Slice Aware Non Terrestrial Networks

Youssouf Drif, Emmanuel Lavinal, Emmanuel Chaput, Pascal Berthou, Boris
Tiomela Jou, Olivier Grémillet, Fabrice Arnal

▶ To cite this version:

HAL Id: hal-03273402
https://laas.hal.science/hal-03273402

Submitted on 29 Jun 2021

Slice Aware Non Terrestrial Networks

Youssouf Drif[*†], Emmanuel Lavinal[†], Emmanuel Chaput[†],
Pascal Berthou[‡], Boris Tiomela Jou[§], Olivier Grémillet[*], Fabrice Arnal[*]
[*]IRT Saint-Exupéry, France; Email: firstname.lastname@irt-saintexupery.com
[†]IRIT, Toulouse, France; Email: firstname.lastname@irit.fr
[‡]LAAS, Toulouse, France; Email: firstname.lastname@laas.fr
[§]Airbus Defence and Space, Toulouse, France; Email: firstname.lastname@airbus.com

Abstract—With the recent integration of Non Terrestrial Networks (NTNs) into 3GPP Release 17, 5G networks are expected to benefit from the NTN large coverage area. This integration will help mobile terrestrial networks reach a worldwide coverage. However, this ultimate ubiquity also comes with its set of challenges to overcome. One of the main issues is the seamless integration of NTNs into the existing mobile network standard. In this paper, we propose a comprehensive architecture integrating NTNs as slice-aware backhaul links. This architecture remains fully compliant with the 3GPP standard. For this purpose, we propose an end-to-end slice model integrating NTNs and 5G networks. Then, we implement this model on a 5G-satellite testbed, adding new functional components to interconnect both networks at the control and data plane levels. Lastly, we evaluate the performances of our method using the aforementioned testbed by monitoring each slice and their related Quality of Service requirements.

Index Terms—5G, Non Terrestrial Network, Network Slicing, Software Defined Network (SDN), Network Function Virtualization (NFV)

I. Introduction

5G networks have been introduced with the main focus on vertical markets integration resulting in the adoption of the network slicing concept. The 5G network can be partitioned into optimized slices dedicated to heterogeneous services with specific requirements. To achieve the end-to-end Key Performance Indicators (KPIs) of each service, the network slicing concept needs to spread across all segments of the network including the Transport Network (TN). Major satellite network research breakthroughs have been made during the last decade resulting in Very High Throughput Satellite (VHTS) systems and mega-constellation projects such as Starlink or Kuiper. These advances increase the capacity, the flexibility and improve the overall performances of satellite systems making them a serious candidate for the TN. Nevertheless, integrating NTNs in 5G networks is a real challenge to leverage the capabilities of end-to-end slices.

Inherent benefits of satellite networks such as their worldwide coverage area and fast deployment can make them a real accelerator for mobile terrestrial networks. The network slicing paradigm, in our opinion, is the enabler for a coherent integration. We can state that, to a certain extent, satellite networks are already "sliced by nature" with the various types of satellites orbiting at different heights dedicated to specific services and thus acting as multiple slices (e.g. Geostationary Orbit (GEO) for broadcast and Low Earth Orbit (LEO) for

high-performance broadband). These satellite systems could be integrated in 5G slices as slice-aware NTN backhaul links but this integration represent considerable challenges.

The 3GPP standard does not yet provide all the tools to deploy and manage end-to-end slices nor provide an overall slice management system or interfaces to integrate external actors such as TN providers. Another major constraint is the lack of unified management interface in NTNs. Indeed, even with the recent accomplishments in the satellite field, systems greatly vary and do not have yet a unified management framework. Consequently, additional functionalities must be added to the 5G and satellite networks to fully support the end-to-end service across the NTN segment.

In this paper, we focus on implementing end-to-end slices across 5G networks and NTNs. We first define the required parameters to interface the NTN with the 5G network to ensure the slice orchestration and life-cycle management. Then, we define an end-to-end slicing model which integrates seamlessly the NTN as a slice-aware link within the 5G end-to-end slice. In order to maintain slice and Quality of Service (QoS) continuity, we interconnect the control and data planes of both networks, introducing new functional components and interfaces. Finally, we implement this model on our satellite-5G testbed, evaluate its performances against multiple real-life scenarios and demonstrate slice and QoS continuity across domains.

This paper is organized as follows: we summarize the state-of-the-art on network slicing within the 3GPP standard and describe related work on the integration of the NTNs with mobile networks in section II. We define the parameters for the management interface and our end-to-end slicing model in section III. Our end-to-end architecture and container-based open-source testbed implementation is described in section IV alongside our test scenarios. We evaluate the performances of our model using our testbed in section V. Finally, we present future work in sectionVI and conclude in section VII.

II. Non Terrestrial Networks in 5G

In this section, we describe the current 3GPP standard and its current state on the network slicing topic. We also describe the broadband NTN architecture and related work on the satellite integration into mobile terrestrial networks.

A. 3GPP Architecture

The new 5G Service Based Architecture (SBA) [1] leveraging virtualization technologies and cloud-ran approaches enables vertical markets and an unprecedented flexibility level in mobile networks. It now supports network slicing in the Radio Access Network (RAN) [2] and the Core Network (CN). A slice is an end-to-end partition in the network and is composed of various network slice subnets [3]. These slice subnets usually match to a network domain such as the 5G CN or the RAN domain and are composed of specific Network Functions (NFs) corresponding to the domain. The service profile [4] defines the technical parameters for each slice such as their maximum latency. Each slice subnet is also represented by a slice profile which specifies its technical parameters. The service and slice profiles are both derived from the GSMA Generic Network Slice Template (GST)/Network Slice Types (NEST) [5] which is the reference for the mobile network slicing in terms of main specification. The service profile stands as the Service Level Specification (SLS) of the slice and the GST/NEST defines its Service Level Agreement (SLA). End-to-end KPIs [6] are defined for the 5G network and some are dedicated to the network slicing.

Slices are identified by the Single - Network Slice Selection Assistance Information (S-NSSAI) which is the composition of the Slice/Service Type (SST) and the Slice Differentiator (SD). The SST identifies the type of the slice and the SD complements this information in case of multiple identical types of slice in the same Public Land Mobile Network (PLMN). A User Equipment (UE) attaches to one or multiple slices using its PDU sessions. When establishing a PDU session [7], a UE indicates in the control plane messages the S-NSSAI it wants to register on. A UE can belong to multiple slices simultaneously and within each PDU session multiple flows could be established with different QoS requirements. The 5G network provides end-to-end QoS using the QoS flow system. Each flow has a QoS Flow ID (QFI) which identifies the associated 5G QoS Identifier (5QI). 5QIs are standardized values with QoS characteristics such as the resource type (e.g. Guaranteed Bit Rate (GBR), non-GBR) or Packet Delay Budget. To process the QoS flows, the RAN slice subnet maps them to Data Radio Bearers (DRB). The CN slice subnet multiplexes them into GTP-U tunnels which implement the IP stack QoS (e.g. DiffServ [8] or GTP extension headers).

Slice orchestration relies on the 3GPP Service Based Management Architecture (SBMA) framework [9], [10]. Management Services (MnSs) are defined in this framework to manage the slice lifecycle. The MnSs follow the producer-consumer pattern and each MnS exposes services which can be consumed by other MnSs. In the current standard, few of those MnSs are specified with their procedures [11] (e.g. instantiate a new network slice or network slice subnet) and it is up to the 5G operator to implement them into Management Functions (MnFs). Work on network slicing is still in progress and the standard does not include yet all the network slicing aspects such as end-to-end management or the integration with other actors. Phase 2 of network slicing standardization has started in Release 17 [12] and aims at improving the slicing support in 5G networks and beyond.

B. NTN Integration into 5G

Similarly to 3GPP terrestrial networks, satellite networks are Wide Area Networks (WANs) with their own standards, technologies and protocol stacks. The architecture is composed of three different segments: the user segment which holds the satellite terminal, the ground segment which holds the satellite gateways and the control components of the network, and finally the space segment which holds one or multiple satellites. The Satellite Terminal (ST) is similar to the UE in 5G networks. It communicates with the space segment which relays its messages to the satellite gateway. Space segments can be composed of one or multiple satellites depending on their orbit and their type of service. For instance, GEO satellites are well suited for broadcasting and broadband services without requirements in terms of low latencies. LEO and Middle Earth Orbit (MEO) have increased performances thanks to their lower altitude, however satellite systems using them are more complex than GEO ones. Satellite gateways have multiple roles such as allocating resources to the STs on the forward and return links, Authentication, Authorization and Accounting (AAA), or data-plane processing. They are a central component of the satellite architecture and can communicate with multiple STs and satellites simultaneously.

Researches have been conducted on the integration of satellite networks in 4G first then in 5G networks [13], [14]. Works on the integration of satellite systems into mobile terrestrial networks started long ago. Pioneered projects such as VITAL, focusing on the need of the adoption of virtualization and softwarization technologies into satellite networks [15], highlighted key challenges and benefits of such an integration. Sat5G followed the path of VITAL and studied the integration of satellite networks into terrestrial networks [16]. It focused on the analysis of various aspects of the 3GPP standard on 5G including the new SBA, the 5G New Radio (NR) and the QoS system. It outlined various levels of integration from a simple TN to a fully 3GPP compliant satellite system, each integration level associated to a short-mid-long term road-map with its own challenges to overcome. The outcome of the project showed that a full compatibility of satellite systems with 3GPP specifications requires considerable work on satellite and mobile networks architectures. Still, it is feasible due to the convergence and wide adoptions of cornerstone technologies such as SDN and NFV in both networks. Following the work of the latter projects, the SATis5 project has developed a platform to demonstrate use-cases of MEO and GEO satellite integration within 5G networks (e.g. Video delivery, NB-IoT). More recently the SUPER-G project has proposed an evolution of the satellite network architecture. Key challenges of network slicing are applied to the satellite network in order to introduce a Satellite Slice as a Service framework [17]. This framework provides a method to "slice" the inner satellite network and also guidelines to use these slices as a TN for 5G networks.

III. END-TO-END SLICING MODEL

In this section, we describe our end-to-end model integrating an NTN as a slice aware TN in the 5G terrestrial network. The goal of this model is to formalize our slice definition in the 3GPP context. In addition, it acts as a reference specification for the architecture we propose in section IV.

A. Model overview

We define our model and architecture with the main focus of establishing 5G end-to-end slices while ensuring the QoS and slice KPI requirements.
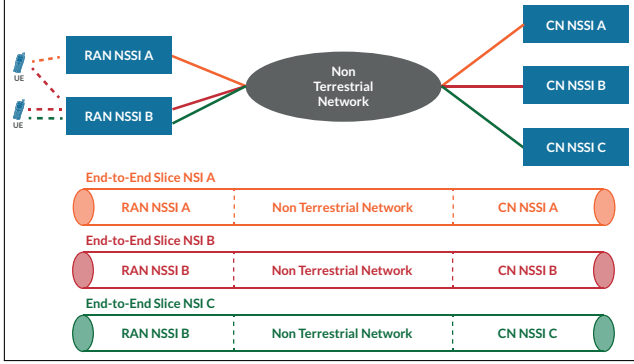

Fig. 1: 5G end-to-end slice with NTN as a slice aware TN

Figure 1 depicts three Network Slice Instances (NSIs) composed of RAN Network Slice Subnet Instances (NSSIs) and CN NSSIs. In our model, the NTN provides a slice-aware link between each RAN NSSI and CN NSSI. The stitching between the three components results in the end-to-end NSI. For instance, the composition of the RAN NSSI A, the link provided by the NTN and the CN NSSI A constitutes the NSI A. We generalize this approach and define in Table I our slicing model.

TABLE I: End-to-end slice model

Symbol	Description
S	The set of 5G end-to-end NSIs
$S_i \in S, i \in \mathbb{N}$	The ith NSI
S_{RAN}	The set of 5G RAN NSSIs
$Sr_j \in S_{\text{RAN}}, j \in \mathbb{N}$	The jth RAN NSSI
S_{CN}	The set of 5G CN NSSIs
$Sc_k \in S_{\text{CN}}, k \in \mathbb{N}$	The kth CN NSSI
T	The set of links provided by the NTN
$T_j^k \in T, (j,k) \in \mathbb{N}^2$	The NTN link between Sr_j and Sc_k

With the help of these parameters, we can now introduce the stitching function, defined in equation (1). This function provides an end-to-end slice as described in equation (2).

$$stitch : S_{\text{RAN}} \times S_{\text{CN}} \times T \to S$$
$$(Sr_j, Sc_k, T_j^k) \mapsto Sr_j \oplus T_j^k \oplus Sc_k \quad (1)$$

$$S_i = stitch(Sr_j, Sc_k, T_j^k), (i,j,k) \in \mathbb{N}^3 \quad (2)$$

B. NTN Management Interface

The first step of the stitching is to define the interface between the NTN and the 5G network management components. Throughout the complete slice life-cycle, this interface will be used to interact with the NTN. From the service profile parameters and satellite specificities, we have extracted the main parameters required for establishing and configuring an NTN link. Additional parameters (such as security ones) will be added in the future. It is up to the 5G operator to instantiate these parameters according to its end-to-end requirements. The satellite operator can then refine and derive these parameters into satellite NFs. The 5-uplet $\Theta(\lambda, \Delta, \mu, \beta, \sigma)$ which identifies the NTN link reference parameters are defined as follows:

- λ: **maximum latency** in milliseconds. This parameter determines which space segment the satellite operator should use when providing the NTN link. Depending on the needs of the 5G operator in terms of QoS, this parameter also drastically influences the satellite operator's choice in ground segments and technologies.
- Δ: **jitter** in millisecond. Due to their architecture and the influence of weather on satellite communications, a non-negligible jitter could be introduced. "Non-constant" delays could provide significant performance issues in 5G networks such as delays in voice communications or more critical control plane messages. With this parameter specified by the 5G operator, satellite operators can deploy proper NFs and configure them in order to reduce this delay.
- μ and β: **target download and upload throughput** in Mbit/s. Forward and return links technologies differ and, depending on the required throughput, satellite operators will use a certain set of frequency band, protocols and equipments.
- σ: **resource sharing** (boolean) with other slices. This parameter influences the isolation level, thus the type of radio resources (e.g. Single Channel Per Carrier (SCPC), MF-TDMA) as well as compute resources.

C. Slice and QoS continuity

Using Θ, we are able to provide a T_j^k NTN link, tune it according to the 5G operators needs and reserve the corresponding resources. However, this is not enough to consider the NTN link as "slice-aware" since slice and QoS continuity must coexist in the data plane. To ensure an end-to-end slice across all network domains and QoS continuity within each slice, satellite and 5G domains must identify, for each packet, first the slice it belongs to and then its requirements in terms of QoS. Every UE establishes multiple sessions and each one belongs to a single slice. Within each PDU session multiple flows with different QoS requirements are also established. Table II defines our model for the UE and PDU sessions.

To ensure the isolation of both network domains and the correct traffic processing, the slice and QoS properties of flows
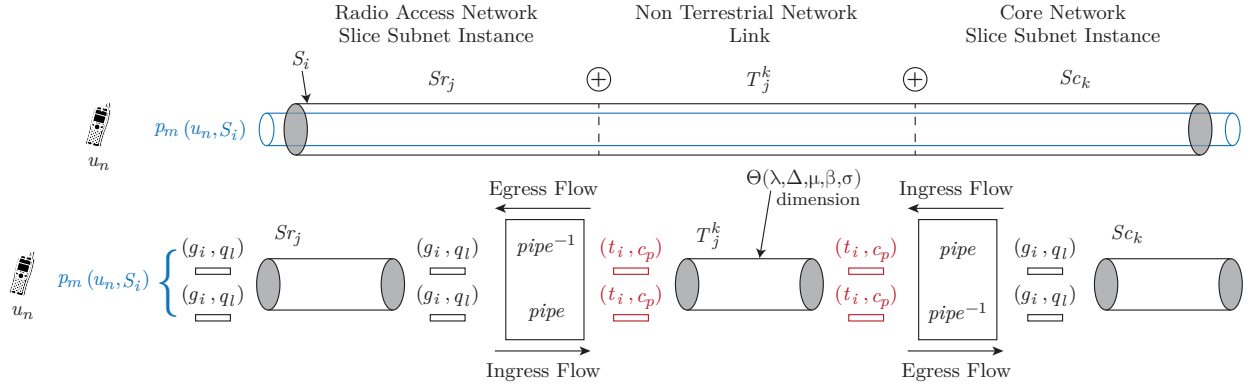
Fig. 2: End-to-end Slice Model

TABLE II: UE and PDU models

Symbol	Description
$U, card(U) = N_{\text{UE}}, N_{\text{UE}} \in \mathbb{N}$	The set of UEs with a total of N_{UE} UEs
$u_n \in U, n \in [\![0, N_{\text{UE}} - 1]\!]$	The nth UE
$P^n, n \in [\![0, N_{\text{UE}} - 1]\!]$	The set of u_n PDU sessions
$p_m(u_n, S_i), i \in \mathbb{N},$ $n \in [\![0, N_{\text{UE}} - 1]\!], m \in \mathbb{N}$	The mth PDU session of u_n established on the ith slice

TABLE III: Flows model

Symbol	Description
Q_G	The set of 5QIs
$q_l \in Q_G, l \in \mathbb{N}$	The QFI identifying each 5QI
Q_S	The set of NTN QoS classes
$c_p \in Q_S, p \in \mathbb{N}$	Each individual NTN QoS class
I_G	The set of information identifying the slices in the 5G domain
$g_i \in I_G, i \in \mathbb{N}$	The information identifying the ith slice in the 5G domain
I_S	The set of information identifying the slices in the satellite domain
$t_i \in I_S, i \in \mathbb{N}$	The information identifying the ith slice in the satellite domain

must be preserved from on domain to another. We define the ingress flow as the incoming traffic from the 5G network to the satellite network and the egress flow as the outgoing traffic from the satellite network to the 5G network. The 5G slice and QoS information of the ingress flow must be translated into satellite traffic and then translated back to the 5G network. The 5G network provides end-to-end QoS using the QoS flow system. The QoS processing in the NTN is different from the 5G QoS system and both QoS systems are isolated and totally independent from each other. Moreover, the satellite QoS differs from one satellite system to another as they have their own specificities. This raises a real challenge as both QoS systems are different, we must ensure that the QoS classes of both systems are compatible. Sat5G has proposed a mapping between 5QI and satellite Class of Services (CoS) [18]. Few mappings are currently supported as most broadband satellites are GEO ones. LEO mega-constellations are game changers and open up the path to more supported 5QIs within the satellite network.

We define our flow model in Table III. Each slice is identified in the 5G domain by the g_i set of information. These pieces of information are refined from the slice S_i and the PDU sessions $p_m(u_n, S_i)$ established on this slice (e.g. g_i could be a set of L2/L3 fields). Within the slice in the 5G domain, q_l identifies the QoS of each flow. The t_i set of information identifies the slice in the satellite domain. The satellite CoS is identified by c_p. The tuples (g_i, q_l) fully identify the flows in terms of slice and QoS requirements within the 5G network. Symmetrically, the tuples (t_i, c_p) identify the flows in terms of slice and QoS requirements within the satellite network. We

define the $pipe$ function and its inverse function in equation (3) which fulfills the role of flow translation, it classifies the slice and the QoS for each network domain, plus, it translates the tuple for each network domain. This completes our end-to-end model which is shown in figure 2.

$$pipe : I_G \times Q_G \to I_S \times Q_S$$
$$(g_i, q_l) \mapsto (t_i, c_p)$$

$$pipe^{-1} : I_S \times Q_S \to I_G \times Q_G$$
$$(t_i, c_p) \mapsto (g_i, q_l) \quad (3)$$

IV. END-TO-END ARCHITECTURE

To validate our model and assess its relevance, we design a functional architecture integrating the satellite-5G networks and we implement it on our container-based testbed. We focus on the integration of the control and data planes, thus we do not develop on "orchestration plane" and consider it out of scope of our work (e.g. defining orchestration functions using the 5G SBMA or orchestration procedures).

A. Functional Architecture

As stated in section III, in order to have end-to-end slices, the NTN link provider must be able to identify slices first and
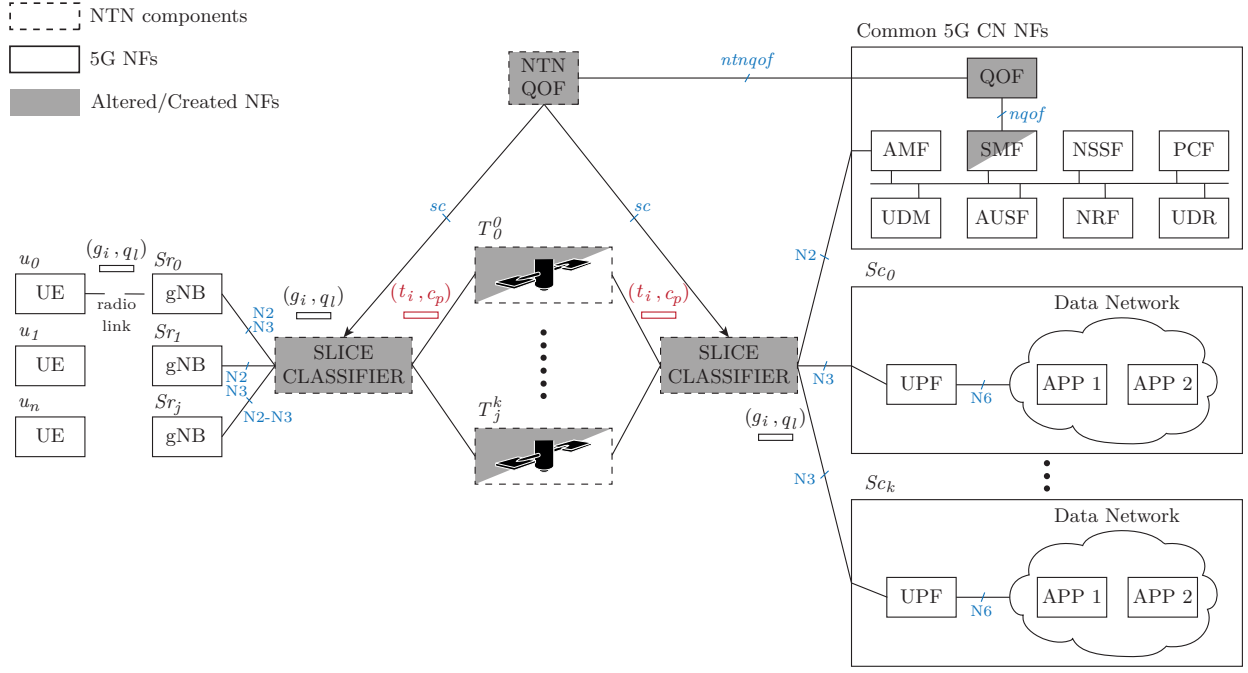
Fig. 3: End-to-end Functional Architecture

then, each individual flow within slices. The 5G architecture is flexible and addresses the vertical market integration. Various Application Functions (AFs) can be instantiated within each network slice subnet and generate traffic during the end-to-end slice lifecycle; likewise UEs can establish new PDU sessions leading to new flows in the network. Consequently, the state of flows must be continuously monitored and maintained. We have therefore extended the traditional operation of the 5G network, created new NFs, procedures and interfaces to enable an automated stitching and state monitoring in the 5G and satellite networks. Our end-to-end architecture as well as the newly introduced interfaces are shown in figure 3.

We introduce the **5G Quality of Service Function (QOF)** and the **NTN QOF** which ensures the communication between 5G and satellite networks control planes over the $ntnqof$ interface. We define a subroutine dedicated to monitor flows within both networks which is called for each PDU session creation/modification/deletion. The subroutine is depicted in figure 4 with the example of the PDU Session Establishment Procedure. The overall procedure is divided in multiple steps: after a successful registration on the network, a UE establishes multiple PDU sessions with the 5G network and triggers the 5G PDU Session Establishment procedure (step 1) which embeds an S-NSSAI. The Session Management Function (SMF) generates the information required to process the PDU session in the 5G data-plane according to the S-NSSAI identifying the slice (step 2). This is where our hook takes place, interrupts the traditional procedure and executes our subroutine. From the previously generated information, the SMF extracts and sends a subset (supported QFIs, GTP endpoint) representing the PDU session to the 5G QOF over the $nqof$ interface (step 3). The

5G QOF then translates those information into multiple (g_i, q_l) tuples identifying each flow within the PDU session (step 4) and sends them to the NTN QOF over $ntnqof$ interface (step 5). In turn, the NTN QOF translates each (g_i, q_l) tuple to the corresponding (t_i, c_p) tuple which identify the flow in the satellite network (step 6).
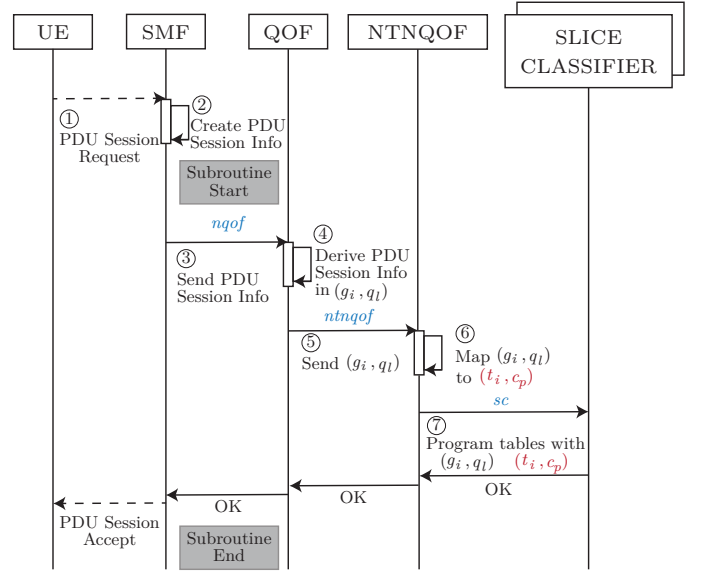

Fig. 4: PDU Session Establishment Procedure

A slice and QoS translation table must be constantly updated according to the 5G and satellites networks configuration. These are mapping tables between the satellite-5G domains and are maintained on the **Slice Classifiers**. The Slice Classi-

fiers are devices interconnecting the 5G and satellite networks data-plane which translate flows between both network domains. By relying on these tables, the Classifiers are able to steer the traffic to the adequate 5G components and NTN links. We have adopted the SDN paradigm between the Classifiers and the NTN QOF and on behalf of its translation role, the NTN QOF programs the Slice Classifiers. Over the sc interface, control messages are sent to populate mapping tables (step 7). Classifiers also monitor the traffic and implement policies related to the SLA of the negotiated link (T_j^k is sized according to Θ which represents the SLA), thus ensuring slice and QoS continuity at the interconnection points between the satellite and 5G networks. They enforce the slice and QoS policy for each flow and implement functions such as QoS based scheduler and traffic shaping policies to match the SLA of each slice.

The introduction of the aforementioned control plane NFs, the Slice Classifiers and the definition of the subroutine match the $pipe$ and $pipe^{-1}$ functions of our model.

In order to have a full end-to-end slice and QoS continuity, the NTN link provider needs to implement mechanisms within the satellite network to differentiate slices, to respect the SLA and to match equations (1) and (2). The resource provisioning, technology selection and network management inside the satellite network is out of the scope of this paper.

B. Testbed Implementation

We implement a docker-based testbed relying on existing open-source tools. We are using free5GC [19] as the 5G CN which is compliant with the 3GPP Release 15. We alter the SMF, introduce the $nqof$ interface, develop and integrate the 5G QOF. To optimize the resource consumption, we centralize all the CN NFs and for each Sc_k we only instantiate the required NFs: a dedicated User Plane Function (UPF) as well as the application servers in each data network as shown in figure 3. These servers generate traffic using the *iperf* tool. For the 5G RAN, we use the UERANSIM project [20] which implements a Release 16 compliant software gNB, provides a software UE implementation and also simulates the radio link between the UE and the gNB. We are using Trunks [21], a lightweight DVB-S2/RCS-2 satellite network simulator, it supports the QoS and Adaptive Coding and Modulation (ACM) mechanisms. We implement the NTN QOF and the $ntnqof$ interface using a SBA. We use Linux networks kernel tools to implement the Slice Classifier and develop a custom HTTP based control protocol for the sc interface.

C. Scenarios

We define a scenario in Table IV in which three end-to-end slices are launched concurrently, each slice supporting different applications and QoS requirements. The NTN providers set-up two physical links. One LEO link L with a 100 Mbps forward link capacity, a 25 Mbps return link capacity and a target latency of 45 ms with a 5 ms variable jitter. One GEO link G with a 100 Mbps forward link capacity, a 25 Mbps return link capacity and a target latency of 550 ms

with a 50 ms variable jitter. We run this scenario twice to evaluate traditional backhaul systems against our model. First, in a "slice-unaware" (SUAW) mode where S_0 and S_1 are multiplexed on L which acts as a traditional backhaul link (i.e. $T_0^0 = T_0^1 = L$) and S_2 traffic is sent on G. Then, we replay this scenario in a "slice-aware" (SAW) mode in which we deploy all the previously described mechanisms to match our model. In the SAW mode, the NTN provider is able to instantiate two logical links T_0^0 and T_0^1 on top of L. S_0 is dedicated to low latency A_0 applications, the VoIP (QFI 7) which needs a constant rate allocation of 128 kbps and web browsing (QFI 9) which generates 3 Mbps of traffic. S_1 is dedicated to video streaming (QFI 6) applications A_1 which consume 10 Mbps. The last slice S_2 is dedicated to streaming applications A_2 which consume 4 Mbps. Each UE establishes PDU sessions $p_m(u_n, S_i)$ on the slices S_i and starts generating traffic corresponding to the set of applications A_i.

TABLE IV: Scenario definition

Parameter	Description
$T = 240s$	Total duration
$S_i, i \in [\![0, 2]\!]$	Total of 3 NSI
$Sr_j, j = 1$	Total of 1 RAN NSSI
$Sc_k, k \in [\![0, 2]\!]$	Total of 3 CN NSSIs
$N_{\text{UE}} = 9$	Total of 9 UE
$T_0 = [0, 240]$	Slice 0 start from 0s to 240s
$\Theta_0 = (45, 5, 35, 10, true)$	NTN link parameter between Sr_0 and Sc_0 using LEO system
$A_0 = \{Web, VoIP\}$	Applications running in slice S_0
$T_1 = [120, 240]$	Slice 1 start from 120s to 240s
$\Theta_1 = (45, 5, 65, 15, true)$	NTN link parameter between Sr_0 and Sc_1 using LEO system
$A_1 = \{Streaming\}$	Application running in slice S_1
$T_2 = [60, 180]$	Slice 2 start from 60s to 180s
$\Theta_2 = (550, 50, 40, 15, false)$	NTN link parameter between Sr_0 and Sc_2 using GEO system
$A_2 = \{Streaming\}$	Application running in slice S_2

V. PERFORMANCE EVALUATION

The scenario generates a testbed of 64 containers running on a 32 vCPU Ubuntu VM with 128 GB of RAMs. We run the testbed and set G as the default link for the 5G control plane. UEs successfully register on the 5G network and are able to establish PDU sessions. This validates that 5G control plane procedures can be sent on GEO links ($RTT \approx 500$ ms). The results presented in this section are extracted from the average of 10 repetitions. For each flow within PDU sessions on the forward link, we measure the end-to-end Packet Delay Budget (PDB), the Packet Error Rate (PER), the jitter and the throughput for the SAW and SUAW modes. We aggregate flows per QFI and results are shown in figure 5.

At $t_0 = 0s$, UEs start generating Web and VoIP traffic on S_0 for $240s$, and at $t_1 = 60s$, UEs generate streaming traffic on S_2. As S_0 and S_2 traffic are routed through different physical satellite infrastructures, both slices are isolated and no performance impact can be measured in SAW and SUAW modes from $t_0 = 0s$ to $t_2 = 120s$. Moreover, whether for S_0 or S_2, the aggregated throughput of flows (respectively 28.15
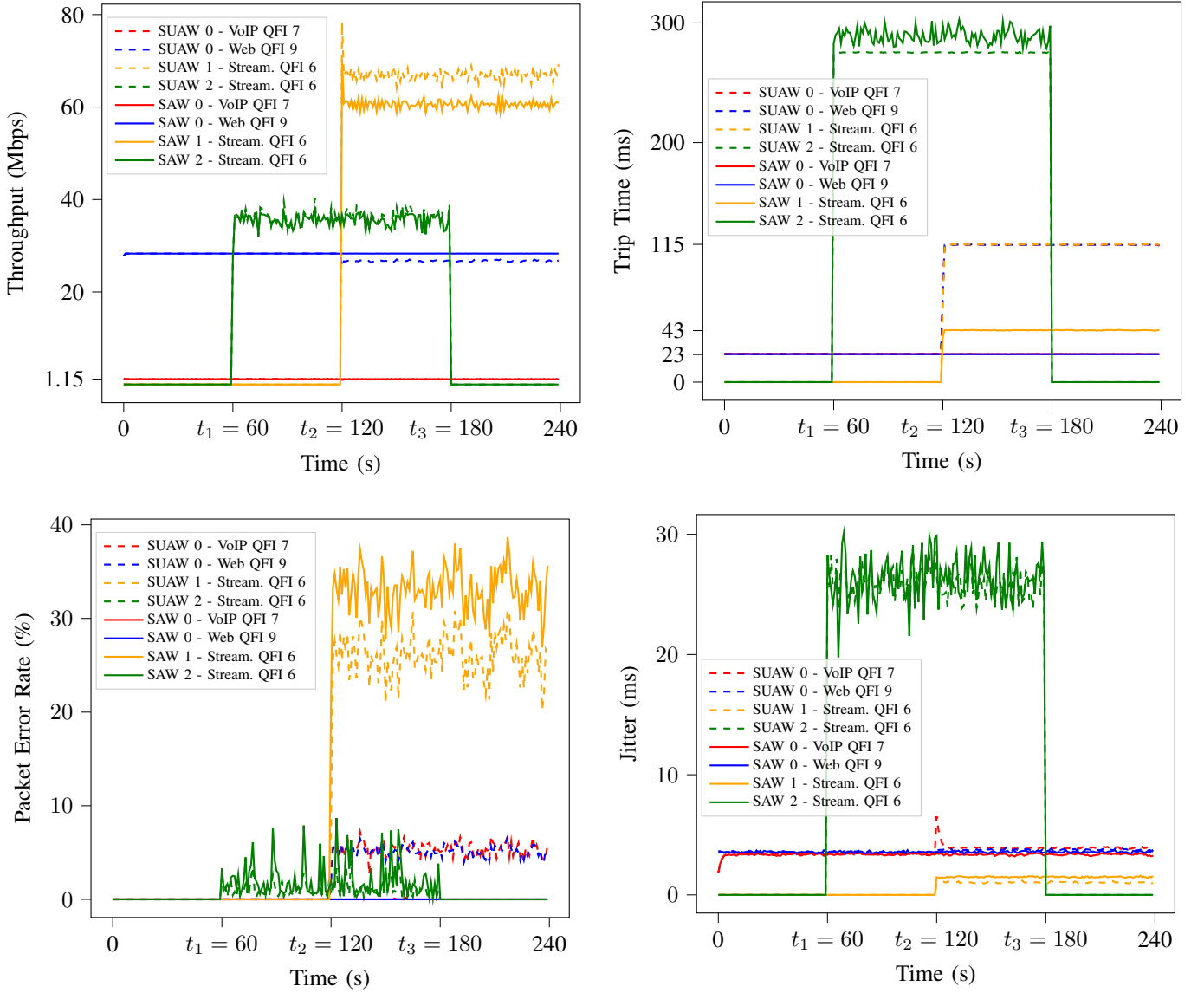
Fig. 5: Throughput, PDB, PER and Jitter of 3 Slices running concurrently

Mbps and 36 Mbps) does not exceed the slices throughput specification (respectively 35 Mbps and 40 Mbps). We are using isochronous Variable Bitrate (VBR) based streaming applications with no adaptive streaming techniques. As the NTN link introduces a non-negligible delay between the streaming servers and UEs, some packets are considered lost by the streaming receiver which is desynchronized with the server. This explains the PER for the streaming application even in ideal conditions.

At $t_1 = 120s$, S_1 starts and "greedy" users consume excessive resources (total of 90 Mbps). Without the "slice-aware" mechanisms, there is no notion of slices within the satellite networks, therefore T_0^0 and T_0^1 NTN links do not exist and we can see drastic performances drop since a congestion occurs on the NTN physical link. The total throughput ≈ 120 Mbps excesses the 100 Mbps link capacity, thus S_1 overlaps on

resources of S_0. The PDB of flows within S_0 sees an increase of 78% from 24 ms to 110 ms which is still acceptable for streaming and web applications (QFI 6 and 9) but not for VoIP ones (QFI 7) in terms of QoS requirements. The various PER spikes of the VoIP flows are also unacceptable and do not respect the QFI 7 specification.

As stated in section IV-A, being "slice-aware" allows us to deploy proper mechanisms to counteract such situations. For each slice, the Slice Classifiers implement a dedicated set of queues and schedulers. It allows us to isolate slices which prevents S_1 from consuming S_0 resources. With these mechanisms deployed, significant improvements are made and errors are not spread from S_1 to S_0. Additionally, we implement a rate monitoring system on each Classifier to avoid congestions on NTN links. As UEs still try to consume a total of 90 Mbps which would create congestion on L ($90+35 > 110$ Mbps) and

not respect the SLA defined by Θ_1, the rate monitor system caps S_1 throughput to 65 Mbps. However, the congestion is not avoided for S_1 traffic and the PER rises from an average of 25% to 35% in the SAW mode as these packet losses are not shared anymore with S_0. The target throughput of S_0 is always reached and does not suffer from S_1 overlapping. The QoS is preserved and VoIP traffic priors on streaming one. The measured PDB of web and VoIP services remains at an average of 24 ms, their respective PER is reduced to 0% and no additional jitter is perceived. S_2 still runs from $t_1 = 60s$ to $t_2 = 180s$ without impacting the performances of S_0 and S_1 and reciprocally. Those results demonstrate the pertinence of our model and outline the importance of the adoption of the network slicing paradigm across satellite networks.

VI. Future Work

At this time, we are exploring several paths for improvement. In particular, we focus on a new implementation of our slice classifier currently based on Linux kernel tools towards a data plane implementation using the P4 language in order to have higher control and flexibility on the processing of packets within the classifier. As stated in section III-A, one of the main benefits of the slicing paradigm is the ability to "play" with the slice in terms of resource allocations and NFs orchestration. Using the management interface, we are therefore evaluating the impact of the slice life-cycle management operation on our approach, such as the dynamic scaling of the slice. Such improvements could be made by integrating more parameters into the management interface and by defining new MnS and MnFs using the 5G SBMA.

VII. Conclusion

In this paper, we have presented our end-to-end slice model and architecture to integrate the satellite network in 5G networks as a slice-aware NTN link. The results clearly show that our approach allows end-to-end slice continuity while ensuring the respect of QoS requirements within each slice. Additionally, resource isolation is preserved across network domains thanks to QoS class mappings and policies enforced on slice classifiers. The architecture implementing our model is based on the 3GPP SBA and the new services we introduce are integrated seamlessly within the overall architecture, which minimizes the adoption effort for satellite and 5G actors. Moreover, our architecture maintains independent 5G and satellite management domains which is a primary requirement for both operators. The interactions between these operators are limited to slice life-cycle management (e.g. provisioning a specific NTN slice) since the interconnection of the control and data planes are fully automated.

Source and Testbed Availability

The entire work presented within this paper can be replayed and verified using our open-source container-based testbed ([Online] https://github.com/shynuu/slice-aware-ntn). Guidelines are also provided to deploy customized testbeds, run your own scenarios and evaluate their performances.

References

[1] 3GPP, "System architecture for the 5G System (5GS)," 3rd Generation Partnership Project (3GPP), Technical Specification (TS) 23.501, 12 2020, version 16.7.0. [Online]. Available: http://www.3gpp.org/DynaReport/23501.htm

[2] 3GPP, "NR; NR and NG-RAN Overall description; Stage-2," 3rd Generation Partnership Project (3GPP), Technical Specification (TS) 38.300, 1 2021, version 16.4.0. [Online]. Available: http://www.3gpp.org/DynaReport/38300.htm

[3] 3GPP, "Management and orchestration; Concepts, use cases and requirements," 3rd Generation Partnership Project (3GPP), Technical Specification (TS) 28.530, 12 2020, version 17.0.0. [Online]. Available: http://www.3gpp.org/DynaReport/28530.htm

[4] 3GPP, "Management and orchestration; 5G Network Resource Model (NRM); Stage 2 and stage 3," 3rd Generation Partnership Project (3GPP), Technical Specification (TS) 28.541, 12 2020, version 17.1.0. [Online]. Available: http://www.3gpp.org/DynaReport/28541.htm

[5] GSMA Association, "Generic Network Slice Template," Specification NG. 116, November 2020, version 4.0. [Online]. Available: http://www.ietf.org/internet-drafts/draft-henry-tsvwg-diffserv-to-qci-04.txt

[6] 3GPP, "Management and orchestration; 5G end to end Key Performance Indicators (KPI)," 3rd Generation Partnership Project (3GPP), Technical Specification (TS) 28.554, 12 2020, version 17.1.1. [Online]. Available: http://www.3gpp.org/DynaReport/28554.htm

[7] 3GPP, "Procedures for the 5G System (5GS)," 3rd Generation Partnership Project (3GPP), Technical Specification (TS) 23.502, 1 2021, version 16.7.1. [Online]. Available: http://www.3gpp.org/DynaReport/23502.htm

[8] J. Henry et al., "Diffserv to qci mapping," Working Draft, IETF, Internet-Draft draft-henry-tsvwg-diffserv-to-qci-04, April 2020.

[9] 3GPP, "Management and orchestration; Generic management services," 3rd Generation Partnership Project (3GPP), Technical Specification (TS) 28.532, 12 2020, version 16.6.0. [Online]. Available: http://www.3gpp.org/DynaReport/28532.htm

[10] 3GPP, "Management and orchestration; Architecture framework," 3rd Generation Partnership Project (3GPP), Technical Specification (TS) 28.533, 12 2020, version 16.6.0. [Online]. Available: http://www.3gpp.org/DynaReport/28533.htm

[11] 3GPP, "Management and orchestration; Provisioning," 3rd Generation Partnership Project (3GPP), Technical Specification (TS) 28.531, 12 2020, version 16.8.0. [Online]. Available: http://www.3gpp.org/DynaReport/28531.htm

[12] 3GPP, "Study on enhancement of network slicing; Phase 2," 3rd Generation Partnership Project (3GPP), Technical Report (TR) 23.700-40, 11 2020, version 1.2.0. [Online]. Available: http://www.3gpp.org/DynaReport/23700-40.htm

[13] B. T. Jou et al., "Architecture options for satellite integration into 5g networks," in 2018 European Conference on Networks and Communications (EuCNC), June 2018, pp. 398–9.

[14] G. Giambene et al., "Satellite-5G Integration: A Network Perspective," IEEE Network, vol. 32, no. 5, pp. 25–31, Sep. 2018.

[15] T. Ahmed et al., "Software-defined satellite cloud ran," International Journal of Satellite Communications and Networking, vol. 36, no. 1, pp. 108–133, 2018. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/sat.1206

[16] K. Liolis et al., "Use cases and scenarios of 5g integrated satellite-terrestrial networks for enhanced mobile broadband: The sat5g approach," International Journal of Satellite Communications and Networking, vol. 37, no. 2, pp. 91–112, 2019.

[17] Y. Drif et al., "An extensible network slicing framework for satellite integration into 5G," International Journal of Satellite Communications and Networking, 2020. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/sat.1387

[18] B. T. Jou et al., "Integrated SaT5G Detailed Network Architecture," Technical Report, SaT5G Consortium, Technical Report D3.2, April 2019. [Online]. Available: https://www.sat5g-project.eu/wp-content/uploads/2019/04/SaT5G_D3.2_ADS_v2.0_D.pdf

[19] Free5GC Consortium, "free5GC R15," 2021, V3.0.5. [Online]. Available: https://www.free5gc.org/

[20] Ali Güngör, "UERANSIM," 2021, V3.1.9. [Online]. Available: https://github.com/aligungr/UERANSIM

[21] Youssouf Drif, "Trunks - Satellite Link Simulator," 2021, V2.0.0. [Online]. Available: https://github.com/shynuu/trunks