CHALMERS
UNIVERSITY OF TECHNOLOGY

Network Slicing

This is a pre-print version of the Network Slicing Chapter - CNIT Technical Report 06 titled "Network Programmability: a (r)evolutionary approach".

The full Report can be purchased (both PDF and printed versions) at https://shop.texmat.it/collana-cnit.html

Network Slicing

Carla Raffaelli[1], Paolo Monti[2], Federico Tonini [2]

[1] DEI - University of Bologna, Bologna - Italy

[2] Chalmers University of Technology, Gothenburg, Sweden

Abstract: *Network slicing is emerging as a key enabling technology to support new service needs, business cases, and the evolution of programmable networking. As an end-to-end concept involving network functions in different domains and administrations, network slicing calls for new standardization efforts, design methodologies, and deployment strategies. This chapter aims at addressing the main aspects of network slicing with relevant challenges and practical solutions.*

1 Introduction

1.1 Background, concepts and motivations

Network slicing is emerging as a key technology for programmable networks thanks to the maturity reached by network virtualization techniques and emerging business opportunities, especially, but not exclusively, in relation to 5G and further generations of cellular networks. Network slicing is targeted to accomodate end-to-end services while maintaining quality of service requirements even in changing network conditions. As a consequence it has triggered the activity of the main standardization bodies, including 3GPP, IETF, ITU-T, supported by many alliances and groups.

The general concept of network slicing is represented by logical correlations of network functions and resources, either virtual or physical, to provide programmable end-to-end services on demand, according to performance requirements. This concept can be seen as an evolution of the Infrastructure as a Service (IaaS) cloud computing model, that has been pushed forward to much more flexible and dynamic paradigm as network slicing, where a thorough adoption of virtualization, potentially for any network function, is applied. As a consequence, network slicing offers extremely high potential in shaping network platform with high flexibility involving different networks, cloud resources, operators and business players, which also translates into a challenging increased complexity in network control, management and orchestration [1].

Some aspects can be identified to characterize network slicing:

- end-to-end: an intrinsic property of network slicing to facilitate service delivery to end users, customers and applications;
- resource sharing: to allow better network resource utilization levels in the presence of many different and differently evolving service needs;

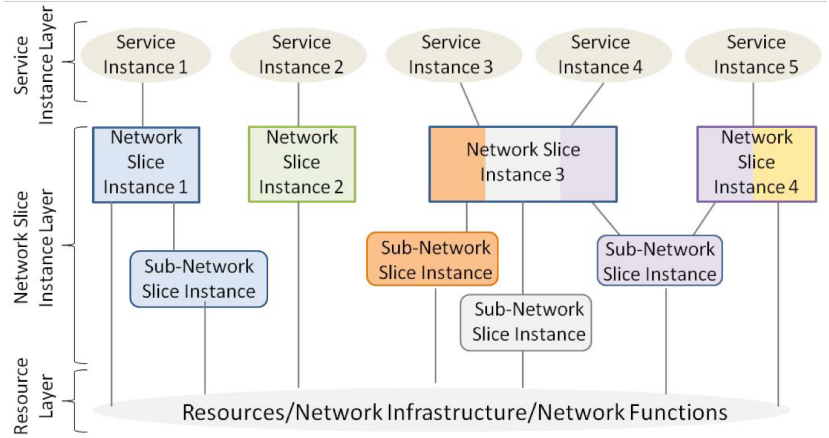

Figure 1: NGMN Network slicing architecture

- isolation: to ensure performance and security while sharing resources;
- elasticity: to ensure Service Level Agreements (SLAs) are met while varying the radio and network condition or geographical service area, as a consequence of user mobility;
- programmability: to allow third parties to control network resources through Application Programming Interfaces (API) to ensure elasticity and customization;
- automation: to enable on demand configuration of a network slice, based on signalling to specify, besides conventional SLAs like latency, jitter, capacity, other timing attributes like its duration or periodicity;
- hierarchical abstraction: a property which allow hierarchical usage of network slices that, once configured can be in their turn further shared by further parties.

A reference architecture for network slicing has been proposed in a Next Generation Mobile Networks (NGMN) document [2] to have a common understanding of the concept and a reference basis for the network slicing process and operation. Network slicing is organized into three layers as shown in figure 1, namely the Service Instance Layer, the Network Slice Instance Layer, and the Resource layer. A service instance (SI) is the implementation of a service as could be provided by a vertical context or application provider or mobile network operator. A network slice instance (NSI) is the set of resources configured to provide the service instance with the level of performance required as the result of a possible composition of sub-network instances, i.e., the Network Slice Subnet Instances (NSSI), which can be isolated or shared among different slices. In this sense the network slice can take advantage of functionalities and resources available in the network, that can be used in support of separated slice implementations [2].

Being the NSI an end-to-end concept, it can involve NSSI belonging to different administrative domains which can be partially shared by other NSIs. In turn, the NSSIs can be isolated or shared depending on the applied policies and configurations. The NSI

itself can be shared among different SIs with compatible requirements. In relation to dynamically changing service demands, NSI can be reconfigured accordingly, by controlling the assigned abstracted relevant resources using open programmable interfaces.

The attention has been recently further pushed towards an end-to-end autonomic framework which consists of embedded self-managing capabilities, thought to be distributed across the entire virtual infrastructure. This autonomic framework will allow dynamic and adaptive adjusting of the infrastructure to system-wide environment changes (e.g. traffic patterns, capacity, coverage, software, new service integration, fault prediction, fault mitigation, fault isolation, security threats, privacy safeguards, energy conservation etc.), while optimizing system-wide behavior, performance, and service experience. The embedded self-managing characteristics of an end-to-end autonomic framework are delineated in terms of self-Configuration, self-healing, self-Optimizing, and self-Protection attributes [3]. These cognitive attributes embedded within cooperating entities, are expected to lead to a zero-touch (no human intervention) automation of operations on a system-wide basis, well-beyond the limits of automation afforded by just a collection of self-managed entities.

1.2 Business opportunities

The interest in network slicing and the consolidation of this concept can be related in a large part to the emerging 5G ecosystem which involves many different industrial sectors and verticals. It can be seen in any case as a new paradigm of network design characterized by high flexibility and openness for the development of products and services at many different level of abstraction and business opportunities.

The business roles that can be facilitated by a thorough deployment of network slicing can be identified in the following players:

- infrastructure providers: provide networking hardware and connectivity in public or private areas which need a dynamic way to share and sell their resources among customers;
- cloud providers: offer computational and storage resources along with potential cloud services to host network slice deployment;
- virtual network operators: lease resources from infrastructure providers and manage the runtime operations to support specialized services in urban or remote areas;
- network service template providers: offer service templates and tuning resources in relation to the volume of traffic and Service Level Agreements (SLAs);
- verticals: offer services to non-telecom communities and it is interested in sharing infrastructure resources by network slicing in adaptive way;
- service brokers: mediate among virtual network operators, verticals, application providers and the infrastructure providers to map their requests on available physical resources.

Interactions and partnerships among different players are needed to facilitate the involved players to properly manage operation and resource usage and can be provided

over networking and cloud resources. Network slicing configuration will progressively turn into an automated process without the need of external manual intervention. This require proper signaling mechanisms to allow third parties to place a slice creation request indicating the desired capacity, latency, jitter, duration or periodicity and changing them beyond conventional SLA contractual agreements [4]. A typical context where network slicing will bring its huge potential is 5G networking where multi-tenancy and enhanced coverage for third parties is provided in a flexible way, thus representing a mean of further revenues for operators, infrastructure and cloud providers.

1.3 Standardization bodies and activities

The NGMN Alliance (Next Generation Mobile Networks Alliance) has taken the initiative to define the network slicing concept in January 2016, as seen by mobile network operators. The goal of the NGMN Alliance is to ensure that the standards for next generation network infrastructure, service platforms and devices will meet the requirements of operators in relation to the end user demand and expectations. NGMN is open to three categories of participants (i.e., NGMN Partners): (*i*) Mobile network operators (Members), (*ii*) telco vendors, software companies and many other leading industry players (Contributors), and (*iii*) research institutes contributing substantially to mid- to long-term innovation (Advisors). Each of these Partner categories has different rights as laid down in the NGMN Articles of Association and the NGMN Participation Guidelines. The aim of network slicing reported in the NGMN document is to design a more flexible and efficient architecture to meet the requirements of emerging 5G use cases which cannot be supported by current architectures.

The 3rd Generation Partnership Project (3GPP) includes telecommunications standard development organizations, known globally as "Organizational Partners" and provides their members with the environment to produce the Reports and Specifications that define 3GPP technologies. 3GPP has finalized the definition of the 5G System architecture, including Access and Core Networks, and it defined the key design principles for End to End Network Slicing. 3GPP considered many aspects of network slicing, such as management, access, charging, security, provisioning, and roles [5],[4]. 3GPP also launched the Service Based Architecture (SBA) for organizing and operating network functions (NFs), evolving from conventional point-to-point interfaces to micro-service interconnected architecture [6].

Standardization activities in the IETF involve the specification of general requirements and the development of 5G network slicing architecture, network slice management and orchestration mechanisms including lifecycle management to coordinate E2E and domain orchestration. It is important to mention that, some of the recent works in the IETF include: applicability of Abstraction and Control of Traffic Engineered Networks (ACTN) to network slicing, gateway function for network slicing, management of precision network slicing, and packet network slicing using segment routing.

The GSMA represents the interests of mobile operators worldwide, with more than 750 operators and almost 400 companies in the broader mobile ecosystem, including handset and device makers, software companies, equipment providers and internet companies, as well as organizations in adjacent industry sectors. GSMA has issued some documents to support the mapping of vertical industry and the use cases to Network slice characteristics. GSMA has defined how to map the use case requirements to network slices

by introducing the Network Slice Template. This concept allows also to describe network slice characteristics to support interoperability between different operators [7]. The Generic Slice Template (GST) defined by GSMA refers to some common slice attributes that the industry can use as the basis to describe the network slice type. These attributes can be used by vendors, mobile network operators and slice customers, in addition to other proprietary attributes for slice customization. GSTs are then filled with values for all or for a subset of the attributes to describe specific slices. A GST filled with values is called the NEtwork Slice Type (NEST), which also serves many purposes. Vendors can use a NEST to define the features of their products. Vertical Industry customers (slice customers) can use a NEST as a reference to understand the contractual agreements with the network operator. Network operators (slice provider) can use a NEST with their roaming partners facilitating the definition of network slices in roaming agreements [8].

Standardization activities from ITU-T SG13 [9] include the development of requirements and a frame-work for network management and orchestration for vertical (service to network resources) and horizontal slicing. The ITU-T SG13 also defines an independent management of each plane (service, control data) and association of a user with multiple type of slices which is very closely coupled with the 3GPP work. It further defines high-level technical characteristics of network softwarization for IMT-2020, and data plane programmability. ITU-T SG15 developed an architecture of Slicing Packet Network (SPN) for 5G transport along with network slicing requirements for a SDN transport network.

2 Slice Design

2.1 Quality of service requirements

One of the main features of network slicing is its ability to customize the capabilities and functionalities of the network to the customers' needs. The main driver for the definition of network slicing customization is nowadays represented by the industrial vision on 5G started by NGMN [2] and further developed by the 3GPP [5].

The relationship between a network slice provider and a slice customer is expressed by the Service Level Agreement (SLA) which defines the terms and conditions of the service, the level of exposure, and the amount of control given to tenants and customer on network slice operation. Network slicing enables programmability and modularity in the provisioning of network resources with respect to the service requirements of a specific vertical segment, thereby presenting high potential in adaptation to customers' needs. Efficient resource orchestration and programmable management are applied to face the needs of the different contexts [1]. In addition, network slices, allocated to verticals, can stretch across greater geographical areas, that is, between different countries, or encompass areas where coverage can only be assured by combining resources from different mobile operators. Such slice deployment requires an efficient combination of federated resources, not only to provide the desired bandwidth, but also to cope with additive (e.g., latency or jitter) and multiplicative constraints (e.g., end-to-end error rate probability) across multiple administrative domains. Fulfilling such requirements across a federated environment is challenging, not only from the perspectives of decomposing a slice request into the respective domain(s), but also for assuring its performance maintenance. [3]

An important property of network slicing is slice isolation. An appropriate level of isolation and QoS provisioning is required to allow a variety of verticals to use a common infrastructure. Each slice may be perceived as isolated from different points of view, such as associated resources, performance or security aspects. Level and strength of isolation may vary depending on the requirements and the usage scenarios for slicing.

3GPP has introduced an orchestration and management architecture in [5] to provide a service management function, which analyzes incoming slice requests, converting service into networking requirements, and a network slice management function, which performs the mapping onto network resources and takes care of the whole Life Cycle Management (LCM) (i.e., from the allocation of network slice resources and their operation, to their final de-allocation). Although the resource mapping process is carried out across different technology domain (i.e., including RAN, transport and core) the current 3GPP efforts concentrate mainly on the NSIs deployed and managed by a single administrative entity.

The key role in the definition of a slice with proper quality of service characterization is taken by the orchestrator which should span across different domains in a hierarchical way [10]. This can happen either for technological reasons, e.g., Radio Access Network (RAN), Transport and core, or for administrative reasons to harmonize resources among different domains. As a consequence, the development of a standard way to exchange information among different providers and domains is needed, so that SLAs are met on the whole span. Also, an appropriate set of data must be selected and sent to the orchestration layer to (*i*) solve the slice admission and mapping problems in an optimal way, and (*ii*) to be able to continuously monitor the SLAs during the slice lifecycle.

To establish a multi-domain slice the principle of recursive virtualization and hierarchical network abstraction is applied [1]. The network resources allocated to a particular tenant can be abstracted and exposed to a third party that can construct a new service on top of the prior one. This approach simplifies the composition of slices allowing the combination of different resources in a flexible way. Upon the arrival of a slice request, the service orchestration layer decides whether to admit the slice or not. This process involves the identification of the domains to be involved. Then, the slice request must be converted into directives for the different domains, each selecting the most appropriate set of resources. This can be done using an intent-based networking paradigm, which allows expressing slice requirements and constraints in the form of policies [11]. Each domain is also responsible for providing monitoring data throughout the slice life-cycle. Data from different domains are collected and elaborated by the service orchestration layer to monitor SLAs and take the necessary actions. The interactions among these entities can be based on a peer to peer approach or a federated infrastructure domain [3]. In the former, orchestrators of different domains interact to find a solution that satisfies specific SLAs. In the latter, a common cross-domain slice coordinator leverages trusted connectivity across administrative domains and carries out domain-specific resource allocation.

2.2 Problem definition and objectives

Each network slice is a logically self-contained network that needs to be designed for a specific requirement and consists of several network functions and resources abstracted from underlying communication and network resources [1].

How to effectively convert the network service requirements to the desired network resources requires several considerations at different levels, including the control level,

data plane level, and network wide level. In addition, one of the most attractive and challenging aspects of network slicing is represented by the elasticity in adjusting resources under varying network conditions to guarantee the required SLA. For example, the number of virtual machines (VMs) with the corresponding computing, storage, and networking resources are hardly determined under time-varying data traffic. Communication network conditions are inherently dynamic: a widely known example is the diurnal fluctuation of network flows that follow human activity. Other phenomena may also lead to time-varying slice requirements: cultural and sports events, service attacks and server downtime, variability of wireless channels, time-varying cost of virtual resource at different locations, failures of optical links, and so on. The primary goal of dynamic network slice is to allow the network operator to reconfigure and migrate the slices in order to match the network variability.

Slice Isolation allows for simpler and more effective design of each slice with the goal of meeting the requirements of the particular vertical applications and services offered by the slice tenant. In addition, network failure, overload, or security attacks in one slice will not affect the operation of other slices in the network. However, slice isolation can have a negative impact on multiplexing efficiency and network utilization. Careful consideration of how the traffic generated by a service will use the instantiated network slice is consequently needed, especially in relation to latency requirements.

End-to-end resource allocation has to cross different technological domains, such as CN, RAN, and transport network (TN) or even different administrative domains. Novel coordination approaches among heterogeneous techniques of different network layers are required and not easy to achieve. Even though the different technology segments of an end-to-end network slice, like the Radio Access Network (RAN) and the core network (CN), have specific requirements, most resource allocation problems of network slicing can be converted to general optimization problems coupled with the related network slice life cycle management [1].

The resource allocation related to network slice design can be solved as a virtual network embedding (VNE) problem [12], [13]. The VNE poses two issues: mapping virtual nodes to physical nodes, and mapping virtual links connecting virtual nodes to paths connecting physical nodes. The physical nodes and paths involve computing, storage, and networking resources. So the network slicing design problem results in a combined optimization problem of placing network functions over a set of candidate locations and deciding their interconnections.

In general terms, the network slice design problem can be formulated as follows:

- **Given**: a network with available resources (e.g radio, transport, cloud resources), the type of service slice to be deployed and its service requirements
- **To find**: most convenient set of resources to be allocated to the slice
- **To ensure**: (*i*) QoS requirements (e.g., latency, bandwidth), (*ii*) available network resources are not exceeded, (*iii*) proper level of resource isolation is ensured.

Its solution is achieved by application of optimization techniques and dynamic network adaptation strategies.

2.3 Network slice optimization and performance

Network slice optimization can be achieved by centralized approaches, such as the facility location problem or the set cover problem, which are unfortunately NP-hard [14]. They provide optimal solutions with a small network slice-sets but do not scale well. As a consequence, distributed approaches based on heuristics are often adopted and demonstrated to be effective and scalable. While typically achieving sub-optimal solutions, distributed approaches better adapt the solutions to network slice evolution and network slice-set sizes [15].

The resource allocation problem of network slicing can in principle be defined as an integer linear programming (ILP) problem or nonlinear programming (NLP) problem. The optimization objectives include the throughput of network slices, the resource utilization ratio, the remaining physical resources for the next assignment, and possibly other parameters. The general constraints are the computational resources available in a transport node, the bandwidth between nodes, and the power consumption, to cite a few. The problem description can become complicated due to varying network environments and diverse requirements of network slices. This kind of optimization problems results difficult to be solved in polynomial time. In [15] solutions are given for a network size of 16 nodes. Heuristic approaches are shown to be able to solve much larger networks to derive near-optimal solutions with low computational complexity. Other approaches include hierarchical resource allocation scheme to maximize the network throughput of all network slices in C-RAN [16].

Monitoring and online optimization can be applied to provide effective dynamic network slicing. A key point is represented by the choice of the parameters to be monitored and how often this happens and when to take actions to adjust them [17],[18]. Current network monitoring tools provide a static view of the utilization of each network resource (link and node capacities, VNF capabilities, etc.). In a dynamic environment, the impact of embedding a particular slice on future slice requests is unclear. For example, consider the case of optimally embedding a slice based on static information. Once a new slice request arrives, a reconfiguration of the old slice might be needed due to resource contention. However, slice reconfiguration comes at a cost and hence a prediction mechanism of future resource utilization can be helpful. Typically, prediction mechanisms rely on the use of historical data. In our context, techniques from machine learning [19] can be used to exploit the raw monitored data from the SDN controllers and obtain useful information for the network orchestrator to make predictions about the impact of new slices on network resource allocation and performance and take suitable actions [20].

The methodology of dynamic NS creates an arena of online optimization problems. To optimize the use of available resources and meet the time-varying slice requirements, the network operator needs to constantly optimize the slice resource allocation, while deciding whether or not to admit new slices. This falls into the area of online optimization, where powerful algorithmic tools exist,such as stochastic network optimization and the domain of online competitive algorithms [21]. The online NS problem is related to other classical online problems such as the online minimum cost multi-commodity problem, the online network embedding problem, the online VNF placement problem the online packing problem, the online facility location problem and different variations of them.

A typical way of solving online problems is by using the offline optimization counterpart in two phases: quick assignment and readjustment [22]. The quick assignment

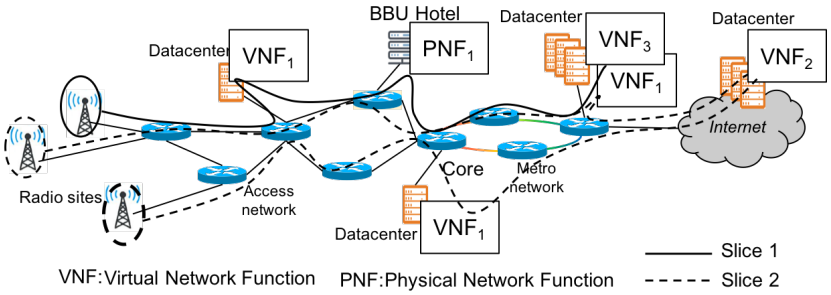

Figure 2: Example of virtual (VNF) and physical (PNF) function chaining to support mobile network slicing

phase exploits quick but sub-optimal algorithms to decide how to embed the slices one by one. Then the reconfiguration algorithms resolve the global offline resource optimization, and the slices are reconfigured into a well performing optimizing configuration. A sample two-phase approach applied to C-RAN is described in [23]. Trade-offs between cost and frequency of reconfigurations have been studied in [22].

3 Slice Deployment

3.1 Main software components

Network slicing is the result of the combination of many different functional elements to support the end-to-end service. Network Function Virtualization (NFV) allows the deployment of network functions, originally performed in hardware, in virtual environment which benefits of cloud computing support. Virtual Network Functions (VNFs) are deployed on Virtual Machines (VMs) that can be suitably chained to enable network services. The hardware virtualization takes place on the host machine, while the guest machine is called the virtual machine (VM). Each virtual machine shares resources like computing, storage, and connectivity. The NFV architectural framework [24] consists of several components, including the VNFs, the NFV infrastructure and the Management and Orchestration responsible for managing and orchestrating VNFs and VNFIs. A single NFV infrastructure (NFVI), such as a data center, may support concurrent instances of multiple network slices, where each network slice can support one or more service verticals, such as mobile broadband, 4k video services, ultra-reliable e-health, and/or autonomous driving applications.

Network function chaining allows to logically relate different functional components to enable the network slice. As a consequence, a network slice results as a set of Virtual Network Functions (VNFs) and Physical Network Functions (PNFs) that are chained in a logical sequence over virtual links (VL) to provide the network service required. A network slice corresponds to one or more function chaining. An example of virtual and physical network function chaining to support a mobile vertical is represented in figure 2. The Service Based Architecture (SBA) defined by 3GPP in releases 15 and 16 offers a set of network functions to implement both the user plane and the control/management

plane o 5G networks.

In the Software Defined Networking (SDN) framework the control plane is fully separated from the data plane, which is moved to a centralized location implemented by SDN controllers. The SDN controller manages network slices effectively by applying rules in accordance with the corresponding network policy. Based on the requirements of the application at hand, the SDN controllers generate different rules in terms of link discovery, topology management, policy deployment, and flow table delivery and send them to the data plane. The forwarding devices in the data plane, such as switches and routers, just apply and execute these rules.

To efficiently and flexibly utilize virtual resources and manage VNFs, the NFV management and orchestrator (NFV-MANO) is proposed by ETSI, which consists of NFV orchestration (NFVO), VNF managers (VNFMs), and virtualized infrastructure managers (VIMs). NFV-MANO manages the lifecycle of VNFs through VNFMs and VIMs. NFVO is responsible for orchestrating a network service incorporated with an external operation/business support system (BSS/OSS). The NFVO allows a network operator to provide a network service by chaining a number of VNFs.

From the implementation point of view, the concept of virtualization requires an additional layer responsible for creating, controlling, and managing virtual machines, called hypervisor. The hypervisor is typically located between the physical infrastructure and the operating system and implement a platform to allow the sharing of the hardware resources [1]. As an alternative to hypervisor-based virtual machines, containers can be adopted to implement an operating system based virtualization and create multiple user space isolated server instances [25].

3.2 End-to-end network slicing

Network slices, allocated to verticals, can stretch across different countries, administrative domains, or encompass areas where coverage can only be assured by combining resources from different mobile operators. Likewise, vertical services may need computing and storage resources that can only be offered by specific cloud providers to complement connnectivity capabilities. Such slice deployment requires an efficient combination of federated resources, not only to provide the desired bandwidth, but also to cope with additive (e.g., latency or jitter) and multiplicative constraints (e.g., end-to-end error rate probability) across multiple administrative domains. The slice request needs to be decomposed into each domain and its performance to be maintained throughout the entire service life-cycle. To handle the dynamics related to federated resource allocation efficiently, a cross-domain coordinator has been introduced [3]. Such a coordinator aligns cloud and networking resources across federated domains and controls inter-domain transport layer connectivity assuring the desired performance.

A possible architecture of the network slicing system for end-to-end network slicing is proposed in [26] based on the SDN principles of full separation of data and control planes, centralized control logic, abstracted view of resources, and states exported to applications. Figure 3 represents the management and orchestration (MANO) architecture with reference to 5G mobile networks. The data plane is represented by the infrastructure layer, which includes mobile edge, mobile transport and core. It is a logical composition of links, forwarding nodes (switches and routers), cloud nodes (data centers) including also connectivity, computing and storage resources. The control plane is divided into

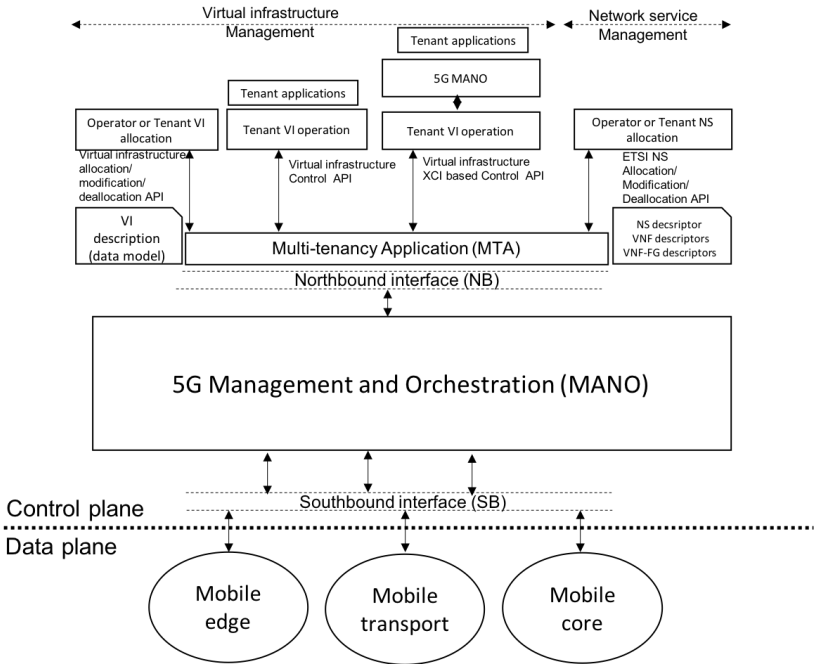

Figure 3: MANO representation for 5G deployment

two sublayers: an application layer above and a management and orchestration layer (MANO) platform below. The multi-tenancy application (MTA) has the specific aim of implementing multi-tenant support throughout domains [26].

MANO provides an abstracted view of available resources and states to the application ecosystem via a northbound interface (NBI). On the other side MANO is connected to the data plane elements via a southbound interface (SBI) to execute control and management functions on the hardware components. To offer end-to-end management through different network domains the MANO system can adopt a hierarchical structure, where a global orchestration engine controls the MANO of each domain, or a peer-to-peer structure, where the MANOs of different domains interact each other. Many open source projects have developed the MANO functionalities, like open Daylight and ONOS for the SDN controllers, and OPNFV, ONAP, OSM, Openstack and OpenMANO as far as the ETSI NFV MANO [26].

3.3 Life cycle management

The deployment of network slicing involves the management of the whole network slice life-cycle. With reference to a virtual infrastructure deployment this involves dynamic allocation of resources, their operation and their de-allocation. Partitioning and book-keeping of resources and instantiation of virtual functions is performed with bandwidth and latency characteristics chosen according to the virtual infrastructure requirements. The allocation of a virtual infrastructure can be triggered by a tenant, like a Mobile

Virtual Network Operator, using an API or contacting the infrastructure operator to agree on the SLA. The partitioning of resources required by the virtual infrastructure can happen in hard or soft quotas, depending whether they are fully assigned at the time of instantiation or at the time of use. A common partitioning approach allows instantiated software node to share the physical ports and rely on the statistical multiplexing enabled by packet switching.

Once the virtual infrastructure has been allocated, the multi-tenant application assign tenants with some level of control over it. Each tenant is allowed to deploy their choice of infrastructure operating system and control plane to optimize resource usage in relation to their own application. In general the tenant has limited control over the abstracted elements of the infrastructure which includes virtual infrastructure view and resource state. Actual configuration and monitoring of flows at the nodes are typically excluded.

The deployment of a network service is complementary to virtual infrastructure deployment and aims at delivering isolated chains of virtual services composed of Virtual network functions (VNF) by sharing the same physical infrastructure, which provides computing, storage, and connectivity resources. A tenant request specify the type of VNF in the NS descriptor and how they need to be connected. For this purpose templates are provided and standardized within ETSI NFV ISG and OASIS TOSCA [26].

When deploying a Network Service the tenant is only interested in operating the application and is not involved in any resource configuration effort. Application-level interfaces are provided to tenants which follows an intent-based approach asking for the composition of some network functions using the available API. The Multi-tenancy Application (MTA) is responsible for the logical mapping and maintenance of tenant's requests on the underlying virtual resources, according to the SLA established.

The life-cycle of a network slice is described by 3GPP as composed by the four phases described in the following. [5].

- Preparation. During the preparation phase the network slice instance does not exist yet. During this phase the network slice template design, the capacity planning are performed in relation to the network slice requirements preparing the network environment to host the NSI.

- Instantiation, configuration and activation phase: during instantiation and configuration all resources needed to support NSI are created and configured so that the NSI is ready for operation; the activation step includes all the actions that makes the NSI active.

- Run time phase: during this phase the NSI is able of support communications services. Supervision and reporting is provided in this phase as well as changes related to NSI scaling, NSI capacity, NSI topology or association with network functions.

- Decommissioning: the NSI is terminated and its resources removed. The NSI does not exist anymore after this phase in the context of network slice lifecycle.

In the process of network slice definition use cases and requirements of the slice customers are expected to be represented in a common language using the Generic Slice Template (GST) and the NEtwork Slice Type (NEST) [8]. The GST is a set of attributes

(e.g. supported throughput, supported functionality, provided application programming interfaces (APIs), etc.) that characterise any slice. It contains the attribute names, definitions and units. The NEST is a GST filled with values and/or ranges based on specific vertical industry use cases. A NEST is essential for a network operator to instantiate equivalent slices, e.g. in terms of performance, functions, etc. Some early trials have been conducted to demonstrate network slicing with cross-industry collaborations among operators, vendors and vertical industries [26].

4 Challenges in Slice Design and Operation

4.1 Slice automation

In terms of challenges, end-to-end management and orchestration frameworks require the implementation of specific functionalities to reach complete automation. Slice deployment should be autonomous, requiring automatic acceptance or denial of slice requests, based on the network resources and the service requirements. The network control should also be able to continuously monitor the state of the resources to adapt slice mapping and operation to the evolving network conditions, i.e., to be able to re-configure itself. This is required, for example in the case of traffic variation and/failures. All this must work when slices traverse different technological and/or administrative domains, requiring to elaborate and expose information among the different entities in a common, standard way [17].

Slice automation concerns the following aspects:

- traffic prediction
- resource management
- admission control
- slice embedding/reconfiguration in the virtual or physical infrastructure

just to mention a few.

Artificial intelligence (AI) can help by enabling systems to autonomously take decisions based on their perceived environment. To do so, information must be collected from the network, where equipment of different suppliers co-exists, requiring the definition of common standard interfaces to create vendor-agnostic monitoring systems [27]. Moreover, telemetry information can be exploited for proactive or reactive network re-configurations. Different models can be used to obtain information about the physical layer and trigger changes at the network level, e.g., in routing, spectrum and modulation assignments [28]. These data can also be used to estimate the traffic and take appropriate actions [29], i.e., triggering reconfiguration strategies to change the current slice resource assignment and avoid SLA violation or slice request rejection. Even though these approaches are effective, further analysis of their computational effort and performance, as well as the amount of data to be collected and elaborated in real-size scenarios require further studies.

Strategies based on ML can also be employed in the slice admission process. Reinforcement learning (RL) strategies can be used to make scheduling decisions based on the

feedback coming from past actions [30]. Application of supervised learning methods can help in predicting traffic evolution and get insights into future resource needs [31].

The key point in the application of AI-based techniques is represented by the identification of a suitable training procedure, depending on the context and on the AI model applied.

4.2 Attacks, security, vulnerability

Network slicing introduces new vulnerabilities with respect to the already investigated critical security threats attributed to the underlying SDN and NFV technologies [32].

Secure network slicing solutions must ensure the main security principles, traditionally categorized into confidentiality, authentication, authorization, availability, and integrity. In such a context, when associated with network slicing, all these concepts need to be reviewed in relation to the sharing of resources and the new needs of interaction between the control and the data planes [33].

Multiple slices have to co-exist on the same shared infrastructure. The main property of isolation ensure reliability and security for each slice [33]. Slice isolation techniques prevent different services to potentially depleting slice resources, or to exhaust common resources with multiple slices, causing Denial of Service (DoS) to other subscribers. Distributed DoS attacks may also be caused by malware on user's devices, and since they may be connected to different slices simultaneously, this could lead to unwanted inter-slice communication [33]. Isolation is also required to avoid resources assigned to a slice to be accessed by other slices, especially for privacy reasons (e.g., personal data stored in a data center). For example, if a network function (NF) is shared, a violation of the NF may allow attackers to steal information from different slices. Isolation of NFs can be done, e.g., at the hardware level, the virtual machine, or kernel level. Complete isolation NF is preferable from a security point of view. However, this usually leads to different dedicated networks with very low multiplexing gains.

As long as KPIs are met, slice isolation is preserved. Interference among network slices sharing the same resources can arise when KPIs are no longer met. Interference can be broadly defined as anything that breaks the possibility to provide the KPIs of a given slice, and in particular security-related KPIs. As a consequence possible attacks can happen and the primary challenge in network slice security design is to ensure that attacks performed against one slice do not affect the others, that means that security functions act independently on each slice.

The multiple kinds of interactions in the network slicing architecture, which take also place through different administration or technological domains, increase the security challenges. Some of the security aspects have been already solved while there are some nontrivial issues, such as avoiding to compromise a NF, defending against side-channels, or dealing with end-devices vulnerabilities that still lack solutions and need further research [33].

4.3 Differentiated Reliability

The resource allocation algorithms of network slicing should not only enhance the resource utilization efficiency, but also handle unpredictable network events to achieve high availability of the network slice instance. Unpredictable network events include network

congestion (i.e., caused by heavy data traffic) or network function failures (i.e., caused by unexpected malfunction of software or hardware). Redundant resource reservation and network function remapping are two efficient approaches to cope with unpredictable events.

Depending on the quality of service requirements, different reliability models can be applied. A differentiated reliability approach can be adopted with protection models specific for each class of service. Dedicated protection can be applied to delay sensitive slicing, such as the URLLC slicing, while shared protection can instead be applied to high bandwidth demanding slicing, such as the eMBB slicing. This differentiated approach allow to statically reserve redundant resources only for those services that need prompt service continuity, while trying to reduce the redundant resources required by high capacity services by sharing.

Two different ILP models are presented in [34] comparing the outcome of dedicated and shared backup path protection (DPP and SPP, respectively) in the transport network. Both strategies allows to select the most convenient baseband split depending on the available network resources while providing reliability against single node or link failure. The objective of these strategies is to minimize the number of nodes where to install cloud resources and the amount of resources to be provisioned. Numerical results are obtained considering the deployment of a URLLC slice in a 6 node network, reported in Fig. 4, under different conditions. In particular, two different resource distributions are considered. In the balanced case, which well represent a mobile edge computing scenario with nodes of limited computational capacity, all the nodes have the same capacity. In the unbalanced case instead, nodes 2 and 5 are assumed to have unlimited resources, while other nodes have limited capacity. All the links have the same and limited capacity. The details of numerical setup are described in [34].

Table 1 shows the number of nodes selected to host either baseband, core or cloud functions (referred to as active nodes) and the saved backup computational capacity when SPP is used, with respect to the DPP, in the sample network. The unconstrained case, reported as a benchmark, provides a lower bound for the number of active nodes, that is the case when no constraints on capacity, bandwidth, and latency are applied. Since this case requires only 2 nodes, it also exhibits no backup resource sharing. In real case scenarios, when resources are limited, the number of nodes increases due to finite node and link capacity. This situation is evaluated under two different delay constraints (2 and 3 hops). In the balanced case, sharing backup resources leads to a reduction in the number of nodes to be activated, regardless of the number of hops. This is due to the sharing of backup resources in both links and nodes, that allows reducing the backup capacity by 66.6%. In the unbalanced case instead, where some nodes provide extensive capacity, the SPP is still effective in sharing backup capacity with a reduction of up to 27.8%.

5 Conclusions

Network slicing has been described in many aspects to show its potential in achieving high flexibility, efficiency and reliability as required by emerging network services. To summarize, slicing strongly relies on network function virtualization and SDN control and management to offer a programmable environment and enable network automation.

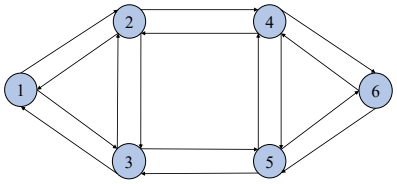

Figure 4: 6 node networks for URLLC slice deployment. Resulting active nodes are shown in table 1 with DPP and SPP.

Table 1: Active nodes and capacity savings for the 6 node network

Network	Active nodes DPP	Active nodes SPP	Saved Capacity
2 hops - bal	6	4	66.6%
3 hops - bal	6	4	66.6%
2 hops - unbal	4	4	23.6%
3 hops - unbal	4	4	27.8%
Unconstrained	2	2	0%

Many companion techniques have been referred such as optimization algorithms and machine learning approaches that need to be suitably tuned for the purpose of network slice design monitoring and dynamic configuration. Security, automation, reliability still are mentioned as possible areas of further research.

References

[1] I. Afolabi et al. Network Slicing and Softwarization: A Survey on Principles, Enabling Technologies and Solutions. *IEEE Commun. Surveys Tutorials*, 20(3):2429–2453, 2018.

[2] NGMN Alliance. Description of Network Slicing Concept, NGMN 5G P1 Requirements Architecture, Work Stream End-to-End Architecture, Version 1.0. 2016.

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

[4] 3GPP. Management, Orchestration and Charging for 5G networks (Network Slicing on the way). 2019.

[5] 3GPP. TS 28.530, Management of 5G networks and network slicing; concepts, use cases and requirements, Rel.15. 2018.

[6] 3GPP. TS 23.501/TS 23.502, Procedures for the 5G system.

[7] GSMA Alliance. Network Slicing: Use Case Requirements. 2018.

[8] GSMA Alliance. From Vertical Industry Requirements to Network Slice Characteristics. 2018.

[9] ITU-T. Framework for the support of network slicing in the IMT-2020 network, Recommendation ITU-T Y.3112. 2018.

[10] F. Tonini et al. Network Slicing Automation: Challenges and Benefits. *Proceedings of Optical Network Design and Modelling 2020 (ONDM 2020)*, 2020.

[11] T. Subramanya, R. Riggio, and T. Rasheed. Intent-based mobile backhauling for 5g networks. *Proceedings of 12th International Conference on Network and Service Management (CNSM)*, pages 348–352, 2016.

[12] A. Fischer et al. Virtual Network Embedding: A Survey. *IEEE Commun. Surveys Tutorials*, 15(4):1888–1906, 2013.

[13] A. Marotta, D. Cassioli, M. Tornatore, Y. Hirota, Y. Awaji, and B. Mukherjee. Reliable slicing with isolation in optical metro-aggregation networks. *Optical Fiber Communications Conference and Exhibition (OFC)*, pages 1–3, 2020.

[14] M. Pioro and D. Medhi. *Routing, Flow, and Capacity Design in Communication and Computer Networks*. Elsevier Science, 2014.

[15] C. Raffaelli, B. M. Khorsandi, and F. Tonini. Distributed Location Algorithms for Flexible BBU Hotel Placement in C-RAN. *Proceedings of ICTON 2018*, 2018.

[16] M. R. Rahman and R. Boutaba. SVNE: Survivable Virtual Network Embedding Algorithms for Network Virtualization. *IEEE Trans. Network and Service Management*, 10(2):105–118, 2013.

[17] A. A. Barakabitze, A. Ahmad, R. Mijumbi, and A. Hines. 5G network slicing using SDN and NFV: A survey of taxonomy, architectures and future challenges. *Computer Networks*, 167, 2020.

[18] A. Clemm, M. F. Zhani, and R. Boutaba. Network Management 2030: Operations and Control of Network 2030 Services. *J Netw Syst Manage*, 2012.

[19] F. Musumeci et al. An Overview on Application of Machine Learning Techniques in Optical Networks. *IEEE Communications Surveys Tutorials*, 21(2):1383–1408, 2019.

[20] L. Nie et al. Traffic matrix prediction and estimation based on deep learning in large-scale ip backbone networks. *Elsevier J. Network and Computer Applications*, 76(C):16–22, 2016.

[21] G. Even, M. Medina, and B. Patt-Shamir. On-Line Path Computation and Function Placement in SDNs. *Lecture Notes in Computer Science*, 10083, 2016.

[22] S. Paris et al. Controlling Flow Reconfigurations in SDN. *Proceedings of 35th Annual IEEE INFOCOM*, 2016.

[23] F. Tonini, B.M. Khorsandi, E.Amato, and C. Raffaelli. Scalable edge computing deployment for reliable service provisioning in vehicular networks. *J. Sens. Actuator Netw*, 24(4), 2019.

[24] ETSI. Network Functions Virtualisation (NFV); Architectural Framework, document GS NFV 002. *Internet Technology Letters*, 2013.

[25] M. G. Xavier et al. Performance Evaluation of container-based virtualization for high performance computing environments. *Proceedings of IEEE PDP*, pages 233–240, 2013.

[26] X. Costa-Perez et al. *Network Slicing for 5G Networks, chapter 9.* John Wiley and Sons, 2018.

[27] D. M. Gutierrez-Estevez et al. Artificial intelligence for elastic management and orchestration of 5G networks. *IEEE Wireless Communications*, 26(5):134–141, 2019.

[28] F. Musumeci et al. An overview on application of machine learning techniques in optical networks. *IEEE Communications Surveys Tutorials*, 21(2):1383–1408, 2019.

[29] D. Rafique and L. Velasco. Machine learning for network automation: overview, architecture, and applications. *IEEE/OSA Journal of Optical Communications and Networking*, 10(10):D126–D143, 2018.

[30] M. R. Raza, C. Natalino, P. Öhlen, L. Wosinska, and P. Monti. Reinforcement learning for slicing in a 5G Flexible RAN. *Journal of Lightwave Technology*, 37(20):5161–5169, 2019.

[31] M. R. Raza, A. Rostami, L. Wosinska, and P. Monti. A slice admission policy based on big data analytics for multi-tenant 5G networks. *Journal of Lightwave Technology*, 37(7):1690–1697, 2019.

[32] T. Taleb S. Lal and A. Dutta. NFV: Security threats and best practices. *IEEE Commun. Mag.*, 55(8):211–217, 2017.

[33] V. A. Cunha et al. Network slicing security: Challenges and directions. *Internet Technology Letters*, 2(5):e125, 2019.

[34] F. Tonini, E. Amato, and C. Raffaelli. Optimization of optical aggregation network for 5G URLLC service. *Proceedings of the 2019 IEEE Global Communications Conference (GLOBECOM)*, pages 1–6, 2019.