



**HAL**  
open science

# Le "network slicing" dans les réseaux 5G : intégration d'un lien satellite avec continuité de service

Youssef Drif

► **To cite this version:**

Youssef Drif. Le "network slicing" dans les réseaux 5G : intégration d'un lien satellite avec continuité de service. Réseaux et télécommunications [cs.NI]. Institut National Polytechnique de Toulouse - INPT, 2022. Français. NNT : 2022INPT0052 . tel-04191136

**HAL Id: tel-04191136**

**<https://theses.hal.science/tel-04191136>**

Submitted on 30 Aug 2023

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Université  
de Toulouse

# THÈSE

En vue de l'obtention du

## DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

**Délivré par :**

Institut National Polytechnique de Toulouse (Toulouse INP)

**Discipline ou spécialité :**

Informatique et Télécommunication

---

**Présentée et soutenue par :**

M. YOUSOUF DRIF

le mercredi 13 juillet 2022

**Titre :**

Le « network slicing » dans les réseaux 5G : intégration d'un lien satellite  
avec continuité de service

---

**Ecole doctorale :**

Mathématiques, Informatique, Télécommunications de Toulouse (MITT)

**Unité de recherche :**

Institut de Recherche en Informatique de Toulouse ( IRIT)

**Directeur(s) de Thèse :**

M. EMMANUEL CHAPUT

M. EMMANUEL LAVINAL

**Rapporteurs :**

M. TOUFIK AHMED, UNIVERSITE DE BORDEAUX

**Membre(s) du jury :**

MME ANNE FLADENMULLER, UNIVERSITE SORBONNE, Président

M. EMMANUEL CHAPUT, TOULOUSE INP, Membre

M. EMMANUEL LAVINAL, UNIVERSITE TOULOUSE 3, Membre

M. FABRICE ARNAL, THALES ALENIA SPACE, Invité(e)

MME GÉRALDINE TEXIER, IMT ATLANTIQUE, Membre

M. OLIVIER GREMILLET, IRT SAINT-EXUPERY, Invité(e)

M. PASCAL BERTHOU, UNIVERSITE TOULOUSE 3, Membre



## Abstract

---

The advent of low-earth orbit mega-constellations and recent advances in satellite technology are reducing the cost of satellite connections and offering greater competitiveness. With connectivity reaching Gbit/s and latency times of less than 40 milliseconds per terminal, satellite networks are achieving unprecedented performance. They are mainly designed to provide connectivity to end-users but can be integrated with existing networks. One promising use case for satellite networks is their integration into 5G networks. 5G mobile networks are being deployed and offer a multitude of heterogeneous services. They are the evolution of 4G networks and integrate in their definition the innovative concept of network slicing which allows the deployment of customized end-to-end network connectivities for a specific service. Each of these connectivities is instantiated on top of a single physical network infrastructure, which allows the mutualization of resources and a better optimization of their usage. Nevertheless, network slicing introduces a number of issues and challenges that must be addressed. One of these challenges is the integration of transport networks that bridge the gap between the radio access and the 5G core network while meeting service requirements. The objective of this thesis is to show that, by relying on the Software Defined Network and Network Function Virtualization technologies as well as the network slicing, it is possible to unify the satellite and 5G networks in order to accelerate the deployment of 5G networks and to increase their capacities. In this context, we propose a novel satellite architecture that supports network slicing and integrates components both in the control plane and in the data plane to ensure end-to-end service continuity. This contribution has been validated on an open-source and extensible experimentation platform on which different scenarios of 5G slices deployment, integrating satellite links, have been tested.

---

**Keywords :** Software Defined Networks, Network Function Virtualization, Network Slicing, 5G, Non Terrestrial Networks.

---





## Résumé

---

L'avènement des mega-constellations satellites à basse orbite et les récentes avancées sur les satellites Very High Throughput Satellite (VHTS) permettent de réduire les coûts des connexions satellites et offrent une plus grande compétitivité. Avec des connectivités atteignant le Gbit/s et des temps de latence inférieurs à une quarantaine de millisecondes par terminal, les réseaux satellites atteignent des performances inédites. Ils sont majoritairement destinés à fournir une connectivité aux utilisateurs finaux mais peuvent être intégrés avec les réseaux existants. L'un des cas d'usage prometteur des réseaux satellites est leur intégration au sein des réseaux 5G. Les réseaux mobiles de 5<sup>ème</sup> génération sont en cours de déploiement et offrent une multitude de services hétérogènes. Ils sont l'évolution des réseaux 4G et intègrent dans leur définition le concept innovant de network slicing qui permet le déploiement de connectivités réseaux personnalisées bout-en-bout pour un service spécifique. Chacune de ces connectivités doit être instanciée au dessus d'une seule et même infrastructure réseau physique, ce qui permet la mutualisation des ressources et une meilleure optimisation de leur utilisation. Néanmoins, l'utilisation du network slicing introduit un certain nombre d'enjeux et de défis qu'il faut relever. L'un de ces défis est l'intégration des réseaux de transport qui font le lien entre l'accès radio et le cœur de réseau 5G tout en respectant les exigences de services. L'objectif de cette thèse est de montrer qu'en s'appuyant sur les technologies Software Defined Network et Network Function Virtualization ainsi que du network slicing, il est possible d'unifier les réseaux satellites et 5G afin d'accélérer le déploiement des réseaux 5G et d'augmenter leurs capacités. Dans ce contexte, nous proposons une architecture satellite novatrice qui supporte le network slicing et qui intègre des composants dans le plan de contrôle et dans le plan de données pour assurer une continuité de service de bout-en-bout. Cette contribution a été validée sur une plateforme d'expérimentation ouverte et extensible sur laquelle différents scénarios de déploiement de slices 5G sur des liens satellites ont été testés.

---

**Mots clés : Software Defined Networks, Network Function Virtualization, Network Slicing, 5G, Non Terrestrial Networks.**

---



---

## Remerciements

---

Cette thèse s'est déroulée dans un contexte collaboratif au sein du projet SUPER-G et a impliqué de multiples acteurs. Ce fut une aventure professionnelle et humaine extrêmement enrichissante qui m'a permis de me découvrir les mondes académiques et industriels, et surtout de grandir. À cet effet, je tiens à remercier nominativement chacune des personnes qui ont rendu ce travail possible et aidé à parcourir ce long chemin.

Tout d'abord, merci aux membres du Jury de thèse, les rapporteurs qui ont analysé mes travaux en détail Toufik Ahmed et Guillaume Urvoy-Keller ainsi que les examinatrices Anne Fladenmuller et Géraldine Texier.

Un grand merci à toute l'équipe encadrante qui m'a soutenue et épaulé : Emmanuel C., merci pour tes conseils, ton soutien moral et ta patience. Ce fut trois années qui m'ont permis de beaucoup apprendre à tes côtés et de développer des compétences trop longtemps délaissées. J'en ressors avec plus de confiance en moi, une vision plus riche de l'écosystème académique et de ses contributions scientifiques ainsi qu'une solide base pour la poursuite de mon travail. Tu m'as énormément soutenu avant les débuts de la thèse et je ne l'oublie pas.

Emmanuel L., merci également pour ton encadrement, ta patience et tes conseils avisés. Ce fut vraiment un plaisir de collaborer sur cette thèse avec toi, tu m'as aidé à atteindre un objectif de vie et je me suis vraiment éclaté à bosser sur de la technique et à écrire nos papiers. Tu as cru en mes capacités et m'as poussé à fournir un travail de qualité avec des accomplissements technique inespérés.

Merci Pascal pour tes sages conseils et ta bonne humeur, nos discussions sur la recherche, nos moments de détente qui m'ont permis de me ressourcer et de repartir de plus bel. Tu as apporté une certaine harmonie au sein de la thèse en assurant une excellente cohésion entre le monde académique et industriel. Ton œil avisé m'a permis de me remettre en question et d'améliorer mon travail qui aujourd'hui est une réussite.

Un grand merci à toi Boris qui m'a suivi depuis mon stage de fin d'études et accompagné jusqu'ici, le chemin a quelques fois été brumeux indépendamment de notre volonté, mais nous y sommes arrivés. Tu as été un élément central pour la réalisation de cette thèse et son aboutissement. Tout comme les autres, tu étais là aux moments cruciaux pour me soutenir et aidé à surmonter les difficultés rencontrées.

Merci à toi Fabrice pour tes conseils pragmatiques basés sur ta longue expérience qui m'ont fait prendre conscience des différences majeures qui subsistent entre les travaux académiques et la R&D industrielle.

Enfin, un immense merci à Olivier pour ta gentillesse hors norme et ta bonne humeur quotidienne. Le résultat aurait très certainement été différent si tu n'avais pas été là à partager tous nos moments quotidiennement.

Un grand merci à mes collègues de l'IRT Christophe, Philippe P., Karim,

Jacques, Phillipe V., Tanissia, Ahmad, Laure, Michel et Jean-Michel pour les moments partagés et leur aide pendant toute la durée de mon travail. Merci à l'ensemble de l'équipe RMESS du laboratoire IRIT : André-Luc, Gentian, Riadh, Béatrice, Jérôme, Katia, Jullien, Jean-Luc et Sylvie.

Merci à mes collègues de bureau Justin et Romain, je retiendrais les parties de VR avec vous et les blagues totalement éclatées qui pouvaient venir ici et là pour oublier le syndrome de l'imposteur qui nous a frappé. Mention spéciale pour toi Justin et ton sauvetage de dernière minute. Merci à Kévin et Mohamed avec qui nous avons partagé de très bons repas et relevé quelques défis d'ingénierie sociale. Un grand merci à tous les doctorants Dorin, Oana, Firmin, Chaïma, Elene, Amal, Akash, Adrien et Chaïma qui ont réalisé leur thèse en parallèle et avec qui nous nous sommes serré les coudes. Je tiens aussi à remercier Louis et Théo que j'ai encadré pendant cette thèse et qui ont apporté leur pierre à l'édifice en me permettant de me conforter dans mes résultats.

Merci à toute ma famille Djamila, Houda, Khadija, Imenne et Ali pour votre soutien chacun à votre manière. Sans vous, il aurait été compliqué de tenir ces trois années. Merci à toi, Ibtissam pour ton soutien quotidien, sans toi, cette fin de thèse se serait révélée beaucoup plus difficile qu'elle ne l'a été. Enfin, je tiens à remercier l'ensemble de ma grande famille Kamel, Leïla, Idriss, Adam, Maryam, Mohamed M., Roufia, Ibrahim, Hafsa, Assiya, Youssouf ainsi que mes amis Loïc, Thomas, Mohamed S., Khalid, Hakim, Mezyan.

---

**Ce travail n'est pas seulement l'aboutissement de trois années de thèse, mais le résultat de bien plus longues années en amont, de ce fait, il convient aussi de remercier toutes les personnes non cités qui m'ont accompagnés durant tout mon parcours scolaire, personnel ou professionnel durant toutes ces années.**

---





# Table des matières

<b>Introduction</b>	<b>1</b>
<b>1 Introduction aux réseaux 5G et satellites</b>	<b>7</b>
1.1 Avancées technologiques qui permettent la 5G	8
1.1.1 Software Defined Networks (SDN)	8
1.1.2 Network Function Virtualization (NFV)	10
1.2 Les Réseaux de 5 <sup>ème</sup> génération	14
1.2.1 L'émergence de nouveaux services dans les réseaux	14
1.2.2 Les nouveautés de la 5G	15
1.2.3 La Service Based Architecture	15
1.2.4 L'accès radio en 5G	19
1.3 Les réseaux satellites	24
1.3.1 L'architecture broadband	24
1.3.2 Les évolutions technologiques des réseaux satellites	25
1.3.3 Les mega-constellations broadband	26
1.4 L'intégration des réseaux 5G et satellites	27
1.4.1 L'utilisation des réseaux satellites en réseau d'accès 5G	29
1.4.2 L'intégration du segment satellite dans les réseaux 5G	30
1.5 Conclusion	31
<b>2 Le Network Slicing</b>	<b>33</b>
2.1 Convergence vers le network slicing	34
2.1.1 Les nouveaux services dans les réseaux	34
2.1.2 Travaux préliminaires	35
2.1.3 Prise en compte du concept dans les différents organismes	35
2.1.4 Modèle de rôle	37
2.1.5 Architecture et Orchestration	39
2.1.6 Plan de contrôle	42
2.1.7 Plan de données	43
2.1.8 Documents de références	44
2.2 Caractéristiques d'un Network Slice	47
2.2.1 Propriétés des slices réseaux	47
2.2.2 Les défis associés	50
2.3 Conclusion	55
<b>3 Vers une architecture satellite supportant le network slicing</b>	<b>57</b>
3.1 Le network slicing pour les réseaux satellites	58
3.2 Allocation des ressources satellites	59
3.3 Mise en commun des infrastructures satellites	62
3.4 Une architecture satellite plus flexible	63



3.4.1	Gateway Satellite . . . . .	63
3.4.2	Satellite Core Network . . . . .	65
3.5	Gestion et orchestration du network slice satellite . . . . .	66
3.5.1	Modèle de rôle pour le network slicing satellite . . . . .	67
3.5.2	Satellite Management System . . . . .	68
3.5.3	Slice Management System . . . . .	70
3.6	Conclusion . . . . .	71
<b>4</b>	<b>Le network slicing pour le NTN dans la 5G</b>	<b>73</b>
4.1	Les grandes phases de l'intégration . . . . .	74
4.2	L'interface entre les systèmes de gestion . . . . .	74
4.3	Définition du niveau d'interconnexion entre les slices . . . . .	78
4.4	Continuité de slice et de QoS . . . . .	79
4.4.1	Modélisation de la continuité de slice et de QoS . . . . .	79
4.4.2	Le slice classifier dans le plan de donnée . . . . .	82
4.4.3	Échange dynamique de paramètres de QoS . . . . .	84
4.4.4	Architecture Fonctionnelle End-to-end . . . . .	85
4.5	Conclusion . . . . .	88
<b>5</b>	<b>La continuité de Slice et de QoS</b>	<b>91</b>
5.1	Réalisation d'une plateforme d'évaluation . . . . .	92
5.1.1	Nos besoins . . . . .	92
5.1.2	Les plateformes existantes . . . . .	92
5.1.3	Bilan et choix d'une plateforme . . . . .	94
5.2	Container Mesh for Network Slicing . . . . .	94
5.2.1	L'architecture de la plateforme . . . . .	94
5.2.2	Les éléments fonctionnels . . . . .	96
5.2.3	Une base pour nos expérimentations . . . . .	98
5.3	Évaluation de notre proposition "Slice Aware NTN" . . . . .	100
5.3.1	Scénarios . . . . .	100
5.3.2	Validation Expérimentale . . . . .	101
5.3.3	Évaluation des performances . . . . .	103
5.3.4	Conclusion . . . . .	108
	<b>Conclusion et Perspectives</b>	<b>109</b>
<b>A</b>	<b>Bandes de fréquences du 5G New Radio</b>	<b>112</b>
A.1	Frequency Range 2 $\geq$ 24 Ghz . . . . .	112
<b>B</b>	<b>Plateforme COMETS</b>	<b>113</b>
B.1	Fichier de configuration des scénarios . . . . .	113
B.2	Scénario générique . . . . .	114
B.3	Cycle de vie des scénarios . . . . .	117

---

<b>C</b>	<b>Serveurs d'applications IPerf</b>	<b>118</b>
C.1	Configuration des clients et serveurs IPerf . . . . .	118
<b>D</b>	<b>Slice Classifier</b>	<b>120</b>
D.1	Structures de données . . . . .	120
D.2	Configuration des files d'attentes . . . . .	121
D.3	Runtime . . . . .	122
D.3.1	Initialisation du Runtime . . . . .	122
D.3.2	Nouvelle Session PDU . . . . .	123
	<b>Définitions</b>	<b>125</b>
	<b>Acronymes</b>	<b>128</b>

# Table des figures

1.1	Architecture SDN [7]	8
1.2	De l'architecture réseau traditionnelle vers NFV	10
1.3	Framework ETSI NFV MANO	12
1.4	Architecture 4G avec CUPS [42]	16
1.5	Service Based Architecture 5G [3]	17
1.6	Séparation de la gNB en CU et DU [44]	20
1.7	Séparation de la pile protocolaire pour Cloud-RAN	21
1.8	Architecture Cloud-RAN	22
1.9	Architecture O-RAN	22
1.10	Architecture Satellite Broadband	24
1.11	Modes d'intégration des réseaux satellites dans la 5G [84]	27
2.1	Partition d'un réseau avec différents slices	34
2.2	Concept de Network Slicing présenté par le NGMN	36
2.3	Modèle de rôle dans la 5G [40]	38
2.4	Modèle relationnel du network slicing dans la 5G [103]	39
2.5	Relation entre le GSMA GST et le Service Profile [104]	40
2.6	Cycle de vie d'un slice défini par le 3GPP [40]	41
3.1	Slice satellite intégrée dans le réseau 5G	58
3.2	Réservation de ressources sur le segment satellite pour le backhau- ling 5G	60
3.3	Mutualisation des systèmes satellites pour découper le network sli- cing	62
3.4	Service Based Architecture-Gateway	64
3.5	Séparation du réseau satellite entre le satellite core and radio networks	66
3.6	Modèle de rôle du network slicing satellite	67
3.7	Satellite Management System	68
3.8	Étapes d'instanciation du network slice satellite	69
3.9	Slice Management System	71
4.1	Phases d'instanciation d'un network slice end-to-end	74
4.2	Interfaces de gestion du slice satellite	75
4.3	5G end-to-end slice avec un Non Terrestrial Network (NTN) slice aware en tant que Transport Network (TN)	76
4.4	Niveau d'interconnexion entre les réseaux satellite et 5G	78
4.5	Modèle Slice end-to-end	82
4.6	Traitement des flux dans le Slice Classifier	83
4.7	Négociation dynamique des paramètres de QoS	84
4.8	Architecture Fonctionnelle End-to-end	86

---

4.9	PDU Session Establishment Procedure . . . . .	87
4.10	Étapes pour le slicing du réseau satellite et l'intégration dans la 5G . . . . .	89
5.1	Architecture de COMETS . . . . .	95
5.2	Architecture fonctionnelle des scénarios . . . . .	99
5.3	Débit par QoS Flow ID (QFI) et par slice . . . . .	104
5.4	Packet Error Rate par QFI et par slice . . . . .	105
5.5	Packet Delay Budget par QFI et par slice . . . . .	106
5.6	Gigue par QFI et par slice . . . . .	107

# Liste des tableaux

1.1	Correspondance entre les fonctions 4G et les fonctions 5G . . . . .	18
1.2	Mega-constellations LEO (basé sur [80] et enrichi jusqu'à Juillet 2022)	26
1.3	Les différents modes d'intégration [84] . . . . .	28
2.1	Organisme travaillant sur le network slicing . . . . .	36
2.2	Les slices standard de la 5G et les services associés . . . . .	43
2.3	Documents de standardisation - Network Slicing . . . . .	44
2.4	Projets 5G traitant du network slicing . . . . .	47
2.5	Défis associés au Network Slicing . . . . .	56
3.1	Catégorie des défis pour le slicing du réseau satellite . . . . .	59
3.2	Défis du network slicing appliqués aux satellites et nos propositions associées . . . . .	71
4.1	Modèle de Network Slice End-to-end . . . . .	76
4.2	Modèles des User Equipment (UE) et PDU . . . . .	80
4.3	Modèle des flux . . . . .	81
4.4	Enjeux du network slicing appliqués aux satellites et propositions associées . . . . .	88
5.1	Définition du scénario . . . . .	100
5.2	Propositions . . . . .	108
A.1	Fréquence 5G FR2 [194] . . . . .	112

# Liste des Implantations

1	UE se raccordant au réseau et établissant des sessions PDUs . . . . .	102
2	Établissement des sessions PDUs par la SMF . . . . .	103
3	Définition des scénarios . . . . .	114
4	Scénario générique COMETS . . . . .	117
5	Cycle de vie des scénarios COMETS . . . . .	117
6	Configuration des applications IPerf . . . . .	119
7	Structures de données . . . . .	121
8	Configuration des files d'attente . . . . .	122
9	Initialisation du Runtime . . . . .	123
10	Nouvelle session PDU . . . . .	124



# Introduction

Les réseaux de télécommunications n'ont eu cesse d'évoluer pour s'adapter aux besoins des sociétés. Aujourd'hui, l'usage que font les utilisateurs des réseaux a évolué et radicalement changé. Une multitude de services hétérogènes [1] sont apparus, tels que les services de streaming vidéo (e.g. Netflix, HULU, YouTube), les réseaux sociaux (e.g. Facebook, Instagram, TikTok), l'e-commerce (e.g. Amazon, Ebay), les systèmes de gouvernances (e.g. votes électroniques, e-citizenship), la messagerie instantanée (e.g. Whatsapp, Telegram, Signal) et des services permettant le travail collaboratif à distance (e.g. Zoom, Microsoft Teams, Google Meet). Chacun de ces services possède ses propres exigences en termes de bande passante, de capacité, de latence et nécessite ses propres applications en plus de la connectivité fournie par le réseau pour y accéder. Pour certains services tels que le streaming vidéo et les outils collaboratifs, une coopération entre le fournisseur de service et l'opérateur réseau est même nécessaire afin de pouvoir déployer un service optimal.

C'est pourquoi, au cours des dernières décennies, les réseaux ont évolué pour accroître leurs performances, augmenter leurs capacités, réduire les latences et optimiser la gestion de leurs ressources afin de rester compétitifs et s'adapter au marché. Néanmoins, ces améliorations à elles seules ne suffisent plus pour permettre le déploiement de nouveaux services de plus en plus hétérogènes. En effet, à hauteur de 2023, Cisco prévoit environ 30 milliards d'équipements connectés au réseau Internet [2] utilisant des millions de services différents. Cela occasionnera une augmentation considérable de la bande passante que les réseaux actuels ne peuvent supporter.

De plus, avec l'arrivée de services encore plus contraints tels que la télé-médecine, le Vehicle-to-Everything (V2X), l'industrie 4.0, le Massive Internet of Things (MIoT), il est nécessaire de repenser totalement l'architecture de déploiement des réseaux afin d'intégrer ces services et d'en permettre le passage à l'échelle. La construction d'une infrastructure dédiée à chaque service peut être envisagée, mais elle serait très coûteuse et presque impossible à gérer au vu du nombre important de services qu'il faudrait déployer dans le réseau additionné à la cascade d'acteurs impliqués dans ces services.

C'est ce qui a conduit les opérateurs réseaux à adopter le paradigme de network slicing au sein des nouveaux réseaux mobiles de 5<sup>ème</sup> génération. Le network slicing est un concept réseau dans lequel on crée de multiples partitions dans le réseau nommées "network slice", chaque partition étant dédiée à un service particulier. Ces slices sont instanciés au dessus d'une seule et même infrastructure physique, et possèdent une totale autonomie au sein du réseau ainsi que leur propre architecture. Ils sont orchestrés, déployés et s'étendent sur tous les segments réseaux allant de l'utilisateur jusqu'au serveur applicatif de manière à répondre aux exigences de services de bout-en-bout (end-to-end).



Au même moment, le support des Non Terrestrial Network (NTN) (et donc en particulier les réseaux satellites) dans la release 15 du standard 3rd Generation Partnership Project (3GPP) [3] ouvre la voie à une intégration complète des deux réseaux. En effet, les avancées technologiques ont permis ces dernières années le déploiement de mega-constellations satellites en orbite basse qui permettent d'atteindre des performances inédites. Les réseaux satellites, avec leurs performances et leurs zones de couverture mondiale, peuvent donc être aujourd'hui considérés comme des accélérateurs de déploiement pour les réseaux mobiles.

Le network slicing ouvre de nouvelles possibilités, vise à fournir des services de bout en bout dans plusieurs domaines administratifs réseaux et est donc un candidat pour assurer une telle intégration. C'est dans ce cadre que s'inscrivent les travaux de recherche décrits dans ce manuscrit.

## Contexte et Problématique de la thèse

### Cadre de la thèse

Cette thèse s'inscrit dans le projet de recherche collaboratif SUPER-G effectué au sein de l'IRT Saint Exupéry initié en 2019. Ce projet regroupe 4 entreprises du spatial (Airbus Defence and Space, Thales Alenia Space, Viveris, QoS Design), 2 laboratoires de recherche (IRIT, LAAS), la fondation de recherche i2CAT ainsi que le centre national d'études spatiales qui collaborent afin d'aboutir à une intégration réussie des réseaux satellites et 5G. Il succède aux projets VITAL et SaT5G qui avaient pour but d'étudier la virtualisation des réseaux satellites et, dans une première phase, l'intégration des réseaux satellites dans la nouvelle architecture 5G. L'objectif de ce projet est de proposer des méthodes d'intégration du réseau satellite au sein des réseaux 5G afin d'en accélérer le déploiement et l'adoption, développer des plateformes d'expérimentation flexibles qui permettent de déployer divers scénarios et d'en évaluer les performances, ainsi que préparer la transition vers les réseaux 5,5G et 6G.

### Problématique de la thèse

L'intégration des réseaux satellites dans les réseaux mobiles est un processus complexe qui nécessite des pré-requis au sein des deux familles de réseaux. Ces pré-requis ne sont aujourd'hui pas implantés tels que la virtualisation au sein des réseaux satellites ainsi que leur architecture qui n'est pas adaptée à la dynamique des réseaux 5G. Les méthodes d'interconnexion aux niveaux architectural, protocolaire et de gestion ne sont pas définies.

Notre travail s'intéresse à l'utilisation du network slicing comme paradigme d'intégration des deux réseaux. Les techniques Software Defined Network (SDN) et Network Function Virtualization (NFV) ne suffisent pas à elle seules à dynamiser les liens satellites pour permettre une intégration complète de ces deux types

de réseaux. L'architecture satellite doit aussi évoluer afin d'introduire des points d'interconnexions avec les réseaux 5G.

Afin de réaliser cette interconnexion un ensemble de verrous technologiques doivent être levés :

- L'architecture satellite actuelle peu flexible qui ne supporte que partiellement les technologies SDN et NFV ;
- la ségrégation qui existe entre les différentes infrastructures satellites fournissant des services équivalents ou différents ;
- la complexité d'allocation des ressources dans les réseaux satellites qui se matérialise par un manque de dynamisme et qui empêche le passage à l'échelle des services 5G ;
- l'absence d'une adoption du mécanisme de network slicing dans les réseaux satellites ;
- l'absence de mécanismes permettant la continuité de service et de Quality of Service (QoS) entre les réseaux satellites et 5G dans le cas d'une interconnexion.

Nos travaux ont pour ambition d'apporter des réponses à ces défis.

## Contributions

Nous présentons dans cette section les contributions qui résultent de notre travail de thèse :

- **Revue du paradigme de network slicing** : Dans cette contribution nous effectuons une revue complète du paradigme de network slicing, de son potentiel, des problèmes qu'il essaie de résoudre, de son développement actuel, de son adoption dans les réseaux terrestres et des défis qu'il faut relever pour le déployer.
- **Architecture satellite supportant le network slicing** : Cette contribution applique le paradigme de network slicing aux réseaux satellites, liste les différents défis qu'il faut relever afin de pouvoir bénéficier des avantages de ce concept et propose des solutions pour y arriver. Au sein de cette contribution, nous définissons une architecture satellite flexible capable de supporter les différents aspects du slicing et permettre une intégration réelle dans les réseaux 5G.
- **Intégration dans les réseaux 5G en utilisant le network slicing** : Dans cette partie, nous proposons une méthode d'intégration des réseaux satellites dans les réseaux 5G en tant que TN (réseau de transport) supportant le network slicing. Nous définissons une architecture d'interconnexion qui permet à un opérateur mobile d'utiliser un segment satellite pour relayer les données de ses network slices de manière automatisée tout en respectant les Key Performance Indicator (KPIs) des services au sein des network slices et la continuité de QoS. Pour cela, nous introduisons des procédures et fonctions réseaux dans le plan de contrôle 5G et satellite, développons

une fonction réseau d'interconnexion dans le plan de données entre les réseaux 5G et satellites.

- **Développement d'une plateforme d'expérimentation dédiée à l'intégration du réseau satellite dans les réseaux 5G :** Dans cette dernière contribution, nous développons une plateforme d'expérimentation extensible open-source pour l'intégration du lien satellite dans la 5G supportant le network slicing. Nous implantons nos précédentes contributions sur cette plateforme et validons nos propositions.

## Plan du manuscrit

Cette section présente le plan du manuscrit qui, suite à cette introduction s'articule en 5 chapitres et une conclusion générale.

- **Chapitre 1 :** Dans ce chapitre, nous effectuons un état de l'art des réseaux mobiles terrestres, des réseaux satellites et des différentes techniques d'intégration. Nous définissons la problématique de notre thèse et les éléments auxquels nous allons apporter des réponses. Nous présentons l'évolution des réseaux 4G, des différentes améliorations qui ont été apportées et qui ont amené aux réseaux mobiles de 5<sup>ème</sup> génération. Nous présentons les réseaux 5G, les innovations technologiques qu'elles apportent et les briques technologiques sur lesquelles ils se reposent. Nous présentons de même les réseaux satellites, les innovations majeures qu'ils ont subi ces dernières années, les mega-constellations et leur niveau d'intégration actuel dans les réseaux mobiles.
- **Chapitre 2 :** Ce chapitre présente le network slicing dans sa globalité, il liste l'ensemble des contributions qui sont liées à ce paradigme, les objectifs qu'il veut atteindre, les différentes définitions qui ont été proposées pour ce concept, les caractéristiques communes à ces définitions, les défis qu'il amène, les architectures existantes et son déploiement actuel.
- **Chapitre 3 :** Ce chapitre présente notre contribution sur l'architecture satellite supportant le network slicing. Il inclut une infrastructure satellite mutualisant l'ensemble des infrastructures existantes actuelles, une gateway satellite désagrégée, un système d'orchestration de slices satellite et un système de gestion pour l'ensemble du réseau satellite.
- **Chapitre 4 :** Ce chapitre présente notre contribution sur l'intégration du lien satellite dans les réseaux 5G. Nous listons l'ensemble des étapes nécessaires pour réaliser cette intégration et détaillons chacune d'elles. Nous présentons notre schéma de négociation dynamique de QoS, les modifications à apporter au cœur de réseau 5G, les différentes fonctions réseaux que nous ajoutons dans les réseaux satellites et 5G, les fonctions réseaux qui servent de points d'interconnexion entre les réseaux ainsi que les mécanismes qui permettent de garantir une continuité de service et de QoS au sein des deux réseaux.

- **Chapitre 5 :** Au sein de ce chapitre, nous présentons notre plateforme d'expérimentation basée sur les conteneurs sur laquelle nous avons validé nos précédentes contributions. Nous effectuons une revue des plateformes existantes à ce jour, leurs faiblesses et les motivations qui nous ont poussé à définir notre plateforme. Nous présentons ses avantages, son extensibilité et l'implantation que nous avons utilisé pour nos scénarios de test.
- **Conclusion et Perspective :** Ce chapitre donne un résumé des travaux réalisés lors de notre thèse, ainsi que les perspectives d'améliorations et d'évolutions sur lesquelles nous travaillons d'ores et déjà et celles qui peuvent être envisagées à plus long terme.



# Introduction aux réseaux 5G et satellites

---

## Sommaire

---

<b>1.1</b>	<b>Avancées technologiques qui permettent la 5G</b>	<b>8</b>
1.1.1	Software Defined Networks (SDN)	8
1.1.2	Network Function Virtualization (NFV)	10
<b>1.2</b>	<b>Les Réseaux de 5<sup>ème</sup> génération</b>	<b>14</b>
1.2.1	L'émergence de nouveaux services dans les réseaux	14
1.2.2	Les nouveautés de la 5G	15
1.2.3	La Service Based Architecture	15
1.2.4	L'accès radio en 5G	19
<b>1.3</b>	<b>Les réseaux satellites</b>	<b>24</b>
1.3.1	L'architecture broadband	24
1.3.2	Les évolutions technologiques des réseaux satellites	25
1.3.3	Les mega-constellations broadband	26
<b>1.4</b>	<b>L'intégration des réseaux 5G et satellites</b>	<b>27</b>
1.4.1	L'utilisation des réseaux satellites en réseau d'accès 5G	29
1.4.2	L'intégration du segment satellite dans les réseaux 5G	30
<b>1.5</b>	<b>Conclusion</b>	<b>31</b>

---

Ce chapitre introduit les différentes technologies qui sont employées dans le monde des réseaux mobiles de 5<sup>ème</sup> génération et satellites. Nous présentons les évolutions des réseaux 4G vers les réseaux 5G, leur apport et les problématiques auxquelles ils répondent. Puis nous présentons les réseaux satellites broadband géostationnaires, leurs cas d'usages, les technologies qu'ils utilisent ainsi que leurs limitations. Enfin, nous effectuons une revue des nouvelles mega-constellations broadband en orbite basse en cours de déploiements, leurs avantages et inconvénients comparés aux satellite géostationnaires ainsi que les opportunités qu'elles ouvrent.

## 1.1 Avancées technologiques qui permettent la 5G

Le déploiement des réseaux 5G est aujourd'hui rendu possible grâce à l'ensemble des avancées scientifiques et technologiques de ces dernières années dans les réseaux en matière de qualité de service, de virtualisation, de gestion et d'orchestration. Ces avancées constituent des composants de base nécessaires au bon développement de la 5G. Nous présentons les technologies SDN dans le paragraphe 1.1.1 ainsi que NFV dans le paragraphe 1.1.2.

### 1.1.1 Software Defined Networks (SDN)

L'un des sujets réseau majeur de ces dernières années fut le développement de la programmabilité des réseaux afin d'augmenter la flexibilité dont dispose un opérateur réseau ainsi que simplifier la gestion des réseaux [4]. L'Internet Engineering Task Force (IETF) proposa une première approche pour séparer le plan de contrôle et le plan de donnée dans les réseaux avec le framework Forwarding and Control Element Separation (ForCES) [5], cette initiative fut poursuivie par l'ITU Telecommunication Standardization Sector (ITU-T) [6] pour les réseaux 5G et l'Open Networking Foundation (ONF) qui proposa l'architecture SDN [7] devenue aujourd'hui l'architecture de référence.

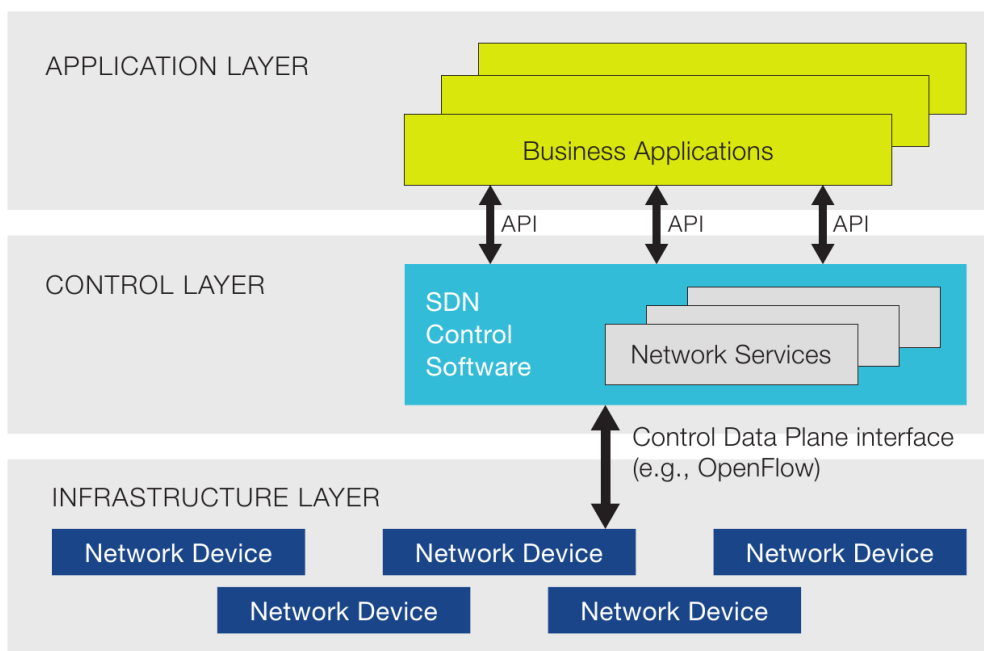


FIGURE 1.1 – Architecture SDN [7]

SDN introduit une architecture similaire à ForCES dans laquelle le plan de contrôle et le plan de données sont séparés au niveau fonctionnel et implantés sur des équipements différents. La figure 1.1 montre l'architecture SDN. Le plan de

contrôle est programmé via une Application Programming Interface (API) nord et va ensuite aller configurer des “switchs SDN” dans le plan de données via une API sud qui utilise des protocoles standardisés tels qu’OpenFlow [8] ou Network Configuration Protocol (NETCONF) [9]. La séparation des plans de contrôle et de données permet une centralisation du plan de contrôle sur un contrôleur SDN unique (e.g. Ryu [10], OpenDaylight [11]) qui embarque les applications réseaux et possède donc une supervision globale du réseau. Les réseaux SDN possèdent trois modes de fonctionnement :

- le mode réactif qui lui permet de reprogrammer le plan de donnée en fonction du trafic réel et d’ajuster les règles de gestion de flux du réseau. Ce mode permet un fonctionnement optimal en ne programmant que les règles nécessaires au trafic réseau, néanmoins il ne permet pas un passage à l’échelle dans les réseaux possédant une quantité de trafic importante. En effet, la communication entre les contrôleurs SDN et les switchs excède très largement la durée de traitement des paquets au sein des switchs ce qui occasionne des problèmes de congestions et donc de performances. Ce mode de fonctionnement est aussi particulièrement sensible aux attaques réseaux qui peuvent cibler les contrôleurs lorsque les switchs lui envoient le trafic [12] (e.g. déni de service par saturation des contrôleurs SDN).
- le mode proactif où le contrôleur pré-programme tous les switchs afin qu’ils soient en mesure de traiter le trafic sans avoir à communiquer avec ce dernier. Les communications ne se font que du contrôleur vers les switchs, ce qui rend la politique de gestion du réseau robuste, réduit les vecteurs d’attaques et permet un passage à l’échelle. Cependant, il y a une perte de flexibilité dans la gestion du réseau puisqu’aucune action ne peut être déclenchée en fonction du trafic en temps réel dans le réseau. Cela ne permet donc pas un fonctionnement optimal de ce dernier et peut aussi aboutir à une sous-utilisation du réseau ou à l’inverse une incapacité à réagir en cas d’événements.
- le mode prédictif qui est une amélioration du mode proactif où l’on utilise un ensemble de données d’historique sur le trafic réseau pour effectuer des prédictions futures. Ces prédictions sont utilisées par le contrôleur pour reprogrammer le plan de donnée périodiquement afin d’assurer un traitement optimal des paquets. Le mode prédictif peut aussi être utilisé avec des techniques plus avancées de machine learning qui permettent au réseau de s’auto-configurer ainsi que de s’auto-gérer.

Les architectures SDN ont à ce jour un large domaine d’application [13-16] et permettent de répondre aux besoins de plusieurs services tels que les Content Delivery Network (CDN) [17] ou le streaming vidéo [18]. SDN a atteint un niveau de maturité suffisant pour être actuellement déployé au niveau industriel sur des applications bien définies [19] et même sur des réseaux longues distances comme le SDN-WAN B4 de Google [20]. SDN constitue une avancée majeure dans la programmation des réseaux et est une brique technologique fondamentale sur laquelle repose la 5G.



## 1.1.2 Network Function Virtualization (NFV)

### 1.1.2.1 Introduction à la virtualisation de fonction réseau

En parallèle des développements qui conduisirent la communauté à définir SDN, la virtualisation des réseaux s'opérait. NFV s'inspire du concept de la virtualisation des systèmes informatiques utilisés à large échelle dans les datacenters pour étendre cette virtualisation aux équipements réseaux. Avec ce concept, il est possible de découpler le matériel physique et les couches logicielles; cela permet de développer des fonctions réseau implantables sur n'importe quel matériel générique Commercial off-the-shelf (COTS). Cela rend aussi utilisable les datacenters existants comme base d'infrastructure pour NFV.

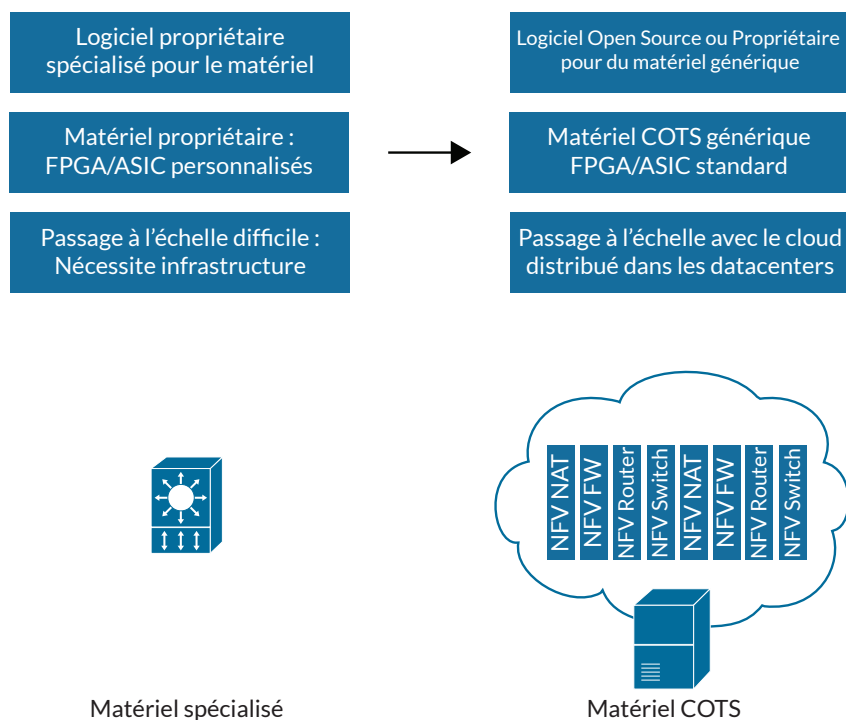


FIGURE 1.2 – De l'architecture réseau traditionnelle vers NFV

Dans les équipements réseaux traditionnels, les équipementiers (e.g. Cisco, Juniper) ne se souciaient pas de la couche matérielle sur laquelle leur logiciel allait s'exécuter puisqu'ils développaient eux-mêmes leur matériel spécialisé (voir figure 1.2). Ils ont un contrôle total du matériel et du logiciel ce qui leur laisse le choix de développer du matériel bien spécifique pour la fonction réseau à effectuer (e.g. un routeur de réseau de cœur n'offrira pas les mêmes garanties qu'un routeur de bordure). Avec NFV il est impossible de prévoir sur quel matériel la couche logicielle va s'exécuter et donc de développer une couche logicielle spécifique. Ceci est un changement majeur avec l'arrivée de NFV offrant alors ainsi la possibilité d'exécuter des fonctions réseaux sur du matériel générique et non plus de constructeurs spécifiques assurant ainsi une portabilité des fonctions réseaux.

Cela amena l'European Telecommunications Standards Institute (ETSI) à développer un framework d'orchestration ETSI NFV MANO [21] afin d'homogénéiser la gestion des réseaux virtualisés et de définir une architecture fonctionnelle de référence commune à toutes les solutions d'orchestration. De plus, SDN et NFV sont intrinsèquement liés car le chaînage des flux entre les fonctions se repose sur la dynamique offerte par SDN.

### 1.1.2.2 Framework ETSI NFV MANO

L'architecture des équipements qui régit les réseaux traditionnels est simple du fait de la dépendance de la couche logicielle avec le matériel. Avec NFV, les fonctions réseaux sont découplées du matériel ce qui offre beaucoup de solutions pour leur mise en œuvre. On appelle Virtual Network Function (VNF) dans NFV une fonction réseau virtualisée. Pour construire une topologie réseau, il faut un ensemble de virtual network functions (VNFs) et donc aussi un système de management pour gérer les ressources physiques et logicielles pour le bon fonctionnement du réseau virtuel. De plus un opérateur peut décider d'utiliser des VNFs de fournisseurs différents tout en gardant un système de gestion unique. NFV apporte donc à son tour de nouveaux défis :

- implantation de VNFs différentes selon les fournisseurs;
- gestion du cycle de vie des VNFs;
- gestion de l'interaction entre les différentes VNFs de différents fournisseurs;
- gestion des ressources physiques de matériel générique;
- supervision de l'utilisation des ressources et les VNFs;
- configuration des VNFs;
- mise en place d'un système de facturation.

Afin de répondre à ces problématiques tout en gardant un système ouvert et non-propriétaire, il est nécessaire de définir un Framework et de le soumettre à une standardisation. Ce framework doit garantir une totale indépendance entre les couches matérielles et logicielles, constituer une référence pour les équipementiers et leur proposer une méthode de développement de VNF. C'est ainsi que l'European Telecommunications Standards Institute (ETSI) forma un groupe de travail en 2013 pour développer le framework ETSI NFV MANO [22, 23] (figure 1.3) ayant pour objectif de proposer une architecture de référence pour NFV. Le groupe de travail de l'ETSI l'a développé en se basant sur trois critères :

- indépendance entre les couches physiques et logicielles;
- flexibilité qui permet une automatisation et un passage à l'échelle du réseau;
- dynamique qui permet une modification de la topologie du réseau en production.

Cette architecture est composée de plusieurs blocs principaux qui interagissent entre eux via des interfaces standardisées. On retrouve les blocs suivants dans l'architecture ETSI NFV MANO :

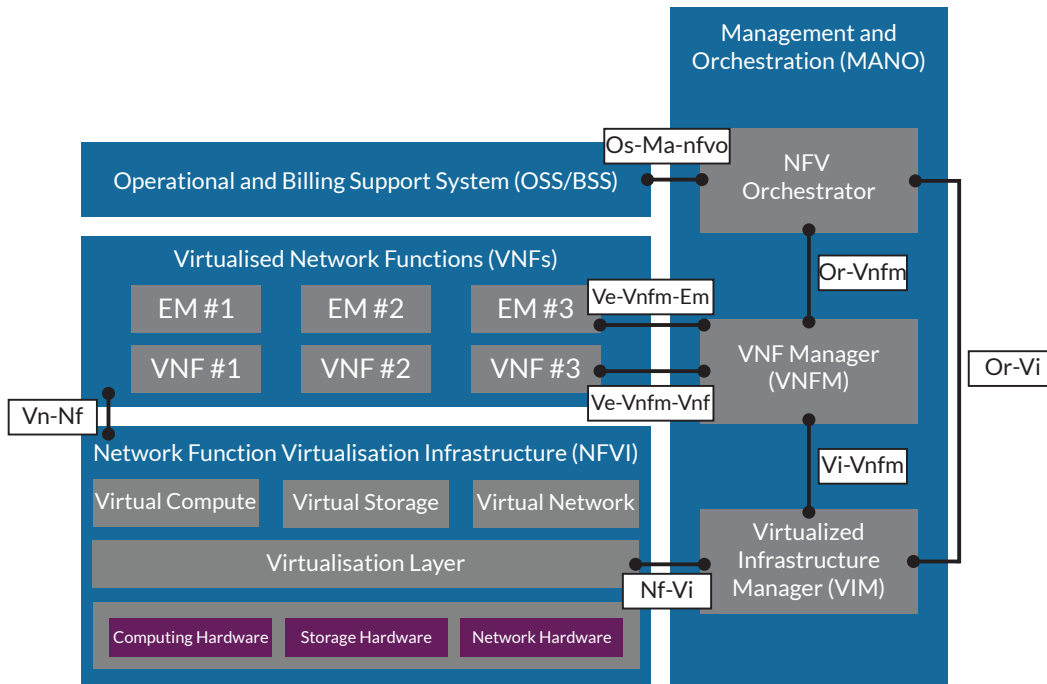


FIGURE 1.3 – Framework ETSI NFV MANO

- **Network Function Virtualisation Infrastructure (NFVI) :** c'est le bloc qui représente l'infrastructure sur laquelle les VNFs vont s'exécuter. Il comprend les ressources virtuelles, la couche de virtualisation et les ressources physiques. On retrouve dans la couche ressources physiques l'ensemble du matériel générique disponible avec ses composants (CPU, GPU, ASIC, FPGA, NIC, RAM, HDD, etc...), ces ressources peuvent être localisées dans un seul lieu physique ou alors réparties à travers plusieurs points physiques (dans des datacenters que l'on appelle plus précisément dans notre cas d'application NFV-POP). On distingue dans les ressources physiques les systèmes de stockages (e.g NAS, SAN ou distribués), les ressources réseaux qui comprennent l'ensemble des cartes réseaux disponibles ainsi que les ports qui leurs sont associés (e.g paire de cuivre, optique ou sans-fil) et enfin la puissance de calcul qui comprend les processeurs et la mémoire vive. Au niveau supérieur, on retrouve la couche de virtualisation qui va permettre d'abstraire les ressources physiques aux VNFs qui auront accès à des ressources virtuelles répondant à leurs exigences. Une VNF sera en mesure d'utiliser les ressources virtuelles par le biais d'une machine virtuelle ou d'un conteneur.
- **Virtual Network Function (VNF) :** On y retrouve les VNFs qui sont des machines virtuelles ou des conteneurs. Pour chaque VNF, il peut exister un **Element Management (EM)** qui est responsable des fonctions **Fault, Configuration, Accounting, Performance and Security (FCAPS)** de la VNF. La granularité d'une VNF n'est pas définie par le framework, une VNF est

seulement définie comme étant une fonction réseau virtualisée. On peut citer des fonctions réseaux connues comme les firewalls, NAT ou encore Domain Name System (DNS).

- Operations Support System (OSS)/Business Support System (BSS) : C'est le système qui assure la relation avec les clients, il contient le catalogue des produits, un système de gestion CRM, les assurances, les exigences des contrats et un système de facturation. Il peut aussi embarquer d'autres fonctions liées à la gestion opérationnelle de l'opérateur.
- Management and Orchestration (MANO) : Le rôle principal de ce bloc est la gestion et l'orchestration de l'écosystème NFV. On y retrouve le NFV Orchestrator (NFVO) qui est responsable de la gestion du cycle de vie des Network Services (NS) (création, modification, destruction), des échanges d'informations concernant le cycle de vie des NS avec le système OSS/BSS, de la gestion des ressources physiques en échangeant des informations avec le Virtualized Infrastructure Manager (VIM) et envoie des notifications sur les ressources disponibles au VNF Manager (VNFM). Le VNFM est l'élément qui s'occupe de la gestion du cycle de vie et la configuration initiale des VNFs et EM. Il permet aussi de déléguer la gestion et la supervision des VNFs au NFVO, EM et VNF elle-même. Enfin, il peut aussi jouer le rôle de gestionnaire des ressources virtualisées en communiquant avec le VIM. Le dernier élément du bloc MANO est le VIM qui est responsable de la gestion des ressources de l'infrastructure NFVI qui peuvent être localisées dans un ou plusieurs NFVI-POPs. Il est responsable de la virtualisation des ressources au sein de l'infrastructure NFVI par le biais de la couche de virtualisation. Comme précédemment, il peut déléguer la gestion des ressources virtualisées au VNFM et NFVO.

En plus de l'architecture fonctionnelle, le framework ETSI NFV MANO définit les processus d'orchestration au sein de l'écosystème NFV [24], l'orchestration multi-domaine [25], son intégration avec le paradigme SDN [26] et le support du network slicing [27].

## 1.2 Les Réseaux de 5<sup>ème</sup> génération

Dans cette section, nous présentons les réseaux mobiles 5G dans leur globalité, les objectifs scientifiques et techniques qu'ils visent ainsi que les innovations qu'ils apportent. Nous présentons dans la section 1.2.2 les nouveautés apportées par la 5G, dans la section 1.2.3 l'architecture des réseaux 5G et dans la section 1.2.4 le Radio Access Network (RAN) 5G. Une attention particulière est donnée au standard spécifié par le 3GPP, l'organisme responsable de la standardisation des réseaux mobiles actuels.

### 1.2.1 L'émergence de nouveaux services dans les réseaux

Avec l'évolution des usages des réseaux de télécommunications par les utilisateurs, la gestion des réseaux et leur objectif principal se sont retrouvés modifiés. L'objectif principal des premiers réseaux longue distance tels qu'ARPANET (1969) [28] et X.25 (1976) [29] fut de fournir une connectivité entre les utilisateurs et un accès à un nombre limité de services. Les services qu'offraient ces réseaux furent dans un premier temps l'échange de messages textuels notamment avec la création des protocoles e-mails (1973) [30] et l'échange de fichiers peu volumineux avec l'utilisation de protocoles tel que FTP (1971) [31]. En raison de leur nouveauté et des coûts de raccordement et des services ciblés, les usagers très largement majoritaires de ces réseaux étaient les organismes gouvernementaux, les universités et les entreprises. Mais ce n'est qu'avec l'arrivée des réseaux longue distance décentralisés tels que Internet Protocol (IP) (1980) [32], les réseaux mobiles de seconde génération (1991) Global System for Mobile Communications (GSM) [33] et le développement massif de l'Internet au niveau mondial que les usages changèrent considérablement et qu'une multitude de nouveaux services fut déployée au sein des réseaux. Depuis, les réseaux fixes et mobiles n'ont cessé d'augmenter leurs capacités, leurs performances et le nombre d'équipements qui peuvent s'y connecter simultanément. Chacune des améliorations technologiques qui a été introduite que ce soit pour les réseaux fixes (e.g. Asymmetric Digital Subscriber Line (ADSL) (1988) [34], Fiber to the everything (FTTx) [35]) ou mobiles fut dans ce sens. Néanmoins, avec la multiplication de services hétérogènes déployés (e.g. Streaming vidéo, Internet of Things (IoT)), augmenter les performances des réseaux ne suffit plus. Il est désormais nécessaire de repenser le déploiement et l'orchestration des réseaux afin de permettre la coexistence optimale de ces services hétérogènes qui nécessitent parfois des architectures, des technologies et des performances totalement disjointes. C'est précisément l'un des objectifs des réseaux mobiles de 5<sup>ème</sup> génération en cours de déploiement et de standardisation qui intègrent en leur sein le concept de "network slicing". Ces réseaux de nouvelle génération répondent aux recommandations de l'ITU Radiocommunication Sector (ITU-R) IMT-2020 [36-39] et souhaitent pouvoir répondre aux besoins des marchés verticaux en étendant la notion de réseaux mobile notamment dans l'industrie 4.0.

### 1.2.2 Les nouveautés de la 5G

Les réseaux mobiles ont évolué d'un paradigme permettant d'apporter une connectivité aux utilisateurs vers un modèle de déploiement de services end-to-end à la demande. Il fut ainsi nécessaire de repenser en profondeur les réseaux mobiles afin qu'ils puissent intégrer cette notion de service et fournisseur de service. Les réseaux mobiles ont ainsi évolués avec l'arrivée de la 5<sup>ème</sup> génération avec pour objectif premier d'homogénéiser les briques technologiques afin de permettre le déploiement de services hétérogènes sur une infrastructure réseau unifiée et mutualisée. Ces réseaux ont été introduits avec la Release 15 définie par l'organisme 3GPP [40]. Des changements majeurs ont été apportés à l'architecture globale, un nouveau modèle de rôle et un nouveau framework de management orienté service appelé Service Based Management Architecture (SBMA) ont été introduits. Un nouveau Core Network (CN) a été proposé qui s'appuie sur les technologies cloud et s'oriente autour de la Service Based Architecture, un nouveau RAN a été proposé et intégré avec le New Radio (NR) et les réseaux 5G intègrent désormais le paradigme de network slicing. Les notions de virtualisation de CN et RAN, initiés au sein de la 5G, ont été définitivement entérinés dans la 5G avec l'adoption de SDN et NFV. Il existe actuellement deux types de déploiements pour les réseaux 5G : le mode Standalone (SA) où l'on déploie un CN et un RAN 5G complété par le mode Non Standalone (NSA) où l'on déploie un CN 4G auquel on adjoint un RAN 5G. Le mode NSA est préféré pour les déploiements réels en première instance ce qui permet de remplacer les Evolved Node B (eNBs) par des next generation Node B (gNBs) et donc de faire la mise à jour du RAN puis en seconde instance l'introduction du mode SA qui permet de pleinement exploiter les capacités des réseaux 5G. Effectivement, le mode NSA ne permet pas de pleinement bénéficier des avancées technologiques qu'a subit le CN 5G ainsi que les nombreuses fonctionnalités qu'il offre en terme d'orchestration ou de services comme par exemple le déploiement et l'utilisation du network slicing.

### 1.2.3 La Service Based Architecture

La Release 8 du 3GPP a introduit Evolved Packet Core (EPC) [41] comme CN pour la 4G. L'EPC est composé de :

- Mobility Management Entity (MME) : Élément du plan de contrôle qui possède des fonctions de traitement du plan de signalisation de la 4G au travers de l'interface S1-MME ;
- Serving Gateway (SGW) : Élément du plan de contrôle et du plan de données qui est responsable de l'interconnexion avec le RAN 4G Evolved Universal Terrestrial Radio Access (E-UTRA) au travers de l'interface S1-U. Il Possède des fonctions de relayage ainsi que de traitement de paquets ;
- PDN Gateway (PGW) : Élément du plan de contrôle et du plan de données qui est responsable de l'interconnexion avec le Packet Data Network (PDN) au travers de l'interface SGi ;

- Home Subscriber Server (HSS) : Serveur de base de données qui contient toutes les informations relatives aux usagers du réseau. Il communique avec le MME au travers de l'interface S6a;
- Policy and Charging Rules Function (PCRF) : Fonction responsable de la politique de QoS du réseau et de la facturation;
- PDN : Le réseau de donnée externe inter-connecté au réseau 4G au travers de l'interface SGi.

Lors du déploiement des réseaux 4G, cette architecture répondait aux besoins mais est aujourd'hui trop coûteuse à maintenir et trop complexe à gérer. En effet, les équipements du CN suivent un paradigme centralisé, ce qui augmente considérablement le coût lors des évolutions du réseau ou lors du passage à l'échelle. Le plan de contrôle et le plan de données des Gateways (GWs) sont sur les mêmes équipements ce qui ne permet pas un passage à l'échelle optimal du réseau et une complexité de gestion importante. C'est pourquoi le 3GPP a introduit la séparation du plan de contrôle et du plan de données au sein du CN 4G grâce à l'amélioration Control and User Plane Separation (CUPS) [42] au sein de la Release 14 (Figure 1.4).

#### Fonctions réseaux 4G avec CUPS

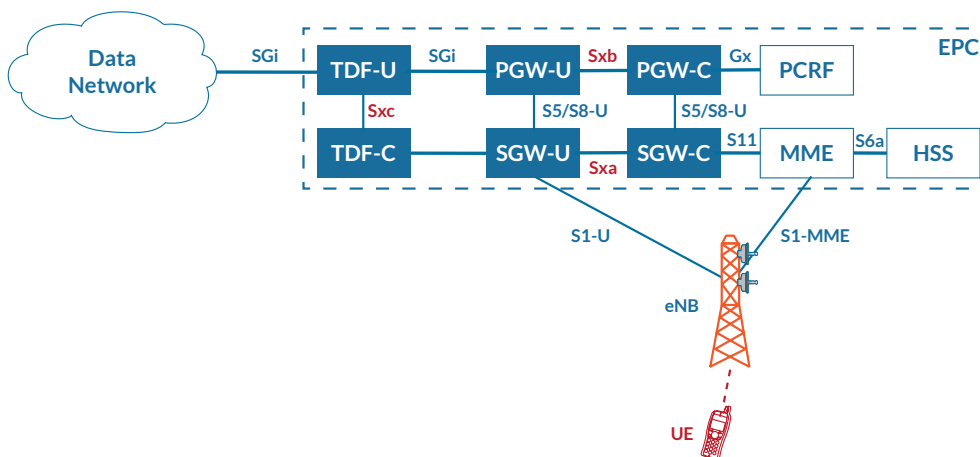


FIGURE 1.4 – Architecture 4G avec CUPS [42]

Avec ce type d'architecture, on adopte une partie du paradigme SDN en centralisant le plan de contrôle de chaque GW permettant ainsi une gestion plus fine des ressources dans le plan de données. On sépare la PGW en deux composants : la PDN Gateway Control Plane (PGW-C) et la PDN Gateway User Plane (PGW-U) qui sont reliées par l'interface Sxb. De même, on effectue la séparation pour la SGW : la Serving Gateway Control Plane (SGW-C) et la Serving Gateway User Plane (SGW-U) au travers de l'interface Sxa.

Néanmoins, même en opérant une séparation claire entre le plan de données et



le plan de contrôle au sein du cœur de réseau avec CUPS, la SGW et la PGW possèdent des fonctionnalités redondantes que ce soit au niveau contrôle ou donnée. Cette redondance peut encore être supprimée et une simplification peut encore s'opérer dans le réseau. On voit donc que même avec les optimisations apportées au sein du réseau 4G, il est encore nécessaire d'apporter plus de flexibilité au fonctionnement du cœur. Le CN 4G a ainsi évolué pour aller vers un CN 5G plus flexible. Comme expliqué précédemment, l'objectif premier du CN 5G est de permettre des déploiements personnalisables en fonction des services.

Avec la maturité des technologies NFV et des changements opérés au niveau des fonctions réseaux, le 3GPP a convergé vers un cœur de réseau se basant sur une Service Based Architecture (SBA). L'introduction du protocole Hypertext Transfer Protocol (HTTP) comme standard permet une flexibilité jamais atteinte aujourd'hui car cela permet de reprendre tous les paradigmes modernes de l'internet tel que les API Representational State Transfer (REST) et les technologies cloud. Chaque network function (NF) possède une Service Based Interface (SBI) qui lui permet d'effectuer des requêtes et d'exposer des API. Cela permet de définir et d'intégrer de nouveaux services qui peuvent répondre à de nouveaux besoins émergents. Il profite aussi des solutions d'orchestration cloud qui permettent le passage à l'échelle du CN rapidement et efficacement.

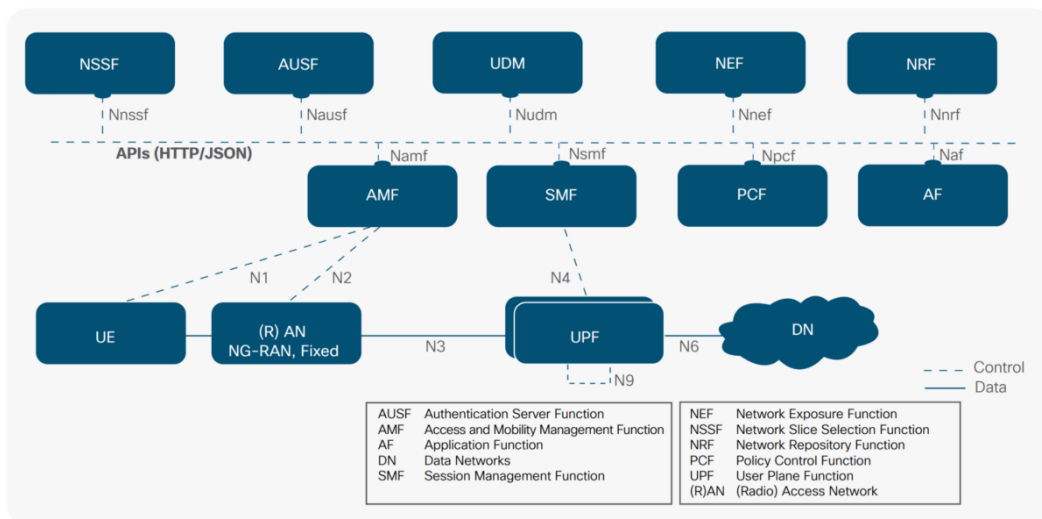


FIGURE 1.5 – Service Based Architecture 5G [3]

Le 3GPP introduit pour ce faire de nouvelles NFs représentées sur l'architecture de la figure 1.5 :

- Access and Mobility Management Function (AMF) : Gestion du plan de contrôle et interconnexion avec le RAN 5G au travers de l'interface N2 ;
- Session Management Function (SMF) : Attribution des adresses IP, création des sessions Protocol Data Units (PDUs) ;
- Policy Control Function (PCF) : Gestion de la politique de QoS du réseau ;
- Unified Data Management (UDM) : Stocke et récupère les informations uti-



- lisateurs. Peut fonctionner de manière stateless ou stateful ;
- Unified Data Repository (UDR) : Stocke les informations utilisateurs lorsque l'UDM fonctionne de manière stateless ;
  - Network Slice Selection Function (NSSF) : Fonction qui sélectionne les slices pour les UE ;
  - NF Repository Function (NRF) : Joue le rôle d'un annuaire. Toutes les NFs du cœur de réseau vont s'enregistrer auprès de cette fonction
  - User plane Function (UPF) : Point d'interconnexion au niveau plan de données entre le RAN 5G et le CN 5G en utilisant l'interface N3 ; Il interconnecte aussi le réseau 5G avec le Data Network (DN) au travers de l'interface N6 ;
  - Network Exposure Function (NEF) : Expose les API des NFs du cœur de réseau aux fonctions extérieures au réseau ;
  - Application Function (AF) : Fonction applicative dédiée à un service qui peut communiquer avec les fonctions du cœur de réseau en utilisant les SBI ;

Fonction 4G	Fonction 5G
MME	AMF
SGW-C, PGW-C	SMF
SGW-U, PGW-U	UPF
HSS	UDM
HSS	UDR
PCRF	PCF
eNB	gNB
—	NSSF
—	NRF
—	AF
—	NEF

TABLE 1.1 – Correspondance entre les fonctions 4G et les fonctions 5G

Cette architecture modulaire permet une flexibilité accrue et inexistante jusque là dans les architectures de télécommunications. Plus largement, ce type d'architecture reprend les standards cloud, permet de pleinement exploiter la flexibilité du cloud et le déploiement de services hétérogènes au sein du réseau 5G. Cela devient donc un activateur pour des architectures de type Mobile Edge Caching (MEC) où l'on peut déporter des fonctions réseaux au plus proche de l'utilisateur en les co-localisant avec le RAN. Ce type d'architecture permet par exemple d'obtenir de meilleures performances pour les CDN, elle permet également l'utilisation du network slicing qui est un des fondamentaux pour permettre le déploiement et l'orchestration de services hétérogènes.

Nous avons synthétisé dans le tableau 1.1 la correspondance entre les fonctions du réseau 4G et du réseau 5G. Il y a une multiplication du nombre de fonctions au

sein du réseau 5G par deux ce qui octroie plus de flexibilité mais paradoxalement accroît la complexité de l'orchestration et de la gestion du réseau. Cela implique le développement de systèmes évolués dédiés à cette orchestration tels que les systèmes Open Network Automation Platform (ONAP) et Open Source MANO (OSM) [43] qui ont pour rôle d'assurer la gestion complète du réseau et de garantir un fonctionnement optimal de ce dernier.

Les changements apportés au CN que nous avons énumérés permettent une plus grande flexibilité au sein du réseau et rendent possible des perspectives d'évolutions accrues par rapport aux réseaux 4G. Néanmoins, eux seuls ne suffisent pas à atteindre le niveau de flexibilité requis pour le déploiement de services end-to-end. Le RAN 5G a donc subi à son tour un ensemble de modifications et d'évolutions afin d'atteindre les objectifs des réseaux 5G.

#### 1.2.4 L'accès radio en 5G

Une multitude d'évolutions ont été apportées au RAN 4G pour lui permettre de supporter les exigences des réseaux 5G et ainsi converger vers le NG-RAN.

##### 1.2.4.1 Architecture

L'architecture des eNBs s'est adaptée aux nouveaux besoins en flexibilité des réseaux 5G. Les eNBs ont évolué pour adopter le paradigme SDN et adopter une architecture simplifiant le déploiement d'éléments radios afin de rapidement et efficacement augmenter la zone de couverture des réseaux 5G. Cette adoption se matérialise par la séparation de l'eNB en composants Central Unit (CU) et Distribution Unit (DU) qui forment la gNB en 5G et qui remplace l'eNB (figure 1.6). Le développement des architectures Cloud-RAN contribue également énormément à atteindre ces objectifs même si elles ne sont pas officiellement développées par le 3GPP dans le standard 5G (cf dans paragraphe 1.2.4.3).

##### 1.2.4.2 Interface Air

Des innovations majeures ont aussi été apportées à l'interface air des réseaux mobiles. L'introduction du 5G New Radio (NR) qui intègre en son sein de multiples avancées technologiques radio permet d'augmenter considérablement la bande passante disponible et la flexibilité de gestion de cette dernière. La 5G utilise dorénavant les ondes millimétriques (> 24 GHz) qui débloquent une utilisation de bande dont la largeur est 25 fois supérieure à la bande Sub-6 GHz (< 6 GHz) des réseaux 3G et 4G [45]. Chacune des bandes de fréquences offre un mode de fonctionnement soit en Time Division Duplex (TDD) qui permet des communications half-duplex ou Frequency Division Duplex (FDD) qui permet des communications full-duplex. Elles possèdent aussi chacune des largeurs de bande différentes, ce qui influe sur la bande passante disponible et donc en fonction des services, on préférera utiliser des bandes spécifiques. Nous avons synthétisés dans l'annexe A les

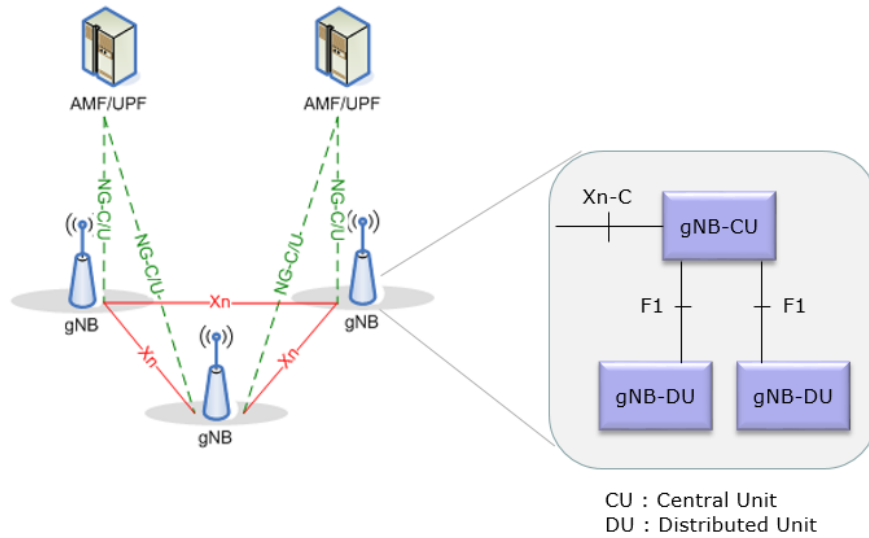


FIGURE 1.6 – Séparation de la gNB en CU et DU [44]

différentes bandes de fréquences utilisées en 5G ainsi que leur mode de fonctionnement. L'utilisation du codage canal avancé tel que le Multi-Edge Low-Density Parity-Check (ME-LDPC) [46] permet d'atteindre des débits de l'ordre du Gbit/s pour chaque UE. Ces deux bandes peuvent être utilisées en simultanément par les UE qui possèdent une double connectivité dans un premier temps en NSA dans lequel les bandes Sub-6 GHz sont utilisées en mode 4G pour les canaux de contrôle et de données et les bandes hautes en mode New Radio (NR) pour les données seulement. En mode SA, toutes les communications se font en utilisant le mode New Radio (NR) sur les bandes Sub-6 GHz et millimétriques. Le mode SA ouvre l'utilisation du CN 5G et des fonctionnalités associées. L'utilisation du Massive MIMO [47] avec le beamforming [48] et le beam tracking [49] dans un environnement où le déploiement du New Radio (NR) à grande échelle est nécessaire pour l'utilisation et l'exploitation des bandes millimétriques. Ce déploiement assure des handover radios efficaces dans les bandes millimétriques ainsi que le support du Non-line-of-sight (NLOS) assurant une continuité de connectivité pour les UE même en utilisant les ondes millimétriques et leurs capacités physiques limitées.

### 1.2.4.3 Cloud-RAN

Dans la continuité des évolutions SDN et NFV, le RAN des réseaux mobiles a aussi évolué afin d'augmenter ses performances et sa flexibilité en développant l'architecture Cloud-RAN [50]. Elle combine le principe de SDN en séparant le plan de contrôle et le plan de données ainsi que le principe de NFV qui permet d'exécuter certains des composants de Cloud-RAN sur du matériel COTS. Elle s'est appliquée aux réseaux 4G avant d'être reprise au sein de la 5G. Cette architecture sépare les eNBs 4G et les gNBs 5G en plusieurs éléments distincts. On effectue

cette décomposition en éléments distincts en identifiant et isolant les différents niveaux protocolaires du plan de données du RAN. La figure 1.7 montre la séparation des couches protocolaires de la 4G et de la 5G en éléments distincts.

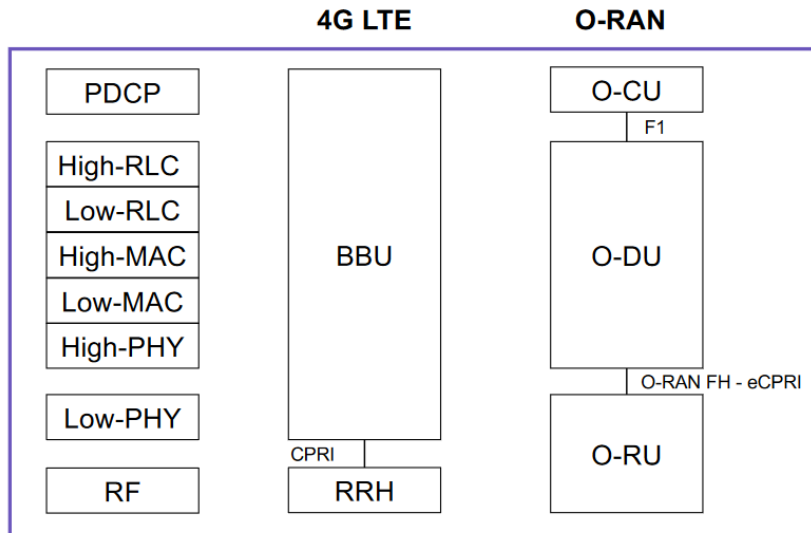


FIGURE 1.7 – Séparation de la pile protocolaire pour Cloud-RAN

Dans la 4G, l'eNB se retrouve distribuée entre les Baseband Units (BBUs) et les Remote Radio Heads (RRHs) qui sont reliés par la Common Public Radio Interface (CPRI). Une BBU agit comme l'élément de contrôle qui centralise tous les algorithmes d'allocation de ressources, prend les décisions et qui est reliée à une ou plusieurs RRHs. Cela permet d'augmenter significativement la couverture radio des réseaux 4G en multipliant le nombre de RRHs tout en minimisant le nombre de BBUs déployées. L'introduction de l'interface CPRI qui standardise les échanges entre les BBUs et les RRHs permet l'interopérabilité entre les équipements de différents constructeurs. On peut donc virtualiser les BBUs en VNFs dans le cloud qui deviennent dès lors des Virtual BBUs (vBBUs) orchestrables avec le framework NFV MANO.

Avec l'évolution des eNBs dans la 5G, le standard décompose les gNBs en éléments Central Unit (CU) et Distribution Unit (DU) séparés par l'interface F1 [51]. Contrairement à la séparation en 4G, l'interface F1 permet d'isoler les couches de niveau 3 et supérieur des couches inférieures. La Central Unit (CU) est séparée en éléments de contrôle et en éléments de donnée qui communiquent respectivement avec les Distribution Units (DUs) en utilisant le plan de contrôle F1-C [52] et le plan de données F1-U [53]. L'architecture Cloud-RAN pour la 5G combine les séparations effectuées par l'interface F1 du standard 5G et la séparation au niveau radio des BBUs et RRHs. On décompose donc finalement, comme le montre la figure 1.8 en : CU relié par le backhaul au CN et par le midhaul aux DUs ainsi que les DUs reliées aux Radio Units (RUs) par le fronthaul. La communication entre les DUs et les RUs s'effectue en utilisant l'interface enhanced CPRI (eCPRI)

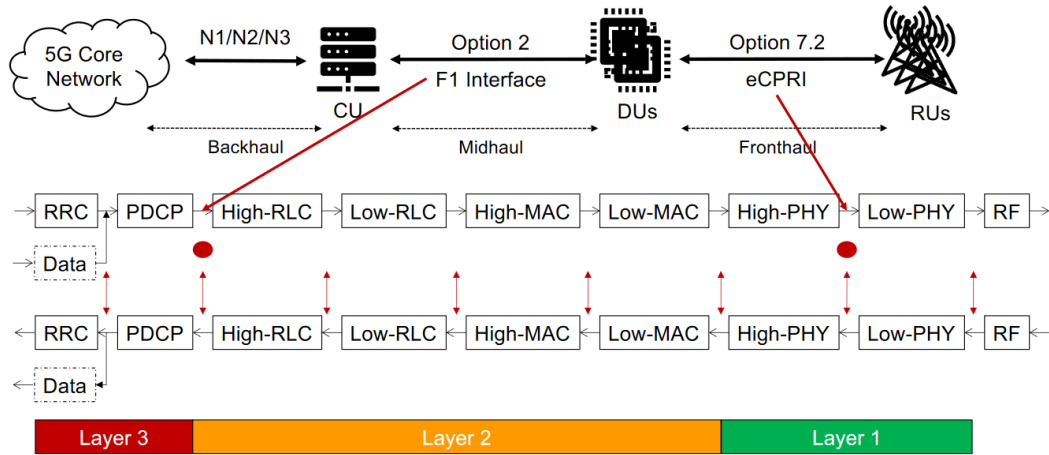


FIGURE 1.8 – Architecture Cloud-RAN

qui permet le dialogue standardisé entre la couche haute et la couche basse de la couche physique radio. Cette séparation fonctionnelle et protocolaire standardisée augmente la flexibilité dont dispose les network operators dans le choix des fournisseurs et des équipements qu'ils déploient dans leur réseau.

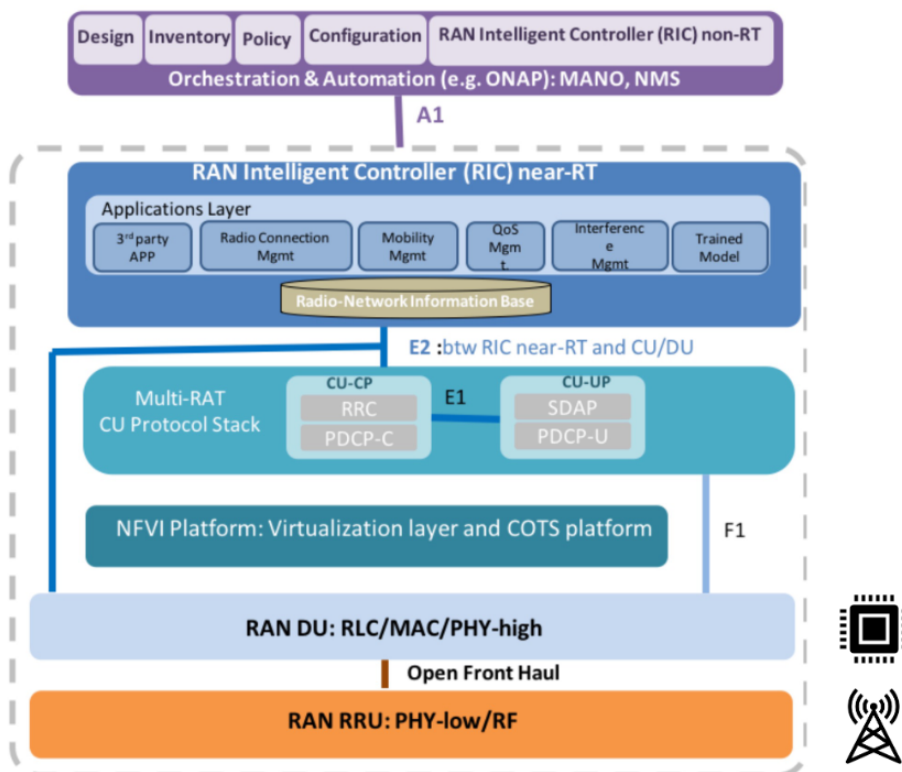


FIGURE 1.9 – Architecture O-RAN

Cette architecture Cloud-RAN a donné lieu à des initiatives importantes qui ont pour objectif de définir une architecture et des standards ouverts open-RAN [54] pour faciliter le développement et l'évolution de la partie radio des futurs réseaux mobiles. Ainsi, le consortium O-RAN [55] regroupant un très grand nombre d'industriels du secteur des télécommunications mobiles, a défini une architecture standardisée Cloud-RAN qui intègre le paradigme SDN, NFV, les technologies de traitement big-data Artificial Intelligence (AI) et compatibles avec le standard 5G défini par le 3GPP. Cette initiative vise à accélérer le développement du RAN, pérenniser l'utilisation des technologies cloud et permettre de cibler les travaux lors de l'évolution des technologies d'accès radio.

La figure 1.9 représente l'architecture fonctionnelle O-RAN. Elle se focalise sur l'orchestration du RAN et standardise les différentes interfaces entre les composants pour permettre une interopérabilité au niveau de l'orchestration. La couche d'orchestration MANO est responsable de l'automatisation de la gestion du RAN et embarque un ensemble de fonctions dédiées à la gestion et à la configuration non temps réel du réseau radio. Cette couche communique avec les fonctions réseaux par le biais de l'interface A1. En plus des fonctions Central Units (CUs), DUs et RUs, O-RAN introduit un contrôleur RAN Intelligent Controller (RIC) near-RT qui embarque une couche applicative dédiée à la gestion temps réel du RAN et des ressources associées (e.g. mobilité, QoS, ressources radio). L'une des spécificités de cette architecture est aussi la séparation des Central Units (CUs) en élément de contrôle et données afin de permettre le support de plusieurs types d'accès radio (e.g. terrestre, satellite).

## 1.3 Les réseaux satellites

### 1.3.1 L'architecture broadband

Les réseaux satellites sont des réseaux wide area network (WAN) permettant de fournir de la connectivité aux utilisateurs finaux. Ils possèdent leur propre architecture et permettent de déployer plusieurs types de services et répondre à plusieurs cas d'usages différents. On retrouve plusieurs types de satellites : les satellites broadband géostationnaires, en orbite moyenne et en orbite basse. En fonction du service à déployer, les technologies diffèrent ainsi que le type de satellite à utiliser. Nous décrivons ici l'architecture traditionnelle des réseaux satellites géostationnaires broadband permettant de fournir une connectivité réseau. L'architecture des réseaux satellites broadband est séparée en plusieurs segments : le user-segment (segment utilisateur), le ground-segment (segment sol) et le space-segment (segment spatial). Les utilisateurs finaux se situent dans le user-segment et communiquent avec les gateways situées dans le ground-segment.

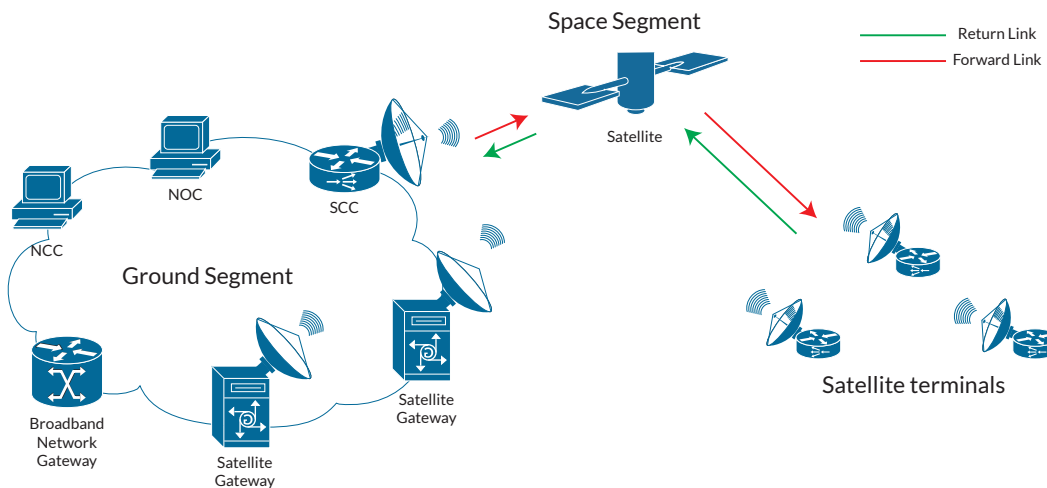


FIGURE 1.10 – Architecture Satellite Broadband

Selon le type de communication, les standards et protocoles sont différents comme avec par exemple le Digital Video Broadcasting - Satellite - Second Generation (DVB-S2) et Digital Video Broadcasting - Second Generation Return Channel Satellite (DVB-RCS2). Ce sont des standards adaptés pour faire du broadband géostationnaire. Les satellites géostationnaires ont évolué vers des modèles VHTS pour augmenter leur capacité globale et atteindre des centaines de Gb/s [56] voir des Tb/s [57, 58] notamment grâce aux technologies à bord telles que les processeurs ré-génératifs [59] les antennes actives [60, 61] et le beam-hopping [62]. La multiplication des stations sols permet aussi d'augmenter la capacité des systèmes géostationnaires néanmoins même avec ces améliorations on atteint toujours une limite fixée par le poids des satellites que l'on ne peut augmenter indéfiniment. Une des pistes explorée aujourd'hui est l'utilisation de laser Free Space Optical

(FSO) [63, 64] qui permettrait de disposer d'une capacité infinie pour émettre et recevoir des données. Ces technologies seraient ainsi applicables sur le feeder link (gateway satellite) mais seraient complexes à mettre en œuvre pour le user link au vu du prix des technologies. Une des fortes contraintes du satellite géostationnaire est la latence élevée des communications qu'il est difficile de minimiser, qui possède un impact non négligeable sur le protocole de transport TCP ainsi que les conditions météorologiques qui peuvent fortement dégrader la qualité du lien radio. En effet, il est nécessaire d'introduire la segmentation des connexions TCP à l'aide de Performance-enhancing Proxies (PEP) [65]. Ces techniques complexes sont coûteuses ce qui a nécessité la recherche de plusieurs alternatives comme l'utilisation d'algorithmes spécifiques pour TCP [66] ainsi que l'utilisation du protocole de transport QUIC [67] qui permet en plus de palier la latence élevée du satellite de sécuriser les communications de bout-en-bout [68]. Nécessitant toujours plus de capacité et des latences plus faibles, les acteurs du monde satellite ont alors œuvré pour déployer des satellites de nouvelle génération plus petits mais plus nombreux avec les mega-constellations.

### 1.3.2 Les évolutions technologiques des réseaux satellites

Avec l'arrivée de SDN et NFV dans les réseaux terrestres, de nombreux projets de recherches tel que le projet VITAL [69] se sont focalisés sur l'application de ces technologies aux réseaux satellites afin d'en intégrer tous les avantages. Ainsi, Ferrús et al. [70] ont listé différents scénarios d'intégration et défis associés. On en relève trois distincts : la virtualisation des réseaux satellites qui faciliterait la gestion et la création de nouveaux opérateurs satellites virtuels, l'amélioration des services de backhauling 4G/5G et le déploiement d'accès hybride satellite-terrestre qui permet une meilleure gestion et optimisation des flux. Dans [71], Ferrús et al. proposent d'utiliser SDN/NFV afin d'effectuer de l'ingénierie de trafic de bout-en-bout en intégrant les réseaux terrestres et satellites. Ahmed et al. ont proposé une application du paradigme SDN couplé avec NFV aux réseaux satellites dans [72] où ils les appliquent dans le segment sol afin d'effectuer de la diversité de gateway satellites et assurer une meilleure disponibilité des feeder links. Cela se corréle aux travaux de Bertaux et al. [73] dans lesquels la diversité de gateway est utilisée pour améliorer les mécanismes de handover. Les auteurs énumèrent aussi les différentes opportunités d'intégration uniforme des réseaux satellites dans les réseaux mobiles terrestres et considèrent les technologies SDN/NFV comme des activateurs pour cette intégration. Dans [74] Toufik et al. proposent d'appliquer l'architecture Cloud-RAN terrestre aux gateway satellites DVB-S2/DVB-RCS2 et définissent une architecture similaire "Software Defined Cloud-RAN" dans lequel le découpage des gateway satellites est effectué afin d'en virtualiser une partie et d'optimiser les coûts ainsi que les allocations de ressources. Cette approche est spécifique au standard DVB mais constitue une solide base pour être étendue à d'autres standards ou bien être rendue générique. Récemment (2021), en ayant des approches similaires, les acteurs du cloud ont lancé des services pay-as-you-go qui



permettent de déployer des gateway satellites à la demande. Microsoft et Amazon ont lancé respectivement leurs services AWS Ground Station [75] et Azure Orbital [76] qui permettent de déployer des gateway en bande X et S et de communiquer avec des satellites en orbite. Ces services sont néanmoins limités et ne supportent que le standard radio VITA49 et nécessitent du matériel spécifique déployés dans des datacenters bien précis, de plus ils sont dédiés à la communication aux satellites en orbite basse.

### 1.3.3 Les mega-constellations broadband

Avec l'arrivée des mega-constellations broadband, de nouvelles opportunités d'intégrations se présentent. Le tableau 1.2 liste les différentes constellations broadband en cours d'élaboration et celles qui sont en cours de déploiement ou en projet. On retrouve plusieurs types de constellations, celles qui par exemple ont pour objectif de proposer de l'accès direct en 5G aux utilisateurs telles que AST SpaceMobile [77] ou encore celles qui ont pour objectif d'offrir un service broadband à des usagers finaux telles que Starlink [78] ou Telesat [79] par le biais de Very Small Aperture Terminal (VSAT) ou Earth Station in Motion (ESIM). Comparativement aux satellites géostationnaires, les constellations offrent des temps de latence beaucoup plus faibles et disposent d'une capacité accrue. La faible latence permet de s'affranchir des coûteuses techniques de segmentation des connexions TCP introduites avec les PEP et permet donc d'accroître les performances des communications. Le développement et le déploiement de ces mega-constellations est aujourd'hui rendu possible par un nombre multiple d'avancées technologiques.

Constellation	Nombre de satellites	Altitude (km)	Type de terminal
Iridium (gen 1)	66/77	780	Spécifique
Globalstar	48/48	1440	Spécifique
AST SpaceMobile	1/240	720	Terminal 5G
OneWeb	428/650	1200	VSAT, ESIM
Starlink	2651/11943	550	VSAT, ESIM
Lightspeed	0/298	1015 et 1325	VSAT, ESIM
Amazon Kuiper	9/3200	600	VSAT, ESIM
Guo Wang	0/12992	Planifié	VSAT, ESIM

TABLE 1.2 – Mega-constellations LEO (basé sur [80] et enrichi jusqu'à Juillet 2022)

Les avancées dans le domaine de la finesse de gravure des composants silicium permet une réduction des coûts pour les composants à bord (hors pandémies, guerres ou pénuries de semi-conducteurs). Le déploiement des technologies Optical Inter-Satellite Links (OISL) permettent une communication entre les différents satellites d'un même plan orbitale [81] et même entre les satellites de différents plans orbitaux. Le déploiement des méga-constellations est aussi accéléré par la réduction des coûts de lancement de satellites avec l'arrivée de nouveaux

lanceurs polyvalents adaptés [82]. Néanmoins, cela nécessite des technologies à bord de plus en plus complexes ainsi qu'au sol puisqu'il devient nécessaire dorénavant de pouvoir suivre les satellites en fonction de la position. Cela induit aussi des changements de cellules avec les handover qu'il faut minimiser afin d'assurer une continuité de service. Une problématique grandissante avec les mega-constellations pourrait être l'occupation intensive des orbites basses et moyennes qui peuvent aboutir à une saturation de ces orbites. Cette problématique serait dans le futur un obstacle majeur au déploiement de nouvelles constellations et au maintien opérationnel des satellites existants.

## 1.4 L'intégration des réseaux 5G et satellites

Une multitude de projets de recherches se sont intéressés à l'intégration des réseaux satellites dans les réseaux mobiles terrestres. Nous pouvons citer le projet VITAL qui a eu pour objectif d'appliquer les concepts de SDN et NFV au réseau satellite afin d'augmenter sa flexibilité et permettre une intégration étroite avec les réseaux 4G. Le projet SaT5G ayant pour objectif l'intégration du satellite dans les réseaux 5G lui a succédé, proposant différents modes d'intégration qui peuvent être classés en deux catégories [83] comme montré sur la figure 1.11.

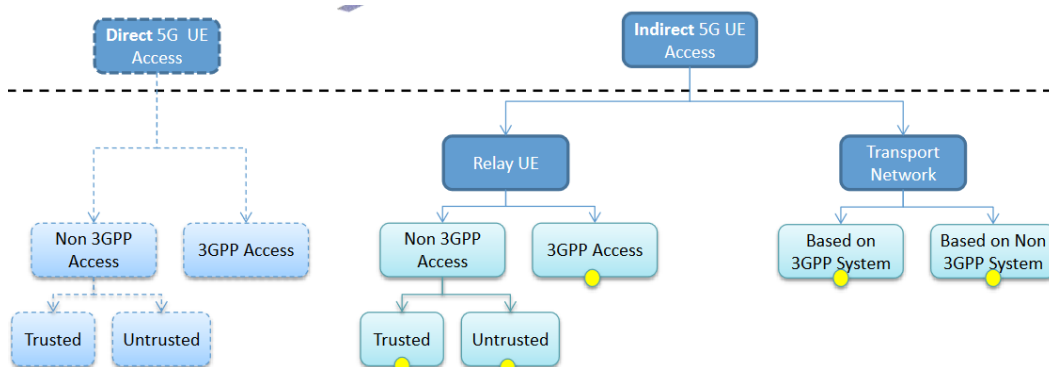


FIGURE 1.11 – Modes d'intégration des réseaux satellites dans la 5G [84]

L'intégration en accès direct que nous décrivons dans le §1.4.1 permet l'utilisation des réseaux satellites directement comme extension radio aux réseaux 5G et peut donc être considéré comme un RAN 5G. L'intégration peut se faire en accès indirect, décrit dans le §1.4.2, où l'on considère le réseau satellite comme un intermédiaire entre le RAN et le CN 5G. Chacun de ces modes d'intégration possède des contraintes différentes et offre des niveaux d'intégration plus ou moins complets et donc complexes. Le tableau 1.3 synthétise les différents modes d'intégration du satellite dans les réseaux 5G : les modes d'accès direct, les modes d'accès indirect en tant que noeud relai (en utilisant des UE passe-relai permettant de connecter d'autres UE plus petit) et l'intégration en tant que Transport Network (TN) pour des services de lien backhaul.

TABLE 1.3 – Les différents modes d'intégration [84]

Intégration	Défis	Gestion du réseau	Échéancier
3GPP Access	Support de la forme d'onde NR au sein du satellite	Composant de gestion unique	Long terme
Trusted non-3GPP Access	Intégration du satellite en mode trusted non-3GPP Access au sein du standard	Composant de gestion unique	Moyen terme
Untrusted non-3GPP Access	Support des mécanismes untrusted access 3GPP au sein du réseau satellite	Composant de gestion unique	Court terme
Relay node with 3GPP Access	Support de la forme d'onde NR au sein du satellite, adaptation des mécanisme de relayage au sein du terminal satellite	Composant de gestion unique	Long terme
Relay node with Trusted non-3GPP Access	Intégration du satellite en mode trusted non-3GPP Access au sein du standard, adaptation des mécanisme de relayage au sein du terminal satellite	Composant de gestion unique	Moyen terme
Relay node with Untrusted non-3GPP Access	Support des mécanismes untrusted access 3GPP au sein du réseau satellite, adaptation des mécanisme de relayage au sein du terminal satellite	Composant de gestion unique	Court-Moyen terme
Transport Network based on 3GPP System specification	Définition d'un système satellite compatible avec le standard 5G	Interconnexion des composant de gestions des réseaux satellite et 5G	Court-Moyen terme
Transport Network non-based on 3GPP System specification	Définir une couche d'adaptation pour le réseau satellite considéré comme un réseau de transport	Interconnexion des composant de gestions des réseaux satellite et 5G	Court terme

### 1.4.1 L'utilisation des réseaux satellites en réseau d'accès 5G

Dans le cas d'une intégration pour un accès direct au réseau 5G par le réseau satellite, un UE qui posséderait une connexion satellite serait en mesure d'accéder directement au CN 5G par le biais de l'infrastructure satellite. Le niveau d'intégration maximale est lorsque le terminal satellite est compatible 5G et permet un accès 3GPP. Il embarque donc tous les protocoles de communications 5G ainsi que l'accès radio ce qui oblige le segment spatial à supporter en son sein la forme d'onde New Radio (NR) 5G. Les satellites agissent donc en tant que gNB pour les terminaux satellites qui sont des UE. Une partie des bandes de fréquences FR2 (n259, n260, n261, n262 en annexe A.1) est compatible avec les systèmes satellites qui utilisent pour leurs communications les bandes Ka et V. Depuis la release 15 du 3GPP [3], des travaux sur l'adaptation de la forme d'onde 5G aux réseaux satellites sont en cours au sein de l'organisme de standardisation 3GPP [85] qui identifie le satellite comme Non Terrestrial Network (NTN). Les résultats préliminaires démontrent qu'il est nécessaire d'adapter la forme d'onde pour les réseaux satellites. En plus d'adapter la forme d'onde, il n'est pas certain qu'elle puisse être supportée sur tout type de satellite, selon l'orbite il serait ou non possible d'utiliser cette forme d'onde. L'un des grands défis est aussi l'adaptation de cette forme d'onde pour fournir des services haut débit en utilisant les réseaux satellites pour des smartphones classiques. En effet, l'objectif majeur d'une intégration en accès direct n'est pas de permettre à des terminaux satellites de supporter la forme d'onde 5G mais de pouvoir étendre la couverture des réseaux 5G sans avoir à déployer d'infrastructure au sol. Ce sont les smartphones traditionnels 5G qui doivent être en mesure de se connecter aux satellites de manière transparente lorsque la couverture radio de gNB terrestres n'est plus assurée afin de garantir une continuité de service. Ce niveau d'intégration est recherché par les mega-constellations du type AST SpaceMobile qui ont pour objectif de proposer une extension de couverture à moindre frais aux opérateurs mobiles terrestres.

Outre les adaptations de la forme d'onde et des contraintes liées à la couche physique, des intégrations au niveau protocolaire doivent être réalisées afin que le CN 5G puisse communiquer avec le RAN 5G satellite. De plus il faut que les systèmes de management satellite et 5G puissent communiquer afin d'assurer une orchestration cohérente des ressources allouées. Dans le cas des mega-constellations satellites, la gestion complexe de la flotte satellite, qui est mobile, doit aussi être prise en compte car elle influe fortement sur les mécanismes de handover qui ne sont plus déclenchés seulement par l'utilisateur qui est en mouvement avec son terminal mais aussi par les satellites qui sont en mouvement. A ce jour, aucun test réel n'a été effectué pour de l'accès direct 5G par satellite, des premiers tests sont annoncés par AST SpaceMobile en 2023.

Il est possible d'utiliser les terminaux satellites en mode Non-3GPP mais cela force l'utilisation de la Non-3GPP Interworking Function (N3IWF). Les réseaux d'accès satellites ne sont donc plus des moyens fiables d'accéder au réseau 5G ce qui limite grandement l'interaction des réseaux entre eux et empêche une conver-

gence vers un réseau uniforme.

### 1.4.2 L'intégration du segment satellite dans les réseaux 5G

Il est possible d'intégrer les réseaux satellites dans les réseaux 5G en accès indirect, ce qui signifie qu'un UE 5G se connectera à une gNB ou à un UE relais en ignorant qu'il utilise potentiellement un lien satellite pour se raccorder au réseau [86]. L'utilisation du réseau satellite en tant que relais UE permet de répondre au cas d'usage spécifique des plate-formes mobiles (e.g. avions, bateau) mais n'apportent que très peu de valeur ajoutée comparé à un lien backhaul entre une gNB localisé sur la plate-forme et le CN 5G. L'avantage d'intégrer le lien satellite en tant que backhaul est qu'il conserve la totalité de ses fonctionnalités et profite ainsi des nombreuses technologies spécifiques au satellite qui permettent de relayer plus de trafic qu'avec un accès direct. Les mega-constellations broadband sont un candidat parfait pour ce type d'intégration qui atteignent aujourd'hui sur des terminaux utilisateurs plusieurs centaines de Mbit/s et prétendent atteindre le Gbit/s sur des terminaux spécifiques.

Le projet SaT5G [84] a défini une architecture d'intégration haut niveau du réseau satellite en tant que Transport Network (TN) fournissant un lien backhaul au réseau 5G et défini les besoins auxquels il faut répondre pour réaliser cette intégration. De même que pour l'intégration en accès direct, les architectures 5G et satellites doivent être mises en concurrence et des mécanismes d'interconnexion introduits. Les liens backhaul classiques ne tiennent pas compte du trafic 5G et fournissent des points de terminaisons pour relayer des données entre deux équipements réseaux. Le réseau satellite est donc considéré, dans ce dernier cas, comme un réseau de transport ce qui n'apporte aucun niveau d'intégration supérieur comme pour l'accès direct. Des modifications sont donc nécessaires afin que le lien satellite ne soit plus transparent, permette une meilleure intégration et atteindre les objectifs fixés par les réseaux 5G.

Plusieurs aspects ont donc été étudiés au sein du projet SaT5G [87] sur ce type d'intégration. Le besoin d'orchestration globale entre les réseaux 5G et satellites nécessite une coordination des équipements de gestion 5G et satellite. L'utilisation de Network Management System (NMS) au sein de chaque réseau est spécifié avec une synchronisation et une coordination par le biais d'interfaces est-ouest entre eux. Les résultats du projet fournissent une analyse haut niveau des communications entre ces NMS mais ne spécifient nullement les interfaces ainsi que les messages échangés. On peut notamment noter l'absence de métriques tels que des KPIs end-to-end au niveau orchestration qui permettent d'établir clairement des communications satellites-5G et d'en mesurer les performances. En effet, il est complexe de mettre en œuvre une orchestration globale qui permettrait la gestion tout en garantissant des métriques qui doivent aussi être instanciées au sein des deux réseaux.

Chaque flux 5G traverse le RAN 5G, puis est relayé par le réseau satellite jusqu'au CN 5G. Une analyse des flux 5G end-to-end a donc été effectuée avec une

proposition architecturale pour la correspondance entre les classes de QoS 5G et satellite. Il est de la responsabilité du terminal satellite et de la gateway satellite de traduire les flux en provenance du réseau 5G afin qu'ils soient traités avec les bonnes politiques de QoS au sein du réseau satellite puis de traduire à nouveau ces flux vers le réseau 5G. Cependant, aucune correspondance entre les classes de QoS 5G et satellite n'a été proposé et le projet SaT5G cite les travaux en cours de standardisation au sein du 3GPP.

Enfin, le projet investigate le support du network slicing au sein d'une architecture 5G-satellite à un très haut niveau. Le projet souligne la nécessité du déploiement de NFV et de SDN au sein des réseaux satellites afin de pouvoir augmenter considérablement leur flexibilité et permettre le paradigme de network slicing. C'est une étape nécessaire afin de pouvoir, au sein d'une architecture intégrant le satellite et les réseaux 5G, déployer des network slices end-to-end.

## 1.5 Conclusion

Nous avons présenté dans ce chapitre les principales avancées technologiques des réseaux mobiles terrestres et des réseaux satellites. Les réseaux 4G ont évolué vers des réseaux mobiles plus flexibles qui permettent le déploiement de services hétérogènes end-to-end au dessus d'une même infrastructure physique. Ces déploiements sont rendus possibles par l'adoption des technologies SDN et NFV qui sont les pierres angulaires des réseaux mobiles du futur, elles s'appliquent tant dans le CN que dans le RAN. Cela permet d'augmenter considérablement la capacité, les performances et l'efficacité des réseaux 5G afin de permettre une importante modularité et paramétrer les déploiements en fonction du contexte cible.

De même, les réseaux satellites ont considérablement évolué afin d'augmenter leur capacité, leur résilience et accroître leur flexibilité. Plusieurs avancées scientifiques majeures ont permis l'amélioration des satellites géostationnaires ainsi que le développement de mega-constellations broadband atteignant des performances jusque là impossibles. C'est dans ce contexte qu'il est aujourd'hui envisageable d'effectuer une intégration des réseaux satellites dans les réseaux mobiles terrestres dans un premier temps et une convergence vers des réseaux uniques dans un second temps.

Les projets de recherche et industriels ont initié ces travaux et ont ouvert la voie à une intégration des réseaux satellites dans le standard 5G qui les intègre depuis la Release 15 [3]. Comme illustré sur le tableau 1.3, plusieurs types d'intégrations sont possibles mais l'on retient les deux grandes catégories accès direct et indirect. Les travaux sur l'accès direct ne sont qu'aux phases préliminaires et impliquent un très grand nombre de défis à résoudre avant de pouvoir ambitionner un déploiement réel. Notre travail s'inscrit dans la continuité de l'accès indirect et plus spécifiquement en considérant le réseau satellite comme un Transport Network (TN) au sein du réseau 5G.

Dans ce contexte, notre objectif est de proposer une intégration transparente

qui préserve les exigences de QoS end-to-end. Les technologies SDN et NFV ne permettent pas, à elles seules, d'effectuer cette intégration. Dans la suite de notre travail, nous proposons de nous appuyer sur le paradigme de network slicing qui a l'ambition d'offrir des communications personnalisées de bout en bout, sur des infrastructures réseaux mutualisées. De plus, l'utilisation du network slicing comme outil d'intégration permet de franchir la première étape vers une intégration en accès direct qui nécessite le support complet de ce paradigme.

# Le Network Slicing

---

## Sommaire

---

<b>2.1</b>	<b>Convergence vers le network slicing</b>	<b>34</b>
2.1.1	Les nouveaux services dans les réseaux	34
2.1.2	Travaux préliminaires	35
2.1.3	Prise en compte du concept dans les différents organismes	35
2.1.4	Modèle de rôle	37
2.1.5	Architecture et Orchestration	39
2.1.6	Plan de contrôle	42
2.1.7	Plan de données	43
2.1.8	Documents de références	44
<b>2.2</b>	<b>Caractéristiques d'un Network Slice</b>	<b>47</b>
2.2.1	Propriétés des slices réseaux	47
2.2.2	Les défis associés	50
<b>2.3</b>	<b>Conclusion</b>	<b>55</b>

---

Dans ce chapitre, nous effectuons une revue complète et synthétique du paradigme de Network Slicing, son origine, ses objectifs, les problématiques auxquelles il propose des réponses ainsi que les nouveautés qu'il apporte dans la gestion et le déploiement des réseaux. Nous commençons par présenter le network slicing abordé par les organismes de standardisation pour les réseaux mobiles de 5<sup>ème</sup> génération. Ensuite, nous listons les différents enjeux que pose ce paradigme et lions cela aux activités de recherche qui essaient de résoudre chacun de ces enjeux.



## 2.1 Convergence vers le network slicing

### 2.1.1 Les nouveaux services dans les réseaux

Bien que le concept de network slicing ne soit pas spécifique aux réseaux mobiles, ce sont ces nouveaux réseaux qui ont su créer un engouement autour duquel sont axés la majorité des travaux académiques et industriels de recherche sur le slicing. Le network slicing a été introduit par le 3GPP dans la Release 15 du standard 5G [3] mais les travaux se poursuivent toujours et les différents aspects du slicing ne sont pas achevés ce qui ne permet pas encore son déploiement. Le network slicing est un concept end-to-end qui permet de déployer des partitions réseaux personnalisées bout en bout allant du serveur applicatif jusqu'à l'utilisateur final. Chaque partition nommée "network slice" est déployée au dessus d'une seule et même infrastructure réseau physique ou virtuelle mutualisée. Chaque network slice est un objet réseau autonome dédié à un service qui peut avoir des besoins totalement différents des autres slices déployés au sein du même réseau. La figure 2.1 illustre le partitionnement d'un réseau mobile ayant une seule et même infrastructure physique au dessus de laquelle trois network slices dédiés à trois services différents ont été déployés. Chaque utilisateur du réseau mobiles est ainsi en mesure d'accéder aux services déployés au sein du réseau en se raccordant aux slices qui les hébergent.

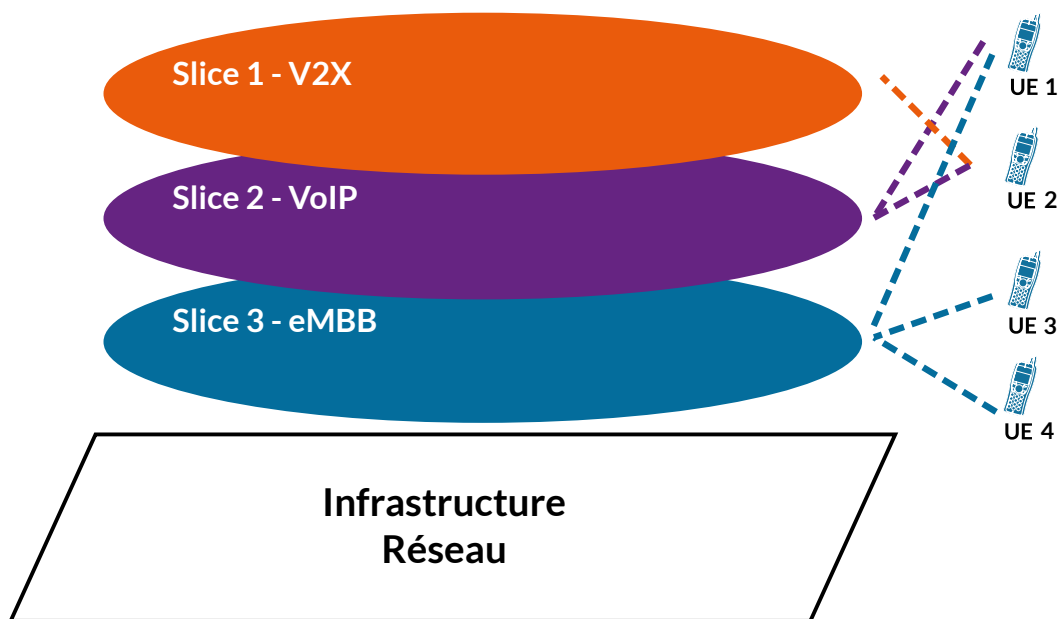


FIGURE 2.1 – Partition d'un réseau avec différents slices

Par exemple l'utilisateur 1 va se raccorder aux network slices 3 et 2 afin de profiter des services Voice over IP (VoIP) et enhanced Mobile Broadband (eMBB) fournis par le réseau. Chacun des slices est associé à un ensemble d'indices de performances bout-en-bout (end-to-end KPIs) [88] qui doivent être assurés afin

de respecter les exigences des services associés au slice. Ces KPIs font partie de la définition du slice au sein du réseau mais ne suffisent pas à définir la totalité de l'éco-système network slicing. Un slice réseau doit répondre à une multitude d'exigences et de critères qui doivent permettre son orchestration, ses performances ainsi que sa bonne gestion dans le réseau. Dans la section 2.1 nous présentons les différents travaux qui ont amené à la définition du slicing contemporain, les acteurs impliqués ainsi que les modèles de rôles et cas d'usages du network slicing. Dans la section 2.2, nous donnons les caractéristiques des slices, les enjeux qui en découlent et les propositions qui essaient d'y répondre. Enfin dans la section 2.3 nous concluons sur l'état actuel du network slicing dans les réseaux.

### 2.1.2 Travaux préliminaires

Le concept de network slicing est le résultat d'années de recherches qui ont débutées dans les années 2000 et qui avaient pour but de réfléchir à l'internet du futur [89, 90], ses contraintes et les modèles qui pouvaient y répondre. Le terme "slice réseau" est apparu pour la première fois avec le projet GENI [91] qui mit en place une plateforme d'expérimentation réseau partagée aux États-Unis. Le projet GENI introduit un slice comme étant une unité d'isolation pour une expérimentation réseau, un ensemble de ressources allouées à l'expérimentation et une unité de contrôle d'accès. Les initiatives en ce sens se sont multipliées résultant en un ensemble d'exigences qui furent formulées dans Management and Service-aware Networking Architectures (MANA) [92] en 2009 comme caractéristiques de slices réseaux généralisés. Cette architecture définit les caractéristiques de slices réseaux au niveau des ressources, de l'orchestration, de l'architecture globale ainsi que de l'infrastructure sous-jacente. En se fondant sur ces travaux, l'ITU-T a spécifié dans ses recommandations sur les réseaux du futur [93] en 2012 [94] le concept de la "softwarisation réseau" ainsi que ses cas d'usage. Chaque slice, au sens ITU-T est une unité Logically Isolated Network Partition (LINP) qui coexiste avec plusieurs autres LINP au dessus d'une seule et même infrastructure réseau physique, chaque LINP étant une unité de ressources programmable (e.g. calcul, stockage). Le concept de cette unité LINP sera repris par d'autres organismes et deviendra par la suite dans leur spécification un "slice".

### 2.1.3 Prise en compte du concept dans les différents organismes

Cette notion est reprise en 2015 par le conglomérat Next Generations Mobile Networks (NGMN) qui précise le concept de network slicing pour la 5G [95] et le sépare en trois couches distinctes [96] (figure 2.2) : le Service Instance Layer (i) (la couche service), le Network Slice Instance Layer (ii) (la couche instance de slices) et le Resource Layer (iii) (la couche ressources). On retrouve dans (i) les services instanciés pour les utilisateurs finaux ou instanciés pour différents acteurs dans le cadre de marchés verticaux, ces services peuvent être fournis par l'opérateur réseau ou un tiers. L'opérateur réseau utilise un "blueprint" (spécification

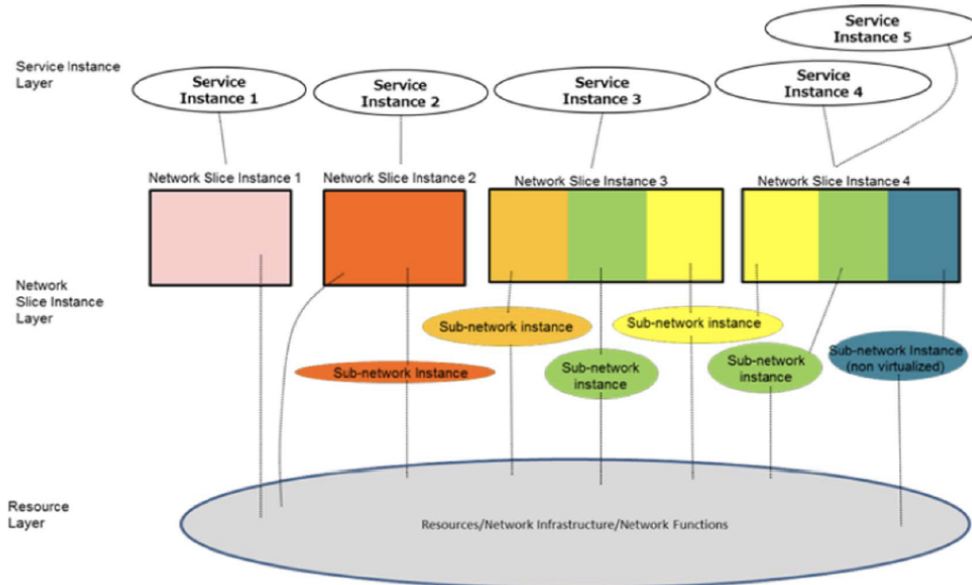


FIGURE 2.2 – Concept de Network Slicing présenté par le NGMN

formelle) pour instancier un ou plusieurs slice dans (ii). Chaque slice peut être la composition de plusieurs sous-réseaux qui peuvent eux-même être partagés par des instances différentes de slices. Un sous-réseau est un ensemble de plusieurs NFs (fonctions réseaux) qui sont instanciées au dessus de (iii) et consomment des ressources physiques ou virtuelles. La même année, l'organisme 3GPP effectua un premier rapport [97] sur le network slicing au sein des réseaux mobiles de 5<sup>ème</sup> génération qui initia les travaux de standardisation du network slicing pour la 5G. L'ONF, en combinant les exigences fonctionnelles des réseaux 5G ainsi que les spécifications du NGMN a décrit de quelle manière le paradigme SDN pouvait s'intégrer et répondre à une partie des exigences du network slicing [98]. De même l'ETSI a proposé des méthodes d'orchestration pour le network slicing en étendant l'utilisation de son framework ETSI NFV MANO [27].

TABLE 2.1 – Organisme travaillant sur le network slicing

	NGMN	ITU	ONF	ETSI	3GPP	GSMA	IETF
<b>Définition</b>	●	●			●	●	
<b>Modèle de rôle</b>	○	●			●		
<b>Architecture</b>		●			●		
<b>Orchestration</b>		○		○	○		
<b>Plan de contrôle</b>			○		●		○
<b>Plan de données</b>			○		●		○

Nous avons listé dans le tableau 2.1 les contributions des différents organismes

sur le network slicing. Il indique quels champs d'application sont abordés et traités par chacun et s'ils sont traités complètement (●) ou partiellement (○). Ce tableau a été construit en se basant sur les propositions de modèles de rôle décrit dans la section 2.1.4, les solutions d'orchestration et l'architecture des slices dans la section 2.1.5, les propositions de plan de contrôle dans la section 2.1.6 et les propositions de plan de données dans la section 2.1.7. L'ensemble des travaux listés dans cette section s'appuie sur les documents normatifs, white papers et spécifications des différents organismes listés dans le tableau 2.1. Nous avons synthétisé dans le tableau 2.3 l'ensemble des documents de standardisation que nous avons compilés afin d'effectuer notre analyse.

#### 2.1.4 Modèle de rôle

Le network slicing étant un concept end-to-end, plusieurs acteurs réseaux existants sont impliqués dans la mise en place d'un slice.

L'ITU-T dans son architecture LINP a défini cinq acteurs [99] ainsi que les interactions existantes entre eux par le biais d'interfaces haut niveau : le network operator (opérateur réseau) (a) dont la responsabilité est la gestion des ressources qui sont allouées aux LINP (virtuelles ou physiques), il est aussi responsables de la gestion d'accès des end-users (utilisateurs finals) (b) aux LINP par le biais de gateways. Les end-users au sein d'un LINP accèdent aux services qui sont développés par les service developers (développeur de services) (c) qui déploient leurs services au sein d'un ou plusieurs LINP. Il est de la responsabilité du LINP operator (opérateur LINP) (d) de gérer un LINP en instanciant des services au sein du LINP et, dans des cas avancés, le LINP operator peut aussi effectuer de la gestion de ressources au sein du slice. Enfin dans le cas où un LINP devrait être instancié au sein de deux infrastructures réseaux distinctes appartenant à deux network operators différents, le LINP exchanger (échangeur LINP) (e) assure une cohérence dans la gestion du LINP en agissant comme une passerelle entre les deux network operators. Ce modèle présenté par l'ITU-T constitue l'un des modèles de base dont plusieurs modèles de rôle dérivent. Il peut être considéré comme un modèle de base générique sur lequel il est possible d'itérer pour les différents types de réseau.

Dans le standard 5G, le 3GPP précise ce modèle de rôle davantage pour s'adapter aux exigences des réseaux mobiles et intègre tous les différents acteurs impliqués dans le network slice [40]. Le modèle de rôle de la 5G modélise aussi les interactions client-fournisseur entre les acteurs respectifs. Les slices au sein des réseaux 5G peuvent être utilisés de deux manières distinctes : en mode Network Slice as a Service (NSaaS) où l'on expose des slices pour qu'ils soient utilisés par des tiers fournissant des services ou alors en mode Network slice as Network operator internal où l'opérateur réseau exploite le paradigme du network slicing pour simplifier la gestion du réseau sans qu'il y ait nécessairement des implications commerciales avec d'autres acteurs. Il est dans ce dernier cas l'équivalent du LINP operator dans le modèle de l'ITU-T. Le modèle de rôle que le 3GPP a défini s'adapte à ces deux

modes d'utilisation. La figure 2.3 montre le modèle de rôle défini pour les réseaux 5G et le network slicing.

Le network operator (a) est client des network equipment providers (fournisseur d'équipements réseaux) et des virtualization infrastructure service providers (fournisseurs de services d'infrastructures virtuelles) qui lui fournissent toute l'infrastructure nécessaire et les équipements nécessaires au déploiement de son réseau et des network slices. Le virtualization infrastructure service provider n'est lui-même qu'un fournisseur de services qui s'appuie sur une infrastructure physique fournie par les NFVI Suppliers (fournisseur d'infrastructures virtuelles) et les data center service providers (opérateur de data centers) eux-mêmes clients des hardware suppliers (fournisseur d'équipements physiques). La pile de ces acteurs constitue le socle nécessaire pour que les network operators puissent instancier les network slices fournis dans le cas du NSaaS aux NSaaS providers (fournisseur de slices). Les NSaaS providers fournissent par la suite ces slices aux NSaaS customers (utilisateurs finaux des slices). Les NSaaS providers correspondent aux LINP operators (e) et les NSaaS customers aux end-users (b) dans le modèle LINP. Dans le mode d'usage Network slice as Network operator internal, le NSaaS provider correspond au communication service provider et le NSaaS customer au communication service customer.

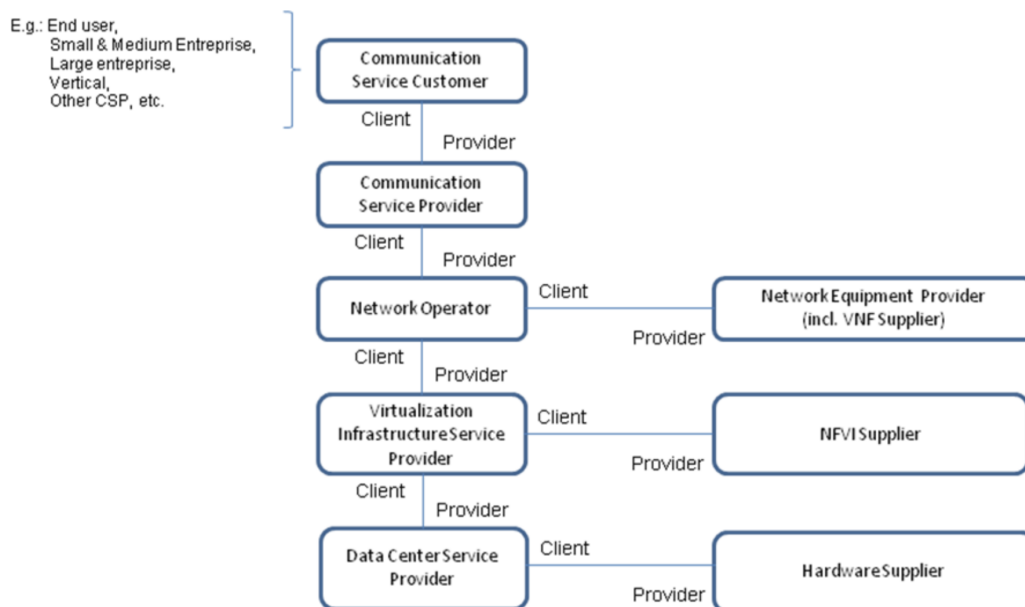


FIGURE 2.3 – Modèle de rôle dans la 5G [40]

Dans notre analyse et dans la suite de notre de travail, ce modèle de rôle défini par le 3GPP constitue le modèle de référence. Comparativement au modèle LINP, il ne modélise pas l'interopérabilité entre les network slices des différents network operators en introduisant des acteurs tels que le LINP exchanger (f). Néanmoins, ces interactions sont présentes au sein des réseaux 5G avec le roaming qui ne né-

cessite pas l'intervention d'un tiers pour effectuer la passerelle entre deux network operators. Les modèles de rôle proposés par les autres organismes et les activités de recherche sont similaires avec une terminologie qui diffère dans laquelle on retrouve régulièrement les slices tenant en tant que LINP operator et les slices providers en tant que end-users.

### 2.1.5 Architecture et Orchestration

L'ITU-T a introduit une architecture [99] haut niveau dans laquelle plusieurs blocs fonctionnels sont définis ainsi que les interfaces entre eux pour la gestion des LINP. Cette architecture s'appuie grandement sur la virtualisation [100] et la ségrégation des ressources réseaux en calcul, stockage et transport. Elle décrit un modèle d'orchestration pour les LINP et se réfère aux modèles et architectures sur le cloud [101] pour permettre l'orchestration multi-domaine [102]. Chaque LINP se voit attribuer un ensemble de ressources virtuelles et physiques qui sont gérées par des management fonctions dédiées. Cette architecture est générique et peut s'appliquer dans plusieurs contextes réseaux, elle reste cependant haut niveau et abstrait donc la complexité de gestion qui peut exister au sein de réseaux réels déployés. Cette architecture semble être reprise et précisée dans les réseaux 5G du 3GPP.

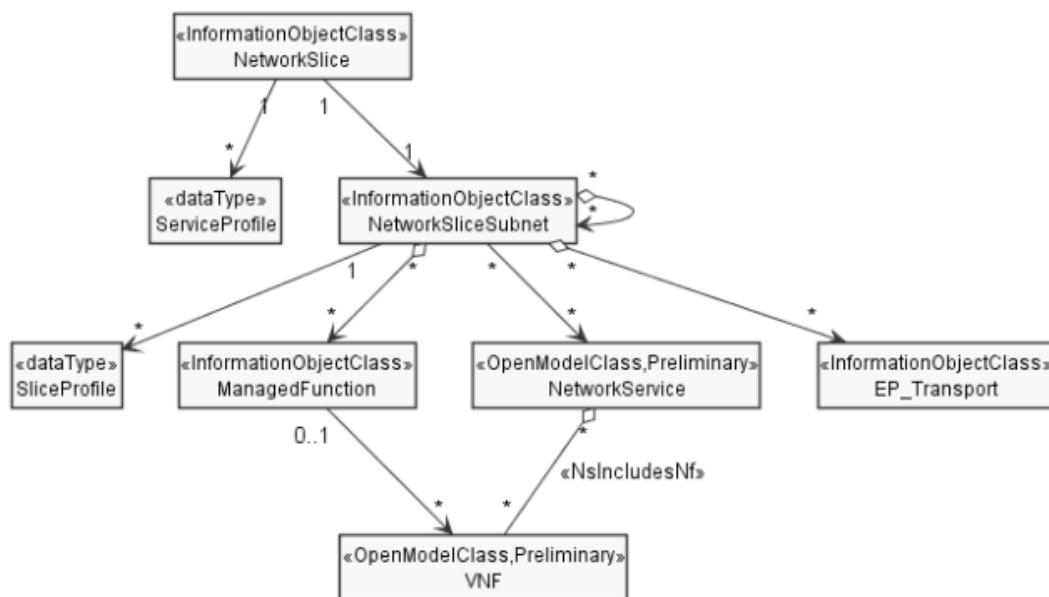


FIGURE 2.4 – Modèle relationnel du network slicing dans la 5G [103]

Le standard 5G aborde depuis la release 15 le network slicing sur plusieurs aspects, ne cesse d'évoluer et répond à plusieurs problématiques : l'architecture et l'orchestration du slice, la définition du plan de contrôle afin d'établir et maintenir les slices ainsi que le plan de données pour chaque slice. Il est défini comme un paradigme dans lequel des réseaux/partitions logiques sont créés, avec une isolation

des ressources et une topologie optimisée appropriées pour servir une catégorie de service (par exemple, un cas d'utilisation/une catégorie de trafic, ou pour des raisons internes à un network operator) ou des end users (système logique créé "à la demande") [3]. Chaque network slice est un objet réseau que l'on peut gérer qui possède ses propres attributs et qui répond à un besoin de connectivité bien spécifique. Un network slice est la composition de network slice subnets successifs qui sont eux-mêmes des objets réseaux indépendants répondant aux exigences d'une partie du réseau (e.g. CN, RAN). Les network slices et les network slice subnets sont respectivement instanciés au sein du réseau 5G en network slice instances et network slice subnet instances afin de répondre aux exigences des network operators ou des communication service providers.

La figure 2.4 représente le modèle relationnel décrivant le network slicing au sein du standard [103]. Le Service Profile définit les paramètres techniques de chaque slice, tels que leur latence ou leur bande passante. Chaque network slice subnet est également représenté par un Slice Profile qui spécifie de même ses paramètres techniques. Les Service Profile et le Slice Profile sont tous deux dérivés du GSMA Generic Network Slice Template (GST)/Network Slice Types (NEST) [104] (figure 2.5), qui est la référence pour le network slicing des réseaux mobiles en terme de spécifications techniques. Le Service Profile correspond au Service Level Specification (SLS) du slice tandis que le GST/NEST définit le Service Level Agreement (SLA) associé. Sur la base du Service Profile, les caractéristiques nécessaires pour le TN sont aussi dérivées et permettent de le configurer. Chaque network slice subnet est aussi composé d'un ensemble de network services lui-même composé de plusieurs VNFs. Les KPIs end-to-end sont définis pour l'ensemble du réseau 5G ainsi que pour le network slicing [88].

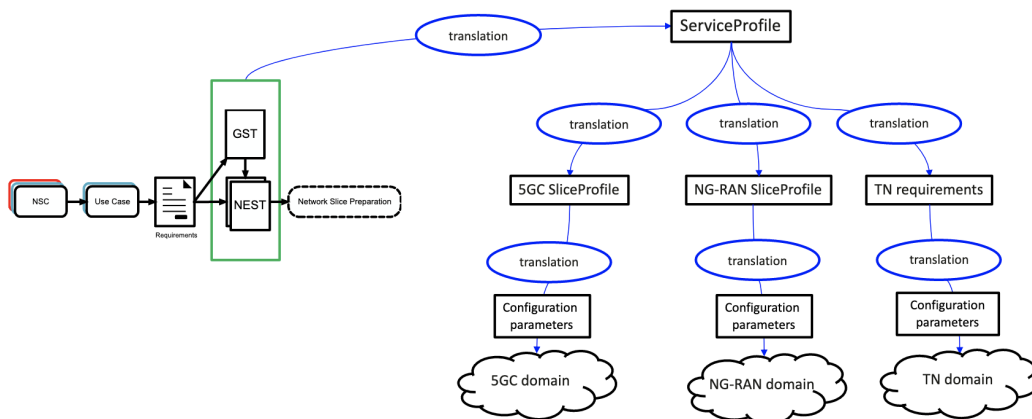


FIGURE 2.5 – Relation entre le GSMA GST et le Service Profile [104]

Chaque network slice doit pouvoir répondre à un ensemble de services hétérogènes qui nécessitent pour certains des architectures fonctionnelles dédiées et des protocoles de communications propriétaires. La 5G définit et standardise les architectures pour les quatre services suivants :



- enhanced Mobile Broadband (eMBB) : service nécessitant des communications à très haut débit (de l'ordre du Gbps) tel que le streaming vidéo ou la réalité augmentée pour système non critique. Ces services peuvent nécessiter de mettre en place du MEC pour augmenter les performances et améliorer la QoS. Les connexions sont bi-directionnelles et permettent d'envoyer du trafic dans les deux sens mais avec une quantité de trafic supérieure dans le sens descendant.
- Ultra-Reliable Low-Latency Communication (URLLC) : service nécessitant un temps de latence particulièrement faible ( $< 5$  ms) et une connectivité réseau extrêmement fiable ( $> 99.999\%$ ). On retrouve dans cette catégorie les services associés aux soins à distance, aux systèmes à réalité augmentée critiques et à l'industrie 4.0 avec les usines connectées.
- MIIoT : services nécessitant un très grand nombre de connexions tels que les capteurs connectés et les villes connectées. Les connexions sont bi-directionnelles mais le trafic est majoritairement montant.
- V2X : services pour véhicules connectés nécessitant un temps de latence extrêmement faible, une connectivité réseaux extrêmement fiable et une bande passante élevée [105]. L'architecture V2X est un cas spécifique du URLLC qui doit aussi permettre une communication Peer 2 Peer (P2P) entre les véhicules.

La flexibilité de l'architecture 5G permet de définir des architectures optimisées pour ces quatre types de service et d'instancier des network slices qui répondent à leurs exigences respectives.

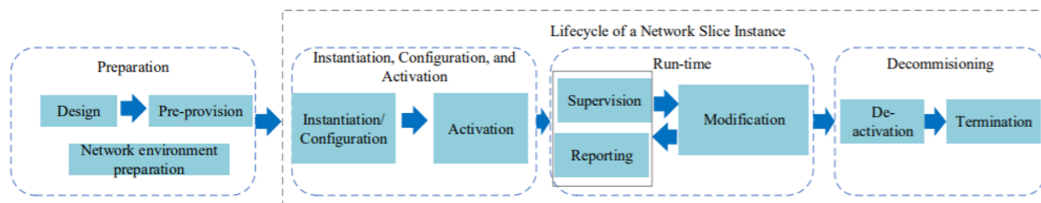


FIGURE 2.6 – Cycle de vie d'un slice défini par le 3GPP [40]

Un network slice étant un objet réseau indépendant et autonome, il possède un cycle de vie bien défini au sein de la 5G comme le montre la figure 2.6 qui liste l'ensemble des étapes de ce cycle de vie [106]. Chaque passage d'un état du cycle de vie à un autre est associé à une procédure [107] qui repose sur l'utilisation du framework d'orchestration orienté service SBMA (SBMA) du 3GPP [108]. Ce framework définit des Management Services (MnS) qui peuvent être de plusieurs types :

- MnS component type A : groupe d'opérations ou de notifications. Par exemple créer, lire, mettre à jour ou supprimer une entité réseau.
- MnS component type B : des entités qui peuvent être gérées et qui possèdent un modèle informationnel au sein du réseau 5G. Un network slice,



par exemple, est un composant de type B qui est aussi identifié comme étant un objet Network Resource Model (NRM) [109] décrit par un modèle informationnel.

- MnS component type C : informations sur les performances et défaillances de l'entité gérée.

Les MnS sont implantés dans des Management Functions (MnFs) qui communiquent entre elles par ce biais. Les MnS suivent le modèle producteur-consommateur et chacun expose des services qui peuvent être consommés par d'autres MnS. Dans la norme actuelle, peu de ces MnS sont spécifiés avec leurs procédures [106] (par exemple, l'instanciation d'un network slice ou d'un network slice subnet) et il est de la responsabilité du network operator 5G de les mettre en œuvre dans les MnFs. Le 3GPP définit les MnS components de type A, B et C ainsi que les procédures associées au network slices [106]. La spécification des MnFs qui implémenteraient ces MnS n'est pas encore présente sous forme de Technical Specifications (TS) mais sous forme de Technical Reports (TRs) qui précèdent les spécifications officielles du standard.

La phase 2 de la normalisation du network slicing a débuté dans la version 17 [110] et vise à améliorer le support du slicing des réseaux 5G et dans les nouvelles générations de réseaux mobiles.

L'ETSI reprend l'architecture du network slicing au sein de la 5G et propose une intégration avec son framework ETSI NFV MANO [27]. Un mapping est effectué entre le framework NFV MANO et les concepts de network slicing du 3GPP. Le bloc MANO permet d'orchestrer les MnS du réseau 5G en charge de la gestion des network slices. Il fait le lien entre les différentes procédures de gestion du cycle de vie des slices du 3GPP et les capacités du framework MANO à répondre à ces procédures. [111-127]

### 2.1.6 Plan de contrôle

On distingue le plan de contrôle du network slice et le plan d'orchestration au niveau du network slicing. Nous avons présenté le plan d'orchestration dans la section précédente. Nous détaillons ici le plan de contrôle du réseau qui permet aux end users de se raccorder aux slices et d'accéder aux services. La nouvelle SBA 5G [3], qui s'appuie sur les technologies de virtualisation et les approches de type "cloud" et SDN, permet d'atteindre un niveau de flexibilité sans précédent dans les réseaux mobiles. Cette flexibilité permet désormais de supporter le network slicing au sein du RAN [44, 85] et du CN. Les network slices sont identifiés dans le plan de contrôle par le Single - Network Slice Selection Assistance Information (S-NSSAI) qui est la composition du Slice/Service Type (SST) et du Slice Differentiator (SD). Le SST identifie le type de slice (tableau 2.2) et le SD complète cette information dans le cas de plusieurs types de slices identiques dans le même Public Land Mobile Network (PLMN). Un UE se raccorde à un ou plusieurs network slices en utilisant ses sessions PDU.

Lors de l'établissement d'une session PDU [128], le UE indique dans les mes-

TABLE 2.2 – Les slices standard de la 5G et les services associés

SST	Valeur	Services dédiés
eMBB	1	Services broadband
URLLC	2	Services à latence extrêmement faibles
MIoT	3	Services IoT
V2X	4	Services pour véhicules connectés

sages du plan de contrôle le S-NSSAI qui identifie le type de slice auquel il souhaite se raccorder. Le réseau identifie ensuite le slice grâce à la NSSF du CN dont le rôle est de sélectionner le network slice instance adéquat. Dans le cas du roaming, une correspondance est effectuée entre le S-NSSAI des différents network operators. Une fois la sélection faite, la SMF crée la session pour le UE et lui attribue un identifiant. Il va ensuite programmer l’User plane Function (UPF) située dans le plan de donnée à l’aide du protocole Packet Forwarding Control Protocol (PFCP).

### 2.1.7 Plan de données

Il n’existe aucun plan de données commun qui assurerait la continuité de slice entre les différents domaines réseaux, puisque chaque domaine réseau possède ses propres protocoles de communication et et nécessitent des passerelles entre chaque domaine. Des protocoles peuvent être utilisés pour faire la convergence au sein de chaque domaine comme par exemple dans les réseaux IP, où l’IETF propose une architecture de chaînage de fonctions [129]. Cependant, dans le contexte des réseaux mobiles, des protocoles sont déjà définis (e.g. GTP-U [130]) et permettent d’identifier les paquets utilisateurs dans le plan de données. Une réutilisation et une amélioration de ces protocoles pour supporter le network slicing serait donc plus judicieuse comme point de départ de travail pour une convergence généralisée. Au sein des réseaux 5G, un UE peut appartenir à un ou plusieurs network slices simultanément et, dans chaque session PDU, plusieurs flux peuvent être établis avec des exigences de QoS différentes. Chaque flux appartient à un QFI qui identifie le 5G QoS Identifier (5QI) associé. Les 5QI sont des valeurs normalisées avec des caractéristiques de QoS telles que le type de ressource (par exemple, Guaranteed Bit Rate (GBR), non-GBR) ou trip-time des paquets. Pour traiter les différents flux de QoS, le network slice subnet RAN met en correspondance chaque paquet avec les Data Radio Bearers (DRB) 5G. Le network slice subnet CN les multiplexe quant à lui dans des tunnels GTP-U qui mettent en œuvre la QoS de la pile IP (par exemple via DiffServ [131] ou les en-têtes d’extension GPRS Tunneling Protocol (GTP)).

Les travaux normatifs sur le network slicing sont toujours en cours et n’incluent encore pas tous les aspects, comme l’intégration et l’orchestration des différents segments réseaux formant le network slice end-to-end.

## 2.1.8 Documents de références

TABLE 2.3 – Documents de standardisation - Network Slicing

Spécification	Domaine	Contenu
<b>ITU</b>		
Y.3001 [93]	Définition	Définition de la virtualisation des ressources pour les réseaux du futur.
Y.3011 [94]	Définition	Définition et objectifs du concept LINP.
Y.3015 [99]	Architecture, Orchestration, Modèle de rôle	Présente une architecture LINP, un modèle rôle et un modèle d'orchestration.
Y.3012 [100]	Définition	Définition des besoins de l'architecture LINP.
Y.3502 [101]	Architecture, Orchestration	Architecture de référence cloud pour l'ITU et sur laquelle le modèle LINP s'appuie.
Y.3511 [102]	Orchestration	Framework d'orchestration entre domaines cloud distincts utilisé par LINP.
<b>GMSA</b>		
NG.116 [104]	Définition	Définition des cas d'utilisation du network slicing, les paramètres techniques des network slices pour chacun des cas d'usage.
NG.127 [132]	Définition	Agrégation des travaux effectués au sein des autres organismes de standardisation, définition du concept de network slicing, du modèle de rôle, de l'architecture haut-niveau, des caractéristiques des network slices.
<b>ETSI</b>		
GR NFV-EVE 012 [27]	Architecture	Correspondance entre le network slicing dans le standard 5G et le framework ETSI NFV MANO

3GPP		
TR 28.801 [133]	Architecture, Orchestration, Modèle de rôle	Rapport d'étude sur le network slicing, ses apports au sein des réseaux 5G, définition de l'architecture et du modèle de rôle.
TS 23.501 [3]	Architecture, Définition, Plan de contrôle, Plan de donnée	Définition de l'architecture des réseaux 5G, du concept de network slicing, ses objectifs et ses caractéristiques, spécification des services déployés au sein des network slices.
TS 23.502 [128]	Plan de contrôle, Plan de donnée	Définition des procédures d'établissement, de modification et de suppression de sessions PDU, définit les procédures de sélection de slice pour les UE.
TS 28.530 [40]	Architecture, Orchestration, Modèle de rôle	Définition des network slice comme étant la composition de network slice subnet et du TN, spécification des modes d'utilisation du network slicing et du modèle de rôle.
TS 28.531 [106]	Orchestration	Spécification des procédures de management des network slices, network slice subnet.
TS 28.532 [107]	Orchestration	Définition des opérations de management de service génériques.
TS 28.533 [108]	Orchestration	Définition des composants de services de management, des relations producteur-consommateur et des fonctions de management du réseau 5G.
TS 29.531 [134]	Architecture, Orchestration	Spécification des services de la Network Slice Selection Function.
TS 38.300 [44]	Architecture, Plan de contrôle, Plan de donnée	Architecture du NR et NG-RAN 5G, séparation de la gNB en CU et DU, définition des piles protocolaires, support du network slicing dans le RAN.
TS 38.413 [135]	Plan de contrôle	Spécification du protocole NGAP.
TS 28.541 [103]	Définition	Spécification des modèles informationnel pour les éléments du network slicing.

---

TR 28.811 [136]	Architecture, Orchestration	Propositions d'amélioration du système d'orchestration du network slicing pour la Release 17.
<b>ONF</b>		
TR-526 [98]	Architecture	Détaille l'intégration du paradigme SDN dans le network slicing et spécifie son usage au sein de ce dernier.
<b>NGNM</b>		
5G White Paper [95]	Définition	Définition et objectifs du concept du network slicing.
Slicing Concept [96]	Définition	Spécification de network slicing et définition de la relation entre les services, les network slices, les network slice subnets et l'infrastructure physique.
<b>IETF</b>		
RFC 7665 [129]	Plan de donnée, Plan de contrôle	Définition de l'architecture Service Function Chaining (SFC).

---

## 2.2 Caractéristiques d'un Network Slice

Les organismes de standardisation s'appuient sur de nombreux travaux de recherche préliminaires qui s'attelaient à définir le concept de network slicing [137-140]. De même, l'organisme européen 5G-PPP a initié un ensemble de projets H2020 dédiés à la 5G qui visaient dans une première phase à définir les réseaux 5G [141] puis, dans une seconde et troisième phase de les améliorer et les faire évoluer [142, 143]. Chaque projet se focalise sur des aspects bien spécifiques des réseaux de 5<sup>ème</sup> génération et génère des résultats scientifiques exploitables par les organismes de standardisation. Nous synthétisons dans le tableau 2.4 les projets qui ont été totalement ou partiellement consacrés au network slicing.

TABLE 2.4 – Projets 5G traitant du network slicing

Projet	Réalisations
NECOS	Définition d'une architecture d'orchestration NECOS pour le "network slicing as a service" dans le cloud. Cette architecture est orientée orchestration multi-domaine et a pour objectif de déployer des services au sein d'infrastructures virtualisées distinctes et de les orchestrer.
SliceNet [144]	Définition d'une architecture SliceNet permettant de déployer des slices end-to-end au sein de plusieurs domaines.
5G SONATA	Utilisation de SDN/NFV pour déployer des services end-to-end au dessus d'infrastructure virtualisée.

L'ensemble de ces travaux a permis d'identifier les caractéristiques des network slices, d'en extraire leurs propriétés et les nombreux défis qu'il revient de relever afin de pouvoir mettre en œuvre le network slicing. Nous avons synthétisé et condensé ces propriétés dans la section 2.2.1 et listé tous les défis associés dans la section 2.2.2.

### 2.2.1 Propriétés des slices réseaux

Les network slices possèdent une multitude d'attributs qui les caractérisent. Ils doivent supporter l'aspect end-to-end (section 2.2.1.1), posséder un haut niveau d'automatisation (section 2.2.1.2), être programmable (section 2.2.1.3), supporter les aspects de passage à l'échelle et d'élasticité (section 2.2.1.4), offrir plusieurs niveaux d'isolation (section 2.2.1.5), offrir plusieurs niveaux de sécurité (section 2.2.1.6) et permettre une abstraction hiérarchique pour la gestion du slice (section 2.2.1.7).

### 2.2.1.1 Bout-en-bout (End-to-end)

L'aspect end-to-end est une propriété inhérente au slicing qui permet à un fournisseur de service de déployer et garantir des services à un utilisateur final. Un slice est instancié au dessus d'une infrastructure mutualisée qui combine plusieurs segments réseaux. Le service doit donc être garanti dans chacun des segments réseaux composant le slice. Chaque segment réseau appartient à un domaine administratif et possède une gestion autonome, les acteurs qui fournissent l'infrastructure réseau au sein de ces domaines sont différents. Les ressources qui sont allouées et gérées au sein de chaque segment sont hétérogènes et doivent être orchestrées de manière cohérente pour que les KPIs du slice soient respectés. Une communication au niveau plan de contrôle entre les différents domaines peut alors s'avérer nécessaire. Néanmoins, le découpage du slice en plusieurs domaines administratifs permet aux acteurs de ces domaines d'orchestrer ces ressources de manière efficace, d'en optimiser l'allocation ainsi que leur utilisation. Dans le cas des réseaux mobiles 5G, les différents segments réseaux au dessus desquels sont déployés les slices sont : le Radio Access Network (RAN), le réseau de transport, le Core Network (CN) ainsi que le réseau cloud du Data Network.

### 2.2.1.2 Automatisation (Automation)

La création, la modification et la suppression du slice doivent être effectuées de manière automatique sans intervention humaine. En plus du traditionnel SLA, des mécanismes additionnels doivent venir s'ajouter afin de permettre à des tiers de demander la création de slices avec des caractéristiques techniques qui peuvent énormément varier sans qu'il n'y ait de modification de SLA (e.g. latence, gigue, nombre d'utilisateurs supportés, bande passante, durée de vie du slice, garanties de qualités de services). Cette automatisation permet une réduction des Operating Expenses (OPEX) et des durées d'exécution qui améliorent la réactivité lorsqu'un événement au sein du slice survient.

### 2.2.1.3 Programmabilité (Programmability)

Un slice doit être programmable par des tiers via des API. Ces interfaces servent à exposer les capacités du slice et d'interagir avec lui. Ces API peuvent être ouvertes ou fermées selon la relation commerciale entre les différents acteurs économiques impliqués. En utilisant ces API, un tiers doit être en mesure de modifier le comportement du slice de manière dynamique (e.g. instancier une NF au sein d'un slice, augmenter la quantité de ressources allouées globalement au slice ou spécifiquement à une NF).

### 2.2.1.4 Élasticité (Elasticity)

L'un des objectifs majeurs avec l'introduction des slices dans les réseaux est d'augmenter considérablement la flexibilité dont dispose les fournisseur de ser-

vices ainsi que les opérateurs réseaux pour la gestion des services et des réseaux. Un slice possède donc par définition la propriété "d'élasticité" qui est une opération essentielle au niveau des ressources afin d'assurer le SLA associé au slice dans des conditions opérationnelles qui peuvent varier. En fonction des conditions et de l'état du réseau, il peut être nécessaire de modifier de façon temporaire la quantité de ressources allouées au sein d'un network slice (e.g. absorber un pic de trafic inattendu, augmenter les ressources pour corriger une dégradation du lien radio). Cette opération peut être manuellement déclenchée par un des acteurs impliqués dans la gestion du slice ou par des mécanismes automatiques. Cette opération peut aussi s'effectuer dans le cadre de mise à niveau du slice en reprogrammant une NF, en instanciant une nouvelle NF, en ajoutant une technologie radio dans le RAN ou en ajustant la politique de gestion du slice pour une durée conséquente. Cette dynamique est en partie rendue possible par le découpage en sous-domaines du slice ; chaque acteur du domaine possède une parfaite maîtrise de ses ressources et est donc en mesure d'effectuer ces opérations de manière optimale. L'introduction de nouvelles technologies telles que SDN et NFV permettent aussi une plus grande flexibilité dans la gestion des ressources. En effet, les ressources allouées à un slice peuvent être physiques ou virtuelles ce qui permet une gestion plus fine de ces ressources. Néanmoins, tous les slices partagent la même infrastructure sous-jacente ce qui complexifie les opérations de modification de slice et peuvent nécessiter une communication inter-slice pour assurer une gestion cohérente de ces derniers.

#### 2.2.1.5 Isolation (Isolation)

Chaque slice au sein du réseau est indépendant, possède ses propres règles de gestion et ses ressources. L'isolation entre slices est donc une propriété fondamentale du network slicing qui permet une totale autonomie des slices entre eux. Cette isolation, présente au niveau du plan de contrôle et du plan de données, permet d'assurer les garanties de performance et de sécurité pour chaque fournisseur de service. Les ressources allouées au slice peuvent être dédiées, partagées ou mixtes. Le choix du type de ressources dépend du SLA associé au slice et assure une isolation plus ou moins forte des différents slices. Ce choix a un impact direct sur les performances et sur la flexibilité dont dispose un tiers pour gérer le slice.

#### 2.2.1.6 Sécurité (Security)

Additionnés aux mécanismes d'isolation, le slice doit pouvoir offrir plusieurs niveaux de sécurité. L'accès à la gestion du slice ainsi que le raccordement des utilisateurs doivent être sécurisés. De plus, les données circulant au sein de chaque slice ne doivent être accessibles qu'au propriétaire du slice et peuvent faire l'objet de chiffrement bout-en-bout en instanciant des NFs dédiées.



### 2.2.1.7 Abstraction hiérarchique (Hierarchical Abstraction)

L'abstraction hiérarchique est une propriété du slicing qui trouve ses racines dans la virtualisation récursive, dans laquelle la procédure d'abstraction des ressources est répétée selon un modèle hiérarchique, offrant une plus grande abstraction pour les différents acteurs et qui permet de cibler précisément leurs besoins. En d'autres termes, les ressources d'un slice, allouées à un tiers, peuvent être ré-allouées en partie ou en totalité à un autre acteur tiers qui exploiterait aussi le même slice. Par exemple, un Mobile Virtual Network Operator (MVNO) qui a acquis un slice auprès d'un fournisseur d'infrastructure, offre une quantité partielle de ces ressources pour permettre à un fournisseur de services publics utilisant son réseau virtuel d'instancier en son sein un slice IoT.

## 2.2.2 Les défis associés

En se fondant sur les paramètres définis précédemment pour le network slicing, nous avons extrait les nombreux enjeux et défis qu'il est nécessaire de résoudre. Ces enjeux sont associés aux ressources, à la QoS, à l'isolation, à l'orchestration, à la programmabilité, à la sécurité et au concept end-to-end des network slices. Pour chacun de ces enjeux, nous présentons les solutions scientifiques qui ont été proposées et effectuons une claire distinction entre les solutions proposées pour le RAN et le CN. De plus, nous positionnons les technologies SDN et NFV pour chacun des défis et montrons en quoi ils sont des composants nécessaires pour permettre le network slicing.

### 2.2.2.1 Garantir l'isolation des ressources (i)

Les ressources au sein de chaque network slice doivent être isolées et gérées indépendamment les unes des autres. Au sein du RAN, plusieurs solutions ont été proposées telles que l'utilisation de méthodes basées sur des problèmes d'optimisations [145-148], de la théorie des enchères [149], et de la théorie des jeux [150-152]. Des solutions basées sur des algorithmes avancées pour les ordonnanceurs [153] sont aussi largement utilisées.

L'utilisation des architectures Cloud-RAN permet de segmenter les ressources consommées au sein du RAN 5G et d'appliquer des techniques d'isolations différentes. On peut donc utiliser des hyperviseurs tels que Xen, ESXi, VMware pour l'isolation des ressources utilisées par les Central Units (CUs).

Ces techniques se basent sur la virtualisation des ressources qui s'opère aussi dans le cloud au niveau des ressources de calcul, de stockage et de mémoire vive. On retrouve l'utilisation de systèmes de gestion d'infrastructures virtualisées plus tels que VMware vSphere ou OpenStack. Ce sont ces mêmes systèmes qui sont utilisés pour l'isolation des ressources dans le CN. On peut y adjoindre les nouveaux systèmes de conteneurisation tels que Docker Swarm ou Kubernetes qui permettent d'isoler les ressources en utilisant les namespaces linux.

NFV est essentiel pour la réalisation de cette isolation car il ajoute une couche d'orchestration afin de gérer de manière optimale le partage des ressources de l'infrastructure et leurs liens avec les hyperviseurs.

#### 2.2.2.2 Permettre un passage à l'échelle des ressources (ii)

Les mécanismes d'élasticité doivent s'appliquer au slice et donc il doit être possible d'effectuer des modifications sur les ressources allouées au network slice.

Dans le RAN, plusieurs aspects doivent être pris en compte pour permettre ce passage à l'échelle. Il n'est réalisable que dans le cas d'utilisation d'architecture Cloud-RAN ou avec des gNBs implantant la séparation au niveau de l'interface F1. En effet, il est primordial de pouvoir modifier indépendamment les ressources radios puis les ressources logiciels qui sont deux types de ressources disjointes. La quantité de ressources radios allouées au slice peut être modifiée par un hyperviseur qui permettrait l'utilisation d'un système de virtualisation radio [50]. La reconfiguration d'ordonnanceur [153, 154] permet aussi d'allouer plus de bande passante si nécessaire à une communication en particulier. L'instanciation supplémentaire d'éléments radios dans le réseau permet aussi d'augmenter la quantité de ressources disponible administrée par l'hyperviseur.

Au sein du CN, le passage à l'échelle des ressources est conditionné par l'utilisation d'infrastructures virtualisées pilotées par des solutions d'orchestration NFV tels que OSM ou ONAP. Ces solutions d'orchestrations doivent mettre en œuvre une politique de gestion de ressources qui soit cohérente avec la disponibilité des ressources ainsi que les besoins respectifs de chaque network slice. L'architecture SBA du CN 5G a été pensée afin d'exploiter pleinement les mécanismes d'élasticité déjà très largement déployés dans le cloud par les fournisseurs de services. SDN est couplé à NFV afin de gérer les flux entre les nouvelles instances de fonctions réseaux qui résulteraient du passage à l'échelle des slices.

#### 2.2.2.3 Support de différent niveaux de QoS au sein du slice (iii)

Un network slice étant dédié à un type de service, il peut supporter en son sein une multitude de politiques de QoS. Il faut pour cela distinguer les mécanismes qui interviennent dans le plan de contrôle et dans le plan de données. Le projet SliceNet dans sa définition d'architecture a proposé un plan de contrôle dédié au network slicing dans la 5G afin de gérer pour chaque slice les politiques de QoS associées [155]. Cependant, même avec un plan de contrôle bien défini, il faut que le plan de données soit en mesure d'effectuer une double classification du trafic et de le traiter en fonction des politiques de slice dans un premier temps puis de QoS dans un second temps. Que ce soit dans le RAN ou dans le CN, les équipements de données qui implantent les algorithmes de priorités doivent supporter la classification à deux niveaux. Par exemple, un ordonnanceur radio dans le RAN effectuerait une première priorisation gros grain pour allouer les ressources adéquates à chaque network slice. Puis, dans un second temps, pour chaque ressource

allouée, il effectuerait à nouveau un ordonnancement afin de prioriser à nouveau le trafic au sein du slice. Il en est de même pour les systèmes de virtualisation dans le CN.

#### 2.2.2.4 Gestion indépendante de la politique de QoS pour chaque slice (iv)

Les politiques de QoS doivent être spécifiques à chaque network slice et ne pas impacter les autres slices qui partagent la même infrastructure. Plusieurs aspects doivent être mis en concurrence ici : l'orchestration, le support de la QoS au sein des slices et le respect de l'isolation des ressources. Chaque slice tenant doit pouvoir disposer d'API lui permettant d'interagir avec le slice dont il dispose. C'est ce qui a été introduit par le projet NECOS dans son architecture de network slice as a service [156] dans laquelle chaque tenant possède un contrôle total de son network slice. Le slice tenant définit les services qu'il veut déployer ainsi que la QoS associée à ces services puis les instancie au sein d'une infrastructure répartie sur plusieurs domaines administratifs différents. Cette architecture est dédiée à la pure gestion de slices au sein d'une infrastructure cloud sans tenir compte du RAN, néanmoins, on peut étendre cette approche et ajouter des composants de contrôle qui permettraient dans l'infrastructure physique de configurer des ressources radios en se basant sur la configuration du slice tenant. L'orchestration fournie par NFV permet de segmenter les différentes ressources allouées aux network slices.

#### 2.2.2.5 Continuité de QoS au sein au sein des segments composant le slice end-to-end (v)

Le network slice est composé de segments réseaux appartenant à différents domaines administratifs. Chaque flux au sein du slice doit être clairement identifié et la QoS associée au flux doit être préservée au sein de chaque domaine. Des mécanismes doivent être déployés afin d'appliquer les différentes politiques de QoS au sein de chaque domaine réseau ainsi qu'aux points d'interconnexions. SDN et NFV ne fournissent pas de mécanismes à eux seuls qui permettent d'assurer cette continuité de service. Des mécanismes additionnels doivent être ajoutés et c'est pourquoi nous focalisons une partie de notre travail sur cet aspect.

#### 2.2.2.6 Gestion indépendante des slices (vi)

De même que pour la gestion indépendante de la politique de QoS pour chaque slice, l'administration des network slices doit être exclusive au slice tenant. On peut utiliser les mêmes techniques afin d'assurer une indépendance de gestion entre chaque slice.

### 2.2.2.7 Indépendance dans la gestion des segments réseaux composant le slice (vii) et intégration transparente (viii)

L'indépendance entre segments réseaux composant le network slice doit être préservée. Les réseaux appartenants à des domaines administratifs différents doivent rester indépendants les uns les autres en termes de gestion et d'infrastructure sous-jacentes. En effet, les nombreux travaux sur l'orchestration multi-domaines ont démontré l'impossibilité de converger vers une gestion unique des domaines réseaux différents pour des raisons de modèle de rôle et d'hétérogénéité technologiques.

De plus, il est aussi impossible de développer et déployer des systèmes d'interconnexions qui prennent en charge tous les domaines réseaux avec leurs technologies propres et spécifiques. Il faut donc qu'il y ait une convergence au niveau des systèmes d'orchestration afin de permettre une communication uniforme et ainsi obtenir une intégration transparente et cohérente entre les systèmes. Dans la suite de nos travaux, nous prendrons comme référence les systèmes déployés pour les réseaux 5G et donc s'adapter à ces systèmes.

Dans notre méthode d'intégration, nous apportons des solutions à ces contraintes.

### 2.2.2.8 Gestion autonome et automatisation du cycle de vie (ix) et déploiement automatique (x)

Les opérations liées au cycle de vie d'un network slice doivent être automatisées. Cette automatisation passe par des outils de déploiement de réseau tels que les orchestrateurs OSM ou ONAP. Pour ce faire, il est nécessaire de se conformer aux différents formats des orchestrateurs et modéliser l'ensemble des composants du network slice selon les standards de ces outils. Afin de pouvoir utiliser ces plateformes efficacement, l'adoption des technologies NFV est primordiale pour permettre une certaine flexibilité dans le déploiement, le maintien et la suppression du slice. Le CN 5G a été spécifié pour ce type d'usage et donc est nativement compatible avec les outils de gestion cloud. Un effort d'intégration avec les orchestrateurs est cependant nécessaire afin de normaliser les composants du CN en VNF standards. L'automatisation au sein du RAN quant à lui est une opération plus complexe puisqu'elle implique un déploiement de ressources radios qui ne sont pas situés sur des VNFs mais des Physical Network Function (PNF).

L'automatisation des réseaux mobiles a été introduit au sein des réseaux 4G avec les Self-Organizing Network (SON) qui ont introduit les notions de self-configuration, self-optimisation, self-healing, self-protection et l'Automatic Neighbor Relations (ANR) afin de permettre une automatisation et un début d'autonomie. Au sein de la 5G, l'arrivée du Machine Learning et des techniques d'Intelligence Artificielle permettent d'améliorer les SON qui sont ré-utilisés au sein du network slicing [157]. Des propositions ont été formulées afin d'utiliser les techniques d'Intelligence Artificielle dans l'allocation de ressources pour les network

slices [158].

#### 2.2.2.9 Programmation du slice (xi)

Afin de permettre une manipulation des ressources au sein de chaque network slice, des API doivent être mise à disposition pour le slice tenant. SliceNet et NE-COS exposent dans leurs architectures des API nord qui permettent au tenant d'interagir avec son slice et de changer sa configuration de manière dynamique sans qu'il n'y ait de modification de SLA.

SDN joue ici un rôle central dans la programmation des réseaux car il permet d'exposer des API aux applications de gestion de réseau afin qu'elles puissent dynamiquement modifier son comportement.

#### 2.2.2.10 Avoir des mécanismes de sécurité dédié pour chaque slice (xii)

Les mécanismes qui permettent d'assurer la sécurité de chaque network slice impliquent l'utilisation de processus Authentication, Authorization and Accounting (AAA) pour gérer l'accès des utilisateurs ainsi que du traitement dans le plan de données. Les paquets qui circulent dans le plan de données peuvent aussi bénéficier de protocoles de chiffrement sécurisés tels que IPSec afin de garantir au tenant un niveau de sécurité supplémentaire.

#### 2.2.2.11 Assurer les performances du slice de bout-en-bout (xiii)

Assurer les KPIs d'un network slice est une opération coûteuse et extrêmement complexe. Il faut garantir les performances du slice ainsi que respecter les exigences de QoS qui se mesurent du serveur applicatif jusqu'à l'utilisateur. Le network slice se décompose en plusieurs segments réseaux appartenant à différents domaines administratifs, ce qui nécessite de l'orchestration multi-domaine ainsi qu'une traduction des exigences de performances pour chaque domaine réseau. A ce jour, il n'y a que très peu de contributions qui proposent des solutions scientifiques et techniques pour assurer que la composition des segments RAN, TN et CN respectent les KPIs des slices. L'IETF a proposé une architecture dans laquelle des éléments d'interconnexion "slice gateway" sont introduits afin de relayer le trafic d'un segment vers un autre [159]. Cette architecture n'indique cependant pas comment les ressources sont modélisées au sein de chaque segment, comment les KPIs end-to-end sont dérivés et sont traduits pour chaque domaine réseau et de quelle manière on peut superviser les performances du slice.

Nous proposons dans notre modèle d'intégration des éléments qui répondent à ce défi.

## 2.3 Conclusion

Le concept de network slicing est un paradigme réseau dans lequel chaque network slice est un objet réseau autonome et indépendant avec ses propres caractéristiques. Ce paradigme vient s'ajouter aux concepts réseaux existants pour répondre aux problématiques de services end-to-end dans les réseaux et d'optimiser la gestion de ces derniers. Sous l'impulsion du développement des réseaux de 5<sup>ème</sup> génération, les travaux de recherche se sont multipliés ainsi que les travaux des organismes de standardisation. Ils abordent tous les aspects du network slicing de l'orchestration jusqu'à la définition du plan de contrôle et du plan de données des network slices.

Le network slicing introduit la notion de slice end-to-end déployé et composé de plusieurs segments réseaux appartenant à des domaines administratifs réseaux différents. La nature end-to-end des network slices nécessite de spécifier pour chaque domaine réseau ce concept, ce qui complexifie la convergence vers une architecture fonctionnelle commune pour le network slicing. En effet, les types de segments composant le network slice peuvent être totalement hétérogènes et donc les technologies sous-jacentes employées aussi. Les segments peuvent être de type cloud, Transport Network (TN), Core Network (CN) ou Radio Access Network (RAN). Dans le CN, il s'appuie majoritairement sur les avancées technologiques telles que la virtualisation et la programmation des réseaux. Ces technologies sont aussi reprises dans le RAN et notamment l'architecture Cloud-RAN qui les combine avec les technologies New Radios (NRs). Néanmoins, nous avons montré la difficulté à pouvoir seulement orchestrer ces deux types de segments conjointement, même s'ils utilisent des technologies similaires sous-jacentes. Ces difficultés sont d'autant plus grandes lorsqu'il s'agit d'intégrer les domaines cloud et TN.

C'est la difficulté majeure pour l'aboutissement du network slicing doit être implanté dans plusieurs domaines réseaux qui s'appuient sur des technologies différentes. Il est donc aujourd'hui nécessaire d'identifier et définir les points communs entre les différents types de réseau afin de converger vers une architecture fonctionnelle commune qui permette le déploiement et l'orchestration de network slice end-to-end au travers de tous les segments réseaux. Les travaux de recherche sont majoritairement effectués dans le contexte des réseaux 5G ce qui circonscrit le cercle des recherches et réduit les possibilités de généralité avec les autres réseaux. Il serait donc bénéfique de considérer le network slicing indépendamment des réseaux mobiles et définir une architecture qui soit commune et qui permette le déploiement de network slices au sein de tout type de réseau.

La flexibilité d'un réseau satellite supportant le network slicing offre des possibilités d'intégration multiples avec les réseaux mobiles. C'est dans cette démarche que nous appliquons le network slicing au réseau satellite, analysons la compatibilité des réseaux actuels avec ce paradigme et proposons des solutions afin de pouvoir instancier des network slices satellites. Nous confrontons dans le chapitre suivant le réseau satellite à une partie des défis du network slicing que nous avons résumé dans le tableau 2.5. Dans les chapitres 4 et 5 nous focalisons nos contri-

TABLE 2.5 – Défis associés au Network Slicing

Défis	Référence
Isolation des ressources	i
Passage à l'échelle des ressources	ii
Support de la QoS au sein du slice	iii
Gestion indépendante la politique de QoS pour chaque slice	iv
Continuité de QoS au sein au sein des segments composant le slice end-to-end	v
Gestion indépendante des slices	vi
Indépendance dans la gestion des segments réseaux composant le slice	vii
Intégration transparente	viii
Gestion autonome et automatisation du cycle de vie	ix
Déploiement automatique	x
Programmation du slice	xi
Mécanisme de sécurité dédié pour chaque slice	xii
Assurer les performances du slice de bout-en-bout	xiii

butions sur la résolutions des défis (v), (vii), (viii) et (xiii) qui sont nécessaire à l'intégration du satellite dans la 5G.

# Vers une architecture satellite supportant le network slicing

---

## Sommaire

---

<b>3.1</b>	<b>Le network slicing pour les réseaux satellites</b>	<b>58</b>
<b>3.2</b>	<b>Allocation des ressources satellites</b>	<b>59</b>
<b>3.3</b>	<b>Mise en commun des infrastructures satellites</b>	<b>62</b>
<b>3.4</b>	<b>Une architecture satellite plus flexible</b>	<b>63</b>
3.4.1	Gateway Satellite	63
3.4.2	Satellite Core Network	65
<b>3.5</b>	<b>Gestion et orchestration du network slice satellite</b>	<b>66</b>
3.5.1	Modèle de rôle pour le network slicing satellite	67
3.5.2	Satellite Management System	68
3.5.3	Slice Management System	70
<b>3.6</b>	<b>Conclusion</b>	<b>71</b>

---

Dans ce chapitre, nous appliquons le paradigme de network slicing au réseau satellite afin d'augmenter la flexibilité dans la gestion des réseaux satellites et de pouvoir les intégrer dans les réseaux 5G. Le but ultime de cette confrontation est l'instanciation de network slices satellites. Nous confrontons le réseau satellite aux enjeux du network slicing, proposons des solutions à ces défis afin d'obtenir l'instanciation d'un network slice satellite.



### 3.1 Le network slicing pour les réseaux satellites

Dans le chapitre 2, nous avons décrit le network slicing du réseau 3GPP et les composants utilisés pour créer et gérer des network slices. Le réseau de transport utilisé dans la 5G peut être de n'importe quel type. Dans ce contexte, le réseau satellite est un candidat idéal pour servir de réseau de transport entre le RAN et le CN 5G. Il possède une zone de couverture planétaire qui permet d'établir des communications longue distance sans fil et sans avoir à re-déployer des infrastructures au sol. L'intégration du réseau satellite en tant que réseau de transport est également la première étape pour intégrer le satellite dans les réseaux 5G afin de pouvoir envisager une intégration en mode d'accès direct, comme nous le montrons dans le chapitre 1. La définition d'API entre le système de gestion 3GPP et les réseaux de transport n'est pas suffisant pour configurer le réseau satellite afin de gérer les flux des network slices et les données de backhaul entre le RAN 5G et le CN 5G. (voir la figure 3.1). En tant que réseau de transport pour le backhauling 5G, le réseau satellite couvre plusieurs gNB 5G (par le biais du Satellite Terminal (ST)) et s'interconnecte avec divers UPF 5G (en utilisant la gateway satellite). Cela implique de réserver des ressources adéquates sur les équipements terrestres et spatiaux (e.g. les Gateway (GW) satellites et les charges utiles des satellites). L'allocation des ressources est l'un des principaux défis pour le réseau satellite en raison de l'hétérogénéité des ressources (e.g. le calcul, le stockage, la radio dans le segment terrestre et le segment spatial). Les ressources peuvent être spécifiques au système (e.g. bande passante, fréquence) ou alors génériques (e.g. calcul, stockage) et être allouables dans chacun des segments satellites. Plus que la réservation et l'allocation des ressources, les équipements doivent être configurés et orchestrés de manière simple, automatisée et complète. Le réseau 5G doit pouvoir demander dynamiquement l'établissement d'une connectivité dédiée et la modifier ultérieurement en fonction de ses besoins.

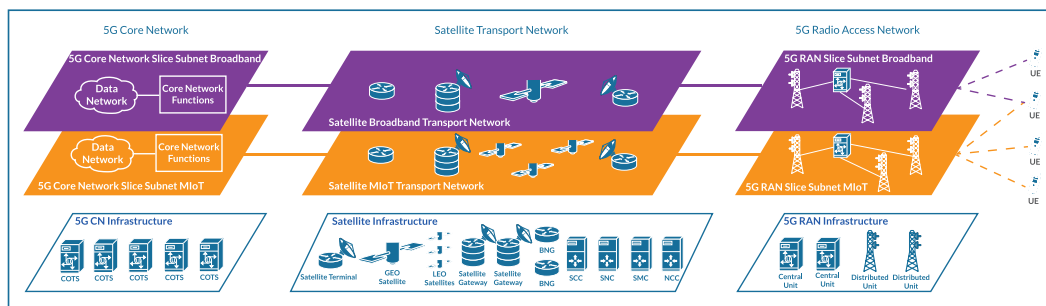


FIGURE 3.1 – Slice satellite intégrée dans le réseau 5G

Les réseaux satellites, tels qu'ils sont construits aujourd'hui, ont la capacité d'acheminer des données 5G, par exemple par encapsulation dans le protocole IP ou par d'autres méthodes de tunnellation. Néanmoins, ils ne sont pas en mesure de supporter pleinement la dynamique induite par le paradigme de network slicing, comme l'allocation dynamique des ressources et le passage à l'échelle. L'ar-

architecture des réseaux satellites doit évoluer afin de dépasser leurs limites actuelles en termes de performance et de gestion. Les technologies SDN et NFV, pierres angulaires du network slicing, peuvent contribuer à la modernisation du réseau satellite. Ces deux technologies sont également les composants de base pour augmenter la flexibilité des réseaux 5G et sur lesquels repose le paradigme du network slicing. Les réseaux satellites doivent donc être capables de supporter ce concept, ce qui est scientifiquement et techniquement complexe. Des questions fondamentales doivent être abordées et des défis majeurs doivent être résolus. Nous reprenons les défis listés dans la section 2.2.2 pour l'application du network slicing, et confrontons les réseaux satellites à ces derniers afin de proposer ensuite des lignes directrices pour les résoudre. Les réseaux satellites seraient alors en mesure de déployer des connectivités personnalisées de manière automatisée, rapide et évolutive sur n'importe quel point du globe leur permettant d'atteindre une flexibilité inédite ainsi que des performances accrues. Ces performances permettent d'envisager une intégration, comme nous le détaillons ici avec les réseaux 5G, mais aussi des réseaux IoT et les réseaux mobiles de future génération. Nous classons tous les défis identifiés dans le chapitre précédent auxquels nous allons apporter des réponses dans le tableau 3.1.

TABLE 3.1 – Catégorie des défis pour le slicing du réseau satellite

Catégorie	Défis
Isolation	i, ii
Management et Orchestration	iv, vi, vii, viii
Autonomie et automatisation	ix, x
Quality of Service (QoS)	iii, v
Programmabilité	xi
Sécurité	xii

L'objectif final est de pouvoir "découper" le réseau satellite en partitions dynamiquement instanciées et entièrement optimisées pour une connectivité entre le CN et RAN 5G. La figure 3.1 montre un tel partitionnement et donne l'exemple de deux partitions satellites instanciées au-dessus de l'infrastructure satellite. Le network slice supérieur est dédiée à la 5G eMBB en utilisant un satellite Geostationary Orbit (GEO) et le network slice inférieur est utilisé pour la MIoT avec des satellites Low Earth Orbit (LEO).

## 3.2 Allocation des ressources satellites

L'isolation des network slices est une condition essentielle pour l'application du network slicing. Un network slice doit offrir plusieurs niveaux d'isolation afin de répondre aux différents besoins du marché. Ce processus d'isolation est le ré-

sultat de l'isolation qui a lieu aux différents niveaux du modèle OSI. Dans cette partie, nous nous concentrons sur l'allocation et l'isolation des ressources.

L'allocation des ressources au sein du réseau satellite est un processus complexe ce qui en fait un sujet de recherche fondamental qu'il est nécessaire de résoudre. Dans la section 1.3.1, nous avons détaillé l'architecture des réseaux satellites qui comprend trois segments : le segment terrestre, le segment spatial et le segment utilisateur. Non seulement les ressources doivent être réservées sur chaque segment, mais les systèmes satellites peuvent aussi être très différents les uns des autres. Les segments terrestres peuvent implanter des normes telles que DVB-S2/DVB-RCS2 ou utiliser des normes propriétaires. Les capacités du segment spatial sont principalement définies par l'orbite du satellite et la capacité de la charge utile (e.g. satellites GEO ou méga-constellation), ce qui influe par conséquent sur la flexibilité globale du système. La figure 3.2 montre la nécessité de réserver des ressources sur chaque segment de satellite pour un network slice end-to-end. L'UPF 5G dessert plusieurs RAN 5G situés dans un ou plusieurs spot-beams satellites. Cela implique d'allouer et de partager les ressources sur les liaisons aller et retour du lien satellite. Chaque spot-beam peut être desservi par une ou plusieurs GW satellite. Ainsi, pour un seul network slice end-to-end, des ressources pourraient être réservées et orchestrées sur plusieurs passerelles satellites.

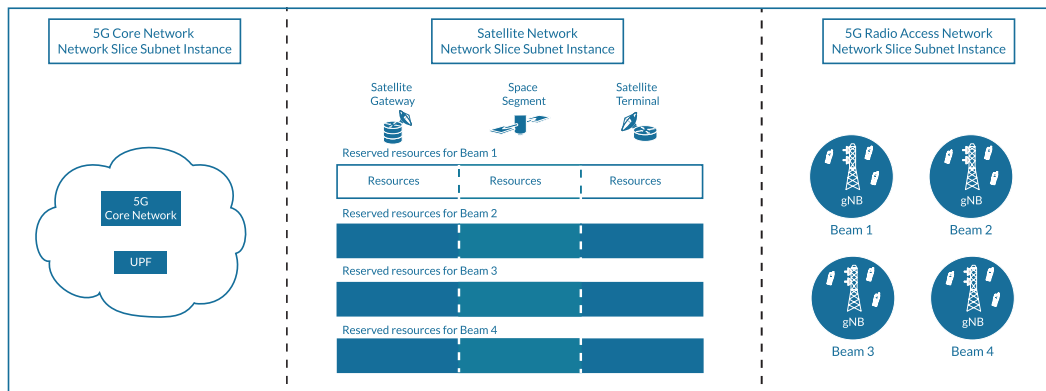


FIGURE 3.2 – Réserve de ressources sur le segment satellite pour le backhauling 5G

Le network slice satellite doit offrir plusieurs types d'allocation et d'isolation des ressources au niveau de la couche infrastructure [enjeu i]. Plusieurs types de ressources sont impliqués sur chaque segment lorsqu'une communication par satellite est établie, comme les NFs ou les ressources radio. Pour isoler correctement ces ressources, nous devons modéliser et définir précisément la granularité des ressources disponibles sur chaque segment (e.g. une porteuse et une bande passante pour la gateway satellite pour le forward link ou une porteuse sur le return link pour le ST). Pour chaque segment satellite, nous devons identifier le type de ressource qui pourrait être réservé et isolé. Le segment terrestre, tel que

décrit dans la section 1.3.1, est composé des éléments de contrôle du réseau satellite et des éléments de données. Le projet VITAL propose une approche de type Cloud-RAN [74] dans laquelle une division fonctionnelle précise d'une passerelle DVB-S2/RCS2 est proposée et effectuée. Au cours de ce processus, trois types de ressources peuvent être identifiés : les ressources NFVI utilisées par les NFs, les ressources radio sur le forward et return links allouées aux ST et les ressources physiques allouées. En fonction de ses capacités et de ses caractéristiques, le segment spatial ne fait que répéter le signal radio de la gateway satellite vers les spot-beams, ce qui implique des ressources radio. Dans le cas des méga-constellations ou de satellites GEO, le signal peut même être démodulé et traité à bord ce qui implique du matériel spécifique et COTS.

Les ressources allouées au network slice satellite peuvent également être augmentées ou diminuées [enjeu ii] en fonction des besoins des utilisateurs du slice. Les ressources allouées à un network slice peuvent être verrouillées exclusivement pour ce slice et ne pas être partagées avec d'autres slices. Les trois types d'allocation de ressources à un network slice sont les suivants :

- Ressources dédiées : Les ressources dédiées peuvent être considérées comme un type d'allocation où un sous-ensemble de ressources matérielles de la couche infrastructure est uniquement dédié à un seul network slice, sans support de partage. La notion de passage à l'échelle du network slice signifie dans ce cas ajouter des ressources physiques directes au slice et ces ressources resteront verrouillées jusqu'à ce que le network slice les libère ou soit terminé. Par exemple, les ressources peuvent être des serveurs physiques, des fonctions réseaux physiques, des liens physiques tels que la fibre noire, des GW satellites ou des ST. Ce type d'allocation de ressources n'exploite pas pleinement l'infrastructure et peut monopoliser des ressources inutilisées, d'où le coût plus élevé du network slice et son utilisation principalement pour les services de communication critiques qui nécessitent un haut niveau d'isolation et des performances accrues.

Nous définissons les ressources comme dédiées lorsque les ressources allouées à un network slice sont garanties pendant tout le cycle de vie du network slice et ne sont pas altérées en fonction des besoins des autres network slices.

- Ressources partagées : lorsqu'il n'y a pas de ressources spécifiques au niveau de la couche d'infrastructure qui sont allouées uniquement au network slice. Les ressources allouées au network slice peuvent varier en fonction du temps et des besoins du slice. La granularité des ressources allouées est plus fine et pourrait être, par exemple, le multiplexage de network slices sur une porteuse radio au lieu de dédier une porteuse complète pour chaque slice. Une telle approche permet un meilleur taux d'utilisation de l'infrastructure et pourrait donc réduire le coût associé à un network slice.
- Ressources hybrides : un network slice peut demander différents types d'allocation de ressources en fonction de ses besoins. La partie la plus coûteuse d'une communication par satellite est le taux d'occupation du spectre ra-

dielectrique. Le partage du spectre peut réduire considérablement le coût d'une communication par satellite pour le client d'un network slice. Ainsi, des allocations hybrides peuvent être envisagées avec des ressources dédiées sur l'infrastructure NFVI et des ressources partagées sur l'infrastructure radio.

### 3.3 Mise en commun des infrastructures satellites

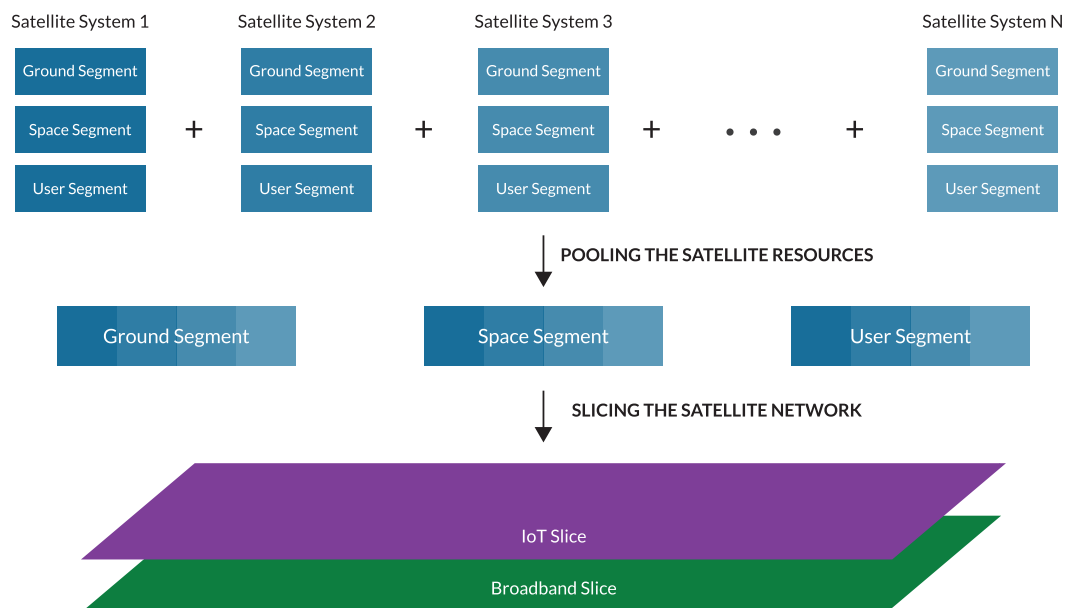


FIGURE 3.3 – Mutualisation des systèmes satellites pour découper le network slicing

Une autre différence majeure entre les réseaux 5G et les réseaux satellites, outre l'architecture, est le fait que chaque système satellite est totalement ou partiellement isolé des autres. Pour chaque nouveau système satellite, les fabricants réutilisent certaines technologies clés et les adaptent, comme les passerelles ou les charges utiles, mais chaque nouveau système satellite se comporte comme un réseau satellite autonome. Chaque segment composant l'architecture satellite implique des acteurs et des technologies différents et évolue à des vitesses différentes, ce qui rend également difficile l'orchestration globale du réseau. Les réseaux satellites ont été principalement conçus pour être efficaces, leur architecture et leurs protocoles sont donc optimisés et sont très spécifiques à chaque système. Comme chaque système satellite est isolé et ne peut offrir que la capacité dont il dispose, ces réseaux ne tirent actuellement pas pleinement parti de leurs capacités agrégées et de leurs zones de couverture. La mise en commun de tous les systèmes satellites (figure 3.3) profiterait grandement au réseau satellite et, ce faisant, dans ce nouveau contexte, le paradigme de network slicing prendrait tout son sens. Ce paradigme et la mutualisation des infrastructures satellites est d'autant plus

intéressant pour les opérateurs disposant de systèmes fournissant des services hétérogènes.

Compte tenu de l'architecture actuelle des réseaux satellites, il est très difficile de regrouper toutes les ressources et de tirer pleinement parti du paradigme du découpage du réseau et des technologies SDN/NFV. L'architecture n'est pas assez flexible et ne tire pas pleinement partie de toute l'infrastructure satellite pour partitionner le réseau satellite en network slices optimisées. Mettre en commun les infrastructures nécessite également des efforts considérables pour accroître l'interopérabilité entre les différents composants utilisés au sein des réseaux satellites. Les efforts de standardisation actuellement déployés au 3GPP pourrait permettre d'atteindre cet objectif qui profiterait aux réseaux satellites et aux réseaux terrestres. Nous proposons une évolution de l'architecture des réseaux satellites permettant de mettre en commun les ressources des satellites et exploiter pleinement le paradigme de network slicing.

### 3.4 Une architecture satellite plus flexible

Les réseaux 5G sont en passe d'implanter le network slicing grâce à leur architecture basée sur les services et à la séparation claire entre les fonctionnalités RAN et CN. Cette architecture s'appuie également sur des outils clés tels que les paradigmes SDN et NFV qui améliorent considérablement la flexibilité de la gestion du réseau et de l'allocation des ressources. Cette séparation claire entre le CN et le RAN n'existe pas dans les réseaux de satellites, ce qui accroît la difficulté d'une allocation dynamique et efficace des ressources. Par conséquent, nous proposons une nouvelle architecture capable de permettre le "slicing" du réseau satellite. Cette architecture tire pleinement parti du paradigme NFV pour allouer efficacement les ressources et du paradigme SDN pour orchestrer les flux au sein du réseau. Nous décrivons une architecture fonctionnelle composée d'une gateway satellite désagrégée (section 3.4.1) et d'un coeur de réseau satellite (section 3.4.2).

#### 3.4.1 Gateway Satellite

Nous commençons par redéfinir le rôle de la gateway satellite qui exécute diverses fonctions. Dans l'architecture satellite traditionnelle, cette gateway est généralement un équipement monolithique et est physiquement située dans un téléport satellite. Elle gère les fonctionnalités liées à trois couches réseau : la couche réseau de niveau 3 qui gère les fonctions supérieures de la gateway telles que le VPN, le PEP ou l'équilibrage de charge ; la couche de niveau 2 MAC et de niveau 1 physique qui gèrent les fonctions radio telles que l'allocation de ressources aux terminaux satellites sur les forward et return links. Dans notre approche, la gateway satellite n'est plus un composant unique mais un ensemble de sous-composants disjoints. Avec l'amélioration des capacités de traitement, de nombreux composants peuvent maintenant être exécutés sur du matériel COTS. Dans notre approche, nous considérons que nous sommes dans un environnement totalement

virtualisé et désagrègeons la gateway satellite en éléments indépendants comme le montre la figure 3.4.

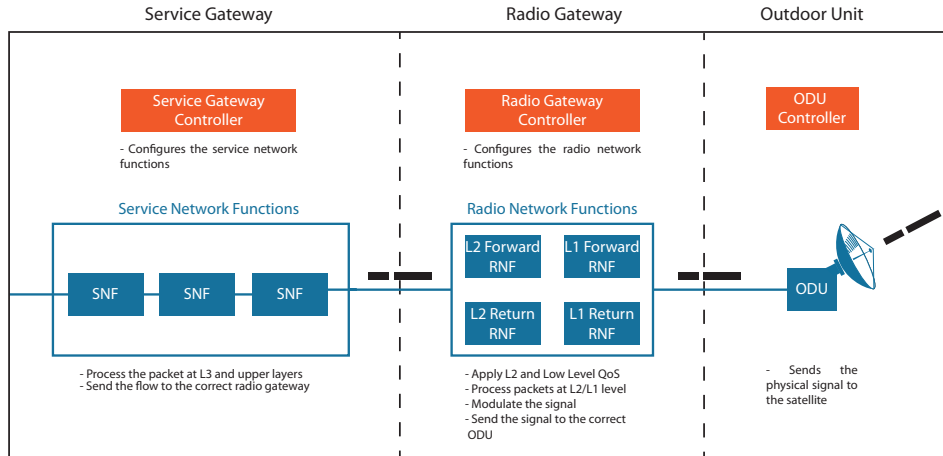


FIGURE 3.4 – Service Based Architecture-Gateway

L'objectif de notre architecture de passerelle satellite est d'allouer la bonne ressource et de supprimer la redondance dans les réseaux de satellites :

- **Service gateway :** cette partie de la gateway gère toutes les fonctionnalités de la couche de niveau 3 et des couches supérieures. La service gateway est composée d'une ou plusieurs Service Network Functions (SNFs) (e.g. Virtual Private Network (VPN), pare-feu, PEP) et d'un contrôleur de services. Le contrôleur expose aussi des API pour permettre la configuration de la service gateway par un tiers. Les fonctions de réseau de services sont chaînées et traitent le flux de données du réseau satellite avant de le transmettre à la radio gateway appropriée. Le contrôleur de services configure toutes les SNFs et les chaîne pour former un ensemble services cohérents. La service gateway est totalement indépendante des autres parties de la gateway et peut servir une ou plusieurs radio gateway. La service gateway doit être instanciée au sein d'une infrastructure NFVI qui peut se trouver dans un cloud public ou privé. Certaines fonctions, telles que le PEP, doivent être aussi proches que possible de la radio gateway. Les SNFs de la service gateway peuvent donc être distribuées sur plusieurs NFVI, l'un qui se situerait dans un domaine cloud et un autre qui serait co-localisé au sein du téléport proche de la radio gateway.
- **Radio gateway :** il s'agit de la partie radio de la gateway qui pourrait être spécifique pour chaque système satellite. Cette partie de la gateway gère uniquement les niveaux 2 et 1 qui nécessitent des fonctions spécifiques et parfois propriétaires. Elle est en charge de la gestion des ressources radio sur les forward et return links et s'occupe aussi du traitement de la QoS pour ces niveaux. La radio gateway est composée d'un contrôleur radio et de Radio Network Functions (RNFs). Les RNF traitent les flux aux niveaux

2 et 1 sur les forward et return links. Elles sont programmées à l'aide de l'interface sud du contrôleur radio (par exemple, configuration du plan de fréquences, configuration DAMA pour un système DVB-RCS2). La radio gateway est également totalement indépendante de la service gateway.

- Outdoor Unit (ODU) : envoie et reçoit le signal au segment spatial. Elle est reconfigurable et permet de communiquer avec un ensemble de systèmes satellite.

Notre approche est compatible avec des composants qui suivraient des normes ETSI ou des composants propriétaires. Les différentes NFs de la service gateway, la radio gateway peuvent être implantés comme des conteneurs ou des Virtual Machine (VM). Cela permet de bénéficier pleinement de l'état de l'art sur NFV et d'exploiter les travaux en cours sur la virtualisation dans les réseaux terrestres. En utilisant ce type de gateway désagrégée, il est possible d'instancier uniquement les ressources nécessaires et de commencer à relever le défi [enjeu ii] sur l'évolutivité du network slice. La modification des ressources du network slice n'affectera que la partie correspondante de la gateway (e.g. l'augmentation de la bande passante de la radio gateway n'augmentera pas nécessairement la puissance de traitement de la service gateway). Néanmoins, cette gateway ne fait que traiter le flux de données et ne traite aucun message de contrôle de politique ou de gestion de réseau. Ces fonctions sont traitées par le satellite core network.

### 3.4.2 Satellite Core Network

Un autre aspect fondamental que nous devons traiter dans le network slicing est la gestion de la QoS. La QoS doit être pris en charge au sein du network slice [enjeu iii] avec une politique de QoS indépendante pour chaque network slice [enjeu iv]. Le satellite core network est similaire au CN 5G au niveau des fonctionnalités, il gère les fonctionnalités du CN telles que l'accounting, l'authentification et définit la politique de gestion du réseau. Il gère également les fonctions spécifiques aux satellites telles que la configuration du segment spatial et les handover et la mobilité pour les méga-constellations MEO/LEO. L'objectif de ce satellite core network est de pouvoir mutualiser les fonctionnalités principales et redondantes présentes dans divers systèmes de satellites indépendants, de les instancier une fois et de les mettre à la disposition d'un ou plusieurs network slice satellites. Le satellite core network est modulaire et, en fonction des besoins du network slice, les fonctions instanciées peuvent énormément varier. Par exemple, un Satellite Network Operator (SNO) pourrait instancier un network slice commun qui aurait pour objectif d'implanter et de centraliser les fonctions réseaux d'authentification. La figure 3.5 montre la similitude avec le SBA 5G. L'ensemble de la radio gateway et le segment spatial est l'équivalent des RAN d'un réseau mobile terrestre. Ils constituent l'interface radio utilisée par le ST pour se connecter au satellite core network. La service gateway est, dans cette représentation, l'équivalent de l'UPF de la 5G, à la différence qu'elle pourrait également être distribuée entre le cloud et le téléport.



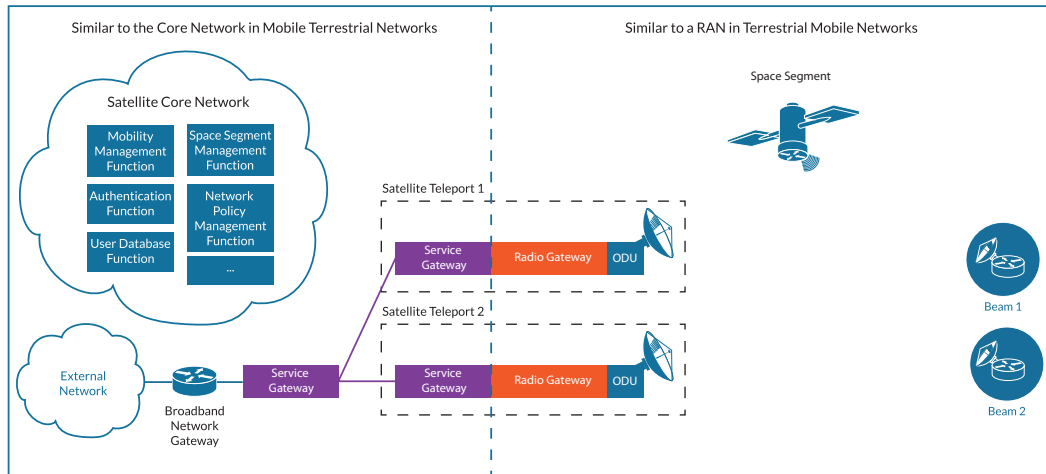


FIGURE 3.5 – Séparation du réseau satellite entre le satellite core and radio networks

Grâce à cette architecture, nous sommes en mesure de définir un network slice satellite comme étant l'ensemble d'un satellite core network, d'une service gateway et d'une ou plusieurs radio gateway connectées à des ODU. Nous pouvons alors relever le défi [enjeu iii] : chaque network slice tel que décrit ci-dessus supporte nativement la QoS puisque nous n'avons pas fondamentalement changé les composants du réseau satellite. Nous relevons également partiellement le défi [enjeu iv] mais la gestion et l'orchestration du network slice doivent être détaillées afin de relever pleinement ce défi. Une autre exigence clé pour le network slicing est le support des aspects de sécurité. Le network slice doit offrir différents niveaux de sécurité [enjeu x]. L'aspect sécurité n'est pas seulement lié à l'architecture mais aussi au type de ressources allouées network slice ainsi qu'à la technologie sous-jacente et aux fonctions réseau impliquées dans le déploiement des slices satellites.

### 3.5 Gestion et orchestration du network slice satellite

Dans la section précédente, nous avons défini un network slice satellite. Nous nous concentrons ici sur la gestion et l'orchestration d'un tel network slice. Un network slice satellite ou un groupe de network slices doit être géré par un tiers. Un tiers peut louer / posséder un ou plusieurs network slices et avoir pour chacun d'eux une politique de gestion spécifique ([enjeu v]). Une politique est un sous-ensemble de règles pour la gestion du cycle de vie du slice, basé sur des paramètres de qualité de service, des capacités de services, des considérations économiques et des ressources dédiées. Les opérations sur les network slices, à savoir la création, l'activation, la supervision, l'établissement de rapports, la modification, la désactivation et la résiliation, ne doivent pas nécessiter d'interventions manuelles et doivent être automatisées [enjeu vi]. Les opérateurs peuvent déclencher les opérations précédentes mais ne doivent pas exécuter manuellement les procédures liées

à chaque opération. L'automatisation dans les réseaux satellites est un véritable défi en raison des composants monolithiques actuels de l'architecture des réseaux satellites. Les technologies SDN et NFV combinées ensemble peuvent faciliter le processus d'automatisation et permettre un déploiement rapide des network slices [enjeu vii]. En outre, le système de gestion des network slices pourrait être suffisamment intelligent pour déclencher les opérations du cycle de vie de manière autonome [enjeu viii] (e.g. demander plus de ressources si nécessaire).

### 3.5.1 Modèle de rôle pour le network slicing satellite

Pour répondre à ces défis, nous définissons notre modèle de rôle (figure 3.6) en nous basant sur le modèle de rôle du 3GPP. Nous identifions les acteurs actuels du satellite qui sont impliqués dans le déploiement et la gestion de network slices :

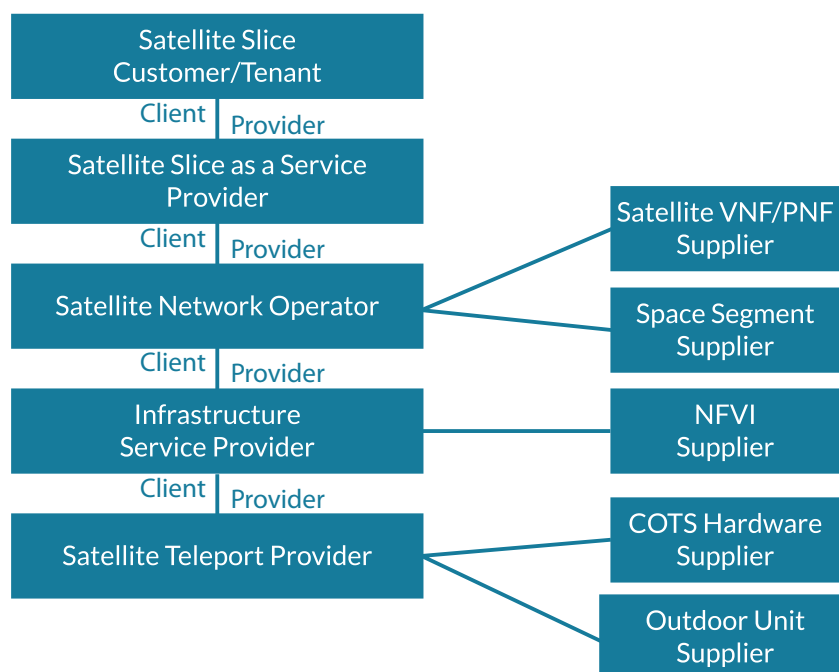


FIGURE 3.6 – Modèle de rôle du network slicing satellite

- Satellite Slice Customer est l'utilisateur ou le locataire du network slice satellite. Il peut s'agir d'un opérateur terrestre 5G qui souhaite acheminer des données 5G ou d'un courtier louant plusieurs network slices à plusieurs fournisseurs de network slices de satellite afin de fournir plusieurs types de connectivité à ses clients. Le client du network slice peut également être un fournisseur de services qui souhaite exécuter des services Over The Top (OTT).
- Satellite Slice as a Service Provider possède un système de gestion de network slice pour concevoir, exécuter et gérer des slices instanciés au sein de l'infrastructure satellite. Le slice provider a la possibilité de fournir des

slices avec différents niveaux de flexibilité à un slice customer. Le type de slices et de services qu’il peut offrir dépend aussi fortement des capacités de l’infrastructure. Le Satellite Network Operator, le Satellite Virtual Network Operator (SVNO) ou un opérateur de network slice peuvent agir en tant que slice provider.

- Satellite Network Operator possède le segment spatial du satellite et les VNFs/PNF mises à disposition par les fournisseurs (e.g. Eutelsat, SES, Via-Sat). Les VNFs/PNF peuvent être des composants monolithiques tels que des gateway satellites, les ST ou une gateway désagrégée comme décrit dans le §3.4.1.
- Infrastructure Service Provider met en place, gère et maintient une infrastructure virtualisée (e.g. OVH, Scaleway). Le NFVI Supplier fournit le système de gestion de l’infrastructure, il s’agit généralement d’une société spécialisée dans l’édition de systèmes de virtualisation (e.g. VMWare, RedHat).
- Satellite Teleport Provider est le propriétaire de l’infrastructure globale. Il construit le téléport satellite et l’équipe avec une multitude d’ODU en fonction de la demande de ses clients. Le téléport peut être utilisé par plusieurs SNO ou SVNO, il n’est pas dédié à un slice tenant spécifique. Le fournisseur de téléport construit et entretient également un backbone entre tous ses téléports et s’assure que chacun d’entre eux est constamment connecté aux réseaux de ses clients.

Ce modèle ne change pas fondamentalement le rôle actuel des acteurs du satellite. Il introduit de nouveaux acteurs liés aux technologies de virtualisation qui permettent de supporter le paradigme du network slicing. Chaque acteur peut bien sûr assumer un ou plusieurs rôles, ce qui dans dans la réalité serait très probablement le cas (e.g. le SNO et le slice provider étant le même acteur).

### 3.5.2 Satellite Management System

Nous définissons maintenant comment le network slice satellite serait instancié et orchestré au sein du réseau satellite.

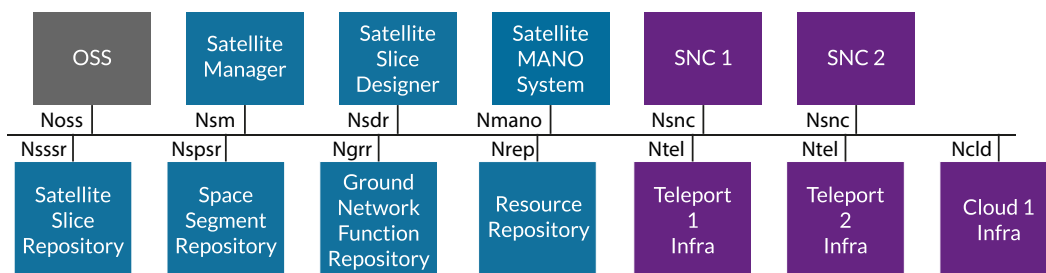


FIGURE 3.7 – Satellite Management System

La figure 3.7 montre le système de gestion que nous avons défini. Nous définissons des composants **de gestion**, **d’infrastructures** et des **composants externes**. Nous décrivons le processus d’instanciation de slice satellite sur la figure 3.8.

Le **satellite manager** expose les API du réseau satellites à l'OSS/BSS externe qui peut être considéré comme l'interface client. Ces API sont utilisées pour effectuer des requêtes de création ou de modification de network slices satellites (étape 1). Elles exposent soit des network slices prédéfinies présents dans le **satellite slice repository**, soit une liste de KPIs personnalisables tels que les débits montant/descendant, la latence ou les zones de couvertures. Le GSMA GST [104] peut être utilisé ici comme référence pour les KPIs car il est défini pour être générique et indépendant du réseau. Lorsqu'il se base sur des KPIs pour demander l'instanciation d'un network slice, le **satellite manager** agit comme un mandataire et transmet la liste des paramètres au **satellite slice designer** (étape 2). Le rôle principal du **satellite slice designer** est de concevoir le network slice satellite optimal de manière automatisée en fonction des paramètres reçus du **satellite manager** (étape 3). Le designer possède l'intelligence nécessaire pour traduire les exigences génériques en paramètres compréhensibles pour les réseaux satellites.

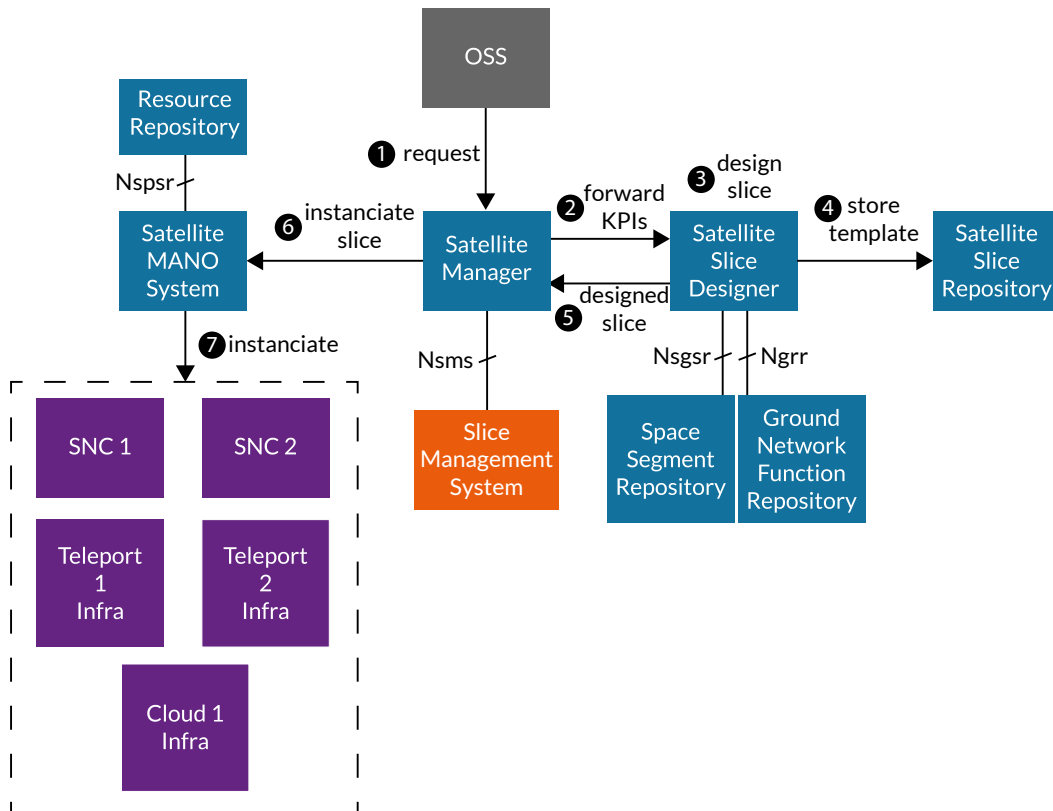


FIGURE 3.8 – Étapes d'instanciation du network slice satellite

Par exemple, les débits montant/descendant et les temps de latence peuvent être traduits par le choix d'un segment spatial spécifique associé à un ensemble de fonctions radio utilisées sur le forward link (e.g. DVB-S2 avec une bande passante dédiée) et sur le return link (e.g. SCPC). Pour déterminer le network slice satellite optimal, le designer s'appuie sur deux **repositories** : le **space segment repository**

et le **ground network functions repository**. Le space segment repository indexe tous les segments spatiaux disponibles des réseaux satellites et leurs Satellite Network Control (SNC) responsables de leur maintien en orbite. Ce repository résulte de la mise en commun des systèmes de satellites. Le ground network functions repository contient les fonctions du satellite core network, les fonctions désagrégées de la service based-architecture gateway et les fonctions ODU. Sur la base des exigences du slice et de ce dernier repository, le satellite slice designer compose le satellite core network, la service gateway, la radio gateway et le ODU adéquats.

Le slice designer produit une définition de network slice satellite et l'ajoute au satellite slice repository (étape 4). Il renvoie au satellite manager la définition du slice au satellite manager (étape 5) qui va demander au satellite MANO system de l'instancier (étape 6). Lorsqu'il instancie le network slice, il détermine le placement optimal des fonctions réseau dans l'infrastructure en utilisant le **ressource repository** (étape 7). Ce repository contient toutes les ressources d'infrastructure disponibles dans chaque cloud/téléport. Certains travaux ont déjà été proposés pour traiter le placement des VNF dans un network slice satellite [160]. En fonction de l'utilisation des ressources, le satellite manager sélectionne les clouds et les téléports appropriés. Il demande ensuite au **satellite MANO system** d'instancier toutes les fonctions réseaux correspondantes dans les infrastructures précédemment sélectionnées. Le processus d'instanciation implique la configuration du segment spatial, le satellite MANO system possède donc des interfaces pour pouvoir communiquer avec le SNC lors de l'instanciation des ressources à divers endroits.

### 3.5.3 Slice Management System

Pour chaque network slice, le satellite manager instancie également un unique **slice orchestrator**. Ceci assure l'indépendance entre les network slices au niveau orchestration et répond au défi [enjeu v] lié à la politique de gestion du slice. Le slice orchestrator est le composant responsable de la configuration initiale du slice, de sa modification, de sa supervision et de sa destruction. Dans le cas où les ressources radio du network slice sont partagées avec d'autres network slices, l'orchestrator se coordonne avec le Teleport NMS. Il doit également se coordonner avec le satellite manager afin de demander plus de ressources si nécessaire. De même, il interagit avec le Satellite Core Network qui est instancié et adapté pour chaque network slice.

Le slice orchestrator expose des API de gestion de slice au slice tenant. Grâce à ces interfaces, le tenant peut interagir avec le network slice et l'orchestrer selon ses besoins. L'interface de gestion exposée par l'orchestrator possède plusieurs niveaux de granularité : elle peut exposer un nombre de paramètres de gestion important pour permettre à un acteur du satellite ayant des connaissances suffisantes de régler avec précision les paramètres du slice (e.g. les paramètres de la radio gateway), ou exposer un nombre de paramètres limité pour permettre une gestion plus simple par un fournisseur de service par exemple.

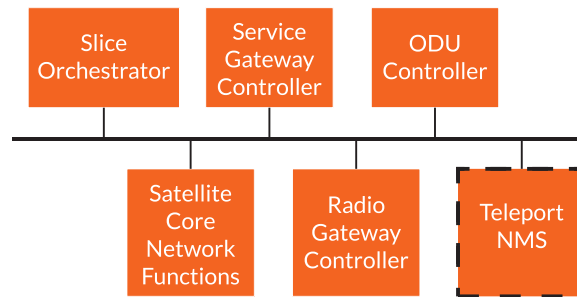


FIGURE 3.9 – Slice Management System

### 3.6 Conclusion

Dans ce chapitre, nous avons proposé une évolution de l'architecture satellite afin d'augmenter sa flexibilité et de pouvoir supporter le network slicing dans le but de l'intégrer au réseau 5G. Nous avons proposé de mettre en commun tous les systèmes satellites dont le résultat est une seule et même infrastructure. Nous ré-architecturons ensuite les composants satellites pour obtenir une distinction claire entre un coeur de réseau satellite et un réseau d'accès radio. Nous définissons un coeur de réseau satellite générique à qui l'on peut adjoindre plusieurs réseaux d'accès qui utiliseraient potentiellement des technologies radios différentes. De même, nous re-définissons le rôle de la gateway satellite en lui appliquant le concept d'architecture orientée service ainsi que le paradigme SDN pour aboutir à une ségrégation des ressources et de pouvoir en optimiser l'allocation. La résultante de ces propositions est la capacité à composer un coeur de réseau et un accès radio totalement spécifié pour le type de services que l'on veut déployer au sein de l'infrastructure mutualisée et virtualisée. Nous définissons par la suite des systèmes de gestion satellite et de slice permettant sur la base de KPIs d'instancier des network slices satellite et de les orchestrer.

Nous avons résumé dans le tableau 3.2 nos différents éléments de réponses aux enjeux que pose le network slicing dans les réseaux satellites en y associant les éléments techniques scientifiques qui permettent de répondre à ce paradigme dans les réseaux 5G comme décrit dans la section 2.2.2 du chapitre 1.

TABLE 3.2 – Défis du network slicing appliqués aux satellites et nos propositions associées

Défis	Références	Propositions
Isolation des ressources	i	Hyperviseur, Network Functions
Passage à l'échelle	ii	Infrastructure virtualisé, Hyperviseur

Supporter la QoS au sein du slice	iii	Utiliser les mécanismes de QoS présent au sein des réseaux satellites actuels
Politique de QoS indépendante entre chaque slice	iv	Orchestration indépendante, isolation des slices
Gestion indépendante pour chaque slice	vi	Orchestrateur unique par slice
Gestion autonome et automatisation du cycle de vie	vi	Infrastructure virtualisée, Management System, Utilisation d'Intelligence Artificielle (non détaillé ici)
Déploiement automatique du slice	x	Infrastructure virtualisée, Management System

---

Ces éléments de réponses nous permette de proposer un schéma d'intégration des réseaux satellites au sein des réseaux 5G en utilisant le concept de network slicing.

# Le network slicing pour le NTN dans la 5G

---

## Sommaire

---

4.1	Les grandes phases de l'intégration . . . . .	74
4.2	L'interface entre les systèmes de gestion . . . . .	74
4.3	Définition du niveau d'interconnexion entre les slices . . . . .	78
4.4	Continuité de slice et de QoS . . . . .	79
4.4.1	Modélisation de la continuité de slice et de QoS . . . . .	79
4.4.2	Le slice classifier dans le plan de donnée . . . . .	82
4.4.3	Échange dynamique de paramètres de QoS . . . . .	84
4.4.4	Architecture Fonctionnelle End-to-end . . . . .	85
4.5	Conclusion . . . . .	88

---

Dans ce chapitre, nous définissons notre modèle d'intégration du NTN dans la 5G, nous identifions les points d'interconnexions entre les systèmes de gestion 5G et NTN, nous spécifions les interfaces pour réaliser cette interconnexion au sein du plan de management, de contrôle et de données, nous introduisons notre composant d'interconnexion le "slice classifier" et définissons une architecture que nous évaluons à posteriori sur notre plateforme dédiée.

Nous itérons sur les propositions du chapitre précédent et raffinons les enjeux du network slicing dans notre contexte d'intégration.



## 4.1 Les grandes phases de l'intégration

L'interconnexion des systèmes de gestion satellite et 5G est la première étape pour l'échange de paramètres afin de négocier l'établissement du network slice end-to-end. Les réseaux satellite et 5G doivent rester indépendants l'un de l'autre en termes de gestion et d'infrastructure sous-jacentes [enjeu vii]. De plus, pour que l'intégration soit transparente, la complexité du réseau satellite doit être abstraite à l'opérateur du réseau 5G [enjeu viii].

Nous identifions trois grandes phases qui doivent être prises en compte lors de l'établissement du network slice end-to-end 5G, comme décrit dans la figure 4.1 : la négociation de haut niveau entre le 5G et satellite management system, l'instanciation de tous les éléments qui composent le network slice end-to-end et l'intégration de tous les éléments, appelé par la suite le "stitching".

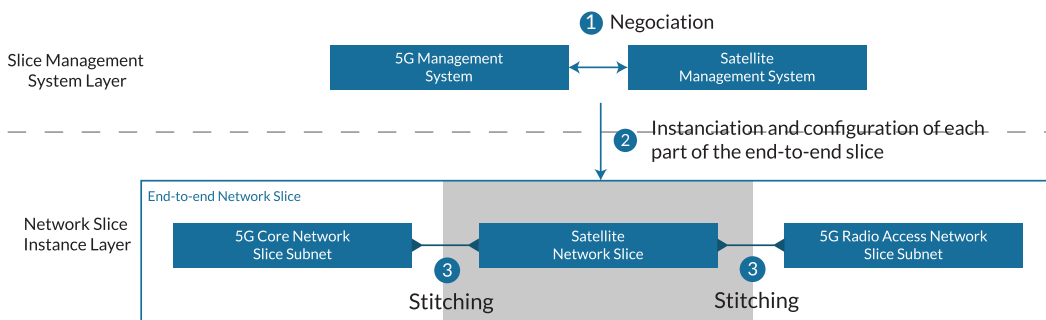


FIGURE 4.1 – Phases d'instanciation d'un network slice end-to-end

L'objectif de la première phase est de définir l'interface de communication entre le satellite et le réseau 5G. Grâce à cette interface, le 5G management system communique avec le satellite management system qui instancie un network slice satellite conforme aux exigences 5G. Ce network slice satellite est utilisé comme lien de backhaul de données entre le RAN et le CN 5G.

Dans la deuxième phase, le satellite management system dérive les exigences 5G en exigences compréhensibles par le réseau satellite, instancie les network slices et les configure. Le chapitre 3 décrit comment le network slice est modélisé dans l'architecture satellite sous-jacente et comment il peut être instancié. Le 5G management system instancie également les network slice subnets 5G dans cette phase, nous avons détaillé ces processus dans le chapitre 2.

La troisième et dernière phase est le "stitching" entre segment satellite, le network slice subnet CN 5G et le network slice subnet RAN 5G. Cette phase est décrite dans la section 4.3.

## 4.2 L'interface entre les systèmes de gestion

Nous considérons dans une première approche le network slice satellite comme un network slice subnet 3GPP pour spécifier l'interface entre les systèmes

de gestion satellite et 5G. Le réseau satellite hérite donc du management framework 5G et met en œuvre les procédures de gestion du cycle de vie de ce dernier [106] (e.g. vérification de la faisabilité, instanciation, modification, résiliation). Le satellite management system est identifié comme une Network Slice Subnet Management Function (NSSMF) vis-à-vis de la Network Slice Management Function (NSMF) 5G et met en œuvre les interfaces de gestion des network slice subnets (*allocateNssi* et *deallocateNssi*). Ces deux interfaces sont utilisées pour instancier et terminer un network slice subnet en envoyant une liste de KPIs (Slice Profile [103]). Néanmoins, les interfaces définies pour la gestion d'un network slice subnet ne sont pas suffisantes pour demander l'établissement d'une connectivité satellite car ils ne permettent pas de modéliser les besoins en terme de KPIs. L'utilisation du seul Slice Profile n'est donc adapté pour décrire le réseau satellite et être géré par le satellite management system, qui devrait mettre en œuvre d'autres mécanismes et interfaces pour compléter l'interface 3GPP (e.g. échange de slice/service type, échange dynamique de points d'interconnexion).

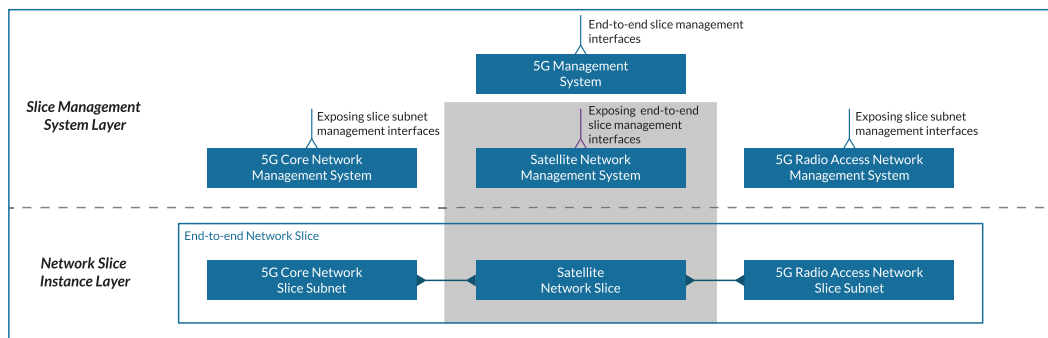


FIGURE 4.2 – Interfaces de gestion du slice satellite

C'est pourquoi, nous considérons le réseau satellite dans une seconde approche comme un network slice 3GPP qui va se comporter comme un Transport Network (TN). Le réseau satellite est dans cette approche un Transport Network (TN) évolué capable d'identifier les flux associés au slice : un "slice-aware non terrestrial network".

Le satellite management system peut alors exposer les interfaces de gestion du network slice end-to-end satellite au 5G management system, comme le montre la figure 4.2. Le 5G management system est donc en mesure d'interagir avec le satellite management system en utilisant les interfaces dédiées à la gestion de network slice end-to-end qu'il implante déjà. Ces interfaces sont complètes et permettent d'échanger des paramètres cruciaux (Service Profile [103]) pour le réseau satellite tels que les débits montants, descendants, la latence, le monitoring des KPIs, la zone de couverture, le niveau de partage des ressources et la gigue. Le satellite management system peut alors dériver et raffiner ces paramètres pour déterminer les équipements et les configurations optimales à déployer. En outre, la gestion du réseau satellite est compatible avec les spécifications 3GPP et permet une intégration transparente avec le réseau 5G puisque d'un point de vue opérateur 5G, le

réseau satellite possède le comportement d'un network slice 3GPP et donc peut en mesurer les KPIs. Cette proposition nous permet de répondre en partie à l'enjeu viii.

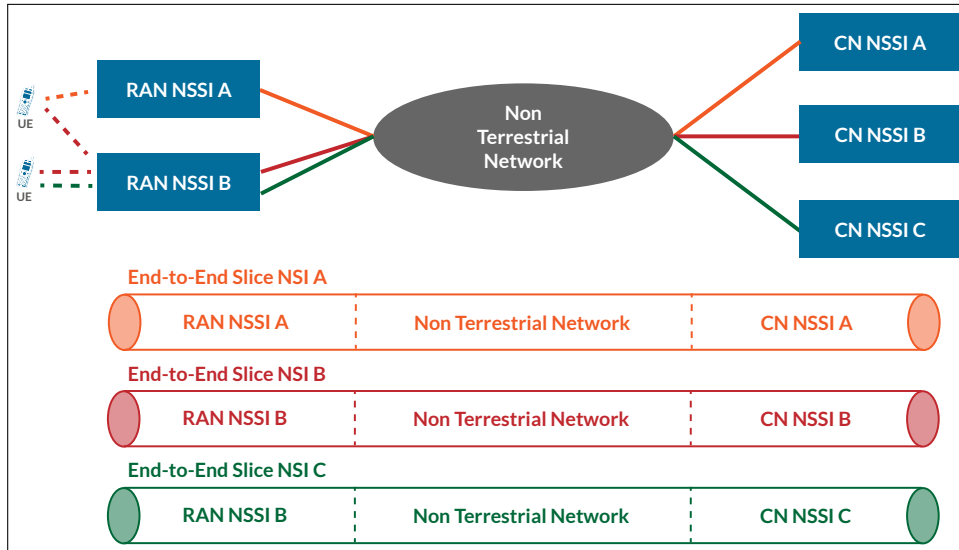


FIGURE 4.3 – 5G end-to-end slice avec un NTN slice aware en tant que TN

La figure 4.3 représente trois “network slice instances” composés de RAN network slice subnet instances et CN network slice subnet instances. Sur cette figure, le NTN fournit un ou plusieurs liens “slice-aware” entre chaque RAN network slice subnet instance et CN network slice subnet instance. L'assemblage entre les trois composants résulte en network slice instance end-to-end. Par exemple, la composition du RAN network slice subnet instance A, le lien fourni par le NTN et le CN network slice subnet instance A constitue le network slice instance A.

TABLE 4.1 – Modèle de Network Slice End-to-end

Symbole	Description
$S$	L'ensemble des network slice instances 5G
$S_i \in S, i \in \mathbb{N}$	Le $i^{\text{ème}}$ network slice instance
$S_{\text{RAN}}$	L'ensemble des RAN network slice subnet instances 5G
$Sr_j \in S_{\text{RAN}}, j \in \mathbb{N}$	Le $j^{\text{ème}}$ RAN network slice subnet instance 5G
$S_{\text{CN}}$	L'ensemble des CN network slice subnet instances 5G
$Sc_k \in S_{\text{CN}}, k \in \mathbb{N}$	Le $k^{\text{ème}}$ CN network slice subnet instance
$T$	L'ensemble des liens fournis par le NTN
$T_j^k \in T, (j, k) \in \mathbb{N}^2$	Le lien NTN entre $Sr_j$ et $Sc_k$

Nous généralisons cette approche et définissons dans le tableau 4.1 notre modèle de network slice. À partir des paramètres du Service Profile et des spécificités du satellite, nous avons extrait les principaux paramètres nécessaires à l'établissement et à la configuration d'une liaison NTN. Des paramètres supplémentaires (tels que ceux relatifs à la sécurité) seront ajoutés à l'avenir. Il appartient à l'opérateur 5G d'instancier ces paramètres en fonction de ses besoins end-to-end. Chaque network slice instance doit respecter ses KPIs associés. Ces paramètres de spécification sont dérivés dans un ensemble unique dédié à chaque domaine de réseau. L'opérateur satellite peut ensuite affiner et dériver ces paramètres en NFs satellites. Le 5-uplet  $\Theta(\lambda, \Delta, \mu, \beta, \sigma)$  qui identifie les paramètres de référence du lien NTN est défini comme suit :

- $\lambda$  : **latence maximale** en millisecondes. Ce paramètre détermine le segment spatial que l'opérateur du satellite doit utiliser pour assurer la liaison NTN. L'utilisation des GEO implique une complexité moindre que celle des LEO et des Middle Earth Orbit (MEO) car le ST n'aurait pas à suivre les satellites et à mettre en œuvre des méthodes de handover. En fonction des besoins de l'opérateur 5G en termes de QoS, ce paramètre influence également de manière drastique le choix de l'opérateur satellite en matière de composants terrestres et de technologies.
- $\Delta$  : **jitter** en milliseconde. En raison de leur architecture et de l'influence de la météo sur les communications par satellite, une gigue non négligeable peut être introduite. Les retards "non constants" pourraient poser des problèmes de performance importants dans les réseaux 5G, tels que des retards dans les communications vocales ou les messages plus critiques du plan de contrôle. Avec ce paramètre spécifié par l'opérateur 5G, les opérateurs de satellites peuvent déployer des NFs appropriés et les configurer afin de réduire et compenser ce retard.
- $\mu$  et  $\beta$  : **débit montants et descendants** en Mbit/s. Les technologies des forward et return links diffèrent et, en fonction du débit requis, les opérateurs de satellites utiliseront un certain ensemble de bandes de fréquences, de protocoles et d'équipements.
- $\sigma$  : **partage des ressources** (booléen) avec d'autres tranches. Ce paramètre influence le niveau d'isolement, donc le type de ressources radio (par exemple, Single Channel Per Carrier (SCPC), MF-TDMA) ainsi que les ressources de calcul.

Les travaux du 3GPP sur le network slicing sont en cours. Les spécifications ne fournissent pas de solution normative d'interconnecter les différents network slice subnets composant les network slice et de garantir un service end-to-end. Elles ne précisent pas les procédures et les méthodes permettant de configurer des points d'extrémités et d'interconnexion, ainsi que d'assurer le transit du trafic entre les différents network slice subnets. Par conséquent, en plus des interfaces déjà spécifiées dans la 5G, des procédures et des mécanismes doivent être définis afin de relier les différents network slice subnets et de garantir un service end-to-end.

### 4.3 Définition du niveau d'interconnexion entre les slices

Dans cette section, nous définissons le stitching des network slice subnet. L'opération de stitching est le processus d'assemblage des network slice subnets pour former le network slice end-to-end. L'objectif de cette interconnexion est de garantir les exigences de performance du network slice 5G. La QoS doit être prise en charge au sein des network slices, la continuité de la QoS à travers des domaines 5G et satellites est donc fondamentale. L'exigence end-to-end implique également la continuité de la QoS au sein de tous les network slice subnets [enjeu v]. Chaque network slice subnet a ses propres plans de gestion, de contrôle et de données, et nous devons donc déterminer à quel niveau et comment ils peuvent être interconnectés. L'interconnexion des network slice subnet est une tâche particulièrement complexe lorsque le network slice doit respecter des exigences de performances bout-en-bout.

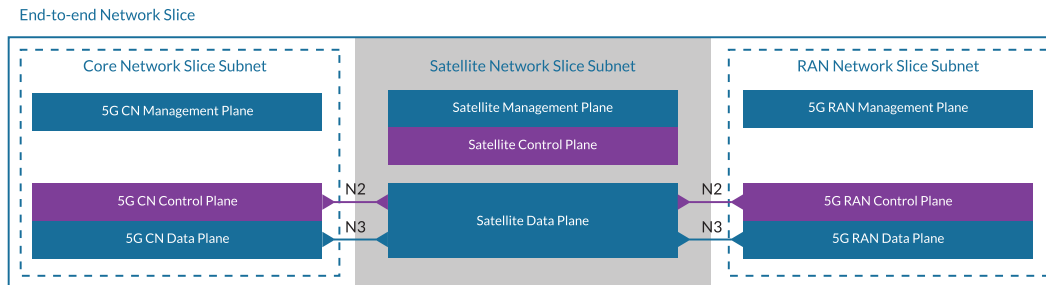


FIGURE 4.4 – Niveau d'interconnexion entre les réseaux satellite et 5G

Chaque network slice subnet est isolé des autres et traite les paquets en utilisant son propre système de QoS interne. Le réseau 5G utilise les classes 5QI qui peuvent être standard ou non standard. Il gère en interne la correspondance entre les classes 5QI, le système de gestion de QoS au niveau des ressources radios et le système de gestion de QoS au niveau CN. Le réseau satellite utilise un système de QoS spécifique personnalisé qui peut être appliqué aux niveaux réseaux 3 et 2. Un Class of Services (CoS) unifié pour la 5G et le réseau satellite est très probablement impossible car les deux réseaux sont trop hétérogènes. L'option immédiate et la plus courante consiste à effectuer une traduction de QoS entre les classes 5G et satellites. Les fonctions réseau du CN communiquent avec le RAN 5G en utilisant les points de référence N2 et N3. Le point de référence N2 est dédié à la communication entre l'AMF et le gNB (plan de contrôle via SCTP) et le point de référence N3 est entre l'UPF et le gNB (plan de données via GTP-Uv1). Les deux points de référence, dans le plan de données du satellite, sont traités sans distinction (figure 4.4). Par conséquent, le plan de contrôle 5G doit être pris en compte dans le flux de donnée qui circule au sein du réseau satellite et intégré dans ces classes de QoS.

Nous introduisons donc dans notre modèle la fonction de stitching associée, définie dans l'équation (1). Elle définit un network slice end-to-end comme décrit dans l'équation (2) qui est le résultat de la composition du RAN, du NTN et du

CN.

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

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

En reprenant la figure 4.3, le NTN fournit trois liens slice-aware au réseau 5G qui permettent de faire le lien entre le RAN et le CN. Ces trois liens peuvent être des liens logiques ou physiques. La composition résulte en network slices  $NSI_A = S_0 = \text{stitch}(Sr_0, Sc_0, T_0^0)$ ,  $NSI_B = S_1 = \text{stitch}(Sr_1, Sc_1, T_1^1)$  et  $NSI_C = S_2 = \text{stitch}(Sr_1, Sc_0, T_1^2)$ .

La flexibilité des réseaux 5G permet également de changer radicalement le déploiement des fonctions 5G (e.g. fonctions de réseau du CN déportée au côté du RAN pour mettre en place du MEC). En effet, l'ajout des AF dans le CN 5G et la possibilité d'intégrer une multitude d'applications hétérogènes amplifie le phénomène de dynamique des instances de réseau déployées. Cela peut introduire par exemple de nouveaux protocoles, donc de nouvelles règles QoS à prendre en compte même après le déploiement du network slice et donc le réseau satellite doit s'adapter et se reconfigurer rapidement pour prendre en compte de nouveaux trafics.

De plus, l'opérateur du réseau 5G peut considérer que le réseau satellite n'est pas un support de transport sécurisé et utiliser des protocoles de chiffrement tels que IPsec pour chiffrer le flux entre le RAN et le CN 5G. Ce flux chiffré obscurcit les interfaces N2 et N3 du réseau satellite qui ne serait plus en mesure de traiter correctement les paquets. Une correspondance statique des flux n'est donc pas une solution efficace et des méthodes qui permettent une dynamique plus importante doivent être mise en oeuvre.

Dans ce contexte, nous proposons donc une solution plus générique supportant le processus d'automatisation et la forte dynamique des réseaux 5G.

## 4.4 Continuité de slice et de QoS

### 4.4.1 Modélisation de la continuité de slice et de QoS

En utilisant  $\Theta$  (défini dans la section 4.2), nous sommes capables de fournir un lien  $T_j^k$  NTN, de le personnaliser en fonction des besoins des opérateurs 5G et de réserver les ressources correspondantes. Cependant, cela n'est pas suffisant pour considérer le lien NTN comme "slice-aware" puisque la continuité de network slice et QoS doit coexister dans le plan de données. Pour garantir un network slice end-to-end à travers tous les domaines réseaux et la continuité de QoS, les domaines

satellite et 5G doivent identifier, pour chaque paquet, d'abord le slice auquel il appartient, puis ses exigences en termes de QoS. Chaque UE établit de multiples sessions PDU et chacune d'entre elles appartient à un seul et unique network slice. Pour chaque session PDU, plusieurs flux avec des exigences de QoS différentes sont également établis. Le tableau 4.2 définit notre modèle pour les UE et les sessions PDU.

TABLE 4.2 – Modèles des UE et PDU

Symbole	Description
$U, \text{card}(U) = N_{\text{UE}}, N_{\text{UE}} \in \mathbb{N}$	L'ensemble des UE avec un total de $N_{\text{UE}}$ UE
$u_n \in U, n \in \llbracket 0, N_{\text{UE}} - 1 \rrbracket$	Le $n^{\text{me}}$ UE
$P^n, n \in \llbracket 0, N_{\text{UE}} - 1 \rrbracket$	L'ensemble des $u_n$ session PDU
$p_m(u_n, S_i), i \in \mathbb{N},$ $n \in \llbracket 0, N_{\text{UE}} - 1 \rrbracket, m \in \mathbb{N}$	La $m^{\text{me}}$ session PDU de $u_n$ établie sur le $i^{\text{eme}}$ slice

Pour assurer l'isolation des deux domaines réseau et le traitement correct du trafic, les propriétés slice et QoS des flux doivent être préservées d'un domaine à l'autre. Nous définissons le flux entrant comme le trafic entrant du réseau 5G vers le réseau satellite et le flux sortant comme le trafic sortant du réseau satellite vers le réseau 5G. Le network slice 5G et les informations de QoS du flux entrant doivent être traduites en trafic satellite, puis retransmises au réseau 5G. Le réseau 5G fournit un système de QoS end-to-end en utilisant le système de QoS flow. Le traitement de la QoS dans le NTN est différent du système de QoS 5G et les deux systèmes QoS sont isolés et totalement indépendants l'un de l'autre. De plus, la QoS des satellites diffère d'un système satellite à l'autre car ils ont chacun leurs propres spécificités. Cela pose un réel défi car les deux systèmes de QoS sont différents, nous devons nous assurer que les classes de QoS des deux systèmes sont compatibles. A ce jour, aucune correspondance n'a été proposée comme nous l'avons indiqué dans le chapitre 1. Peu de correspondances sont actuellement prises en charge car la plupart des satellites broadband sont des satellites GEO. Les méga-constellations LEO changent la donne et ouvrent la voie à davantage de prises en charge de 5QI au sein du réseau satellite. De plus, nous complétons cette correspondance en ajoutant notre méthode de négociation dynamique décrite dans la section 4.4.3.

Nous définissons notre modèle de flux dans le tableau 4.3. Chaque network slice est identifié dans le domaine 5G par l'ensemble d'informations  $g_i$ . Ces informations sont dérivées à partir du network slice  $S_i$  et des sessions PDU  $p_m(u_n, S_i)$  établies sur ce slice (e.g.  $g_i$  peut être un ensemble de champs de niveau 2 et 3). Au sein du network slice dans le domaine 5G,  $q_l$  identifie la QoS de chaque flux. L'ensemble d'informations  $t_i$  identifie le network slice dans le domaine satellite.

TABLE 4.3 – Modèle des flux

Symbole	Description
$Q_G$	L'ensemble des 5QI
$q_l \in Q_G, l \in \mathbb{N}$	Le QFI identifiant chaque 5QI
$Q_S$	L'ensemble des classes de QoS NTN
$c_p \in Q_S, p \in \mathbb{N}$	Chaque classe de QoS NTN
$I_G$	L'ensemble des informations identifiant les slices dans le domaine 5G
$g_i \in I_G, i \in \mathbb{N}$	L'information identifiant le $i^{\text{ème}}$ slice dans le domaine 5G
$I_S$	L'ensemble des informations identifiant les slices dans le domaine satellite
$t_i \in I_S, i \in \mathbb{N}$	L'ensemble des informations identifiant $i^{\text{ème}}$ slice dans le domaine satellite

La CoS du satellite est identifié par  $c_p$ . Les tuples  $(g_i, q_l)$  identifient complètement les flux et leur attribuent un network slice et des exigences de QoS dans le réseau 5G. Symétriquement, les tuples  $(t_i, c_p)$  identifient les flux et leur attribuent un network slice et des exigences de QoS au sein du réseau satellite. Nous définissons la fonction *pipe* et sa fonction inverse dans l'équation (3) qui remplit le rôle de traduction des flux, elle classe les network slice et la QoS pour chaque domaine réseau, de plus, elle traduit le tuple pour chaque domaine du réseau.

$$\begin{aligned} \text{pipe} : I_G \times Q_G &\rightarrow I_S \times Q_S \\ (g_i, q_l) &\mapsto (t_i, c_p) \end{aligned}$$

$$\begin{aligned} \text{pipe}^{-1} : I_S \times Q_S &\rightarrow I_G \times Q_G \\ (t_i, c_p) &\mapsto (g_i, q_l) \end{aligned} \quad (3)$$

Ceci complète notre modèle end-to-end illustré dans la figure 4.5.

Nous décrivons sur cette figure les différents UE  $U_n$  qui établissent des sessions PDUs  $p_m(u_n, S_i)$  au sein du réseau 5G. Au sein de chaque session PDU  $p_m$ , les UE  $U_n$  émettent un ensemble de flux 5G  $(g_i, q_l)$  qui traversent le network slice subnet RAN  $Sr_j$  avant d'être traduit par la fonction *pipe* en trafic  $(t_i, c_p)$  vers le lien NTN  $T_j^k$ . Ce trafic est ensuite transmis par le lien NTN vers la fonction *pipe*<sup>-1</sup> qui va re-traduire le trafic vers  $(g_i, q_l)$  avant d'être transmis au network slice subnet CN  $Sc_k$ .

Les fonctions *pipe* et *pipe*<sup>-1</sup> doivent être implantés dans le plan de donnée et



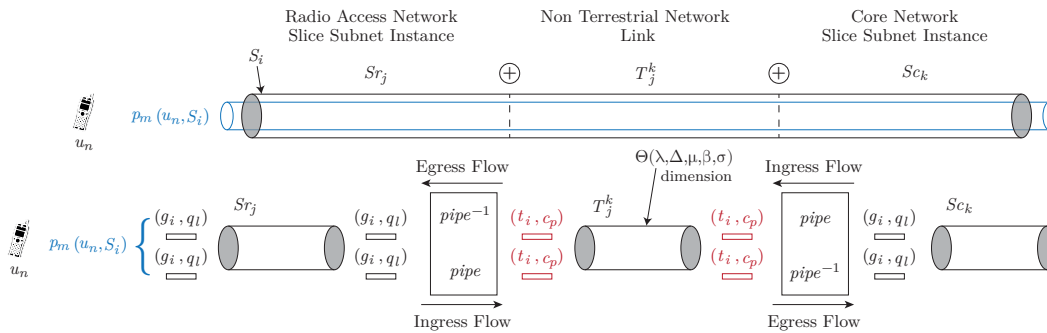


FIGURE 4.5 – Modèle Slice end-to-end

être en mesure de classifier, traduire, ainsi qu'appliquer des mécanismes permettant la continuité de slice de QoS. C'est pourquoi nous introduisons une fonction réseau dédiée à cette d'interconnexion : le slice classifier.

#### 4.4.2 Le slice classifier dans le plan de donnée

Les Slice Classifiers sont des fonctions placées à la frontière de chaque network slice subnet et sont les points d'interconnexion entre le slice satellite et les network slice subnets CN et RAN 5G qui peuvent être physiques ou virtuelles. Elles remplacent les Broadband Network Gateway (BNG) traditionnels et gèrent le trafic entre les network slice subnets interconnectés et assurent la continuité de QoS dans le plan de données au sein du network slice end-to-end. De même, elles sont aussi responsables de la traduction des flux entre chaque domaine réseau. Elles s'assurent que chaque paquet appartient au bon network slice, possède la classe de QoS adéquate et donc qu'il sera traité avec la bonne politique de gestion au sein de chaque segment réseau. Les network slice subnets sont interconnectés par les interfaces "east" (est) et "west" (ouest) des classifiants. Dans notre contexte, les slices classifiants traduisent les flux entre le CN 5G et le réseau satellite puis retraduisent les flux entre le réseau satellite et le RAN 5G. La figure 4.6 montre le processus interne du Slice Classifier, elle représente le classifiant qui interconnecte le CN 5G (west interface) avec le slice satellite (east interface).

Chaque interface a des flux entrants qui doivent être traduits vers des flux sortants corrects. Chaque flux entrant est marqué comme appartenant à un slice spécifique et à une classe de QoS. Le classifiant maintient constamment en mémoire une "Slice Table" et une "QoS Table" unique associées à chaque slice. Chaque flux est traité par le classifiant qui inspecte chaque paquet du flux et fait correspondre un ensemble de champs pour identifier son appartenance au slice et la QoS associée. Il traduit ensuite chaque paquet du flux en un paquet compréhensible par le segment réseau suivant. Par exemple, puisque le classifiant doit gérer les interfaces N2 et N3 du réseau 5G (comme précisé dans la section 4.3), les flux entrants sur la west interface pourraient être discriminés en se basant sur les ports SCTP ou les champs TEID GTP-U pour identifier le network slice et le champ DSCP IP pour

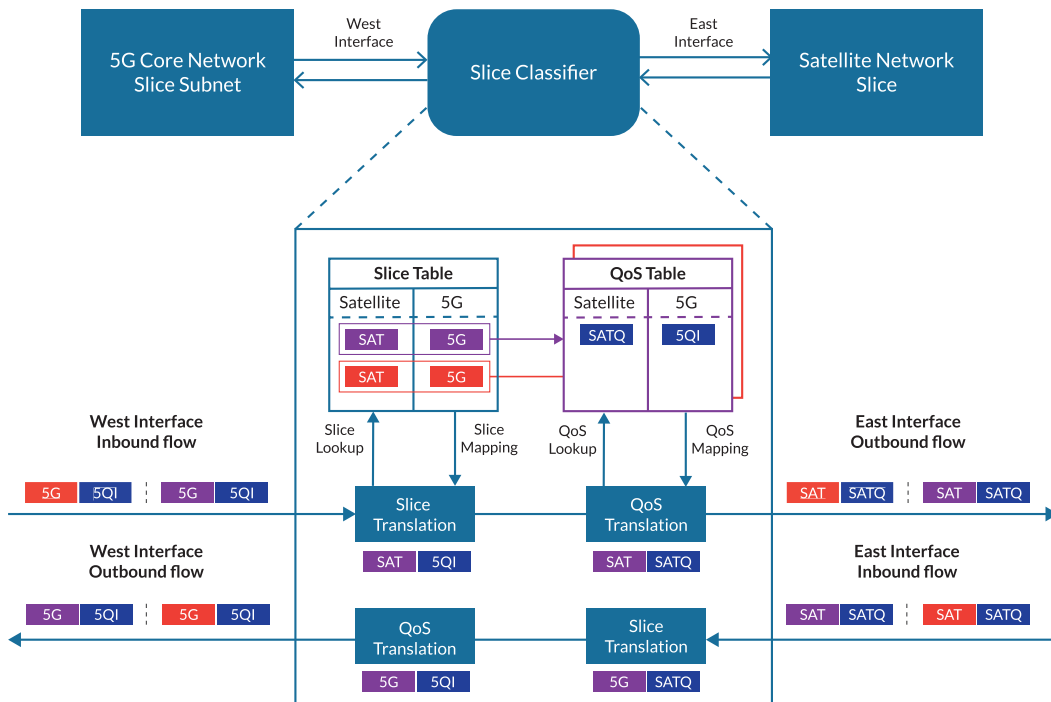


FIGURE 4.6 – Traitement des flux dans le Slice Classifier

identifier la classe de QoS. Le classifieur prend ensuite des mesures en fonction de la correspondance (e.g. insérer un label MPLS) pour traduire les flux qui seront traités par le réseau satellite. D'autres correspondances et actions sont possibles car le but de ce processus est de traduire les flux entrants en flux compréhensibles sur l'interface opposée pour le network slice subnet suivant dans le slice. Ce processus de traduction doit être rapide et efficace pour ne pas altérer les performances (e.g. ajouter de l'overhead dans les paquets qui minimise l'efficacité spectrale de la liaison radio dans le réseau satellite). Des correspondances simples telles que les tags VLAN ou les labels MPLS sont des options préférables pour identifier à la fois le network slice et la QoS.

Les tables du classifieur sont régulièrement mises à jour par le plan de contrôle qui est géré par le satellite management system. Le slice classifieur peut également jouer le rôle de dispositif de chiffrement pour le network slice satellite et implémenter des mécanismes de sécurité supplémentaires. Cela assure une couche de sécurité supplémentaire (répond au défi [enjeu xii]).

En définitive, les slice classifieurs permettent d'assurer dans le plan de donnée une continuité de slice et de QoS, ils permettent d'effectuer une sélection du lien NTN à utiliser pour garantir les exigences liées aux KPIs sur slice et la QoS ainsi qu'effectuer une traduction du domaine satellite vers le domaine 5G. Cependant, il est nécessaire de mettre en place le plan de contrôle associé pour répondre à la dynamique des réseaux 5G.

### 4.4.3 Échange dynamique de paramètres de QoS

Le réseau 5G doit se coordonner avec le réseau satellite pour établir le lien entre le CN et le RAN 5G. Nous proposons une solution qui supporte la dynamique des réseaux 5G et qui pourrait être étendue au-delà de l'intégration du satellite dans la 5G. La figure 4.7 montre le processus de négociation dynamique pour échanger la QoS entre les deux réseaux. Le satellite management system expose une interface dédiée à cette coordination sur laquelle les requêtes du réseau 5G sont effectuées une fois la demande d'établissement de slice effectuée. En utilisant cette interface le réseau 5G va envoyer les KPIs au réseau satellite, qui va les traduire, déterminer l'allocation optimale des ressources et instancier le network slice (étape 1 et 2).

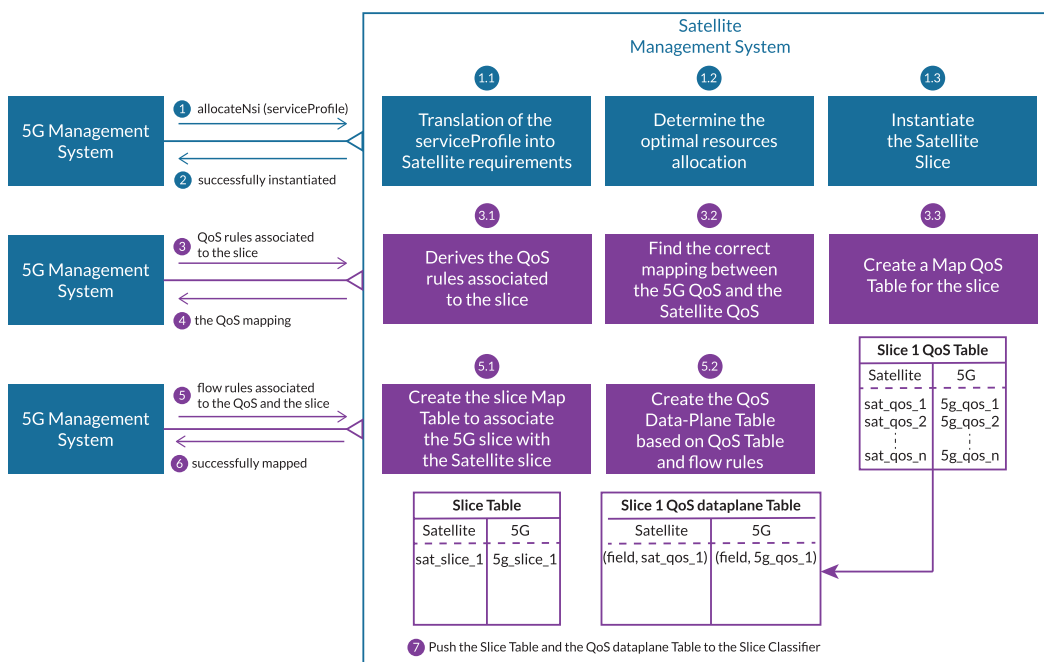


FIGURE 4.7 – Négociation dynamique des paramètres de QoS

Au travers de cette interface, le 5G management system envoie un ensemble de règles de QoS associées aux flux du network slice qu'il souhaite établir (étape 3). Les règles envoyées par le 5G management system doivent être compréhensibles par le satellite management system, l'interface exposée doit donc être construite conjointement avec les acteurs 5G. Ensuite, le satellite management system dérive les règles QoS associées au network slice : le satellite management system analyse l'ensemble des règles QoS envoyées par le réseau 5G (e.g. latence, priorité) et fait correspondre ces règles à son système de QoS interne. Le satellite management system crée une table qui associe règles QoS de la 5G au règles QoS du réseau satellite. Pour chaque network slice qui serait requêté par le réseau 5G, une table unique est créée et cette procédure est répétée.

L'unicité de la table QoS pour chaque network slice garantit l'indépendance des différentes politiques de QoS entre chaque network slice. Si le satellite mana-

gement system ne peut dériver et respecter une ou plusieurs règles QoS envoyées par le réseau 5G, il propose des alternatives ou rejette ces règles QoS. Ce processus permet de trouver une correspondance optimale entre les QoS de chaque network slice subnet et contribue à la continuité de QoS au sein du network slices end-to-end. Après avoir établi une correspondance optimale entre les différentes classes de QoS (étape 4), le réseau 5G envoie un autre lot de règles (étape 5) mais pour identifier les flux dans le plan de données associés au network slice et les règles de QoS envoyées auparavant. Les différentes règles permettent des correspondances sur les champs des paquets dans le plan de données (e.g. VLAN, IP, MPLS). Le satellite management system crée ensuite une table de plan de données QoS basée sur la table QoS et les règles de flux envoyées par le 5G management system. Si de nouveaux flux apparaissent dans le slice 5G (e.g. une nouvelle session PDU), le 5G management system enverra régulièrement de nouvelles règles au réseau satellite. Lors de changements structurels au sein du network slice les étapes 3 à 6 peuvent être répétées afin d'intégrer de nouvelles politiques de QoS et traiter les flux en fonction. Le plan de contrôle programme régulièrement les slice et QoS Table des slices classifiés (étape 7) afin qu'ils puissent traiter le trafic réseau. On adopte un paradigme SDN où le plan de contrôle va pouvoir programmer le plan de donnée de manière réactive et prédictive.

#### 4.4.4 Architecture Fonctionnelle End-to-end

Pour valider notre modèle, évaluer sa pertinence ainsi que nos propositions, nous avons défini une architecture fonctionnelle intégrant les réseaux satellite-5G et nous la mettons en œuvre sur notre banc d'essai basé sur des conteneurs dans le chapitre 5. Nous nous concentrons sur l'intégration des plans de contrôle et de données (étape 3 à 7 de la figure 4.7), nous ne développons donc pas ici le "plan d'orchestration" (e.g. la définition de fonctions d'orchestration utilisant le SBMA 5G ou les procédures d'orchestration). L'instanciation des liens NTN et leur configuration n'est pas abordée ici.

Comme indiqué dans la section 4.2, pour obtenir des network slice end-to-end, le NTN qui fournit un lien "slice-aware" entre le RAN et le CN doit être en mesure d'identifier les slices d'abord et ensuite, chaque flux individuel au sein du slice ainsi que sa politique de QoS. L'architecture 5G étant flexible afin de permettre une intégration dans les marchés verticaux et étendre l'utilisation des télécoms à une industrie plus variée, divers AF peuvent donc être instanciés au sein des network slice subnets. Cela va générer du nouveau trafic pendant le cycle de vie du slice; de même, les UE peuvent établir de nouvelles sessions PDU conduisant à de nouveaux flux dans le réseau. Par conséquent, l'état des flux doit être surveillé et maintenu en permanence. Nous avons donc étendu le fonctionnement traditionnel du réseau 5G, créé de nouvelles NFs, procédures et interfaces pour permettre un stitching automatisé et une surveillance de l'état dans les réseaux 5G et satellite. Notre architecture end-to-end ainsi que les interfaces nouvellement introduites sont présentées dans la figure 4.8.

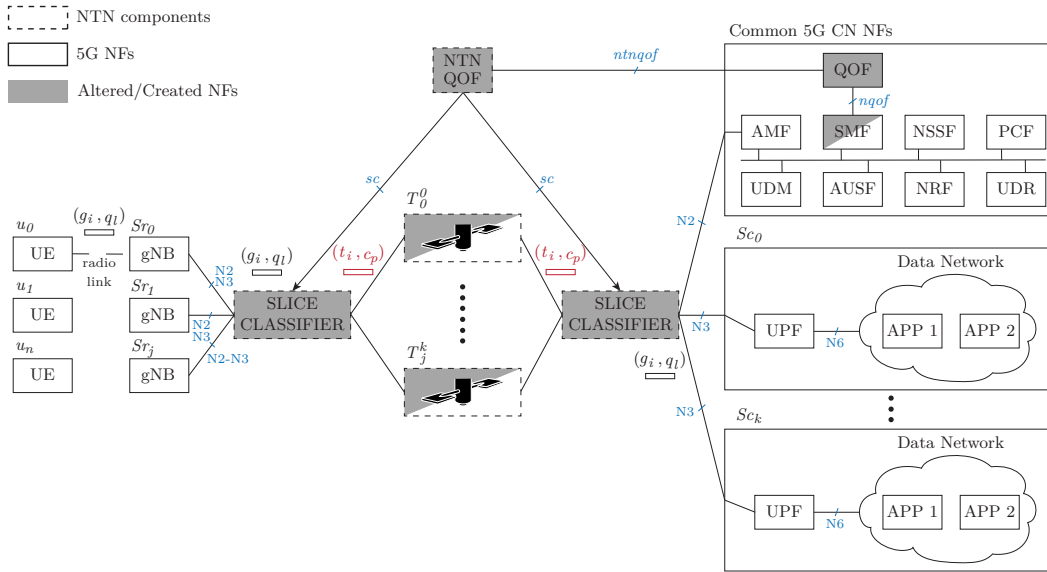


FIGURE 4.8 – Architecture Fonctionnelle End-to-end

Nous introduisons la **5G Quality of Service Function (QOF)** et la **NTN QOF** qui assurent la communication entre les plans de contrôle des réseaux 5G et satellite par le biais de l'interface *ntnqof*. Nous définissons une sous-routine dédiée au monitoring des flux dans les deux réseaux qui est appelée pour chaque création/modification/suppression de session PDU. La sous-routine est décrite dans la figure 4.9 avec l'exemple de la procédure d'établissement de session PDU. La procédure globale est divisée en plusieurs étapes : après un enregistrement réussi sur le réseau, un UE établit plusieurs sessions PDU avec le réseau 5G et déclenche la procédure d'établissement de session PDU 5G (étape 1) qui intègre un S-NSSAI. La SMF génère les informations nécessaires au traitement de la session PDU dans le plan de données 5G en fonction du S-NSSAI identifiant le network slice (étape 2). C'est ici que notre "hook" se déclenche, interrompt la procédure traditionnelle et exécute notre sous-routine. À partir des informations générées précédemment, la SMF extrait et envoie un sous-ensemble (supporté par le QFI, point de terminaison GTP) représentant la session PDU à la QOF 5G sur l'interface *nqof* (étape 3). La QOF 5G traduit ensuite ces informations en plusieurs tuples  $(g_i, q_i)$  identifiant chaque flux au sein de la session PDU (étape 4) et les envoie à la NTN QOF par le biais de l'interface *ntnqof* (étape 5). À son tour, la NTN QOF traduit chaque tuple  $(g_i, q_i)$  en tuple  $(t_i, c_p)$  correspondant qui identifie le flux dans le réseau satellite (étape 6).

Les slice et QoS tables doivent être constamment mise à jour en fonction de la configuration des réseaux 5G et satellites. Il s'agit de tables de correspondance entre les domaines satellite-5G et elles sont maintenues au sein des **Slice Classifiers**. En s'appuyant sur ces tables, les classifieurs sont capables de diriger le trafic vers les composants 5G et les liens NTN adéquats. Nous avons adopté le para-

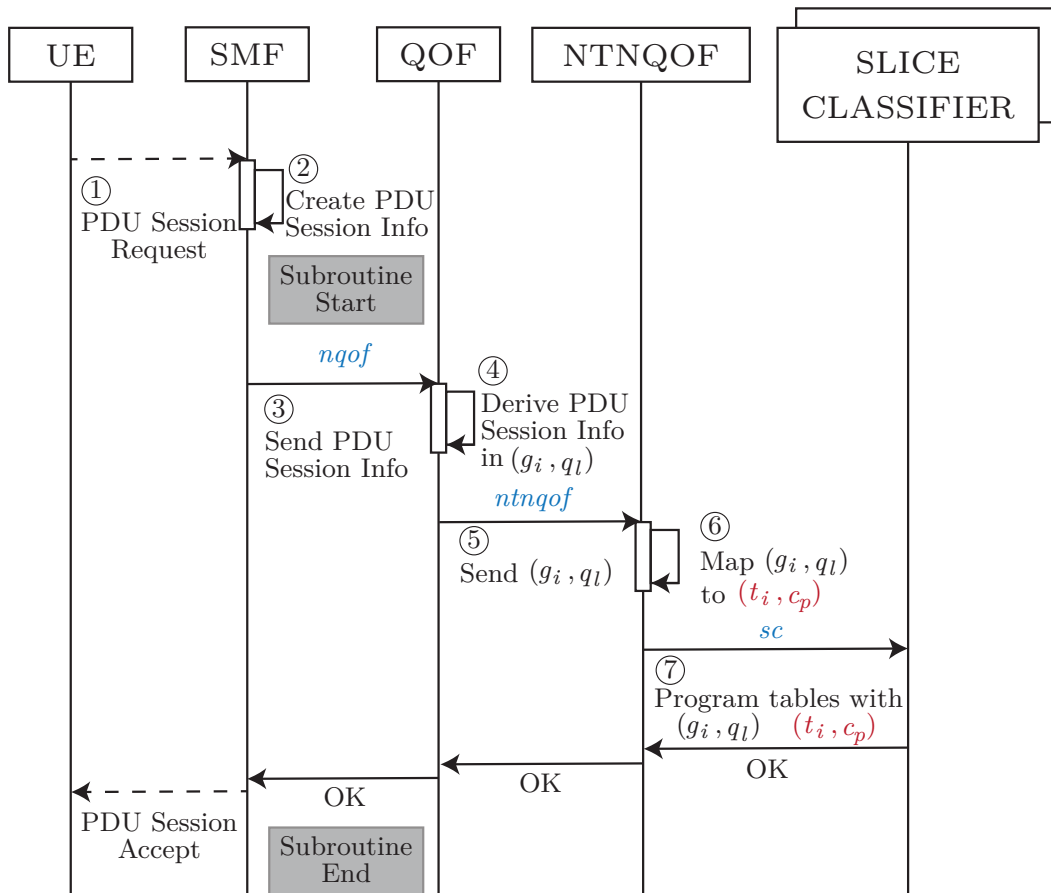


FIGURE 4.9 – PDU Session Establishment Procedure

digme SDN entre les classifieurs et les NTN QOF, qui en plus de son rôle de traduction des tuples, programme les classifieurs du network slice. Sur l'interface *sc*, des messages de contrôle sont envoyés pour peupler les tables de correspondance (étape 7). Les classifieurs surveillent également le trafic et mettent en œuvre des politiques liées au SLA du lien négocié ( $T_j^k$  est dimensionné en fonction de  $\Theta$  qui représente le SLA), assurant ainsi la continuité des network slices et de la QoS aux points d'interconnexion entre les réseaux satellite et 5G. Ils appliquent la politique des network slices et de QoS pour chaque flux et mettent en œuvre des fonctions telles que l'ordonnanceur basé sur la QoS et les politiques de traffic shaping pour correspondre au SLA de chaque network slice.

Pour valider nos propositions, nous avons développé un banc de test basé sur des conteneurs docker. Nous détaillons le développement de ce testbed dans le chapitre 5 ainsi que les résultats de nos propositions.

## 4.5 Conclusion

Dans ce chapitre, nous avons appliqué le concept de network slicing au réseau satellite afin d'améliorer la flexibilité de ce dernier, nécessaire pour une intégration dans les réseaux terrestres. Nous avons présenté notre modèle et notre architecture de network slicing pour intégrer le réseau satellite dans les réseaux 5G en tant que lien NTN "slice-aware". Notre approche permet une continuité de network slice end-to-end tout en assurant le respect des exigences de QoS au sein de chaque slice. Le tableau 4.4 synthétise nos propositions pour répondre aux enjeux liés au network slicing additionnés à ceux dédiés pour l'intégration du NTN dans la 5G.

TABLE 4.4 – Enjeux du network slicing appliqués aux satellites et propositions associées

Enjeux	Références	Propositions
Indépendance entre la 5G et le réseau satellite	vii	Modèle de rôle, API, Slice Classifiers
Intégration transparente du satellite dans la 5G	viii	Implanté les interfaces du 3GPP
Continuité de la QoS au sein du network slice	v	Slice Classifiers, QOF, NTN QOF, Hook dans la SMF
Assurer les performances du slice de bout-en-bout	xiii	API, Slice Classifiers, QOF, NTN QOF, Hook dans la SMF

De plus, l'isolation des ressources est préservée grâce aux correspondances de classes de QoS et aux politiques appliquées au sein des slice classifiers. L'architecture qui met en œuvre notre modèle est basée sur la SBA 3GPP et les nouveaux services que nous introduisons sont intégrés de manière transparente dans l'architecture globale, ce qui minimise l'effort d'adoption pour les acteurs du satellite et de la 5G. De plus, notre architecture maintient des domaines de gestion indépendants pour la 5G et le satellite, ce qui est une exigence primordiale pour les deux opérateurs. Les interactions entre ces opérateurs sont limitées à la gestion du cycle de vie des slices (e.g. le provisionnement d'un slice NTN spécifique) puisque l'interconnexion des plans de contrôle et de données est entièrement automatisée.

Nous avons résumé et explicité chaque étape nécessaire pour la mise en œuvre de notre intégration dans la figure 4.10. Les étapes 1, 2 et 3 ont été décrites dans le chapitre précédent et sont nécessaires pour obtenir une flexibilité suffisante au sein du réseau satellite afin de pouvoir réaliser les étapes 4 et 5 d'intégration décrites dans ce chapitre. Notre approche peut cependant être complétée pour permettre une modélisation uniforme de chaque network slice subnet ainsi que du network slice résultant et donc une orchestration générique. C'est précisément cette sixième étape que nous détaillons et étayons dans le chapitre suivant que nous consacrons à définir cette modélisation et à démontrer son efficacité en y appliquant un algorithme d'auto-scaling distribué.

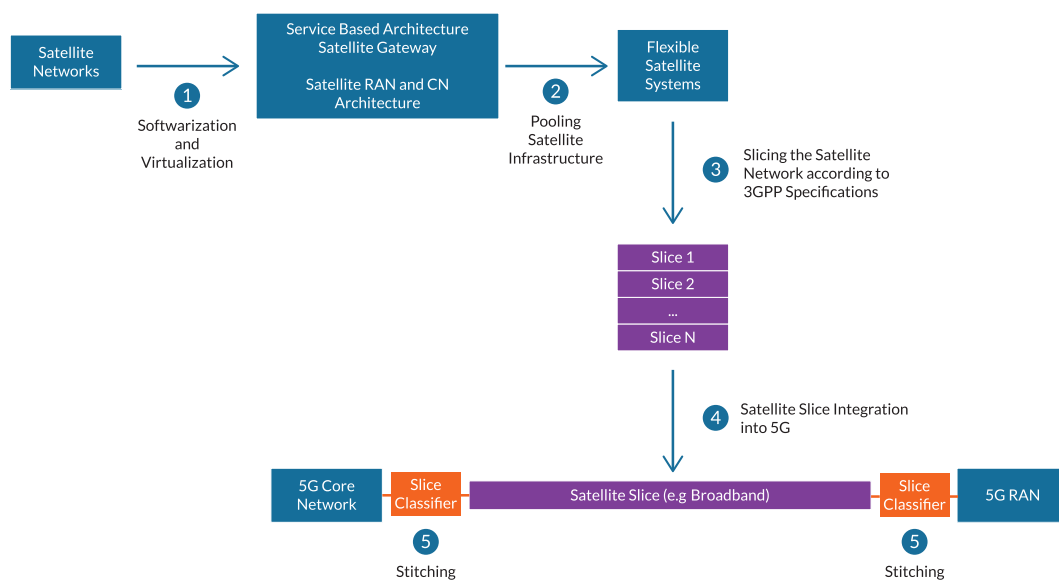


FIGURE 4.10 – Étapes pour le slicing du réseau satellite et l'intégration dans la 5G





# La continuité de Slice et de QoS

---

## Sommaire

---

<b>5.1</b>	<b>Réalisation d'une plateforme d'évaluation</b>	<b>92</b>
5.1.1	Nos besoins	92
5.1.2	Les plateformes existantes	92
5.1.3	Bilan et choix d'une plateforme	94
<b>5.2</b>	<b>Container Mesh for Network Slicing</b>	<b>94</b>
5.2.1	L'architecture de la plateforme	94
5.2.2	Les éléments fonctionnels	96
5.2.3	Une base pour nos expérimentations	98
<b>5.3</b>	<b>Évaluation de notre proposition "Slice Aware NTN"</b>	<b>100</b>
5.3.1	Scénarios	100
5.3.2	Validation Expérimentale	101
5.3.3	Évaluation des performances	103
5.3.4	Conclusion	108

---

Dans ce chapitre, nous présentons notre plateforme d'évaluation et l'utilisons pour évaluer la pertinence de nos précédentes contributions. Nous effectuons une revue des différentes plateformes d'expérimentations réseaux existantes ainsi que les apports de notre propre plateforme. Nous implantons ensuite nos propositions et les validons sur cette plateforme.

## 5.1 Réalisation d'une plateforme d'évaluation

### 5.1.1 Nos besoins

Afin de pouvoir tester et valider nos propositions, une plateforme d'expérimentation est nécessaire. Nos travaux impliquent l'utilisation de réseaux 5G RAN, 5G CN et NTN. Il est donc nécessaire de pouvoir intégrer ces différents réseaux au sein d'une même plateforme, tout en pouvant s'appuyer sur des composants réseaux existants dans le monde du logiciel libre. Par ailleurs, nous souhaitons pouvoir définir et rajouter nos propres composants, ainsi que générer différents scénarios de network slicing et leurs topologies réseaux associées et évaluer les performances de ces scénarios. La plateforme doit donc être flexible et évolutive.

Nous avons donc étudié un ensemble de plateformes d'expérimentations réseaux, orientées virtualisation pour la plupart, et disponibles à ce jour afin d'évaluer si elles répondaient à nos besoins.

### 5.1.2 Les plateformes existantes

#### 5.1.2.1 Network Simulator 3 (NS3)

L'outil Network Simulator 3 (NS3) [161] permet d'exécuter des simulations réseaux et de générer du trafic réseau terrestre. Des modèles mathématiques de trafic réseau sont définis et évalués sur des simulations. L'outil dispose également d'un mode émulation qui lui permet de se connecter à des réseaux externes et d'exécuter un environnement hybride émulation-simulation. Ce type de simulation est adapté aux phases préliminaires pour effectuer les premières preuves de concept, nous souhaitons dans notre campagne d'évaluation être au plus près de déploiements réels afin d'évaluer des performances plus réalistes et pertinentes. De plus, la nécessité de définir des modèles de trafic pour la totalité des composants 5G ainsi que certaines limitations de la plateforme [162] nous ont conduit à ne pas retenir cet outil.

#### 5.1.2.2 GNS-3

Tout comme NS3, le logiciel de prototypage réseau GNS-3 [163] permet de définir et d'évaluer des topologies réseaux. Il se base en revanche sur de la virtualisation en émulant des équipements réseaux sous forme de machines virtuelles. Néanmoins, les possibilités d'extension de la plateforme sont limitées et elle ne fournit pas un environnement suffisamment flexible pour implanter notre définition de network slice et en évaluer les performances.

#### 5.1.2.3 Mininet

Mininet [164] est un outil d'émulation extrêmement populaire qui permet de définir des topologies réseaux et de les évaluer dans un environnement virtualisé. C'est un outil flexible qui est utilisé pour la définition et l'expérimentation

des réseaux SDN. Néanmoins, Mininet ne permet une intégration que très limitée de composants externes, ce qui rend la plateforme difficilement extensible aux réseaux NTN et 5G.

#### 5.1.2.4 Open Source MANO (OSM)

OSM [165] est une solution d'orchestration qui permet de déployer et d'orchestrer des services réseaux définis et formalisés selon la norme ETSI NFV MANO. Il correspond au bloc MANO de l'architecture ETSI et s'interface avec des gestionnaires d'infrastructures virtuelles. OSM nécessite dans un premier temps de "packager" ses composants réseaux sous forme de VNF, puis de network services afin de pouvoir les instancier. Afin de pouvoir réaliser cette instanciation, il est aussi nécessaire de déployer une infrastructure virtualisée se basant sur des VM ou des conteneurs (e.g. kubernetes, VMWare, OpenStack). Nos expérimentations n'ont pas vocation à être déployées sur une plateforme ETSI NFV, donc cette surcouche d'orchestration ne nous apporte aucun avantage. Au contraire, cela ajoute une couche de complexité et de rigidité qui nous est inutile pour nos expérimentations.

#### 5.1.2.5 ONAP

ONAP [166] est une plateforme d'orchestration similaire à OSM avec une orientation opérateur de télécommunication. Elle est destinée à déployer et orchestrer des éléments réseaux au sein de réseaux opérateurs. De même qu'OSM, elle nécessite de déployer une infrastructure sous-jacente sur laquelle les composants réseaux vont s'exécuter. Cet outil ajoute une surcouche d'orchestration dédiée opérateur qui n'est pas assez flexible et donc non adaptée à notre contexte de recherche.

#### 5.1.2.6 Machines Virtuelles

L'utilisation directe de VM sans de surcouche d'orchestration est l'une des solutions les plus flexibles qui permet d'exécuter de nombreux scénarios dans notre cas. La virtualisation offerte par les hyperviseurs tels que ESXI [167], VirtualBox [168] ou Proxmox [169] permet la configuration et le lancement de VM sur lesquels un ou plusieurs composants réseau peuvent s'exécuter. Ces composants peuvent s'appuyer sur des logiciels existants mais aussi intégrer nos propositions. C'est donc la solution que nous avons retenue dans un premier temps et avec laquelle nous avons effectué nos premiers développements et premiers tests. Cependant la consommation élevée de ressources additionnée au manque d'automatisation et aux temps élevés de créations de VM augmentent de façon significative les temps d'exécutions des scénarios et donc les possibilités d'expérimentation. C'est pourquoi nous avons privilégié dans un second temps l'utilisation de virtualisation légère avec les conteneurs.

### 5.1.2.7 Conteneurs

Les conteneurs [170] utilisent des techniques de virtualisation légère, ce qui réduit grandement les temps de création et d'exécution des conteneurs. Ils utilisent la notion de namespace qui permet d'exécuter des environnements isolés au sein d'un même hôte qui partage son noyau à la différence des VM qui ont un noyau dédié. Les technologies Linux Containers (LXC) [171] et Docker [172] supportent l'exécution de conteneurs au sein de machines linux. Docker offre des services d'orchestration plus élaborés que LXC avec Docker Compose [173] et Docker Swarm [174] qui permettent de déployer des conteneurs de manière distribuée sur plusieurs machines physiques ou virtuelles ainsi que renforcer ces déploiements (e.g. autoscaling, keepalive). Similaire à Docker Swarm, Kubernetes [175] est l'orchestrateur de référence des conteneurs et est largement utilisé dans le monde cloud pour le déploiement de conteneurs en production. Swarm et Kubernetes ajoutent une surcouche d'orchestration complexe qui n'est pas nécessaire à nos travaux.

### 5.1.3 Bilan et choix d'une plateforme

Comme expliqué dans la section précédente, les différentes plateformes NS3, GNS3, Mininet, OSM et ONAP, ne permettent pas de répondre à nos besoins. Nous avons aussi envisagé l'utilisation d'OpenBACH [176] mais en est encore à des stades de développement très précoces et spécifique aux réseaux satellites ce qui ne permet pas d'expérimentation avec les réseaux 5G.

L'utilisation de VM et de conteneurs pour effectuer de premières implantations et valider des concepts est suffisant mais nécessite un travail d'intégration, de configuration et d'automatisation pour déployer et tester nos contributions. Nous nous sommes donc basés sur Docker et Docker Compose afin de développer notre propre plateforme d'expérimentation COntainer MEsh for nETwork Slicing (COMETS) en ajoutant une surcouche logicielle à ces outils. Cette surcouche implante notre définition du network slicing, définit nos scénarios, automatise le déploiement de ces scénarios et leur évaluation en s'appuyant sur des conteneurs et leurs capacité d'émulation.

## 5.2 Container Mesh for Network Slicing

### 5.2.1 L'architecture de la plateforme

Nous avons développé une plateforme automatisée capable de déployer des scénarios de test personnalisés intégrant les réseaux 5G et NTN. La figure 5.1 décrit les étapes mises en oeuvre par COMETS pour l'évaluation d'un scénario. Chaque type de scénario ainsi que son paramétrage est défini dans un fichier de configuration (voir annexe B.1) qui va être utilisé par le générateur de testbed. Pour nos besoins, nous avons défini deux types de scénario (voir annexe B.2) ainsi que des paramètres modifiables pour chaque scénario qui sont :

- Le type de scénario;

- la durée d'exécution du scénario ;
- le nombre de slices ;
- les applications déployées au sein de chaque slice ;
- la durée de vie de chaque slice ;
- le nombre de UE ;
- la configuration des liens NTN physiques : les débits maximum montant et descendant, la latence, la gigue et le packet error rate ;
- la répartition des slices sur les liens NTN.

Chaque scénario est associé à un ensemble de fonctions réseaux, de scripts et à une configuration globale spécifique qui constitue le testbed de ce scénario (voir annexe B.3). Les composants réseaux sont chacun définis comme un conteneur docker et l'ensemble du testbed est synthétisé dans un fichier **docker-compose.yaml**.

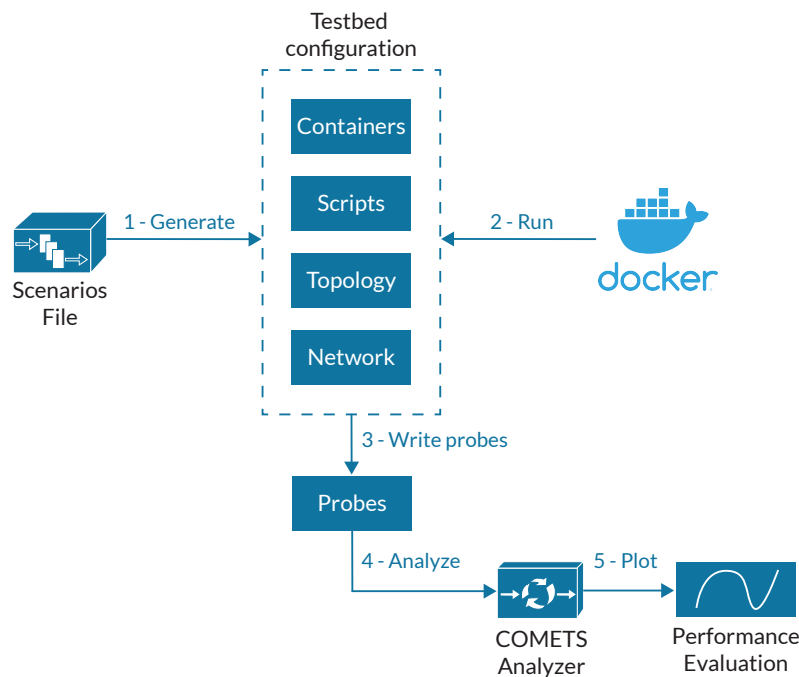


FIGURE 5.1 – Architecture de COMETS

Le runtime COMETS lance le testbed en utilisant le docker engine et génère des fichiers rapports qui sont ensuite utilisés par l'analyseur COMETS qui évalue les performances de chaque scénario et les compare. Nous avons défini cette plateforme afin qu'elle soit réutilisable à l'avenir, qu'elle puisse évoluer et que nous puissions constamment intégrer de nouveaux scénarios totalement hétérogènes, générer les architectures réseaux associées et évaluer leur pertinence et leurs performances. L'ensemble du système d'orchestration de la plateforme représente environ 26 000 lignes de code développé en Python et intègre une vingtaine de fonctions réseaux développées en Golang et C++.

## 5.2.2 Les éléments fonctionnels

Dans cette section, nous listons les différentes fonctions réseaux que nous avons intégrés dans notre plateforme. Chaque scénario COMETS dédié au network slicing génère une architecture réseau qui est décrit sur la figure 5.2. Le choix des éléments a été effectué avec la volonté d'être au plus proche d'un déploiement réel afin d'obtenir des résultats réalistes. C'est pourquoi nous favorisons les composants émulés et les outils réseaux réels.

### 5G Core Network

Nous utilisons free5GC [177] comme CN 5G qui est conforme à la release 15 du 3GPP. C'est le premier CN standalone open-source et est plus avancé dans son développement que les projets OpenAirInterface CN [178] et Open5GS [179]. Il est le cœur de référence de l'ONF qui l'intègre dans ses différents projets de recherche et dans des projets industriels. Il intègre toutes les fonctions du plan de contrôle du CN 5G ainsi que celles du plan de données. Un CN 5G est un ensemble de fonctions réseaux dont le fonctionnement est extrêmement complexe, c'est pourquoi l'équipe de développement de free5GC a conçu un framework logiciel commun à toutes les fonctions réseaux 5G en Golang [180] excepté pour l'UPF qui utilise des composants dédiés dans le noyau linux [181]. Ce framework peut être réutilisé afin de définir de nouvelles fonctions ou étendre les fonctions existantes.

En utilisant ce framework, nous avons modifié la SMF initiale [182] en ajoutant notre "hook" qui s'exécute à chaque création, modification ou suppression de session PDU. De même, nous avons implanté dans notre SMF [183] l'interface *nqof* en utilisant le protocole HTTP (la SMF est client HTTP). Nous avons implanté la QOF [184] et l'interface *nqof* associée qui expose un serveur HTTP. Nous avons également utilisé le protocole HTTP pour implanter l'interface *ntnqof* dans la QOF.

Pour optimiser la consommation des ressources, nous centralisons toutes les CN NFs du plan de contrôle qui sont communes à tous les network slices et pour chaque  $S_{c_k}$  nous n'instancions que les NFs nécessaires. Dans le plan de données, pour chaque slice, une seule UPF est dédiée ainsi que les serveurs applicatifs dans le DN.

### Serveurs applicatifs

Nous avons utilisé l'outil IPerf2 [185] pour générer du trafic de différent type. Chaque UE dans le réseau embarque des clients IPerf qui vont échanger des données avec des serveurs dans le DN. Pour palier certaines limites de l'application IPerf, nous avons dédié et instancié des serveurs pour chaque UE. Le trafic généré par les applications est identifié pour chaque type d'application par le Differentiated Services Code Point (DSCP) IP dans le plan de données (implantation en annexe C).

### 5G Radio Access Network

Pour le RAN 5G, nous avons utilisé le projet UERANSIM [186] qui implante une gNB logicielle conforme à la release 16 du 3GPP. Il fournit une implantation UE logicielle et simule également la liaison radio entre l'UE et la gNB. Cependant, des limitations existent avec cet outil qui ne supporte aucun mécanisme de QoS dans le plan de données. Lors de l'établissement du tunnel GTP, les paquets utilisateurs ayant une QoS bien spécifique ne sont pas marqués. Notre approche nécessite de pouvoir identifier la QoS associée à chaque flux. C'est pourquoi nous avons développé puis déployé sur la gNB un module basé sur l'extension u32 d'iptables [187] qui effectue du Deep Packet Inspection (DPI) sur les paquets utilisateurs et reporte le champ DSCP sur le paquet IP GTP.

### Lien NTN

Nous avons conçu et utilisé Trunks [188], un simulateur de lien satellite qui supporte les mécanismes QoS de niveau 3 et capable de simuler une dégradation de lien avec la mise en place de l'Adaptive Coding and Modulation (ACM). Il se base sur des outils linux natifs traffic control et iptables auxquels est adjoint une surcouche logicielle en Golang qui ajoute de la reconfiguration dynamique. Il est donc facilement exécutable sur une machine virtuelle ou un conteneur. Nous avons développé Trunks dans notre contexte pour remplacer l'émulateur OpenSAND [189] qui est dédié à l'émulation de systèmes satellites géo-stationnaires et difficilement conteneurisable. De même qu'OpenSAND, l'extension du simulateur NS3, Satellite NS3 (S-NS3) [190] ajoute des modules permettant de simuler des systèmes VHTS géostationnaires mais n'est pas adapté à notre solution de conteneurs.

Dans le plan de contrôle NTN, nous avons implanté la NTN QOF [191] en se basant sur le framework de free5GC. Cette fonction expose un serveur HTTP sur l'interface *ntnqof* sur laquelle la QOF va communiquer. Nous avons aussi implanté une API sud sur l'interface *sc* afin de programmer les slice classifiers.

### Slice Classifier

Les slice classifiers sont composés de deux modules : le premier module implante les slices et QoS tables en utilisant les outils du kernel linux netfilter et son interface iptables. Ces tables sont régulièrement mises à jour par la NTN QOF qui programme les classifiers. C'est pourquoi nous avons ajouté un second module développé en Golang qui expose une API nord et permet la communication sur l'interface *sc* avec la NTN QOF. Nous avons développé un protocole de contrôle personnalisé basé sur HTTP pour l'interface *sc* qui échange des règles de flux et qui utilise un classifier runtime afin de modifier le comportement de netfilter lorsqu'il y a une modification de sessions PDU. La gestion de la politique de network slice et de QoS au sein des classifiers est gérée grâce à une combinaison de files



d'attente Hierarchy Token Bucket (HTB) et Stochastic Fairness Queueing (SFQ). Tous les détails d'implantation du classifieur sont donnés en annexe D.

### 5.2.3 Une base pour nos expérimentations

En utilisant cette plateforme, nous allons déployer dans la section suivante et automatiser plusieurs scénarios de tests afin de valider nos propositions faites dans le chapitre précédent sur l'intégration du NTN dans la 5G. Nous avons publié la plateforme COMETS en open-source [192] pour permettre à la communauté scientifique d'effectuer ses propres expérimentations de manière fiable et rapide.

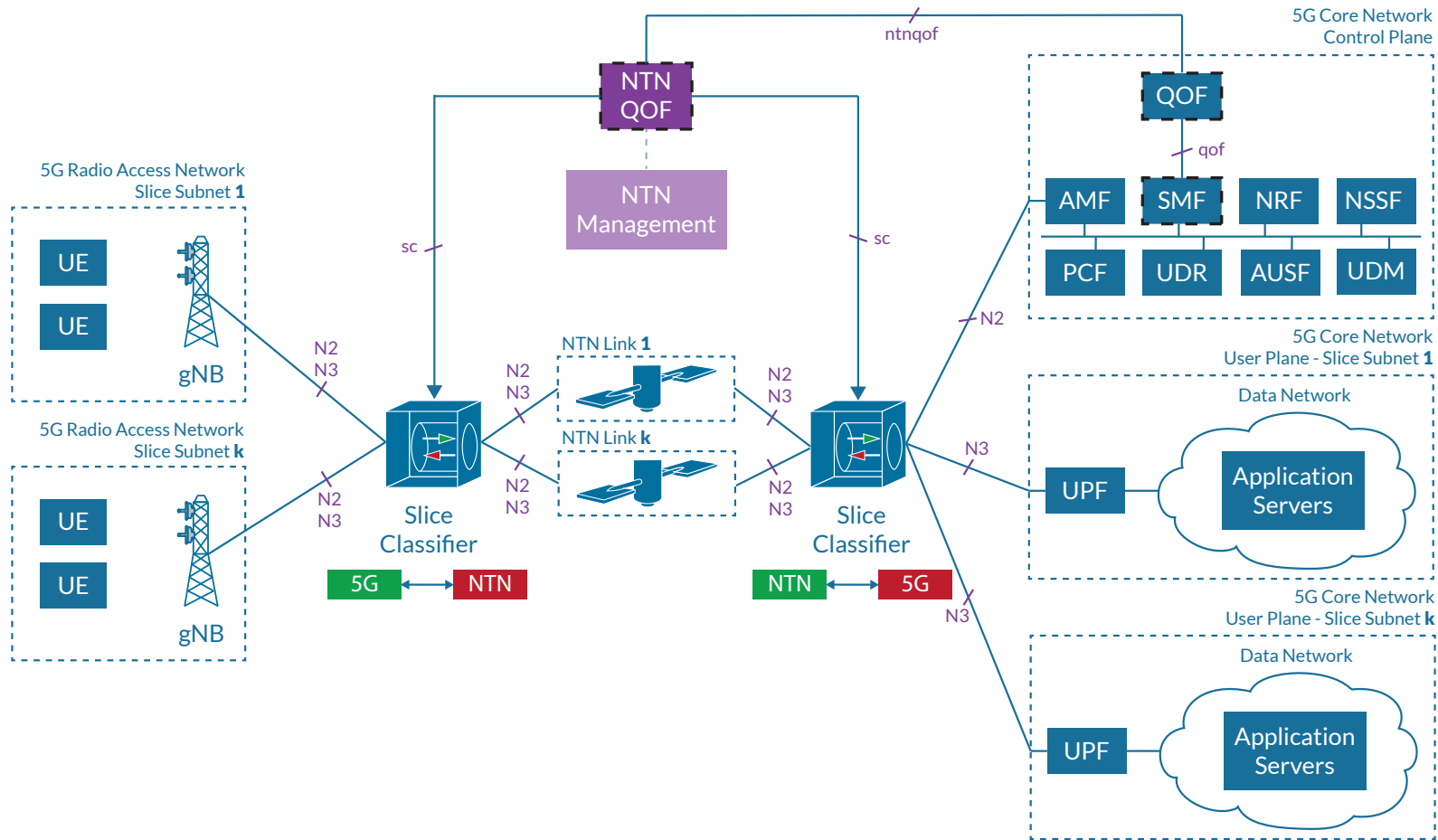


FIGURE 5.2 – Architecture fonctionnelle des scénarios

### 5.3 Évaluation de notre proposition “Slice Aware NTN”

#### 5.3.1 Scénarios

Nous définissons un scénario dans le tableau 5.1 dans lequel trois network slices end-to-end sont lancés simultanément, chaque slice proposant différentes applications avec des exigences de QoS spécifiques. Les fournisseurs de lien NTN mettent en place deux liaisons physiques. Une liaison LEO  $L$  avec une capacité de liaison aller de 100 Mbps, une capacité de liaison retour de 25 Mbps et une latence cible de 45 ms avec une gigue variable de 5 ms. Un lien GEO  $G$  avec une capacité de liaison aller de 100 Mbps, une capacité de liaison retour de 25 Mbps et une latence cible de 550 ms avec une gigue variable de 50 ms. Nous exécutons ce scénario deux fois pour évaluer les systèmes de liaison traditionnels par rapport à notre modèle.

TABLE 5.1 – Définition du scénario

Paramètre	Description
$D = 240s$	Durée totale
$S_i, i \in \llbracket 0, 2 \rrbracket$	Total de 3 network slice instances
$Sr_j, j = 1$	1 RAN network slice subnet instance
$Sc_k, k \in \llbracket 0, 2 \rrbracket$	3 CN network slice subnet instances
$N_{UE} = 9$	9 UE
$D_0 = [0, 240]$	Slice 0 “ON” de 0s à 240s
$\Theta_0 = (45, 5, 35, 10, true)$	Paramètres du lien NTN entre $Sr_0$ et $Sc_0$ en utilisant le système LEO
$A_0 = \{\text{Web, VoIP}\}$	Applications dans le slice $S_0$
$D_1 = [120, 240]$	Slice 1 “ON” de 120s à 240s
$\Theta_1 = (45, 5, 65, 15, true)$	Paramètres du lien NTN entre $Sr_0$ et $Sc_1$ en utilisant le système LEO
$A_1 = \{\text{Streaming}\}$	Application dans le slice $S_1$
$D_2 = [60, 180]$	Slice 2 “ON” de 60s à 180s
$\Theta_2 = (550, 50, 40, 15, false)$	Paramètres du lien NTN entre $Sr_0$ et $Sc_2$ en utilisant le système GEO
$A_2 = \{\text{Streaming}\}$	Application dans le slice $S_2$

Tout d’abord, dans un mode “slice-unaware” (SUAW) où  $S_0$  et  $S_1$  sont multiplexés sur  $L$  qui agit comme une liaison de retour traditionnelle (c’est-à-dire  $T_0^0 = T_0^1 = L$ ) et le trafic  $S_2$  est envoyé sur  $G$ . L’interconnexion des réseaux NTN et 5G est réalisée à l’aide d’une BNG classique. Ensuite, nous rejouons ce scénario dans un mode “slice-aware” (SAW) dans lequel nous déployons tous les mécanismes décrits dans le chapitre précédent qui implantent notre modèle et doivent

garantir une continuité de slice et de QoS. Nous ajoutons la QOF dans le CN 5G, la NTN QOF dans le réseau satellite et ajoutons les slice classifiers pour réaliser l’interconnexion du NTN avec le réseau 5G. Dans le mode SAW, le fournisseur NTN est capable d’instancier deux liens logiques  $T_0^0$  et  $T_0^1$  au dessus du lien physique  $L$ . Le slice  $S_0$  est dédié aux applications à faible latence  $A_0$ , la VoIP (QFI 7) qui nécessite une allocation de débit constant de 128 kbps et la navigation web (QFI 9) qui génère 3 Mbps de trafic. Le slice  $S_1$  est dédié aux applications de streaming vidéo (QFI 6)  $A_1$  qui consomment 10 Mbps. Le dernier network slice  $S_2$  est dédié aux applications de streaming  $A_2$  qui consomment 4 Mbps. Chaque UE établit des sessions PDU  $p_m(u_n, S_i)$  sur les slices  $S_i$  et commence à générer le trafic correspondant à l’ensemble des applications  $A_i$ .

Nous choisissons trois slices et deux liens physiques NTN afin de valider la redirection du trafic au sein des slice classifiers entre les deux liens NTN. Le multiplexage de  $S_0$  et  $S_1$  sur le lien  $L$  est effectué pour valider les communications entre les plans de contrôle satellites et 5G ainsi que la programmation dynamique des classifiers et leurs algorithmes de gestion de trafic entre les network slices et l’application des politiques de QoS. Les durées d’exécution différentes des slices mettent en évidence le fonctionnement réactif de nos mécanismes et des communications du plan de contrôle qui s’exécutent dynamiquement lorsque de nouveaux flux sont détectés.

### 5.3.2 Validation Expérimentale

Le scénario génère un testbed de 64 conteneurs fonctionnant sur une VM Ubuntu de 32 vCPU avec 128 Go de RAM. Dans notre scénario de raccordement, seuls les liens NTN sont disponibles entre les RAN et CN de la 5G. Ainsi, pour les modes SAW et SUAW, nous voulons également valider que le plan de contrôle 5G peut être envoyé sur des liens GEO ( $RTT \approx 500$  ms). Nous exécutons le testbed et définissons donc  $G$  comme lien par défaut pour le plan de contrôle 5G ce qui signifie que le trafic SCTP de l’interface N2 ne transitera que par le satellite géostationnaire. Les UE s’enregistrent avec succès sur le réseau 5G et sont capables d’établir des sessions PDU.

```

1  UERANSIM v3.2.6
2  [2022-06-03 12:32:52.134] [nas] UE switches to state [MM-DEREGISTERED/PLMN-SEARCH]
3  [2022-06-03 12:32:52.136] [rrc] New signal detected for cell[1], total [1] cells in
   ↪ coverage
4  [2022-06-03 12:32:52.137] [nas] Selected plmn[001/01]
5  [2022-06-03 12:32:54.635] [rrc] Selected cell plmn[001/01] tac[1] category[SUITABLE]
6  [2022-06-03 12:32:54.635] [nas] UE switches to state [MM-DEREGISTERED/PS]
7  [2022-06-03 12:32:54.635] [nas] UE switches to state [MM-DEREGISTERED/NORMAL-SERVICE]
8  [2022-06-03 12:32:54.635] [nas] Initial registration required due to
   ↪ [MM-DEREG-NORMAL-SERVICE]
9  [2022-06-03 12:32:54.635] [nas] UAC access attempt is allowed for identity[0],
   ↪ category[MO_sig]
10 [2022-06-03 12:32:54.635] [nas] Sending Initial Registration
11 [2022-06-03 12:32:54.635] [nas] UE switches to state [MM-REGISTER-INITIATED]
12 [2022-06-03 12:32:54.635] [rrc] Sending RRC Setup Request
13 [2022-06-03 12:32:54.636] [rrc] RRC connection established

```

```

14 [2022-06-03 12:32:54.636] [rrc] UE switches to state [RRC-CONNECTED]
15 [2022-06-03 12:32:54.636] [nas] UE switches to state [CM-CONNECTED]
16 [2022-06-03 12:32:54.711] [nas] Authentication Request received
17 [2022-06-03 12:32:54.711] [nas] Sending Authentication Failure due to SON out of range
18 [2022-06-03 12:32:54.764] [nas] Authentication Request received
19 [2022-06-03 12:32:54.816] [nas] Security Mode Command received
20 [2022-06-03 12:32:54.816] [nas] Selected integrity[2] ciphering[0]
21 [2022-06-03 12:32:54.882] [nas] Registration accept received
22 [2022-06-03 12:32:54.882] [nas] UE switches to state [MM-REGISTERED/NORMAL-SERVICE]
23 [2022-06-03 12:32:54.882] [nas] Sending Registration Complete
24 [2022-06-03 12:32:54.882] [nas] Initial Registration is successful
25 [2022-06-03 12:32:54.882] [nas] Sending PDU Session Establishment Request
26 [2022-06-03 12:32:54.882] [nas] UAC access attempt is allowed for identity[0],
↪ category[MO_sig]
27 [2022-06-03 12:32:55.266] [nas] PDU Session Establishment Accept received
28 [2022-06-03 12:32:55.266] [nas] PDU Session establishment is successful PSI[1]
29 [2022-06-03 12:32:55.273] [app] Connection setup for PDU session[1] is successful, TUN
↪ interface[uesimtun0, 10.0.222.2] is up.
30 [2022-06-03 12:32:55.266] [nas] PDU Session establishment is successful PSI[2]
31 [2022-06-03 12:32:55.273] [app] Connection setup for PDU session[2] is successful, TUN
↪ interface[uesimtun1, 10.0.223.2] is up.
32 [2022-06-03 12:32:55.266] [nas] PDU Session establishment is successful PSI[3]
33 [2022-06-03 12:32:55.273] [app] Connection setup for PDU session[3] is successful, TUN
↪ interface[uesimtun2, 10.0.224.2] is up.
34 [2022-06-03T12:32:56+00:00] Route to Data Network created successfully

```

### Implantation 1 – UE se raccordant au réseau et établissant des sessions PDUs

La capture de log 1 montre l'enregistrement avec succès d'un UE ayant comme identifiant l'imsi-001010000000001 ainsi que l'établissement avec succès des sessions PDU ayant les identifiants 1, 2 et 3. On peut voir que la SMF crée ces sessions PDU du côté du CN 5G avant de programmer les UPF et envoyer une notification à l'AMF qui va la relayer au UE sur la capture de log 2.

```

1 2022-06-03T12:32:48Z [36m [SMF][CFG] [0mSMF config version [1.0.2]
2 [...]
3 2022-06-03T12:32:48Z [36m [SMF][App] [0mSend PFCP Association Request to
↪ UPF[10.0.213.3]
4 2022-06-03T12:32:48Z [36m [SMF][App] [0mSend PFCP Association Request to
↪ UPF[10.0.210.3]
5 2022-06-03T12:32:48Z [36m [SMF][App] [0mSend PFCP Association Request to
↪ UPF[10.0.215.3]
6 2022-06-03T12:32:48Z [36m [SMF][PFCP] [0mIn HandlePfcPAssociationSetupResponse
7 2022-06-03T12:32:48Z [36m [SMF][PFCP] [0mHandle PFCP Association Setup Response with
↪ NodeID[10.0.210.3]
8 2022-06-03T12:32:48Z [36m [LIB][PFCP] [0mRemove Request Transaction [2]
9 2022-06-03T12:32:48Z [36m [SMF][PFCP] [0mUPF(10.0.210.3)[{sliceA}] setup association
10 2022-06-03T12:32:48Z [36m [SMF][PFCP] [0mIn HandlePfcPAssociationSetupResponse
11 2022-06-03T12:32:48Z [36m [LIB][PFCP] [0mRemove Request Transaction [1]
12 2022-06-03T12:32:48Z [36m [SMF][PFCP] [0mHandle PFCP Association Setup Response with
↪ NodeID[10.0.213.3]
13 2022-06-03T12:32:48Z [36m [SMF][PFCP] [0mUPF(10.0.213.3)[{sliceB}] setup association
14 2022-06-03T12:32:48Z [36m [SMF][PFCP] [0mIn HandlePfcPAssociationSetupResponse
15 2022-06-03T12:32:48Z [36m [LIB][PFCP] [0mRemove Request Transaction [3]
16 2022-06-03T12:32:48Z [36m [SMF][PFCP] [0mHandle PFCP Association Setup Response with
↪ NodeID[10.0.215.3]
17 2022-06-03T12:32:48Z [36m [SMF][PFCP] [0mUPF(10.0.215.3)[{sliceC}] setup association
18 2022-06-03T12:32:53Z [36m [SMF][PduSess] [0mReceive Create SM Context Request
19 2022-06-03T12:32:53Z [36m [SMF][PduSess] [0mIn HandlePDUSessionSMContextCreate

```

```

20 2022-06-03T12:32:53Z [36m [SMF][PduSess] [OmSend NF Discovery Serving UDM Successfully
21 [...]
22 2022-06-03T12:32:53Z [36m [SMF][GSM] [OmIn HandlePDUSessionEstablishmentRequest
23 2022-06-03T12:32:53Z [36m [SMF][PduSess] [OmPCF Selection for SMContext
   ↳ SUPI[imsi-001010000000001] PDUSessionID[1]
24 2022-06-03T12:32:53Z [36m [SMF][PduSess] [OmSUPI[imsi-001010000000001] has no
   ↳ pre-config route
25 2022-06-03T12:32:53Z [36m [SMF][Consumer] [OmSendNFDiscoveryServingAMF ok
26 [...]
27 2022-06-03T12:32:53Z [36m [SMF][GSM] [OmIn HandlePDUSessionEstablishmentRequest
28 2022-06-03T12:32:53Z [36m [SMF][PduSess] [OmPCF Selection for SMContext
   ↳ SUPI[imsi-001010000000001] PDUSessionID[2]
29 2022-06-03T12:32:53Z [36m [SMF][PduSess] [OmSUPI[imsi-001010000000001] has no
   ↳ pre-config route
30 2022-06-03T12:32:53Z [36m [SMF][Consumer] [OmSendNFDiscoveryServingAMF ok
31 [...]
32 2022-06-03T12:32:53Z [36m [SMF][GSM] [OmIn HandlePDUSessionEstablishmentRequest
33 2022-06-03T12:32:53Z [36m [SMF][PduSess] [OmPCF Selection for SMContext
   ↳ SUPI[imsi-001010000000001] PDUSessionID[3]
34 2022-06-03T12:32:53Z [36m [SMF][PduSess] [OmSUPI[imsi-001010000000001] has no
   ↳ pre-config route
35 2022-06-03T12:32:53Z [36m [SMF][Consumer] [OmSendNFDiscoveryServingAMF ok
36 [...]
37 2022-06-03T12:40:26Z [36m [SMF][Init] [OmTerminating SMF...
38 2022-06-03T12:40:26Z [36m [SMF][Consumer] [OmSend Deregister NFInstance
39 2022-06-03T12:40:27Z [36m [SMF][Init] [OmDeregister from NRF successfully

```

### Implantation 2 – Établissement des sessions PDUs par la SMF

Cette expérimentation valide l’exécution des procédures du plan de contrôle 5G sur des liens GEO ( $RTT \approx 500$  ms).

### 5.3.3 Évaluation des performances

Les résultats présentés dans cette section représentent la moyenne de 10 répétitions. Pour chaque flux dans les sessions PDU sur le forward link, nous mesurons le Packet Delay Budget (PDB) de bout en bout, le Packet Error Rate (PER), la gigue et le débit pour les modes SAW et SUAW. Nous agrégeons les flux par QFI et par network slice.

#### Slice-Unaware (SUAW)

À  $t_0 = 0s$ , les UE commencent à générer du trafic Web et VoIP sur  $S_0$  pendant 240s, et à  $t_1 = 60s$ , les UE génèrent du trafic streaming sur  $S_2$ . Comme les trafics  $S_0$  et  $S_2$  sont acheminés par des infrastructures satellites physiques différentes, les deux network slices sont isolées et aucun impact sur les performances ne peut être mesuré en modes SAW et SUAW de  $t_0 = 0s$  à  $t_2 = 120s$ .

De plus, que ce soit pour  $S_0$  ou  $S_2$ , le débit agrégé des flux (respectivement 28,15 Mbps et 26 Mbps) ne dépasse pas les spécifications de débit des slices (respectivement 35 Mbps et 40 Mbps) comme le montre la figure 5.3. Nous utilisons des applications de streaming isochrones basées sur un débit Variable Bitrate (VBR). Comme le lien NTN introduit un délai non négligeable entre les serveurs

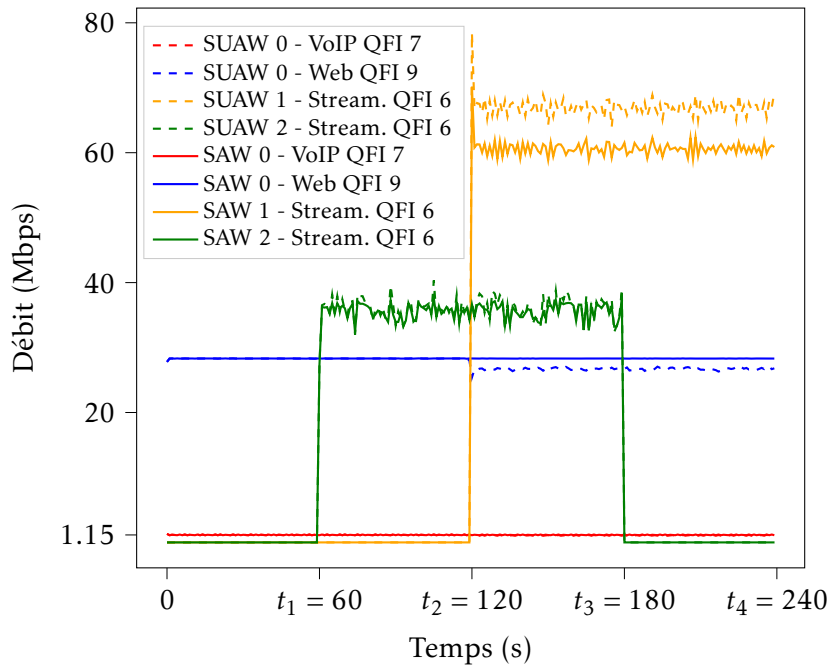


FIGURE 5.3 – Débit par QFI et par slice

de streaming et les UE, certains paquets sont considérés comme perdus par le récepteur de streaming qui est désynchronisé avec le serveur d'envoi. Ceci explique le PER sur la figure 5.4 pour l'application de streaming même dans des conditions idéales. Notez que nous n'utilisons pas de techniques de streaming adaptatif qui évaluent fréquemment la qualité de la liaison réseau et décident de réduire/améliorer la qualité des segments vidéos envoyés, et donc la consommation de bande passante. Par conséquent, l'application de streaming génère toujours un trafic isochrone entre les serveurs et les UE avec une consommation moyenne de bande passante de 36 Mbps. Jusqu'à  $t_2 = 120s$  les conditions réseaux sont idéales, les utilisateurs disposent de suffisamment de ressources pour les deux slices, il n'y a pas de pertes excessives comme le montre la figure 5.4, pas d'excès de PDB comme on le voit sur la figure 5.5 et la gigue oscille entre des valeurs maximales et minimales comme le montre la figure 5.6.

A  $t_2$ , les utilisateurs "avidés" commencent à consommer des ressources excessives au sein de  $S_1$  pour un total de 90 Mbps. Sans les mécanismes "slice-aware", il n'y a pas de notion de network slice dans les réseaux satellites, donc les liens NTN  $T_0^0$  et  $T_0^1$  logiques n'existent pas et on peut voir sur la figure 5.4 des chutes de performances drastiques dès qu'une congestion se produit sur le lien physique NTN. Le PER de  $S_0$ , nul jusqu'à  $t_2$ , augmente pour se stabiliser à environ 6% et où le PER de  $S_1$  oscille entre 22.5% et 30% du fait de la nature du trafic streaming (cf figure 5.4). Le débit total d'environ 120 Mbps dépasse la capacité du lien  $L$  de 100 Mbps, ainsi  $S_1$  chevauche et consomme les ressources de  $S_0$ . Le PDB des flux au sein de  $S_0$  connaît une augmentation de 78%, passant de 24 ms à 110 ms (cf figure

5.5), ce qui est encore acceptable pour les applications web et de streaming (QFI 6 et 9) mais pas pour celle de VoIP (QFI 7) en termes d'exigences de QoS.

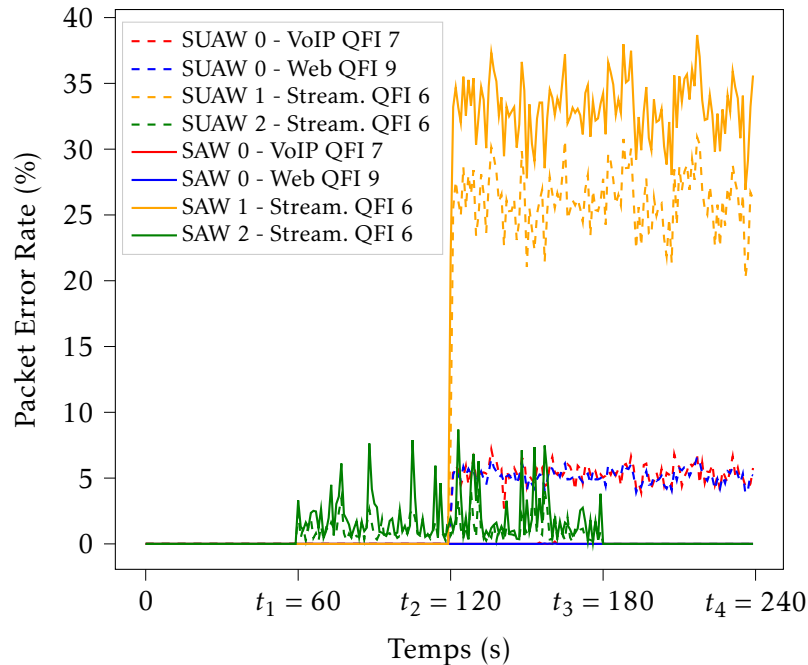


FIGURE 5.4 – Packet Error Rate par QFI et par slice

Les différents pics de PER des flux VoIP sont également inacceptables et ne respectent pas la spécification QFI 7 (cf figure 5.4 entre  $t_2$  et  $t_4$ ).

### Slice-Aware (SAW)

Comme mentionné dans la section 5.3.1, le fait d’être “slice-aware” nous permet de déployer des mécanismes appropriés pour garantir une continuité de slice et de QoS. Pour chaque slice, nous remplaçons les BNG par les slice classifiers, déployons la QOF dans le réseau 5G ainsi que la NTN QOF dans le réseau satellite. Nous déployons les mécanismes capables d’identifier chaque flux dans le réseau, déterminer à quel network slice il appartient, sa politique de QoS associée afin d’appliquer les politiques de traitement de trafic dédiées.

L’un des effets majeurs de ces mécanismes est l’isolation inter-slice ce qui prévient les surconsommations de ressources. Avec ces mécanismes déployés, des améliorations significatives sont apportées et les erreurs ne se propagent pas de  $S_1$  à  $S_0$ . De  $t_2$  à  $t_4$  (cf figure 5.4), on constate que  $S_0$  ne subit aucune perte de paquets que ce soit sur le trafic streaming ou VoIP, le taux de perte est nul. En revanche, le PER de  $S_1$  croît comparativement au scénario précédent qui augmente de 33.3% et passe de 25% à 35% car ses pertes de paquets ne sont plus partagées avec  $S_0$ . On constate aussi sur la figure 5.6 une légère augmentation de la gigue. Ces résultats montrent donc bien que les mécanismes proposés empêchent  $S_1$  de consommer les



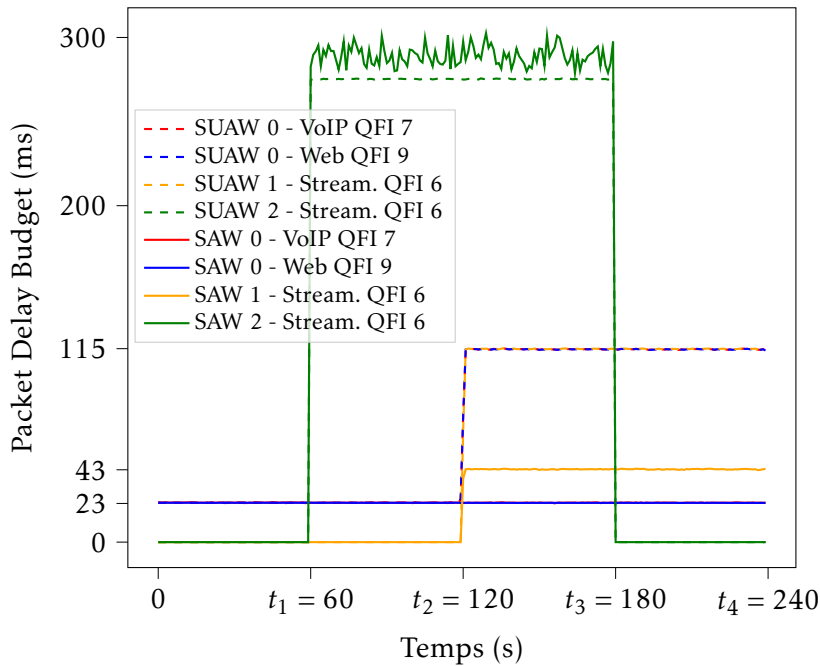


FIGURE 5.5 – Packet Delay Budget par QFI et par slice

ressources de  $S_0$  qui partagent pourtant le même lien physique NTN.

De plus, nous implantons un système de surveillance de débit (“traffic policing”) sur chaque classifieur afin d’éviter les congestions sur les liens NTN et assurer les KPIs du network slice. Les UE cherchent toujours à consommer un total de 90 Mbps, ce qui créerait une congestion sur  $L$  ( $90 + 35 > 110$  Mbps) et ne respecterait pas le SLA défini par  $\Theta_1$ . Le trafic du slice  $S_0$  devient donc prioritaire sur le trafic de  $S_1$  et les slice classifieurs appliquent les politiques d’ordonnancement et de files d’attente dédiés ce qui réduit la consommation de  $S_1$  à un maximum de 60 Mbps (cf figure 5.3 entre  $[t_2, t_4]$ ). L’isolation et la régulation de trafic étant opérante, le débit cible de  $S_0$  est toujours atteint et ne souffre pas du chevauchement de  $S_1$ .

On voit sur la figure 5.6 que la gigue du trafic VoIP est inférieure à la gigue du trafic streaming sur toute la durée du scénario en mode SAW, cela nous montre que la QoS est préservée et le trafic VoIP prime sur les autres. On note que même de  $t_2$  à  $t_4$ , lorsque  $S_1$  est actif, il n’y a aucune altération du trafic de  $S_0$  et la gigue reste constante.

Le PDB mesuré des services web et VoIP au sein de  $S_0$  reste à une moyenne de 23 ms pendant toute la durée du scénario soit une réduction de 133.3% comparé au scénario SUAW comme le montre la figure 5.5. Les flux streaming de  $S_1$  quant à eux subissent des pertes et donc un PDB de 43 ms en moyenne. On a donc aussi une amélioration du PDB pour ces flux comparé au scénario précédent mais cela s’explique en partie par la perte de près d’un tiers des paquets. Cependant, le réseau NTN est constamment informé par le réseau 5G des flux qui apparaissent

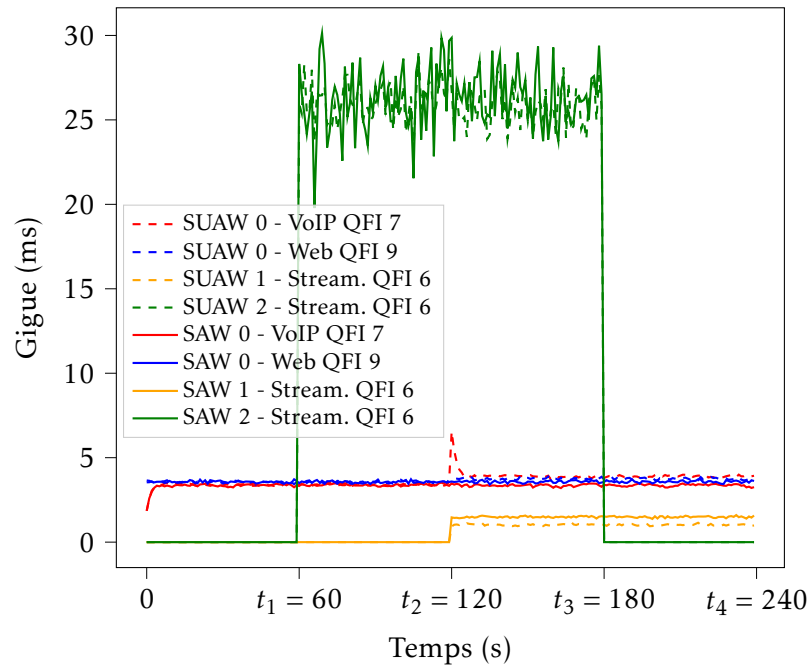


FIGURE 5.6 – Gigage par QFI et par slice

dans le réseau, et donc dans on pourrait choisir de discriminer des flux plus précisément dans une version améliorée de notre travail.

$S_2$  fonctionne toujours de  $t_1 = 60s$  à  $t_3 = 180s$  sans que les performances de  $S_0$  et  $S_1$  ne soient impactées et réciproquement. Ces résultats démontrent la pertinence de notre modèle et de nos propositions qui assurent une isolation inter-slice ainsi qu'une continuité de slice et de QoS au sein du segment NTN. Cela souligne l'importance de l'adoption du paradigme de network slicing dans les réseaux de satellites pour une intégration avec les réseaux 5G.

Nous avons aussi démontré l'efficacité de notre plateforme d'expérimentation COMETS en exécutant des scénarios complexes de network slicing intégrant les réseaux satellite et 5G. De plus, nous l'avons conçu pour être utilisable et extensible à n'importe quel scénario par l'ensemble de la communauté de recherche. Cette plateforme constitue dorénavant une base pour la validation expérimentale de nos propositions car elle accélère le développement et l'évaluation de nouvelles architectures. Des améliorations ont déjà été apportées et de nouvelles propositions sont en cours de validation sur COMETS.

### 5.3.4 Conclusion

Dans ce chapitre, nous avons présenté une implantation de notre modèle et de notre architecture de network slicing pour intégrer le réseau satellite dans les réseaux 5G en tant que lien NTN tenant compte des network slice. Nous avons résumé dans le tableau 5.2 les différents défis que nous avons validé avec nos propositions.

TABLE 5.2 – Propositions

Défis	Références	Atteint
Politique de QoS indépendante entre chaque slice	iv	✓
Continuité de QoS au sein des segments	v	✓
Indépendance dans la gestion des segments réseaux	vii	✓
Intégration transparente	viii	✓
Assurer les performances du slice de bout-en-bout	xiii	✓

Les résultats montrent clairement que notre approche permet une continuité de slice bout-en-bout tout en assurant le respect des exigences QoS au sein de chaque slice. De plus, l'isolation des ressources est préservée à travers les domaines du réseau grâce aux mapping de classes de QoS et aux politiques appliquées par les slice classifieurs. L'architecture qui met en œuvre notre modèle est basée sur le 3GPP SBA et les nouveaux services que nous introduisons sont intégrés de manière transparente dans l'architecture globale, ce qui minimise l'effort d'adoption pour les acteurs du satellite et de la 5G. De plus, notre architecture maintient des domaines de gestion indépendants pour la 5G et le satellite, ce qui est une exigence primordiale pour les deux opérateurs. Les interactions entre ces opérateurs sont limitées à la gestion du cycle de vie des slices (par exemple, le provisionnement d'un slice NTN spécifique) puisque l'interconnexion des plans de contrôle et de données est entièrement automatisée.

# Conclusion et Perspectives

## Conclusion

Dans ce travail, nous nous sommes intéressés à l'intégration des réseaux satellites dans les réseaux 5G en utilisant le paradigme de network slicing. Nos objectifs étaient multiples : le premier était d'améliorer la flexibilité des réseaux satellites afin d'en simplifier l'interconnexion avec les réseaux externes tout en homogénéisant leur gestion. L'arrivée des mega-constellations broadband replace le satellite comme un moyen de télécommunication accessible par tous et ayant des performances qui supplantent celles de certains réseaux d'accès terrestres et mobiles (e.g. ADSL, 3G, 4G) en termes de latence, de débits et zone de couverture. C'est pourquoi, notre second objectif était de proposer une méthode d'intégration des réseaux satellites dans les réseaux 5G en les considérant comme des liens NTN qui sont des réseaux officiellement supportés par le 3GPP.

Pour atteindre nos objectifs, nous avons effectué un état de l'art des technologies mobiles terrestres, des technologies satellites broadband et des différents travaux qui ont été effectués pour une intégration de ces réseaux. Deux familles d'intégration peuvent être distinguées : l'intégration directe où les réseaux satellites peuvent être utilisés comme réseau d'accès 5G et l'intégration indirecte où les réseaux satellites sont exploités comme lien de communication entre le RAN et le CN 5G. Nous avons poursuivi un type d'intégration indirecte en se fondant sur le paradigme de network slicing qu'ont adopté les réseaux 5G. Cette intégration constitue une première étape, nécessaire à une intégration plus complète et plus optimale des réseaux satellites dans les réseaux 5G.

Le paradigme de network slicing nous semble être un outil idéal pour espérer atteindre cette optimalité. Nous avons donc par la suite effectué une revue complète de ce concept dans les réseaux terrestres, défini son objectif, listé les technologies sur lesquelles le slicing repose, listé toutes les propriétés et les défis associés au slicing pour son adoption dans les réseaux. Dans le but de générer des network slices satellite, nous avons confronté le slicing aux réseaux satellites et proposé une architecture plus flexible qui permettrait d'instancier des liens satellites à la demande et les intégrer dans les réseaux 5G.

Un enjeu majeur est alors la garantie d'une QoS homogène de bout-en-bout au sein du slice. Dans le but d'intégrer ces liens satellites dans les réseaux 5G, nous avons alors proposé une méthode d'intégration où les liens NTN relaient le trafic entre le RAN 5G et le CN tout en assurant une continuité de slice au sein du réseau satellite et une continuité de QoS. Pour ce faire, nous nous sommes basés sur les spécifications 3GPP et nous avons proposé une interface de communication entre les systèmes de gestion satellite et 5G, et nous avons introduit une méthode de négociation de QoS dynamique qui permet d'adapter le support de la QoS entre les réseaux 5G et satellite. Nous avons rajouté des fonctions réseaux dans les plans de

contrôle satellite et 5G afin d'identifier chaque flux des slices et déployer dynamiquement des règles sur des slices classifiées qui sont des éléments d'interconnexion dans le plan de données et qui assurent cette continuité de slice et de QoS.

Enfin, pour valider nos propositions, nous avons développé une plateforme d'expérimentation réseau COMETS basée sur la virtualisation par conteneurs. Cette plateforme permet de définir ses propres scénarios réseaux utilisant des composants existants ou développés spécifiquement dans un contexte précis et d'automatiser leur déploiement et leur évaluation de performances. En utilisant cette plateforme, nous avons implanté nos scénarios de network slicing intégrant les réseaux satellites et 5G et validé nos contributions. Ce déploiement nous a permis de valider nos contributions aussi bien d'un point de vue fonctionnel qu'en termes de performances. L'outil COMETS est de plus maintenant disponible pour la communauté.

## Perspectives

Les recherches menées au cours de cette thèse ont amené à faire un certain nombre de considérations concernant l'étendue de la problématique abordée. Nous présentons ici diverses ouvertures de recherche.

### Expérimentation avec un RAN 5G et des liens satellites physiques

Il s'agit ici de reprendre l'expérimentation et la plateforme qui a été développée dans le chapitre 5 afin d'y intégrer un RAN physique 5G et utiliser des liens satellites physiques. Cela permettrait de démontrer que notre méthode est applicable aux réseaux opérateurs actuellement déployés.

Nous avons d'ores et déjà entamé les travaux et avons des résultats préliminaires en utilisant une gNB Amarisoft [193], un smartphone 5G et un lien satellite fourni par la mega-constellation Starlink de SpaceX. N'ayant pas accès au système de gestion Starlink, obtenir des résultats a nécessité d'abord d'évaluer le fonctionnement de cette mega-constellation et de déployer les mécanismes nécessaires au raccordement d'un RAN 5G au terminal Starlink.

### Modélisation uniforme des ressources dans le slice de bout-en-bout

Un network slice est la composition de plusieurs segments réseaux appartenant à différents domaines administratifs comme nous l'avons souligné dans le chapitre 2. Chaque segment possède ses propres ressources et son propre système de gestion qui doivent répondre aux KPIs du slice. Lors de l'établissement d'un network slice, il est de la responsabilité de chaque fournisseur de segment d'assurer l'allocation de ressources adéquate et de la maintenir pendant toute la durée de vie du slice.

Notre méthode d'interconnexion présentée dans le chapitre 4 nous permet de constamment superviser le réseau et de connaître le nombre exact de flux au sein

de chaque segment NTN. En généralisant cette approche sur tous les segments, il serait donc possible de caractériser précisément chaque flux en termes de quantité de ressources nécessaires au sein de chaque segment puis de quantifier les ressources nécessaires pour un flux au sein du slice de bout-en-bout. On pourrait par la suite agréger ces ressources et proposer des modèles d'uniformisation qui caractériseraient les besoins de chaque flux en fonction des KPIs du slice. Cela permettrait d'appliquer des algorithmes d'auto-scaling, d'optimiser l'allocation des ressources au niveau de chaque segment et ainsi améliorer la gestion de bout-en-bout du slice.

### **Extension de notre méthode d'intégration à d'autres NTN et à d'autres réseaux de transport**

Notre méthode d'intégration a été définie en prenant comme cas d'usage les réseaux satellites. Ils font partie de la famille des NTN qui incluent d'autres réseaux notamment les High-altitude Platform Station (HAPS) (e.g. drone, ballons atmosphériques). Nous pensons que notre méthode est assez générique et serait applicable à ces réseaux qui ont aussi leurs spécificités. En plus des réseaux NTN, il serait aussi intéressant d'appliquer nos mécanismes sur des liens terrestres actuels ne supportant pas le network slicing. En effet, les fonctions réseaux que nous avons introduites sont génériques et peuvent aussi être ajoutées dans d'autres types de réseaux.

Nous serions alors, dans le cas d'une expérimentation et d'une validation, en possession d'une méthode totalement générique qui permettrait à n'importe quel réseau de transport de facilement s'intégrer au sein des réseaux 5G et supporter la continuité de service et de QoS.

# Bandes de fréquences du 5G New Radio

---

## A.1 Frequency Range 2 $\geq$ 24 Ghz


 — Bandes de fréquences partagées avec les réseaux satellites

TABLE A.1 – Fréquence 5G FR2 [194]

Bande	Fréquence (Ghz)	Uplink / Downling (Ghz)
n257	28	26.50 — 29.50
n258 (K-band)	26	24.25 – 27.50
n259 (V-band)	41	39.50 — 43.50
n260 (Ka-band)	39	37.00 — 40.00
n261 (Ka-band)	28	27.50 — 28.35
n262 (V-band)	47	47.20 — 48.20

# Plateforme COMETS

---

Implantation en Python de la plateforme COMETS qui génère un fichier `docker-compose.yaml` qui est exécuté par le runtime docker. A la fin de l'exécution du scénario, COMETS agrège les résultats et évalue les performances.

## B.1 Fichier de configuration des scénarios

```
1  saw-ntn: # Slice Aware Scenario
2      type: "slice-aware"
3      duration: 240
4      user: 9 # Number of UE to simulate
5      links: # NTN Link list
6          - default: false # NTN Link parameters
7            forward: 100
8            return: 25
9            delay: 45
10           jitter: 5
11           acm: false
12           slices: # Slice within the NTN Link
13               - start: 0
14                 end: 240
15               theta: # 5G requested parameters
16                   lambda: 45
17                   delta: 5
18                   mu: 10
19                   beta: 35
20                   sigma: true
21           applications: # Applications in the slice
22               - name: voip
23                 data_rate: 128 # kbps
24               - name: web
25                 data_rate: 3 # mbps
26 suaw-ntn: # Slice Unaware Scenario
27     type: "non-slice-aware"
28     duration: 240
29     user: 9
30     links: # NTN Link list
31         - default: false
32           forward: 100
33           return: 25
34           delay: 45
35           jitter: 5
36           acm: false
```



```

37     slices: # Slice within the NTN Link
38         - start: 0
39           end: 240
40           theta:
41             lambda: 45
42             delta: 5
43             mu: 10
44             beta: 35
45             sigma: true
46         applications:
47             - name: voip
48               data_rate: 128
49             - name: web
50               data_rate: 3
51

```

### Implantation 3 – Définition des scénarios

## B.2 Scénario générique

```

1  class Scenario(object):
2      """Generic Scenario Interface"""
3
4      validation_schema = None
5      scenario_type = "generic"
6
7      def __init__(self, definition: Dict) -> None:
8          self.name = definition['name']
9          self.definition = definition
10         self.networker = Networker()
11         self.repository = Repository()
12
13     def set_path(self, results_folder: str, containers_folder: str,
14 ↪ configurations_folder: str, output_configuration_folder: str,
15 ↪ scenario_folder: str):
16         """
17         Set the path for the output folders
18         """
19
20         self.repository.results_folder = results_folder
21         self.repository.containers_folder = containers_folder
22         self.repository.configurations_folder = configurations_folder
23         self.repository.output_configuration_folder =
24 ↪ output_configuration_folder
25         self.repository.scenario_folder = scenario_folder
26
27     def prepare_scenario(self) -> None:
28         """
29         Prepare all the requirements prior to the generation of the scenario
30 ↪ topology

```

```
27         """
28
29         logging.info(
30             f"[{self.name}] prepare_scenario function undefined, I QUIT !")
31         sys.exit(1)
32
33     def generate_networks(self) -> None:
34         """
35         Generate the networks of the testbed
36         """
37
38         logging.info(
39             f"[{self.name}] generate_networks function undefined, I QUIT !")
40         sys.exit(1)
41
42     def generate_topology(self) -> None:
43         """
44         Create all the services and generate the network topology of the
45 ↪ scenario
46         """
47
48         logging.info(
49             f"[{self.name}] generate_topology function undefined, I QUIT !")
50         sys.exit(1)
51
52     def configure_services(self) -> None:
53         """
54         Configure all the services
55         """
56
57         for service in sorted(self.repository.services):
58             self.repository.services.get(service).configure(self.repository)
59
60     def configure_compose(self) -> None:
61         """
62         Configure the services docker compose part
63         """
64
65         for service in sorted(self.repository.services):
66             self.repository.services.get(
67                 service).configure_compose(self.repository)
68
69     def configure_entrypoint(self) -> None:
70         """
71         Configure the services entrypoint
72         """
73
74         for service in sorted(self.repository.services):
75             self.repository.services.get(
76                 service).configure_entrypoint(self.repository)
77
78     def write_configuration(self) -> None:
79         """
```

```

79     Write the configuration files of each service
80     """
81
82     for service in sorted(self.repository.services):
83         s = self.repository.services.get(service)
84
85         ↪ s.write_configuration(self.repository.output_configuration_folder)
86
87     def write_entrypoint(self) -> None:
88         """
89         Write entrypoints for each service
90         """
91
92         for service in sorted(self.repository.services):
93             s: Service = self.repository.services.get(service)
94             s.write_entrypoint(self.repository)
95
96     def write_compose(self) -> None:
97         """
98         Write docker-compose.yaml file
99         """
100
101         output = {
102             "version": "3.9",
103             "services": {},
104             "networks": {},
105             "volumes": {"dbdata": None}
106         }
107
108         for service in sorted(self.repository.services):
109             if self.repository.services[service].compose != None:
110                 output["services"][service] =
111                 ↪ self.repository.services[service].compose.generate()
112
113         network_index: int = 0
114         for network in sorted(self.networker.networks_name):
115             output["networks"][network] = {
116                 "name": network,
117                 "driver": "bridge",
118                 "ipam": {
119                     "driver": "default",
120                     "config": [
121                         {
122                             "subnet": str(
123                                 self.networker.networks_name.get(network))
124                         }
125                     ]
126                 },
127                 "driver_opts":
128                 {
129                     "com.docker.network.bridge.name":
130                     ↪ f"network-{{network_index}}"
131                 }

```

```
129         }
130         network_index += 1
131
132         path: str = f"{self.repository.scenario_folder}/{self.name}/docker-
↪ compose.yaml"
133         yam = yaml.YAML()
134         yam.indent(sequence=4, offset=2)
135         with codecs.open(path, "w", encoding="utf-8") as compose_file:
136             yam.dump(output, compose_file)
137
138     def run(self) -> None:
139         logging.info(f"[{self.name}] run function undefined, I QUIT !")
140         sys.exit(1)
```

Implantation 4 – Scénario générique COMETS

### B.3 Cycle de vie des scénarios

```
1  scenario = Scenario()
2
3  scenario.prepare_scenario()
4  scenario.generate_networks()
5  scenario.generate_topology()
6  scenario.configure_services()
7  scenario.write_configuration()
8  scenario.configure_compose()
9  scenario.configure_entrypoint()
10 scenario.write_entrypoint()
11 scenario.write_compose()
```

Implantation 5 – Cycle de vie des scénarios COMETS

# Serveurs d'applications IPerf

Implantation en Python dans la plateforme COMETS des serveurs applicatifs utilisant l'outil IPerf.

## C.1 Configuration des clients et serveurs IPerf

```

1  class VoIP(Application):
2
3      def __init__(self, data_rate=128):
4          super().__init__("VoIP")
5          self.qi = 7
6          # EF
7          self.ntn_dscp = 0x2c
8          # AF
9          self.terrestrial_dscp = 0x2e
10         self.port = 5060
11         self.data_rate = data_rate
12         self.pdb = 100
13         self.code = APPLICATION_VOIP
14
15         def generate_server(self, bind, sl, server):
16             return f"iperf -s -B {bind} -p {self.port} -u -l 1400 -i 1 -f b --tos
17                 ↪ {self.generate_tos()} > {self.get_log_server(server, sl)}"
18
19         def generate_client(self, bind, destination, ue, sl, duration):
20             return f"iperf -c {destination} -B {bind} -p {self.port} --trip-times
21                 ↪ --tos {self.generate_tos()} -i 1 -u -l 1400 --reverse -t
22                 ↪ {duration}s -b {self.data_rate}k -f b > {self.get_log_client(ue,
23                 ↪ sl)} &"
24
25         def get_data_rate(self):
26             return f"{self.data_rate} kbit/s"
27
28 class Streaming(Application):
29
30     def __init__(self, data_rate=5):
31         super().__init__("Streaming")
32         self.qi = 6
33         # AF12
34         self.ntn_dscp = 0x0c
35         # AF12
36         self.terrestrial_dscp = 0x0c

```

```

34     self.port = 8085
35     self.data_rate = data_rate
36     self.pdb = 300
37     self.code = APPLICATION_STREAMING
38
39     def generate_server(self, bind, sl, server):
40         return f"iperf -s -B {bind} -p {self.port} -u -l 1400 -i 1 -f b --tos
↳ {self.generate_tos()} > {self.get_log_server(server, sl)}"
41
42     def generate_client(self, bind, destination, ue, sl, duration):
43         return f"iperf -c {destination} -B {bind} -p {self.port} --trip-times
↳ --tos {self.generate_tos()} -i 1 -u -l 1400 --reverse -t
↳ {duration}s
↳ --isochronous=25:{self.data_rate}m,{round(self.data_rate * 0.2)}m
↳ -f b > {self.get_log_client(ue, sl)} &"
44
45     def get_data_rate(self):
46         return f"{self.data_rate} Mbit/s"
47
48
49 class Web(Application):
50
51     def __init__(self, data_rate=3):
52         super().__init__("Web")
53         self.qi = 9
54         # AF11
55         self.ntn_dscp = 0x0a
56         # AF11
57         self.terrestrial_dscp = 0x0a
58         self.port = 8080
59         self.data_rate = data_rate
60         self.pdb = 300
61         self.code = APPLICATION_WEB
62
63     def generate_server(self, bind, sl, server):
64         return f"iperf -s -B {bind} -p {self.port} -b {self.data_rate}M -u -l
↳ 1400 -i 1 -f b --tos {self.generate_tos()} >
↳ {self.get_log_server(server, sl)}"
65
66     def generate_client(self, bind, destination, ue, sl, duration):
67         return f"iperf -c {destination} -B {bind} -p {self.port} --trip-times
↳ --tos {self.generate_tos()} -i 1 -u -l 1400 --reverse -t
↳ {duration}s -b {self.data_rate}M -f b > {self.get_log_client(ue,
↳ sl)} &"
68
69     def get_data_rate(self):
70         return f"{self.data_rate} Mbit/s"

```

### Implantation 6 – Configuration des applications IPerf

# Slice Classifier

Implantation en Golang du Slice Classifier qui expose une API nord REST pour la configuration et qui repose sur iptables ainsi que linux TC.

## D.1 Structures de données

```

1  type PDU struct {
2      TEID      uint32 `json:"teid" yaml:"teid" bson:"teid"`
3      DSCP5     uint8  `json:"dscp_5g" yaml:"dscp_5g" bson:"dscp_5g"`
4      DSCPS     uint8  `json:"dscp_satellite" yaml:"dscp_satellite"
5      ↪ bson:"dscp_satellite"`
6      SliceID  uint8  `json:"slice_id" yaml:"slice_id" bson:"slice_id"`
7      Endpoint string `json:"endpoint" yaml:"endpoint" bson:"endpoint"`
8      IPv4     string `json:"ipv4" yaml:"ipv4" bson:"ipv4"`
9      Ingress  string `json:"ingress" yaml:"ingress" bson:"ingress"`
10     IsRAN     bool   `json:"is_ran" yaml:"is_ran" bson:"is_ran"`
11 }
12
13 type ADMControl struct {
14     SliceID  uint8  `json:"slice_id" yaml:"slice_id" bson:"slice_id"`
15     Throughput int    `json:"throughput" yaml:"throughput"
16     ↪ bson:"throughput"`
17     Endpoint string `json:"endpoint" yaml:"endpoint" bson:"endpoint"`
18 }
19
20 type ADM struct {
21     Controls []ADMControl `json:"controls" yaml:"controls"
22     ↪ bson:"controls"`
23     Aware    bool          `json:"slice_aware" yaml:"slice_aware"
24     ↪ bson:"slice_aware"`
25 }
26
27 type ClassID struct {
28     Root int
29     VoIP int
30     Rest int
31 }
32
33 type Classification struct {
34     Classes map[uint8]ClassID
35     Queues  map[string]int
36     Ifaces  []string
37 }

```

```

34
35 type Runtime struct {
36     iptable *iptables.IPTables
37     Aware   bool
38 }

```

### Implantation 7 – Structures de données

## D.2 Configuration des files d'attentes

```

1  func (c *Classification) buildTree(Controls []ADMControl) error {
2
3      for k, adm := range Controls {
4          log.Println(k, adm.Endpoint, adm.SliceID, adm.Throughput)
5          iface, err := FindInterface(adm.Endpoint)
6          if err != nil {
7              return err
8          }
9
10         if !c.queueIsSet(iface) {
11             runTC("qdisc", "add", "dev", iface, "root", "handle",
↪ "1:", "htb", "r2q", "10")
12             c.Ifaces = append(c.Ifaces, iface)
13             c.Queues[iface] = 1
14         }
15
16         voipClass := c.Queues[iface]*10 + 2
17         restClass := c.Queues[iface]*10 + 1
18         c.Classes[adm.SliceID] = ClassID{
19             VoIP: voipClass,
20             Rest: restClass,
21             Root: c.Queues[iface],
22         }
23
24         rootClass := fmt.Sprintf("1:%d", c.Queues[iface])
25         runTC("class", "add", "dev", iface, "parent", "1:",
↪ "classid", rootClass, "htb", "rate", fmt.Sprintf("%dmbit",
↪ adm.Throughput), "burst", c.calculateBurst(adm.Throughput), "cburst",
↪ c.calculateBurst(adm.Throughput), "mtu", "1500")
26
27         voipClasss := fmt.Sprintf("1:%d", voipClass)
28         restClasss := fmt.Sprintf("1:%d", restClass)
29         runTC("class", "add", "dev", iface, "parent", rootClass,
↪ "classid", voipClasss, "htb", "rate", "2mbit", "prio", "0", "burst",
↪ "3k", "cburst", "3k", "mtu", "1500")
30         runTC("qdisc", "add", "dev", iface, "parent", voipClasss,
↪ "sfq", "perturb", "10")
31         runTC("class", "add", "dev", iface, "parent", rootClass,
↪ "classid", restClasss, "htb", "rate", fmt.Sprintf("%dmbit",
↪ adm.Throughput-1), "ceil", fmt.Sprintf("%dmbit", adm.Throughput-1),
↪ "prio", "1", "burst", c.calculateBurst(adm.Throughput-1), "cburst",
↪ c.calculateBurst(adm.Throughput-1), "mtu", "1500")

```



```

32         runTC("qdisc", "add", "dev", iface, "parent", restClass,
↳ "sfq", "perturb", "10")
33
34         c.Queues[iface]++
35     }
36
37     return nil
38 }

```

### Implantation 8 – Configuration des files d’attentes

## D.3 Runtime

### D.3.1 Initialisation du Runtime

```

1  func InitRuntime(flush bool, config string) error {
2      err := ParseConf(config)
3      if err != nil {
4          log.Println(err.Error())
5          return err
6      }
7      table, err := iptables.New()
8      if err != nil {
9          return err
10     }
11
12     table.ClearAndDeleteChain(MANGLE_TABLE, PIPE_CHAIN)
13     table.ClearAndDeleteChain(MANGLE_TABLE, SLICE_CHAIN)
14     table.ClearAndDeleteChain(MANGLE_TABLE, SLICE_CHAIN)
15     table.ClearAndDeleteChain(NAT_TABLE, PREROUTING_CHAIN)
16
17     table.NewChain(MANGLE_TABLE, PIPE_CHAIN)
18     table.NewChain(MANGLE_TABLE, SLICE_CHAIN)
19     table.NewChain(MANGLE_TABLE, QOS_CHAIN)
20
21     table.AppendUnique(MANGLE_TABLE, PREROUTING_CHAIN, "-j", PIPE_CHAIN)
22     table.AppendUnique(MANGLE_TABLE, PREROUTING_CHAIN, "-j", SLICE_CHAIN)
23     table.AppendUnique(MANGLE_TABLE, PREROUTING_CHAIN, "-j", QOS_CHAIN)
24
25     classifier = &Classification{
26         Classes: make(map[uint8]ClassID),
27         Queues:   make(map[string]int),
28     }
29
30     runtime = &Runtime{
31         iptable: table,
32         Aware:   false,
33     }
34     InitRouter(false, false)
35     return nil

```

```
36 }
```

## Implantation 9 – Initialisation du Runtime

### D.3.2 Nouvelle Session PDU

```

1  func (r *Runtime) NewPDU(pdu *PDU) error {
2
3      if !r.Aware {
4          return nil
5      }
6
7      label := slicing.NewLabel(pdu.SliceID, true, 0x2e)
8
9      iface, err := FindInterface(pdu.Ingress)
10     if err != nil {
11         return err
12     }
13     oface, err := FindInterface(pdu.Endpoint)
14     if err != nil {
15         return err
16     }
17
18     var dscp5GString string = fmt.Sprintf("0x%s",
↳ hex.EncodeToString([]byte{0x2e}))
19
20     u32GTP := u32.BuildMatchGTP(pdu.TEID, pdu.IPv4, pdu.IsRAN)
21
22     // Add marking to incoming packets
23     runtime.iptable.AppendUnique(MANGLE_TABLE, PIPE_CHAIN, "-i", iface,
↳ "-p", "udp", "--dport", "2152", "--sport", "2152", "-j", "MARK",
↳ "--set-xmark", label.GeneratePipe())
24     runtime.iptable.AppendUnique(MANGLE_TABLE, QOS_CHAIN, "-i", iface,
↳ "-p", "udp", "--dport", "2152", "--sport", "2152", "-m", "dscp",
↳ "--dscp", dscp5GString, "-j", "MARK", "--set-xmark",
↳ label.GenerateDSCP())
25     runtime.iptable.AppendUnique(MANGLE_TABLE, SLICE_CHAIN, "-i", iface,
↳ "-m", "u32", "--u32", u32GTP, "-j", "MARK", "--set-xmark",
↳ label.GenerateSliceID())
26
27     voipClass := classifier.Classes[pdu.SliceID].VoIP
28     restClass := classifier.Classes[pdu.SliceID].Rest
29     voipClasss := fmt.Sprintf("1:%d", voipClass)
30     restClasss := fmt.Sprintf("1:%d", restClass)
31
32     runtime.iptable.AppendUnique(MANGLE_TABLE, POSTROUTING_CHAIN, "-o",
↳ oface, "-m", "mark", "--mark", label.GeneratePipeSliceID(), "-j",
↳ "CLASSIFY", "--set-class", restClasss)
33     runtime.iptable.AppendUnique(MANGLE_TABLE, POSTROUTING_CHAIN, "-o",
↳ oface, "-m", "mark", "--mark", label.Generate(), "-j", "CLASSIFY",
↳ "--set-class", voipClasss)

```

```
34  
35     return nil  
36 }  
37
```

Implantation 10 – Nouvelle session PDU

# Définitions

- Artificial Intelligence (AI)** Intelligence Artificielle. 23
- Baseband Unit (BBU)** Dispositif au sein d'une eNB qui interprète et traite les fréquences en bande de base.. 21
- Blueprint** Descriptif d'un network slice.. 35
- Central Unit (CU)** Élément de contrôle d'une gNB.. 19, 21, 23, 45, 50
- Commercial off-the-shelf (COTS)** Matériel générique. 10, 20, 61, 63
- Common Public Radio Interface (CPRI)** Interface de communication entre les éléments de contrôle radios et les éléments de donnée radio. 21
- Communication Service Customer (CSC)** Client d'un service de communication. 38
- Communication Service Provider (CSP)** Fournisseur d'un service de communication. 38, 40
- Core Network (CN)** Cœur de réseau. 15, 16, 17, 18, 19, 20, 21, 27, 29, 30, 31, 40, 42, 43, 48, 50, 51, 52, 53, 54, 55, 58, 59, 63, 65, 74, 76, 78, 79, 81, 82, 84, 85, 92, 96, 100, 101, 102, 109
- Data Center Service Provider (DCSP)** Opérateur de data center. 38
- Distribution Unit (DU)** Élément de donnée d'une gNB.. 19, 21, 23, 45
- End User** Utilisateur final. 40, 42
- End-to-end (E2E)** Bout en bout. xii, 1, 15, 19, 30, 31, 32, 34, 40, 43, 47, 48, 50, 52, 54, 55, 56, 60, 74, 75, 76, 77, 78, 79, 81, 82, 85, 88, 100
- enhanced Common Public Radio Interface (CPRI)** Interface de communication entre les éléments de contrôle radios et les éléments de donnée radio améliorée. 21
- Evolved Node B (gNB)** Équipement permettant le raccordement des utilisateurs au réseau d'accès radio 4G.. 15, 18, 19, 20, 21
- Forward Link** Liaison aller.. 60, 69, 103
- Fronthaul** Réseau de transport qui relie les RUs aux DUs dans l'architecture Cloud-RAN.. 21
- Gateway** Passerelle réseau. 4, 16, 25, 26, 31, 58, 60, 61, 63, 64, 65, 66, 68, 70, 71
- GPRS Tunneling Protocol (GTP)** Protocole d'encapsulation défini initialement pour les réseaux mobiles GPRS, il se définit par GTP-C dans le plan de contrôle et GTP-U dans le plan de données.. 43, 97

- Hardware Supplier** Fournisseur d'équipement physique. 38
- Key Performance Indicator (KPI)** Indice de performance. 3, 30, 34, 35, 40, 48, 54, 69, 71, 75, 76, 77, 83, 84, 106, 110, 111
- Management Function (MnF)** Fonction de gestion. 42
- Management Service (MnS)** Service de gestion. 41, 42
- Management Function (MF)** Fonction réseau de gestion. 39
- Midhaul** Réseau de transport qui relie les CUs aux DUs dans l'architecture Cloud-RAN.. 21
- Mobile Edge Caching (MEC)** Technique de cache dans le réseau. 18, 41, 79
- Namespace** Espace de nom linux.. 94
- Network Resource Model (NRM)** Modèle de ressources réseaux. 42
- Network Service** Ensemble de fonctions réseaux qui permettent d'assurer un service. 40, 93
- Network Slice Subnet Instance (NSSI)** Instance d'un sous-segment de slice. 40, 76, 100
- Network Slice Instance (NSI)** Instance d'un network slice. 40, 43, 76, 77, 100
- Network Slice Subnet (NSS)** Sous-segment d'un slice. 40, 42, 43, 45, 46, 74, 75, 77, 78, 81, 82, 83, 85, 88
- Network Slice as a Service Customer (NSaaS)** Client de Network Slice as a Service. 38
- Network Slice as a Service Provider (NSaaSP)** Fournisseur de Network Slice as a Service. 38
- Network Equipment Provider (NEP)** Fournisseur d'équipement réseau. 38
- Network Operator (NOP)** Opérateur réseau. 22, 37, 38, 39, 40, 42, 43
- Network Function (NF)** Fonction réseau. 17, 18, 36, 48, 49, 60, 61, 65, 77, 85, 96
- Network Slice (NS)** Un network slice. xii, 1, 3, 31, 34, 37, 38, 40, 41, 42, 43, 44, 45, 46, 47, 50, 51, 52, 53, 54, 55, 57, 58, 59, 60, 61, 62, 63, 65, 66, 67, 68, 69, 70, 71, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 92, 96, 97, 100, 101, 103, 104, 105, 106, 108, 109, 110
- Network Slicing** Le network slicing. i, iii, xii, xiv, 1, 2, 3, 4, 13, 14, 15, 18, 31, 32, 33, 34, 37, 39, 40, 42, 43, 44, 45, 46, 47, 50, 51, 53, 55, 57, 58, 59, 62, 63, 65, 66, 68, 71, 72, 73, 77, 88, 92, 94, 96, 107, 108, 109, 110, 111
- New Radio (NR)** Nouvelle forme d'onde pour les réseaux 5G. 15, 19, 20, 29, 55
- next generation Node B (gNB)** Équipement permettant le raccordement des utilisateurs au réseau d'accès radio 5G. Évolution des eNBs dans la 4G. 15, 18, 19, 20, 21, 29, 30, 45, 51, 97

- NFVI Supplier** Fournisseur d'infrastructures virtualisées NFVI. 38
- Peer 2 Peer** Pair à pair. 41
- Protocol Data Unit (PDU)** Unité de données protocolaire.. 17, 42, 81, 96
- Quality of Service (QoS)** Qualité de service. 3, 4, 16, 17, 23, 31, 32, 41, 43, 50, 51, 52, 54, 56, 59, 64, 65, 66, 72, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 97, 100, 101, 105, 106, 107, 108, 109, 110, 111
- Radio Network Function (RNF)** Fonction réseaux de la radio gateway.. 64
- Radio Unit (RU)** Élément de donnée d'une gNB traitant seulement de la couche radio.. 21, 23
- Radio Access Network (RAN)** Réseau d'accès radio. 14, 15, 17, 18, 19, 20, 21, 23, 27, 29, 30, 31, 40, 42, 43, 48, 49, 50, 51, 52, 53, 54, 55, 58, 59, 60, 63, 65, 74, 76, 78, 79, 81, 82, 84, 85, 92, 97, 100, 101, 109, 110
- Remote Radio Head (RRH)** Émetteur-récepteur radio à distance qui se connecte à un panneau de commande radio de l'opérateur réseau via une interface électrique ou sans fil.. 21
- Return Link** Liaison retour.. 60, 69
- Service Network Function (SNF)** Fonction réseaux dans la service gateway.. 64
- Service Function Chaining (SFC)** Méthode de chaînage de fonctions réseaux proposé par l'IETF.. 46
- Technical Report (TR)** Technical Report dans le 3GPP. Suggestion pour la définition du standard 3GPP.. 42
- Technical Specification (TS)** Technical Specification dans le 3GPP. Défini le standard 3GPP. 42
- Transport Network (TN)** Réseau de transport. xii, 3, 27, 30, 31, 40, 45, 54, 55, 75, 76
- Virtual Private Network (VPN)** Réseau privé virtuel.. 64
- Virtual Baseband Unit (vBBU)** Baseband Unit virtualisée. 21
- Virtualization Infrastructure Service Provider (VISP)** Fournisseur d'infrastructures virtualisée. 38
- Virtual Network Function (VNF)** Fonction réseau virtualisée. 11, 12, 13, 21, 40, 53, 68, 93
- Wide Area Network (WAN)** Réseau longue distance. 24

# Acronymes

**3GPP** 3rd Generation Partnership Project. 2, 14, 15, 16, 17, 19, 23, 29, 31, 34, 36, 37, 38, 39, 42, 63, 75, 76, 88, 97, 109

**5QI** 5G QoS Identifier. 43, 78, 80, 81

**AAA** Authentication, Authorization and Accounting. 54

**ACM** Adaptive Coding and Modulation. 97

**ADSL** Asymmetric Digital Subscriber Line. 14, 109

**AF** Application Function. 18, 79, 85

**AMF** Access and Mobility Management Function. 17, 18, 102

**ANR** Automatic Neighbor Relations. 53

**API** Application Programming Interface. 9, 17, 18, 48, 52, 54, 58, 64, 69, 70, 88, 97, 120

**BNG** Broadband Network Gateway. 82, 100, 105

**BSS** Business Support System. 13

**CDN** Content Delivery Network. 9, 18

**COMETS** COntainer MESH for nETwork Slicing. 94, 95, 96, 98, 107, 110, 113, 118

**CoS** Class of Services. 78, 81

**CUPS** Control and User Plane Separation. 16, 17

**DN** Data Network. 18, 48, 96

**DNS** Domain Name System. 13

**DPI** Deep Packet Inspection. 97

**DRB** Data Radio Bearers. 43

**DSCP** Differentiated Services Code Point. 96, 97

**DVB-RCS2** Digital Video Broadcasting - Second Generation Return Channel Satellite. 24, 25

**DVB-S2** Digital Video Broadcasting - Satellite - Second Generation. 24, 25

**eMBB** enhanced Mobile Broadband. 34, 41, 43, 59

**ESIM** Earth Station in Motion. 26

**E-UTRA** Evolved Universal Terrestrial Radio Access. 15

**EM** Element Management. 12, 13

**EPC** Evolved Packet Core. 15

- ETSI** European Telecommunications Standards Institute. 11, 36, 42, 65
- FCAPS** Fault, Configuration, Accounting, Performance and Security. 12
- FDD** Frequency Division Duplex. 19
- ForCES** Forwarding and Control Element Separation. 8
- FSO** Free Space Optical. 24
- FTTx** Fiber to the everything. 14
- GBR** Guaranteed Bit Rate. 43
- GEO** Geostationary Orbit. 59, 61, 77, 80, 100, 101, 103
- GSM** Global System for Mobile Communications. 14
- GST** Generic Network Slice Template. 40
- GW** Gateway. 58, 60, 61
- HAPS** High-altitude Platform Station. 111
- HSS** Home Subscriber Server. 16, 18
- HTB** Hierarchy Token Bucket. 98
- HTTP** Hypertext Transfer Protocol. 17, 96, 97
- IETF** Internet Engineering Task Force. 8, 43, 54
- IoT** Internet of Things. 14, 43, 50, 59
- IP** Internet Protocol. 14, 43, 58
- ITU** International Telecommunication Union. 44
- ITU-R** ITU Radiocommunication Sector. 14
- ITU-T** ITU Telecommunication Standardization Sector. 8, 35, 37, 39
- LEO** Low Earth Orbit. 59, 77, 80, 100
- LINP** Logically Isolated Network Partition. 35, 37, 38, 39, 44
- LXC** Linux Containers. 94
- MANA** Management and Service-aware Networking Architectures. 35
- MANO** Management and Orchestration. 13, 23, 93
- ME-LDPC** Multi-Edge Low-Density Parity-Check. 20
- MEO** Middle Earth Orbit. 77
- MIoT** Massive Internet of Things. 1, 41, 43, 59
- MME** Mobility Management Entity. 15, 16, 18
- MVNO** Mobile Virtual Network Operator. 50
- N3IWF** Non-3GPP Interworking Function. 29



- NEF** Network Exposure Function. 18
- NEST** Network Slice Types. 40
- NETCONF** Network Configuration Protocol. 9
- NFV** Network Function Virtualization. i, iii, 2, 3, 8, 10, 11, 13, 15, 17, 20, 23, 25, 27, 31, 32, 47, 49, 50, 51, 52, 53, 59, 63, 65, 67
- NFVI** Network Function Virtualisation Infrastructure. 12, 13, 61, 62, 64
- NFVO** NFV Orchestrator. 13
- NGMN** Next Generations Mobile Networks. 35, 36
- NLOS** Non-line-of-sight. 20
- NMS** Network Management System. 30, 70
- NRF** NF Repository Function. 18
- NS3** Network Simulator 3. 92, 94, 97
- NSA** Non Standalone. 15, 20
- NSaaS** Network Slice as a Service. 37, 38
- NSMF** Network Slice Management Function. 75
- NSSF** Network Slice Selection Function. 18, 43, 45
- NSSMF** Network Slice Subnet Management Function. 75
- NTN** Non Terrestrial Network. xii, 2, 29, 73, 76, 77, 78, 79, 80, 81, 83, 85, 86, 87, 88, 92, 93, 94, 95, 97, 98, 100, 101, 103, 104, 105, 106, 107, 108, 109, 111
- ODU** Outdoor Unit. 65, 66, 68, 70
- OISL** Optical Inter-Satellite Links. 26
- ONAP** Open Network Automation Platform. 19, 51, 53, 93, 94
- ONF** Open Networking Foundation. 8, 36, 96
- OPEX** Operating Expenses. 48
- OSM** Open Source MANO. 19, 51, 53, 93, 94
- OSS** Operations Support System. 13
- OTT** Over The Top. 67
- PCF** Policy Control Function. 17, 18
- PCRF** Policy and Charging Rules Function. 16, 18
- PDB** Packet Delay Budget. 103, 104, 106
- PDN** Packet Data Network. 15, 16
- PEP** Performance-enhancing Proxies. 25, 26, 63, 64
- PER** Packet Error Rate. 103, 104, 105
- PFCP** Packet Forwarding Control Protocol. 43

- PGW** PDN Gateway. 15, 16, 17
- PGW-U** PDN Gateway User Plane. 16, 18
- PGW-C** PDN Gateway Control Plane. 16, 18
- PLMN** Public Land Mobile Network. 42
- PNF** Physical Network Function. 53, 68
- QFI** QoS Flow ID. xiii, 43, 81, 86, 101, 103, 104, 105, 106, 107
- QOF** Quality of Service Function. 86, 87, 88, 96, 97, 101, 105
- REST** Representational State Transfer. 17
- RIC** RAN Intelligent Controller. 23
- SFQ** Stochastic Fairness Queueing. 98
- S-NS3** Satellite NS3. 97
- SCPC** Single Channel Per Carrier. 77
- SD** Slice Differentiator. 42
- SNC** Satellite Network Control. 70
- SNO** Satellite Network Operator. 65, 68
- S-NSSAI** Single - Network Slice Selection Assistance Information. 42, 43, 86
- SA** Standalone. 15, 20
- SBA** Service Based Architecture. 15, 17, 42, 51, 65, 88, 108
- SBI** Service Based Interface. 17, 18
- SBMA** Service Based Management Architecture. 15, 41, 85
- SDN** Software Defined Network. i, iii, 2, 3, 8, 9, 10, 11, 13, 15, 16, 19, 20, 23, 25, 27, 31, 32, 36, 42, 47, 49, 50, 51, 52, 54, 59, 63, 67, 71, 85, 87, 93
- SGW** Serving Gateway. 15, 16, 17
- SGW-U** Serving Gateway User Plane. 16, 18
- SGW-C** Serving Gateway Control Plane. 16, 18
- SLA** Service Level Agreement. 40, 48, 49, 54, 87, 106
- SLS** Service Level Specification. 40
- SMF** Session Management Function. 17, 18, 43, 86, 88, 96, 102
- SON** Self-Organizing Network. 53
- SST** Slice/Service Type. 42, 43
- ST** Satellite Terminal. 58, 60, 61, 65, 68, 77
- SVNO** Satellite Virtual Network Operator. 68
- TDD** Time Division Duplex. 19

- UDM** Unified Data Management. 17, 18
- UDR** Unified Data Repository. 18
- UE** User Equipment. xiv, 18, 20, 27, 29, 30, 42, 43, 45, 80, 81, 85, 86, 95, 96, 97, 100, 101, 102, 103, 104, 106
- UPF** User plane Function. 18, 43, 58, 60, 65, 96, 102
- URLLC** Ultra-Reliable Low-Latency Communication. 41, 43
- V2X** Vehicle-to-Everything. 1, 41, 43
- VBR** Variable Bitrate. 103
- VHTS** Very High Throughput Satellite. iii, 24, 97
- VIM** Virtualized Infrastructure Manager. 13
- VM** Virtual Machine. 65, 93, 94, 101
- VNFM** VNF Manager. 13
- VoIP** Voice over IP. 34
- VSAT** Very Small Aperture Terminal. 26

# Bibliographie

- [1] Mary MEEKER. « Internet Trends 2019 ». Bond, 2019, p. 334 (cf. p. 1).
- [2] CISCO SYSTEMS. « Cisco Annual Internet Report (2018-2023) ». White Paper. Cisco, 2020, p. 35. URL : <https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.pdf> (visité le 28/05/2022) (cf. p. 1).
- [3] 3GPP. « System architecture for the 5G System (5GS) ». Technical Specification (TS) 23.501. Version 17.3.0. 3rd Generation Partnership Project (3GPP), déc. 2021. URL : <http://www.3gpp.org/DynaReport/23501.htm> (cf. p. 2, 17, 29, 31, 34, 40, 42, 45).
- [4] Akram HAKIRI et Pascal BERTHOU. « Leveraging SDN for The 5G Networks: Trends, Prospects and Challenges ». In : (2015), p. 23 (cf. p. 8).
- [5] L. YANG et al. « Forwarding and Control Element Separation (ForCES) Framework ». RFC 3746. Internet Engineering Task Force, avr. 2004, p. 1-40. DOI : {10.17487/RFC3746}. URL : <http://www.rfc-editor.org/rfc/rfc3746.txt> (cf. p. 8).
- [6] ITU-T. « Framework of software-defined networking ». Recommendation Y.3300. International Telecommunication Union, juin 2014 (cf. p. 8).
- [7] ONF. « SDN Architecture ». Rapp. tech. ONF TR-502. Open Networking Foundation, juin 2014. URL : [https://opennetworking.org/wp-content/uploads/2013/02/TR\\_SDN\\_ARCH\\_1.0\\_06062014.pdf](https://opennetworking.org/wp-content/uploads/2013/02/TR_SDN_ARCH_1.0_06062014.pdf) (visité le 18/02/2022) (cf. p. 8).
- [8] ONF. « OpenFlow Switch Specification ». Rapp. tech. ONF TS-001. Version 1.0.0 (Wire Protocol 0x01). Open Networking Foundation, déc. 2009. URL : <https://opennetworking.org/wp-content/uploads/2013/04/openflow-spec-v1.0.0.pdf> (visité le 18/02/2022) (cf. p. 9).
- [9] R. ENNS et al. « Network Configuration Protocol (NETCONF) ». RFC 6241. Internet Engineering Task Force, juin 2011, p. 1-113. DOI : {10.17487/RFC6241}. URL : <http://www.rfc-editor.org/rfc/rfc6241.txt> (cf. p. 9).
- [10] Faucet SDN. « Ryu SDN framework ». GitHub. 2021. URL : <https://github.com/faucetsdn/ryu> (visité le 28/04/2022) (cf. p. 9).
- [11] The Linux FOUNDATION. « OpenDaylight ». OpenDaylight. 2022. URL : <https://www.opendaylight.org/> (visité le 28/04/2022) (cf. p. 9).

- [12] Antoine FEGHALI, Rima KILANY et Maroun CHAMOUN. « SDN security problems and solutions analysis ». In : *2015 International Conference on Protocol Engineering (ICPE) and International Conference on New Technologies of Distributed Systems (NTDS)*. 2015 International Conference on Protocol Engineering (ICPE) and International Conference on New Technologies of Distributed Systems (NTDS). Paris, France : IEEE, juill. 2015, p. 1-5. ISBN : 978-1-4673-9265-5. DOI : 10.1109/NOTERE.2015.7293514. URL : <http://ieeexplore.ieee.org/document/7293514/> (visité le 27/04/2022) (cf. p. 9).
- [13] Diego KREUTZ et al. « Software-Defined Networking: A Comprehensive Survey ». In : *Proceedings of the IEEE* 103.1 (2015), p. 14-76. DOI : 10.1109/JPROC.2014.2371999 (cf. p. 9).
- [14] Sandra SCOTT-HAYWARD, Gemma O'CALLAGHAN et Sakir SEZER. « Sdn Security: A Survey ». en. In : *2013 IEEE SDN for Future Networks and Services (SDN4FNS)*. Trento, Italy : IEEE, nov. 2013, p. 1-7. ISBN : 978-1-4799-2781-4. DOI : 10.1109/SDN4FNS.2013.6702553. URL : <http://ieeexplore.ieee.org/document/6702553/> (visité le 18/02/2022) (cf. p. 9).
- [15] Celio TROIS et al. « A Survey on SDN Programming Languages: Toward a Taxonomy ». en. In : *IEEE Communications Surveys & Tutorials* 18.4 (2016), p. 2687-2712. ISSN : 1553-877X. DOI : 10.1109/COMST.2016.2553778. URL : <http://ieeexplore.ieee.org/document/7452335/> (visité le 18/02/2022) (cf. p. 9).
- [16] Juan Camilo Correa CHICA, Jenny Cuatindioy IMBACHI et Juan Felipe Botero VEGA. « Security in SDN: A comprehensive survey ». In : *Journal of Network and Computer Applications* 159 (2020), p. 102595. ISSN : 1084-8045. DOI : <https://doi.org/10.1016/j.jnca.2020.102595>. URL : <https://www.sciencedirect.com/science/article/pii/S1084804520300692> (cf. p. 9).
- [17] Matthias WICHTLHUBER, Robert REINECKE et David HAUSHEER. « An SDN-Based CDN/ISP Collaboration Architecture for Managing High-Volume Flows ». In : *IEEE Transactions on Network and Service Management* 12.1 (2015), p. 48-60. DOI : 10.1109/TNSM.2015.2404792 (cf. p. 9).
- [18] Michael JARSCHER et al. « SDN-Based Application-Aware Networking on the Example of YouTube Video Streaming ». In : *2013 Second European Workshop on Software Defined Networks*. 2013, p. 87-92. DOI : 10.1109/EWSDN.2013.21 (cf. p. 9).
- [19] Suhail AHMAD et Ajaz Hussain MIR. « Scalability, Consistency, Reliability and Security in SDN Controllers: A Survey of Diverse SDN Controllers ». In : *Journal of Network and Systems Management* 29.1 (nov. 2020), p. 9. ISSN : 1573-7705. DOI : 10.1007/s10922-020-09575-4. URL : <https://doi.org/10.1007/s10922-020-09575-4> (cf. p. 9).

- [20] Sushant JAIN et al. « B4: Experience with a Globally Deployed Software Defined WAN ». In : *Proceedings of the ACM SIGCOMM Conference*. Hong Kong, China, 2013. URL : <http://cseweb.ucsd.edu/~vahdat/papers/b4-sigcomm13.pdf> (cf. p. 9).
- [21] ETSI. « Use Cases ». Group Report GR NFV 001. Serie Network Functions Virtualisation (NFV); V1.3.1. European Telecommunications Standards Institute, mars 2021. URL : [https://www.etsi.org/deliver/etsi\\_gr/NFV/001\\_099/001/01.03.01\\_60/gr\\_NFV001v010301p.pdf](https://www.etsi.org/deliver/etsi_gr/NFV/001_099/001/01.03.01_60/gr_NFV001v010301p.pdf) (visité le 03/03/2022) (cf. p. 11).
- [22] ETSI. « Network Functions Virtualisation (NFV) Release 2; Management and Orchestration; Architectural Framework Specification ». Rapp. tech. ETSI GS NFV 006 V2.1. v 2.1.1. European Telecommunications Standards Institute, jan. 2021. URL : [https://www.etsi.org/deliver/etsi\\_gs/NFV/001\\_099/006/02.01.01\\_60/gv006v020101p.pdf](https://www.etsi.org/deliver/etsi_gs/NFV/001_099/006/02.01.01_60/gv006v020101p.pdf) (visité le 21/02/2022) (cf. p. 11).
- [23] ETSI. « Architectural Framework ». Group Specification GS NFV 002. Serie Network Functions Virtualisation (NFV); V1.2.1. European Telecommunications Standards Institute, déc. 2014. URL : [https://docbox.etsi.org/ISG/NFV/Open/Publications\\_pdf/Specs-Reports/NFV%20002v1.2.1%20-%20GS%20-%20NFV%20Architectural%20Framework.pdf](https://docbox.etsi.org/ISG/NFV/Open/Publications_pdf/Specs-Reports/NFV%20002v1.2.1%20-%20GS%20-%20NFV%20Architectural%20Framework.pdf) (visité le 03/03/2022) (cf. p. 11).
- [24] ETSI. « Management and Orchestration ». Group Specification GS NFV-MAN 001. Serie Network Functions Virtualisation (NFV); V1.1.1. European Telecommunications Standards Institute, déc. 2014. URL : [https://www.etsi.org/deliver/etsi\\_gs/nfv-man/001\\_099/001/01.01.01\\_60/gv001v010101p.pdf](https://www.etsi.org/deliver/etsi_gs/nfv-man/001_099/001/01.01.01_60/gv001v010101p.pdf) (visité le 03/03/2022) (cf. p. 13).
- [25] ETSI. « Report on architecture options to support multiple administrative domains ». Group Report GR NFV-IFA 028. Serie Network Functions Virtualisation (NFV) Release 3; Management and Orchestration; V3.1.1. European Telecommunications Standards Institute, jan. 2018. URL : [https://www.etsi.org/deliver/etsi\\_gr/NFV-IFA/001\\_099/028/03.01.01\\_60/gr\\_NFV-IFA028v030101p.pdf](https://www.etsi.org/deliver/etsi_gr/NFV-IFA/001_099/028/03.01.01_60/gr_NFV-IFA028v030101p.pdf) (visité le 03/03/2022) (cf. p. 13).
- [26] ETSI. « Report on SDN Usage in NFV Architectural Framework ». Group Specification GS NFV-EVE 005. Serie Network Functions Virtualisation (NFV); Ecosystem; V1.1.1. European Telecommunications Standards Institute, déc. 2015. URL : [https://www.etsi.org/deliver/etsi\\_gs/NFV-EVE/001\\_099/005/01.01.01\\_60/gv005v010101p.pdf](https://www.etsi.org/deliver/etsi_gs/NFV-EVE/001_099/005/01.01.01_60/gv005v010101p.pdf) (visité le 03/03/2022) (cf. p. 13).
- [27] ETSI. « Report on Network Slicing Support with ETSI NFV Architecture Framework ». Group Report GR NFV-EVE 012. Serie Network Functions Virtualisation (NFV) Release 3; Evolution and Ecosystem; V3.1.1. Euro-

- pean Telecommunications Standards Institute, déc. 2017 (cf. p. 13, 36, 42, 44).
- [28] Bolt BERANEK et Newman Inc for DARPA. « Interface Message Processor ». Report 1822. Defense Advanced Research Projects Agency, 1969 (cf. p. 14).
- [29] CCITT. « CCITT Recommendation X.25 Orange Book ». Specification. Draft. Comité Consultatif International Téléphonique et Télégraphique, 1976 (cf. p. 14).
- [30] A.K. BHUSHAN et al. « Standardizing Network Mail Headers ». RFC 0561. Internet Engineering Task Force, sept. 1973. DOI : { 10 . 17487 / RFC0561}. URL : <http://www.rfc-editor.org/rfc/rfc561.txt> (cf. p. 14).
- [31] A.K. BHUSHAN. « File Transfer Protocol ». RFC 0114. Internet Engineering Task Force, avr. 1971, p. 1-17. DOI : { 10 . 17487 / RFC0114}. URL : <http://www.rfc-editor.org/rfc/rfc114.txt> (cf. p. 14).
- [32] J. POSTEL. « DoD standard Internet Protocol ». RFC 0760. Internet Engineering Task Force, jan. 1980, p. 1-42. DOI : { 10 . 17487 / RFC0760}. URL : <http://www.rfc-editor.org/rfc/rfc760.txt> (cf. p. 14).
- [33] ETSI. « European Digital Cellular Telecommunications System (Phase 1); MS-BSS Layer 1 - General Requirements ». Technical Report (TR) GSM 04.04. Version 3.3.4. European Telecommunications Standards Institute, fév. 1992 (cf. p. 14).
- [34] Richard D. GITLIN et al. « Method and apparatus for wideband transmission of digital signals between, for example, a telephone central office and customer premises ». US4924492A (US). Mai 1990. URL : <https://patents.google.com/patent/US4924492A/en> (visité le 17/02/2022) (cf. p. 14).
- [35] Imrich CHLAMTAC, Ashwin GUMASTE et Csaba SZABO. « Broadband Services: Business Models and Technologies for Community Networks ». en. John Wiley & Sons, juin 2005. ISBN : 978-0-470-02249-8 (cf. p. 14).
- [36] ITU-R. « IMT Vision – Framework and overall objectives of the future development of IMT for 2020 and beyond ». Report ITU-R M.2083-0. International Telecommunication Union, sept. 2015 (cf. p. 14).
- [37] ITU-R. « Future technology trends of terrestrial IMT systems ». Report ITU-R M.2320-0. International Telecommunication Union, nov. 2014 (cf. p. 14).
- [38] ITU-R. « IMT traffic estimates for the years 2020 to 2030 ». Rapp. tech. ITU-R M.2370-0. International Telecommunication Union, juill. 2015 (cf. p. 14).
- [39] ITU-R. « Technical feasibility of IMT in bands above 6 GHz ». Rapp. tech. ITU-R M.2376-0. International Telecommunication Union, juill. 2015 (cf. p. 14).

- [40] 3GPP. « Management and orchestration; Concepts, use cases and requirements ». Technical Specification (TS) 28.530. Version 17.2.0. 3rd Generation Partnership Project (3GPP), déc. 2021. URL : <http://www.3gpp.org/DynaReport/28530.htm> (cf. p. 15, 37, 38, 41, 45).
- [41] 3GPP. « General Packet Radio Service (GPRS) enhancements for Evolved Universal Terrestrial Radio Access Network (E-UTRAN) access ». Technical Specification (TS) 23.401. Version 17.3.0. 3rd Generation Partnership Project (3GPP), déc. 2021. URL : <http://www.3gpp.org/DynaReport/23401.htm> (cf. p. 15).
- [42] 3GPP. « Architecture enhancements for control and user plane separation of EPC nodes ». Technical Specification (TS) 23.214. Version 17.0.0. 3rd Generation Partnership Project (3GPP), juin 2021. URL : <http://www.3gpp.org/DynaReport/23214.htm> (cf. p. 16).
- [43] Girma M. YILMA et al. « Benchmarking open source NFV MANO systems: OSM and ONAP ». In : *Computer Communications* 161 (sept. 2020), p. 86-98. ISSN : 01403664. DOI : 10.1016/j.comcom.2020.07.013. URL : <https://linkinghub.elsevier.com/retrieve/pii/S0140366420305946> (visité le 22/03/2022) (cf. p. 19).
- [44] 3GPP. « NR; NR and NG-RAN Overall description; Stage-2 ». Technical Specification (TS) 38.300. Version 16.8.0. 3rd Generation Partnership Project (3GPP), déc. 2021. URL : <http://www.3gpp.org/DynaReport/38300.htm> (cf. p. 20, 42, 45).
- [45] Theodore S. RAPPAPORT et al. « Millimeter Wave Mobile Communications for 5G Cellular: It Will Work! ». In : *IEEE Access* 1 (2013). Conference Name: IEEE Access, p. 335-349. ISSN : 2169-3536. DOI : 10.1109/ACCESS.2013.2260813 (cf. p. 19).
- [46] Tom RICHARDSON. « Multi-Edge Type LDPC Codes ». In : (2004), p. 37 (cf. p. 20).
- [47] Lu LU et al. « An Overview of Massive MIMO: Benefits and Challenges ». In : *IEEE Journal of Selected Topics in Signal Processing* 8.5 (oct. 2014), p. 742-758. ISSN : 1932-4553, 1941-0484. DOI : 10.1109/JSTSP.2014.2317671. URL : <http://ieeexplore.ieee.org/document/6798744/> (visité le 25/03/2022) (cf. p. 20).
- [48] Wonil ROH et al. « Millimeter-wave beamforming as an enabling technology for 5G cellular communications: theoretical feasibility and prototype results ». In : *IEEE Communications Magazine* 52.2 (fév. 2014), p. 106-113. ISSN : 0163-6804. DOI : 10.1109/MCOM.2014.6736750. URL : <http://ieeexplore.ieee.org/document/6736750/> (visité le 25/03/2022) (cf. p. 20).



- [49] Suhanya JAYAPRAKASAM et al. « Robust Beam-Tracking for mmWave Mobile Communications ». In : *IEEE Communications Letters* 21.12 (déc. 2017), p. 2654-2657. ISSN : 1089-7798. DOI : 10.1109/LCOMM.2017.2748938. URL : <http://ieeexplore.ieee.org/document/8025577/> (visité le 25/03/2022) (cf. p. 20).
- [50] Aleksandra CHECKO et al. « Cloud RAN for Mobile Networks—A Technology Overview ». In : *IEEE Communications Surveys & Tutorials* 17.1 (2015), p. 405-426. ISSN : 1553-877X. DOI : 10.1109/COMST.2014.2355255. URL : <https://ieeexplore.ieee.org/document/6897914> (visité le 19/02/2022) (cf. p. 20, 51).
- [51] 3GPP. « NG-RAN; F1 general aspects and principles ». Technical Specification (TS) 38.470. Version 16.5.0. 3rd Generation Partnership Project (3GPP), juill. 2021. URL : <http://www.3gpp.org/DynaReport/38470.htm> (cf. p. 21).
- [52] 3GPP. « NG-RAN; NR user plane protocol ». Technical Specification (TS) 38.425. Version 16.3.0. 3rd Generation Partnership Project (3GPP), avr. 2021. URL : <http://www.3gpp.org/DynaReport/38425.htm> (cf. p. 21).
- [53] 3GPP. « NG-RAN; F1 Application Protocol (F1AP) ». Technical Specification (TS) 38.473. Version 16.8.0. 3rd Generation Partnership Project (3GPP), déc. 2021. URL : <http://www.3gpp.org/DynaReport/38473.htm> (cf. p. 21).
- [54] Liljana GAVRILOVSKA, Valentin RAKOVIC et Daniel DENKOVSKI. « From Cloud RAN to Open RAN ». In : *Wireless Personal Communications* 113.3 (1<sup>er</sup> août 2020), p. 1523-1539. ISSN : 1572-834X. DOI : 10.1007/s11277-020-07231-3. URL : <https://doi.org/10.1007/s11277-020-07231-3> (cf. p. 23).
- [55] O-RAN ALLIANCE. « O-RAN Minimum Viable Plan and Acceleration towards Commercialization ». Rapp. tech. O-RAN Alliance, juin 2021. URL : <https://static1.squarespace.com/static/5ad774cce74940d7115044b0/t/60f9b144abdc902712f43475/1626976585796/O-RAN+Minimum+Viable+Plan+and+Acceleration+towards+Commercialization+White+Paper+29+June+2021.pdf> (visité le 21/02/2022) (cf. p. 23).
- [56] O. VIDAL et al. « Next generation High Throughput Satellite system ». In : *2012 IEEE First AESS European Conference on Satellite Telecommunications (ESTEL)*. 2012 IEEE First AESS European Conference on Satellite Telecommunications (ESTEL). Oct. 2012, p. 1-7. DOI : 10.1109/ESTEL.2012.6400146 (cf. p. 24).
- [57] Domenico MIGNOLO et al. « Approaching Terabit/s satellite ». In : *Navigation and Earth Observation Conference*. 2011, p. 11 (cf. p. 24).

- [58] P. INIGO et al. « Review of terabit/s satellite, the next generation of HTS systems ». In : **2014 7th Advanced Satellite Multimedia Systems Conference and the 13th Signal Processing for Space Communications Workshop (ASMS/SPSC)**. 2014 7th Advanced Satellite Multimedia Systems Conference and the 13th Signal Processing for Space Communications Workshop (ASMS/SPSC). ISSN: 2326-5949. Sept. 2014, p. 318-322. DOI : 10 . 1109 / ASMS-SPSC . 2014 . 6934562 (cf. p. 24).
- [59] A. LE PERA et al. « Digital Transparent Processor for Satellite Telecommunication Services ». In : **2007 IEEE Aerospace Conference**. 2007 IEEE Aerospace Conference. ISSN: 1095-323X. Mars 2007, p. 1-9. DOI : 10 . 1109 / AERO . 2007 . 352932 (cf. p. 24).
- [60] Baptiste PALACIN et al. « Multibeam antennas for very high throughput satellites in Europe: Technologies and trends ». In : **2017 11th European Conference on Antennas and Propagation (EUCAP)**. 2017 11th European Conference on Antennas and Propagation (EUCAP). Mars 2017, p. 2413-2417. DOI : 10 . 23919 / EuCAP . 2017 . 7928493 (cf. p. 24).
- [61] Sudhakar K. RAO. « Advanced Antenna Technologies for Satellite Communications Payloads ». In : IEEE Transactions on Antennas and Propagation 63.4 (avr. 2015). Conference Name: IEEE Transactions on Antennas and Propagation, p. 1205-1217. ISSN : 1558-2221. DOI : 10 . 1109 / TAP . 2015 . 2391283 (cf. p. 24).
- [62] J. ANZALCHI et al. « Beam hopping in multi-beam broadband satellite systems: System simulation and performance comparison with non-hopped systems ». In : **2010 5th Advanced Satellite Multimedia Systems Conference and the 11th Signal Processing for Space Communications Workshop**. 2010 5th Advanced Satellite Multimedia Systems Conference and the 11th Signal Processing for Space Communications Workshop. ISSN: 2326-5949. Sept. 2010, p. 248-255. DOI : 10 . 1109 / ASMS - SPSC . 2010 . 5586860 (cf. p. 24).
- [63] Karim ELAYOUBI, Angélique RISSONS et Aniceto BELMONTE. « Optical test bench experiments for 1-Tb/s satellite feeder uplinks ». In : **Laser Communication and Propagation through the Atmosphere and Oceans VII**. T. 10770. International Society for Optics et Photonics. 2018, p. 1077006 (cf. p. 25).
- [64] K. ELAYOUBI et al. « RZ-DPSK optical modulation for free space optical communication by satellites ». In : **2017 Opto-Electronics and Communications Conference (OECC) and Photonics Global Conference (PGC)**. 2017 Opto-Electronics and Communications Conference (OECC) and Photonics Global Conference (PGC). ISSN: 2166-8892. Juill. 2017, p. 1-2. DOI : 10 . 1109 / OECC . 2017 . 8115015 (cf. p. 25).

- [65] E. DUBOIS et al. « Enhancing TCP based communications in mobile satellite scenarios: TCP PEPs issues and solutions ». In : *2010 5th Advanced Satellite Multimedia Systems Conference and the 11th Signal Processing for Space Communications Workshop*. 2010 5th Advanced Satellite Multimedia Systems Conference and the 11th Signal Processing for Space Communications Workshop. ISSN: 2326-5949. Sept. 2010, p. 476-483. DOI : 10 . 1109/ASMS-SPSC.2010.5586878 (cf. p. 25).
- [66] Renaud SALLANTIN et al. « An end-to-end alternative to TCP PEPs: Initial Spreading, a TCP fast start-up mechanism ». In : *International Journal of Satellite Communications and Networking* 34.1 (2016). \_eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1002/sat.1097>, p. 75-91. ISSN : 1542-0981. DOI : 10 . 1002 / sat . 1097. URL : <https://onlinelibrary.wiley.com/doi/abs/10.1002/sat.1097> (visité le 26/04/2022) (cf. p. 25).
- [67] Jana IYENGAR et Martin THOMSON. « QUIC: A UDP-Based Multiplexed and Secure Transport ». Rapp. tech. 9000. Mai 2021. 151 p. DOI : 10 . 17487 / RFC9000. URL : <https://www.rfc-editor.org/info/rfc9000> (cf. p. 25).
- [68] JC PAVUR et al. « QPEP: An actionable approach to secure and performant broadband from geostationary orbit ». In : Internet Society, 2021 (cf. p. 25).
- [69] VITAL CONSORTIUM. « H2020 VITAL project ». 2015. URL : <http://www.ict-vital.eu/> (visité le 02/10/2019) (cf. p. 25).
- [70] R. FERRÚS et al. « SDN/NFV-enabled satellite communications networks: Opportunities, scenarios and challenges ». In : *Physical Communication* 18 (mars 2016), p. 95-112. ISSN : 18744907. DOI : 10 . 1016 / j . phycom . 2015 . 10 . 007. URL : <https://linkinghub.elsevier.com/retrieve/pii/S1874490715000543> (visité le 13/05/2019) (cf. p. 25).
- [71] R. FERRUS et al. « Towards SDN/NFV-enabled satellite ground segment systems: End-to-End Traffic Engineering use case ». In : *2017 IEEE International Conference on Communications Workshops (ICC Workshops)*. 2017 IEEE International Conference on Communications Workshops (ICC Workshops). Paris, France : IEEE, mai 2017, p. 888-893. ISBN : 978-1-5090-1525-2. DOI : 10 . 1109 / ICCW . 2017 . 7962771. URL : <http://ieeexplore.ieee.org/document/7962771/> (visité le 13/05/2019) (cf. p. 25).
- [72] T. AHMED et al. « Satellite Gateway Diversity in SDN/NFV-enabled satellite ground segment systems ». In : *2017 IEEE International Conference on Communications Workshops (ICC Workshops)*. 2017 IEEE International Conference on Communications Workshops (ICC Workshops). Paris, France : IEEE, mai 2017, p. 882-887. ISBN : 978-1-5090-1525-2. DOI : 10 . 1109 / ICCW . 2017 . 7962770. URL : <http://ieeexplore.ieee.org/document/7962770/> (visité le 13/05/2019) (cf. p. 25).

- [73] Lionel BERTAUX et al. « Software defined networking and virtualization for broadband satellite networks ». In : IEEE Communications Magazine 53.3 (mars 2015). Number: 3, p. 54-60. ISSN : 0163-6804. DOI : 10.1109/MCOM.2015.7060482. URL : <http://ieeexplore.ieee.org/document/7060482/> (visité le 13/05/2019) (cf. p. 25).
- [74] Toufik AHMED et al. « Software-defined satellite cloud RAN ». In : International Journal of Satellite Communications and Networking 36.1 (jan. 2018). Number: 1, p. 108-133. ISSN : 15420973. DOI : 10.1002/sat.1206. URL : <http://doi.wiley.com/10.1002/sat.1206> (visité le 13/05/2019) (cf. p. 25, 61).
- [75] Amazon Web SERVICES. « AWS Ground Station ». Amazon Web Services, Inc. 2022. URL : <https://aws.amazon.com/fr/ground-station/> (visité le 28/04/2022) (cf. p. 26).
- [76] MICROSOFT. « Azure Orbital | Microsoft Azure ». 2022. URL : <https://azure.microsoft.com/fr-fr/services/orbital/> (visité le 28/04/2022) (cf. p. 26).
- [77] AST. « SpaceMobile | AST & Science ». URL : <https://ast-science.com/spacemobile/> (visité le 22/03/2022) (cf. p. 26).
- [78] Jonathan C. McDOWELL. « The Low Earth Orbit Satellite Population and Impacts of the SpaceX Starlink Constellation ». In : The Astrophysical Journal 892.2 (6 avr. 2020), p. L36. ISSN : 2041-8213. DOI : 10.3847/2041-8213/ab8016. URL : <https://iopscience.iop.org/article/10.3847/2041-8213/ab8016> (visité le 22/03/2022) (cf. p. 26).
- [79] TELESAT. « Telesat Lightspeed LEO Network ». 20 mai 2020. URL : <https://www.telesat.com/leo-satellites/> (visité le 22/03/2022) (cf. p. 26).
- [80] Xingqin LIN et al. « On the Path to 6G: Embracing the Next Wave of Low Earth Orbit Satellite Access ». In : IEEE Communications Magazine 59.12 (déc. 2021). Number: 12, p. 36-42. ISSN : 0163-6804, 1558-1896. DOI : 10.1109/MCOM.001.2100298. URL : <https://ieeexplore.ieee.org/document/9681631/> (visité le 31/01/2022) (cf. p. 26).
- [81] CARLOS CARRIZO et al. « Optical inter-satellite link terminals for next generation satellite constellations ». In : Proc.SPIE. T. 11272. 2 mars 2020. DOI : 10.1117/12.2545629. URL : <https://doi.org/10.1117/12.2545629> (cf. p. 26).
- [82] Timo WEKERLE et al. « Status and Trends of Smallsats and their Launch Vehicles — An Up-to-date Review ». In : Journal of Aerospace Technology and Management 9.3 (3 août 2017), p. 269-286. ISSN : 2175-9146. DOI : 10.5028/jatm.v9i3.853. URL : <http://www.jatm.com.br/ojs/index.php/jatm/article/view/853> (visité le 22/03/2022) (cf. p. 27).

- [83] Boris Tiomela Jou et al. « Architecture Options for Satellite Integration into 5G Networks ». In : **2018 European Conference on Networks and Communications (EuCNC)**. 2018 European Conference on Networks and Communications (EuCNC). ISSN: 2575-4912. Juin 2018, p. 398-9. DOI : 10.1109/EuCNC.2018.8442436 (cf. p. 27).
- [84] Boris TIOMELA JOU et al. « Integrated SaT5G General Network Architecture ». D3.1. Issue: D3.1. SaT5G, 15 oct. 2019. URL : [https://www.sat5g-project.eu/wp-content/uploads/2019/04/SaT5G-D3.1-Integrated-SaT5G-general-network-architecture\\_ADS\\_v1.00\\_Inter...pdf](https://www.sat5g-project.eu/wp-content/uploads/2019/04/SaT5G-D3.1-Integrated-SaT5G-general-network-architecture_ADS_v1.00_Inter...pdf) (visité le 15/10/2019) (cf. p. 27, 28, 30).
- [85] 3GPP. « Study on New Radio (NR) to support non-terrestrial networks ». Technical Report (TR) 38.811. Version 15.4.0. 3rd Generation Partnership Project (3GPP), oct. 2020. URL : <http://www.3gpp.org/DynaReport/38811.htm> (cf. p. 29, 42).
- [86] 3GPP. « Study on using satellite access in 5G ». Technical Report (TR) 22.822. Version 16.0.0. 3rd Generation Partnership Project (3GPP), juill. 2018. URL : <http://www.3gpp.org/DynaReport/22822.htm> (cf. p. 30).
- [87] Boris TIOMELA JOU et al. « Integrated SaT5G Detailed Network Architecture ». D3.2. SaT5G, 2019. URL : [https://www.sat5g-project.eu/wp-content/uploads/2019/04/SaT5G\\_D3.2\\_ADS\\_v2.0\\_D.pdf](https://www.sat5g-project.eu/wp-content/uploads/2019/04/SaT5G_D3.2_ADS_v2.0_D.pdf) (visité le 28/04/2022) (cf. p. 30).
- [88] 3GPP. « Management and orchestration; 5G end to end Key Performance Indicators (KPI) ». Technical Specification (TS) 28.554. Version 17.5.0. 3rd Generation Partnership Project (3GPP), déc. 2021. URL : <http://www.3gpp.org/DynaReport/28554.htm> (cf. p. 34, 40).
- [89] Alex GALIS et al. « A Flexible IP Active Networks Architecture ». In : **Active Networks**. Sous la dir. de Gerhard Goos et al. T. 1942. Berlin, Heidelberg : Springer Berlin Heidelberg, 2000, p. 1-15. ISBN : 978-3-540-40057-8. DOI : 10.1007/3-540-40057-5\_1. URL : [http://link.springer.com/10.1007/3-540-40057-5\\_1](http://link.springer.com/10.1007/3-540-40057-5_1) (visité le 18/02/2022) (cf. p. 35).
- [90] Alex GALIS et al. « Programmable Networks for IP Service Deployment ». en. Artech House, 2004. ISBN : 978-1-58053-746-9 (cf. p. 35).
- [91] Larry PETERSON et al. « GENI Design Principles ». en. In : **Computer** 39.9 (sept. 2006), p. 102-105. ISSN : 0018-9162. DOI : 10.1109/MC.2006.307. URL : <http://ieeexplore.ieee.org/document/1703325/> (visité le 20/02/2022) (cf. p. 35).
- [92] Alex GALIS et al. « Management and service-aware networking architectures (MANA) for future Internet; Position paper: System functions, capabilities and requirements ». In : **2009 Fourth International Conference on Communications and Networking in China**. Xian, China : IEEE, août 2009, p. 1-13. ISBN : 978-1-4244-4337-6. DOI : 10.1109/CHINACOM.2009.

5339964. URL : <http://ieeexplore.ieee.org/document/5339964/> (visité le 20/02/2022) (cf. p. 35).
- [93] ITU-T. « Future networks: Objectives and design goals ». Recommendation Y.3001. International Telecommunication Union, mai 2011, p. 26 (cf. p. 35, 44).
- [94] ITU-T. « Framework of network virtualization for future networks ». Recommendation Y.3011. International Telecommunication Union, jan. 2012 (cf. p. 35, 44).
- [95] NGMN. « 5G White Paper ». Technical Report. Version 1.0.8. Next Generations Mobile Networks, fév. 2015 (cf. p. 35, 46).
- [96] NGMN. « Description of Network Slicing Concept ». Technical Report. Version 1.0.8. Next Generations Mobile Networks, sept. 2016 (cf. p. 35, 46).
- [97] 3GPP. « Study on Architecture for Next Generation System ». Technical Report (TR) 23.799. Version 14.0.0. 3rd Generation Partnership Project (3GPP), déc. 2016. URL : <http://www.3gpp.org/DynaReport/23799.htm> (cf. p. 36).
- [98] ONF. « Applying SDN Architecture to 5G Slicing ». Rapp. tech. TR-526. Issue 1. Open Networking Foundation, avr. 2016. URL : [https://opennetworking.org/wp-content/uploads/2014/10/Applying\\_SDN\\_Architecture\\_to\\_5G\\_Slicing\\_TR-526.pdf](https://opennetworking.org/wp-content/uploads/2014/10/Applying_SDN_Architecture_to_5G_Slicing_TR-526.pdf) (visité le 20/02/2022) (cf. p. 36, 46).
- [99] ITU-T. « Functional architecture of network virtualization for future networks ». Recommendation Y.3015. International Telecommunication Union, avr. 2016 (cf. p. 37, 39, 44).
- [100] ITU-T. « Requirements of network virtualization for future networks ». Recommendation Y.3012. International Telecommunication Union, avr. 2014 (cf. p. 39, 44).
- [101] ITU-T. « Information technology – Cloud computing – Reference architecture ». Recommendation Y.3502. International Telecommunication Union, août 2014 (cf. p. 39, 44).
- [102] ITU-T. « Framework of inter-cloud computing ». Recommendation Y.3511. International Telecommunication Union, mars 2014 (cf. p. 39, 44).
- [103] 3GPP. « Management and orchestration; 5G Network Resource Model (NRM); Stage 2 and stage 3 ». Technical Specification (TS) 28.541. Version 17.5.0. 3rd Generation Partnership Project (3GPP), déc. 2021. URL : <http://www.3gpp.org/DynaReport/28541.htm> (cf. p. 39, 40, 45, 75).
- [104] GSMA. « Generic Network Slice Template ». Rapp. tech. NG 116. Version 6.0. GSM Association, nov. 2021. URL : <https://www.gsma.com/newsroom/wp-content/uploads/NG.116-v6.0-1.pdf> (visité le 20/02/2022) (cf. p. 40, 44, 69).

- [105] 3GPP. « Vehicle-to-Everything (V2X) services in 5G System (5GS); Stage 3 ». Technical Specification (TS) 24.587. Version 17.4.1. 3rd Generation Partnership Project (3GPP), jan. 2022. URL : <http://www.3gpp.org/DynaReport/24587.htm> (cf. p. 41).
- [106] 3GPP. « Management and orchestration; Provisioning ». Technical Specification (TS) 28.531. Version 17.2.0. 3rd Generation Partnership Project (3GPP), déc. 2021. URL : <http://www.3gpp.org/DynaReport/28531.htm> (cf. p. 41, 42, 45, 75).
- [107] 3GPP. « Management and orchestration; Generic management services ». Technical Specification (TS) 28.532. Version 16.10.0. 3rd Generation Partnership Project (3GPP), déc. 2021. URL : <http://www.3gpp.org/DynaReport/28532.htm> (cf. p. 41, 45).
- [108] 3GPP. « Management and orchestration; Architecture framework ». Technical Specification (TS) 28.533. Version 17.1.0. 3rd Generation Partnership Project (3GPP), déc. 2021. URL : <http://www.3gpp.org/DynaReport/28533.htm> (cf. p. 41, 45).
- [109] 3GPP. « Telecommunication management; Generic Network Resource Model (NRM) Integration Reference Point (IRP); Information Service (IS) ». Technical Specification (TS) 28.622. Version 17.0.0. 3rd Generation Partnership Project (3GPP), déc. 2021. URL : <http://www.3gpp.org/DynaReport/28622.htm> (cf. p. 42).
- [110] 3GPP. « Study on enhancement of network slicing; Phase 2 ». Technical Report (TR) 23.700-40. Version 17.0.0. 3rd Generation Partnership Project (3GPP), mars 2021. URL : <http://www.3gpp.org/DynaReport/23700-40.htm> (cf. p. 42).
- [111] 3GPP. « Telecommunication management; Management concept, architecture and requirements for mobile networks that include virtualized network functions ». Technical Specification (TS) 28.500. Version 16.0.0. 3rd Generation Partnership Project (3GPP), juill. 2020. URL : <http://www.3gpp.org/DynaReport/28500.htm> (cf. p. 42).
- [112] 3GPP. « Telecommunication management; Performance Management (PM) for mobile networks that include virtualized network functions; Requirements ». Technical Specification (TS) 28.520. Version 16.0.0. 3rd Generation Partnership Project (3GPP), juill. 2020. URL : <http://www.3gpp.org/DynaReport/28520.htm> (cf. p. 42).
- [113] 3GPP. « Telecommunication management; Performance Management (PM) for mobile networks that include virtualized network functions; Procedures ». Technical Specification (TS) 28.521. Version 16.0.0. 3rd Generation Partnership Project (3GPP), juill. 2020. URL : <http://www.3gpp.org/DynaReport/28521.htm> (cf. p. 42).

- [114] 3GPP. « Telecommunication management; Performance Management (PM) for mobile networks that include virtualized network functions; Stage 2 ». Technical Specification (TS) 28.522. Version 16.0.0. 3rd Generation Partnership Project (3GPP), juill. 2020. URL : <http://www.3gpp.org/DynaReport/28522.htm> (cf. p. 42).
- [115] 3GPP. « Telecommunication management; Performance Management (PM) for mobile networks that include virtualized network functions; Stage 3 ». Technical Specification (TS) 28.523. Version 16.0.0. 3rd Generation Partnership Project (3GPP), juill. 2020. URL : <http://www.3gpp.org/DynaReport/28523.htm> (cf. p. 42).
- [116] 3GPP. « Telecommunication management; Life Cycle Management (LCM) for mobile networks that include virtualized network functions; Requirements ». Technical Specification (TS) 28.525. Version 16.0.0. 3rd Generation Partnership Project (3GPP), juill. 2020. URL : <http://www.3gpp.org/DynaReport/28525.htm> (cf. p. 42).
- [117] 3GPP. « Telecommunication management; Life Cycle Management (LCM) for mobile networks that include virtualized network functions; Procedures ». Technical Specification (TS) 28.526. Version 16.0.0. 3rd Generation Partnership Project (3GPP), juill. 2020. URL : <http://www.3gpp.org/DynaReport/28526.htm> (cf. p. 42).
- [118] 3GPP. « Telecommunication management; Life Cycle Management (LCM) for mobile networks that include virtualized network functions; Stage 2 ». Technical Specification (TS) 28.527. Version 16.0.0. 3rd Generation Partnership Project (3GPP), juill. 2020. URL : <http://www.3gpp.org/DynaReport/28527.htm> (cf. p. 42).
- [119] 3GPP. « Telecommunication management; Life Cycle Management (LCM) for mobile networks that include virtualized network functions; Stage 3 ». Technical Specification (TS) 28.528. Version 16.0.0. 3rd Generation Partnership Project (3GPP), juill. 2020. URL : <http://www.3gpp.org/DynaReport/28528.htm> (cf. p. 42).
- [120] 3GPP. « Telecommunication management; Configuration Management (CM) for mobile networks that include virtualized network functions; Requirements ». Technical Specification (TS) 28.510. Version 16.0.0. 3rd Generation Partnership Project (3GPP), juill. 2020. URL : <http://www.3gpp.org/DynaReport/28510.htm> (cf. p. 42).
- [121] 3GPP. « Telecommunication management; Configuration Management (CM) for mobile networks that include virtualized network functions; Procedures ». Technical Specification (TS) 28.511. Version 16.0.0. 3rd Generation Partnership Project (3GPP), juill. 2020. URL : <http://www.3gpp.org/DynaReport/28511.htm> (cf. p. 42).



- [122] 3GPP. « Telecommunication management; Configuration Management (CM) for mobile networks that include virtualized network functions; Stage 2 ». Technical Specification (TS) 28.512. Version 16.0.0. 3rd Generation Partnership Project (3GPP), juill. 2020. URL : <http://www.3gpp.org/DynaReport/28512.htm> (cf. p. 42).
- [123] 3GPP. « Telecommunication management; Configuration Management (CM) for mobile networks that include virtualized network functions; Stage 3 ». Technical Specification (TS) 28.513. Version 16.0.0. 3rd Generation Partnership Project (3GPP), juill. 2020. URL : <http://www.3gpp.org/DynaReport/28513.htm> (cf. p. 42).
- [124] 3GPP. « Telecommunication management; Fault Management (FM) for mobile networks that include virtualized network functions; Requirements ». Technical Specification (TS) 28.515. Version 16.0.0. 3rd Generation Partnership Project (3GPP), juill. 2020. URL : <http://www.3gpp.org/DynaReport/28515.htm> (cf. p. 42).
- [125] 3GPP. « Telecommunication management; Fault Management (FM) for mobile networks that include virtualized network functions; Procedures ». Technical Specification (TS) 28.516. Version 16.0.0. 3rd Generation Partnership Project (3GPP), juill. 2020. URL : <http://www.3gpp.org/DynaReport/28516.htm> (cf. p. 42).
- [126] 3GPP. « Telecommunication management; Fault Management (FM) for mobile networks that include virtualized network functions; Stage 2 ». Technical Specification (TS) 28.517. Version 16.0.0. 3rd Generation Partnership Project (3GPP), juill. 2020. URL : <http://www.3gpp.org/DynaReport/28517.htm> (cf. p. 42).
- [127] 3GPP. « Telecommunication management; Fault Management (FM) for mobile networks that include virtualized network functions; Stage 3 ». Technical Specification (TS) 28.518. Version 16.0.0. 3rd Generation Partnership Project (3GPP), juill. 2020. URL : <http://www.3gpp.org/DynaReport/28518.htm> (cf. p. 42).
- [128] 3GPP. « Procedures for the 5G System (5GS) ». Technical Specification (TS) 23.502. Version 17.3.0. 3rd Generation Partnership Project (3GPP), déc. 2021. URL : <http://www.3gpp.org/DynaReport/23502.htm> (cf. p. 42, 45).
- [129] J. HALPERN et C. PIGNATARO. « Service Function Chaining (SFC) Architecture ». RFC 7665. Internet Engineering Task Force, oct. 2015, p. 1-32. DOI : {10.17487/RFC7665}. URL : <http://www.rfc-editor.org/rfc/rfc7665.txt> (cf. p. 43, 46).
- [130] 3GPP. « General Packet Radio System (GPRS) Tunnelling Protocol User Plane (GTPv1-U) ». Technical Specification (TS) 29.281. Version 17.1.0. 3rd Generation Partnership Project (3GPP), sept. 2021. URL : <http://www.3gpp.org/DynaReport/29281.htm> (cf. p. 43).

- [131] Jerome HENRY et al. « Diffserv to QCI Mapping ». Internet-Draft draft-henry-tsvwg-diffserv-to-qci-04. IETF, avr. 2020 (cf. p. 43).
- [132] GSMA. « E2E Network Slicing Architecture ». White Papier NG.127. GSM Association, mars 2021. URL : <https://www.gsma.com/newsroom/wp-content/uploads//NG.127-v1.0-1.pdf> (visité le 03/03/2022) (cf. p. 44).
- [133] 3GPP. « Telecommunication management; Study on management and orchestration of network slicing for next generation network ». Technical Report (TR) 28.801. Version 15.1.0. 3rd Generation Partnership Project (3GPP), jan. 2018. URL : <http://www.3gpp.org/DynaReport/28801.htm> (cf. p. 45).
- [134] 3GPP. « 5G System; Network Slice Selection Services; Stage 3 ». Technical Specification (TS) 29.531. Version 17.3.0. 3rd Generation Partnership Project (3GPP), déc. 2021. URL : <http://www.3gpp.org/DynaReport/29531.htm> (cf. p. 45).
- [135] 3GPP. « NG-RAN; NG Application Protocol (NGAP) ». Technical Specification (TS) 38.413. Version 16.8.0. 3rd Generation Partnership Project (3GPP), déc. 2021. URL : <http://www.3gpp.org/DynaReport/38413.htm> (cf. p. 45).
- [136] 3GPP. « Management and orchestration; Study on Network Slice Management Enhancement ». Technical Report (TR) 28.811. Version 17.0.0. 3rd Generation Partnership Project (3GPP), déc. 2021. URL : <http://www.3gpp.org/DynaReport/28811.htm> (cf. p. 46).
- [137] Xenofon FOUKAS et al. « Network Slicing in 5G: Survey and Challenges ». en. In : *IEEE Communications Magazine* 55.5 (mai 2017), p. 94-100. ISSN : 0163-6804. DOI : 10.1109/MCOM.2017.1600951. URL : <http://ieeexplore.ieee.org/document/7926923/> (visité le 15/02/2022) (cf. p. 47).
- [138] Ibrahim AFOLABI et al. « Network Slicing and Softwarization: A Survey on Principles, Enabling Technologies, and Solutions ». en. In : *IEEE Communications Surveys & Tutorials* 20.3 (2018). Number: 3, p. 2429-2453. ISSN : 1553-877X, 2373-745X. DOI : 10.1109/COMST.2018.2815638. URL : <https://ieeexplore.ieee.org/document/8320765/> (visité le 24/09/2019) (cf. p. 47).
- [139] Latif U. KHAN et al. « Network Slicing: Recent Advances, Taxonomy, Requirements, and Open Research Challenges ». en. In : *IEEE Access* 8 (2020), p. 36009-36028. ISSN : 2169-3536. DOI : 10.1109/ACCESS.2020.2975072. URL : <https://ieeexplore.ieee.org/document/9003208/> (visité le 15/02/2022) (cf. p. 47).
- [140] Mohammed CHAHBAR et al. « A Comprehensive Survey on the E2E 5G Network Slicing Model ». en. In : *IEEE Transactions on Network and Service Management* 18.1 (mars 2021). Number: 1, p. 49-62. ISSN : 1932-4537, 2373-7379. DOI :

- 10 . 1109 / TNSM . 2020 . 3044626. URL : <https://ieeexplore.ieee.org/document/9295415/> (visité le 31/01/2022) (cf. p. 47).
- [141] 5G-PPP. « First Wave of Research & Innovation Projects ». Rapp. tech. The 5G Infrastructure Public Private Partnership (5G PPP), 2015. URL : <https://5g-ppp.eu/wp-content/uploads/2015/10/5GPPP-brochure-final-web.pdf> (visité le 02/05/2022) (cf. p. 47).
- [142] 5G-PPP. « Second Wave of Research & Innovation Projects ». Rapp. tech. The 5G Infrastructure Public Private Partnership (5G PPP), 2017. URL : [https://5g-ppp.eu/wp-content/uploads/2021/06/5GPPP\\_Phase3\\_Brochure\\_WEB-2.pdf](https://5g-ppp.eu/wp-content/uploads/2021/06/5GPPP_Phase3_Brochure_WEB-2.pdf) (visité le 02/05/2022) (cf. p. 47).
- [143] 5G-PPP. « 5G PPP Projects - Phase 3 ». Rapp. tech. The 5G Infrastructure Public Private Partnership (5G PPP), 2021. URL : <https://5g-ppp.eu/wp-content/uploads/2017/11/5GPPP-brochure-phase2-final-web.pdf> (visité le 02/05/2022) (cf. p. 47).
- [144] Q. WANG et al. « SliceNet: End-to-End Cognitive Network Slicing and Slice Management Framework in Virtualised Multi-Domain, Multi-Tenant 5G Networks ». In : **2018 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB)**. 2018 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB). Juin 2018, p. 1-5. DOI : 10.1109/BMSB.2018.8436800 (cf. p. 47).
- [145] Pengtao ZHAO et al. « Information Prediction and Dynamic Programming-Based RAN Slicing for Mobile Edge Computing ». In : IEEE Wireless Communications Letters 7.4 (août 2018), p. 614-617. ISSN : 2162-2337, 2162-2345. DOI : 10.1109/LWC.2018.2802522. URL : <https://ieeexplore.ieee.org/document/8281474/> (visité le 19/02/2022) (cf. p. 50).
- [146] Pablo CABALLERO et al. « Multi-Tenant Radio Access Network Slicing: Statistical Multiplexing of Spatial Loads ». In : IEEE/ACM Transactions on Networking 25.5 (oct. 2017), p. 3044-3058. ISSN : 1063-6692, 1558-2566. DOI : 10.1109/TNET.2017.2720668. URL : <http://ieeexplore.ieee.org/document/7982693/> (visité le 19/02/2022) (cf. p. 50).
- [147] Anteneh A. GEBREMARIAM et al. « SoftSLICE: Policy-Based Dynamic Spectrum Slicing in 5G Cellular Networks ». In : **2018 IEEE International Conference on Communications (ICC)**. Kansas City, MO : IEEE, mai 2018, p. 1-6. ISBN : 978-1-5386-3180-5. DOI : 10.1109/ICC.2018.8422148. URL : <https://ieeexplore.ieee.org/document/8422148/> (visité le 19/02/2022) (cf. p. 50).
- [148] Phuong Luu Vo et al. « Slicing the Edge: Resource Allocation for RAN Network Slicing ». In : IEEE Wireless Communications Letters 7.6 (déc. 2018), p. 970-973. ISSN : 2162-2337, 2162-2345. DOI : 10.1109/LWC.2018.

2842189. URL : <https://ieeexplore.ieee.org/document/8370043/> (visité le 19/02/2022) (cf. p. 50).
- [149] Pablo CABALLERO et al. « Network Slicing Games: Enabling Customization in Multi-Tenant Mobile Networks ». In : *IEEE/ACM Transactions on Networking* 27.2 (avr. 2019), p. 662-675. ISSN : 1063-6692, 1558-2566. DOI : 10 . 1109 / TNET . 2019 . 2895378. URL : <https://ieeexplore.ieee.org/document/8642526/> (visité le 19/02/2022) (cf. p. 50).
- [150] Yang JIA et al. « Bankruptcy game based resource allocation algorithm for 5G Cloud-RAN slicing ». In : *2018 IEEE Wireless Communications and Networking Conference (WCNC)*. Barcelona : IEEE, avr. 2018, p. 1-6. ISBN : 978-1-5386-1734-2. DOI : 10 . 1109 / WCNC . 2018 . 8377187. URL : <https://ieeexplore.ieee.org/document/8377187/> (visité le 19/02/2022) (cf. p. 50).
- [151] Omer NARMANLIOGLU et Engin ZEYDAN. « Learning in SDN-based multi-tenant cellular networks: A game-theoretic perspective ». In : *2017 IFIP/IEEE Symposium on Integrated Network and Service Management (IM)*. Lisbon, Portugal : IEEE, mai 2017, p. 929-934. ISBN : 978-3-901882-89-0. DOI : 10.23919/INM.2017.7987414. URL : <http://ieeexplore.ieee.org/document/7987414/> (visité le 19/02/2022) (cf. p. 50).
- [152] Salvatore D'ORO et al. « Low-Complexity Distributed Radio Access Network Slicing: Algorithms and Experimental Results ». In : *IEEE/ACM Transactions on Networking* 26.6 (déc. 2018), p. 2815-2828. ISSN : 1063-6692, 1558-2566. DOI : 10 . 1109 / TNET . 2018 . 2878965. URL : <https://ieeexplore.ieee.org/document/8532127/> (visité le 19/02/2022) (cf. p. 50).
- [153] Peter ROST et al. « Network Slicing to Enable Scalability and Flexibility in 5G Mobile Networks ». In : *IEEE Communications Magazine* 55.5 (mai 2017), p. 72-79. ISSN : 0163-6804. DOI : 10 . 1109 / MCOM . 2017 . 1600920. URL : <http://ieeexplore.ieee.org/document/7926920/> (visité le 19/02/2022) (cf. p. 50, 51).
- [154] Rajesh MAHINDRA et al. « Radio Access Network sharing in cellular networks ». In : *2013 21st IEEE International Conference on Network Protocols (ICNP)*. Goettingen, Germany : IEEE, oct. 2013, p. 1-10. ISBN : 978-1-4799-1270-4. DOI : 10 . 1109 / ICNP . 2013 . 6733595. URL : <http://ieeexplore.ieee.org/document/6733595/> (visité le 19/02/2022) (cf. p. 51).
- [155] L. BALDINI et al. « SliceNet Control Plane for 5G Network Slicing in Evolving Future Networks ». In : *2019 IEEE Conference on Network Softwarization (NetSoft)*. 2019 IEEE Conference on Network Softwarization (NetSoft). Juin 2019, p. 450-457. DOI : 10 . 1109 / NETSOFT . 2019 . 8806707 (cf. p. 51).

- [156] Stuart CLAYMAN et al. « The NECOS Approach to End-to-End Cloud-Network Slicing as a Service ». In : *IEEE Communications Magazine* 59.3 (mars 2021), p. 91-97. ISSN : 0163-6804, 1558-1896. DOI : 10.1109/MCOM.001.2000702. URL : <https://ieeexplore.ieee.org/document/9422346/> (visité le 02/05/2022) (cf. p. 52).
- [157] Ved P. KAFLE et al. « Consideration On Automation of 5G Network Slicing with Machine Learning ». In : *2018 ITU Kaleidoscope: Machine Learning for a 5G Future (ITU K)*. 2018 ITU Kaleidoscope: Machine Learning for a 5G Future (ITU K). Santa Fe : IEEE, nov. 2018, p. 1-8. ISBN : 978-92-61-26921-0. DOI : 10.23919/ITU-WT.2018.8597639. URL : <https://ieeexplore.ieee.org/document/8597639/> (visité le 05/05/2022) (cf. p. 53).
- [158] Mustufa Haider ABIDI et al. « Optimal 5G network slicing using machine learning and deep learning concepts ». In : *Computer Standards & Interfaces* 76 (juin 2021), p. 103518. ISSN : 09205489. DOI : 10.1016/j.csi.2021.103518. URL : <https://linkinghub.elsevier.com/retrieve/pii/S0920548921000131> (visité le 05/05/2022) (cf. p. 54).
- [159] Shunsuke HOMMA et al. « Gateway Function for Network Slicing ». Internet-Draft draft-homma-rtgwg-slice-gateway-02. Work in Progress. Internet Engineering Task Force, mars 2020. 30 p. URL : <https://datatracker.ietf.org/doc/html/draft-homma-rtgwg-slice-gateway-02> (cf. p. 54).
- [160] Toufik AHMED et al. « On-Demand Network Slicing using SDN/NFV-enabled Satellite Ground Segment Systems ». In : *2018 4th IEEE Conference on Network Softwarization and Workshops (NetSoft)*. 2018 4th IEEE Conference on Network Softwarization and Workshops (NetSoft). ISSN: null. Juin 2018, p. 242-246. DOI : 10.1109/NETSOFT.2018.8460139 (cf. p. 70).
- [161] George F. RILEY et Thomas R. HENDERSON. « The ns-3 Network Simulator ». In : *Modeling and Tools for Network Simulation*. Sous la dir. de Klaus WEHRLE, Mesut GÜNEŞ et James GROSS. Berlin, Heidelberg : Springer, 2010, p. 15-34. ISBN : 978-3-642-12331-3. DOI : 10.1007/978-3-642-12331-3\_2. URL : [https://doi.org/10.1007/978-3-642-12331-3\\_2](https://doi.org/10.1007/978-3-642-12331-3_2) (visité le 23/05/2022) (cf. p. 92).
- [162] Alberto ALVAREZ et al. « Limitations of network emulation with single-machine and distributed ns-3 ». In : *Proceedings of the 3rd International ICST Conference on Simulation Tools and Techniques*. SIMUTools '10. Brussels, BEL : ICST (Institute for Computer Sciences, Social-Informatics et Telecommunications Engineering), 15 mars 2010, p. 1-9. ISBN : 978-963-9799-87-5. DOI : 10.4108/ICST.SIMUTOOLS2010.8630. URL : <https://>

- doi.org/10.4108/ICST.SIMUTOOLS2010.8630 (visité le 23/05/2022) (cf. p. 92).
- [163] Jason C NEUMANN. « The book of GNS3: build virtual network labs using Cisco, Juniper, and more ». No Starch Press, 2015 (cf. p. 92).
- [164] MININET. « Mininet: Rapid Prototyping for Software Defined Networks ». original-date: 2011-04-21T17:46:06Z. 24 mai 2022. URL : <https://github.com/mininet/mininet> (visité le 24/05/2022) (cf. p. 92).
- [165] ETSI. « Open Source MANO ». 2022. URL : <https://osm.etsi.org/> (visité le 23/05/2022) (cf. p. 93).
- [166] THE LINUX FOUNDATION. « Open Networking Automation Platform (ONAP) ». ONAP. 2022. URL : <https://www.onap.org/> (visité le 23/05/2022) (cf. p. 93).
- [167] VMWARE. « Bare Metal Hypervisor | ESX ». VMware. 2022. URL : <https://www.vmware.com/products/esxi-and-esx.html> (visité le 24/05/2022) (cf. p. 93).
- [168] ORACLE. « Oracle VM VirtualBox ». 2022. URL : <https://www.virtualbox.org/> (visité le 24/05/2022) (cf. p. 93).
- [169] PROXMOX. « Proxmox VE - Virtualization Management Platform ». 2022. URL : <https://www.proxmox.com/en/proxmox-ve> (visité le 24/05/2022) (cf. p. 93).
- [170] David BERNSTEIN. « Containers and Cloud: From LXC to Docker to Kubernetes ». In : IEEE Cloud Computing 1.3 (sept. 2014). Conference Name: IEEE Cloud Computing, p. 81-84. ISSN : 2325-6095. DOI : 10.1109/MCC.2014.51 (cf. p. 94).
- [171] LXC. « LXC ». original-date: 2012-09-07T18:50:27Z. 24 mai 2022. URL : <https://github.com/lxc/lxc> (visité le 24/05/2022) (cf. p. 94).
- [172] DOCKER. « The Moby Project ». original-date: 2013-01-18T18:10:57Z. 24 mai 2022. URL : <https://github.com/moby/moby> (visité le 24/05/2022) (cf. p. 94).
- [173] DOCKER. « Docker Compose v2 ». original-date: 2013-12-09T11:40:58Z. 24 mai 2022. URL : <https://github.com/docker/compose> (visité le 24/05/2022) (cf. p. 94).
- [174] DOCKER. « Docker Swarm ». Docker Documentation. 20 mai 2022. URL : <https://docs.docker.com/engine/swarm/> (visité le 24/05/2022) (cf. p. 94).
- [175] GOOGLE. « Kubernetes (K8s) ». original-date: 2014-06-06T22:56:04Z. 24 mai 2022. URL : <https://github.com/kubernetes/kubernetes> (visité le 24/05/2022) (cf. p. 94).

- [176] CNES et VIVÉRIS TECHNOLOGIE. « OpenBACH ». original-date: 2018-06-19T09:13:02Z. 21 avr. 2022. URL : <https://github.com/CNES/openbach> (visité le 02/06/2022) (cf. p. 94).
- [177] FREE5GC. « Free5GC ». original-date: 2020-01-31T08:45:17Z. 23 mai 2022. URL : <https://github.com/free5gc/free5gc> (visité le 24/05/2022) (cf. p. 96).
- [178] OPENAIRINTERFACE ALLIANCE. « OpenAirInterface 5G Core Network ». Git-Lab. 2022. URL : <https://gitlab.eurecom.fr/oai/cn5g> (visité le 24/05/2022) (cf. p. 96).
- [179] OPEN5GS ALLIANCE. « Open5GS ». Version 2.4.7. original-date: 2017-02-01T10:48:49Z. 24 mai 2022. URL : <https://github.com/open5gs/open5gs> (visité le 24/05/2022) (cf. p. 96).
- [180] GOOGLE. « The Go Programming Language ». original-date: 2014-08-19T04:33:40Z. 24 mai 2022. URL : <https://github.com/golang/go> (visité le 24/05/2022) (cf. p. 96).
- [181] FREE5GC. « gtp5g - 5G compatible GTP kernel module ». original-date: 2021-06-30T09:14:43Z. 2 mai 2022. URL : <https://github.com/free5gc/gtp5g> (visité le 30/05/2022) (cf. p. 96).
- [182] FREE5GC. « Session Management Function (SMF) ». original-date: 2020-04-26T19:53:06Z. 10 avr. 2022. URL : <https://github.com/free5gc/smf> (visité le 30/05/2022) (cf. p. 96).
- [183] Youssouf DRIF. « SMF - 5G Core Network with support of NTN ». original-date: 2021-06-01T09:40:36Z. 1<sup>er</sup> juin 2021. URL : <https://github.com/shynuu/5g-core-ntn/tree/main/NFs/smf> (visité le 30/05/2022) (cf. p. 96).
- [184] Youssouf DRIF. « QOF - 5G Core Network with support of NTN ». original-date: 2021-06-01T09:40:36Z. 1<sup>er</sup> juin 2021. URL : <https://github.com/shynuu/5g-core-ntn/tree/main/NFs/qof> (visité le 30/05/2022) (cf. p. 96).
- [185] ROBERT McMAHON. « IPerf2 ». 2022. URL : <https://sourceforge.net/p/iperf2/code/ci/master/tree/> (visité le 30/05/2022) (cf. p. 96).
- [186] Ali GÜNGÖR. « UERANSIM ». Version 3.2.6. original-date: 2019-09-25T05:44:14Z. 23 mai 2022. URL : <https://github.com/aligungr/UERANSIM> (visité le 23/05/2022) (cf. p. 97).
- [187] Youssouf DRIF. « GTP DSCP Copy ». original-date: 2021-04-29T23:41:05Z. 1<sup>er</sup> juin 2021. URL : <https://github.com/shynuu/gtp-dscp> (visité le 31/05/2022) (cf. p. 97).
- [188] Youssouf DRIF. « Trunks ». original-date: 2021-03-13T22:26:22Z. 8 avr. 2022. URL : <https://github.com/shynuu/trunks> (visité le 24/05/2022) (cf. p. 97).

- [189] CNES. « OpenSAND ». Version v6.0.0. original-date: 2019-10-03T13:14:52Z. 26 avr. 2022. URL : <https://github.com/CNES/opensand> (visité le 24/05/2022) (cf. p. 97).
- [190] MAGISTER, CENTRE NATIONAL D'ÉTUDES SPATIALES et EUROPEAN SPACE AGENCY. « Satellite ns3 ». original-date: 2018-04-03T15:37:39Z. 20 mai 2022. URL : <https://github.com/sns3/sns3-satellite> (visité le 23/05/2022) (cf. p. 97).
- [191] YOUSSEUF DRIF. « NTNQOF - 5G Core Network with support of NTN ». original-date: 2021-06-01T09:40:36Z. 1<sup>er</sup> juin 2021. URL : <https://github.com/shynuu/5g-core-ntn/tree/main/NFs/ntnqof> (visité le 30/05/2022) (cf. p. 97).
- [192] YOUSSEUF DRIF. « Slice Aware Non Terrestrial Networks ». original-date: 2021-05-31T22:05:15Z. 8 avr. 2022. URL : <https://github.com/shynuu/slice-aware-ntn> (visité le 24/05/2022) (cf. p. 98).
- [193] AMARISOFT. « AMARI Callbox Mini ». 2022. URL : <https://www.amarisoft.com/app/uploads/2021/10/AMARI-Callbox-Mini.pdf> (visité le 25/05/2022) (cf. p. 110).
- [194] 3GPP. « NR; User Equipment (UE) radio transmission and reception ». Technical Specification (TS) 38.101. Version 0.0.1. 3rd Generation Partnership Project (3GPP). URL : <http://www.3gpp.org/DynaReport/38101.htm> (cf. p. 112).