

# On-Demand Network Slicing using SDN/NFV-enabled Satellite Ground Segment Systems

Toufik Ahmed<sup>1</sup>, Abdelhamid Alleg<sup>1</sup>, Ramon Ferrus<sup>2</sup>, Roberto Riggio<sup>3</sup>

<sup>1</sup>CNRS-LaBRI (UMR5800), Univ. Bordeaux / Bordeaux INP, France.

<sup>2</sup>Universitat Politècnica de Catalunya, Spain.

<sup>3</sup>Future Networks (FuN), FBK CREATE-NET, Trento, Italy.

tad@labri.fr, aalleg@labri.fr, ferrus@tsc.upc.edu, rriggio@fbk.eu

**Abstract**—This paper proposes an architecture framework for the realization of on-demand satellite network slicing that is built on the introduction of Software Defined Networking (SDN) and Network Function Virtualization (NFV) technologies. In this way, service delivery with satellite networks is shifted from a network for connectivity model to a network for service model with a high degree of service customization and adaptability, including satellite bandwidth on-demand. Under this framework, we study the resource orchestration of satellite network services by formulating the on-demand network slicing as an optimization problem that provides flexible service chaining and provisioning taking into account diversified service requirements. The objective is to determine the optimal resource allocation for supporting a satellite network slice that minimizes resources consumption while meeting service specification requirements such as the end-to-end delay.

**Keywords:** Network slicing, service chaining, Software Defined Networking (SDN), Network Function Virtualization (NFV).

## I. INTRODUCTION

Key features of satellite communications such as wide-scale coverage, broadcast/multicast support and high availability, together with significant amounts of new satellite capacity coming online, anticipate new opportunities for satellite communications services as an integral part within upcoming 5G systems. To materialize these opportunities, satellite communications services have to be provisioned and operated in a more flexible, agile and cost-effective manner than it is done today. In this context, it is anticipated that satellite networks shall embrace network slicing support, which is one of the foundations introduced in 5G as a network architecture evolution to support diversified services requirement (e.g. broadband communication, mission critical communications, massive IoT) over a common network infrastructure.

Two central enabling technologies for network slicing support are Software Defined Networking (SDN) and Network Function Virtualization (NFV), which are also being established as key technologies in the evolution of satellite ground segment systems and its integration in 5G network [1]-[3]. SDN and NFV technologies are expected to bring greater flexibility to Satellite Network Operators (SNOs), reducing both operational and capital expenses in deploying and managing SDN/NFV-compatible networking equipment as well as facilitating the integration and operation of combined satellite and terrestrial networks [4]-[6]. The adoption of SDN/NFV technologies facilitate the deployment of several network slices so that each network slice is delivered as a sequence of Virtual Network Function (VNF) instances, chained together to compose a Service Function that requires a particular amount of resources to provide specific performances in terms of latency, throughput, capacity, and availability. For example, deployment of mission critical services such as public safety over a network slice imposes capabilities related to

always-available coverage, low-latency, and high availability/reliability one-to-many and many-to-many communications. This can be properly achieved by ensuring that network resources allocated to a slice are well provisioned and deployed with specific quality of service (QoS) policy support. Furthermore, the capacity and traffic within the slice considering the specific requirements (e.g. coverage, capacity, mobility, reliability...) are correctly managed and optimized.

This paper provides the design of an innovative architecture framework for on-demand satellite network slicing built on top of SDN/NFV-enabled satellite ground segment systems. A focus will be on modeling the on-demand network slicing as an optimization problem that distributes network resources on-the-fly and on demand using flexible service chaining and provisioning while taking into account diversified service requirements. This allows improving flexibility in terms of scaling up/down network resources and configurability in terms of resource control programmability and dynamic QoS policy to achieve required levels of performance.

The rest of this paper is organized as follows. Section II introduces the proposed architecture framework for SDN/NFV-enabled satellite systems offering on-demand adaptive network slicing. Section III presents the slicing problem. Section IV formulates the on-demand resource allocation model for satellite slicing (OnDReAMS) as an optimization problem for flexible placement and chaining of VNFs. On this basis, section V evaluates the performance of the proposed mechanism. Finally, conclusions are drawn in Section VI.

## II. ARCHITECTURE FRAMEWORK FOR ON-DEMAND SATELLITE NETWORK SLICING

A satellite network slice is conceived as a logical, virtual and self-contained network built on top of a physical satellite network infrastructure. Multiple slices may coexist over the same physical satellite network infrastructure. A network slice aggregates multiple physical network resources and uses specific abstraction and isolation mechanisms at topology, node and link levels to achieve the required levels of performance. More specifically, it is considered that most of the functions of a satellite network slice can be supplied as VNFs running in a distributed Network Functions Virtualization Infrastructure (NFVI). Conversely, the non-virtualized functions of the slice, denoted as Physical Network Functions (PNF) in ETSI ISG NFV terms, are provided through one or several physical hardware appliances, which could be dedicated to a given slice or shared among several ones. Therefore, each network slice is represented by a sequence of VNF and PNF instances, chained together to compose a Service Function Chain (SFC) which lasts for a specific period. The VNF resources can be scaled up or down and they may include a variety of network functions such as Performance Enhancement Proxy (PEP) for TCP acceleration, Firewall, Deep Packet Inspection (DPI), Virtual

Private Network (VPN), Packet-based QoS, DNS cache, and so on. The placement, management, chaining, and orchestration operations of these VNFs should be carefully considered to meet the required performances for supporting diverse services.

In particular, as illustrated in Fig. 1 the following entities is considered as a building block of a satellite network slice:

- One or several Satellite Network Function (SNF) VNFs, namely SNF-VNFs (PEP, TCP acceleration, etc.) and Satellite Baseband Gateway (SBG) VNFs, namely SBG-VNFs for baseband mechanisms, that run over one or several NFVI Point-of-Presences in the network side. In addition, the non-virtualized part of the SBG functions, namely SBG-PNFs such as frequency block resources, together with SNF-VNFs and SBG-VNFs constitutes data plane functions.
- SDN-based control applications and SDN controllers (all running as VNF instances) for the realization of some control functions (e.g., QoS control, radio resource management [RRM], gateway diversity [GWD], Fading Mitigation Techniques [FMT], etc.).
- Network Management (NM) and Element Management (EM) functions, also running as VNFs, which provide a package of management functions (e.g. Fault, Configuration, Accounting, Performance and Security [FCAPS] management).
- One or several Customer Premise Equipment (CPE) VNFs namely CPE-VNFs that run over Lightweight NFVI-PoP such as service provider Whitebox.

As shown in Fig. 1, the network service orchestration capabilities are logically centralized in the so-called Service Orchestrator (SO) management component, which forms part of the Operation Support Systems / Business Support Systems (OSS/BSS) of a Satellite Network Operator (SNO). Beyond this, functionalities related to the instantiation, modification and termination of the VNFs composing the satellite network slice are covered by the NFV Manager. The functionalities provided by the SO and the NFV Manager are related to the following:

- Lifecycle management of the slice, which can be defined as the set of functions required to manage the instantiation, maintenance (e.g. adaptive scaling up / down, QoS configuration, etc.), and termination.
- Composition of the service function chain described by a network service descriptor (NSD) that represents the part of the slice that is implemented as VNFs and executed over NFVI-PoP(s).
- Determination of the application-specific aspects of both VNFs and PNFs that form part of a slice.
- Fault, Configuration, Accounting, Performance, Security (FCAPS) management of the slice and its components.
- Lifecycle management of the service chain composing the slice through interaction with the NFV Manager
- Management of VNF packages that can be already on-boarded on the NFV Manager or can be managed/on-boarded onto it by the dashboard of the SO.

The operation of the SO and NFV Manager relies on a set of descriptors that are needed for the characterization of a slice and its components. In general terms, a Satellite Network Slice Descriptor (SNSD) is the input provided to the SO that describes the characteristics of the slice as requested by the customer/tenant. Based on the SNSD, the SO composes the NSD, which describes the virtualized part of the slice, and the slice application-specific descriptors, which contain the configuration of both VNFs and PNFs within the satellite network slice.

### III. SATELLITE NETWORK SLICING MODEL

In this section, we formulate the proposed satellite network slicing model as an optimization problem and we define different notations, parameters and terminologies relative to network slicing topic.

#### A. Network Slicing

The satellite network slice is considered as virtual, self-contained and isolated network built on top of aggregated distributed physical resources at core, edge, access and user levels. It is a network of capabilities rather than a network of entities aiming to provide specialized functions deployed at different points to support diverse services requirements. It is represented by a sequence of VNFs instances and PNFs resources, chained together to compose a SFC.

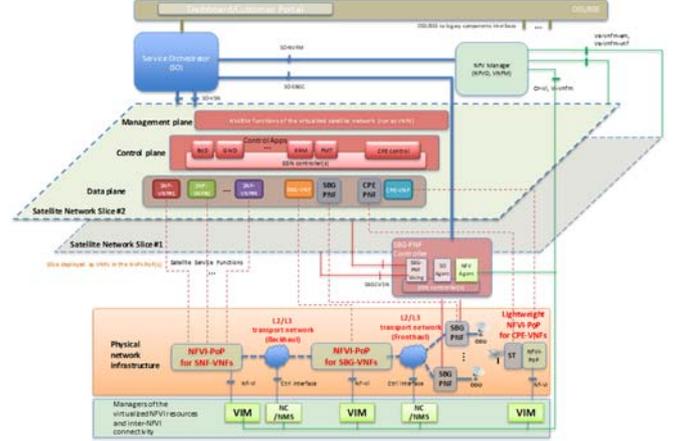


Fig. 1. An architecture for SDN/NFV-enabled satellite ground segment systems.

For simplicity, we ignore the PNF resources at both SBG and CPE as an SBG-VNF is always attached to an SBG-PNF and a CPE-PNF is always attached to a CPE-VNF. However, the proposed model provides appropriate requirements to support normal satellite gateway operation when the functional splitting between VNF and PNF parts is performed. The link between SBG-VNF and SBG-PNF called fronthaul link is carefully examined as the introduction of latency over the fronthaul link can affect the normal network operation such as synchronization, handover decision, guard time, etc. An SFC defines an ordered (resp. partially ordered) set of virtual network functions VNFs [8] that require tailored resources to guarantee predictable network performances defined by a traffic class associated to a slice. We define a Slice Request (SR) by the following parameters: SFC description ( $\rho$ ), Slice lifetime ( $\tau$ ), Tenant identifier ( $\nabla$ ), and Traffic Class ( $\lambda$ ).

We note  $R_s$  the set of slice requests that needs to be instantiated on the satellite network infrastructure. Each slice request  $\delta \in R_s$  is modeled as a quadruple  $\delta (\rho, \tau, \nabla, \lambda)$  where  $\rho$  is the SFC that corresponds to the deployed service,  $\tau$  corresponds to slice lifetime,  $\nabla$  is the tenant identifier and  $\lambda$  is the class of traffic to which belongs the slice request  $\delta$ . The on-demand network slicing is managed as a new request for scaling up / down the SFC network resource. Each SFC  $\rho$  is modeled as a subgraph  $G_v^\rho (N_v^\rho, E_v^\rho)$  where  $N_v^\rho \subseteq N_v$  is a set of VNFs and  $E_v^\rho \subseteq E_v$  is a set of directed edges called virtual links connecting these VNFs. In addition, each VNF instance  $n' \in N_v^\rho$  has its own requested amount of resource denoted  $\theta_{n'}^{n'}$ . Also, each virtual link  $(k, l) \in E_v^\rho$  connecting two VNFs  $k, l \in N_v^\rho$  is characterized by key performance metrics (capacity, performance, delay, etc.) denoted  $\psi_\delta^{(k,l)}$ .

In this work, we concentrate on the slice end-to-end delay (or end-to-end latency) as the key performance indicator of a specific traffic class and we define  $D_{th}^\lambda$ , the end-to-end delay threshold associated to each traffic class  $\lambda \in R_c$ . The value of  $D_{th}^\lambda$  is expected to meet specific requirements for diverse services that will be running on a slice (see Table I. ). We define the end-to-end delay provided by a deployed SFC as the sum of processing delay  $D_{Proc}^n$  of its component VNFs instances and the time needed to forward the flow between these VNFs.

Table I. SERVICE REQUIREMENTS

Type of Service / traffic class	Requirements
Satellite Multimedia Broadcast Multicast Services (MBMS)	High capacity, large packet size, low loss rate, caching at the edge, bulk data, one and many-to-many communications,
Satellite IoT and Massive Machine Type Communications (MTC)	Large converge, one-to-one and one-to-many communication
Satellite Mission Critical Communications (MCC)	Low-latency, high reliability, real-time, jitter sensitive and high interaction
Satellite Mobile Backhauling (MB)	High bandwidth, low latency
Satellite Mobile Direct Access (MDA)	Mobility support, low latency, high reliability, large converge,

## B. NFVI Model

The infrastructure layer hosts the physical and virtual resources needed to create the satellite network slices. These include both virtualization software and hardware comprised of memory, compute, storage, and networking resources. Following the terminology presented in [9], the distributed NFVI is modelled as a graph  $G_i(N_i, E_i)$ , where  $N_i$  is the set of Point of Presence (PoPs) that compose network and  $E_i$  is the set of bidirectional links (PLs). Each PoP  $n \in N_i$  represents a possible location that can host a single or multiple VNFs instances depending on their resource capacities. PoPs are connected via Physical Links (PLs) that forward traffic between VNFs composing a SFC

Each PoP  $n \in N_p$  represents the quantity of available resources in terms of Computing, Memory and Storage denoted  $\theta_n^\lambda$  and reserved to SR using traffic class  $\lambda$ . Similarly, each PL  $(n, m) \in E_p$  connecting two PoPs  $n, m \in N_p$  has its capacity (Bandwidth, Bitrate, etc.) denoted  $\Psi_{(n,m)}^\lambda$  used exclusively by SR with traffic class  $\lambda$ . Table II summarizes the NFVI and SR notation and parameters used in our model.

Table II. NFVI AND SR NOTATION

PAR.	DESCRIPTION
<b>NFVI</b>	
$G_i$	NFVI graph
$N_i$	Set of PoPs in $G_i$
$E_i$	Set of physical links between PoPs
$\theta_n^\lambda$	Available resource at PoP $n \in N_i$ reserved for traffic class $\lambda$
$\Psi_{(n,m)}^\lambda$	Available capacity of physical link $(n, m) \in E_i$ reserved for traffic class $\lambda$
$D_{Trans}^{(n,m)}$	Transmission delay of the physical link in terms of latency $(n, m) \in E_i$
<b>Slice Request "SR"</b>	
$R_s$	Set of slice requests
$R_c$	Set of traffic classes
$\rho$	SFC that corresponds to the deployed service
$\tau$	Slice lifetime
$\nabla$	Tenant identifier
$\lambda$	Class of traffic to which belongs the slice request $\delta$ .
$G_v$	SFCs graph
$N_v$	Set of VNFs in $G_v$
$E_v$	Set of virtual links between VNFs in $G_v$
$N_v^\rho$	Set of VNFs composing the request $r$ where $N_v^\rho \subseteq N_v$
$E_v^\rho$	Set of links between VNFs $\in N_v^\rho$ such as $E_v^\rho \subseteq E_v$
$\psi_\delta^{(k,l)}$	Required capacity of virtual link $(k, l) \in E_v^\rho$
$\theta_{n'}^{\rho}$	Requested resources of VNF $n' \in N_v^\rho$
$D_{Proc}^{n'}$	Processing delay generated by VNF $n' \in N_v^\rho$ using exactly the required amount of resources $\theta_{n'}^{\rho}$
$D_{th}^\lambda$	End-to-end delay threshold associated to $\rho \subseteq R_{sfc}$

## IV. ON DEMAND RESOURCE ALLOCATION MODEL FOR SATELLITE SLICING (ONDREAMS)

Our proposal solution is based on a mathematical program combined with an online algorithm. First, we model the slicing problem using a Mixed Integer Linear Program (MILP). The optimization objective of our MILP is to minimize the amount of allocated resource to VNFs (Equation 1). This objective could be easily adapted to aim other purposes such as number of active PoPs or cost utilization, etc. The optimization objective and the constraints of the MILP are presented below.

$$\text{Min} \left( \sum_{\delta \in R_s} \sum_{n \in N_i} \sum_{n' \in N_v^\rho} \theta_{n'}^{\rho} \cdot C_\lambda^\delta \cdot B_n^{\rho} \right) \quad \forall \lambda \in R_c \quad (1)$$

Subject to:

$$\sum_{\delta \in R_s} \sum_{n' \in N_v^\rho} (\theta_{n'}^{\rho} \cdot C_\lambda^\delta \cdot B_n^{\rho}) \leq \theta_n^\lambda \quad \forall n \in N_i \quad \forall \lambda \in R_c \quad (2)$$

$$\sum_{\delta \in R_s} \sum_{(k,l) \in E_v^\rho} (\psi_\delta^{(k,l)} \cdot C_\lambda^\delta \cdot B_{(n,m)}^{(k,l)}) \leq \Psi_{(n,m)}^\lambda \quad \forall (n, m) \in E_i \quad \forall \lambda \in R_c \quad (3)$$

$$\sum_{n' \in N_v^\rho} B_n^{\rho} = 1 \quad \forall n \in N_i \quad (4)$$

$$\sum_{m \in N_i} B_{(n,m)}^{(k,l)} - \sum_{m \in N_i} B_{(m,n)}^{(k,l)} = B_n^k - B_n^l \quad \forall n \in N_i, \forall (k, l) \in E_v^\rho \quad (5)$$

$$\sum_{n \in N_i} \sum_{n' \in N_v^\rho} (D_{Proc}^{n'} \cdot B_n^{\rho}) + \sum_{(n,m) \in E_i} \sum_{(k,l) \in E_v^\rho} (D_{Trans}^{(k,l)} \cdot B_{(n,m)}^{(k,l)}) \leq D_{th}^\lambda \quad \forall \lambda \in R_c \quad (6)$$

$$\sum_{n' \in N_v^{\rho^*}} B_n^{\rho} = 1 \quad \forall n \in N_i^* \quad (7)$$

$B_n^{\rho}$  (resp.  $B_{(n,m)}^{(k,l)}$ ) is a binary variable indicating whether VNF instance  $n'$  (resp. virtual link  $(k, l) \in E_v$ ) is mapped into a particular PoP  $n$  (resp. into the physical link  $(n, m) \in E_i$ ). Also, we note  $C_\lambda^\delta$  a binary variable indicating whether  $\lambda$  corresponds to the traffic class of the slice request  $\delta$ .

Constraint (2) ensures that the sum of allocated computing resources required by VNF  $n'$  mapped into PoP  $n$  does not exceed the amount of available resources in its class of traffic. Similarly, constraint (3) ensures that each link has enough available capacity to support the virtual links mapped over it. Constraint (4) states that each VNF has to be mapped only once into the physical infrastructure. In other words, the whole amount of resource (Computer, Memory and Storage) allocated to a given VNF must be provided by exactly one physical node to avoid dispatching a VNF over multiple POPs.

Constraint (5) consists in building the virtual paths between the required endpoints. This chaining constraint is used to enforce the condition that for each virtual link there must exist a continuous path allocated between the pair of physical nodes in which VNFs have been mapped. Constraint (6) ensures that each deployed SFC will not exceed the end-to-end delay threshold that is specific to each traffic class. The first part of the equation is a sum of the delay incurred by packet processing on VNFs, while the second part defines the delay incurred by transmitting packets between these VNFs. Last, constraint (7) allows to place a specific type of VNF ( $n' \in N_v^{\rho^*}$ ) into a particular physical placement ( $n \in N_i^*$ ). For example, the SBG-VNF must be placed in the PoPs near the satellite Hubs that implements the physical part of the gateway in terms of SBG-PNFs. The fronthaul link between SBG-VNF and SBG-PNF imposes strict requirements in terms of performance such as latency. This allows the satellite gateway to work properly.

In a second step, the on demand aspect to our solution is achieved by an online algorithm that at the begging of each time window, the online algorithm updates available resources. For this solution, we propose to favor Slice Request (SR) with a long lifetime, other variants may be proposed such as short lifetime slice request first. Therefore, the algorithm calls the original MILP for each class of traffic using as inputs an ordered set of SRs according to their lifetime  $\tau$ . The MILP returns the set of accepted SRs and the placement of their corresponding SFCs. During a time window, the algorithm cannot accept any new SR until the next round. However, it keeps track of the possible upscaling of SRs that have been already instantiated. Also, when a given SR reaches its lifetime its allocated resources will be immediately released and available for resource pool ready to be allocated. At the end of the time window, extra resource are allocated to SRs with upscaling needs. Depending on the available resources and the current placement of the SFC, the upscaling of SR can be realized by satisfying locally the new required resources (e.g. a VNF may obtain new CPUs from the same PoP in which it is already mapped. Similarly, a virtual link capacity can be enhanced over the same physical link without need to be mapped over another link) or by applying a global optimization and calling the original MILP. It is worth noting that upscaling the resources over the satellite link will use the mechanism for Bandwidth on Demand presented such as presented in [7]. Another upscaling scenario that should be carefully considered could be envisioned with the migration of VNFs to other POPs. In the other hand, when a resources downscaling are released, they

will be available in the pool for the next time window. In addition, several scenarios such as simple request / response (without negotiation) and complex request/response (with negotiation) can be considered in this procedure.

To understand our model for QoS constraint, Fig. 2 illustrates a creation of slice by deploying SFC composed of 5 VNFs over a simple satellite segment topology composed of 6 PoPs. Three QoS classes are implemented in different PoPs and links providing differentiated performances. For example, the link between PoP1 and PoP2 has a latency of 5ms for C1 (QoS Class 1) and 7ms for C2 (QoS class 2) and 15ms for C3. Furthermore, each PoP may provide a set of differentiated resources (CPU, disk, memory) corresponding to a specific QoS class, which impacts the processing delay of a given VNF and generates different performances depending on the attributed resource (for example high performance vCPU vs. low performance vCPU). In this case, the slice QoS ensure the aggregation of individual QoS offered to the SFC.

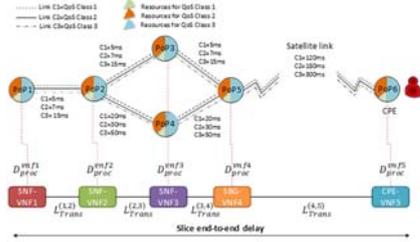


Fig. 2. Example of 5 VNFs SFC and 6 POPs topology with two QoS classes

For simplicity, we suppose that the processing delay of each VNF is set to be  $D_{proc}^v = 2ms$ . The result of our optimal placement and chaining of this SFC will be over {PoP1, PoP2, PoP3, PoP5, PoP6} with QoS class 1. This slice will provides an end-to-end delay of  $\{2+5+2+5+2+5+2+120+2\}=145ms$  which is less than the end-to-end delay threshold (180ms). The other placement {PoP1, PoP2, PoP4, PoP5, PoP6} will not be selected as it cannot guarantee the delay threshold.

## V. PERFORMANCE EVALUATION

In this section, we evaluate the slice creation in terms of placement performances of both OnDReAMS and QoSAM models using different type of SFC. Both models were implemented using AIMMS Modeling Optimization version 4.3 [10] and experiments were conducted on Windows 8 server with Intel Core i7-3740QM processor with 16GB of memory. All evaluations are repeated 20 times. We first describe the simulation environment and then we discuss the performances evaluation metrics.

### A. Simulation environment

We use the topology described in Fig. 2 as baseline topology. In our simulation, the available PoPs resources (resp. requested resources) and the available capacities of physical links (PLs) (resp. requested virtual links capacities) are configured to have fixed values as presented in Table III. In addition, we define three type of generic slice requirement using three traffic classes (QoS1, QoS2, and QoS3) to cater different categories of service with different QoS levels. Each QoS level imposes performance parameters mainly the end-to-end delay threshold  $D_{th}^{\lambda}$  as expressed in Table III.

All formalized models are evaluated using four structural variants of SFC. The first component **L1** is a linear chain composed of a sequence of VNFs connecting two endpoints “S” and “D”. The second component **B1** consists of a bifurcated chain using different VNFs in each path connecting two endpoints “S” and “D”. The third and fourth components (**L2** and **B2**) use the same structure of the ones described previously but with varying the number of VNFs (see Table III. ).

Table III. SIMULATION PARAMETERS

Parameters	Value range
------------	-------------

Number of VNFs per service for linear (L) and bifurcated (B) SFC	L1, B1	[1, 3]	
	L2, B2	[4, 6]	
Delay threshold $D_{th}^{\lambda}$ for a slice	Traffic class « QoS 1 »	150 ms	
	Traffic class « QoS 2 »	300ms	
	Traffic class « QoS 3 »	600ms	
Available resources at PoPs	set at 100%		
Requested resource	set at 1%		
Available capacity of PL	set at 100%		
Required capacity of virtual link	set at 1%		
Processing delay $D_{proc}^v$	2ms		
PoP and link resource distribution per QoS class	Scenario	1	2
	QoS1	20%	50%
	QoS2	30%	25%
	QoS3	50%	25%

In order to evaluate the two formulations (OnDReAMS and QoSAM), we selected some result metrics adopted in several works [11][12]. Thus, for each model, we measured the average end-to-end delay, the average number of QoS violation and the average of accepted slice requests, using two type of the SFCs (linear and bifurcated) with different resource distribution per QoS class according to 2 scenarios for resources distribution as described in Table III.

### B. Simulation Results

First, we analyze the slice end-to-end delay provided with both OnDReAMS and QoSAM. Fig. 3 depicts the average delay measured between endpoints when increasing the number of slice requests for scenario 1 (same results for scenario 2). The end-to-end delay is computed as a sum of VNFs processing delays and transmission delays along the SFC path (as illustrated in topology of Fig. 2).

The results Fig. 3 show that OnDReAMS provides the adequate end-to-end delay compared to QoSAM regardless the number of processed slice requests. When a slice requests is accepted by OnDReAMS, its end-to-end delay is guaranteed. This difference in performance is due to the delay constraint that guides the solver to place VNFs of a given SFC in a manner to ensure not exceeding the required delay threshold specific to each QoS class (result for QoS1, QoS2, and QoS3). In addition, in the case of OnDReAMS, the fact of partitioning resources between different QoS classes guarantees a more efficient placement by allocating the appropriate resource to reach the needed performance in terms of delay. While, QoSAM is unable to differentiate between QoS classes, still less meeting their delay requirements, which may increase the resulted end-to-end delay and generate QoS violation cases for the slice.

When increasing the number of slice requests, OnDReAMS provides a better end-to-end delay, especially with QoS1 and QoS2. This is due to its ability to place the VNFs in a manner to meet its end-to-end delay by allocating exclusively the dedicated resources according to QoS level. However, we observe that allocation of slice with QoS3 experiences less delay with QoSAM compared to OnDReAMS. Such result is mainly due to the ability of QoSAM to use resources without distinction that leads to a convergence of end-to-end delay to an average delay over the path.

Furthermore, by conception, QoSAM is not supposed to respect the distribution of resources per QoS class. This, allows QoSAM to minimize resources consumption by performing a free-class placement of VNFs that uses any available resources without considering delay requirements.

In order to investigate the possible hidden problems behind QoSAM, we measured the number of QoS violation defined as the percentage of slice request exceeding the end-to-end delay threshold among total number of request. Fig. 3, depicts the evolution of QoS violation percentage observed by QoSAM for different portion of QoS1 slice when increasing the number of requests. In the case of OnDReAMS there is no QoS violation because of the strict delay threshold constraint that obliges the solver to reject a request when its QoS class requirement (mainly end-to-end delay) cannot be honored.

We notice that QoSAM starts generating QoS violation cases since 5 requests and their number depends on the portion of QoS1 slice requests. Indeed, such stringent QoS class requirements are more likely to be violated since QoSAM has no delay constraint to respect. Furthermore, the QoS violation reaches its maximum when 50% of slices requests are of type QoS1 class while with 10% QoSAM provides a low QoS violation level. In other words, when using a model that ignores completely delay constraints, the number of QoS violation depends on the number of requests with strict QoS requirements.

To better understand the behavior of OnDReAMS and QoSAM in terms of requests acceptance, Fig. 4 shows the average rate of accepted slice requests for both models in different scenarios using different SFC. As expected, QoSAM solution achieves a better rate of globally accepted slice requests in overall scenarios whereas OnDReAMS tends to reject requests when exceeding a specific number of slices. QoSAM continue to accept slice requests until overloading the network resources but without guaranteeing a convenient QoS performance. Additionally, QoSAM may provide unnecessarily high QoS performances to satisfy slice requests of class QoS2 or QoS3, which leads to a possible QoS violation of QoS1 slice request. In the other hand, the delay constraint adopted by OnDReAMS avoids QoS violation but provokes an early slice requests rejection (for example in scenario 2 “linear” SFC, rejection is noticeable from 55 slice requests). OnDReAMS, as it is designed, begins to reject requests of a given QoS class when the resource reserved to this class is overloaded, even though unused resources reserved to the other class are available. Such problem can be addressed by integrating an auction-based mechanism to deal with idle non-used resources or by allowing sharing / borrowing resources among slices [13]. In addition, introducing QoS negotiation mechanism to allow one slice to move from one QoS level to another (upgrading or downgrading) can be a solution to this problem.

## VI. CONCLUSION

In this paper, we presented an approach that pushes SDN/NFV technology enabler into the satellite domain to provide enhanced satellite communications service delivery and achieve a better integration of the satellite segment within the 5G ecosystem. In particular, we designed an SDN/NFV-based architecture framework for on-demand satellite network slicing. This architecture provides flexible service chaining and provisioning taking into account diversified service requirement while meeting performance expectations from service level perspective. Performance measurements of the slicing procedure have been conducted and results show that the proposed OnDReAMS model provides better QoS level in terms of end-to-end delay to meet service requirement.

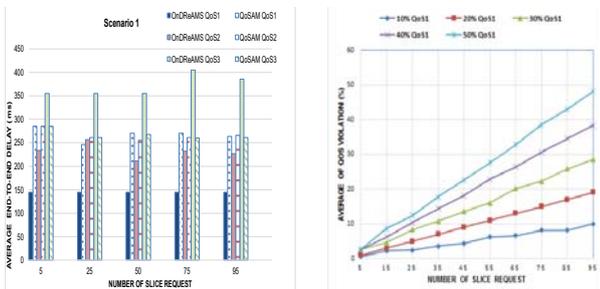


Fig. 3. The E2E delay performance and QoS violation case

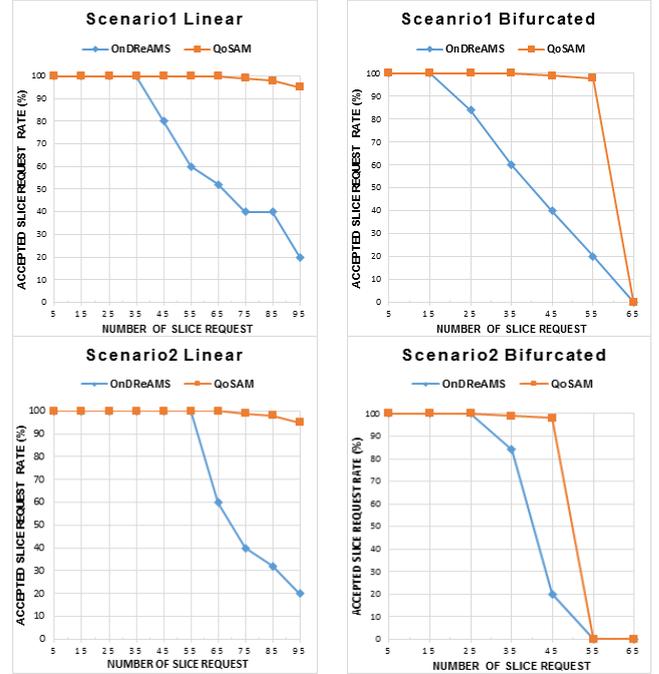


Fig. 4. SFC acceptance comparison in different scenarios

## ACKNOWLEDGMENT

Research leading to these results has received funding from the European Union’s H2020 Research and Innovation Programme (H2020-ICT-2014-1) under the Grant Agreement H2020-ICT-644843.

## REFERENCES

- [1] H2020 VITAL research project website at <http://www.ict-vital.eu/>
- [2] Bertaux L., Medjah S., Berthou P., Abdellatif S., Hakiri A., Gelard P., “Software Defined Networking and Virtualization for Broadband Satellite Networks”. IEEE Communications Magazine, March 2015
- [3] R. Ferrus, H. Koumaras, O. Sallent, G. Agapiou, T. Rasheed, M.-A. Kourtis, C. Boustie, P. Gelard, T. Ahmed, SDN/NFV-enabled satellite communications networks: Opportunities, scenarios and challenges, Physical Communication, November 2015
- [4] Sacchi, C.; Bhasin, K.; Kadowaki, N.; Vong, F., "Toward the "space 2.0" Era [Guest Editorial]," Communications Magazine, IEEE , vol.53, no.3, pp.16,17, March 2015
- [5] NetWorld2020’s – SatCom WG The role of satellites in 5G, Version 5 – 31th July 2014
- [6] 3GPP TR 22.891 V1.1.0, “Feasibility Study on New Services and Markets Technology Enablers;Stage 1 (Release 14)”, November 2015
- [7] T. Ahmed, R. Ferrus, R., Fedrizzi, R., and Sallent, O. (2017, May). Towards SDN/NFV-enabled satellite ground segment systems: Bandwidth on Demand use case. In Communications Workshops (ICC Workshops), IEEE International Conference on (pp. 894-899). IEEE. May 2017.
- [8] NetWorld2020, “Public Private Partnership in Horizon 2020: Creating a Smart Ubiquitous Network for the Future Internet”, November 2013.
- [9] P. Quinn and T. Nadeau, “Problem Statement for Service Function Chaining” IETF RFC 7498, April 2014.
- [10] J.Bisschop, AIMMS optimization modeling. Lulu. com, 2006.
- [11] R. Riggio, A. Bradai, T. Rasheed, J. Schulz-Zander, S. Kuklinski, and T. Ahmed, “VNFs Orchestration in Wireless Networks,” In Proc. of IEEE CNSM, 2015.
- [12] M. Barshan, H. Moens, S. Latre, and F. De Turck, “Algorithms for efficient data management of component-based applications in cloud environments”, in Proc. of IEEE NOMS, 2014
- [13] M. Jiang, M. Condoluci, and T. Mahmoodi, “Network slicing in 5G: an auction-based model”. In IEEE International Conference on Communications. 2017.