



**HAL**  
open science

# Livraison de contenus sur un réseau hybride satellite / terrestre

Elie Bernard Bouttier

► **To cite this version:**

Elie Bernard Bouttier. Livraison de contenus sur un réseau hybride satellite / terrestre. Réseaux et télécommunications [cs.NI]. Institut National Polytechnique de Toulouse - INPT, 2018. Français. NNT : 2018INPT0056 . tel-04215144

**HAL Id: tel-04215144**

**<https://theses.hal.science/tel-04215144v1>**

Submitted on 22 Sep 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 :**

Réseaux, Télécommunications, Systèmes et Architecture

---

**Présentée et soutenue par :**

M. ELIE BERNARD BOUTTIER

le jeudi 5 juillet 2018

**Titre :**

Livraison de contenus sur un réseau hybride satellite / terrestre

---

**Ecole doctorale :**

Mathématiques, Informatique, Télécommunications de Toulouse (MITT)

**Unité de recherche :**

Institut de Recherche en Informatique de Toulouse (I.R.I.T.)

**Directeur(s) de Thèse :**

M. ANDRE LUC BEYLOT

M. RIADH DHAOU

**Rapporteurs :**

M. DANIEL NEGRU, INP BORDEAUX

M. THIERRY TURLETTI, INRIA SOPHIA ANTIPOLIS

**Membre(s) du jury :**

M. STEVEN MARTIN, UNIVERSITE PARIS 11, Président

M. ANDRE LUC BEYLOT, INP TOULOUSE, Membre

M. EMMANUEL DUBOIS, CENTRE NATIONAL D'ETUDES SPATIALES CNES, Membre

M. FABRICE ARNAL, THALES ALENIA SPACE, Membre

Mme DOMINIQUE GAITI, UNIVERSITE DE TECHNOLOGIE DE TROYES, Membre

M. RIADH DHAOU, INP TOULOUSE, Membre



## Remerciements

Parfois la route est longue ... et elle le fut. Ce n'est qu'avec le soutien d'un grand nombre de personnes que les obstacles, les moments de doutes et d'hésitation ont pu être surmontés.

Je remercie en premier lieu mon directeur de thèse, M. André-Luc BEYLOT, qui m'aura fait progresser autant scientifiquement qu'orthographiquement ! Je le remercie pour m'avoir donné la possibilité de faire cette thèse, son encadrement riche d'une grande expérience et ses nombreuses relectures de ce manuscrit. Je remercie également M. Riadh DHAOU qui s'est toujours rendu promptement disponible lors de mes sollicitations, pour ses commentaires et ses relectures également, et avec qui j'ai pu avoir de nombreuses discussions stimulant mes activités de recherche.

Je remercie aussi M. Emmanuel DUBOIS pour son soutien, par ses mots et par sa mise à disposition des ressources du CNES. Ainsi que MM. Fabrice ARNAL et Cédric BAUDOIN, pour leur suivi de cette thèse, leurs remarques et leur accueil à Thales Alenia Space. L'encadrement industriel dont j'ai bénéficié durant cette thèse aura été d'une grande richesse et extrêmement stimulant.

Je remercie les rapporteurs de cette thèse, MM. Thierry TURLETTI et Daniel NEGRU, pour leur relecture attentive de ces travaux. Je remercie également M. Steven MARTIN qui a présidé mon jury ainsi que Mme Dominique GAÏTI pour sa participation malgré l'éloignement.

Je tiens à remercier également Mme Corinne MAILHES, directrice du TESA, qui m'a accueilli très chaleureusement dans ses locaux pour une période de ma thèse.

Il me faut remercier M. Emmanuel CHAPUT qui m'a permis de rencontrer l'équipe d'encadrement qui deviendra la mienne par la suite. Je remercie également les autres permanents de l'équipe, M. Julien FASSON, M. Carlos AGUILAR, et tous ceux avec qui j'ai pu échanger, scientifiquement ou non.

Je remercie aussi notre secrétaire, Mme Sylvie ARMENGO-METCHE, qui s'est montré d'une très grande réactivité à chacune de mes sollicitations.

Merci également à l'ensemble des doctorants que j'ai pu côtoyer durant cette thèse. Je remercie mes prédécesseurs, Jean-Gabriel, Émilie, Nesrine, Aziz, Farouk, et en particulier MM. Renaud SALLANTIN et Jean-Baptiste DUPÉ, qui ont su me guider lors de mes premiers pas dans cette aventure. Je remercie également mes successeurs, Yoann qui m'aura rendu accro à

l'escalade, Dorin pour nos discussions sur le secourisme, Oana qui a partagé généreusement beaucoup de ses desserts avec moi, Cédric pour nos discussions sur le chiffrement, Kuljaree et Amal qui ont partagé mon bureau, Adrien, Elene, Santiago, Hugo, Mohammed, Raphaël, ainsi que tous les autres.

Je remercie également ma famille, notamment mes parents qui m'ont toujours fait confiance et soutenu dans tous mes choix, et sur qui j'ai toujours pu compter.

Je remercie tous mes compagnons de net7, auprès de qui j'ai énormément appris, je leur dois pour beaucoup les connaissances techniques qui m'ont permises de mener à bien cette thèse. Merci également pour l'aide qu'ils m'ont apportée durant la thèse, en particulier M. Vincent DUVERT qui m'a énormément fait profiter de ses compétences rares. Pour finir, je remercie tous mes amis, Drômois, Toulousain, et d'ailleurs, qui auront su me faire lever la tête et revigorer durant ma thèse.

Pour conclure, je souhaite à tout doctorant de profiter d'un cadre aussi riche que celui dont j'ai eu la chance de bénéficier. Bonne chance aux nouveaux venus, Amal et Kévin!

## Résumé

L'augmentation et le renforcement des usages d'Internet rendent nécessaire l'évolution des réseaux existants. Cependant, on constate de fortes inégalités entre les zones urbaines, bien desservies et qui concentrent l'essentiel des investissements, et les zones rurales, mal desservies et délaissées.

Face à cette situation, les utilisateurs de ces zones se tournent vers d'autres moyens d'accès, et notamment vers les accès Internet par satellite. Cependant, ces derniers souffrent d'une limitation qui est le délai important induit par le temps de propagation du signal entre la terre et l'orbite géostationnaire. Dans cette thèse, nous nous intéressons à l'utilisation simultanée d'un réseau d'accès terrestre, caractérisé par un faible débit et un faible délai, et d'un réseau d'accès satellite, caractérisé par une forte latence et un débit plus important.

D'autre part, les réseaux de diffusion de contenus ou CDNs, constitués d'un grand nombre de serveurs de cache, apportent une réponse à l'augmentation du trafic et des besoins en termes de latence et de débit. Cependant, localisés dans les réseaux de cœur, les caches restent éloignés des utilisateurs et n'atteignent pas les réseaux d'accès. Ainsi, les fournisseurs d'accès Internet (FAI) se sont intéressés au déploiement de ces serveurs au sein de leur propre réseau, que l'on appelle alors TelCo CDN. La diffusion des contenus nécessite idéalement l'interconnexion des opérateurs CDN avec les TelCo CDNs, permettant ainsi la délégation de la diffusion à ces derniers. Ils sont alors en mesure d'optimiser la diffusion des contenus sur leur réseau dont ils ont une meilleure connaissance.

Ainsi, nous nous intéresserons à l'optimisation de la livraison de contenus sur un réseau hybride satellite / terrestre intégré à une chaîne de livraison CDN. Nous nous attacherons dans un premier temps à décrire une architecture permettant, grâce à l'interconnexion de CDNs, de prendre en charge la diffusion des contenus sur le réseau hybride. Dans un second temps, nous étudierons l'intérêt de la connaissance des informations apportées par le contexte CDN pour le routage sur une telle architecture. Dans ce cadre, nous proposerons un mécanisme de routage fondé sur la taille des contenus. Finalement, nous montrerons la supériorité de notre approche sur l'utilisation du protocole de transport multichemin MP-TCP.



## Abstract

The increase and reinforcement of Internet uses make necessary to improve existing networks. However, we observe strong inequalities between urban areas, well served and which concentrate the major part of investments, and rural areas, underserved and forsaken.

To face this situation, users in underserved areas are moving to others Internet access, and in particular satellite Internet access. However, the latter suffer from a limitation which is the long delay induced by the propagation time between the earth and the geostationary orbit. In this thesis, we are interested in the simultaneous use of a terrestrial access network, characterized by a low delay and a low throughput, and a satellite access network, characterized by a high throughput and an long delay.

Elsewhere, Content Delivery Networks (CDNs), consisting of a large number of cache servers, bring an answer to the increase in traffic and needs in terms of latency and throughput. However, located in core networks, cache servers stay far from end users and do not reach access networks. Thus, Internet Service Providers (ISPs) have taken an interest in deploying their own CDNs, which will be referred to as TelCo CDNs. The content delivery ideally needs the interconnection between CDN operators and TelCo CDNs, allowing the delegation of the content delivery to the TelCo CDNs. The latter are then able to optimize the content delivery on their network, for which they have a better knowledge.

Thus, we will study the optimization of the contents delivery on a hybrid satellite / terrestrial network, integrated in a CDN delivery chain. We will initially focus on the description of a architecture allowing, thanks to a CDN interconnection, handling contents delivery on the hybrid network. In a second stage, we will study the value of the information provided by the CDN context in the routing on such architecture. In this framework, we will propose a routing mechanism based on contents size. Finally, we will show the superiority of our approach over the multipath transport protocol MP-TCP.



# Table des matières

Glossaire . . . . .	xv
Acronymes . . . . .	xv
<b>1 Introduction</b>	<b>1</b>
1.1 Les réseaux de diffusion de contenus . . . . .	2
1.2 Les réseaux hybrides satellite / terrestre . . . . .	4
1.2.1 Zones mal desservies . . . . .	4
1.2.2 Internet par satellite . . . . .	5
1.2.3 Les limitations de l'accès à Internet par satellite . . . . .	5
1.2.4 Les réseaux hybrides satellite / terrestre . . . . .	6
1.2.5 Le routage multichemin . . . . .	6
1.3 Contributions . . . . .	7
1.3.1 Architecture de livraison hybride satellite / terrestre . . . . .	7
1.3.2 Le routage fondé sur la taille . . . . .	8
1.3.3 MinFCT . . . . .	8
1.3.4 Le routage par flux et par paquet . . . . .	8
1.3.5 Comparaison avec MP-TCP . . . . .	9
<b>2 Réseau de diffusion de contenus</b>	<b>11</b>
2.1 Les réseaux de diffusion de contenus . . . . .	11
2.1.1 Introduction . . . . .	11
2.1.2 Principe de fonctionnement d'un CDN . . . . .	12
2.1.3 Avantages à l'utilisation d'un CDN . . . . .	12
2.1.4 Problématiques opérationnelles de gestion d'un CDN . . . . .	13
2.1.5 Les TelCo CDN . . . . .	16
2.2 L'interconnexion de CDN . . . . .	17
2.2.1 Routage des requêtes . . . . .	19
2.2.2 Journalisation et métrologie . . . . .	21
2.2.3 Méta-données . . . . .	22
2.2.4 Déclenchement d'opération de contrôle . . . . .	23
2.2.5 Gestion du trafic chiffré . . . . .	23
<b>3 Les communications multichemins</b>	<b>27</b>
3.1 Le routage multichemin . . . . .	27
3.1.1 Les réseaux multichemins . . . . .	27

3.1.2	Problèmes de routage multichemin dans les réseaux hybrides . . . . .	27
3.1.3	Routage multichemin et partage de charge au niveau de la couche réseau . . . . .	29
3.1.4	Protocoles de transport multichemins . . . . .	32
3.1.5	Protocoles multichemins applicatifs . . . . .	35
3.1.6	Autres travaux . . . . .	36
3.1.7	Synthèse . . . . .	36
3.2	MP-TCP et ses évolutions / variantes . . . . .	38
3.2.1	Fonctionnement général . . . . .	38
3.2.2	Contrôle de congestion . . . . .	39
3.2.3	Ordonnancement . . . . .	41
3.3	Le projet BATS . . . . .	41
3.3.1	Contexte et objectifs du projet BATS . . . . .	42
3.3.2	Architecture . . . . .	42
3.3.3	Classification du trafic . . . . .	43
3.3.4	Routage du trafic . . . . .	44
3.3.5	Résultats . . . . .	44
<b>4</b>	<b>Architecture hybride et CDN</b>	<b>47</b>
4.1	Acteurs . . . . .	47
4.2	Architecture de routage hybride satellite / terrestre . . . . .	48
4.3	Mise en cache et interconnexion CDNI . . . . .	49
4.3.1	La fin de la mise en cache transparente . . . . .	49
4.3.2	Interconnexion CDNI . . . . .	49
4.3.3	Informations utiles au routage . . . . .	50
4.3.4	Mise en cache par l'UG . . . . .	50
4.3.5	Diffusion en <i>multicast</i> par satellite . . . . .	54
4.4	Conclusion . . . . .	54
<b>5</b>	<b>Routage fondé sur la taille</b>	<b>55</b>
5.1	Modélisation . . . . .	56
5.1.1	Hypothèses simplificatrices . . . . .	57
5.1.2	Critère de performance . . . . .	58
5.1.3	Équité, Qualité de Service (QoS) et discipline de service . . . . .	58
5.2	Algorithmes . . . . .	59
5.2.1	Routage probabiliste . . . . .	59
5.2.2	Routage par seuil . . . . .	59
5.2.3	MinFCT : <i>Minimum Flow Completion Time</i> . . . . .	60
5.3	Étude analytique des performances . . . . .	62
5.3.1	Réseau de files d'attente . . . . .	62
5.3.2	Analyse des valeurs moyennes (MVA) . . . . .	64
5.3.3	L'algorithme MVA . . . . .	64
5.3.4	Implantation de la méthode MVA . . . . .	65

5.4	Étude des performances par simulation . . . . .	65
5.4.1	Ordonnancement . . . . .	65
5.5	Étude des performances par émulation . . . . .	67
5.5.1	Architecture . . . . .	67
5.5.2	Client . . . . .	67
5.5.3	Serveurs mandataires intelligents . . . . .	68
5.5.4	Serveur . . . . .	68
5.6	Résultats . . . . .	68
5.6.1	Paramètres . . . . .	68
5.6.2	Routage probabiliste . . . . .	69
5.6.3	Routage par seuil . . . . .	73
5.6.4	MinFCT . . . . .	76
5.6.5	Comparaison des trois méthodes de routage . . . . .	76
<b>6</b>	<b>Routage sous l'échelle du flux</b>	<b>79</b>
6.1	Agrégation . . . . .	79
6.1.1	Agrégation consciente de la taille . . . . .	80
6.1.2	Résultats . . . . .	80
6.2	Agrégation opportuniste des liens inactifs . . . . .	82
6.2.1	Résultats . . . . .	83
6.3	Réallocation . . . . .	83
6.3.1	Résultats . . . . .	85
6.4	Conclusion . . . . .	86
<b>7</b>	<b>Routage avec transport fiable</b>	<b>87</b>
7.1	Influence du protocole de transport . . . . .	87
7.2	Estimation du temps de téléchargement . . . . .	88
7.3	Estimation des paramètres des liens . . . . .	91
7.3.1	Délai . . . . .	91
7.3.2	Capacité . . . . .	92
7.3.3	Taux de perte . . . . .	92
7.3.4	Nombre de flux concurrents . . . . .	93
7.4	Expérimentations . . . . .	93
7.4.1	Architecture . . . . .	94
7.4.2	Résultats . . . . .	96
7.5	Conclusion . . . . .	100
<b>8</b>	<b>Conclusion et perspectives</b>	<b>101</b>
8.1	Conclusion . . . . .	101
8.2	Perspectives . . . . .	102
8.2.1	Architecture . . . . .	102
8.2.2	Protocole de transport . . . . .	103
8.2.3	Mécanisme de routage . . . . .	104
8.2.4	Affinement de la modélisation . . . . .	107

8.2.5	Conclusion . . . . .	107
<b>Annexe A Plateforme d'expérimentation</b>		<b>109</b>
A.1	Implantation des composants . . . . .	109
A.1.1	Client . . . . .	109
A.1.2	Serveurs mandataires . . . . .	110
A.1.3	Serveur . . . . .	112
A.1.4	Estimation des paramètres des liens . . . . .	112
A.2	Orchestration . . . . .	112

# Table des figures

1.1	Exemple de réseau hybride faisant intervenir un lien terrestre ( <i>e.g.</i> ADSL) et un lien satellite. . . . .	6
2.1	Récupération d'un contenu par un client depuis un CDN. . .	12
2.2	Exemple d'architecture CDN ( <i>Content Delivery Network</i> ) hiérarchisée. . . . .	17
2.3	Interconnexion d'un TelCo CDN avec un opérateur CDN en charge de la diffusion des contenus d'un fournisseur. Les clients du fournisseur d'accès accèdent ainsi aux contenus depuis un cache plus proche. . . . .	18
2.4	Mécanisme de redirection récursif . . . . .	20
2.5	Établissement d'une session TLS en faisant appel à un serveur de clef distant opéré par le fournisseur de contenus. . . . .	25
3.1	Phénomène de blocage en tête de file : le message 3 doit attendre la réception du message 2 avant de pouvoir être appliqué. . . . .	28
3.2	Exemple d'initialisation d'une connexion MP-TCP. . . . .	39
3.3	Deux situations inévitables au bénéfice de MP-TCP. . . . .	40
3.4	Architecture BATS . . . . .	43
4.1	Architecture hybride satellite / terrestre . . . . .	48
4.2	Architecture hybride satellite / terrestre . . . . .	50
4.3	Redirection HTTP itérative du client. . . . .	52
4.4	Redirection DNS puis redirection HTTP avec usage limité d'une clef distante. . . . .	52
4.5	Redirection HTTP récursive du client. . . . .	53
5.1	Étapes d'un cycle requête-réponse. . . . .	56
5.2	L'affectation du flux jaune au même lien augmente le temps résiduel de téléchargement des flux rouge et bleu. . . . .	61
5.3	Représentation du scénario par un réseau de files d'attente. . .	63
5.4	Architecture du banc de test. . . . .	67
5.5	Temps de téléchargement moyen en fonction du nombre d'utilisateurs pour différentes probabilités de routage. . . . .	69

5.6	Temps de téléchargement moyen en fonction de la probabilité de routage par le réseau terrestre. . . . .	70
5.7	Taux d'utilisation des liens satellite et terrestre calculé à partir de MVA. . . . .	71
5.8	Probabilités optimales selon MVA et temps de téléchargement moyen associé obtenu par MVA, simulations et expérimentations. . . . .	72
5.9	Temps de téléchargement moyen en fonction du nombre d'utilisateurs pour différents seuils. . . . .	73
5.10	Temps de téléchargement moyen en fonction du seuil. . . . .	74
	5.11 Seuil optimal selon MVA en fonction du nombre d'utilisateurs, et temps de téléchargement moyen associé obtenu MVA, simulations et expérimentations. . . . .	75
5.12	Temps de téléchargement moyen en fonction du nombre d'utilisateurs dans le système pour la méthode de routage MinFCT. . . . .	76
	5.13 Temps de téléchargement moyen en fonction du nombre d'utilisateurs dans le système pour les trois méthodes de routage étudiées. . . . .	77
6.1	Comparaison des performances des mécanismes d'agrégation avec la méthode de routage par flux MinFCT. . . . .	81
6.2	Taux d'occupation des liens terrestre et satellite avec les stratégies de routage MinFCT et d'agrégation consciente de la taille des flux. . . . .	82
6.3	Comparaison de MinFCT avec et sans le mécanisme d'agrégation opportuniste des liens inactifs. . . . .	83
6.4	Comparaison de MinFCT avec et sans réallocation. . . . .	85
7.1	Plateforme d'expérimentation avec limitation des débits. . . . .	94
7.2	Comparaison des performances de MinFCT par rapport à un routage probabiliste et par seuil ainsi qu'à un routage monochemin. . . . .	97
7.3	Comparaison des performances de MinFCT et MP-TCP. . . . .	98
7.4	Comparaison du taux d'occupation des liens terrestre et satellite entre le routage probabiliste, MinFCT et MP-TCP. . . . .	99

# Liste des tableaux

3.1 Comparaison des avantages et inconvénients des différentes couches pour la mise en place d'un mécanisme de routage multichemin. . . . .	37
7.1 Paramètres utilisés pour la <code>qdisc tbf</code> . . . . .	95

# Glossaire et acronymes

## Glossaire

**AS-Path** Lors de la réception d'une route par BGP (*Border Gateway Protocol*), l'AS-Path indique la liste des ASs (*Autonomous Systems*) ayant retransmis la route, de l'annonceur initial jusqu'au récepteur. 15

**insensibilité** Propriété dite d'une file d'attente lorsque la distribution stationnaire du nombre d'utilisateurs dans celle-ci ne dépend pas de la loi de service, si ce n'est par sa moyenne. 64

**serveur DNS récursif** Serveur DNS procédant à la résolution récursive des noms de domaines à la demande des clients DNS des utilisateurs finaux. 15, 21, 36, 53

## Acronymes

**ACM** *Adaptive Coding and Modulation* 5

**ACME** *Automatic Certificate Management Environment* 25

**ADSL** *Asymmetric Digital Subscriber Line* xii, 1, 4, 6

**AS** *Autonomous System* xv, 14

**BATS** *Broadband Access via integrated Terrestrial and Satellite systems* 41, 59

**BBRR** *Bit-by-Bit Round-Robin* 66

**BCMP** BASKETT, CHANDY, MUNTZ, PALACIOS 64

**BGP** *Border Gateway Protocol* xv, 15, 16

**CDN** *Content Delivery Network* xii, 2–4, 6, 7, 11–25, 36, 47, 51, 52, 55–57, 60, 102, 106

**CDNI** *Content Delivery Network Interconnection* 4, 18, 50, 51, 102, 103

**CIT** *Control Interface / Triggers* 23



- DHCP** *Dynamic Host Configuration Protocol* 36
- DNS** *Domain Name System* 15, 20
- dPIN** *double Pareto log-normale* 64, 74
- DSCP** *Differentiated Services Code Point* 43
- DSLAM** *Digital Subscriber Line Access Multiplexer* 4, 7
- DSS** *Data Sequence Signal* 39
- ETSI** *European Telecommunications Standards Institute* 42
- FAI** fournisseur d'accès Internet 3, 4, 16
- FCFS** *First-Come First-Served* 66
- FCS** *Footprint and Capabilities Semantics* 19, 20
- FCT** *Flow Completion Time* 8, 58, 62
- FIFO** *First-In First-Out* 13
- FQ** *Fair Queuing* 111
- FTP** FILE TRANSFER PROTOCOL 35
- HTS** *High-Throughput Satellite* 6
- HTTP** *HyperText Transfert Protocol* 15, 19–21, 35, 50
- IETF** *Internet Engineering Task Force* 4, 18, 36
- ING** *Intelligent Network Gateway* 48–52, 55–58, 60
- IP** *Internet Protocol* 7, 14, 15, 35, 36
- IUG** *Intelligent User Gateway* 48–52, 55–58
- JSON** *JavaScript Object Notation* 19
- LEO** *Low Earth Orbit* 5
- LFN** *Long Flat Network* 35
- LFU** *Least Frequently Used* 13
- LI** *Logging Interface* 22
- LRU** *Least Recently Used* 13
- LTE** *Long-Term Evolution* 1, 3, 106
- LURK** *Limited Use of Remote Keys* 24
- MDMC** *Multiple Description Motion Compensation* 34
- MEO** *Medium Earth Orbit* 5
- MIF** *Multiple InterFace* 36
- MIME** *Multipurpose Internet Mail Extensions* 8

- MinFCT** *Minimum Flow Completion Time* 8, 60
- MP-QUIC** *Multipath Extension for QUIC* 34
- MP-TCP** *Multipath TCP* 7, 34
- MVA** *Mean-Value Analysis* 64
- NRA** Nœud de Raccordement d'Abonnés 3
- NRO** Nœud de Raccordement Optique 3
- ORM** *Object-Relational Mapping* 112
- PEP** *Performance-Enhancing Proxy* 2, 5, 92, 103, 104
- PFQ** *Packet-by-Packet Fair Queueing* 66
- POP** *Point-of-Presence* 3, 16
- PS** *Processor Sharing* 63–66
- QUIC** *Quick UDP Internet Connections* 34
- RCT** *Remaining Completion Time* 62
- RFC** *requests for comment* 35
- RR** *Round-Robin* 66
- RRI** *Request Routing Interface* 20, 21
- RSA** RIVEST, SHAMIR, ADLEMAN 24
- RTT** *Round-Trip Time* 90, 91, 95, 104
- SACK** *Selective Acknowledgment* 92
- SFQ** *Stochastic Fairness Queueing* 66
- SNI** *Server Name Indicator* 24
- SRPT** *Shortest-Remaining-Processing-Time* 59
- SRTT** *Smoothed RTT* 91
- STAR** *Short-Term, Automatically-Renewed* 25
- TCP** *Transmission Control Protocol* 5, 7, 21, 28, 33–35
- TelCo CDN** *Telecommunication Company CDN* 2–4, 7, 16, 17, 51, 52, 103
- TLS** *Transport Layer Security* 5, 21, 24
- TTL** *Time-To-Live* 15
- UDL** *Unidirectional Link* 6
- UDP** *User Datagram Protocol* 32–34
- URI** *Uniform Resource Indicator* 15, 111

**URL** *Uniform Resource Locator* 49

**VDSL** *Very-high-bit-rate Digital Subscriber Line* 1, 4

**VSAT** *Very Small Aperture Terminal* 6

**WiMAX** *Worldwide Interoperability for Microwave Access* 1

**WRR** *Weighted Round-Robin* 32

# Chapitre 1

## Introduction

L'utilisation d'Internet connaît une croissance exponentielle et ininterrompue avec des usages qui se renforcent et se diversifient. Les services proposés s'étendent à une population d'utilisateurs plus large et à de nouveaux types de terminaux. Parallèlement, de nouveaux services émergent, rendant les utilisateurs toujours plus dépendants d'Internet. Ainsi, les exigences en termes de qualité, fiabilité et résilience des réseaux augmentent.

Afin de répondre à ces nouvelles attentes, il est nécessaire de continuer à faire évoluer les réseaux. Cependant, on constate de fortes inégalités entre les territoires européens, aussi bien dans l'état actuel de leur connectivité que dans l'évolution de celle-ci.

En effet, on observe d'une part des zones urbaines, relativement bien desservies. Les réseaux cuivrés y sont d'une longueur faible, assurant un débit ADSL (*Asymmetric Digital Subscriber Line*) élevé et permettant de bénéficier de ses évolutions telles que le VDSL (*Very-high-bit-rate Digital Subscriber Line*). Ces zones ont pu voir le déploiement de réseaux câblés et connaissent actuellement le déploiement de la fibre optique.

D'autre part, on observe des zones rurales mal desservies et qui n'évoluent pas. Les lignes de cuivre y sont plus longues, offrant des débits ADSL faibles et ne permettant pas de bénéficier de ses évolutions. Le remplacement du cuivre par la fibre optique y est délaissé au profit des zones plus densément peuplées. Ainsi, les écarts de débit entre zones urbaines et zones rurales s'accroissent.

Devant cette impasse, les utilisateurs des zones mal desservies sont obligés de se tourner vers d'autres moyens d'accès moins coûteux et plus rapides à déployer. Ainsi, les solutions sans-fil, plus souples, sont privilégiées. Certains villages ont pu connaître le déploiement d'antennes WiMAX (*World-wide Interoperability for Microwave Access*), dont les fréquences ont été très récemment ré-allouées en France pour l'utilisation du standard LTE (*Long-Term Evolution*), plus performant. Des offres « 4G fixe » reposant sur les réseaux 4G traditionnels ont également fait leur apparition dernièrement.

L'Internet par satellite est également une solution prisée par les utilisateurs et ce depuis plusieurs années.

Ainsi, l'Internet par satellite apparaît comme une solution complémentaire, moins contraignante et capable d'atteindre les zones les plus reculées. À ce titre, on observe une forte croissance de la demande pour ce type d'accès [66]. Parallèlement, l'offre se développe avec l'apparition de nouveaux services commerciaux, et, de manière emblématique, le lancement de KASAT, premier satellite européen intégralement dédié à l'accès Internet par satellite en 2011 [44].

Cependant, les accès Internet par satellite ne sont pas des solutions suffisantes. En effet, ils ne constituent qu'un maillon de la chaîne de livraison, qui doit être optimisée dans son ensemble. D'autre part, l'Internet par satellite souffre d'une limitation spécifique qui demeure le délai important induit par le temps de propagation du signal entre la terre et l'orbite géostationnaire.

Les PEPs (*Performance-Enhancing Proxies*) ont constitué une piste de recherche afin de masquer l'importante latence propre aux réseaux satellitaires. Dans cette thèse, nous nous intéresserons à une autre approche qui est l'utilisation simultanée d'un réseau d'accès terrestre et d'un réseau d'accès par satellite. En effet, malgré leur bas débit dans les zones considérées, les réseaux filaires terrestres ne subissent pas des délais semblables à ceux des réseaux satellitaires. Fort de cette observation, il est alors intéressant de combiner ces deux réseaux aux caractéristiques distinctes, dans l'objectif de tirer profit de leurs avantages respectifs. Dans la suite, on désignera ce type d'architecture sous l'appellation de réseau hybride satellite / terrestre.

D'autre part, les réseaux de diffusion de contenus ou CDNs (*Content Delivery Networks*) constituent une réponse à l'augmentation du trafic et des besoins en termes de latence et de débit. Composés d'un grand nombre de serveurs de caches habilement répartis, ils ont permis de réduire la charge pesant sur les serveurs de contenus et sur les réseaux tout en diminuant le temps d'accès pour les utilisateurs. Cette approche est aujourd'hui largement répandue, la part du trafic d'Internet servie par ces derniers est majoritaire et toujours en augmentation [28]. Elle se renforce encore avec l'apparition de TelCo CDNs (*Telecommunication Company CDNs*), plus proches des utilisateurs, pouvant là encore améliorer l'expérience ressentie par ces derniers.

Ainsi, nous étudierons le problème du routage multichemin sur un réseau hybride satellite / terrestre intégré dans une architecture CDN. Dans ce contexte, nous montrerons l'intérêt des informations apportées par le CDN dans le mécanisme de routage.

## 1.1 Les réseaux de diffusion de contenus

Les réseaux de diffusion de contenus (CDNs) sont apparus comme une réponse majeure aux problèmes soulevés par la diffusion. Leur essor s'est ac-

compagné de l'émergence d'acteurs spécifiques, tels que Akamai, CloudFlare ou encore Amazon CloudFront.

Les serveurs de cache d'un CDN sont répartis géographiquement au niveau des points d'interconnexion ou POPs (*Points-of-Presence*) d'Internet. Les contenus dont le CDN doit assurer la livraison sont répliqués sur les différents serveurs. Les requêtes des utilisateurs sont alors redirigées vers le cache le plus approprié, souvent choisi pour sa proximité.

Les avantages à la mise en place d'une telle architecture pour la livraison des contenus sont nombreux. Tout d'abord, l'utilisateur accédant à un cache proche de lui, le délai s'en trouve réduit, ainsi que l'impact sur les réseaux de cœur. Il en va de même pour la charge subie par le fournisseur de contenus, celui-ci n'ayant plus qu'à fournir le contenu les quelques fois nécessaires à sa mise en cache. Les CDN garantissent également une grande fiabilité et résilience, leur conception distribuée permettant de rediriger de manière transparente l'utilisateur vers un cache fonctionnel en cas de problème.

La gestion d'un CDN nécessite la mise en œuvre d'un grand nombre de techniques dont résultent les performances globales du réseau. Plusieurs thématiques de recherche se distinguent telles que le choix de redirection des utilisateurs, la politique d'éviction des contenus, leur prépositionnement. . . Ces thématiques ont donné lieu à de très nombreux travaux et ne seront pas traités dans cette thèse.

**Les TelCo CDNs** Les CDNs apportent une réponse à la diffusion de contenus pour les fournisseurs de contenus et pour les réseaux de cœur. Cependant, localisés au niveau des POPs, leurs caches restent éloignés des utilisateurs et n'atteignent pas les réseaux d'accès. Ces derniers ne bénéficient pas des mécanismes de mise en cache, ni du contrôle qu'un CDN peut avoir sur la livraison des contenus.

Ainsi, les fournisseurs d'accès Internet (FAI) se sont intéressés au déploiement de serveurs de cache au sein de leur propre réseau, que l'on appelle alors TelCo CDN. En installant des caches dans les stations de base LTE (« *edge caching* ») ou dans les nœuds de raccordement cuivrés ou optiques (NRA/NRO), ceux-ci sont alors bien plus proches des utilisateurs. Les FAI sont alors en mesure d'optimiser la livraison des contenus à l'intérieur même de leur réseau.

**L'interconnexion de CDNs** La diffusion des contenus nécessite idéalement l'interconnexion des opérateurs CDN avec les TelCo CDNs, permettant ainsi la délégation de la diffusion aux TelCo CDNs. Ces derniers sont en effet plus à même de mener à bien cette tâche en raison de la connaissance de leur réseau. Cette coopération entre TelCo CDN et opérateur CDN est formalisée par un contrat de livraison de contenus et est matérialisée par l'utilisation de protocoles d'interconnexion spécifiques.

Dans le but de faciliter la coopération entre les CDNs, le groupe de travail *Content Delivery Network Interconnection* (CDNI) au sein de l'IETF (*Internet Engineering Task Force*) œuvre à la standardisation de protocoles d'interconnexion. On retrouve ainsi dans ce groupe de travail des FAI tels qu'Orange et des opérateurs CDN tels que Velocix. Ces protocoles prévoient une extension des fonctionnalités d'un CDN, avec des mécanismes de redirection des requêtes vers un CDN partenaire, des méthodes d'échange d'informations comme la couverture géographique ou les fonctionnalités prises en charge par les serveurs de cache ou encore l'échange d'informations sur les contenus telles que le type ou la taille. Le CDN en aval a alors plusieurs cartes en main pour mettre en place une stratégie de livraison optimisée pour son réseau.

Le fonctionnement d'un CDN et des mécanismes d'interconnexion CDNI sera présenté dans le chapitre 2. Dans la suite, nous nous intéresserons à l'optimisation de la livraison de contenus par un TelCo CDN sur un réseau hybride satellite / terrestre.

## 1.2 Les réseaux hybrides satellite / terrestre

### 1.2.1 Zones mal desservies

L'accroissement des services numériques s'accompagne d'une modernisation des réseaux d'accès. La fibre optique fait l'objet d'un déploiement continu afin de remplacer les réseaux cuivrés et câblés. Pourtant un grand nombre de zones restent mal desservies ou le deviennent, avec des débits qui n'évoluent pas alors que les besoins augmentent. Ainsi, certaines zones stagnent avec des débits inférieurs à 1 ou 2Mbit/s alors que l'Union Européenne fixe un objectif d'accès à 30Mbit/s pour tous à l'horizon 2020 [72], traduit en France par le Plan Très Haut Débit [120].

Ce constat est dû essentiellement à des raisons économiques. En effet, le déploiement de nouveaux réseaux répondant aux nouveaux besoins est coûteux. Ainsi, les opérateurs favorisent les zones urbaines, moins étendues et plus densément peuplées, réduisant d'une part les coûts de cette évolution et d'autre part garantissant un retour sur investissement rapide en raison du nombre élevé de clients.

Les zones rurales, moins attractives, se retrouvent délaissées par les opérateurs, alors même qu'elles sont les plus en retard. De plus, elles bénéficient difficilement des évolutions de l'ADSL telles que l'ADSL2 ou le VDSL en raison des contraintes sur la longueur des lignes pour leur utilisation. Ainsi, seules 60% [7] des lignes en France y sont éligibles. D'autre part, ces évolutions non seulement profitent aux usagers les plus proches des DSLAMs (*Digital Subscriber Line Access Multiplexers*) et donc déjà les mieux desservis, mais d'autre part brouillent et dégradent les performances de l'ADSL classique encore utilisé par les usagers les plus éloignés [55].

Face à cet accroissement de la fracture numérique, les zones grises apparaissent comme les premières nécessitant une amélioration de la qualité de service de leurs réseaux.

### 1.2.2 Internet par satellite

Les solutions terrestres montrant leurs limites dans les zones rurales pour les raisons évoquées précédemment, les usagers de ces zones se tournent alors vers des solutions sans-fil, notamment vers l'Internet par satellite. Cette solution offre en effet un accès à haut débit pour un coût d'installation modeste limité à l'installation d'une parabole de réception. L'intérêt pour ce moyen d'accès est tel que les opérateurs sont dans la nécessité de refuser par endroit les nouveaux clients [68] [108]. Le développement des accès Internet haut débit par satellite est intégré au plan d'aménagement numérique des territoires en France et a donné lieu au financement de plusieurs projets tels que le projet « THD-Sat » [141].

### 1.2.3 Les limitations de l'accès à Internet par satellite

Les offres d'accès Internet par satellite pour le grand public reposent très majoritairement sur les satellites géostationnaires. Le service rendu par ce type d'accès se caractérise par une latence élevée due au temps de propagation du signal entre la terre et l'orbite géostationnaire ainsi qu'aux larges tampons nécessaires au fonctionnement du système. Cette forte latence a un impact important sur la qualité des services. En effet, beaucoup de protocoles et applications sont très sensibles au délai. Notamment, le protocole TCP (*Transmission Control Protocol*), utilisé par une majorité d'applications, utilise deux mécanismes qui y sont très sensibles : la poignée de main initiale, ainsi que l'algorithme de *slow-start* au démarrage des connexions. Le protocole TLS (*Transport Layer Security*), servant à la sécurisation des échanges sur Internet, est lui aussi tributaire de la latence en raison de son utilisation d'une poignée de main au début des échanges.

Afin de faire face à ces problèmes, des systèmes spécifiques tels que les PEPs ont été déployés. L'utilisation de l'ACM (*Adaptive Coding and Modulation*) permet de réduire le taux d'erreur binaire afin de limiter les retransmissions dues aux pertes, coûteuses en délai. Ces mécanismes amoindrissent les effets de la latence, sans pour autant les supprimer.

Des constellations LEO (*Low Earth Orbit*) et MEO (*Medium Earth Orbit*) de satellites de communication telles qu'Iridium ou O3b ont également fait leur apparition. L'altitude plus réduite des satellites permet d'offrir un service avec un moindre délai. Toutefois, les satellites de communications géostationnaires conservent un avenir certain. En effet, certaines constellations sont plus contraintes que les satellites géostationnaires dans les latitudes qu'elles permettent d'atteindre. D'autre part, les satellites géostation-



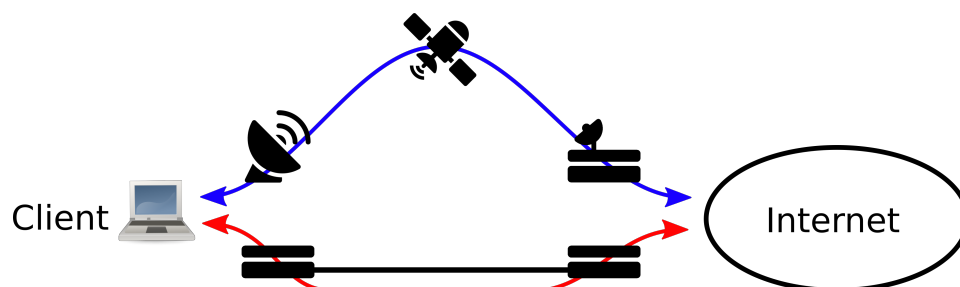


FIGURE 1.1 – Exemple de réseau hybride faisant intervenir un lien terrestre (e.g. ADSL) et un lien satellite.

naires permettent l'utilisation d'une parabole VSAT (*Very Small Aperture Terminal*) fixe alors que les constellations de satellites nécessitent des systèmes plus coûteux avec de multiples paraboles mobiles. De plus, l'utilisation de multiples faisceaux permettant la réutilisation de fréquences ainsi que l'utilisation de la bande de fréquences  $K_a$  a permis une colossale augmentation des débits fournis par les satellites de communication géostationnaires. Enfin, les systèmes existants ont su trouver un public important, la technologie est mieux maîtrisée et de nombreux projets de satellites HTS (*High-Throughput Satellite*) continuent de voir le jour. Ainsi, nous nous intéresserons, dans cette thèse, aux accès à Internet par satellite géostationnaire.

#### 1.2.4 Les réseaux hybrides satellite / terrestre

Contrairement aux réseaux satellitaires, les réseaux terrestres ne connaissent pas un tel temps de propagation. Il devient alors intéressant d'utiliser conjointement les deux réseaux d'accès, par satellite et terrestre, afin de tirer profit de leurs qualités respectives, débit élevé ou faible temps de propagation.

Certains travaux se sont intéressés par le passé à l'utilisation de liens terrestres et satellites unidirectionnels (UDL ou *Unidirectional Link*) [40]. Cette approche est aujourd'hui délaissée avec l'hégémonie des systèmes satellitaires bidirectionnels. D'autre part, l'étude d'une hybridation terrestre / satellite (bidirectionnel) dans un contexte d'interconnexion CDN est, à notre connaissance, une approche nouvelle. Ainsi, nous nous intéresserons à l'optimisation de la diffusion des contenus sur un tel réseau hybride.

#### 1.2.5 Le routage multichemin

L'utilisation conjointe de plusieurs réseaux n'est pas nouvelle et a été longuement étudiée par le passé. De nombreux protocoles et algorithmes existent pour utiliser simultanément plusieurs chemins. Cependant, l'utilisation conjointe de deux réseaux, satellite et terrestre, de technologies différentes et aux propriétés radicalement opposées est très spécifique.

Les différents réseaux d'accès envisagés sont opérés par des entités distinctes et se terminent en des lieux différents, respectivement dans les DSLAMs et dans les passerelles vers le satellite. Certains travaux étudient l'hybridation au niveau des couches physiques ou liaison de données des réseaux par satellite avec les réseaux terrestres 5G [8] [71]. Ces propositions nécessitent une évolution des systèmes existants, or il s'agit bien de leur stagnation qui nous a orientés vers les réseaux hybrides. Ainsi, nous nous intéressons aux méthodes permettant l'exploitation des systèmes dans leur état actuel. Finalement, les réseaux d'accès considérés proposant tous un service IP (*Internet Protocol*), nous étudierons la mise en place d'un mécanisme de routage multichemin au niveau 3 ou supérieur.

Les méthodes de routage classiques sont généralement conçues pour des réseaux homogènes, aux caractéristiques similaires, et s'avèrent inefficaces voir inapplicables sur les réseaux hétérogènes, dont les différents chemins possèdent de grands écarts de latence, débit et / ou taux de perte comme nous le verrons au chapitre 3. Les solutions fonctionnelles sont plus complexes en raison de l'ajout de mécanismes spécifiques au cas hétérogène, les rendant difficiles à mettre en œuvre. Un problème majeur est celui du déséquencement des paquets, causé par les systèmes d'agrégation des couches basses jusqu'à la couche IP. Il est interprété comme de la congestion par le protocole TCP qui diminue alors son débit d'émission, réduisant les performances. Ce problème peut toutefois être corrigé, au prix d'une complexité accrue, à l'aide d'un tampon au niveau du récepteur permettant le reséquencement des paquets. Le protocole de transport MP-TCP (*Multipath TCP*), qui met en place ce type de mécanisme, s'est toutefois distingué par sa facilité de déploiement grâce à une rétrocompatibilité avec le protocole dominant TCP.

Malgré cela, les solutions applicables restent intrinsèquement conçues pour les réseaux homogènes, les réseaux hétérogènes apparaissant uniquement comme un cas pathologique. Les objectifs en termes de métriques visées par ces mécanismes (taux d'occupation des liens, maximisation du débit. . .) ne sont pas appropriés à des chemins disparates.

Les techniques de communication multichemin ainsi que les problématiques associées seront présentées dans le chapitre 3.

## 1.3 Contributions

### 1.3.1 Architecture de livraison hybride satellite / terrestre

Dans cette thèse, nous nous sommes tout d'abord intéressés à la proposition d'une architecture de livraison de contenus sur réseau hybride satellite / terrestre. Celle-ci fait intervenir un fournisseur d'accès à Internet hybride, opérant son propre CDN (approche TelCo CDN) et sera décrite dans le chapitre 4. Elle permet à ce dernier de mettre en place des mécanismes

spécifiques pour optimiser la livraison des contenus sur ce réseau hybride.

### 1.3.2 Le routage fondé sur la taille

Partant du constat que les contenus courts sont majoritairement contraints en délai (contenus html/css, requêtes interactives. . .) alors que les contenus longs profitent avantageusement d'un meilleur débit, la taille des contenus apparaît comme un critère déterminant pour la mise en place d'un routage efficace.

La taille des contenus est une information relevant de la couche applicative et n'est donc usuellement pas connue par les mécanismes de routage. Cependant, notre contexte d'interconnexion nous permet d'acquérir diverses informations sur les contenus envoyés. Ces informations, telles que la taille des contenus ou le type MIME, peuvent alors être utilisées dans le processus de routage.

### 1.3.3 MinFCT

L'utilisation de la taille a déjà été suggérée à travers un mécanisme de routage fondé sur l'utilisation d'un seuil fixe [33]. Cependant, cette approche à un inconvénient majeur résidant dans le paramétrage de ce seuil. D'autre part, elle est limitée à l'utilisation de deux liens et ne s'adapte pas aux évolutions des conditions du réseau.

Pour surpasser ces contraintes, nous avons conçu un nouveau mécanisme de routage, MinFCT, fondé sur le temps de téléchargement des flux ou FCT (*Flow Completion Time*). Ce mécanisme consiste, pour chaque flux, à estimer son temps de téléchargement sur chaque chemin disponible; le flux est alors routé par le chemin minimisant cette estimation. L'estimation du temps de téléchargement repose sur la connaissance de la taille du flux ainsi que sur les caractéristiques des chemins considérés (latence, capacité, taux de perte).

Ce mécanisme sera présenté et évalué dans le chapitre 5 et a donné lieu à la publication suivante : E. Bouttier, R. Dhaou, F. Arnal, C. Baudoin, E. Dubois, and A. L. Beylot, Analysis of content size based routing schemes in hybrid satellite & terrestrial networks, *IEEE Global Communications Conference (GLOBECOM)*, Dec 2016 [16]

### 1.3.4 Le routage par flux et par paquet

Lors de l'utilisation d'un réseau multichemin, deux stratégies différentes s'opposent traditionnellement. D'une part, le routage à l'échelle des flux consiste à prendre la même décision de routage pour l'ensemble des paquets d'un même flux. D'autre part, le routage à l'échelle des paquets permet de prendre une décision plus fine, différente pour chaque paquet, ouvrant la

porte par exemple à l'agrégation du débit de plusieurs liens pour un même flux.

Afin d'améliorer notre première proposition de routage par flux, nous avons alors étudié l'intérêt d'un routage par paquet dans notre contexte hétérogène avec connaissance de la taille. Cette approche sera traitée dans le chapitre 6 et s'est concrétisée par la publication suivante : (mêmes auteurs) Heterogeneous multipath networks : Flow vs. packet based routing in a size-aware context, *IEEE Global Communications Conference (GLOBECOM)*, Dec 2017 [17].

### 1.3.5 Comparaison avec MP-TCP

Pour terminer cette thèse, nous nous sommes focalisés sur la comparaison entre notre solution paramétrée pour TCP et la solution très en vogue MP-TCP. Cette comparaison nous a permis de confirmer l'intérêt de la connaissance de la taille des flux par le mécanisme de routage, ainsi que l'inefficacité de l'agrégation en milieu hétérogène.

Cette étude prendra place dans le chapitre 7 et s'est traduite par la publication suivante : (mêmes auteurs) Improving content delivery with size-aware routing in hybrid satellite / terrestrial networks, *IEEE International Conference on Communications (ICC)*, Mai 2018 (à paraître) [18].



## Chapitre 2

# Réseau de diffusion de contenus

### 2.1 Les réseaux de diffusion de contenus

#### 2.1.1 Introduction

Internet connaît une croissance exponentielle qui se traduit par une explosion du nombre d'utilisateurs, du nombre de requêtes et des quantités de trafic échangé [29]. Les architectures client/serveur traditionnelles ne sont plus capables de faire face aux nouveaux besoins et se retrouvent saturées. La réplication des contenus sur de multiples serveurs nommés caches répartis en de nombreux emplacements est une technique très couramment utilisée pour faire face aux défis posés par la livraison des contenus. En redirigeant les requêtes des utilisateurs vers le cache le plus proche, ces nouvelles architectures distribuées permettent de réduire l'impact sur les réseaux et sur les serveurs de contenus tout en réduisant le temps nécessaire à la satisfaction des demandes des utilisateurs.

Ce nouveau paradigme pour la distribution des contenus s'est traduit par la création de réseaux dédiés à cette tâche, nommés réseaux de diffusion de contenus ou CDNs (*Content Delivery Networks*). Ces réseaux viennent s'insérer entre les fournisseurs de contenus et les utilisateurs; la livraison des contenus leur étant déléguée. La spécificité de cette tâche a conduit à l'apparition d'acteurs nouveaux que sont les opérateurs de CDN. Ils agissent comme intermédiaires entre les fournisseurs de contenus, qui produisent ces derniers, et les utilisateurs finaux qui les consomment. Les fournisseurs de contenus ont particulièrement recours aux CDNs pour les contenus statiques tels que les images, les vidéos ou les scripts en javascript.

Plusieurs travaux décrivent le fonctionnement d'un CDN et présentent les différentes architectures mises en œuvre [111] [112]. Dans ce chapitre, nous nous restreindrons à une présentation générale du fonctionnement d'un

CDN. Nous exposerons les principales difficultés opérationnelles et présenterons quelques solutions majeures pour les traiter.

### 2.1.2 Principe de fonctionnement d'un CDN

Un CDN est composé d'un ensemble de serveurs disposant d'un espace de stockage important permettant la mise en cache de contenus. Un mécanisme de routage redirige les requêtes des utilisateurs vers le cache le plus approprié, le contenu étant alors servi par ce dernier. Le réseau de distribution possède également une infrastructure de gestion et de journalisation. L'infrastructure de gestion permet de maintenir la cohérence des contenus stockés dans les caches tandis que l'infrastructure de journalisation permet de facturer la distribution aux fournisseurs de contenus ainsi que de leur fournir des statistiques.

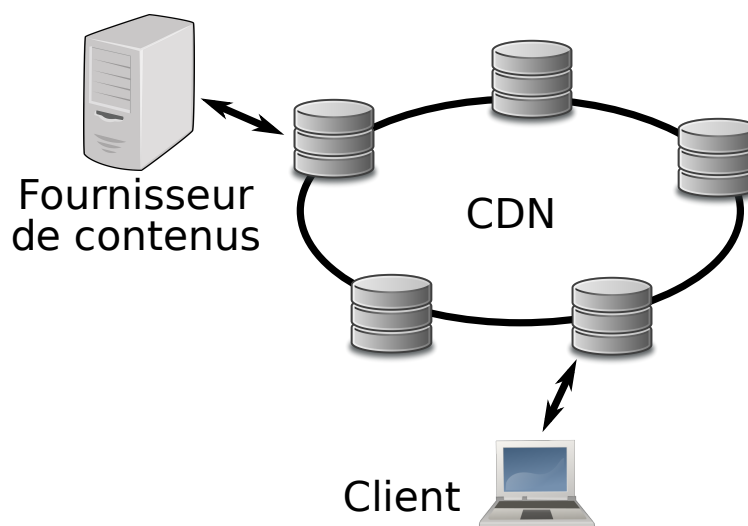


FIGURE 2.1 – Récupération d'un contenu par un client depuis un CDN.

### 2.1.3 Avantages à l'utilisation d'un CDN

Les avantages à l'utilisation d'un CDN sont nombreux. Tout d'abord, les CDNs permettent aux fournisseurs de contenus de déléguer une tâche compliquée qui ne constitue pas leur cœur de métier. Ils permettent également d'absorber les pics de trafic, qu'ils soient prévus (par exemple en soirée) ou imprévus (par exemple la brusque popularité d'un site web ou un déni de service distribué (DDoS)). En effet, l'infrastructure d'un CDN est dimensionnée pour servir les contenus de nombreux fournisseurs ; le surdimensionnement nécessaire pour absorber les pics de trafic est mutualisé et de fait bien moindre au regard de la taille du CDN. L'étendue géographique

d'un CDN permet aux utilisateurs de récupérer des contenus depuis un cache plus proche et donc avec une latence réduite. Les coûts de distribution sont également réduits en raison d'une diminution de l'utilisation du réseau.

#### 2.1.4 Problématiques opérationnelles de gestion d'un CDN

La gestion d'un CDN vise à optimiser la diffusion des contenus afin d'assurer la meilleure qualité de service pour les utilisateurs. Pour cela, l'opérateur dispose d'un certain nombre de leviers. Nous en présentons ici certains ainsi que quelques solutions majeures.

**Gestion des caches** Lorsqu'un utilisateur est redirigé vers un cache, le contenu qu'il demande peut être absent, on parle alors de défaut de cache (*cache miss*). Dans ce cas, le cache peut décider de rediriger l'utilisateur vers un autre cache disposant *a priori* du contenu. Il peut également décider d'acquérir le contenu depuis un autre cache du CDN ou directement depuis les serveurs du fournisseur de contenus.

Lorsqu'un nouveau contenu est mis en cache, il est nécessaire de supprimer un ou plusieurs contenus afin de libérer la place nécessaire au nouvel arrivant. Plusieurs algorithmes existent pour décider des objets à supprimer en priorité. L'objectif majeur de ces algorithmes est de minimiser les défauts de cache.

L'algorithme *First-In First-Out* (FIFO) consiste à supprimer, parmi les contenus présents, le plus ancien. *Least Recently Used* (LRU) supprime le contenu auquel on n'a plus accédé depuis le plus longtemps, favorisant le maintien dans le cache des contenus populaires. Dans le même ordre d'idée, l'algorithme *Least Frequently Used* (LFU) calcule pour chaque contenu sa fréquence d'utilisation, afin de supprimer le moins populaire. Ce dernier algorithme est influencé par la méthode avec laquelle la fréquence est estimée, et plus particulièrement par l'échelle de temps sur laquelle la fréquence est estimée. Les contenus ont en effet souvent une popularité passagère sur diverses échelles de temps. Outre la complexité de l'estimation de la popularité des contenus [30], et notamment de la prédiction de leur popularité future [89], d'autres algorithmes tiennent compte par exemple de la taille des contenus [25]. En effet, il peut être plus intéressant pour un opérateur de garder en cache de nombreux petits contenus à la popularité relative plutôt qu'un unique gros contenu même s'il s'avère plus populaire. Il s'agit d'une optimisation conjointe entre minimisation des défauts de cache et minimisation des échanges réseau nécessaires au rapatriement de gros contenus plus éloignés.

**Placement des contenus** Le placement des contenus est un problème auquel beaucoup de travaux se sont intéressés [74] [149]. En effet, l'espace disque est coûteux, et ce d'autant plus qu'il est proche des utilisateurs. D'autre part, de nombreux contenus sont très populaires et nécessitent d'être



répliqués un grand nombre de fois afin d'éviter la saturation des caches. Ainsi, il s'agit d'un problème d'optimisation conjointe entre répllication et placement des contenus.

Les contenus peuvent être mis en cache lors d'un défaut de cache, mais aussi de façon proactive [140]. Ceci permet notamment d'éviter les défauts de cache liés à la première demande des contenus. Cette approche permet également de profiter de périodes où les coûts d'utilisation du réseau sont réduits (par exemple la nuit) pour mettre en cache les contenus. D'autre part, il peut être décidé d'appliquer un placement statique des contenus : les utilisateurs sont toujours redirigés vers un cache qui possède le contenu. On supprime ainsi les défauts de cache tout en optimisant de manière globale le placement des réplias ; la politique de placement est régulièrement mise-à-jour pour s'adapter à l'évolution de la popularité des contenus.

**Algorithme de redirection** L'algorithme de redirection doit choisir le cache le plus approprié vers lequel rediriger un utilisateur. Cette décision est cruciale pour assurer le fonctionnement optimal du CDN.

La proximité est un critère prédominant dans la réalisation de ce choix. En effet, un cache proche permet de réduire le temps nécessaire à un utilisateur pour récupérer un contenu ainsi que le nombre de réseaux que le contenu devra traverser. Cela nécessite de localiser l'utilisateur, ce qui est réalisé majoritairement grâce à l'adresse IP de ce dernier. En effet, des bases de données dites « géoip » associent aux adresses IP leur position géographique approximative (pays, voire ville). L'adresse IP permet également d'identifier l'AS dans lequel se situe l'utilisateur, permettant d'identifier le chemin entre le client et un cache au sens topologique du réseau.

La proximité n'est cependant pas le seul critère pris en compte. La présence du contenu dans les caches peut également être intégrée à l'algorithme de redirection [109]. On pourra alors préférer rediriger un utilisateur vers un cache un peu plus éloigné mais ayant déjà le contenu en cache ou au contraire rediriger l'utilisateur vers un cache proche qui ne dispose pas encore du contenu, misant sur l'utilité de sa mise en cache pour de futures requêtes. La charge des serveurs de cache ou des réseaux les connectant est également un facteur intéressant à optimiser pour garantir une bonne qualité de service aux utilisateurs [155]. La célèbre méthode « *Power of Two Choices* » [99] consistant à choisir le serveur le moins chargé parmi deux tirés au hasard est efficace et facile à implanter.

Les algorithmes les plus avancés sont couplés à la gestion des caches, optimisant le placement des contenus dans les caches, de manière globale, et en tenant compte du maximum de critères [6].

**Méthode de routage des requêtes** Les utilisateurs sont redirigés vers le cache le plus approprié pour récupérer leurs contenus. Pour procéder à

cette redirection, plusieurs méthodes existent, trois méthodes principales étant fondées sur l'utilisation des protocoles DNS (*Domain Name System*) et HTTP (*HyperText Transfert Protocol*) et l'utilisation d'adresses IP *anycast* [148].

La redirection fondée sur l'utilisation du DNS consiste à héberger les contenus à servir dans un domaine précis, par exemple `media.acme.org`, dont la gestion sera déléguée aux serveurs DNS de l'opérateur CDN. Cette délégation peut se faire soit directement par l'utilisation de champs `NS`, soit indirectement par l'utilisation d'un champ `CNAME` pointant sur un domaine contrôlé par l'opérateur, par exemple `acme.cdn.org`. Lorsqu'un utilisateur accède à un contenu, il procède tout d'abord à la résolution du domaine `media.acme.org`. Les serveurs DNS de l'opérateur CDN lui répondent alors avec une ou plusieurs adresses IP de caches susceptibles de répondre à sa requête. Cette méthode possède quelques limitations. Tout d'abord, l'opérateur CDN a pour seule information pour localiser l'utilisateur l'adresse IP du serveur DNS récursif utilisé par ce dernier, or celui-ci peut être éloigné du client conduisant à une mauvaise décision de redirection. Il ne connaît pas non plus le contenu exact auquel l'utilisateur accède. D'autre part, le DNS possède un système de mise en cache des réponses. Ainsi, la décision de redirection sera conservée par le serveur récursif pour une durée égale au TTL (*Time-To-Live*). La décision pourra également être appliquée à d'autres clients utilisant le même serveur DNS récursif sans intervention de l'opérateur CDN.

La redirection fondée sur le protocole HTTP permet des décisions plus fines. Dans ce cas, le domaine `media.acme.org` pointe vers un (ou plusieurs) serveur(s) web de l'opérateur CDN. À la réception de la requête du client, le serveur répond avec un code HTTP 307 ou 308 indiquant ainsi une redirection temporaire ou permanente. L'utilisateur va alors émettre une nouvelle requête vers l'URI (*Uniform Resource Indicator*) fournie via l'en-tête `Location`. Lors du choix du cache vers lequel rediriger la requête, l'opérateur CDN dispose de plusieurs informations : l'adresse IP exacte du client permettant sa géolocalisation, son agent utilisateur permettant de connaître le type de terminal que celui-ci utilise ainsi que l'URI complète du contenu permettant son identification formelle. Cette méthode de redirection a cependant l'inconvénient d'être plus lente que la redirection DNS, en raison de la nécessité d'établir une connexion HTTP avec un serveur potentiellement distant.

L'utilisation d'adresse IP *anycast* consiste à annoncer avec le protocole BGP la même adresse IP depuis plusieurs endroits. Cette même adresse IP, pointée par le domaine `media.acme.org`, est alors utilisée par plusieurs caches distincts. Les tables de routage des routeurs de l'Internet sont mises-à-jour par BGP pour aiguiller les requêtes vers le cache le plus proche. La proximité est ici déterminée par la métrique BGP, celle-ci étant fondée sur la longueur de l'AS-Path, c'est-à-dire le nombre de Systèmes Autonomes

traversés pour atteindre la cible. Cette méthode est relativement simple à mettre en œuvre puisque fondée sur l'utilisation du protocole BGP. Bien que la méthode de routage soit restreinte, elle offre de bonnes performances [21].

**Placement des caches** Outre la question du placement des objets se pose celle du placement des caches. En effet, il s'agit d'une ressource limitée par la capacité d'investissement d'un opérateur CDN. Leur placement est alors primordial, celui-ci ayant un impact direct sur les possibilités offertes aux algorithmes de placement des contenus et de redirection des utilisateurs.

Les stratégies visent alors principalement à minimiser la distance des utilisateurs au cache le plus proche. Ainsi, cela s'apparente à un partitionnement en  $K$ -moyennes, problème combinatoire pouvant être traité à l'aide d'heuristiques. Notons que la distance peut être entendue de différentes manières telles que le délai [139] ou le nombre de sauts [86]. Des variantes intègrent d'autres critères d'optimisation tels que la charge [123].

**Structuration d'un CDN** Afin de faciliter sa gestion ainsi qu'à des fins de performance, un CDN est souvent organisé de manière hiérarchique. Les caches peuvent alors être agencés sous forme arborescente, avec des caches intermédiaires situés avant les feuilles. Lorsqu'une feuille rencontre un défaut de cache, celle-ci va alors récupérer le contenu demandé depuis un autre cache. Plusieurs stratégies existent pour sélectionner ce cache [35].

Avec une approche hiérarchique, les caches s'adressent à leur cache parent, qui lui-même s'adresse au fournisseur de contenus en cas de défaut. Une autre stratégie peut consister à s'adresser aux autres caches voisins du cache pris en défaut. On évite ainsi une potentielle requête au fournisseur de contenus. Une dernière stratégie consiste à fédérer les caches voisins ; dans ce cas un cache pris en défaut va rediriger l'utilisateur vers son voisin en possession du contenu au lieu de le récupérer lui-même afin d'éviter une duplication des contenus dans des caches géographiquement proches. Il peut être toutefois décidé de répliquer un contenu très populaire si un cache unique n'est pas suffisant pour répondre aux nombreuses sollicitations.

### 2.1.5 Les TelCo CDN

Les CDN sont principalement opérés par des entités spécifiques dont ils constituent l'activité principale. Leurs caches sont placés bien souvent au niveau des points de présence ou POPs de l'Internet, éloignés des utilisateurs. Les fournisseurs d'accès Internet se sont intéressés au déploiement des CDNs dont les nœuds se situent plus près des usagers afin d'améliorer la qualité de service de leurs abonnés et de profiter des avantages de la mise en cache, faisant ainsi des économies sur les coûts d'interconnexion de leur réseau. On parle alors de TelCo CDN.

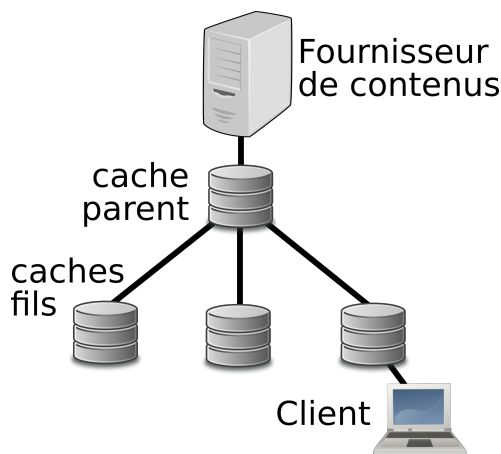


FIGURE 2.2 – Exemple d'architecture CDN hiérarchisée.

Les TelCo CDN possèdent l'avantage d'avoir la main sur leur réseau d'accès contrairement aux opérateurs CDN traditionnels. Cela leur permet de déployer des caches proches des utilisateurs, par exemple dans les nœuds de raccordement d'abonnés ou optique (NRA / NRO) ou dans les stations de base des réseaux de téléphonie mobile. Par une connaissance approfondie du réseau d'accès, un TelCo CDN est capable d'optimiser la diffusion des contenus.

La souscription pour un fournisseur de contenus d'un contrat de diffusion avec un TelCo CDN lui permet d'offrir un meilleur service à ses usagers situés sur ce réseau d'accès, et malheureusement uniquement sur celui-ci. Il s'agit là d'une limitation des TelCo CDN liée à leur portée.

## 2.2 L'interconnexion de CDN

Nous avons présenté dans la partie précédente l'intérêt des réseaux de diffusion de contenus. De nos jours, leur déploiement est massif, incluant les fournisseurs d'accès Internet qui déploient leur propre CDN dans les réseaux d'accès.

Les CDNs sont donc nombreux, avec des caractéristiques, notamment géographiques, distinctes. Par exemple, un CDN peut être restreint à un continent, amenant à une insatisfaction des utilisateurs situés sur d'autres continents. Pour pallier ce problème, un fournisseur de contenus peut signer un contrat de diffusion avec d'autres CDNs présents en d'autres lieux. Cependant, cette approche est limitée : chaque fournisseur de contenus doit souscrire potentiellement un grand nombre de contrats de diffusion pour obtenir un service correct. C'est ici qu'apparaît l'intérêt de l'interconnexion des CDN, afin de permettre à un CDN de déléguer la diffusion d'un contenu à

un autre CDN mieux placé. Le fournisseur de contenus n'a ainsi plus qu'un unique interlocuteur, le CDN en amont, qui aura un contrat de diffusion avec le CDN en aval.

Outre l'extension de l'empreinte géographique, l'interconnexion de CDN présente d'autres cas d'utilisation. Ainsi, les TelCo CDN sont particulièrement intéressés par cette approche : en signant un contrat de diffusion avec un opérateur CDN, il sera alors capable d'optimiser la distribution des contenus sur son réseau et de réduire le trafic échangé avec le CDN. L'interconnexion est également utile en cas d'augmentation temporaire de la charge portant sur un CDN, par exemple liée à la diffusion d'un événement très populaire [150]. Il pourra alors faire momentanément appel aux ressources d'un autre CDN, évitant ainsi de devoir surdimensionner sa propre infrastructure pour gérer ce type de situation. D'autres cas peuvent amener un CDN à faire appel à un autre CDN, comme une attaque par déni de service, une panne, ou encore la demande par le client d'une fonctionnalité qu'il ne prend pas en charge (par exemple diffuser les contenus avec MP-TCP ou en IPv6).

Procéder à une interconnexion demande la mise en place d'interfaces d'échange entre les deux CDNs. Ceci nécessite des spécifications puis leur implantation ce qui en alourdit la mise en place. Afin de simplifier le procédé, l'IETF s'est doté du groupe de travail CDNI qui vise à publier des standards ouverts sur les interfaces à adopter pour s'interconnecter. Ainsi, la signature d'un contrat de distribution de contenus et l'échange de quelques informations techniques sont suffisants pour mettre en place une telle coopération.

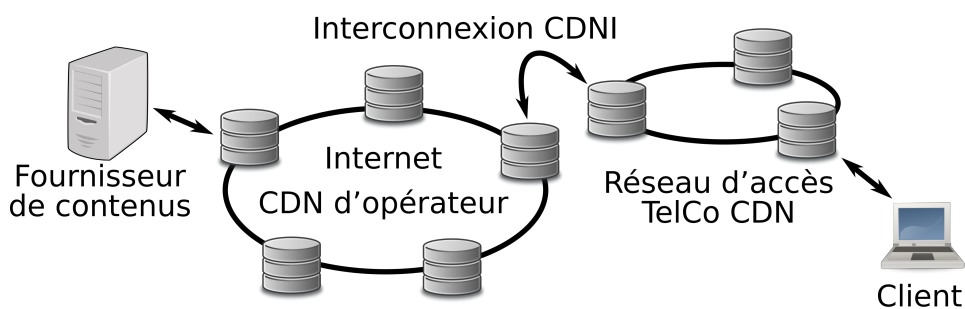


FIGURE 2.3 – Interconnexion d'un TelCo CDN avec un opérateur CDN en charge de la diffusion des contenus d'un fournisseur. Les clients du fournisseur d'accès accèdent ainsi aux contenus depuis un cache plus proche.

Le groupe de travail CDNI a ainsi standardisé quatre interfaces :

- Une interface de routage permettant de rediriger les requêtes des utilisateurs vers le CDN le plus à même d'y répondre. Cette interface est séparée en deux :
  - Une interface permettant d'obtenir les caractéristiques et capacités d'un CDN ;

- Une interface permettant la redirection des requêtes à proprement dit ;
- Une interface de journalisation permettant de faire remonter les statistiques de diffusion, notamment à des fins de facturation ;
- Une interface de méta-données permettant d'échanger des informations techniques liées à la diffusion des contenus telles que, par exemple, une zone géographique de diffusion limitée, mais l'interface peut être étendue pour transmettre n'importe quelle information ;
- Une interface de contrôle, permettant de contrôler l'interconnexion : récupération des points d'accès aux autres interfaces, lancement de la coopération, prépositionnement et éviction des contenus. . .

Toutes ces interfaces prévoient l'échange de messages normalisés au format JSON (*JavaScript Object Notation*) sur le protocole HTTP. Dans la suite de cette partie, nous détaillerons le fonctionnement des interfaces dont le rôle nous semble le plus important dans le cadre de cette thèse.

### 2.2.1 Routage des requêtes

Le routage des requêtes des utilisateurs vers le CDN le plus approprié et *a fortiori* vers le cache le plus à-même d'y répondre constitue la tâche principale de l'interconnexion. Il a été choisi de diviser en deux l'interface en charge de cette tâche, avec d'une part l'obtention des informations nécessaires à la prise de décision de routage d'une requête et d'autre part le mécanisme de redirection effectif.

#### Interface d'annonce des capacités

Cette première composante de l'interface de routage est normalisée dans la RFC 8008 [130] et est appelée FCS (*Footprint and Capabilities Semantics*). Son but est de permettre aux CDNs d'échanger les informations nécessaires au routage des requêtes.

Notamment, elle permet à un CDN d'informer de sa couverture limitée et de ses capacités de diffusion restreintes à un groupe précis d'utilisateurs. Alors qu'un opérateur CDN indiquera être capable de servir n'importe quel utilisateur, un TelCo CDN pourra utiliser cette interface pour informer ne servir que les utilisateurs de son AS ou d'une certaine plage d'adresses IP.

D'autres informations peuvent être transmises : les protocoles pris en charge par le CDN (par exemple, est-il capable de servir un utilisateur avec le protocole RTMP ?), le mode de redirection souhaité (DNS ou HTTP), auprès de qui les contenus doivent être acquis (fournisseur de contenus ou CDN amont), etc. Les informations fournies doivent également permettre à un CDN en amont de choisir parmi plusieurs CDNs en aval potentiels le plus adapté à servir une requête, par exemple en se fondant sur leur charge.

### Mécanismes de redirection

Deux méthodes de redirection peuvent être envisagées, une redirection itérative ou une redirection récursive. Dans le cas d'une redirection itérative, l'utilisateur sera redirigé par un premier CDN vers un second qui le redirigera, de manière usuelle, vers le nœud final sélectionné pour servir sa requête. Cette méthode suppose une acceptation préalable du second CDN pour servir les requêtes qui sont redirigées vers lui ; cette acceptation est négociée via l'interface FCS qui permet également de définir où les utilisateurs doivent être redirigés et par quel protocole. Dans le cas d'une redirection récursive, le CDN en aval va contacter un CDN en amont afin de lui demander où l'utilisateur doit être redirigé. Ce deuxième cas nécessite la normalisation des échanges entre CDN amont et CDN aval nécessaires à la récursion. Ces échanges se font grâce à l'interface RRI (*Request Routing Interface*) normalisée dans le RFC 7975 [105]. Nous détaillons ici le fonctionnement d'une redirection récursive utilisant cette interface.

Deux protocoles sont envisagés pour rediriger l'utilisateur, le protocole DNS et le protocole HTTP. Le choix du protocole est là encore négocié par l'interface FCS. Quel que soit le protocole choisi, les mécanismes de redirection récursive restent identiques.

On suppose qu'un utilisateur cherche à accéder à un contenu hébergé par un CDN A. Il transmet alors une requête vers le CDN A. À la réception de celle-ci, le CDN A contacte le CDN B grâce à l'interface RRI en lui fournissant les informations connues sur l'utilisateur et sur la requête, celles-ci dépendant fortement du protocole considéré. Le CDN B va alors répondre en indiquant s'il accepte de satisfaire la requête, et si oui, où il souhaite que le CDN A redirige l'utilisateur. Ce procédé est schématisé sur la figure 2.4.

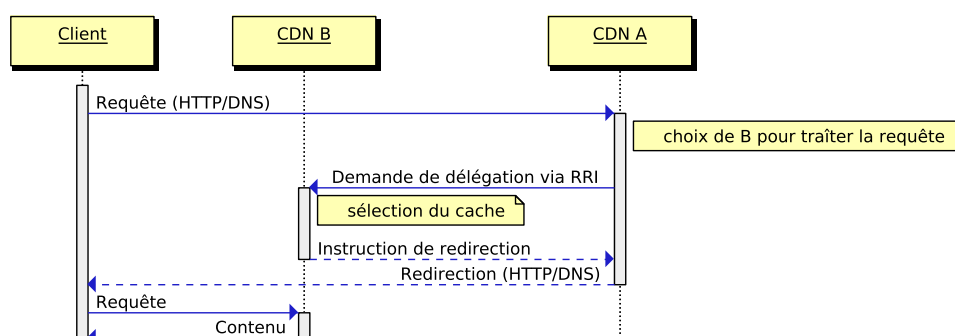


FIGURE 2.4 – Mécanisme de redirection récursif

Le CDN B peut indiquer à A si l'instruction de redirection peut être gardée en cache afin d'accélérer les redirections des futures requêtes. L'interface RRI reposant sur le protocole HTTP, B dispose de tous les mécanismes de mise en cache de ce protocole décrits dans la RFC 7234 [50] afin d'indiquer à A la validité de sa réponse. Par exemple, il peut indiquer que celle-ci est

valable une heure et s'applique à une plage d'adresses IP donnée. Un client ayant un autre préfixe d'adresse IP provoquera alors une nouvelle demande de redirection.

**Redirection DNS** Lorsque la redirection est effectuée au niveau DNS, le CDN en aval indique au CDN en amont la réponse DNS à fournir à l'utilisateur. Cette réponse peut constituer directement une ou plusieurs adresses IP correspondant au cache à contacter, ou bien il peut s'agir d'un enregistrement CNAME. Dans ce dernier cas, le serveur DNS récursif de l'utilisateur devra effectuer une nouvelle requête pour le domaine indiqué. Ce choix est donc *a priori* moins performant, mais il peut s'avérer en réalité plus efficace si cette réponse est accompagnée d'une durée de validité élevée.

La redirection fondée sur le protocole DNS dans un cadre d'interconnexion présente les mêmes avantages et inconvénients que lorsqu'elle est utilisée par un CDN pour rediriger un utilisateur vers le cache le plus approprié. En particulier, cette méthode est pénalisée par le manque d'informations sur l'utilisateur (accès uniquement à l'adresse IP de son serveur DNS récursif) et le manque d'informations sur la requête de l'utilisateur (accès uniquement au nom de domaine de la requête et non à son URI). En revanche, elle peut s'avérer plus rapide, le protocole DNS ne nécessitant pas de poignée de main et le serveur DNS récursif pouvant suivre plusieurs redirections rapidement par une position plus proche du cœur de réseau.

**Redirection HTTP** Là encore, la redirection HTTP fonctionne de manière analogue dans un cadre d'interconnexion par l'utilisation d'un code de retour signifiant une redirection (généralement 302). Le code exact et l'adresse de redirection sont fournis par le CDN aval au CDN amont par l'interface RRI. Comme précédemment, la réponse sur l'interface RRI peut être mise en cache afin d'accélérer les redirections futures.

Une redirection HTTP s'avère plus lente qu'une redirection DNS en raison du temps d'établissement de la connexion TCP, généralement sécurisée avec TLS, ce qui nécessite plusieurs échanges de bout-en-bout entre le client et le serveur du CDN amont. En revanche, les deux CDNs disposent de plus d'informations pour motiver leurs choix (choix du CDN aval et choix du nœud par le CDN aval), notamment l'adresse IP exacte du client, l'URI complète de la requête ou encore le type de terminal utilisé par le client.

### 2.2.2 Journalisation et métrologie

Le rôle d'un CDN inclut l'établissement de statistiques sur la diffusion des contenus, pour son propre usage ainsi que de celui des fournisseurs de contenus. Ces informations sont utilisées à diverses fins : facturation de la diffusion, étude économique, optimisation du réseau, etc. Le recours à un



autre CDN dépossède le premier CDN des informations portant sur le trafic délégué.

Pour pallier ce problème, une interface, nommé LI (*Logging Interface*), permettant l'échange d'informations métrologiques a été définie dans la RFC 7937 [46]. Cette interface permet également l'échange d'informations de journalisation, portant sur la diffusion des contenus mais également sur le bon fonctionnement de l'interconnexion. Ainsi, le bon déroulement de l'éviction d'un contenu pourra être remonté au CDN parent par ce biais.

L'interface LI permet d'indiquer les informations souhaitées, si celles-ci peuvent être agrégées ou encore où et comment elles doivent être remontées. La RFC normalise également le format des fichiers de logs, qu'ils concernent la diffusion ou les événements de gestion du CDN.

### 2.2.3 Méta-données

Les méta-données concernent le type ou le format d'un contenu, mais également les informations nécessaires à la diffusion de celui-ci, telles que la zone géographique de diffusion ou encore où obtenir le contenu pour sa mise en cache. Il s'agit donc d'informations nécessaires à un CDN en amont pour répondre aux sollicitations de routage via l'interface RRI ou aux requêtes directes des consommateurs des contenus.

Ces méta-données étant transmises par le CDN en amont au CDN en aval, une interface d'échange est nécessaire. Celle-ci est normalisée par la RFC 8006 [106] et nommée MI (*Metadata Interface*).

L'interface est conçue de manière à séparer la sémantique de la syntaxe du protocole. Seules certaines informations sémantiques sont standardisées, on compte notamment :

- La source du contenu, c'est-à-dire l'emplacement où doit être récupéré le contenu par le CDN en aval ;
- Les restrictions géographiques à respecter pour la diffusion d'un contenu, celles-ci pouvant s'exprimer sous la forme d'une liste de préfixes IP ou d'AS ;
- La fenêtre de temps pendant laquelle la diffusion est permise, par exemple pour spécifier la date de publication d'un épisode de série TV ;
- Les restrictions sur les protocoles de diffusion permis, par exemple pour indiquer si la diffusion en clair en HTTP est autorisée ou si l'utilisation du protocole HTTPS est rendue obligatoire ;
- Les mécanismes d'authentification que doivent respecter les clients.

Afin d'alléger la quantité de méta-données, celles-ci sont agrégées : une même information peut s'appliquer à plusieurs contenus. Ainsi, il est possible de grouper les contenus, ou d'indiquer qu'une même méta-information

s'applique à un ensemble de contenus partageant un même nom de domaine ou dont le chemin d'accès vérifie une certaine expression régulière.

Conçu de manière extensible, l'interface permet une sémantique modulaire et l'échange d'autres informations qui ne font pas l'objet de cette RFC.

#### 2.2.4 Déclenchement d'opération de contrôle

Lorsqu'un opérateur CDN fait appel à un CDN partenaire pour diffuser certains contenus, il nécessite de pouvoir demander à son partenaire l'exécution de certaines opérations de contrôle. Cette possibilité est offerte par l'interface CIT (*Control Interface / Triggers*) normalisée par la RFC 8007 [102].

Trois opérations sont offertes par cette interface au CDN parent :

**Prépositionnement** Cette opération permet d'ordonner à un CDN aval d'acquérir et de placer dans ses caches certains contenus populaires par exemple, ou que l'on souhaite être déjà disponibles à des fins de performance.

**Invalidation** L'invalidation de contenus et/ou de leurs méta-données ordonne au CDN aval de procéder à une revalidation des données. Il s'agit, lors d'une requête d'un utilisateur, de contacter le serveur d'acquisition des données afin de vérifier la validité de l'objet en cache, et à défaut de le mettre à jour.

**Purge** Cette opération ordonne au CDN aval de supprimer de ses caches certains contenus et/ou méta-données.

Afin de faciliter la gestion des opérations, l'interface prévoit notamment la possibilité de planifier des opérations. Il est également possible de suivre l'évolution d'une requête, une demande de prépositionnement par exemple pouvant être longue à traiter, ou de l'annuler.

#### 2.2.5 Gestion du trafic chiffré

La large majorité des contenus diffusés par les CDN le sont avec le protocole HTTP. Or une part croissante et déjà majoritaire du trafic HTTP est chiffrée avec le protocole TLS. Il est donc primordial pour un CDN d'être capable de gérer le trafic chiffré.

Certains fournisseurs d'accès Internet ont par le passé cherché à améliorer la qualité de service et réduire les coûts de leur réseau en mettant en place une mise en cache transparente des contenus. L'augmentation du trafic chiffré vient mettre à mal cette stratégie, renforçant d'autant plus le développement des TelCo CDN, et *a fortiori* le besoin d'interconnexion qui en résulte.

Le fonctionnement de la chaîne de confiance régissant la validité des connexions TLS nécessite, pour un serveur de cache, l'accès à une clef de chiffrement associée à un certificat valide pour le nom de domaine hébergeant le contenu. Plusieurs stratégies peuvent être mises en place pour permettre à un CDN en aval de proposer un certificat valide à un client. Ces solutions nécessitent l'échange d'informations avec le CDN en amont et sont actuellement discutées au sein du groupe de travail CDNI dans un document de travail non finalisé [48].

À noter que dans le cas d'une redirection HTTP, l'utilisateur peut être redirigé vers un domaine dont le CDN aval est maître. Ainsi, ce dernier peut diffuser les contenus avec ses propres certificats TLS. Cependant, cette approche est limitée :

- les méthodes de redirection fondées sur le DNS ne sont alors plus envisageables ;
- le fournisseur de contenus peut souhaiter que ses contenus soient servis sur son domaine et masquer l'utilisation d'un CDN à ses utilisateurs ;
- la redirection vers un autre domaine pose des problèmes d'authentification de l'utilisateur dus aux restrictions sur le partage des cookies.

### Utilisation limitée de clefs distantes

La première solution est appelée LURK (*Limited Use of Remote Keys*). Ce mécanisme a été présenté en 2014 par CloudFlare [137] [13]. Il permet à un serveur de présenter à un client un certificat TLS valide sans posséder les clefs de chiffrement nécessaires. Pour cela, il délègue à un serveur de clefs, après une nécessaire authentification, les opérations cryptographiques nécessaires à l'établissement de la session chiffrée.

Dans notre cas, le serveur de clefs peut être géré par le CDN amont ou par le fournisseur de contenus directement. Le serveur aval peut s'adresser directement au serveur de clefs, ou alors au CDN amont qui relayera la requête au serveur de clefs. Pour être mis en place, ce procédé nécessite la spécification du champ SNI (*Server Name Indicator*) du protocole TLS afin de permettre au CDN aval de connaître le serveur de clefs qu'il doit contacter.

À titre d'exemple, la figure 2.5 représente le fonctionnement de l'établissement d'une session TLS entre un client et un CDN aval. Le serveur de clefs est géré par le fournisseur de contenus et le CDN amont sert de relais jusqu'au serveur de clefs. On suppose ici l'utilisation de RSA (RIVEST, SHAMIR, ADLEMAN) mais un mécanisme analogue permet l'utilisation d'autres algorithmes cryptographiques fondés sur Diffie-Hellman.

L'intérêt de la méthode est d'éviter de partager les clefs privées du fournisseur de contenus avec son opérateur CDN et l'ensemble de ses potentiels CDNs associés.

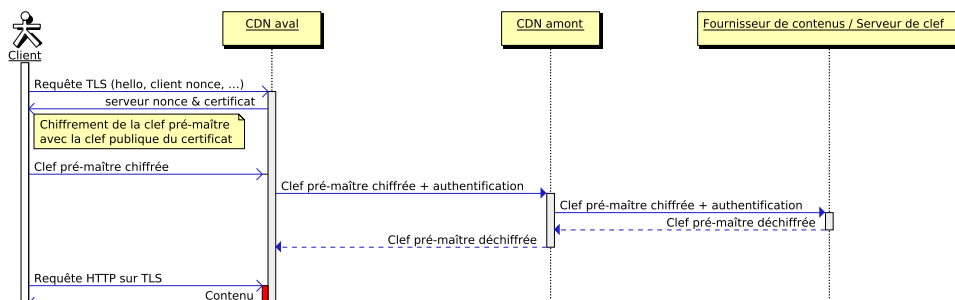


FIGURE 2.5 – Établissement d’une session TLS en faisant appel à un serveur de clef distant opéré par le fournisseur de contenus.

### Diffusion de contenu HTTPS hors-bande

Cette méthode repose sur l’utilisation de l’en-tête HTTP `Accept-Encoding`. Une nouvelle valeur pour celui-ci, `out-of-band`, est actuellement discutée au sein de l’IETF [126]. S’il peut être pris en charge par le client, le serveur peut alors rediriger l’utilisateur vers une nouvelle adresse sur un domaine maîtrisé par le CDN aval. Bien que similaire à une réponse avec code de retour 302, cette méthode est plus avancée :

- il est possible de spécifier plusieurs adresses, par ordre de préférence. Ainsi, en cas de panne du CDN aval, un autre CDN peut être contacté ;
- il est possible de fournir une clef de chiffrement permettant de déchiffrer le contenu. Ceci permet de diffuser un objet confidentiel sans crainte de fuite de la part des CDNs impliqués dans sa diffusion.

### Sous-certificats et certificats à courte durée de vie

Il est proposé ici de délivrer aux CDNs partenaires un certificat valide pour la diffusion des contenus pour une courte durée de vie. Celle-ci permet une expiration rapide des certificats délivrés à des CDNs partenaires avec lesquels le partenariat s’arrête. Cependant, pour être envisagé, il est nécessaire d’automatiser la signature de tels certificats, ceux-ci devant être très souvent renouvelés.

En ce sens, il est proposé une extension du protocole ACME (*Automatic Certificate Management Environment*). Ce protocole, en cours de normalisation à l’IETF [10], permet d’automatiser le déploiement d’une infrastructure à clefs publiques. L’extension proposée nommée STAR (*Short-Term, Automatically-Renewed*) [132] permet au propriétaire d’un nom de domaine d’autoriser une autorité de certification à signer des certificats de courte durée pour une entité tierce.



## Chapitre 3

# Les communications multichemins

### 3.1 Le routage multichemin

#### 3.1.1 Les réseaux multichemins

Pour des raisons de tolérance aux pannes, la topologie des réseaux de communication a été depuis toujours multichemin. L'exploitation de ces différents chemins peut être envisagée de nombreuses manières et avec différents objectifs.

Cette thématique est presque aussi ancienne que les réseaux informatiques et plutôt que d'effectuer un panorama complet de toutes les possibilités offertes, de tous les objectifs visés et de tous les types de réseaux, nous allons nous concentrer sur les niveaux protocolaires réseau, transport et application pour des réseaux IP. Pour un aperçu plus complet, nous renvoyons le lecteur vers d'autres états de l'art [122] [88].

#### 3.1.2 Problèmes de routage multichemin dans les réseaux hybrides

Comme annoncé, le routage multichemin sur un réseau hybride dont les chemins fournissent un service IP peut être envisagé à différents niveaux : au niveau réseau, au niveau transport ou encore au niveau applicatif. On retrouve, à travers ces différentes couches des problématiques transverses et des questions communes. Bien qu'une partie des problèmes rencontrés soit intrinsèquement liée aux communications multichemins, certains problèmes proviennent ou sont renforcés par l'histoire d'Internet et sa conception pour les communications monochemins.

### Déséquilibrage

Les protocoles de la pile TCP/IP ont été conçus et surtout optimisés pour les communications monochemins; cela a des conséquences sur leur extension pour les réseaux multichemins. Une des hypothèses retenues dans ce contexte historique est une réception des paquets par le destinataire dans leur ordre d'émission. Cependant, l'introduction de chemins distincts introduit un potentiel déséquilibrage, d'autant plus probable que les délais des chemins considérés sont déséquilibrés. Ce phénomène a de grandes répercussions sur les communications, le protocole TCP, très majoritairement utilisé, étant particulièrement affecté [84].

En effet, sa conception a amené à interpréter la réception de paquets hors séquence (par exemple, réception du 3<sup>ème</sup> segment alors que le dernier segment reçu est le 1<sup>er</sup>) comme symptomatique de la perte de paquets précédents non reçus (dans notre exemple, le 2<sup>ème</sup> segment). Cette perte supposée est attribuée à un épisode de congestion, TCP exerçant en conséquence, outre la retransmission des segments incriminés, une réaction négative sur le débit d'émission. La présence de déséquilibrage mène ainsi à de mauvaises interprétations par TCP qui adopte un comportement inadéquat; il en résulte une dégradation considérable des performances de la transmission.

À cela s'ajoute le problème du blocage en tête de file (*head-of-line blocking*). La majeure partie des applications nécessite un service de communication avec livraison des données dans l'ordre, fourni par le protocole TCP. Afin de pallier les déséquilibrages, les mécanismes font généralement appel à un tampon de reséquilibrage. Plus les latences observées sur les chemins diffèrent, plus le tampon est sollicité. Il en résulte une consommation mémoire accrue au niveau du récepteur ainsi qu'une augmentation du délai de bout-en-bout, borné par la latence la plus grande des chemins considérés.

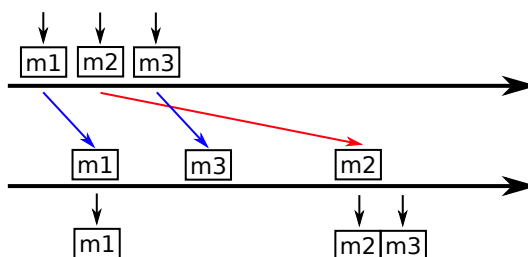


FIGURE 3.1 – Phénomène de blocage en tête de file : le message 3 doit attendre la réception du message 2 avant de pouvoir être délivré à l'application.

Les applications ne nécessitant pas la livraison dans l'ordre des données sont confrontées directement au phénomène de déséquilibrage qui se traduit par une gigue potentiellement importante. Bien que généralement

conçue pour tolérer une faible gigue, l'utilisation de chemins multiples et hétéroclites peut entraîner une baisse de la qualité d'expérience utilisateur comme dans le cas de la voix sur IP par exemple.

### Problématique de MTU

Un chemin entre deux correspondants est caractérisé par sa MTU (*Maximum Transmission Unit*). La connaissance de celle-ci par les protocoles de transport est nécessaire afin d'éviter la fragmentation des paquets et de minimiser la surcharge protocolaire (*overhead*). Afin de la déterminer, TCP a une option MSS (*Maximum Segment Size*) envoyée avec le premier segment SYN mais cette solution connaît de nombreuses limites [82]. Alors que des problèmes sont déjà rencontrés lorsque les communications font face à une situation de routage asymétrique, le routage multichemin se trouve être un cas encore plus pathologique.

### Évolution des architectures

L'évolution vers des communications multichemins est, outre les difficultés liées à ce mode de communication, contrainte par l'architecture existante du réseau. L'expérience montre que celle-ci est difficile à faire évoluer. L'exemple le plus frappant est sans doute celui de l'IPv6 qui, finalisé en 1998 [36] et malgré l'affectation des dernières adresses IPv4 en 2012 [67], a un taux d'acceptation de seulement 18% fin 2017 [59]. En effet, les équipements doivent être mis-à-jour ou remplacés afin de prendre en charge les nouveaux protocoles déployés. Ainsi, l'interopérabilité a une influence importante dans la conception de solutions de routage hybride. Pourtant, même conçues dans cette optique, certaines solutions se trouvent confrontées à des *middleboxes* aux règles de filtrage très strictes ou à l'implantation boguée [142].

#### 3.1.3 Routage multichemin et partage de charge au niveau de la couche réseau

La convergence des réseaux d'accès vers les réseaux IP offre la possibilité de mettre en place relativement aisément un système d'hybridation au niveau de la couche réseau. Cette couche a vu l'apparition de nombreux mécanismes multichemins avec différents objectifs : agrégation, partage de charge, résilience... Certains mécanismes proposent une méthode de routage multichemin accompagnée d'une méthode de relayage spécifique tandis que d'autres proposent un protocole alternatif ou superposé au protocole IP.

Ces mécanismes peuvent être divisés en deux catégories : les mécanismes de routage par flux et les mécanismes de routage par paquet. Dans le premier cas, la décision de routage est prise une seule fois pour un même flux de données ; tous les paquets liés à ce flux sont relayés par le même chemin.



Dans le second cas, la décision de routage est prise à chaque paquet et peut alors différer pour deux paquets appartenant à un même flux.

### Stratégie par flux

Les stratégies par flux ont le grand avantage d'éviter le déséquilibrage des paquets d'un même flux, évitant de ce fait tous les problèmes liés aux protocoles de transport présentés précédemment. Elles permettent l'équilibrage de la charge entre plusieurs chemins ou encore d'offrir des fonctionnalités de résistance aux pannes.

Une famille de solutions repose sur une évolution des protocoles de routage et des méthodes de relayage pour permettre l'utilisation de plusieurs chemins. Ces méthodes reposent généralement sur la stratégie de routage *Equal-Cost Multi-Path* (ECMP) [63]. Celle-ci consiste à configurer, pour une destination donnée, plusieurs routes considérées comme les meilleures et de coût égal. Cette stratégie peut être améliorée en pondérant les multiples routes par exemple à partir d'informations sur la capacité des liens ou sur leur occupation. Les flux sont alors aiguillés aléatoirement sur une des routes possibles, en respectant les pondérations configurées par le protocole de routage.

Les paramètres de pondération peuvent être fixés statiquement, ou évoluer dynamiquement suite par exemple à l'évolution de la congestion (routage adaptatif). Les flux actifs peuvent également être re-routés afin de rééquilibrer la charge ; [73] conseille alors de modifier la décision de routage uniquement pour les flux inactifs depuis quelques secondes afin d'éviter tout déséquilibrage.

Pour déployer une telle stratégie, les routeurs nécessitent d'identifier pour chaque paquet le flux auquel il appartient afin de garantir la même décision de routage que pour les précédents. Pour des raisons de performances, on évite de retenir les décisions de routage pour chaque flux. On utilise à la place une fonction pseudo-aléatoire et déterministe reposant sur une fonction de hachage appliquée à l'en-tête des paquets (adresse IP source, adresse IP de destination, protocole de transport, port source et port de destination). À noter que dans le cas où le protocole de transport n'est pas connu ou n'utilise pas de ports, il est prudent de considérer les paquets comme appartenant à un même flux bien que cela puisse s'avérer sous-optimal.

Des variantes optimisées pour les réseaux multichemins (*Optimized Multipath*, OMP) des protocoles de routage OSPF et ISIS ont ainsi été proposées, OSPF-OMP [145] et ISIS-OMP [87]. Leur utilisation a également été étendue des réseaux IP aux réseaux MPLS [144]. Le protocole BGP a également donné lieu à des propositions de modification afin de permettre un routage multichemin [56].

Les méthodes reposant sur les protocoles de routage sont relativement simples à mettre en œuvre, ne nécessitant pas de changement du plan de

données. Cependant, elles sont limitées par l'utilisation du protocole IP qui ne différencie pas la localisation de l'identification des machines du réseau. Ainsi, de nouveaux protocoles faisant cette distinction ont pu être proposés. Un changement de route est alors facilité par la seule mise à jour du localisateur en laissant l'identifiant intact.

Les protocoles HIP (Host Identity Protocol) [100], MIP [115] [116] et Shim6 [107] introduisent ce concept. Alors qu'IP sert de localisateur, une couche d'identification est ajoutée entre la couche réseau et la couche transport. Devant la difficulté de déploiement de tels mécanismes, d'autres protocoles tels que LISP (*Locator/Identifier Separation Protocol*) [45] et PMIPv6 (*Proxy Mobile IPv6*) [60] ont été introduits, reposant sur une coopération du réseau et laissant ainsi les hôtes intacts. Avec LISP, le trafic IP classique est encapsulé dans un nouvel en-tête IP chargé de la localisation ainsi qu'un en-tête LISP chargé de l'identification. PMIPv6 définit un ensemble de protocoles pour contrôler la mobilité de l'utilisateur et mettre à jour le routage en conséquence.

### Stratégie par paquet

Les stratégies de routage par paquet offrent une granularité de décision plus fine que les stratégies par flux. Elles rendent notamment possibles l'agrégation des flux sur plusieurs liens simultanément. En répartissant les données d'un même flux sur différents liens contraints, l'application peut bénéficier d'un débit agrégé pouvant atteindre la somme des débits des chemins considérés. Plusieurs méthodes permettent de mettre en œuvre une telle stratégie.

[119] propose d'encapsuler les paquets IP dans un en-tête IP externe servant de localisateur. Les auteurs arguent une augmentation du débit utile des flux lorsque les différents chemins ont un débit et une latence proches, situation malheureusement rare en milieu hybride. D'autres travaux proposent d'encapsuler les paquets dans plusieurs tunnels PPP (*Point-to-Point Protocol*) eux-mêmes établis au-dessus d'une connexion SSL ou L2TP par exemple [134] [135] [93].

Comme discuté précédemment, ces stratégies introduisent un potentiel déséquencement qui se traduit par de mauvaises performances du protocole TCP. Ainsi, plusieurs travaux ont cherché à contrecarrer ce problème. [23] propose un mécanisme nommé EDPF (*Earliest Delivery Path First*) consistant à planifier l'envoi des paquets en fonction de leur date d'arrivée supposée dans l'objectif de réduire leur déséquencement. L'algorithme Eifel [92] utilise l'option d'horodatage de TCP pour détecter les situations ne nécessitant pas de retransmission et ainsi limiter les retransmissions inopportunes (*spurious retransmit*). L'algorithme RR-TCP [154] (*Reordering-Robust TCP*) adapte dynamiquement le seuil d'accusés de réception dupliqués déclenchant une retransmission proactive (*fast retransmit*) afin de limiter les retransmissions in-

opportunes. Cette idée est également étudiée dans [85]. L'algorithme F-RTO [127] (*Forward F-RTO*) propose de détecter les retransmissions inopportunes ayant lieu suite à un RTO (*Retransmission Timeout*) trop faible (*Spurious Retransmission Timeouts*). [61] étend l'utilisation de HIP au niveau paquet en s'inspirant des différents mécanismes d'atténuation précédents.

Les mécanismes qui suivent font aveu d'échec des stratégies fondées sur une planification intelligente des envois, et font usage d'un tampon de resé- quencement au niveau du récepteur. Les méthodes similaires à l'algorithme EDPF, bien qu'insuffisantes, restent intéressantes afin de diminuer l'uti- lisation du tampon et les problèmes de blocage en tête de file. Ainsi, [24] propose un ordonnanceur nommé PET (*Packet-Pair based Earliest-Delivery- Path-First algorithm for TCP applications*) utilisant la différence de temps dans la réception de deux paquets afin d'estimer le débit des liens. PET est associé à une politique de gestion du tampon de réception (BMP, *buffer ma- nagement policy*) visant à réordonner les paquets mais également à détecter les pertes proactivement. PRISM [79] ne modifie pas l'ordre des données en lui-même mais utilise un mécanisme nommé RPC (*Reverse Path Controller*) réordonnant les accusés de réception afin de masquer le désé- quencement à l'émetteur. Le projet MLVPN [32] permet de fournir un réseau privé virtuel sur plusieurs chemins agrégés; un ordonnanceur WRR (*Weighted Round- Robin*) prend la décision de routage des paquets; le resé- quencement à la réception est optionnel, il est conseillé de l'activer dans les milieux hétéro- gènes.

### 3.1.4 Protocoles de transport multichemins

Les flux de la couche transport sont généralement associés à une seule transmission (transfert d'un objet web, conversation téléphonique. . .). Ainsi, les transmissions multichemins envisagées à ce niveau ont pour objectif d'agréger les capacités des différents chemins à disposition et tombent donc dans la catégorie des stratégies de routage par paquet.

La difficulté à faire évoluer les architectures existantes a amené à conce- voir la majorité des protocoles de transport multichemins comme des va- riantes des protocoles TCP et SCTP. Ces protocoles de transport garan- tissent une livraison dans l'ordre des paquets, et utilisent pour cela des numéros de séquence. Ainsi, la gestion des problèmes de désé- quencement, bien que non résolue, est facilitée à ce niveau. Le protocole UDP (*User Datagram Protocol*), en revanche, a été ignoré. En effet, les communications multichemins en mode non connecté peuvent être traitées plus facilement au niveau réseau, ou le sont au niveau applicatif lorsque les besoins spécifiques de l'application l'exigent.

### Variantes multichemins de SCTP

Le protocole SCTP apportent des fonctionnalités identiques à TCP et UDP, avec à la fois un service d'échange de datagrammes et un service de transmission de données fiable, en ordre et avec contrôle de congestion. De plus, il ajoute la prise en charge du *multihoming*, facilitant sa modification pour l'étendre aux communications multichemins.

Ainsi, CMT (*Concurrent Multipath Transfer*) [70] est une extension de SCTP permettant la transmission simultanée d'un flux sur deux chemins distincts. Le mécanisme prévoit plusieurs contre-mesures pour faire face aux différents problèmes déjà énoncés pour TCP (retransmissions inopportunes, limitations de l'augmentation de la fenêtre. . .). Par exemple, l'émetteur indique dans chaque paquet le dernier numéro de séquence envoyé sur le même lien, afin de permettre des retransmissions rapides indépendantes sur chaque chemin. D'autres extensions existent telles que cmpSCTP [90] ou LS-SCTP (*Load-Sharing SCTP*) [1], ce dernier maintenant deux numérotations, globale et par chemin, résolvant les problèmes d'ambiguïté liés à l'utilisation d'une seule numérotation. Les extensions WiMP-SCTP [65] ou WiSE (*Wireless SCTP Extension*) [54] [52] [117] s'intéressent spécifiquement à l'utilisation de liens sans-fil. Les problèmes posés par l'utilisation de chemins hétérogènes, caractérisés par des délais très différents, et en particulier les problèmes d'occupation des tampons au niveau de récepteur ont été étudiés par Dreibholz [38] et Adhari [2]. Une des solutions proposées consiste à retransmettre sur le chemin ayant le plus faible délai les paquets causant un blocage en tête de file excessif. La variante mSCTP-CMT [20] traite également ce problème dans le cas particulier du basculement d'un chemin à un autre (*handover*).

### Variantes multichemins de TCP

Le protocole TCP ne prend pas en charge le *multihoming* et nécessite donc de plus lourdes modifications que SCTP. En revanche, il est bien plus utilisé et donc mieux compris et implanté par les équipements des réseaux.

pTCP (*parallel TCP*) [64] établit plusieurs sous-flux TCP et un ordonnanceur envoie les données sur les sous-flux proportionnellement au débit estimé de chacun d'entre eux. Dans un contexte pathologique sans-fil où TCP n'est pas capable de saturer les liens considérés, [143] arrive à obtenir, en envoyant les ACK sur un autre chemin, un débit agrégé supérieur à la somme des débits monochemins.

L'agrégation de plusieurs sous-flux TCP pose cependant plusieurs problèmes d'équité. Par exemple, une situation dans laquelle deux sous-flux partagent un goulot d'étranglement commun, le flux agrégé multichemin obtient un débit double comparativement à un flux TCP. mTCP [153] est le premier protocole à prendre en compte ce problème mais il se contente de dé-

tecter cette situation et écarte alors un sous-flux. Une technique utilisée par E-TCP (*Ensemble-TCP*) [41], CM (*Congestion Management*) [9] ou encore MPAT (*Multi-Probe Aggregate TCP*) [133] consiste à utiliser une fenêtre de congestion commune à l'ensemble des sous-flux. BMC (*Bidimensional-Probe Multipath Congestion Control*) [62] propose une solution intéressante reposant sur l'utilisation conjointe d'un gestionnaire en charge de l'agressivité ainsi qu'un gestionnaire en charge de l'équilibrage entre les différents sous-flux. Finalement, les travaux de Kelly [76] définissent le problème de la congestion couplée et apportent une réponse théorique. Ceux-ci seront repris dans la conception de MP-TCP (*Multipath-TCP*) [53], qui devient une référence, à la fois en raison de sa supériorité théorique mais également en raison d'une implantation réelle et d'une rétro-compatibilité avec TCP le rendant très simple à déployer progressivement.

Depuis, un groupe de travail à l'IETF (MPTCP WG) est apparu, le protocole a été standardisé et de nombreux travaux ont fait et font encore progresser son algorithme de gestion de congestion. La section suivante de ce chapitre lui est consacrée et présente plus en détails son fonctionnement et ses différentes évolutions.

### Variante multichemin de QUIC

L'utilisation croissante [101] du nouveau protocole de transport QUIC (*Quick UDP Internet Connections*) [69] a amené, à l'instar de TCP, à vouloir l'adapter pour l'utilisation de plusieurs chemins. C'est l'objectif de MP-QUIC (*Multipath Extension for QUIC*) [34] [31], très fortement inspiré de MP-TCP.

Alors que des protocoles tels qu'UDP ou TCP sont implantés dans le noyau des systèmes d'exploitation, QUIC et MP-QUIC sont implantés en tant que bibliothèque applicative. Il est ainsi bien plus aisé de procéder au déploiement de leurs évolutions. En revanche, une modification des applications est nécessaire pour utiliser ce nouveau protocole.

### Autres protocoles de transport multichemins

Le protocole Multi-flow Realtime Transport Protocol (MRTP) [94] étend le protocole RTP à l'utilisation de plusieurs chemins, en tenant compte des besoins spécifiques liés au *streaming* multimédia. On relève notamment l'utilisation du codec MDMC (*Multiple Description Motion Compensation*) permettant de diviser le flux vidéo en autant de sous-flux que de chemins, le codec étant capable d'afficher une vidéo dégradée mais sans interruption y compris en cas de coupure ou de congestion sur un des chemins.

### 3.1.5 Protocoles multichemins applicatifs

Le choix de la couche applicative pour implanter des fonctionnalités multichemins est nécessairement lourd en modification des applications. C'est également une couche éloignée du réseau et potentiellement peu informée sur l'état de ce dernier. Cependant, c'est également le niveau le moins contraint par les architectures existantes, une application pouvant aisément reposer sur l'utilisation de protocoles monochemins standards. D'autre part, ce niveau permet de se connecter à plusieurs serveurs distincts simultanément, alors que jusqu'ici les communications avaient été envisagées uniquement entre deux nœuds.

Il est courant pour une application de récupérer simultanément plusieurs contenus afin d'exploiter la diversité de ses accès et / ou la diversité des différents serveurs disponibles. C'est le cas par exemple d'un navigateur Web qui parallélise la récupération des contenus d'une page. Cette utilisation, bien qu'utile, ne nécessite aucun protocole applicatif spécifique et sera donc ignorée ici. En revanche, une application peut, pour les mêmes raisons, vouloir récupérer un contenu depuis plusieurs serveurs ou via des accès différents. Il est alors nécessaire pour le protocole applicatif utilisé d'offrir des fonctionnalités d'agrégation, ce qui se traduit par la possibilité de récupérer uniquement une partition d'un contenu.

Dans ce but, un nouvel en-tête `Accept-Ranges` a été défini pour HTTP dans la *requests for comment* (RFC) 7233 [49]. Celui-ci permet de spécifier la portion précise d'un contenu que l'on souhaite recevoir de la part d'un serveur. Une capacité de *pipelining* y a également été ajoutée par la RFC 2616 [51]. Cette capacité permet à un client de demander un nouvel objet dans une même connexion TCP sans attendre la fin de la requête en cours, éliminant les temps morts inhérents à la réutilisation des connexions. Le *pipelining* est d'autant plus intéressant lorsque utilisé avec les requêtes partielles, permettant d'éviter que ces dernières ne dégradent les performances.

Ces fonctionnalités sont exploitées par plusieurs outils et travaux. Le client HTTP `aria2` est ainsi capable d'obtenir un débit de téléchargement agrégé important.

Des modifications du protocole FTP (FILE TRANSFER PROTOCOL) [121] ont également été proposées comme XFTP [5] ou GridFTP [83]. Ces deux variantes divisent le contenu en morceaux qui sont envoyés à travers plusieurs connexions TCP. Cependant, l'objectif de ces modifications était de pallier les limitations de performances de TCP, notamment dans les réseaux ayant un produit débit / délai important (nommé aussi *Long Flat Network* (LFN)), par l'utilisation de plusieurs flux TCP sur un même chemin. Ainsi, elles ne prévoient pas l'utilisation de plusieurs couples d'adresses IP différentes, et posent un problème évident d'équité. De plus, TCP viole l'indépendance des couches en transmettant au niveau applicatif des informations des niveaux réseau et transport, rendant d'autant plus difficile l'utilisation de plusieurs

adresses IP.

Ces approches laissent à la charge du client le routage des morceaux de contenus sur les différents liens. D'autres protocoles, supposant l'utilisation d'un seul serveur, laissent à celui-ci le choix du routage. Ainsi, Dynamic MPath-Streaming (DMP) [147] ou [81] propose une méthode de routage côté serveur pour le *streaming* multimédia. Notons que le serveur a une meilleure connaissance des contenus à délivrer, informations qui peuvent lui être utiles dans sa décision de routage. En revanche, les informations sur l'état du réseau sont connues plus tard par le serveur que par le client, entraînant une moindre réactivité.

### 3.1.6 Autres travaux

Le groupe de travail MIF (*Multiple InterFace*) au sein de l'IETF vise à s'interroger sur les meilleures pratiques pour la configuration et la gestion d'un système connecté à plusieurs réseaux. Par exemple se pose la question de la détermination du serveur DNS récursif à utiliser si plusieurs serveurs ont été obtenus par DHCP (*Dynamic Host Configuration Protocol*) [128]. La réponse à cette question a des implications importantes en termes de performance, les serveurs récursifs ayant un impact sur la décision de redirection des CDNs.

### 3.1.7 Synthèse

Nous avons vu dans cette section différents protocoles de communication multichemins. Ces protocoles opèrent à différents niveaux et reposent sur diverses approches, chacune présentant des avantages et des inconvénients. Nous les avons répertoriés dans le tableau synthétique 3.1.

Couche		Avantages / inconvénients	
3	stratégie	flux	+ séparation plan de contrôle / plan de données + pas de déséquencement – nécessite l'identification des flux
		paquet	+ agrégation possible – déséquencement
	encapsulation	non	– identificateur / localisateur confondus
		oui	– lourde modification des hôtes nécessaire
4	fondé sur	UDP	+ informations sur l'état du réseau + numérotation et reséquencement – ambiguïté de la numérotation – incompatibilité avec les <i>middleboxes</i> – le niveau 4 n'apporte rien p/r au niveau 3
		TCP	– problématique d'équité complexe + faible modification des applications + très répandu
		SCTP	+ <i>multihoming</i> déjà présent – moins répandu
		QUIC	+ forte croissance de son utilisation – encore marginal devant TCP + modifications faciles à déployer
7		– lourdes modifications des clients nécessaires – couche éloignée du réseau, peu d'informations + informations sur les contenus applicatifs + peu contraint par les architectures existantes + réception possible depuis plusieurs émetteurs	

TABLE 3.1 – Comparaison des avantages et inconvénients des différentes couches pour la mise en place d'un mécanisme de routage multichemin.



## 3.2 MP-TCP et ses évolutions / variantes

Le protocole MP-TCP est une extension de TCP lui permettant d'utiliser conjointement plusieurs chemins à des fins de robustesse et d'augmentation du débit fourni. Il a été conçu de manière à garantir une compatibilité descendante avec les équipements réseaux et les applications, lui conférant ainsi une certaine notoriété due à sa facilité de déploiement. Certains services commerciaux proposent d'ores et déjà son utilisation, notamment à travers l'utilisation de relais MP-TCP. Il s'agit à ce jour du protocole ayant concentré le plus de travaux de recherche sur le délicat problème du contrôle de congestion multichemin, les autres propositions de protocole de communications multichemins ayant bien souvent ignoré, ou uniquement survolé, la question.

### 3.2.1 Fonctionnement général

Le protocole de transport MP-TCP a été standardisé par le groupe de travail MPTCP au sein de l'IETF. Il a été conçu avec pour contraintes de pouvoir être déployé sans modifier les infrastructures existantes d'Internet, sans modifier les applications, et d'être stable et résistant vis-à-vis de la congestion quels que soient le ou les chemins considérés. Les contraintes de compatibilité descendante ont amené à l'implanter comme une extension du protocole TCP.

D'un point de vue applicatif, l'utilisation du protocole MP-TCP est transparente : l'application demande à ouvrir une connexion, le système d'exploitation choisissant alors d'utiliser le protocole MP-TCP en lieu et place du traditionnel protocole TCP. Une interface de programmation applicative (API) offre cependant à l'application la possibilité de contrôler plus finement le comportement du protocole.

D'un point de vue réseau, une connexion MP-TCP se traduit par plusieurs flux TCP qui, par le recours à des options spécifiques, sont logiquement associés à une même connexion. L'établissement d'une connexion MP-TCP sur plusieurs chemins se déroule en deux temps : d'abord l'établissement d'une première connexion TCP puis l'établissement d'une ou plusieurs connexions TCP secondaires.

Afin d'indiquer l'utilisation du protocole MP-TCP, les messages SYN, SYN-ACK et ACK de la poignée de main initiale contiennent l'option MP\_CAPABLE ainsi que d'autres informations telles qu'une clef à des fins de sécurisation. L'option ADD\_ADDR permet au serveur d'informer le client qu'il est également joignable sur une seconde adresse. L'établissement des flux secondaires est réalisé grâce à l'option MP\_JOIN accompagnée d'informations supplémentaires telle qu'un nonce.

Le transfert d'informations entre deux correspondants nécessite la numérotation des données et leur éventuel reséquencement puisque MP-TCP

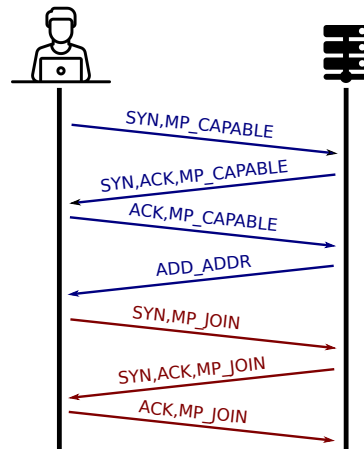


FIGURE 3.2 – Exemple d’initialisation d’une connexion MP-TCP.

garantit, tout comme TCP, leur réception dans l’ordre par l’application destinataire. Cette numérotation est déjà prévue par TCP à travers l’utilisation de numéros de séquence. Cependant, afin d’assurer la plus grande compatibilité possible de MP-TCP, les multiples flux TCP utilisent chacun leurs propres numéros de séquences, garantissant la continuité de ces derniers de manière similaire à une connexion TCP classique. Il en résulte la nécessité d’une re-numérotation, propre à MP-TCP, effectuée via l’option DSS (*Data Sequence Signal*), présente également dans les accusés de réception.

### 3.2.2 Contrôle de congestion

Afin d’adopter un comportement approprié à chaque chemin, chaque sous-flux TCP opère son propre algorithme de contrôle de congestion. Cependant, l’utilisation des algorithmes de congestion classiques de TCP conçus pour assurer une équité entre les flux s’avère inadaptée au protocole MP-TCP. En effet, certaines situations peuvent s’avérer avantageuses pour MP-TCP, la figure 3.3 en présente deux exemples : un premier cas où deux sous-flux MP-TCP partagent un goulot d’étranglement avec un flux TCP et un deuxième cas où un flux MP-TCP partage deux goulots d’étranglement avec deux flux TCP distincts.

Ainsi, une partie des recherches porte sur la conception d’algorithmes de contrôle de congestion spécifiques à MP-TCP qui soient équitables face aux flux TCP classiques, y compris dans les situations pathologiques évoquées.

**Linked-Increases Algorithm** L’algorithme *Linked Increases* (LIA), standardisé par le RFC6356, est une adaptation de NewReno inspirée par les travaux de Kelly et Voice [76]. Ceux-ci introduisent un modèle fluide pour les communications multichemins et déterminent des conditions permettant

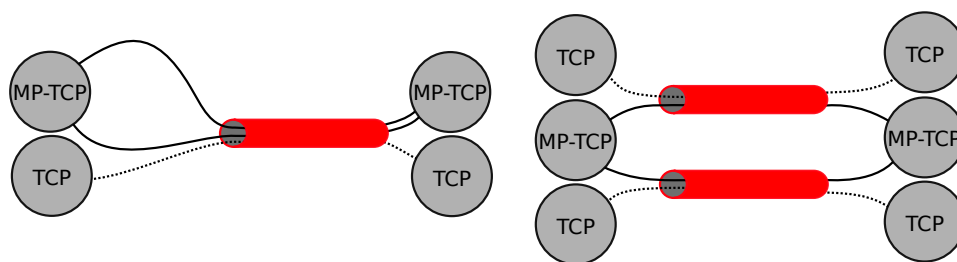


FIGURE 3.3 – Deux situations inévitables au bénéfice de MP-TCP.

l'équité. LIA propose d'incrémenter la fenêtre d'émission d'un sous-flux  $i$  à chaque ACK reçu sur celui-ci de la quantité suivante :

$$\min \left( \frac{\alpha \cdot \text{bytes\_acked} \cdot \text{MSS}_i}{\text{cwnd}_{\text{total}}}, \frac{\text{bytes\_acked} \cdot \text{MSS}_i}{\text{cwnd}_i} \right)$$

Ainsi, l'augmentation que la fenêtre connaîtrait avec TCP (deuxième terme) est bornée par l'augmentation maximale autorisée pour le sous-flux MP-TCP (premier terme). Le paramètre  $\alpha$  détermine l'agressivité du sous-flux et est calculé dynamiquement de manière à assurer un débit agrégé équivalent à ce que TCP obtiendrait sur le meilleur des deux chemins.

Le comportement adopté lors d'événements de congestion (pertes) est celui de NewReno, c'est-à-dire la division par deux de la fenêtre de congestion. À noter qu'au contraire de l'accroissement de la fenêtre de congestion, sa diminution n'est pas liée aux autres sous-flux MP-TCP.

**Opportunistic Linked-Increases Algorithm** Un des objectifs pour un algorithme de contrôle de congestion multichemin, fixé dès la conception de LIA, est d'assurer une congestion équivalente sur tous les chemins. Cependant, LIA fut un compromis entre contrôle optimal et réactivité ainsi qu'évitement des phénomènes d'oscillation, et ne satisfait pas à ce titre ce critère d'équivalence. L'algorithme *Opportunistic Linked-Increases* (OLIA) [77] a été proposé pour atteindre cet objectif. L'accroissement de la fenêtre de congestion est également inspiré par [76] auquel s'ajoute un terme visant à garantir la réactivité de l'algorithme et la robustesse aux oscillations.

**wVegas** L'algorithme *Weighted Vegas* [22] utilise, tout comme l'algorithme TCP *Vegas*, le délai comme information de congestion au lieu des pertes. Cette méthode permet une réaction plus rapide aux évolutions du réseau, un équilibrage de la congestion plus fin et plus rapide et une diminution des pertes. La pondération à laquelle le nom fait référence désigne la répartition du débit agrégé entre les différents sous-flux, et permet à l'algorithme d'une part d'assurer son équité avec les autres flux et d'autre part d'équilibrer son trafic de manière à éviter les liens les plus congestionnés.

**Balanced Linked Adaptation** L'algorithme BALIA [113] [146] est issu de travaux plus récents et généralise les algorithmes précédents en cherchant à obtenir le meilleur compromis entre équité avec TCP, réactivité et oscillations.

D'autres algorithmes de contrôle de congestion pour MP-TCP ont également été proposés [58] [91] [78] mais ceux-ci ne sont pas présents dans l'implantation Linux de référence.

### 3.2.3 Ordonnement

Bien qu'ayant suscité moins d'intérêt, l'ordonnement est également une fonctionnalité importante pour le fonctionnement de MP-TCP. L'ordonneur a pour rôle de choisir, pour chaque paquet, le sous-flux sur lequel il doit être envoyé. Il existe à ce jour deux ordonneurs dans l'implantation Linux du protocole.

- L'ordonneur `round-robin`, qui envoie les paquets de manière cyclique sur chacun des sous-flux pour lesquels la fenêtre d'émission le permet ;
- L'ordonneur `minRTT` qui envoie les paquets, parmi les sous-flux pour lesquels la fenêtre d'émission autorise l'envoi, sur celui dont le délai du chemin est le plus faible.

Certains mécanismes de MP-TCP peuvent être influencés par l'ordonneur. C'est le cas des retransmissions, ainsi que des retransmissions opportunistes, ces dernières consistant à renvoyer certaines données qui ont déjà été transmises sur un autre sous-flux mais pas encore perdues, dans le but par exemple d'accélérer la fin des connexions. Les durées des temporisations sont également concernées, puisque généralement déterminées à partir du délai du lien, posant la question de la détermination de celui qui doit être pris en compte. Ces considérations sont essentielles à la minimisation du blocage en tête de file, phénomène qui se produit facilement avec la différence de délai mais également avec les retransmissions inhérentes au fonctionnement de la plupart des algorithmes de contrôle de congestion, et qui est accentué par la présence de multiple sous-flux.

## 3.3 Le projet BATS

Le projet BATS (*Broadband Access via integrated Terrestrial and Satellite systems*) est un projet de recherche à destination des territoires européens non desservis ou mal desservis au regard du haut débit. Ce projet, financé par la commission européenne, a démarré fin 2012 pour une durée de 3 ans. Il visait à combler la fracture numérique de ces territoires en leur fournissant un accès à haut débit. Pour traiter ce problème, une approche

hybride satellite / terrestre totalement intégrée fut explorée. Ces travaux ont par la suite été repris par l'ETSI (*European Telecommunications Standards Institute*) [42] [43]. En raison des similitudes avec les travaux de cette thèse, nous exposons ici une partie des propositions issues de ce projet.

### 3.3.1 Contexte et objectifs du projet BATS

La Commission Européenne a fixé dans son plan d'aménagement numérique du territoire comme objectif de fournir un accès Internet à haut-débit d'au moins 30 Mbit/s à l'ensemble des foyers européens. Le projet BATS cible les territoires ruraux pour lesquels les coûts de déploiement de nouvelles infrastructures terrestres freinent la modernisation. Ainsi, il est proposé de fournir pour ces territoires un accès à Internet par satellite, hybridé à un accès terrestre à bas débit pour augmenter la qualité du réseau.

BATS propose un système intégré capable d'utiliser divers moyens d'accès de manière intelligente et transparente pour l'utilisateur. Les technologies d'accès envisagées incluent des communications par satellite ainsi que des technologies xDSL, LTE ou Wi-Fi. À ce titre, les pistes de recherche abordées par le projet sont très variées. Elles incluent l'optimisation des réseaux satellitaires, l'intégration et la gestion d'un accès hybride, l'optimisation de la fourniture des services aux utilisateurs à travers l'architecture proposée, ainsi que l'étude de la viabilité économique d'un tel projet. Nous nous intéresserons plus spécifiquement, dans le cadre de cette thèse, aux apports en termes d'architecture et de méthodes pour l'hybridation.

### 3.3.2 Architecture

L'architecture proposée par BATS est représentée sur la figure 3.4. Celle-ci permet d'interconnecter un réseau domestique avec l'Internet via différentes technologies d'accès. On suppose qu'elles incluent un accès satellite ainsi qu'un ou plusieurs accès terrestres (xDSL, LTE...).

Les différents réseaux d'accès sont gérés par le biais de deux serveurs mandataires, nommés *Intelligent User Gateway* (IUG) et *Intelligent Network Gateway* (ING). Un serveur IUG est situé dans chaque foyer familial, son rôle étant de router le trafic de l'utilisateur vers l'ING. *A contrario*, l'ING se situe dans le cœur de réseau d'un fournisseur d'accès et s'occupe du routage du trafic en sens inverse.

Les réseaux d'accès étant très hétérogènes, avec des caractéristiques de latence ou de débit très différentes, le choix d'un chemin particulier a de fortes conséquences en termes de performance. Chaque routeur intelligent est donc chargé de sélectionner le chemin le plus adapté en fonction des besoins des applications et des caractéristiques des liens. Ainsi, les routeurs sont structurés en trois blocs fonctionnels. Le premier est chargé de la classification du trafic suivant le type de service (*Classes of Service*, CoS) ainsi

que la longueur des objets (court ou long). Le second est en charge de la sélection du chemin le plus adapté et repose sur l'utilisation du protocole de transport multichemin MP-TCP. Pour finir, les informations de la couche transport sont utilisées pour en déduire les caractéristiques des liens.

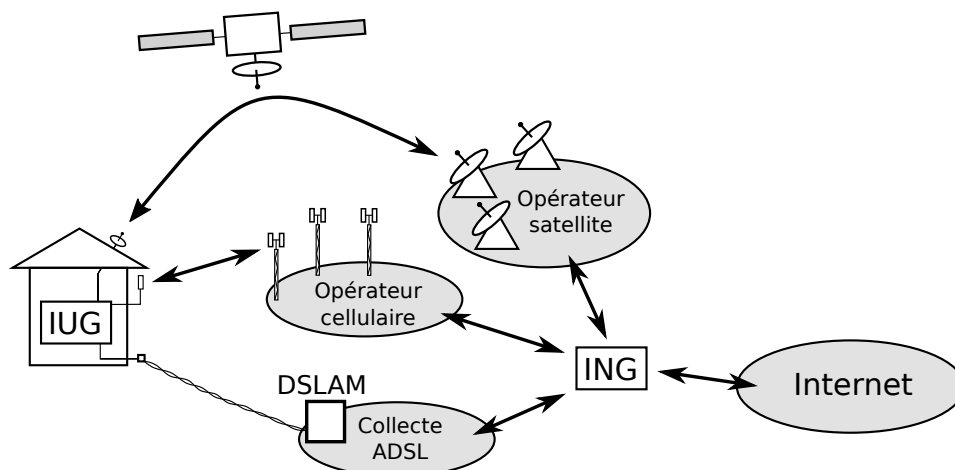


FIGURE 3.4 – Architecture BATS

### 3.3.3 Classification du trafic

La classification du trafic par BATS est décomposée en plusieurs méthodes qui sont testées successivement pour chaque nouveau flux jusqu'à ce que l'une d'entre elles se montre capable de déterminer la classe appropriée. La première méthode de classification consiste à reconnaître le trafic de signalisation tel que SIP ou H.323 et d'en déduire après analyse les informations nécessaires pour reconnaître les flux de données concernés (par exemple le port source d'un flux audio). La seconde méthode consiste à observer le champ DSCP (*Differentiated Services Code Point*) présent dans l'en-tête des paquets IP. Si ce dernier n'est pas présent, BATS se repose alors sur l'inspection des paquets en profondeur (DPI). Cette dernière méthode inclut notamment la reconnaissance des flux à partir de tables de connaissances contenant une liste de protocoles et de ports d'applications bien connues.

Il est à noter que l'identification d'une application peut conduire à des politiques de service différentes pour l'IUG et l'ING. Par exemple, le trafic IMAP (mail) est traité comme sensible au délai par l'IUG et comme trafic élastique par l'ING, l'utilisateur envoyant des commandes de contrôle alors que le serveur envoie des e-mails contenant potentiellement de lourdes pièces jointes.

### 3.3.4 Routage du trafic

BATS distingue le trafic TCP, majoritaire, des autres protocoles de transport dans sa méthode de routage.

#### Trafic TCP

L'IUG et l'ING agissent comme serveurs mandataires pour le trafic TCP. À ce titre, les flux TCP sont coupés à leur niveau les divisant en trois segments. Alors que TCP reste utilisé entre le client et l'IUG ainsi qu'entre l'ING et le serveur, BATS utilise le protocole MP-TCP entre l'IUG et l'ING.

Cependant, MP-TCP n'est utilisé qu'en tant que *framework* pour l'établissement de différents sous-flux sur chacun des chemins disponibles. BATS propose son propre mécanisme d'ordonnancement, ce dernier transmettant chacun des contenus que sur un unique sous-flux. La sélection du chemin sur lequel est transmis un contenu se fait en deux étapes.

Premièrement, les chemins dont les caractéristiques répondent aux exigences en termes de qualité de service sont identifiés, ces exigences étant déterminées par le processus de classification. Ensuite, parmi les chemins retenus, le plus approprié est sélectionné. Cette sélection repose sur la taille de l'objet transféré : si elle est inférieure à un seuil donné, le chemin au délai le plus faible est sélectionné ; sinon, le chemin avec la plus forte capacité est retenu. S'il advenait qu'aucun chemin ne satisfasse les exigences du flux, alors BATS adopte un comportement par défaut en envoyant ce dernier sur le chemin au plus faible délai.

Cet algorithme décisionnel nécessitant la taille des objets, BATS propose de l'estimer en observant le délai d'inter-arrivée entre les paquets TCP. Si ce délai est inférieur à un certain seuil, les paquets sont alors supposés véhiculer le même objet dont la taille est estimée à partir des charges utiles des segments TCP. Dans le cas contraire, les données sont supposées appartenir à un nouvel objet. Notons que la taille d'un objet peut être sous-estimée si celui-ci n'a été que partiellement reçu par le serveur mandataire se livrant à son estimation.

#### Trafic non-TCP

La gestion du trafic non-TCP, essentiellement UDP, se fait par reconnaissance du trafic (port bien connu. . .) associée à une politique de routage pré-définie.

### 3.3.5 Résultats

Finalement, le projet BATS a donné lieu à différentes expérimentations. Les résultats obtenus avec le mécanisme de routage par seuil proposé par BATS ont ainsi pu être confrontés aux performances de MP-TCP muni de

son ordonnanceur par défaut `minRTT`. Notons qu'un seuil fixe de 10ko est utilisé sans précision sur le choix de cette valeur. Les résultats montrent alors un gain notable de la stratégie par seuil sur MP-TCP.





## Chapitre 4

# Architecture de diffusion de contenus sur réseau hybride satellite / terrestre

Nous avons motivé en introduction de cette thèse notre intérêt pour le problème du routage multichemin sur un réseau hybride satellite / terrestre intégré dans une architecture CDN. Cette motivation repose à la fois sur l'intérêt d'un réseau hybride satellite / terrestre pour les zones grises ainsi que la place centrale des CDNs dans la livraison des contenus. Ce chapitre présente une telle architecture qui nous permettra, par la suite, d'étudier le routage.

### 4.1 Acteurs

Avant de présenter notre architecture, nous définissons ici les différents acteurs qui interagiront au sein de celle-ci.

- Les **fournisseurs de contenus** sont les créateurs des contenus (pages web, vidéos, images, . . .) auxquels accèdent les utilisateurs.
- Les **opérateurs CDNs** gèrent et opèrent chacun un CDN. C'est auprès d'eux que les fournisseurs de contenus souscrivent un contrat de livraison de contenus afin de déléguer cette tâche.
- Les **fournisseurs d'accès terrestre**.
- Les **fournisseurs d'accès satellite**.
- Les **fournisseurs d'accès Internet hybride**. C'est auprès de ces derniers que les utilisateurs finaux souscrivent une offre d'accès hybride.
- Les **utilisateurs** finaux, accédant aux contenus à travers un réseau hybride.

Dans la pratique, le fournisseur de service Internet hybride peut aussi être l'un des deux fournisseurs d'accès terrestre ou satellite, ajoutant alors une plus-value à ses offres commerciales.

## 4.2 Architecture de routage hybride satellite / terrestre

L'architecture hybride satellite / terrestre proposée est représentée sur la figure 4.1. Elle est très similaire à celle proposée par le projet BATS [118], mais diffère, nous le verrons par la suite, dans les méthodes de routage, leur mise en œuvre ou encore par les interactions avec les autres acteurs.

On considère ainsi un réseau domestique, partagé par un ou plusieurs utilisateurs et raccordé à Internet par l'intermédiaire d'un réseau hybride satellite / terrestre. On suppose que les utilisateurs accèdent à des objets web produits par un fournisseur de contenus, ce dernier ayant fait appel à un opérateur CDN pour leur diffusion auprès des utilisateurs.

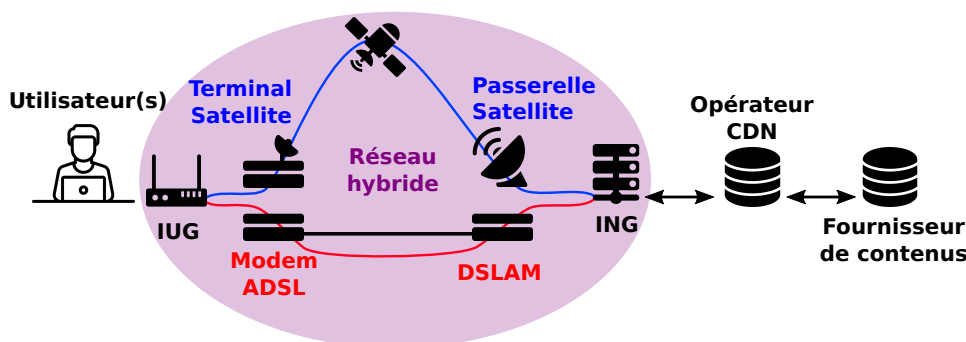


FIGURE 4.1 – Architecture hybride satellite / terrestre

Le domicile familial dispose d'un réseau terrestre bas débit et d'un accès satellite haut débit. Ces deux réseaux d'accès sont associés logiquement par deux serveurs mandataires intelligents, que l'on nommera, pour reprendre la terminologie introduite par le projet BATS, IUG (*Intelligent User Gateway*) et ING (*Intelligent Network Gateway*). L'IUG fait la jonction entre les utilisateurs et le réseau hybride tandis que l'ING fait la jonction entre le réseau hybride et Internet. Ils sont tous les deux opérés par le fournisseur d'accès hybride.

La partie hybride du réseau est ainsi entièrement gérée par le fournisseur d'accès hybride par l'intermédiaire de ses deux serveurs mandataires ; elle est transparente du point de vue du reste du réseau. C'est donc l'opérateur hybride qui décide de la politique d'hybridation, son bon fonctionnement étant alors sous sa responsabilité.

## 4.3 Mise en cache et interconnexion CDNI

Afin d'améliorer la qualité de service de son réseau, le fournisseur d'accès hybride a tout intérêt à déployer dans son réseau ses propres serveurs de cache, que l'on appellera TelCo CDN.

### 4.3.1 La fin de la mise en cache transparente

Durant de nombreuses années, l'ajout de serveurs mandataires transparents procédant à la mise en cache des contenus fut une solution appréciée par les fournisseurs d'accès mais également par les gestionnaires de réseaux d'entreprise pour améliorer les performances de la diffusion des contenus. L'IUG est dans notre cas un candidat idéal, en raison de sa position, pour le déploiement d'un serveur de cache transparent. L'ING a également une position intéressante, au plus près de l'interface de sortie du réseau hybride, et profitable à l'ensemble des clients du fournisseur d'accès hybride.

Cependant, la progression rapide du nombre de flux chiffrés ces dernières années [47] vient mettre à mal cette stratégie. En effet, 60% des URLs (*Uniform Resource Locators*) référencés sur Internet sont des liens HTTPS [125]. D'après les statistiques de télémétrie du navigateur Firefox [114], plus de 70% des pages web auxquelles on accède le sont en utilisant le protocole HTTPS alors qu'il ne s'agissait que de 30% des pages début 2014.

Les serveurs mandataires transparents ne sont ainsi plus applicables, il est alors nécessaire de se tourner vers d'autres méthodes permettant la mise en cache des objets chiffrés.

### 4.3.2 Interconnexion CDNI

Une solution au problème du chiffrement est l'interconnexion de CDN présentée au chapitre 2 dans la section 2.2. Ainsi, on suppose que le TelCo CDN est connecté à l'opérateur CDN à l'aide des interfaces proposées par le groupe de travail CDNI. L'opérateur CDN délègue donc la diffusion des contenus au fournisseur d'accès hybride pour les requêtes en provenance du réseau de ce dernier. Les requêtes des utilisateurs sont redirigées, par les mécanismes prévus par les interfaces CDNI, vers le TelCo CDN. Celui-ci a alors toute la liberté pour optimiser la diffusion du contenu jusqu'à l'utilisateur à travers le réseau hybride.

La figure 4.2 présente plus en détails la succession des redirections auxquelles est confronté l'utilisateur. La requête initiale de l'utilisateur est transmise par l'IUG à l'ING par le réseau hybride puis acheminée jusqu'au fournisseur de contenus. Celui-ci va alors la rediriger vers l'opérateur CDN qui va à son tour la rediriger vers le TelCo CDN. Finalement, le TelCo CDN sélectionne un de ses caches afin de transmettre le contenu à l'ING qui pourra démarrer sa diffusion à travers le réseau hybride.

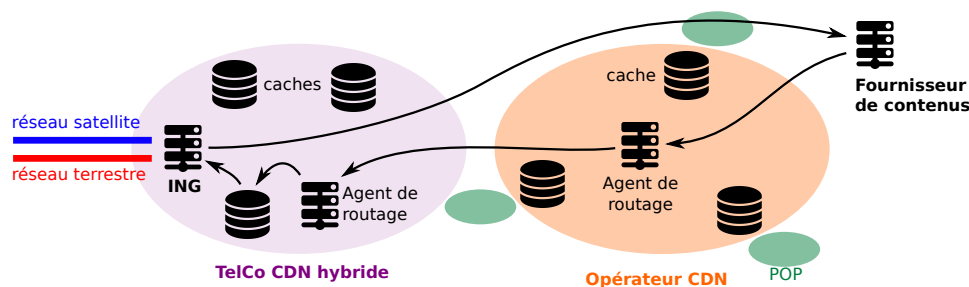


FIGURE 4.2 – Architecture hybride satellite / terrestre

### 4.3.3 Informations utiles au routage

Grâce à cette architecture, l'ING est en mesure d'obtenir un grand nombre d'informations sur les contenus à diffuser, telles que leur taille, leur extension ou encore leur type MIME. Ces informations pourront alors être utilisées dans le mécanisme de routage, et les chapitres suivants s'intéresseront explicitement à l'utilisation de la taille du contenu dans cet objectif.

La taille, en particulier, peut être récupérée de plusieurs manières. Tout d'abord, par un en-tête `Content-Length` lors de l'acquisition en HTTP. Cet en-tête est connu pour être souvent erroné [129] ; cependant ceci s'explique par la génération dynamique de certains contenus tels que les pages web, ne permettant pas au serveur web de connaître leur taille à l'avance. Dans le cas de contenus statiques et pouvant être mis en cache, on peut supposer l'information exacte.

La taille peut également être récupérée par l'utilisation d'une métadonnée spécifique acquise à travers l'interface CDNI de métadonnées (MI). Pour finir, l'ING peut également attendre la fin de l'acquisition du contenu, connaissant ainsi sa taille totale, avant de commencer son transfert vers l'IUG. Cette dernière approche est moins recommandée en raison du délai introduit avant le démarrage du transfert. On peut toutefois supposer celui-ci relativement court en raison de la localisation de l'ING proche des dorsales Internet.

### 4.3.4 Mise en cache par l'IUG

La redirection des requêtes par l'opérateur CDN vers le TelCo CDN permet, outre l'accès à des informations utiles au routage, de mettre en cache les contenus au plus près de l'utilisateur. En effet, leur mise en cache au niveau du TelCo CDN les positionne au plus proche de l'interface de sortie du réseau hybride, c'est-à-dire de l'ING.

Cependant, il est possible de considérer l'IUG lui-même comme un cache appartenant au TelCo CDN. Comme expliqué dans la section 4.3.1, la mise en cache de manière transparente des contenus montre ses limites, en raison

notamment de l'augmentation du trafic chiffré. Considérer l'IUG comme appartenant au TelCo CDN permet de s'affranchir de cette limite d'une part, et de mettre en place une politique de mise en cache optimisée, tenant compte de l'état des autres caches du TelCo CDN. Nous supposons dans la suite avoir toujours affaire à un contenu à récupérer de manière sécurisée.

La requête initiale d'un utilisateur pour un certain contenu se fait généralement sur un domaine dont l'opérateur CDN possède les clés de chiffrement, soit parce qu'il s'agit d'un de ses propres sous-domaines, soit parce que le fournisseur de contenus a fourni à l'opérateur CDN un certificat valide pour un de ses propres sous-domaines. Pour diffuser les contenus depuis les caches du TelCo CDN, il est nécessaire de fournir à ce dernier un accès au certificat du domaine en question, ou alors de rediriger l'utilisateur vers un domaine maîtrisé par le TelCo CDN. Nous avons pu voir dans la section 2.2.5 plusieurs extensions pour CDNI afin de permettre à un CDN aval de diffuser par différentes méthodes un contenu en limitant les implications en termes de sécurité. Cependant, la position très sensible de l'IUG, directement chez les utilisateurs, rend peu probable la mise en place d'une telle stratégie sur ce dernier ; ainsi, on s'orientera vers une redirection HTTP de l'utilisateur vers un domaine maîtrisé par le TelCo CDN.

En ce sens, nous présentons 3 scénarios permettant une telle redirection. D'autres scénarios sont bien évidemment envisageables, suivant les choix des protocoles ou des types de redirection (récursive ou itérative). D'autre part, le TelCo CDN peut choisir de servir directement un contenu sans rediriger l'utilisateur vers l'IUG, par exemple pour un contenu à la faible popularité dont la mise en cache sur l'IUG est jugée non nécessaire.

### Scénario n° 1

La première méthode que nous présentons est une redirection HTTP itérative telle que représentée sur la figure 4.3. Notons que l'indication ING sur notre figure représente le cœur de réseau de l'opérateur hybride, qui est également TelCo CDN.

On suppose que l'utilisateur accède à un contenu hébergé sur le domaine `acme.op-cdn.org`. La requête de l'utilisateur est alors redirigée en HTTP par l'opérateur CDN vers le domaine `acme.telco-cdn.org`, domaine maîtrisé par le TelCo CDN. Les paramètres de cette redirection ont été négociés à l'avance entre l'opérateur CDN et le TelCo CDN lors de la mise en place de l'interconnexion CDNI.

L'utilisateur initie alors une deuxième requête vers l'ING, qui redirige une deuxième fois l'utilisateur, cette fois-ci vers un domaine `acme.client1.telco-cdn.org`. Ce dernier sous-domaine est directement contrôlé par l'IUG de l'utilisateur. Il s'agit d'une redirection du TelCo CDN vers un de ses propres caches, le plus proche de l'utilisateur, en l'occurrence son propre IUG.

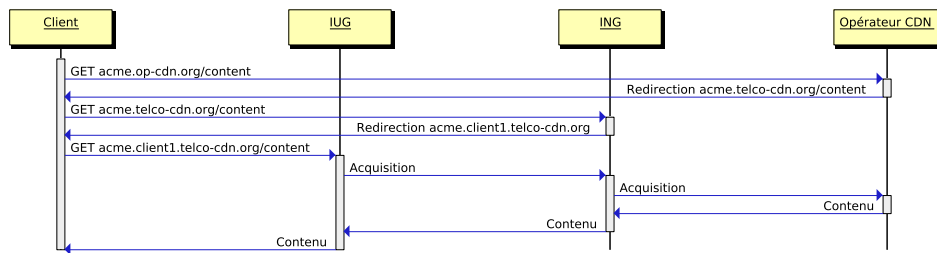


FIGURE 4.3 – Redirection HTTP itérative du client.

L'utilisateur peut alors se connecter directement à l'IUG afin de récupérer son contenu. L'IUG va alors acquérir le contenu depuis l'ING qui va elle-même procéder à son acquisition depuis l'opérateur CDN. Bien sûr, si l'une de ces entités possède déjà le contenu en cache, la diffusion du contenu s'en trouvera accélérée.

Notons que l'acquisition du contenu par l'IUG depuis l'ING se fait à travers le réseau hybride, et que la qualité de service ressentie par l'utilisateur sera directement liée aux performances de l'exploitation du réseau hybride.

## Scénario n° 2

Dans ce deuxième scénario représenté sur la figure 4.4, l'utilisateur est confronté à une redirection HTTP récursive. Lors de la requête initiale de l'utilisateur, l'opérateur CDN contacte le TelCo CDN et lui transmet les informations connues sur l'utilisateur : adresse IP, type de terminal utilisé. . . Le TelCo CDN en déduit alors le client dont il s'agit, et demande à ce qu'il soit redirigé vers le domaine `acme.client1.telco-cdn.org`. La suite de la livraison du contenu est toujours identique.

Ce scénario, comparativement au scénario n° 1, permet l'économie d'une requête HTTP entre l'utilisateur et l'ING.

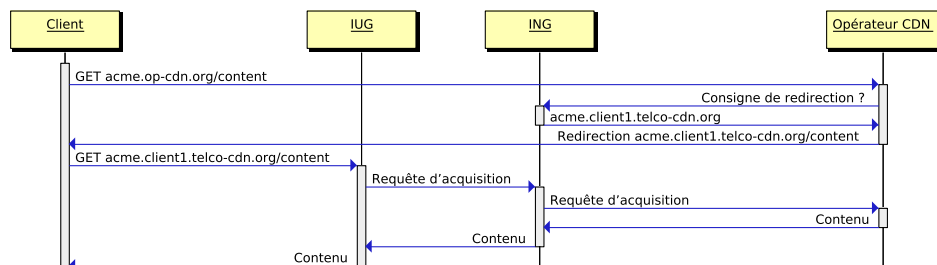


FIGURE 4.4 – Redirection DNS puis redirection HTTP avec usage limité d'une clef distante.

### Scénario n° 3

Pour ce troisième scénario, présenté sur la figure 4.5, l'utilisateur est tout d'abord redirigé par une redirection DNS vers l'ING. En réalité, on suppose que l'utilisateur utilise le serveur DNS récursif intégré à son IUG qui lui-même retransmet les requêtes à un serveur récursif situé dans le cœur de réseau de l'opérateur hybride, et donc proche de l'ING. Ce serveur DNS récursif étant intégré au TelCo CDN, il connaît les domaines pour lesquels le TelCo CDN a une délégation et répond donc avec l'adresse IP d'un de ses serveurs de cache.

Lorsque l'utilisateur se connecte au cache du TelCo CDN, celui-ci doit lui présenter un certificat valide pour le domaine `acme.op-cdn.org`. On suppose ici que le TelCo CDN a un usage limité d'une clef distante possédée par l'opérateur CDN (voir section 2.2.5). Le TelCo CDN peut alors rediriger l'utilisateur en HTTP vers le domaine `acme.client1.telco-cdn.org`.

L'utilisateur va alors récupérer le contenu depuis son IUG comme dans le scénario précédent.

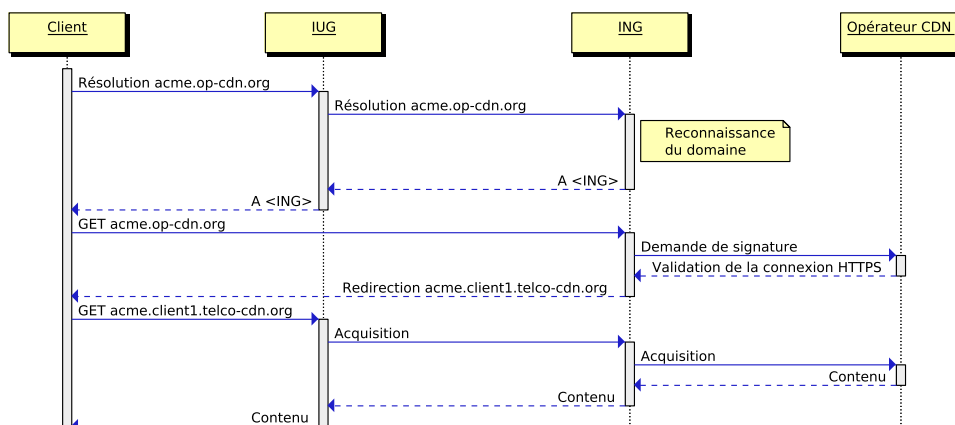


FIGURE 4.5 – Redirection HTTP récursive du client.

### Conclusion sur ces scénarios

Nous avons présenté ici trois scénarios permettant de rediriger la requête d'un utilisateur vers le TelCo CDN, c'est-à-dire vers l'ING et *a fortiori* vers l'IUG. La multiplicité des scénarios possibles, non limitée aux trois présentés, montre la liberté qui existe dans le choix des méthodes, protocoles et localisation des fonctions. En pratique, le choix d'un scénario sera aiguillé par :

- des contraintes opérationnelles : le CDN d'opérateur permet-il l'utilisation limitée de clefs distantes ? Les redirections récursives ? Est-il envisageable de partager un certificat avec un TelCo CDN ?



- les performances : préférence des redirections récursives aux redirections itératives, prépositionnement des contenus et / ou réponse DNS...
- le fonctionnement des algorithmes de placement des contenus : est-il nécessaire de mettre en cache un contenu sur l'IUG (et donc d'y rediriger l'utilisateur) ?

Dans la suite de cette thèse, nous nous intéresserons plus spécifiquement à la diffusion des contenus sur le réseau hybride, et supposerons à ce titre que l'utilisateur a déjà été redirigé vers le TelCo CDN, peu importe le scénario mis en œuvre.

#### 4.3.5 Diffusion en *multicast* par satellite

Bien qu'il ne s'agisse pas de la thématique explorée dans cette thèse, l'architecture proposée permet une application très intéressante qu'est la diffusion en *multicast* par le satellite de certains contenus. En effet, outre son débit important comparativement au réseau terrestre considéré ici, le satellite possède une propriété notoire qu'est sa capacité de diffusion à grande échelle d'un même contenu. Il est alors possible de l'exploiter afin de mettre en cache un même contenu chez un grand nombre d'utilisateurs [80].

Cette possibilité peut être exploitée lors de la demande d'un contenu afin de rentabiliser sa diffusion, ou bien pour la mise en cache proactive des contenus populaires. Cette dernière activité a lieu généralement la nuit, lorsque les réseaux sont moins occupés, et constitue une technique classique des réseaux CDNs, y compris dans le cas de réseau terrestre sans capacité de *multicast* [104].

## 4.4 Conclusion

Dans ce chapitre, nous avons présenté une architecture intégrant un réseau hybride satellite / terrestre dans une chaîne de diffusion de contenus. Cette intégration repose sur l'interconnexion d'un TelCo CDN géré par un fournisseur d'accès hybride avec les opérateurs CDN traditionnels. La diffusion des contenus à travers le réseau hybride est isolée et entre les mains de l'opérateur hybride. Elle sera traitée plus spécifiquement dans les chapitres suivants de cette thèse.

## Chapitre 5

# Mécanismes de routage fondés sur l’usage de la taille

Dans le chapitre précédent, nous avons présenté une architecture intégrant un réseau hybride satellite / terrestre dans une chaîne de diffusion CDN. Cette architecture permet à l’IUG (*Intelligent User Gateway*) et l’ING (*Intelligent Network Gateway*), en position de serveurs mandataires, de choisir le réseau par lequel transite chaque contenu. Cette position permet également d’accéder à plusieurs informations sur les contenus, notamment leur taille.

Dans ce chapitre, nous étudierons l’impact du mécanisme de routage sur les performances de cette architecture. En particulier, nous nous intéresserons à l’intérêt de la connaissance de la taille des contenus et son utilisation par le mécanisme de routage. Ainsi, nous comparerons différents mécanismes, faisant intervenir – ou non – la taille des contenus, et montrerons l’avantage significatif que procure cette connaissance.

L’évaluation de ces mécanismes donnera lieu à une résolution analytique, des simulations et des expérimentations. Ces différentes approches nous permettront d’une part de vérifier nos résultats et d’autre part de valider nos hypothèses et implantations. Les simulations et expérimentations nous seront également utiles pour évaluer notre dernier mécanisme pour lequel nous n’avons pas de résolution analytique.

Les mécanismes que nous étudierons dans ce chapitre seront présentés dans la section 5.2 et opéreront tous à l’échelle des flux. Supposant que chaque flux véhicule **un** contenu, cela signifie que l’ensemble des paquets appartenant à un même flux transiteront par le même réseau. Ainsi, la décision de routage sera prise une seule fois pour chaque contenu, avant le début de son transfert. Cette contrainte sera relâchée dans le chapitre suivant.

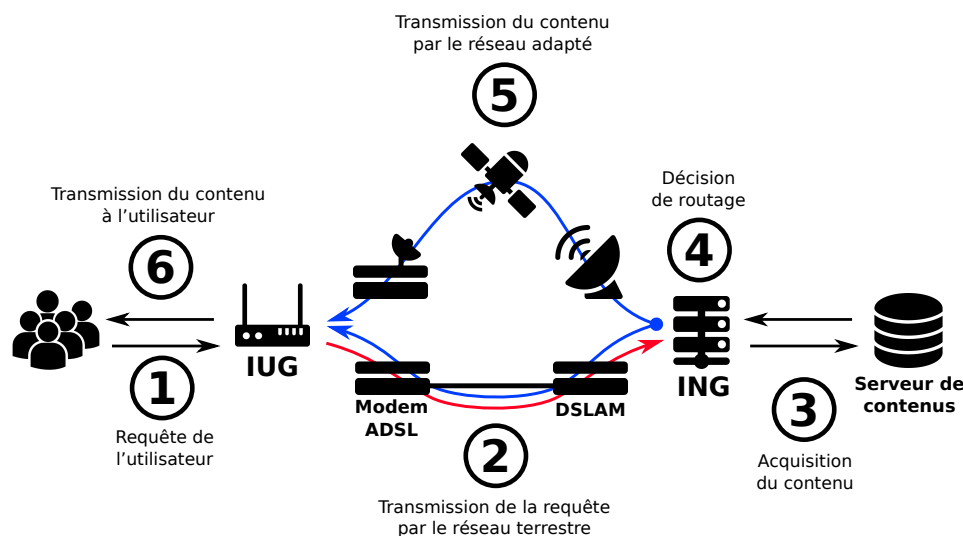


FIGURE 5.1 – Étapes d'un cycle requête-réponse.

## 5.1 Modélisation

Cette section présente la modélisation d'une architecture de diffusion reliant un foyer familial à Internet par un réseau hybride satellite / terrestre.

Le trafic engendré par les membres de la famille est modélisé par une population de  $N$  utilisateurs. Les utilisateurs sont indépendants et répètent indéfiniment un cycle attente-requête-réponse. Un habitant effectuant généralement plusieurs requêtes en parallèle, ce que nous appelons ici un utilisateur ne modélise qu'une fraction de l'activité d'une personne réelle.

Un cycle, tel que représenté sur la figure 5.1, se déroule de la manière suivante :

- **Attente** : l'utilisateur réfléchit à la prochaine requête qu'il va faire durant un temps aléatoire supposé distribué selon une loi exponentielle.
- **Génération de la requête** : un contenu ayant été choisi, l'utilisateur transmet sa requête à l'IUG.
- **Transmission de la requête à l'ING** : l'IUG transmet la requête de l'utilisateur à l'ING par le réseau au plus faible délai (généralement le réseau terrestre dans notre cas).
- **Acquisition du contenu par l'ING** : l'ING ayant reçu la requête, il procède à l'acquisition du contenu depuis le CDN amont.
- **Choix du chemin de transmission** : l'ING exécute son algorithme de routage afin de choisir le chemin par lequel le contenu sera transmis.
- **Transmission du contenu à l'IUG** : le contenu est transmis à l'IUG par l'ING *via* le chemin choisi préalablement.

- **Transmission du contenu à l'utilisateur** : le contenu est finalement transmis à l'utilisateur par l'IUG.

Lors de la réception complète du contenu, l'utilisateur retourne alors en attente pendant une période aléatoire avant de lancer une nouvelle requête.

### 5.1.1 Hypothèses simplificatrices

Notre objectif étant d'observer les performances du choix du mécanisme de routage, notre modélisation simplifie ou ignore sciemment certains éléments.

Premièrement, nous considérons que les utilisateurs et l'IUG se trouvent sur un même réseau local. L'impact de ce réseau est supposé négligeable devant celui des réseaux d'accès ; nous le considérons donc à débit illimité et latence nulle.

Le temps d'acquisition des contenus par l'ING depuis le serveur de contenus est négligé. En effet, il donne lieu à un problème classique de livraison dans une architecture CDN qui est d'une part ici hors de propos, et d'autre part indépendant du mécanisme de routage. Il s'agit d'un temps supplémentaire subi quelle que soit la décision de routage et que l'on peut par ailleurs supposer relativement faible, l'ING étant supposé être bien connecté aux dorsales d'Internet. Ainsi, nous supposons également que l'ING et le serveur de contenus sont sur un même réseau local au débit illimité et à latence nulle.

D'autre part, les chemins terrestres et satellite, incluant modem ADSL, terminal satellite, etc, séparant l'IUG et l'ING sont modélisés par deux liens. La latence de chaque lien correspond à la latence du chemin, et son débit à celui du goulot d'étranglement présent sur le chemin, que l'on suppose par ailleurs être le lien d'accès. Ce modèle pourrait être affiné pour tenir compte de l'impact de la variation météo sur la capacité du lien satellite ou de l'encombrement sur le chemin terrestre.

Les requêtes étant de faible taille et la décision de routage n'étant pas étudiée dans le sens montant (IUG vers ING), nous considérons un débit illimité dans ce sens. Le temps de propagation est tout de même pris en compte.

Pour finir, nous modélisons les communications entre l'ING et l'IUG par des transmissions sans perte, retransmission, ni connexion. La capacité des liens est équitablement répartie, en termes de débit, entre les différents flux partageant le même chemin. De plus, la capacité des liens est supposée être exploitée au maximum. Nous supposons tout de même des communications constituées de paquets, dont le poids des en-têtes viendra s'ajouter au poids des contenus transférés. Il s'agit ici d'une modélisation simple des communications, permettant une étude analytique et par simulation et facilitant là encore l'identification des effets du routage. Nous relâcherons une

partie des contraintes dans le chapitre 7 dans lequel nous nous focaliserons sur l'impact du protocole de transport.

### 5.1.2 Critère de performance

Dans le cadre de cette étude, nous avons choisi le temps de téléchargement des contenus ou FCT (*Flow Completion Time*) comme critère de performance. Celui-ci est défini par le temps écoulé entre l'envoi de la requête par l'utilisateur jusqu'à la réception complète du contenu.

Le temps de téléchargement des contenus est une métrique qui possède plusieurs caractéristiques intéressantes. Tout d'abord, c'est une métrique dont la minimisation reflète à la fois la minimisation de la latence des contenus sensibles au délai et la maximisation du débit des flux élastiques [39]. En ce sens, elle reflète la volonté des utilisateurs de voir leurs requêtes aboutir le plus rapidement possible, qu'il s'agisse d'une requête interactive ou d'un transfert de fichier.

D'autre part, c'est une métrique dont l'utilisation est appropriée à la fois sur le réseau terrestre, au faible débit, et sur le réseau satellite, au long délai. Ainsi, il est possible de comparer la performance de deux flux ayant emprunté des chemins différents. Cette comparaison n'a que peu de sens pour certaines métriques telles que le débit – un flux court pouvant avoir intérêt à emprunter le réseau terrestre en dépit de son faible débit – ou la latence, un flux vidéo ayant un intérêt certain à emprunter le chemin satellite alors que le chemin terrestre est incapable de satisfaire le débit de lecture de la vidéo.

On retrouve ainsi cette métrique dans plusieurs travaux [152] [3] [4].

### 5.1.3 Équité, Qualité de Service (QoS) et discipline de service

Comme annoncé précédemment, le débit des liens entre l'ING et l'TUG est supposé partagé équitablement entre les différents flux empruntant un même lien. L'arbitrage entre les liens sera assuré par un mécanisme d'ordonnement qui tâchera d'atteindre cette équité.

Ce choix découle de l'équité que l'on attend entre les différents flux TCP d'un système réel, thématique à laquelle se sont intéressés de nombreux travaux dans un contexte de réseau à commutation de paquets [26] [103]. D'autre part, ce choix rend le temps de traitement des flux proportionnel à leur taille [14], propriété permettant de comparer les performances du système avec plusieurs distributions de taille, en conservant tout de même la moyenne, afin d'observer ainsi l'effet du mécanisme de routage et non de la discipline de service. Cette propriété sera expliquée lors de l'étude analytique du modèle.

Pourtant, certains systèmes poursuivent volontairement un but d'iniquité à des fins, par exemple, de différenciation entre plusieurs classes de service. Notre contexte, où la taille des flux est connue, est d'autant plus propice à la mise en place d'un mécanisme d'ordonnancement inéquitable. En effet, la discipline de service SRPT (*Shortest-Remaining-Processing-Time*), servant en priorité le flux ayant le moins de données résiduelles à transmettre, donc inéquitable en faveur de ces derniers, est connue pour minimiser le temps moyen de téléchargement [136]. Cependant, cette discipline est aussi connue pour provoquer la famine des flux longs, amenant à une insatisfaction de l'utilisateur, d'où l'impopularité de cette méthode en pratique.

De manière générale, l'équité entre les flux assure une qualité de service plus stable [75]. Finalement, notamment à des fins de simplification, la différenciation de service ne sera pas traitée dans ce travail.

## 5.2 Algorithmes

Cette section vise à présenter les différentes méthodes de routage qui seront par la suite évaluées. Nous rappelons notre objectif d'observer l'intérêt apporté par la connaissance de la taille des contenus dans la décision de routage. Ainsi, la première méthode que nous présenterons n'utilisera pas cette information et servira de référence. La deuxième méthode est issue du projet BATS (*Broadband Access via integrated Terrestrial and Satellite systems*) [33] et représentera les performances de l'état de l'art. Finalement, la dernière méthode constituera notre contribution.

### 5.2.1 Routage probabiliste

Nous étudierons tout d'abord une méthode ne faisant pas intervenir la taille des contenus à des fins de référence. La décision de routage sera prise aléatoirement ; les flux seront routés par le réseau terrestre avec une probabilité  $p_{\text{ter}}$  ou par le réseau satellite avec une probabilité  $p_{\text{sat}}$ . Nous étudierons également le choix de ces probabilités respectives pour cette méthode.

Si les caractéristiques du système sont connues (loi de trafic, latence et débit des chemins), nous verrons qu'il est possible de choisir des probabilités garantissant des performances moyennes optimales. Bien sûr, cette optimalité s'entend uniquement en l'absence de distinction des flux ou de connaissance sur l'état du système.

### 5.2.2 Routage par seuil

Cette deuxième méthode a déjà été abordée dans le projet BATS [33]. Elle consiste à fixer un seuil  $s$  ; les contenus dont la taille dépasse ce seuil sont alors routés par le réseau satellite tandis que les flux de taille inférieure sont routés par le réseau terrestre. Il s'agit donc d'une méthode faisant intervenir

la taille des flux dans le processus de routage. C'est à notre connaissance la seule à utiliser cette information. En effet, la taille des contenus est généralement inconnue expliquant le faible intérêt pour ce genre de stratégie.

Cette méthode est motivée par l'observation que les contenus courts sont en général plus sensibles au délai (par exemple, le trafic interactif) alors que les contenus les plus longs nécessitent, en général, un débit plus important (par exemple, une vidéo ou le téléchargement d'un gros fichier). On retrouve cette idée dans certains algorithmes d'ordonnancement donnant une priorité aux paquets de faible taille (ACK, commande SSH interactive...).

Les performances de cette méthode résultent directement du choix du seuil de routage. Là encore, nous verrons qu'il est possible, connaissant les caractéristiques du trafic et du réseau, de déterminer un seuil donnant des performances moyennes optimales pour cette méthode. Cela ne garantit cependant pas l'optimalité de la méthode elle-même, d'autres approches pouvant faire un meilleur usage de la taille d'une part, ou pouvant tirer profit de la connaissance de l'état du système d'autre part, ce qui n'est pas le cas ici.

### 5.2.3 MinFCT : *Minimum Flow Completion Time*

La méthode MinFCT vise explicitement à minimiser le temps moyen de téléchargement des contenus. Ainsi, lorsqu'une nouvelle demande de contenu arrive, MinFCT estime, pour chaque chemin, le temps de téléchargement nécessaire à la transmission du contenu. Le chemin, pour lequel l'estimation du temps de téléchargement est la plus faible, est alors sélectionné.

Les performances de cette méthode subiront l'impact d'une part de la précision de l'estimation du temps de téléchargement, et d'autre part des conséquences d'une telle décision sur les autres flux. Nous allons à présent présenter la méthode d'estimation du temps de téléchargement d'un contenu de taille  $s$  sur un chemin  $i$ .

Notre modèle suppose les délais et les capacités des liens connus ; d'autre part la taille d'un contenu est portée à la connaissance de l'ING par l'architecture CDN. Ainsi, le temps de téléchargement  $t(i)$  d'un contenu sur le chemin  $i$  peut être estimé par la formule suivante :

$$\begin{aligned} \text{fct}(i) &= d_{\text{ter}} + \frac{s}{\frac{c_i}{n_i+1}} + d_i \\ &= d_{\text{ter}} + (n_i + 1) \cdot \frac{s}{c_i} + d_i \end{aligned} \tag{5.1}$$

avec :

- $d_{\text{ter}}$  le délai du chemin terrestre (propagation de la requête) ;
- $d_i$  le délai du chemin  $i$  (propagation de la réponse) ;
- $n_i$  le nombre de flux actuellement assignés au chemin  $i$  ;

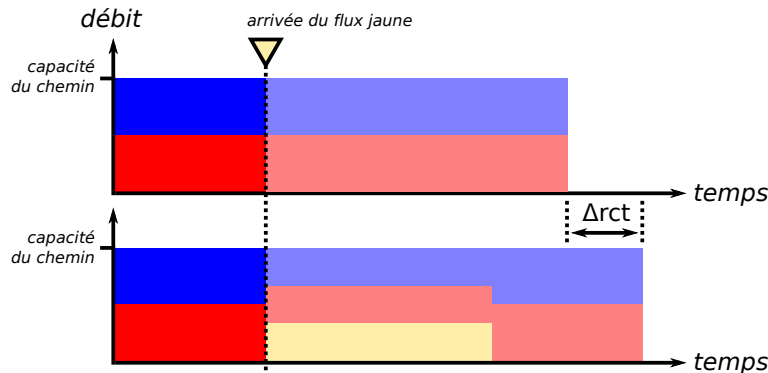


FIGURE 5.2 – L’affectation du flux jaune au même lien augmente le temps résiduel de téléchargement des flux rouge et bleu.

- $s$  la taille du flux, *i.e.* la taille du contenu et du coût des en-têtes ;
- $c_i$  la capacité du chemin  $i$ .

La capacité du chemin étant équitablement répartie entre les différents flux, le nouveau flux obtient une capacité utile de  $\frac{c_i}{n_i+1}$  d’où un temps d’émission de  $(n_i + 1) \cdot \frac{s}{c_i}$ .

À noter que le nombre de flux partageant le même chemin peut varier au cours du temps, à la baisse comme à la hausse. La connaissance de la taille des flux permet de prédire les moments de départ des flux ; cependant les arrivées sont par nature aléatoires. Certains travaux visent à prédire le trafic futur par apprentissage du trafic passé [19]. Cependant, ce type de méthode est d’une part d’une efficacité toute relative [12], et d’autre part difficilement envisageable pour un réseau familial faisant intervenir peu d’acteurs, et pour lequel la présence d’un seul flux supplémentaire peut suffire à faire basculer une décision de routage et dont l’échelle de temps nécessaire à la prédiction est très faible (durée de vie du flux). Finalement, notre estimation sera conduite en supposant le nombre de flux concurrents constant pour la durée de vie du flux, bien qu’il s’agisse là d’une approximation.

Le choix d’affectation d’un nouveau flux a pour conséquence de pénaliser les flux déjà affectés au même chemin, réduisant leur capacité individuelle. La figure 5.2 représente une telle situation où l’arrivée du flux jaune rallonge le temps de téléchargement des flux rouge et bleu (qui sont supposés ici se terminer en même temps).

Afin de prendre en compte cet accroissement qui se répercute sur le temps moyen de téléchargement des contenus, celui-ci est estimé pour chaque flux  $f$  en cours d’émission. Cette diminution de la capacité individuelle peut être momentanée ou s’étendre jusqu’au départ du flux, selon que la quantité de données résiduelle à transmettre  $r_f$  est plus petite ou plus grande que la taille  $s$  du nouveau flux sur le point d’être routé. Ainsi, si un nouveau flux



de taille  $s$  venait à être affecté au même chemin qu'un flux existant  $f$ , le temps de téléchargement restant ou RCT (*Remaining Completion Time*) du flux  $f$  s'exprimerait ainsi :

$$\text{rct}_f(i) = (n_i + 1) \cdot \frac{\min(r_f, s)}{c_i} + n_i \cdot \frac{r_f - \min(r_f, s)}{c_i} + d_i \quad (5.2)$$

L'augmentation du RCT du flux serait donc égale à :

$$\Delta \text{rct}_f(i) = \frac{\min(r_f, s)}{c_i} \quad (5.3)$$

Finalement, on ajoute ces temps supplémentaires au FCT estimé du nouveau flux :

$$\text{fct}^*(i) = d_{\text{ter}} + (n_i + 1) \cdot \frac{s}{c_i} + d_i + \sum_{f=0}^{n_i} \frac{\min(r_f, s)}{c_i} \quad (5.4)$$

Le chemin, noté  $a$ , sélectionné par la méthode est alors celui qui minimise cette quantité :

$$a = \underset{i}{\text{argmin}} \{ \text{fct}^*(i) \} \quad (5.5)$$

### 5.3 Étude analytique des performances

Le modèle proposé peut être représenté par un réseau de files d'attente. Les deux premières méthodes (probabiliste et par seuil) donneront lieu à une étude analytique des performances. La méthode MinFCT ne pourra malheureusement pas être évaluée ainsi et sera donc évaluée par simulation et expérimentation dans les sections suivantes.

#### 5.3.1 Réseau de files d'attente

La représentation du modèle sous forme d'un réseau de files d'attente, fermé, est schématisée sur la figure 5.3. Ce réseau est composé de 6 files :

- La file  $A$  représente le temps de réflexion des utilisateurs entre deux requêtes. Il s'agit d'une file à délai, comportant un nombre infini de serveurs.
- La file  $B$  représente le temps de propagation de la requête de l'utilisateur par le réseau terrestre. Il s'agit également d'une file à délai, de temps de service constant égal au délai du réseau terrestre.

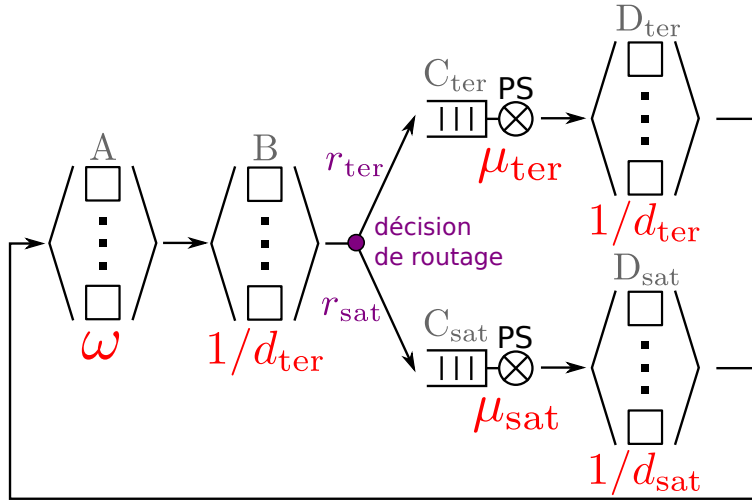


FIGURE 5.3 – Représentation du scénario par un réseau de files d'attente.

- Les files  $C_{\text{ter}}$  et  $C_{\text{sat}}$  représentent le temps de transmission des contenus vers l'utilisateur, respectivement par le réseau terrestre et par le réseau satellite. Il s'agit de files comportant un seul serveur avec une discipline de service *Processor Sharing* (PS), modélisant le partage équitable du lien entre les différents flux [15]. Le temps moyen de service est égal à la longueur moyenne des flux divisé par la capacité du lien.
- Finalement, les files  $D_{\text{ter}}$  et  $D_{\text{sat}}$  représentent le temps de propagation des contenus jusqu'à l'utilisateur. Il s'agit là encore de files à délai, de temps de service constant égal au délai des liens respectifs qu'elles représentent.

Nous considérons un routage sans mémoire, les quantités  $r_{\text{ter}}$  et  $r_{\text{sat}}$  indiquent les ratios d'utilisateurs empruntant respectivement le chemin terrestre et le chemin satellite. Cette représentation ne s'applique donc qu'à la méthode probabiliste et à la méthode par seuil ; la méthode de routage MinFCT n'étant pas sans mémoire de par l'utilisation de la quantité  $n_i$  dans l'estimation du temps de téléchargement des flux.

Avec la méthode probabiliste, on a trivialement  $r_{\text{ter}} = p_{\text{ter}}$  et  $r_{\text{sat}} = p_{\text{sat}}$ . Avec la méthode par seuil, les proportions de flux routés par les réseaux terrestre et satellite peuvent être reliées à la fonction de répartition de la taille des flux  $F_S$ . On a ainsi, pour un seuil de routage  $b$ ,  $r_{\text{ter}} = F_S(b)$  et  $r_{\text{sat}} = 1 - F_S(b)$ .

L'utilisation de la discipline de service PS pour les files  $C$  constitue une simplification du modèle, celle-ci ne reproduisant pas l'effet de la paquetsation des contenus. Nous prendrons tout de même en compte la taille des en-têtes par une augmentation du temps de service nécessaire à leur émission. Dans nos simulations, nous nous affranchirons des approximations liées

à l'utilisation du modèle PS en mettant en œuvre une simulation de paquets.

### 5.3.2 Analyse des valeurs moyennes (MVA)

La méthode MVA (*Mean-Value Analysis*) est une technique permettant de calculer les temps de réponse, nombres moyens de clients et débit moyen de chaque file dans un réseau de files d'attente fermé. Elle est fondée sur la propriété de SEVCIK-MITRANI [131] et la loi de LITTLE [138]. Cette technique numérique procède par récurrence sur le nombre d'utilisateurs dans le réseau.

Le théorème BCMP (BASKETT, CHANDY, MUNTZ, PALACIOS) [11] permet d'exprimer, pour chaque file, la distribution stationnaire du nombre d'utilisateurs et les temps moyens de réponse. Cependant, les contraintes sont plus élevées, MVA pouvant s'appliquer dans certains réseaux sans forme produit alors que BCMP y est restreint. D'autre part, l'expression analytique de la distribution de l'état des files, si elle existe, a de fortes chances d'être difficilement exploitable.

Nous supposons que le temps d'attente entre deux requêtes suit une loi exponentielle. La distribution de la taille des objets Web est connue pour avoir un corps log-normale mais une queue lourde, modélisable par une loi dPIN (*double Pareto log-normale*) [124] [98] [97]. Les temps de propagation des liens et leur capacité seront supposés constants.

Avec le routage probabiliste, les temps de service des files  $C$  suivent une loi similaire à la taille des flux, à un facteur multiplicatif près, inversement proportionnel à leur capacité respective. Avec le routage par seuil, les temps de service des files  $C$  suivent, cette fois-ci, une troncature de la loi de probabilité suivie par la taille des flux, toujours à un facteur près. Les temps de service des files  $C$  suivent donc des lois générales.

En toute rigueur, BCMP impose pour les files à délai et celles de type  $\cdot/G/1/PS$  des temps de service dont la transformée de LAPLACE de la densité de probabilité est rationnelle, ce qui permet de se ramener à des lois de Cox et un découpage en étages. On a l'intuition que ces résultats peuvent être étendus à des lois quelconques. Par exemple, [151] montre l'insensibilité des files  $M/G/1/PS$  en relâchant la contrainte précédente ce qui conforte notre hypothèse sur les files  $C$ .

### 5.3.3 L'algorithme MVA

L'algorithme démarre avec un système vide, la taille de chaque file étant initialisée à 0. À chaque itération, le nombre de clients  $m$  dans le système est incrémenté d'une unité, l'algorithme s'arrête lorsque le nombre d'utilisateurs requis est atteint. Lors d'une itération, sont calculées les valeurs suivantes :

1. Pour chaque file  $k$ , son temps de réponse moyen  $\overline{R}_k(m)$ . Celui-ci est calculé comme la somme du temps d'attente moyen dans un système

avec  $m - 1$  clients en appliquant le théorème de SEVCIK-MITRANI, et du temps de service moyen du  $m$ -ième client :

$$\overline{R}_k(m) = \frac{\overline{L}_k(m-1) + 1}{\mu_k} \quad (5.6)$$

avec  $\overline{L}_k(m-1)$  le nombre moyen de client dans la file  $k$ .

Dans le cas des files à délai, il n'y a pas de temps d'attente et le temps de réponse moyen est directement égale à  $1/\mu_k$ .

2. Le débit du système, à partir de la loi de LITTLE :

$$\lambda(m) = \frac{m}{\sum_{k=1}^K r_k \overline{R}_k(m)} \quad (5.7)$$

où  $r_k$  est le nombre moyen de visites de la file  $k$  pour un client entre deux passages dans la file  $A$ .

3. Le nombre de clients moyen  $\overline{L}_k(m)$  dans chaque file  $k$ , grâce à la loi de LITTLE appliquée à chaque file :

$$\overline{L}_k(m) = r_k \lambda(m) \overline{R}_k(m) \quad (5.8)$$

### 5.3.4 Implantation de la méthode MVA

Nous avons implanté la méthode MVA en python afin d'obtenir des résultats numériques pour les deux premières méthodes de routage.

## 5.4 Étude des performances par simulation

Afin de valider ces premiers résultats mathématiques, et d'étudier la méthode MinFCT pour laquelle MVA ne s'applique pas, nous avons mis en œuvre un simulateur en python. Il s'agit d'un simulateur à événements discrets, fondé sur la bibliothèque SimPy, et reproduisant le comportement au niveau paquet. Le calcul des temps d'émission des paquets tient compte du poids des en-têtes de ces derniers.

Ce simulateur nous a ainsi permis d'obtenir des résultats pour les trois méthodes de routage. Il nous permettra notamment de vérifier, pour les deux premières méthodes de routage, la validité de l'utilisation de MVA, et réciproquement l'adéquation de l'ordonnanceur proposé avec le modèle PS.

### 5.4.1 Ordonnancement

Le fait de supposer les liens entre ING et IUG sans perte et de débit connu, ainsi que l'utilisation d'un protocole sans connexion comme UDP a quelques conséquences non négligeables. Tout d'abord, la limitation de la capacité des liens est reproduite par une limitation par l'ING de son

débit d'émission des datagrammes UDP. D'autre part, il incombe à l'ING de respecter l'équité en termes de débit entre les différents flux. Ainsi, lorsqu'un lien devient disponible (*i.e.* son utilisation pour l'émission d'un nouveau paquet respecte son débit supposé), l'ING doit choisir le flux à servir parmi ceux affectés au lien.

NAGLE fut l'un des premiers à s'intéresser à l'équité dans les réseaux, et publia en 1985 la RFC 970 [103] introduisant le principe de *fair queuing*. Il propose un premier mécanisme dans ce but d'équité : la file FCFS (*First-Come First-Served*) présente jusque-là dans les routeurs est remplacée par un ensemble de files FCFS, une par flux, et chaque file est servie alternativement, à la manière d'un tourniquet (*Round-Robin*). Cette proposition permet d'assurer une équité entre les flux en termes de paquets mais s'avère inéquitable en débit dès lors que les paquets sont de longueur variable. L'algorithme BBR (*Bit-by-Bit Round-Robin*) pallie cette limitation en servant un unique bit à chaque tour, émulant ainsi parfaitement l'équité du modèle *Processor Sharing* (PS), mais son implantation est irréaliste.

Finalement, plusieurs mécanismes réalisables ont été proposés, tels que SFQ (*Stochastic Fairness Queueing*) [96] ou PFQ (*Packet-by-Packet Fair Queueing*) [37]. Notre propre algorithme est inspiré des travaux de GOLESTANI [57] qui propose une implantation très efficace reposant sur l'introduction d'un temps virtuel. Chaque paquet, à son arrivée, se voit attribuer un temps virtuel de fin de service calculé à partir de l'instant d'arrivée et du temps d'attente estimé, compte tenu des autres paquets présents et de l'équité à atteindre. Finalement, le paquet servi en premier est celui avec le temps virtuel de fin de service le moins élevé.

Notre implantation vise à s'adapter à notre cas où l'ING opère au niveau applicatif et ne considère pas des paquets mais uniquement des flux. Ainsi, chaque flux se voit attribuer un *temps virtuel de dernier service*, initialisé à l'instant d'arrivée du flux. Dès qu'un lien est disponible, on calcule pour chaque flux le temps virtuel de fin de service pour son prochain paquet :

$$t_{\text{fin de service}} = t_{\text{dernier service}} + n \cdot \frac{s}{c}$$

où  $n$  est le nombre de flux concurrents,  $s$  la taille du paquet et  $c$  la capacité du lien. Le temps d'émission  $\frac{s}{c}$  du paquet est multiplié par  $n$  afin d'obtenir le temps de séjour virtuel nécessaire pour ce paquet compte tenu de la discipline de service PS.

Le flux servi est alors celui qui minimise ce temps virtuel de fin de service. Son temps virtuel de dernier service est augmenté du temps de séjour du paquet servi. Ainsi, l'équité entre les flux est garantie, quelle que soit la taille des paquets.

## 5.5 Étude des performances par émulation

Finalement, nous avons mis en œuvre une plateforme d'expérimentation nous ayant permis d'obtenir des résultats là encore pour les trois méthodes de routage. Son comportement est très similaire au simulateur, mais l'exécution se déroule en temps réel et utilise la pile réseau (UDP/IP) du noyau Linux.

### 5.5.1 Architecture

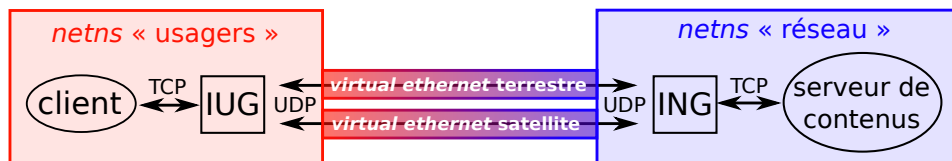


FIGURE 5.4 – Architecture du banc de test.

Cette plateforme d'expérimentation comprend d'une part la réalisation de quatre composants développés en C et d'autre part l'utilisation des outils réseau de Linux pour interconnecter ces derniers.

Ainsi, nous avons développé :

- un client, qui reproduit l'activité des utilisateurs ;
- deux serveurs mandataires intelligents, l'ING et l'IUG ;
- un serveur de contenus.

Ces composants sont détaillés dans la section qui suit ; plus d'informations sur les choix d'implantation sont données dans l'annexe A. Leur code source est disponible sur la forge interne du CNES.

Ces quatre composants ont été déployés sur la même machine. Nous avons utilisé la fonctionnalité des `network namespace` de Linux pour les isoler.

Ainsi, nous avons deux `netns` que l'on nommera respectivement « usagers » et « réseau ». Dans le `netns` « usagers » se trouvent le client et le serveur mandataire utilisateur (IUG) ; dans le `netns` « réseau » se trouvent le serveur mandataire réseau (ING) et le serveur de contenus. Les deux `netns` sont reliés par deux liens distincts (`veth` ou *virtual ethernet*) émulant les liens d'accès terrestre et satellite. La latence de ces derniers est émulée par l'utilisation de la discipline de service (`qdisc`) *network emulator* (`netem`). Le débit d'émission est limité logiciellement directement par l'ING.

### 5.5.2 Client

Le client émule l'activité de  $n$  utilisateurs par l'exécution de  $n$  threads en parallèle. Chaque thread implante le cycle attente-requête-réponse présenté précédemment.

Lors d'une requête, le client se connecte à l'IUG en TCP, localement puisqu'ils partagent le même `netns`. Il utilise le protocole SOCKS5 pour indiquer l'adresse du serveur de contenus qu'il souhaite atteindre (bien qu'on n'en considère qu'un seul ici). Il transmet alors sa requête, en utilisant un protocole simplifié se résumant à l'envoi au format texte de la taille supposée du contenu à récupérer.

### 5.5.3 Serveurs mandataires intelligents

L'IUG implante un serveur SOCKS5, recevant les requêtes du client. Il transmet ces dernières dans un datagramme UDP à destination de l'ING, via le lien terrestre. Chaque datagramme comporte un petit en-tête avec un numéro de flux et un numéro de séquence. Le numéro de séquence numéro 0 est réservé à la transmission d'informations sur le flux nouvellement créé (IP et port de destination indiqué par SOCKS).

L'ING, à la réception d'une requête, se connecte alors en TCP (localement) au serveur de contenus afin de récupérer l'objet demandé. En analysant la requête transmise par ce protocole simplifié, l'ING connaît alors la taille du contenu. L'ING choisit le lien le plus approprié pour envoyer le contenu conformément à la méthode de routage que l'on vise à tester. L'algorithme d'ordonnancement visant à garantir l'équité entre les différents flux est une réimplantation en C de celui décrit dans la section précédente.

### 5.5.4 Serveur

Le serveur de contenus se contente de transmettre, lorsqu'une requête est reçue, autant d'octets (aléatoires) que demandés et ferme la connexion établie par l'ING. Bien évidemment, plusieurs contenus peuvent être servis en parallèle.

## 5.6 Résultats

Nous présentons ici les résultats obtenus analytiquement, par simulation et par expérimentation sur notre banc de test.

### 5.6.1 Paramètres

Le temps d'attente entre deux requêtes suit une loi exponentielle de moyenne une seconde. Le lien terrestre a une capacité de 2 Mbit/s et un délai de 25 ms tandis que le lien satellite a une capacité de 8 Mbit/s et un délai de 300 ms.

Les paquets échangés entre l'IUG et l'ING ont une charge utile de 1450 octets pour une taille totale de 1518 octets. Ainsi, en simulation, l'émission d'un contenu de 500 octets sera supposée prendre le temps nécessaire à

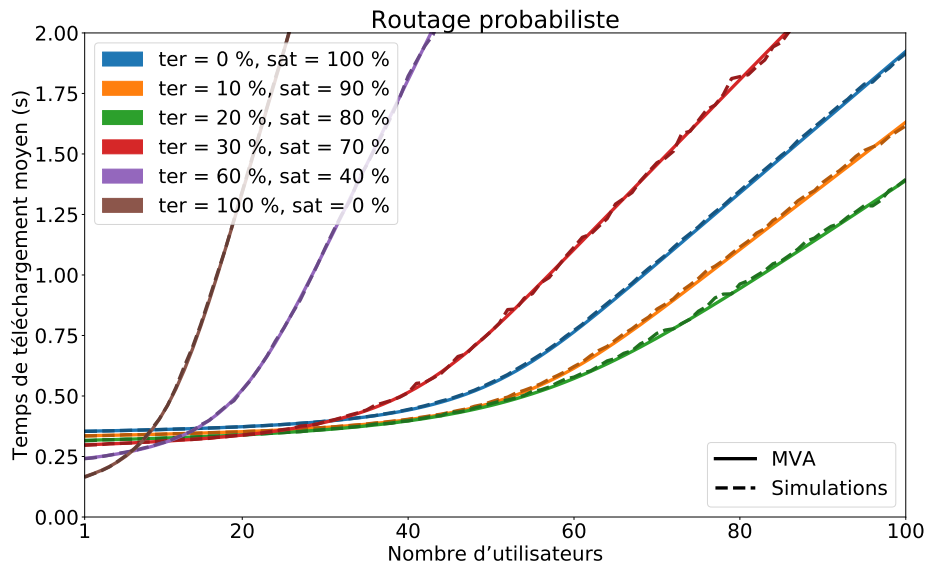


FIGURE 5.5 – Temps de téléchargement moyen en fonction du nombre d'utilisateurs pour différentes probabilités de routage.

l'émission de 568 octets au débit donné du lien. Pour la partie expérimentation, nous enverrons 1450 octets de données dans un datagramme UDP qui ne sera réellement envoyé qu'après une période d'attente correspondant à l'émission de 1500 octets au débit du lien. L'émission réelle est supposée instantanée, le lien Ethernet virtuel ayant une capacité réelle illimitée.

La taille des contenus suit une loi double Pareto-lognormale de paramètres  $\alpha = 2.5$ ,  $\beta = 1.5$ ,  $\nu = 9.5$  et  $\tau = 1.5$ . On a ainsi des contenus d'une taille moyenne de 28 Ko avec un écart-type de 83 Ko.

### 5.6.2 Routage probabiliste

La figure 5.5 représente le temps moyen de téléchargement en fonction du nombre d'utilisateurs dans le système, et ce pour différents jeux de probabilité de routage entre le réseau terrestre et le réseau satellite. Chaque courbe présente une zone relativement plate, à gauche, puis une augmentation linéaire à forte charge, à droite. Cette augmentation s'explique par la saturation du réseau terrestre et / ou satellite due au nombre d'utilisateurs présents. Cette saturation apparaît plus ou moins tôt selon le choix des probabilités de routage, témoignant de l'importance de ces paramètres.



### Analyse à faible charge

Compte tenu de la distribution des tailles de flux, ceux-ci pèsent, en moyenne et en-têtes inclus, 29 237 octets. Le temps de téléchargement moyen, pour un unique utilisateur dans le système (le temps de séjour étant alors égal à la somme des temps de service), est donc de :

— sur le réseau terrestre (courbe marron) :

$$0.025 + \frac{8 \times 29237}{2000000} + 0.025 = 50\text{ms de RTT} + 117\text{ms d'émission} = 167\text{ms}$$

— sur le réseau satellite (courbe bleue) :

$$0.025 + \frac{8 \times 29237}{8000000} + 0.300 = 325\text{ms de RTT} + 29\text{ms d'émission} = 354\text{ms}$$

Le temps d'émission sur le réseau terrestre est, malgré son plus faible débit, bien plus faible que le délai du réseau satellite, expliquant les meilleures performances sur le premier.

On retrouve ces observations sur la courbe bleue (1 utilisateur) de la figure 5.6, représentant le temps moyen de téléchargement en fonction de la probabilité de routage sur le réseau terrestre.

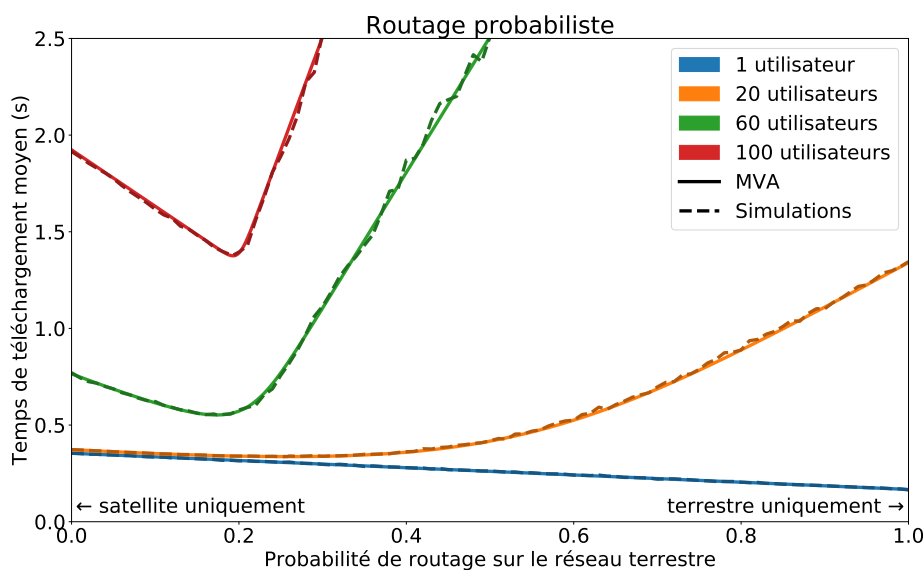


FIGURE 5.6 – Temps de téléchargement moyen en fonction de la probabilité de routage par le réseau terrestre.

### Analyse à forte charge

Lorsque le nombre d'utilisateurs augmente, la capacité des liens se trouve alors partagée entre plusieurs flux augmentant mécaniquement le temps de téléchargement de ces derniers. À partir d'un certain point, les réseaux terrestre et / ou satellite se trouvent totalement congestionnés. Ceci est obser-

vable sur la figure 5.7 représentant le taux d'utilisation des liens satellite et terrestre calculé à partir des résultats de la méthode MVA.

On observe que le jeu de probabilités (0.2, 0.8), égal au ratio des débits des liens, amène une congestion conjointe des réseaux alors que dans les autres cas, un seul des réseaux est congestionné, l'autre étant alors sous-utilisé. De ce fait, c'est également le jeu de probabilités donnant les meilleures performances à forte charge comme en témoigne la courbe « 100 utilisateurs » de la figure 5.6.

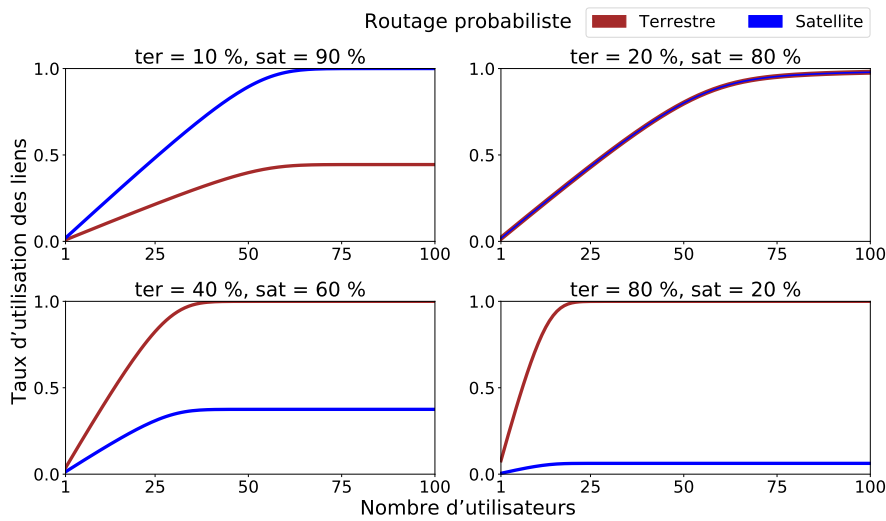


FIGURE 5.7 – Taux d'utilisation des liens satellite et terrestre calculé à partir de MVA.

Lorsque qu'un chemin  $i$  de capacité  $C_i$  est saturé, le temps de séjour  $\tau_i$  dans la file d'émission est alors proportionnel au nombre de flux  $n_i$  sur ce chemin. On peut alors calculer le débit du système  $\lambda$  :

$$\tau_i = n_i \frac{\bar{s}}{C_i} \stackrel{\text{LITTLE}}{=} p_i \lambda \tau_i \frac{\bar{s}}{C_i} \Leftrightarrow \lambda = \frac{C_i}{p_i \bar{s}}$$

avec  $p_i$  la proportion des flux routés sur le chemin  $i$ . Le débit étant maximal lorsque les deux liens sont saturés, c'est-à-dire lorsque  $\frac{C_t}{p_t \bar{s}} = \frac{C_s}{p_s \bar{s}}$ , on retrouve bien que le jeu de probabilité optimal est donné par  $\frac{p_t}{p_s} = \frac{C_t}{C_s}$ .

D'autre part, connaissant le réseau saturé et donc le débit du système, on peut calculer les asymptotes que l'on observe sur la figure 5.5. Il s'agit de calculer le temps de séjour  $\tau_S$  dans le sous-système  $S$  composé des files  $B$ ,  $C$  et  $D$  de la figure 5.3. Pour cela, on applique la loi de LITTLE à la file  $A$  pour obtenir le nombre moyen d'utilisateurs  $n_A = \lambda \omega$ . On en déduit le

nombre d'utilisateurs  $n_S = N - n_A$  dans le sous-système  $S$  et en appliquant la loi de LITTLE à ce dernier, on obtient :

$$\tau_S = \frac{n_S}{\lambda} = \frac{N - \lambda\omega}{\lambda} = \text{fct}(N) \text{ avec } \lambda = \frac{1}{\bar{s}} \cdot \min\left(\frac{C_t}{p_t}, \frac{C_s}{p_s}\right)$$

qui est bien une fonction affine du nombre d'utilisateurs  $N$ .

### Conclusion sur le routage probabiliste

Au vu des résultats précédents, il apparaît plus avantageux d'envoyer les flux sur le réseau terrestre à faible charge, et au contraire sur le réseau satellite à forte charge. On peut observer ceci sur le premier graphique de la figure 5.8 représentant les paramètres optimaux pour une charge donnée (*i.e.* pour un nombre d'utilisateurs donné).

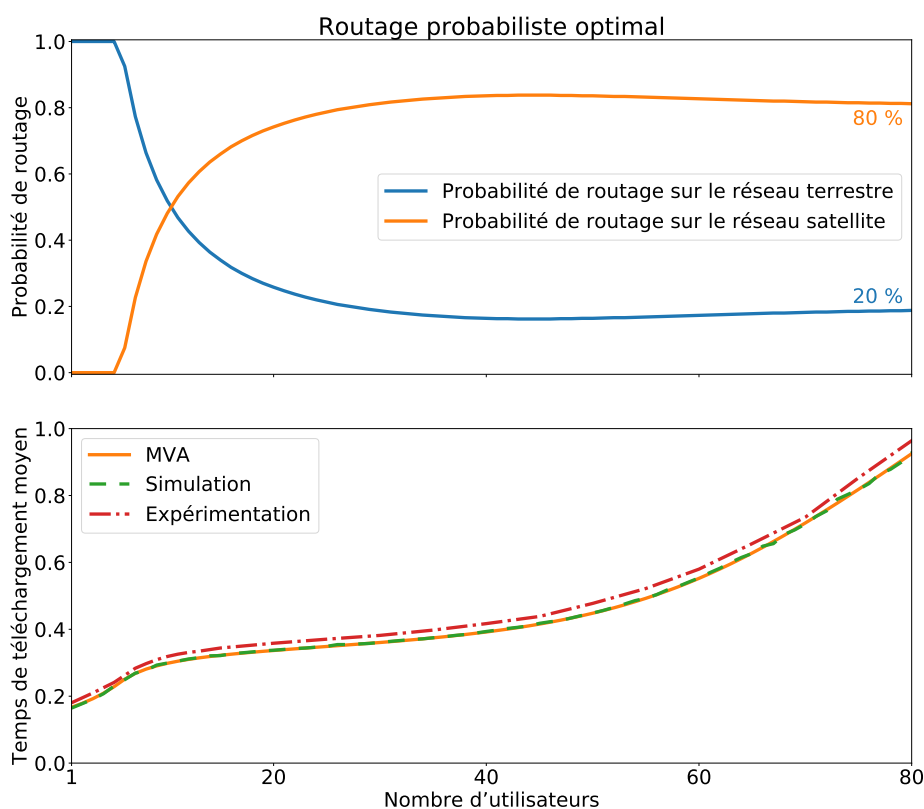


FIGURE 5.8 – Probabilités optimales selon MVA et temps de téléchargement moyen associé obtenu par MVA, simulations et expérimentations.

Le graphique ci-après représente le temps de téléchargement moyen associé. Nous représentons également sur cette dernière figure les résultats

obtenus par expérimentation. Ces derniers sont un peu plus élevés en raison d'épiphénomènes négligés par notre modèle et nos simulations : temps de connexion de l'utilisateur à l'IUG, de l'ING au serveur de contenu, temps de calcul. . . On obtient tout de même un comportement très fidèle à MVA et aux simulations.

On constate ainsi la saturation du réseau terrestre dès la présence d'une dizaine d'utilisateurs dans le système, amenant alors à aiguiller rapidement la majorité du trafic sur le réseau satellite. Finalement ce deuxième réseau se trouve à son tour congestionné aux alentours d'une soixantaine d'utilisateurs.

### 5.6.3 Routage par seuil

La figure 5.9 représente le temps de téléchargement moyen en fonction du nombre d'utilisateurs, et ce pour différents seuils. On retrouve des courbes aux allures similaires aux résultats obtenus précédemment avec la méthode de routage probabiliste. Lorsque le seuil est faible, l'essentiel du trafic est routé sur le réseau satellite et on observe alors un temps de téléchargement moyen borné par le délai de ce dernier. En revanche, lorsque le seuil est plus élevé (50 Ko ou plus), une partie du trafic profite du faible délai du réseau terrestre, abaissant le temps de téléchargement moyen à faible charge, mais amenant également à une rapide congestion de ce dernier avec l'augmentation de la charge.

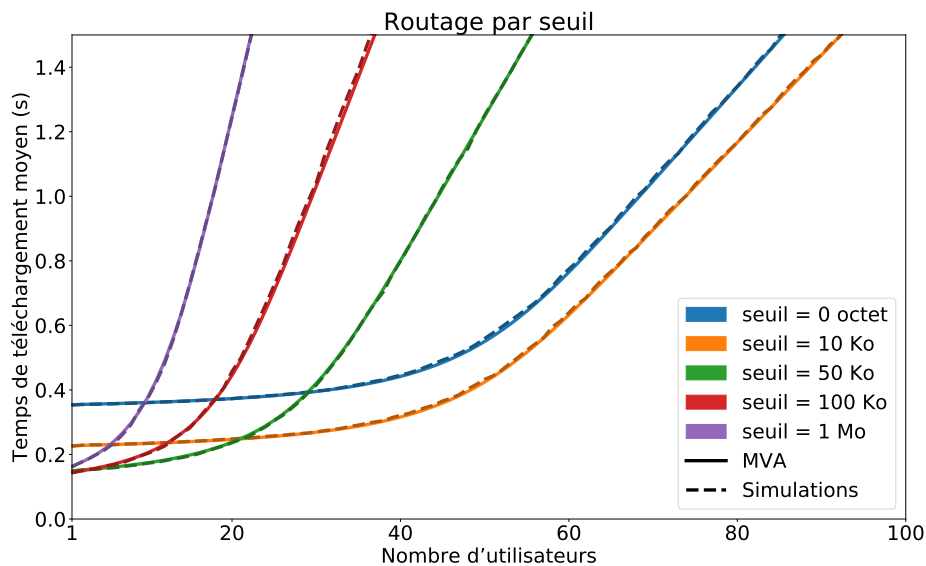


FIGURE 5.9 – Temps de téléchargement moyen en fonction du nombre d'utilisateurs pour différents seuils.

Sur la figure 5.10 sont représentées en fonction du seuil :

- en haut, la proportion des flux routés sur les réseaux terrestre et satellite (*i.e.* la fonction de répartition de la loi dPIN et son complémentaire);
- en bas, les performances de la méthode en termes de temps de téléchargement moyen.

On constate là encore qu'il s'agit de trouver, pour obtenir les meilleures performances, un juste équilibre en fonction de la charge dans la répartition du trafic entre les réseaux terrestre et satellite.

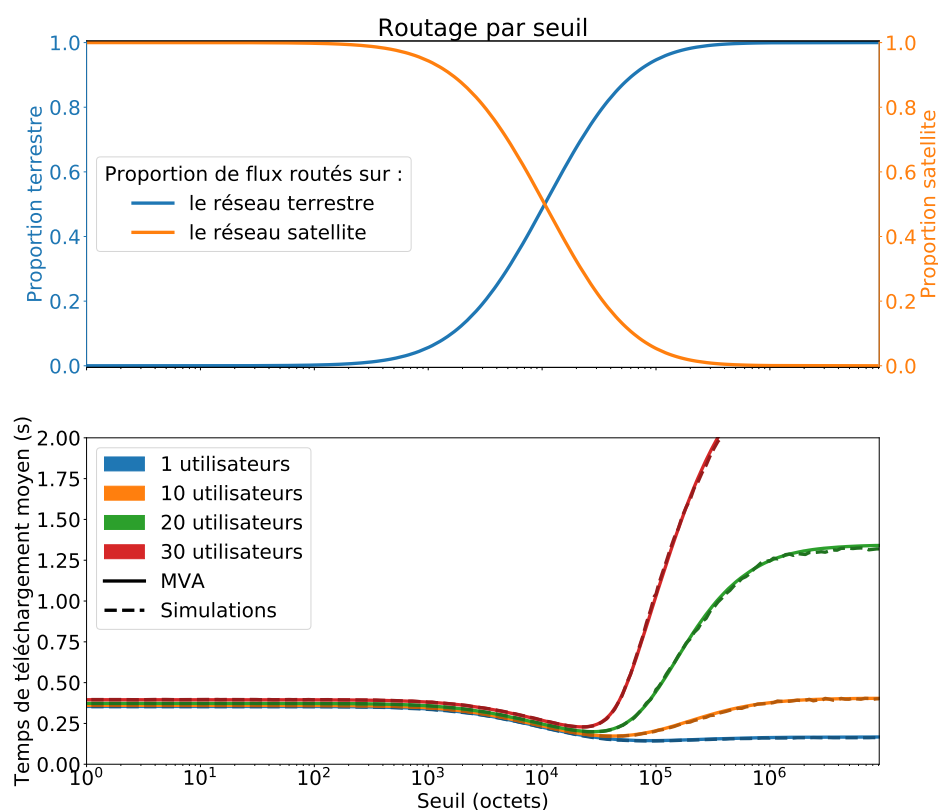


FIGURE 5.10 – Temps de téléchargement moyen en fonction du seuil.

Le premier graphique de la figure 5.11 représente à la fois le seuil optimal pour cette méthode de routage, déterminé à partir des résultats donnés par MVA, ainsi que la proportion des flux routés sur le réseau terrestre lors de l'utilisation de ce seuil. En dessous se trouve le temps de téléchargement moyen associé.

Avec un seul utilisateur, le seuil optimal s'élève à 88.6 Ko : 94% du trafic est alors routé sur le réseau terrestre et profite ainsi de son faible délai. Nous

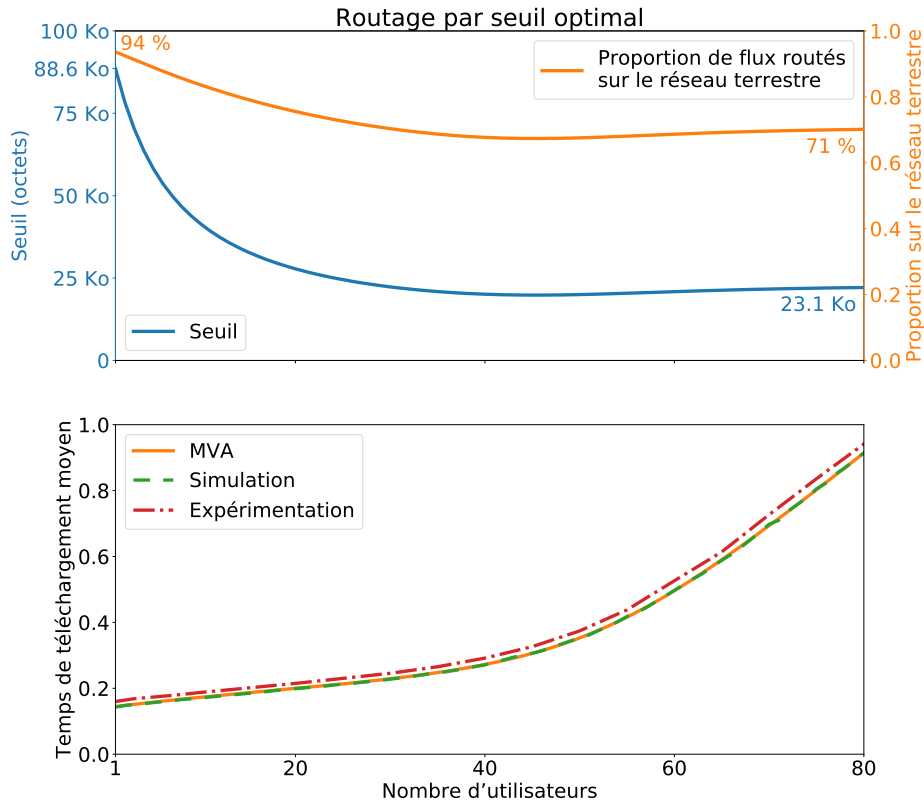


FIGURE 5.11 – Seuil optimal selon MVA en fonction du nombre d'utilisateurs, et temps de téléchargement moyen associé obtenu par MVA, simulations et expérimentations.

avons là une première différence avec la méthode probabiliste qui, pour sa part, transmettait 100 % du trafic sur le réseau terrestre : ici, il est possible d'identifier les flux très longs qui profiteront avantageusement du réseau satellite. Ce seuil peut être retrouvé analytiquement : c'est celui pour lequel le temps de transfert sur le réseau terrestre et sur le réseau satellite sont égaux :

$$d_t + \frac{s_f}{C_t} + d_t = d_t + \frac{s_f}{C_s} + d_s \Leftrightarrow s_f = C_s C_t \frac{d_s - d_t}{C_s - C_t}$$

avec  $s_f$  le seuil portant sur la taille des **flux**. Compte tenu des en-têtes, le seuil  $s_c$  portant sur la taille des **contenus** est égal à :  $s_c = s_f \cdot \frac{1450o}{1500o}$

**Application numérique :**

$$2\text{Mbit/s} \cdot 8\text{Mbit/s} \cdot \frac{300\text{ms} - 25\text{ms}}{8\text{Mbit/s} - 2\text{Mbit/s}} \cdot \frac{1450}{1500} = 91.6\text{Ko} \cdot \frac{1450o}{1500o} = 88.6\text{Ko}$$

### 5.6.4 MinFCT

Les résultats obtenus avec la méthode MinFCT, par simulation et émulation, sont représentés sur la figure 5.12. La droite verte représente les résultats attendus asymptotiquement en régime saturé, tel que précisé dans la section 5.6.2.

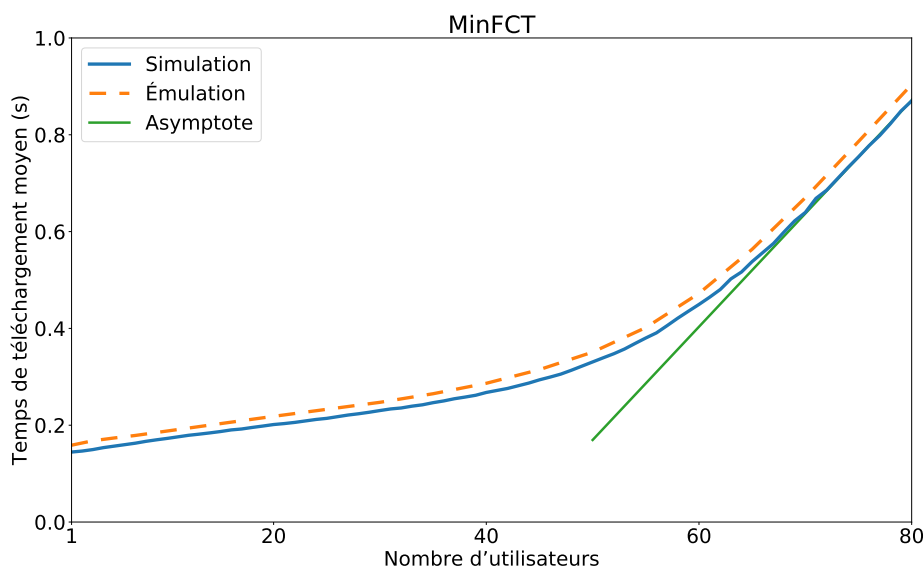


FIGURE 5.12 – Temps de téléchargement moyen en fonction du nombre d'utilisateurs dans le système pour la méthode de routage MinFCT.

On observe ainsi l'aptitude de la méthode à profiter du faible délai du réseau terrestre, mais également à correctement exploiter les deux liens disponibles à forte charge comme en témoigne la jonction avec l'asymptote.

### 5.6.5 Comparaison des trois méthodes de routage

Finalement, nous avons représenté sur la figure 5.13 une comparaison des performances obtenues, par simulation et expérimentation, avec les trois méthodes de routage étudiées.

On observe ainsi l'important gain des méthodes de routage tenant compte de la taille sur la méthode de routage probabiliste. Par ailleurs, les premières méthodes présentent des performances similaires.

Alors que la méthode probabiliste voit le réseau terrestre rapidement congestionné, les deux autres méthodes se montrent capables de limiter la congestion sur ce dernier en y aiguillant uniquement les flux les plus courts. Il en résulte qu'un plus grand nombre de flux peuvent profiter du délai avantageux du réseau terrestre, et ainsi les performances moyennes s'en trouvent

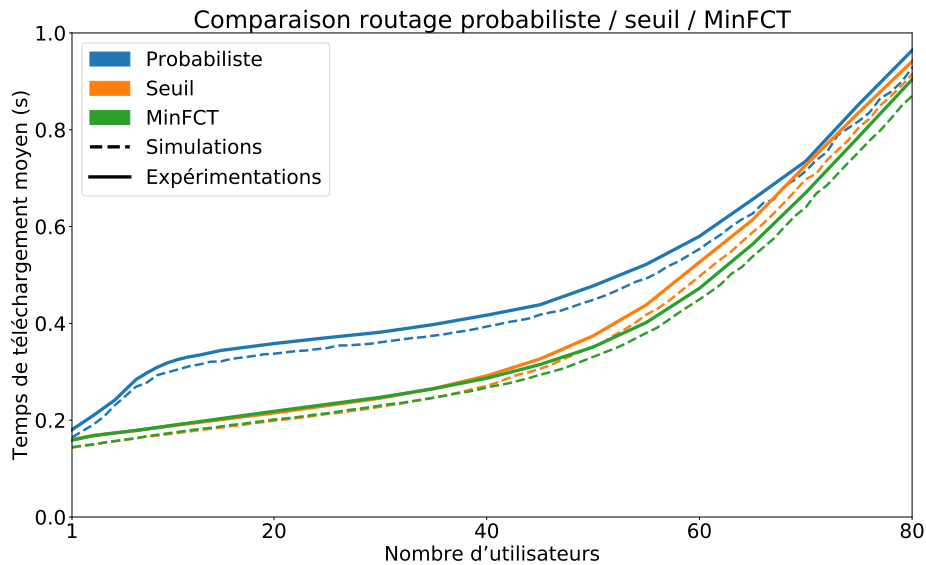


FIGURE 5.13 – Temps de téléchargement moyen en fonction du nombre d'utilisateurs dans le système pour les trois méthodes de routage étudiées.

améliorées.

Notons tout de même que pour les méthodes probabilistes et par seuil, les résultats présentés correspondent aux performances optimales atteignables en présence d'une charge connue et constante permettant le choix de paramètres optimaux. Bien que la méthode par seuil présente des performances équivalentes à la méthode MinFCT, cette dernière ne connaît pas cette importante limitation. Comme l'a montré la figure 5.9, aucune valeur de seuil ne permet des performances optimales en toute situation.

On constate également à forte charge un petit intérêt supplémentaire pour la méthode MinFCT, capable de saturer les deux liens dès 70 utilisateurs comme en témoigne la figure 5.12, alors que la méthode par seuil se rapproche plus lentement de l'asymptote. Ceci s'explique par l'aptitude de MinFCT à prendre des décisions dynamiques, en fonction de l'état courant du système, alors que la méthode par seuil est paramétrée statiquement, malgré le choix d'un seuil statistiquement optimal.





## Chapitre 6

# Mécanismes de routage à une échelle plus fine que le flux

Dans le chapitre précédent, nous avons comparé plusieurs mécanismes de routage sur un réseau hybride aux propriétés hétérogènes. Nous avons montré la supériorité des mécanismes faisant usage de la taille des contenus dans leur décision de routage.

Cependant, nous avons considéré jusque-là uniquement des mécanismes de routage travaillant au niveau des flux. Il est possible de s'affranchir de cette limitation, et de considérer la transmission d'un même contenu sur plusieurs réseaux – simultanément ou non –. Il s'agit donc de descendre à une échelle de routage plus fine, notamment au niveau des paquets.

Cette échelle de décision ouvre de nouvelles perspectives pour améliorer les performances du routage. Ce chapitre vise donc à explorer plusieurs possibilités offertes par ce niveau de décision, et à en comparer les performances avec celles obtenues dans le chapitre précédent.

### 6.1 Agrégation

Le premier mécanisme que nous étudierons dans ce chapitre est un mécanisme d'agrégation. L'agrégation est, comme en témoigne notre inventaire des méthodes de routage par paquets dans le chapitre 3, la méthode usuelle de ce niveau de décision pour profiter de la diversité de chemins qui peut exister entre une source et une destination. Elle permet, par une répartition des données d'un même flux sur plusieurs chemins, d'obtenir un débit agrégé égal à la somme des débits des chemins considérés. En revanche, les délais de bout-en-bout différeront et se traduiront par des attentes potentielles au niveau du récepteur (blocage en tête de file).

Notre mécanisme d'agrégation fonctionnera ainsi : chaque lien, lorsqu'il est disponible (*i.e.* à la fin de l'émission d'un paquet), servira alors l'un des flux actifs. L'ensemble des flux est considéré, sans aucune restriction ;

un même flux pouvant donc être servi par un lien comme par un autre. Le choix du flux à servir reposera sur l'ordonnanceur déjà décrit dans la section 5.4.1 et visant l'équité entre les flux en termes de débit.

À la réception par l'IUG, les fragments seront reséquenceés avant d'être transmis au client. Ce mécanisme, inutile au chapitre précédent, devient nécessaire en raison du déséquenceement introduit par l'utilisation, pour un même contenu, de liens ayant une forte différence de délai.

### 6.1.1 Agrégation consciente de la taille

Dans notre situation très hétérogène, les flux se trouvent tributaires du long délai du chemin passant par le satellite. Nous avons déjà pu observer dans le chapitre précédent l'intérêt du routage des flux courts sur le réseau terrestre, leur temps d'émission étant faible devant le délai du chemin satellite. On comprend donc aisément le problème posé par ce mécanisme de routage.

Ainsi, nous proposons un nouveau mécanisme d'agrégation tenant compte de la taille des contenus afin de pallier cet inconvénient. Avec ce mécanisme, le lien terrestre servira tous les flux tandis que le lien satellite ne servira que les flux les plus longs. Plus précisément, le lien satellite ne servira que les flux dont la quantité de données résiduelle est suffisamment importante pour rendre l'opération bénéfique. C'est le cas lorsque le temps de transmission résiduel sur le lien terrestre seul s'avère plus élevé que le délai du lien satellite, c'est-à-dire lorsque :

$$n \cdot \frac{r_f}{c_t} + d_t > d_s \quad (6.1)$$

avec :

- $n$  le nombre de flux en cours de transfert ;
- $r_f$  la quantité de données résiduelle à transmettre pour le flux  $f$  ;
- $c_t$  la capacité du lien terrestre ;
- $d_t$  (resp.  $d_s$ ) le délai du lien terrestre (resp. satellite).

Notons que la connaissance de la quantité de données résiduelle est possible car la taille du contenu transmis est connue.

### 6.1.2 Résultats

Nous représentons sur la figure 6.1 les performances des mécanismes d'agrégation – avec et sans la condition sur la taille des contenus – et nous les comparons aux performances de la méthode de routage par flux MinFCT. Ces résultats ont été obtenus par simulation ; les paramètres utilisés sont identiques à ceux choisis dans le chapitre précédent.

On observe comme attendu les faibles performances du mécanisme d'agrégation sans condition sur la taille des contenus. Le temps de téléchargement

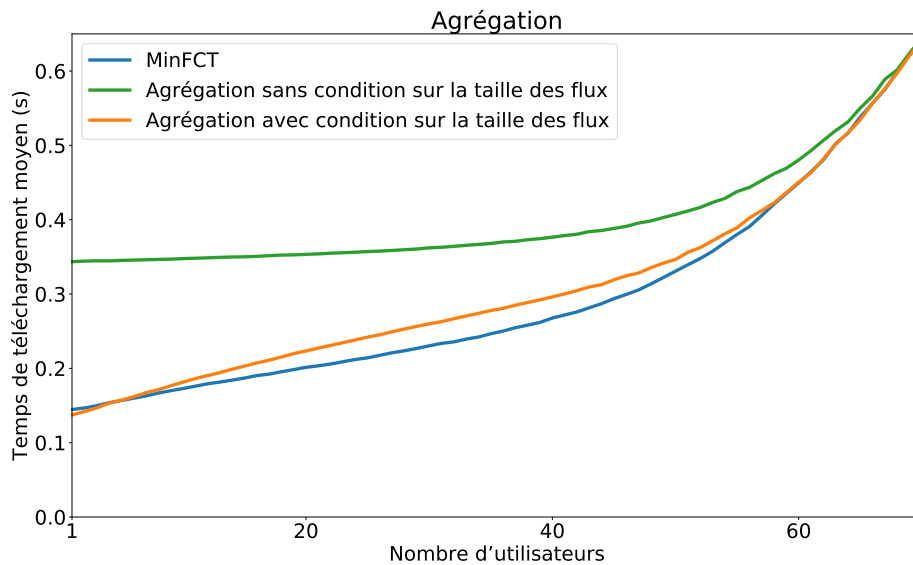


FIGURE 6.1 – Comparaison des performances des mécanismes d’agrégation avec la méthode de routage par flux MinFCT.

moyen de cette méthode ne passe en effet jamais sous le seuil de 325 ms, durée minimale due à l’emprunt au retour du chemin satellite.

En second lieu, on observe une régression des performances comparativement à la méthode MinFCT pour la méthode d’agrégation avec condition sur la taille. En effet, l’introduction de cette condition ne suffit pas à contrebalancer l’effet négatif de cette stratégie. Ses mauvaises performances s’expliquent par l’utilisation conjointe du réseau terrestre par l’ensemble des flux, peu importe leur taille. Il en résulte un accroissement de la congestion de ce dernier, préjudiciable aux flux courts transmis uniquement sur ce réseau. D’autre part, le gain pour les flux longs est faible : le débit du lien satellite étant important, il n’y a qu’un faible intérêt à y additionner le débit du lien terrestre, bien plus faible, d’autant plus que ce réseau est vite congestionné !

On observe tout de même un petit intérêt de l’agrégation lorsque seulement un ou deux utilisateurs sont présents dans le système. Cet intérêt reste faible en raison du débit réduit du réseau terrestre comparativement au réseau satellite comme énoncé précédemment.

La stratégie de routage par flux MinFCT présente de meilleures performances en raison de sa capacité à tirer profit du réseau terrestre pour accélérer les flux courts. En effet, les flux longs, voire certains flux courts en cas de charge importante, sont routés sur le réseau satellite, et uniquement sur ce dernier. On évite ainsi que des flux pénalisés par le délai du réseau satellite encombrant inutilement le réseau terrestre. Celui-ci reste alors dispo-

nible pour accélérer un plus grand nombre de flux courts, diminuant ainsi le temps moyen de téléchargement des contenus. Ceci est confirmé par la figure 6.2 représentant le taux d'occupation des liens. On observe une importante réduction de l'encombrement du lien terrestre avec MinFCT, expliquant le gain de performance.

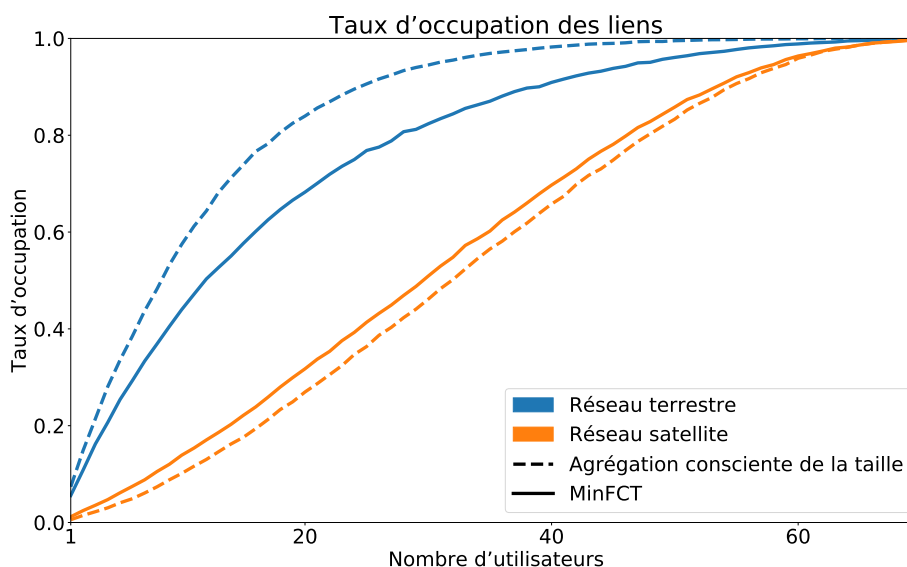


FIGURE 6.2 – Taux d'occupation des liens terrestre et satellite avec les stratégies de routage MinFCT et d'agrégation consciente de la taille des flux.

## 6.2 Agrégation opportuniste des liens inactifs

L'agrégation ayant montré son intérêt à faible charge mais s'étant montrée contre-productive lorsque plusieurs flux sont en concurrence, nous proposons ici d'étudier un mécanisme d'agrégation opportuniste. Celui-ci repose sur la méthode par flux MinFCT qui se verra complétée par un mécanisme d'agrégation déclenché lorsque jugé opportun. L'algorithme MinFCT est donc joué classiquement à chaque nouveau flux afin de lui attribuer un réseau d'accès. Le flux est alors transmis uniquement par ce chemin, sauf si les deux conditions suivantes sont remplies :

- aucun flux n'est en cours sur le second réseau d'accès ;
- la condition d'agrégation 6.1 portant sur la taille est respectée.

Dans ce cas, le flux peut être agrégé, c'est-à-dire que le second réseau d'accès peut également transmettre une partie des paquets du flux.

On s'assure ainsi de ne dégrader ni les performances des autres flux, puisque l'on agrège uniquement lorsque l'autre lien est disponible, ni les performances du flux agrégé grâce à la condition sur la taille.

### 6.2.1 Résultats

Les performances de la méthode sont comparées avec celles de MinFCT sur la figure 6.3. On observe ainsi une très légère amélioration des performances en faisant usage de ce mécanisme. Ce succès reste mitigé en raison d'une part de l'inintérêt à agréger les flux courts sur le réseau satellite à faible charge, ainsi que la faible augmentation du débit liée à l'agrégation pour les flux longs comme expliqué précédemment. Finalement, la méthode montre un intérêt lorsque la charge est relativement importante, c'est-à-dire lorsque le réseau terrestre est encombré et qu'il est alors intéressant d'agrégé certains flux initialement affectés au réseau terrestre au réseau satellite.

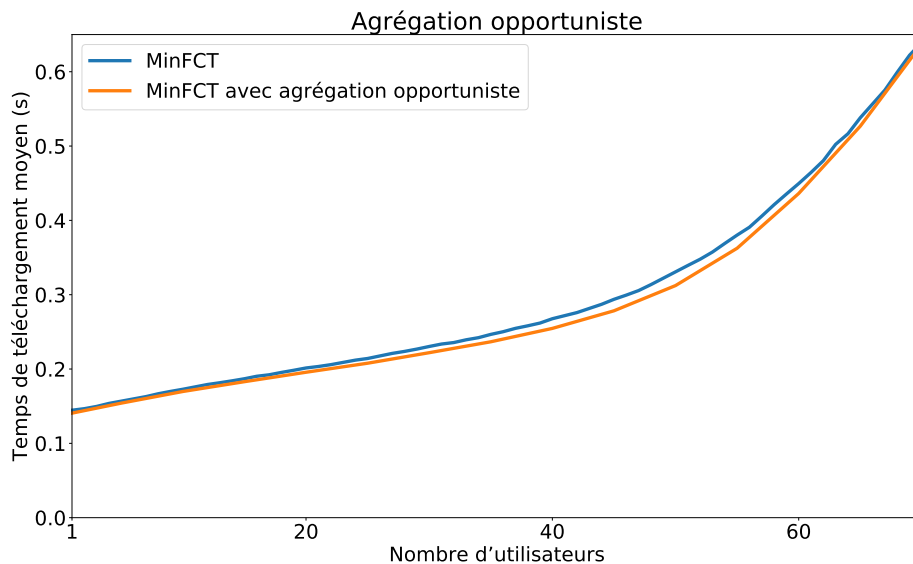


FIGURE 6.3 – Comparaison de MinFCT avec et sans le mécanisme d'agrégation opportuniste des liens inactifs.

## 6.3 Réallocation

Ce dernier mécanisme vise à améliorer les performances de MinFCT en modifiant les décisions de routage au cours du temps afin de les adapter aux évolutions du trafic. En effet, l'estimation du temps de téléchargement faite par MinFCT repose sur le nombre de flux actifs au moment de la prise de décision. Si de nombreux flux arrivent, il peut y avoir un intérêt à déplacer

certain flux initialement affectés au réseau terrestre, les plus longs, sur le réseau satellite afin de décongestionner le premier. À l'inverse, une période calme peut permettre d'accélérer certains flux initialement affectés au réseau satellite grâce au réseau terrestre.

Ainsi, l'algorithme MinFCT est joué classiquement à l'arrivée des flux. À des instants que nous allons présenter, l'algorithme MinFCT sera rejoué pour l'ensemble des flux. L'évolution de la décision de routage, en particulier du réseau satellite au réseau terrestre, est susceptible d'introduire un déséquencement. Le reséquencement des données au niveau de l'IUG est donc, tout comme pour l'agrégation, nécessaire.

**Choix des instants de rejeu** L'intérêt du rejeu étant l'adaptation des affectations aux évolutions du trafic, il semble pertinent de rejouer MinFCT aux instants d'arrivées et de départ des flux, moments où la situation est susceptible d'évoluer. Cependant, à forte charge, l'arrivée de chaque nouveau flux amène à constamment réallouer les flux, posant des problèmes de performance.

Une autre approche consiste à réallouer les flux de manière périodique. La période devra être choisie de l'ordre de grandeur du temps de transmission des flux courts, afin de s'adapter à leur variation. Cette seconde méthode est *a priori* moins performante en raison du délai qui sépare les rejeux des évolutions du réseau. Toutefois, la réallocation périodique est intéressante pour s'adapter aux évolutions liées à des événements extérieurs (*e.g.* variation du débit ou du délai).

**Ordre de traitement des flux** Usuellement, les flux sont affectés par MinFCT dans leur ordre d'arrivée. Lors d'une réallocation, les flux seront traités dans l'ordre croissant de la quantité de données résiduelle. Ainsi, les flux affectés au réseau terrestre seront les plus courts. Procéder dans un autre ordre risquerait de permettre à des flux de longueur moyenne d'être affectés au réseau terrestre, accroissant ainsi sa congestion.

**Adaptation de l'estimateur de MinFCT** Utilisé tel quel, l'estimateur du temps de téléchargement présenté à la section 5.2.3 n'est pas adapté. En effet, il pourrait estimer plus rapide de transmettre les quelques octets restants d'un flux long sur le réseau terrestre, alors que des données sont encore en cours de propagation sur le lien satellite. Il en résulte une augmentation de l'utilisation du lien terrestre au détriment des flux courts, et ce, sans bénéfice pour notre flux long. Ainsi, notre estimation sera minorée par le temps de propagation résiduel des données « en vol ». Un simple suivi de la date d'émission du dernier paquet envoyé sur chaque lien, associé à la connaissance de leur délai, suffit à cela.

### 6.3.1 Résultats

La figure 6.4 présente les résultats associés à cette méthode et les compare à ceux de MinFCT.

Nos différents tests ont confirmé l'inefficacité des rejeux aux instants d'arrivée et de départ à forte charge. Nous avons donc opté pour des rejeux périodiques, avec une période de 100 millisecondes ; d'autres valeurs ont été testées sans amélioration notable.

Là encore, l'intérêt de la méthode se montre très faible. Les situations avec peu d'utilisateurs semblent peu propices à la réallocation, ce qui s'explique par la faible évolution du système dans ces cas-là. En revanche, lorsque la charge augmente, la réallocation regagne de l'intérêt en raison d'une plus grande variabilité du système.

Cependant, ces améliorations restent limitées, en raison de la forte hétérogénéité des liens. En effet, alors que les flux courts seront *a priori* toujours affectés au réseau terrestre et les flux longs au satellite, seul les flux de taille moyenne peuvent espérer bénéficier de ce mécanisme. Or les flux initialement routés sur le réseau satellite sont, de fait, déjà pénalisés par la latence de ce dernier et n'ont ainsi qu'un faible gain à espérer du mécanisme de réallocation. D'autre part, les flux initialement routés sur le réseau terrestre et déplacés sur le réseau satellite auront consommé une bande passante précieuse pour finalement n'en tirer aucun profit.

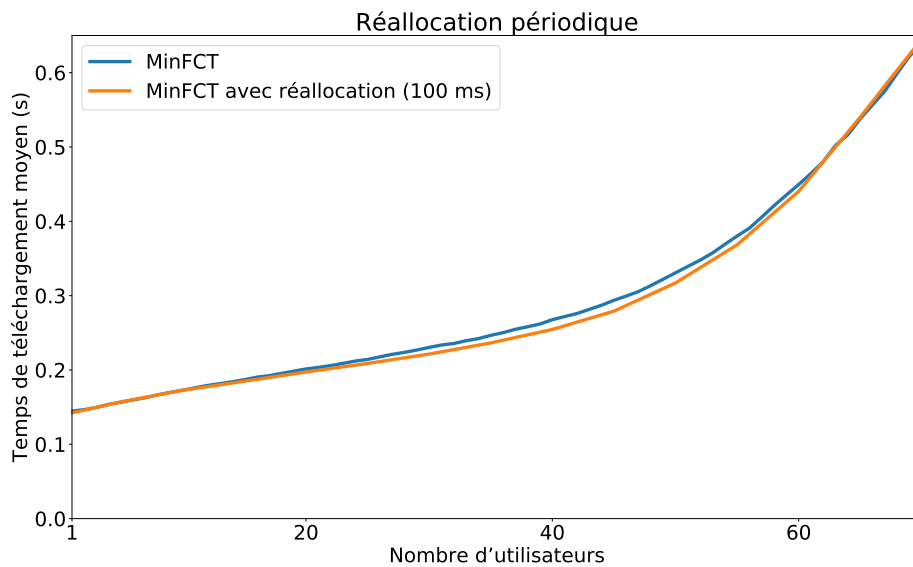


FIGURE 6.4 – Comparaison de MinFCT avec et sans réallocation.



## 6.4 Conclusion

Dans ce chapitre, nous avons tout d'abord étudié l'intérêt de l'agrégation dans notre cas d'hétérogénéité. Nous avons proposé une condition sur la taille des flux, qui s'est montrée absolument nécessaire pour obtenir des performances acceptables. Celles-ci sont tout de même restées en deçà du mécanisme MinFCT, montrant la supériorité d'une approche par flux sur l'agrégation dans un contexte de routage conscient de la taille des contenus.

D'autre part, nous avons proposé deux mécanismes visant à améliorer les performances de MinFCT en sortant du cadre *stricto sensu* du routage par flux. Nous avons montré alors le faible intérêt de ces mécanismes, en raison principalement du contexte fortement hétérogène.

Ainsi, au vu de ces résultats, il apparaît peu intéressant de complexifier un système de manière à permettre un routage à une échelle plus fine que le flux, ceci nécessitant notamment la mise en place d'un mécanisme de reséquencement et une complexification des algorithmes de routage.

Un cadre plus propice de l'utilisation de tels mécanismes pourrait être celui de réseau avec des chemins moins hétérogènes et / ou présentant des variabilités de délai et de trafic plus importantes, en particulier sur des échelles de temps courts. La question de l'intérêt de tels mécanismes reste donc ouverte.

## Chapitre 7

# Mécanisme de routage avec protocole de transport fiable

Dans les chapitres 5 et 6, nous avons démontré l'intérêt de la connaissance de la taille des contenus dans le processus de décision de routage, ainsi que la supériorité des mécanismes par flux sur les mécanismes par paquet. Ces études ont été conduites par modélisation, simulation et expérimentation faisant intervenir des communications sans connexion et sans fiabilisation sur la partie hybride du réseau.

Dans la pratique, la plupart des communications nécessitent une fiabilisation apportée généralement par TCP. Ainsi, nous allons vérifier dans ce chapitre que les conclusions tirées des résultats précédents sont toujours valides lorsque le protocole TCP est utilisé entre nos deux serveurs mandataires. Le mécanisme MinFCT sera préalablement adapté afin de tenir compte de ce changement de protocole. Ces nouveaux résultats seront comparés avec MP-TCP, protocole de transport multichemin de référence et qui utilise une approche par paquet.

### 7.1 Influence du protocole de transport

Les protocoles de transport UDP et TCP se sont largement imposés sur Internet. Alors que le premier fournit un service de transport de datagrammes, sans connexion et sans fiabilisation, le deuxième fournit un service de transmission d'un flux de données, avec connexion, fiabilisation, contrôle de congestion et contrôle de flux. Les expérimentations conduites jusqu'ici transmettaient les flux applicatifs sur la partie hybride du réseau, c'est-à-dire entre les deux serveurs mandataires, en utilisant le protocole UDP. Ce choix s'est imposé pour tenter de reproduire un comportement le plus fidèle possible à la modélisation proposée. Ces expérimentations ont pu être menées à bien par l'utilisation de liens sans perte, avec une limitation logicielle du débit d'émission par les serveurs mandataires.

Dans la pratique, les réseaux présentent des taux de perte non nuls. Bien que les progrès au niveau de la couche physique aient permis de rendre peu probables les pertes liées aux erreurs de transmission dans les systèmes de communication par satellite géostationnaire, les pertes dues à la présence de congestion font partie du fonctionnement normal des communications. Ainsi, la très large majorité du trafic Internet de nos jours repose sur l'utilisation du protocole de transport fiable TCP, notamment exigé par le protocole applicatif HTTP.

Dans ce chapitre, nous gagnerons en réalisme en limitant la capacité des chemins terrestre et satellite par l'introduction d'un goulot d'étranglement sur leur parcours. Des pertes pourront alors survenir lors d'épisodes de congestion provoqués par leur sur-utilisation. Afin d'assurer la continuité des fonctionnalités de TCP sur le réseau hybride, notamment sa fiabilisation des échanges, le même protocole sera utilisé également sur la partie hybride du réseau en lieu et place du protocole UDP utilisé dans les chapitres précédents.

Le mécanisme de routage par flux MinFCT sera adapté afin de tenir compte du comportement de TCP dans son estimation du temps de téléchargement. Les capacités des liens terrestre et satellite, supposées connues précédemment et nécessaires à MinFCT pour ses estimations, seront déduites empiriquement.

## 7.2 Estimation du temps de téléchargement

La méthode MinFCT repose sur l'estimation des temps de téléchargement des contenus sur chaque lien afin de choisir le plus adapté. Le temps de téléchargement dépendant directement du protocole de transport utilisé et du comportement de ce dernier, nous allons adapter notre méthode d'estimation pour TCP.

Le protocole TCP utilise un mécanisme à fenêtre pour déterminer le rythme auquel les données doivent être envoyées. On distingue une fenêtre de réception, permettant le contrôle de flux, et une fenêtre de congestion, permettant l'évitement de congestion (*congestion avoidance*). La fenêtre d'émission est alors la plus petite des fenêtres précédentes. Dans notre cas, le réseau hybride est supposé être le facteur limitant dont on cherche à améliorer les performances ; ainsi, nous supposerons la fenêtre de réception toujours supérieure à la fenêtre de congestion, celle d'émission étant alors égale à celle de congestion.

La fenêtre de congestion a un comportement non trivial : elle est d'abord régie par le mécanisme de démarrage lent (*slow-start*) puis par l'algorithme d'évitement de congestion. Celle-ci est pourtant déterminante dans l'estimation du temps de téléchargement d'un contenu au-dessus de TCP.

Afin d'estimer le temps de téléchargement des contenus, nous détermi-

nous la durée des phases successives de la connexion TCP ainsi que d'autres facteurs influençant le temps total :

- la poignée de main initiale ;
- le temps d'émission du flux ;
- le temps de propagation ;
- le temps supplémentaire dû au démarrage lent ;
- les temps supplémentaires infligés aux flux concurrents ;
- le temps supplémentaire lié aux retransmissions dues aux pertes.

Nous allons maintenant déterminer ces quantités. Pour cela, nous avons besoin des caractéristiques des chemins considérés ; les méthodes d'estimation mises en œuvre seront détaillées dans la section suivante.

### Poignée de main initiale

Le temps lié à la poignée de main sera pris égal à trois fois l'estimation du délai  $d_i$  du chemin considéré.

### Temps d'émission

Le temps d'émission est calculé en divisant la taille  $s$  du flux par la capacité individuelle à laquelle peut prétendre le flux. TCP étant conçu pour assurer une certaine équité entre les différents flux que l'on supposera parfaite au premier ordre, on peut estimer la capacité individuelle en divisant la capacité totale estimée du chemin par le nombre de flux concurrents. Finalement, on obtient un temps d'émission  $t_i^e$  égal à :

$$t_i^e = (n_i + 1) \cdot \frac{s}{c_i} \quad (7.1)$$

avec  $n_i$  le nombre de flux déjà affectés au chemin  $i$ , et  $c_i$  la capacité de ce dernier.

### Temps de propagation

Le temps de propagation est supposé constant durant le transfert du contenu et pris égal à l'estimation du délai du chemin considéré. Les chemins considérés sont soit terrestre et le temps de propagation est faible, soit satellitaire et le temps de propagation est élevé.

### Démarrage lent (*Slow-Start*)

Sur les chemins avec un important produit bande passante  $\times$  délai, typiquement un lien satellite, la connexion TCP peut rester pendant une longue période dans la phase de démarrage lent avant d'atteindre la phase d'évitement de congestion. En effet, lors de la phase de démarrage lent, la fenêtre

de congestion est doublée à chaque RTT (*Round-Trip Time*), jusqu'à la survenue de la première perte provoquant alors le passage de la connexion en phase d'évitement de congestion. Cette première perte survient lorsque la fenêtre de congestion atteint une taille qui permet d'émettre à un débit plus élevé que le débit disponible, provoquant alors de la congestion et donc des pertes.

Pour déterminer ce temps supplémentaire, nous calculons la fenêtre de congestion maximale (`cwnd_max`) avant de passer en phase d'évitement de congestion et le nombre de RTTs nécessaire pour l'atteindre. La fenêtre de congestion maximale, exprimée en segments, est proportionnelle au produit bande passante  $\times$  délai :

$$\text{cwnd}_{\max} = \frac{2d_i}{\text{mss}} \cdot \frac{c_i}{n_i + 1} \quad (7.2)$$

avec usuellement une taille maximale de segment (MSS) de 1448 octets. Le nombre de RTTs alors nécessaire pour l'atteindre est :

$$m_{\text{RTT}}(\text{cwnd}_{\max}) = \text{ceil} \left\{ \log_2 \left( \max \left( 1, \frac{\text{cwnd}_{\max}}{\text{initcwnd}} \right) \right) \right\} \quad (7.3)$$

avec une fenêtre initiale (`initcwnd`) de 10 segments comme défini par la RFC 6928 [27]. Cependant, la fenêtre de congestion maximale n'est pas nécessairement atteinte pour les flux courts. Le nombre maximal de RTTs pour un flux de longueur  $s$  est :

$$m_{\text{RTT}}(\text{end}) = \text{floor} \left\{ \log_2 \left( 1 + \frac{s/\text{mss}}{\text{initcwnd}} \right) \right\} \quad (7.4)$$

Finalement, le temps additionnel introduit par le démarrage lent est donné par :

$$t_{\text{ss}} = 2 \cdot d_i \cdot \min \{ m_{\text{RTT}}(\text{cwnd}_{\max}), m_{\text{RTT}}(\text{end}) \} \quad (7.5)$$

### Temps supplémentaires infligés aux flux concurrents

L'affectation d'un nouveau flux à un chemin provoque une diminution du débit individuel pour les flux concurrents, augmentant ainsi le temps de téléchargement de ces derniers. Si la somme des augmentations est supérieure au gain provoqué par l'affectation du flux sur ce chemin plutôt qu'un autre, l'opération devient contre-productive. Ainsi, pour éviter un tel mauvais choix, la pénalisation subie par les autres flux est ajoutée au « coût » du chemin, et est estimée ainsi :

$$t_{\text{comp}} = \sum_{j=1}^{n_i} \frac{\min(r_j, s)}{c_i} \quad (7.6)$$

où  $r_j$  est la quantité de données résiduelles à transmettre pour le flux  $j$ . On pourra se référer à la description de la version originelle de MinFCT dans la section 5.2.3 qui décrit déjà ce problème.

### Temps supplémentaire lié aux retransmissions dues aux pertes

Chaque paquet perdu doit être retransmis, plusieurs fois si nécessaire jusqu'à sa bonne réception. En supposant indépendantes les pertes des différentes tentatives liées au même paquet, le nombre d'envoi, en moyenne, d'un paquet est égale à :

$$\mathcal{T} = 1 \cdot (1 - \tau_i) + 2 \cdot \tau_i \cdot (1 - \tau_i) + 3 \cdot \tau_i^2 \cdot (1 - \tau_i) + \dots = \frac{1}{1 - \tau_i} \quad (7.7)$$

avec  $\tau_i$  le taux de perte sur le chemin  $i$ .

Ainsi, un flux de longueur  $s$  nécessitera en réalité en moyenne la transmission de  $s \cdot \mathcal{T}$  octets. Les quantités  $s$  ou  $r_j$  présentes dans les formules (7.1), (7.4) et (7.6) doivent être multipliées par  $\mathcal{T}$ .

D'autre part, les retransmissions introduisent en moyenne  $\mathcal{T}$  RTTs supplémentaires, ajoutant au temps de téléchargement  $t_\ell = 2 \cdot d_i \cdot \mathcal{T}$ .

### Temps total de téléchargement

Finalement, le chemin  $p$  sélectionné par MinFCT compte tenu de cette nouvelle méthode d'estimation est celui qui minimise la somme de ces différents termes :

$$p = \underset{i}{\operatorname{argmin}} \left\{ 4 \cdot d_i + t_e + t_{ss} + t_{\text{comp}} + t_\ell \right\} \quad (7.8)$$

## 7.3 Estimation des paramètres des liens

Notre méthode d'estimation du temps de téléchargement des contenus sur chaque lien nécessite la connaissance de la latence, du débit, du taux de perte et du nombre de flux concurrents pour chacun d'entre eux. Nous expliquons ici comme ceux-ci sont estimés.

### 7.3.1 Délai

Le délai est simple à estimer à partir du SRTT (*Smoothed RTT*) des flux ayant précédemment empruntés le chemin. La valeur peut être affinée par moyenne mobile.

### 7.3.2 Capacité

La capacité du chemin satellite est supposée connue. En effet, les réseaux par satellite sont des systèmes très intégrés, avec un grand nombre d'interactions entre les différents composants. Par exemple, les PEP peuvent être couplés avec les allocateurs de ressources dans l'objectif d'obtenir la capacité attribuée à un utilisateur donné. Nous supposons une telle coopération pour notre serveur mandataire.

Une telle coopération est, en revanche, exclue pour obtenir la capacité du réseau terrestre, et il est donc nécessaire d'en effectuer la mesure. Une estimation du débit disponible à partir des informations contenues dans la pile TCP du noyau Linux a été étudiée, mais ces informations se sont avérées non fiables. En effet, les flux sont bien souvent trop courts pour permettre à la connexion TCP de passer en phase d'évitement de congestion ou pour permettre à celui-ci de converger.

Finalement, nous estimons le débit à partir d'une observation passive des accusés de réception TCP (Ack) de l'ensemble des flux transitant par le chemin. Pour cela, chaque flux est suivi, le dernier numéro de séquence acquitté étant enregistré et mis-à-jour lorsqu'un Ack est observé. L'écart entre l'ancienne valeur et la nouvelle indique le nombre d'octets fructueusement reçus par l'IUG. Finalement, le débit de données acquittées est estimé par une moyenne glissante sur une fenêtre temporelle.

L'estimation du débit est relativement précise, mais retardée par le délai du chemin considéré et les périodes de reprise après pertes. En effet, lorsqu'une perte survient, les accusés de réception qui suivent indiquent invariablement le même numéro de séquence du segment manquant jusqu'à sa réception après retransmission. L'option TCP SACK (*Selective Acknowledgment*), introduite dans la RFC 2018 [95], permet de transmettre des informations plus précises en spécifiant les blocs de données correctement reçus. Son observation permet, au prix d'une complexité d'implantation accrue, de continuer à évaluer la quantité de données reçues pour un flux durant une phase de réparation.

### 7.3.3 Taux de perte

Le taux de perte est déterminé en divisant le nombre de retransmission par le nombre total de segments émis. Il est donc nécessaire d'identifier les retransmissions parmi les segments envoyés. Pour cela, on retient, pour chaque flux, le plus élevé des numéros de séquence observés. Si un segment est émis avec un numéro de séquence inférieur, il s'agit alors d'une retransmission ; dans le cas contraire, il s'agit d'une émission régulière.

En pratique, TCP n'est pas toujours en mesure de déterminer précisément les segments qui ont été reçus correctement, malgré l'option SACK, et retransmet donc certains segments pourtant correctement reçus. Ainsi, cette

méthode surestime le taux de perte. Cependant, cette manière d'évaluer le taux de perte correspond bien à l'usage que l'on en fait puisque l'on cherche à déterminer le temps d'émission total compte tenu des retransmissions de TCP.

D'autre part, le démarrage lent des connexions TCP provoque nécessairement quelques pertes et retransmissions lorsque la fenêtre de congestion maximale est atteinte, et ce, même en l'absence de congestion notable. Alors qu'à forte charge ce phénomène est anecdotique face aux retransmissions liées à la congestion, à faible charge il prend une proportion très importante devant le faible nombre de segments transmis. On observe alors un taux de perte très élevé sur un lien pourtant faiblement utilisé, amenant MinFCT à transmettre certains flux sur le réseau satellite quand le réseau terrestre se serait montré plus rapide.

Pour gommer ce problème, le nombre de segments retransmis est minoré d'un segment par seconde, correspondant à une diminution de 0.6% du taux de perte en régime saturé pour le lien terrestre. Cette quantité a été déterminée expérimentalement, celle-ci étant suffisamment grande pour réduire ce phénomène, tout en permettant la détection de la congestion. La validité de cette valeur n'a cependant pas été vérifiée avec des paramètres de liens différents; cette méthode d'estimation du taux de perte est sujette à améliorations.

Le taux de perte est lui aussi estimé par une moyenne glissante sur une fenêtre temporelle.

#### 7.3.4 Nombre de flux concurrents

Afin de déterminer le débit individuel dont dispose un flux, la capacité totale du chemin est divisée par le nombre de flux concurrents partageant ce dernier. Le serveur mandataire connaît le nombre de flux affectés au chemin considéré, mais tous ne sont pas nécessairement actifs. En effet, une partie d'entre eux attend la fin de leur poignée de main initiale tandis que d'autres, bien qu'ayant terminé l'émission du contenu, attendent toujours l'accusé de réception des données en vol. Ainsi, seuls les flux ayant terminé leur poignée de main et ayant toujours des données en attente, dans leur tampon applicatif ou dans la pile TCP, sont pris en compte. Sur les liens congestionnés, définis par un taux de perte supérieur à 1%, les flux avec des données en attente d'accusés de réception sont également pris en compte en raison d'une probable retransmission.

## 7.4 Expérimentations

Dans cette section, nous comparons les performances de différentes méthodes de routage utilisant un protocole de transport fiable sur la partie hybride du réseau. Ces évaluations ont plusieurs buts. Tout d'abord, il s'agira



de vérifier le bon fonctionnement de la version adaptée à TCP du mécanisme de routage MinFCT. Ensuite, on vérifiera que la conclusion tirée du chapitre 5, c'est-à-dire l'avantage de la connaissance de la taille lors de la décision de routage, est toujours valide en présence de TCP. Finalement, les performances de MinFCT seront comparées à celles de MP-TCP, protocole de transport multichemin de référence et ayant adopté une approche par paquet.

Alors que les résultats des chapitres précédents faisaient également intervenir un modèle analytique et des simulations, les résultats qui seront présentés dans cette section ont tous été obtenus uniquement par expérimentation. En effet, notre modèle analytique ne prend pas en compte le comportement compliqué du protocole TCP. Il en est de même pour notre simulateur. D'autre part, l'évaluation par expérimentation nous permet de comparer nos résultats à l'implantation de référence de MP-TCP dans le noyau Linux.

#### 7.4.1 Architecture

L'architecture de notre plateforme d'expérimentation est très similaire à celle utilisée dans les chapitres précédents et est représentée par la figure 7.1.

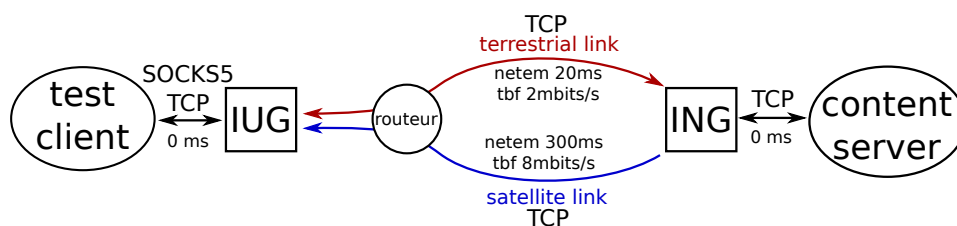


FIGURE 7.1 – Plateforme d'expérimentation avec limitation des débits.

Les débits des liens terrestre et satellite sont à présent limités. Cette limitation est obtenue par l'usage de la `qdisc tbf` (*Token Bucket Filter*).

Un routeur additionnel est placé entre l'IUG et le réseau hybride, de manière à cacher à l'IUG la présence du réseau hybride. Cette modification n'est utile que pour les expérimentations avec MP-TCP et son utilité sera détaillée dans la sous-section suivante.

#### Paramètres

Les paramètres des lois régissant le temps d'attente des utilisateurs ou la taille des flux sont les mêmes que dans les chapitres précédents.

La table 7.1 reproduit le choix des paramètres de la `qdisc tbf`, nouveau composant de notre plateforme d'expérimentation.

	rate	burst	latency	peakrate	mtu
Lien terrestre	2 Mbit/s	2500 octets	40 ms	3 Mbit/s	1513 octets
Lien satellite	8 Mbit/s	10 ko	160 ms	12 Mbit/s	1513 octets

TABLE 7.1 – Paramètres utilisés pour la `qdisc tbf`

Le paramètre `rate` limite la capacité du lien, tandis que le paramètre `latency` limite la taille du tampon dans lequel sont placés les paquets en attente de transmission. La taille du tampon peut être obtenue en multipliant les deux paramètres précédents, soit respectivement  $2 \text{ Mbit/s} \times 40 \text{ ms} = 10 \text{ ko}$  et  $8 \text{ Mbit/s} \times 160 \text{ ms} = 160 \text{ ko}$ . Lorsque le tampon est plein, les paquets sont perdus ; il est donc nécessaire de choisir une taille suffisamment importante de manière à absorber les *bursts* de paquets. En revanche, une taille trop importante augmente la latence des transmissions, problème connu sous le terme *bufferbloat*. La valeur de compromis retenue est égale à  $\frac{\text{rate}}{50000}$ .

Le paramètre `burst` indique la quantité maximale de jetons que `tbf` peut accumuler ; sa valeur est prise égale à  $\frac{\text{rate}}{8 \cdot \text{HZ}}$  conformément aux conseils de paramétrage des créateurs de la `qdisc`. Le paramètre `HZ` est issu du noyau Linux et l'implantation de `tbf` en dépend ; il vaut 100 MHz dans notre cas. L'utilisation du paramètre `peakrate`, égal à  $\frac{3}{2} \cdot \text{rate}$ , permet d'atténuer l'effet des *bursts* en les lissant, rapprochant ainsi notre banc de test d'un réseau contenant un grand nombre de routeurs intermédiaires.

La latence des liens est toujours émulée par l'usage de la `qdisc netem`.

Les expérimentations menées avec TCP le seront avec l'algorithme d'évitement de congestion par défaut sous Linux, CUBIC. La version de MP-TCP utilisée est la 0.94 reposant sur le noyau Linux 4.14.24. Elle sera utilisée avec l'algorithme de congestion couplé OLIA (wVegas et BALIA donnent des résultats similaires). Nous utiliserons le gestionnaire de chemin `fullmesh`, visant à établir autant de connexions qu'il existe de couple (adresse IP source, adresse IP destination) ; l'introduction du routeur intermédiaire entre l'IUG et le réseau hybride assure ainsi que MP-TCP utilise deux sous-flux, un sur le chemin terrestre et un sur le chemin satellite. Finalement, nous emploierons l'ordonnanceur par défaut « *lowest RTT* » dont le comportement vise à privilégier le chemin avec le RTT le plus faible, c'est-à-dire le réseau terrestre.

Le mécanisme de routage probabiliste utilisera des probabilités proportionnelles à la capacité des liens, c'est-à-dire 0.2 et 0.8.

### Estimation des paramètres des liens

Un estimateur des paramètres des liens tel que décrit dans la section 7.3 a été implanté en C. Deux instances en sont lancées dans le même `netns` que l'ING : une observant le réseau terrestre, l'autre observant le réseau satellite.

Les informations recueillies sont partagées avec le serveur mandataire par l'utilisation de segments de mémoire partagée. Les moyennes glissantes sont effectuées sur des fenêtres temporelles de 10 secondes.

### Serveurs mandataires

Les serveurs mandataires sont les mêmes que ceux utilisés précédemment mais ont été adaptés pour prendre en charge le protocole TCP.

L'utilisation de TCP sur la partie hybride du réseau nécessite, contrairement à UDP, l'établissement des connexions. Ainsi, lorsque l'IUG reçoit une demande de connexion du client, deux connexions TCP sont ouvertes en parallèle : une sur le chemin terrestre et une sur le chemin satellite. La requête de l'utilisateur reste transmise strictement par le réseau terrestre, plus rapide. Lors de l'émission de la réponse par l'ING, il se peut que seule la connexion terrestre soit établie, la poignée de main sur le réseau satellite n'étant pas terminée en raison de l'important délai. Dans ce cas, suivant les directives du mécanisme de routage considéré, l'ING pourra démarrer l'émission des données sur le réseau terrestre ou devra attendre l'établissement complet de la connexion sur le réseau satellite.

Dans le cas de MP-TCP, la connexion sera établie sur le réseau terrestre avant l'établissement d'un second sous-flux sur le réseau satellite. Le comportement inverse serait de nature à introduire une importante latence préjudiciable aux performances.

L'utilisation d'un ordonnanceur visant à assurer l'équité en débit entre les différents flux n'est plus nécessaire. En effet, l'équité entre les flux est une propriété de TCP et MP-TCP ; c'est même une exigence avant la mise en service d'un nouvel algorithme d'évitement de congestion. Ainsi, tous les flux sont servis par le serveur mandataire aussi rapidement que le permet leur fenêtre d'émission TCP.

## 7.4.2 Résultats

### MinFCT adapté au protocole TCP

La figure 7.4.2 présente les performances obtenues pour différentes méthodes de routage avec TCP sur la partie hybride du réseau. Nous avons ainsi représenté le temps de téléchargement moyen en fonction du nombre d'utilisateurs pour :

- un routage monochemin sur le lien terrestre (respectivement satellite) uniquement ;
- un routage probabiliste et par seuil ; le jeu de probabilités ainsi que le seuil ont été choisis pour des performances moyennes optimales à forte charge ;

- un routage avec la méthode MinFCT dans sa version adaptée à TCP que nous venons de présenter.

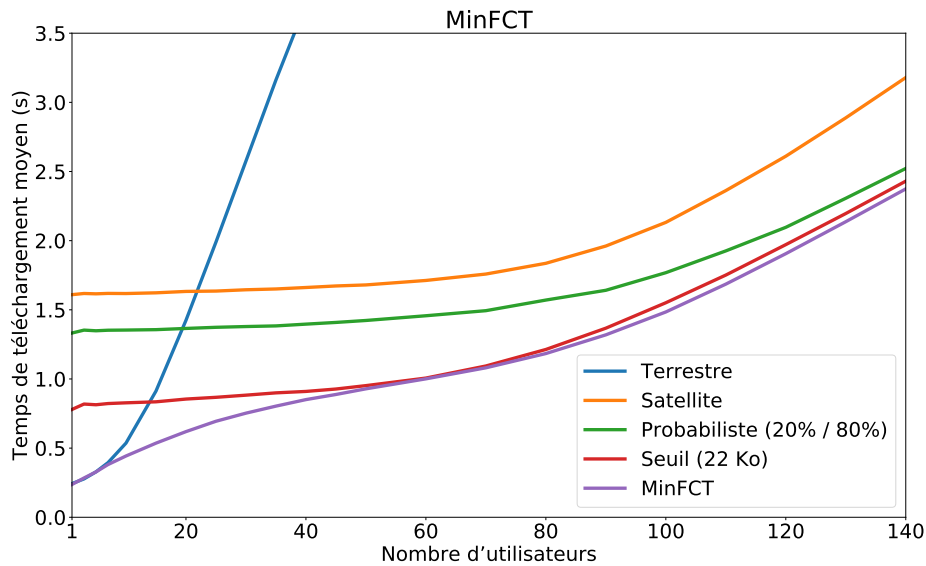


FIGURE 7.2 – Comparaison des performances de MinFCT par rapport à un routage probabiliste et par seuil ainsi qu'à un routage monochemin.

On observe, pour les communications monochemins, des résultats similaires à ceux observés dans le chapitre 5. Pour le réseau terrestre, le temps de téléchargement moyen est court à faible charge et augmente rapidement avec l'accroissement du nombre d'utilisateurs, révélant la faible capacité de ce réseau. Au contraire, le réseau satellite témoigne de son importante latence, mais connaît une congestion bien plus tardive et moins brusque.

Le mécanisme de routage probabiliste montre sa capacité à améliorer d'environ 20% les performances du réseau satellite, en accord avec le jeu de probabilités utilisé. Tout comme lorsque le protocole UDP était utilisé, les performances sont peu intéressantes à faible charge en raison de l'important délai du réseau satellite. En revanche, le mécanisme de routage par seuil témoigne d'un temps de téléchargement bien plus faible. Ce résultat confirme la conclusion du chapitre 5 affirmant la supériorité des mécanismes de routage ayant accès à la taille des contenus.

Le mécanisme MinFCT montre sa supériorité sur l'ensemble des mécanismes précédents. Ses performances à faible charge sont proches de celles du réseau terrestre seul, montrant sa faculté à exploiter le faible temps de propagation de ce dernier. Le temps de téléchargement moyen de la méthode rejoint progressivement celui du routage par seuil lorsque le nombre d'utilisateurs augmente, avec des performances asymptotiques équivalentes. On

observe, tout comme dans le chapitre 5, deux points d'inflexion : un premier aux alentours de 20 utilisateurs lorsque le réseau terrestre se trouve saturé, puis un deuxième vers 80 utilisateurs lorsque le réseau satellite se trouve à son tour saturé. Ainsi, notre estimation du temps de téléchargement semble correctement détecter la congestion des réseaux terrestre et satellite de manière à équilibrer le trafic entre eux. Nous pouvons ainsi valider l'adaptation du mécanisme MinFCT pour son usage avec le protocole de transport TCP.

### Comparaison avec MP-TCP

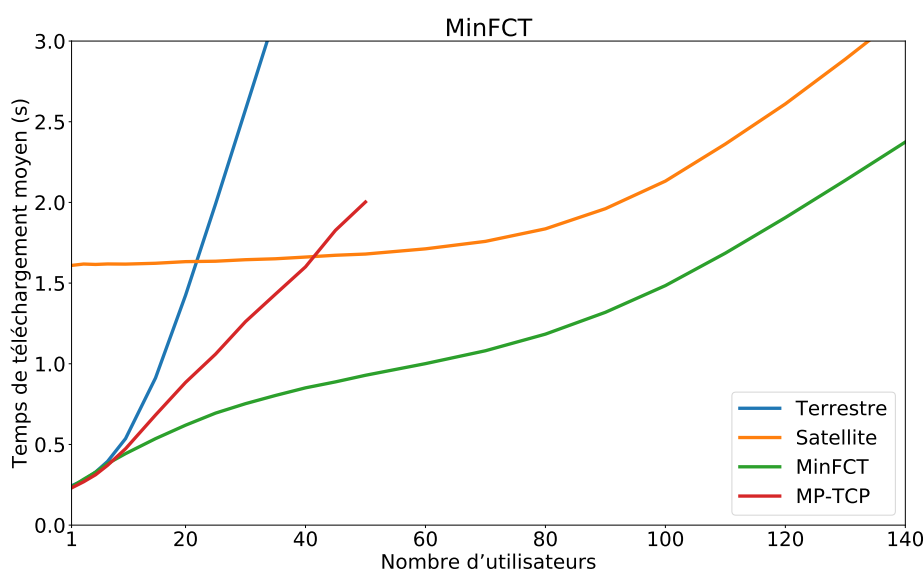


FIGURE 7.3 – Comparaison des performances de MinFCT et MP-TCP.

La figure 7.3 compare les performances de MinFCT avec celles obtenues avec le protocole de transport multichemin de référence MP-TCP.

Le protocole MP-TCP montre sa capacité à améliorer les performances du réseau terrestre en agrégeant les flux longs sur le réseau satellite. Il est à noter que l'ordonnanceur *lowest RTT* de MP-TCP privilégie l'émission des données sur le flux terrestre. Les flux ne sont agrégés avec le réseau satellite qu'une fois le deuxième sous-flux établi, et ne le sont donc pas si la transmission se termine avant. Ce comportement évite ainsi d'émettre sur le réseau satellite des données appartenant à des flux courts, comportement qui aurait augmenté dramatiquement le temps de téléchargement moyen en raison du temps de propagation de ce réseau.

Néanmoins, les performances de MP-TCP se dégradent avec l'augmentation de la charge. En effet, le réseau terrestre devient rapidement très congestionné, l'ensemble des flux l'exploitant. Ce réseau devient alors peu fiable, le

temps nécessaire à l'établissement des connexions augmente, de nombreuses retransmissions sont nécessaires. Au-delà d'une quarantaine d'utilisateurs, le réseau terrestre est tellement dégradé que les performances de MP-TCP deviennent même moins bonnes que TCP sur le réseau satellite seul ! Enfin, au-delà d'une cinquantaine d'utilisateurs, des rejets ont lieu ; les résultats n'ont donc pas été retenus en raison du biais introduit par le rejet plus fréquent des flux les plus longs.

Contrairement à MP-TCP, MinFCT arrive à optimiser le niveau de congestion du réseau terrestre afin d'en tirer profit pour accélérer certains flux bien choisis. Ceci nous conforte dans nos conclusions sur l'inefficacité de l'agrégation dans un contexte hétérogène établies au chapitre 6.

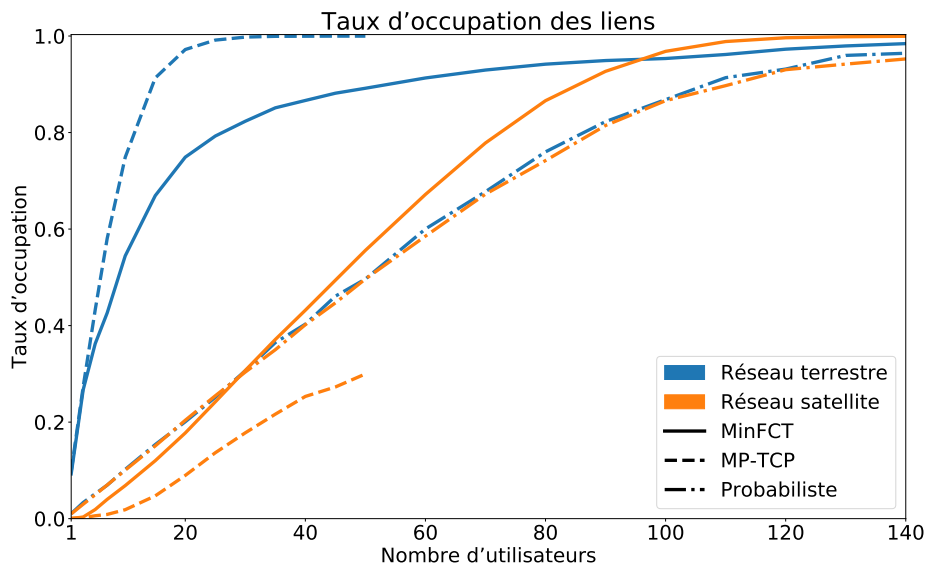


FIGURE 7.4 – Comparaison du taux d'occupation des liens terrestre et satellite entre le routage probabiliste, MinFCT et MP-TCP.

Ces analyses sont confirmées par la figure 7.4 comparant les taux d'occupation des liens terrestre et satellite pour les différentes stratégies de routage. La méthode probabiliste affiche des taux d'occupation identiques sur les deux liens en raison du jeu de probabilités choisi. Le routage par seuil, bien que non représenté, mène à des taux d'occupation identiques, le seuil ayant été choisi pour répartir le trafic équitablement entre les deux chemins.

MinFCT montre bien son utilisation avantageuse du réseau terrestre tout en évitant la congestion de ce dernier. Celui-ci apparaît même légèrement sous-utilisé ; cette observation laisse entrevoir une possible amélioration des performances du mécanisme en le rendant plus agressif vis-à-vis du réseau terrestre. Toutefois, ce réseau servant à la transmission de l'ensemble des requêtes, une marge de sécurité semble judicieuse afin de ne pas voir les

performances s'effondrer.

## 7.5 Conclusion

Dans ce chapitre, nous avons adapté l'algorithme d'estimation du temps de téléchargement des contenus du mécanisme de routage MinFCT pour l'utilisation du protocole de transport TCP. Les résultats obtenus ont montré la supériorité des algorithmes de routage ayant accès à la taille des contenus, confirmant l'intérêt de la connaissance de cette information en accord avec les résultats du chapitre 5. D'autre part, les résultats ont montré la supériorité du mécanisme MinFCT sur les autres méthodes de routage, confirmant le fonctionnement correct de l'algorithme d'estimation du temps de téléchargement. Pour finir, la méthode a été comparée au protocole de routage multichemin MP-TCP, et il a été montré, là-encore, la supériorité de l'algorithme de routage MinFCT. Ceci confirme également les résultats du chapitre 6, qui concluait à la supériorité des mécanismes de routage à l'échelle des flux sur les mécanismes d'agrégation à l'échelle des paquets dans un contexte d'hétérogénéité.

## Chapitre 8

# Conclusion et perspectives

### 8.1 Conclusion

Les réseaux sont mus par une nécessaire évolution pour satisfaire les exigences croissantes des utilisateurs d'Internet. Cependant, on observe un accroissement de la fracture numérique entre les zones urbaines et rurales.

Pour répondre aux besoins des zones mal desservies, nous avons étudié dans cette thèse la diffusion de contenus sur un réseau hybride satellite / terrestre. Cette approche marie deux thématiques que sont les CDNs et les réseaux multichemins hétérogènes.

Nous avons dans un premier temps proposé une architecture permettant de prendre la main sur la diffusion des contenus sur la partie hybride du réseau. Cette architecture est composée de deux serveurs mandataires positionnés aux extrémités du réseau hybride. Ils sont opérés par un opérateur hybride, qui possède également son propre CDN, qualifié alors de TelCo CDN et interconnecté *via* CDNI aux CDNs d'opérateurs.

Ainsi, l'opérateur hybride s'inscrit dans une chaîne de diffusion CDN, et se voit déléguer la livraison des contenus jusqu'à l'utilisateur. Ce contexte CDN permet l'accès à plusieurs informations sur les contenus, notamment leur taille, qui pourront être utilisées dans la gestion de la diffusion des contenus sur le réseau hybride.

Par la suite, nous avons étudié l'impact du routage sur les performances de cette architecture hybride satellite / terrestre, et en particulier l'apport de la connaissance de la taille des contenus par le mécanisme de routage. Nous avons évalué trois méthodes de routage à l'échelle des flux : une méthode probabiliste, une méthode par seuil et enfin notre proposition de mécanisme, MinFCT. Ces évaluations ont donné lieu à une résolution analytique pour les deux premières, ainsi qu'à des simulations et expérimentations pour l'ensemble d'entre elles.

Nous avons alors montré l'intérêt important apporté par la connaissance de la taille, les deux stratégies en faisant usage montrant de bien meilleures



performances que la méthode naïve probabiliste. D'autre part, nous avons pu observer, pour notre mécanisme MinFCT, des performances équivalentes aux résultats optimaux atteignables par la stratégie par seuil, tout en s'affranchissant de la difficulté du paramétrage d'un seuil.

Par la suite, nous avons étudié les possibilités d'amélioration des performances en descendant à une échelle de routage plus fine, c'est-à-dire à l'échelle des paquets. Nous avons alors montré la contre-productivité de l'agrégation, y compris lorsque la taille des flux est connue, en raison de la congestion qui se crée sur le réseau terrestre. D'autre part, nous avons proposé deux modifications de la méthode MinFCT : une agrégation opportuniste ainsi qu'une réallocation. Cependant, la faible amélioration apportée par celles-ci décourage leur utilisation au vu de la complexité supplémentaire nécessaire à leur mise en œuvre.

Finalement, nous avons fait évoluer notre plateforme d'expérimentation afin d'utiliser le protocole de transport TCP sur la partie hybride du réseau. Nous avons préalablement adapté la méthode d'estimation du temps de téléchargement des contenus du mécanisme MinFCT afin de tenir compte de l'influence de ce protocole. Les résultats obtenus sont venus conforter les conclusions tirées précédemment, validant l'intérêt de la connaissance de la taille dans le mécanisme de routage.

D'autre part, les performances du mécanisme MinFCT ont pu être comparées à celles du protocole de routage multichemin MP-TCP. Nous avons alors montré la supériorité de notre proposition sur MP-TCP, et par là-même confirmé le mauvais fonctionnement de l'agrégation en milieu hétérogène.

## 8.2 Perspectives

Le routage hybride mis en œuvre dans cette thèse nécessite la synergie d'un grand nombre de composants, tant au niveau de l'architecture que des protocoles ou des mécanismes de routage. Ainsi, les perspectives d'améliorations sont nombreuses.

### 8.2.1 Architecture

Dans l'architecture décrite au chapitre 4, nous proposons l'utilisation des protocoles d'interconnexion CDNI afin de rediriger les requêtes des utilisateurs vers un serveur de contenus maîtrisé par le fournisseur d'accès hybride. Par la suite, nous avons supposé que l'utilisateur était redirigé vers l'IUG. Celui-ci acquiert alors les contenus depuis l'ING, et nous avons étudié l'utilisation de la taille dans la décision de routage des contenus sur le réseau hybride.

Cependant, nous n'avons pas étudié les possibilités de routage au niveau CDN et CDNI.

### Routage au niveau CDN

Le fournisseur d'accès Internet hybride opère son propre TelCo CDN, et peut donc à ce titre rediriger l'utilisateur vers le cache jugé le plus adapté. Ainsi, il lui est possible de rediriger l'utilisateur non pas vers l'IUG mais vers l'ING.

Le but est ici d'estimer l'intérêt de la mise en cache sur l'IUG, afin d'économiser éventuellement une redirection et le temps qui lui est associé. Il peut s'agir également d'économiser l'espace de stockage disponible sur l'IUG, que l'on peut supposer restreint. Ainsi, cette économie est intéressante pour les contenus peu populaires et les contenus de taille trop importante.

### Routage au niveau CDNI

Nous avons limité notre architecture à l'interconnexion d'un opérateur CDN avec le TelCo CDN du fournisseur d'accès hybride. Cependant, nous pouvons étendre cette architecture et supposer que les opérateurs terrestre et satellite disposent de leur propre TelCo CDN. En supposant ces nouveaux TelCo CDN également interconnectés par CDNI, il est alors possible de rediriger les utilisateurs vers ces derniers.

L'intérêt d'une telle redirection réside dans la position avantageuse des serveurs de contenus des TelCo CDNs terrestre et satellite. En effet, alors que le réseau hybride est un *overlay* dont l'ING est situé à l'extrémité, les caches des fournisseurs d'accès terrestre et satellite peuvent être placés plus près des utilisateurs. D'autre part, ils sont également en meilleure position pour optimiser la livraison des contenus sur leur réseau respectif.

Une autre solution serait, une fois l'utilisateur redirigé vers l'IUG, d'acquérir le contenu demandé en s'adressant à l'un des TelCo CDN terrestre ou satellite.

Néanmoins, ces approches rendent *a priori* plus compliquée l'estimation de l'état des réseaux, utile à une telle décision de routage.

## 8.2.2 Protocole de transport

### Interaction avec les PEPs

Dans le chapitre 7, nous avons adapté l'estimateur du temps de téléchargement de la méthode de routage MinFCT pour l'utilisation du protocole TCP. Cependant, les réseaux satellitaires utilisent généralement des *Performance-Enhancing Proxies* afin d'améliorer les performances de TCP. Leur utilisation nécessite donc l'adaptation de l'estimateur du temps de téléchargement à leur comportement. Notons que l'ING utiliserait alors un estimateur différent pour chaque réseau. Nous pourrions prolonger nos travaux dans ce sens en regardant en particulier l'impact de ces estimateurs et de leurs imperfections.

### Réduction du temps de connexion

Nos implantations des serveurs mandataires IUG et ING procèdent, lors d'une requête émanant d'un client, à l'ouverture d'une connexion TCP sur chaque chemin. L'établissement de ces connexions est coûteux pour les performances. Diverses pistes peuvent être explorées pour réduire cet effet :

**TCP Fast-Open** Cette option TCP permet à l'ING de débiter l'émission du contenu dès la réception du SYN initial sans attendre la fin de la poignée de main, économisant ainsi un RTT.

Toutefois, pour éviter certaines attaques, l'option ne peut être utilisée qu'à partir de la deuxième tentative de connexion entre deux hôtes. D'autre part, TCP Fast-Open possède une API spécifique nécessitant une modification des applications.

Notre scénario semble néanmoins très propice à son utilisation. En effet, l'ING et l'IUG établissent un grand nombre de connexions ; TCP Fast-Open peut donc être constamment utilisée. De plus, le contrôle des serveurs mandataires permet facilement son implantation.

Son bénéfice étant la réduction du temps de téléchargement d'un RTT, celle-ci semble d'autant plus efficace sur le réseau satellite. Il faudra néanmoins vérifier son comportement avec les PEP, qui, par ailleurs, peuvent proposer une fonctionnalité similaire.

**Pool de connexions et réutilisation** Plutôt que d'ouvrir les connexions à la demande, il est possible d'en ouvrir plusieurs pro-activement formant ainsi un *pool* de connexion pré-établies. En outre, après le transfert d'un contenu, les connexions peuvent être réutilisées. Il est également possible de limiter le nombre maximal de connexions de manière à éviter une congestion excessive sur un réseau donné, préjudiciable aux performances globales.

### Utilisation d'un autre protocole tel que QUIC

Nous avons mené nos expérimentations avec le protocole de transport fiable TCP. Il serait intéressant d'en comparer les résultats avec l'utilisation du protocole QUIC qui connaît une certaine croissance de notoriété. Tout comme pour les TCP-PEPs, la méthode d'estimation du temps de téléchargement doit préalablement être adaptée à ce protocole.

### 8.2.3 Mécanisme de routage

#### Trafic web spécifique

Nous avons supposé, à des fins de simplification, que chaque flux véhiculait un unique contenu. La réalité diffère de cette situation idéalisée et nous pourrions adapter notre mécanisme de routage pour en tenir compte.

**Réutilisation des connexions HTTP** Le protocole HTTP/1.1 permet de réutiliser une connexion TCP pour transmettre plusieurs requêtes. Celles-ci peuvent être transmises à la suite de la réception de la réponse à la requête précédente, ou alors les unes après les autres, sans délai, de manière à éliminer la durée d'un RTT présente entre deux requêtes (*pipelining*).

Ce mécanisme nécessite d'adapter notre algorithme de routage à l'échelle des flux pour considérer de manière distincte les différents contenus et leur propre taille. Lorsque les requêtes sont successives, l'adaptation semble relativement aisée. Cependant, le *pipelining* soulève des questions plus complexes. En effet, les contenus doivent être délivrés dans le même ordre que la transmission des requêtes par l'application, ce qui pose la question de l'intérêt d'accélérer les contenus courts sur le réseau terrestre si ces derniers se trouvent bloqués en tête de file dans l'attente de contenus plus longs transmis sur le réseau satellite.

**HTTP/2** Le protocole HTTP/2 est une évolution majeure du protocole HTTP/1.1. Il apporte à ce titre de nouvelles fonctionnalités telles que le multiplexage de plusieurs requêtes et réponses dans une même connexion ainsi que la possibilité pour un serveur d'envoyer proactivement des contenus au client. L'utilisation d'HTTP/2 est intéressante, le multiplexage permettant de résoudre le problème de blocage en tête de file présent avec HTTP/1.1. En effet, il est possible de délivrer les contenus à l'application sans respecter l'ordre de transmission des requêtes.

**Tronçonnage des vidéos** Les vidéos sont bien souvent subdivisées en tronçon de quelques Ko ou Mo. Cette technique permet l'adaptation dynamique de la qualité de la vidéo téléchargée par le client, en réponse aux variations de la congestion du réseau.

Il est important pour notre mécanisme de routage d'identifier les tronçons comme appartenant à un contenu vidéo plus important, au risque de les transmettre sur le réseau terrestre et ainsi de congestionner ce dernier. D'autre part, il serait intéressant de vérifier le bon fonctionnement des algorithmes d'adaptation de la qualité des vidéos sur un réseau hybride tel que le nôtre.

## Qualité de Service (QoS)

Tout au long de cette thèse, nous avons évalué la performance des mécanismes de routage en observant le temps de téléchargement moyen. Les contenus ne sont différenciés que par leur taille et connaissent un traitement identique.

Il serait intéressant de considérer des critères de qualité de service différents suivant les types de contenus (par exemple, images, javascript, page html...). En effet, le contexte CDN nous permet d'accéder à des informations très précises sur les contenus, permettant facilement d'en déduire des critères de QoS. L'intégration de critères de QoS dans le mécanisme de routage nécessite également l'utilisation de nouvelles métriques pour en vérifier l'intérêt.

### Trafic non CDN

Dans cette thèse, nous avons uniquement considéré le routage de contenus hébergés sur CDN. Cependant, un système réel doit être en mesure de traiter l'ensemble du trafic produit par les utilisateurs, y compris le trafic non CDN.

Cette thématique a déjà été traitée dans le projet BATS [33] pour le trafic non HTTP par l'utilisation de règles de routage statiques reposant sur l'identification des flux à partir d'une liste de ports bien connus. Cependant, nous devons pour notre part être capables de traiter également le trafic HTTP non redirigé par CDNI. Cet objectif est double : il s'agit d'une part de prendre en charge le trafic HTTP non CDN, et d'autre part de prendre en charge le trafic CDN pour lequel nous n'avons pas d'accord d'interconnexion pour la diffusion. La gestion du trafic CDN sans interconnexion est également un moyen pour déployer progressivement l'architecture étudiée dans cette thèse.

Le trafic HTTP non chiffré peut être intercepté et traité par un serveur mandataire transparent. En revanche, le trafic chiffré nécessite la mise en place de mécanismes spécifiques. Une technique détaillée dans un document ETSI [43] consiste à observer le temps d'inter-arrivée entre les segments TCP de manière à distinguer les différents contenus véhiculés par un même flux et à estimer par là-même leur poids.

Le mécanisme de réallocation, présenté dans le chapitre 6, peut également être une solution à cette situation. La taille d'un contenu est estimée à partir de la quantité de données reçues par l'ING depuis le serveur en amont. Cette taille est donc initialement sous-estimée en début de réception, amenant à une décision de routage potentiellement mauvaise. La réallocation permet alors de faire évoluer ce choix lorsque la taille est mieux connue. Nous avons procédé à quelques expérimentations qui ont montré un grand intérêt pour cette méthode, qui mériterait donc d'être approfondie.

### Utilisation de plus de deux réseaux d'accès

Un des intérêts de la méthode MinFCT est de pouvoir, sans modification, gérer autant de liens que voulu. On peut ainsi considérer des scénarios avec trois liens, par exemple un lien satellite, un lien ADSL et un lien LTE. Il

serait alors intéressant d'observer le fonctionnement de ce mécanisme dans un tel scénario.

### ***Multicast* par satellite**

Comme évoqué au chapitre 3, le satellite possède une capacité de *multicast*. On peut ainsi imaginer faire évoluer notre algorithme de routage de manière à tenir compte de l'intérêt d'une diffusion en *multicast* compte tenu, par exemple, de la popularité d'un contenu.

#### **8.2.4 Affinement de la modélisation**

Dans nos expérimentations, nous avons abstrait le comportement des couches basses. La plateforme expérimentale OpenSAND [110] qui émule de façon plus réaliste la couche physique et la couche d'accès avec un partage des ressources plus précis pourrait constituer un cadre intéressant en partie vis-à-vis des fluctuations de trafic et des conditions de propagation.

Notons encore que les scénarios applicatifs pourraient être plus variés. Certaines situations peuvent s'avérer particulièrement intéressantes ; par exemple, un trafic de navigation web en parallèle du téléchargement d'une vidéo. Ce dernier scénario est très propice à profiter d'un réseau hybride, la vidéo étant délestée sur le réseau satellite afin de ne pas avoir d'impact sur le trafic web.

#### **8.2.5 Conclusion**

Comme nous pouvons le voir, les pistes d'améliorations sont nombreuses. Le routage multichemin et / ou par partage de charge est toujours un sujet d'actualité, en particulier pour les systèmes hétérogènes que ce soit dans un cadre terrestre ou satellitaire.



# Annexe A

## Plateforme d'expérimentation

Dans cette annexe, nous décrivons plus en détails la plateforme d'expérimentation initialement présentée dans le chapitre 5.

### A.1 Implantation des composants

L'ensemble des composants, clients, serveurs mandataires et serveurs de contenus, ont tous été implantés en C. Nous présentons ici quelques choix techniques, et les possibilités offertes en termes d'évolution.

#### A.1.1 Client

Le client a été implanté en suivant un paradigme de programmation asynchrone. Ainsi, le programme est composé d'un unique *thread*, consistant en une boucle événementielle passive fondée sur l'usage de l'appel système `pselect`. Le drapeau `SOCK_NONBLOCK` est levé sur chaque *socket* grâce à l'appel système `setsockopt`.

Le client peut simuler l'activité de plusieurs utilisateurs. Les informations liées à chaque utilisateur sont stockées dans une structure allouée au démarrage du programme, le fonctionnement du client ne nécessitant alors plus d'appel à `malloc`. Les *sockets* et temporisations de chaque client sont inclus dans la boucle d'attente passive.

Chaque utilisateur possède un état dont la machine à états implante le cycle attente-requête-réponse.

L'attente est d'une durée aléatoire, et utilise une temporisation créée avec `timer_create`. Le choix de la durée repose sur un générateur aléatoire chargé dynamiquement avec `dlopen` et recevant des paramètres lors de l'initialisation. Plusieurs lois de probabilité ont été ainsi implantées sous forme d'objets partagés, permettant de tester facilement l'influence d'une



nouvelle loi. Nous avons ainsi implanté des modules pour les lois uniformes, de Bernoulli, exponentielle, normale, Zipf et dPIN.

Les ouvertures de connexion se font également de manière asynchrone grâce au drapeau `O_NONBLOCK` installé avec l'appel système `fcntl`. Une fois ouverte, un utilisateur procède à une poignée de main `SOCKS5` (facultative) avant de transmettre sa requête. La requête consiste simplement en la transmission de la taille du contenu voulu, celle-ci étant générée aléatoirement. Là encore, un générateur de nombres aléatoires pour une loi donnée est chargé dynamiquement au lancement du client pour cet usage.

Le client attend alors la réception complète du contenu ; il compte les octets reçus mais ceux-ci ne sont pas sauvegardés. Le temps écoulé entre le début de la connexion et la réception complète du contenu est enregistré (ainsi que la taille du contenu) à des fins de statistiques.

Une seconde temporisation est initialisée en début de connexion, et remise à zéro à chaque octet reçu. Si cette dernière dépasse un certain seuil, la connexion est considérée comme bloquée, permettant de détecter certains problèmes (mauvaise gestion du réseau hybride, congestion trop élevée, ...).

À la fin de l'expérimentation, une estimation de la moyenne et de l'écart type du temps de téléchargement sont calculés. Les intervalles de confiance de tous les paramètres de performance sont systématiquement calculés. Les données brutes peuvent être sauvegardées pour effectuer des statistiques plus fines par la suite.

### A.1.2 Serveurs mandataires

Les deux serveurs mandataires, IUG et ING, reposent sur le même code. La différenciation se fait lors de l'initialisation du serveur mandataire.

Une approche asynchrone a également été choisie mais repose ici sur l'utilisation de la bibliothèque haute performance `libevent2`. Cette bibliothèque facilite également la gestion des tampons et des temporisations.

Les serveurs mandataires sont très modulaires, plusieurs organes étant implantés sous forme de modules chargés dynamiquement au démarrage :

- la couche transport pour la partie hybride du réseau ;
- les ordonnanceurs ;
- les algorithmes de routage ;
- divers modules applicatifs pouvant être combinés pour différentes applications.

Chaque module implante un certain nombre de fonctions, obligatoires pour certaines et facultatives pour d'autres, suivant le type de module dont il s'agit.

**Transport** Pour la partie transport, des modules pour UDP, TCP et SCTP ont été créés. MP-TCP repose sur l'utilisation du module TCP.

L'ajout d'un module pour QUIC a été envisagé mais pas encore mené à bien.

Un module est chargé pour chaque lien à exploiter, pas nécessairement le même, permettant d'utiliser des protocoles distincts sur chaque lien (par exemple, un TCP-PEP uniquement pour le lien satellite).

Deux fonctionnements distincts existent :

- un unique canal de communication est établi, et les flux sont multiplexés à travers celui-ci ;
- un nouveau canal de communication est établi pour chaque flux.

Nos modules UDP et SCTP multiplexent les flux tandis qu'une connexion TCP est établie pour chacun d'entre eux.

**Ordonnanceurs** L'ordonnancement est nécessaire lors de l'utilisation de protocoles multiplexant les flux, et décide de l'ordre dans lequel les flux sont multiplexés et donc servis. C'est sous cette forme qu'est implantée notre adaptation de FQ (*Fair Queuing*) pour obtenir un débit équitable entre les différents flux. D'autres ordonnanceurs, favorisant par exemple les flux les plus courts, peuvent être facilement implantés.

**Algorithmes de routage** Ceux-ci implantent les algorithmes présentés et étudiés dans cette thèse.

**Modules applicatifs** Ces modules permettent un traitement des données applicatives qui transitent par les serveurs mandataires. Il est possible de charger plusieurs modules pour appliquer plusieurs traitements. Chaque module consiste en l'implantation d'un *filtering bufferevent*, fonctionnalité propre à la bibliothèque `libevent2`.

Nous avons ainsi implanté :

- le protocole SOCKS5, utile à l'IUG pour recevoir de la part du client les paramètres de destination de la connexion ;
- un mini protocole permettant la transmission des paramètres de la connexion de l'IUG à l'ING ;
- un analyseur pour notre mini protocole applicatif utilisé par le client pour transmettre la taille du contenu souhaité au serveur de contenus, permettant ainsi à l'ING de connaître cette taille ;
- un analyseur HTTP, fondé sur la bibliothèque `http-parser`, permettant par exemple de déterminer la taille des contenus à partir de l'entête `Content-Length` ou de récupérer l'URI exacte de la requête ;
- un serveur mandataire TLS afin de prendre en charge le trafic chiffré (non abouti).

### A.1.3 Serveur

Le serveur suit également le paradigme asynchrone. Il s'articule autour d'une boucle événementielle reposant sur l'appel système `select`. Les allocations de mémoire sont effectuées au démarrage du serveur, évitant de recourir à `malloc` par la suite.

Le serveur attend les nouvelles connexions et les accepte avec l'appel système `accept4`, permettant de mettre du même coup le drapeau `SOCK_NONBLOCK`. L'option `TCP_NODELAY` est également définie à l'aide de `setsockopt` afin de désactiver l'algorithme de `NAGLE`. En effet, celui-ci introduit un petit retard dans l'envoi des paquets qui est de nature à fausser les résultats.

Le serveur s'attend à recevoir la quantité de données à envoyer, qu'il décode avec `scanf`. Les données envoyées correspondent au contenu d'un tampon statiquement alloué dont le contenu importe peu.

Le débit d'émission du serveur peut être limité afin de reproduire le débit limité d'un serveur de contenus. Cette fonctionnalité repose sur l'utilisation de la `qdisc fq` (*fair queueing*) et de l'option `SO_MAX_PACING_RATE` de l'appel système `setsockopt`.

### A.1.4 Estimation des paramètres des liens

L'estimation des paramètres des liens nécessite la capture du trafic sur les différentes interfaces de l'ING. Notre implantation repose, pour cela, sur la bibliothèque `libpcap`. Les événements (émissions, accusés de réception, pertes, ...) sont enregistrés dans un tampon rotatif de taille variable, afin de contenir les événements des 10 dernières secondes.

## A.2 Orchestration

Afin de faciliter les expérimentations, nous avons utilisé différents outils et écrit quelques scripts en python.

Ainsi, les expérimentations à conduire, avec des jeux de paramètres différents, sont mises dans une file d'attente grâce à `RQ` (*Redis Queue*), bibliothèque python dédiée à cet usage. Plusieurs *workers*, suivant le nombre de machines disponibles et leur puissance de calcul, lancent les expérimentations en attente.

Un script python gère le bon déroulement des expérimentations. Il configure préalablement un banc de test : création des `netns`, paramétrage des liens, de leur latence et débit, etc. Les différents composants, serveur de contenus, serveurs mandataires et client sont démarrés successivement et surveillés.

À la fin de l'expérimentation, les composants sont arrêtés et les résultats enregistrés dans une base de données grâce à l'ORM (*Object-Relational*

*Mapping*) du projet `django`. Nous utilisons un champ JSON pour l'enregistrement des paramètres et des résultats afin d'avoir une certaine souplesse pour l'enregistrement d'expérimentations très diverses. Cela nous permet également d'interroger la base de données afin de récupérer les résultats que nous souhaitons tracer, les courbes étant réalisées avec `matplotlib`. La base de données nous permet également de lui déléguer le calcul de certaines métriques telles que le taux d'occupation des liens.



# Bibliographie

- [1] Ahmed ABD, Tarek SAADAWI, Myung LEE et al. “LS-SCTP : a bandwidth aggregation technique for stream control transmission protocol”. In : *Computer Communications* 27.10 (2004), p. 1012-1024.
- [2] Hakim ADHARI et al. “Evaluation of concurrent multipath transfer over dissimilar paths”. In : *Advanced Information Networking and Applications (WAINA), 2011 IEEE Workshops of International Conference on*. IEEE. 2011, p. 708-714.
- [3] Mohammad ALIZADEH et al. “Data Center TCP (DCTCP)”. In : *Proceedings of the ACM SIGCOMM 2010 Conference*. SIGCOMM '10. New Delhi, India : ACM, 2010, p. 63-74. ISBN : 978-1-4503-0201-2. DOI : 10.1145/1851182.1851192.
- [4] Mohammad ALIZADEH et al. “pfabric : Minimal near-optimal data-center transport”. In : *ACM SIGCOMM Computer Communication Review*. T. 43. 4. ACM. 2013, p. 435-446.
- [5] Mark ALLMAN, Hans KRUSE et Shawn OSTERMANN. “An application-level solution to TCP’s satellite inefficiencies”. In : *Proceedings of the First International Workshop on Satellite-based Information Services (WOSBIS)*. 1996.
- [6] M. M. AMBLE et al. “Content-aware caching and traffic management in content distribution networks”. In : *2011 Proceedings IEEE INFOCOM*. Avr. 2011, p. 2858-2866. DOI : 10.1109/INFOCOM.2011.5935123.
- [7] ARCEP. *La régulation de l’Arcep au service des territoires connectés - Edition 2018 - Rapport annuel issu des travaux du GRACO*. Rapp. tech. ARCEP, déc. 2017. URL : [https://www.arcep.fr/uploads/tx%5C\\_gspublication/rapport-GRACO-2018%5C\\_dec2017.pdf](https://www.arcep.fr/uploads/tx%5C_gspublication/rapport-GRACO-2018%5C_dec2017.pdf).
- [8] X. ARTIGA et al. “Terrestrial-satellite integration in dynamic 5G backhaul networks”. In : *2016 8th Advanced Satellite Multimedia Systems Conference and the 14th Signal Processing for Space Communications Workshop (ASMS/SPSC)*. Sept. 2016, p. 1-6. DOI : 10.1109/ASMS-SPSC.2016.7601470.

- [9] Hari BALAKRISHNAN, Hariharan S RAHUL et Srinivasan SESHAN. “An integrated congestion management architecture for Internet hosts”. In : *ACM SIGCOMM Computer Communication Review* 29.4 (1999), p. 175-187.
- [10] Richard BARNES, Jacob HOFFMAN-ANDREWS et James KASTEN. *Automatic Certificate Management Environment (ACME)*. Internet-Draft draft-ietf-acme-acme-08. IETF Secretariat, oct. 2017. URL : <http://www.ietf.org/internet-drafts/draft-ietf-acme-acme-08.txt>.
- [11] Forest BASKETT et al. “Open, closed, and mixed networks of queues with different classes of customers”. In : *Journal of the ACM (JACM)* 22.2 (1975), p. 248-260.
- [12] André-Luc BEYLOT, Riadh DHAOU et Cédric BAUDOIN. “Resource allocation in DVB-RCS satellite systems”. In : *International Conference on Research in Networking*. Springer. 2007, p. 511-521.
- [13] K. BHARGAVAN et al. “Content delivery over TLS : a cryptographic analysis of keyless SSL”. In : *2017 IEEE European Symposium on Security and Privacy (EuroS P)*. Avr. 2017, p. 1-16. DOI : 10.1109/EuroSP.2017.52.
- [14] Thomas BONALD et Alexandre PROUTIERE. “Insensitivity in processor-sharing networks”. In : *Performance Evaluation* 49.1-4 (2002), p. 193-209.
- [15] Thomas BONALD et James ROBERTS. “Scheduling network traffic”. In : *ACM SIGMETRICS Performance Evaluation Review* 34.4 (2007), p. 29-35.
- [16] E. BOUTTIER et al. “Analysis of Content Size Based Routing Schemes in Hybrid Satellite & Terrestrial Networks”. In : *2016 IEEE Global Communications Conference (GLOBECOM)*. Déc. 2016. DOI : 10.1109/GLOCOM.2016.7841601.
- [17] E. BOUTTIER et al. “Heterogeneous Multipath Networks : Flow vs. Packet Based Routing in a Size-Aware Context”. In : *2017 IEEE Global Communications Conference (GLOBECOM)*. Déc. 2017. DOI : 10.1109/GLOCOM.2017.8255095.
- [18] E. BOUTTIER et al. “Improving content delivery with size-aware routing in hybrid satellite / terrestrial networks”. In : *2018 IEEE International Conference on Communications (ICC)*. (à paraître). Mai 2018.
- [19] Justin A BOYAN et Michael L LITTMAN. “Packet routing in dynamically changing networks : A reinforcement learning approach”. In : *Advances in neural information processing systems*. 1994, p. 671-678.

- [20] Lukasz BUDZISZ et al. “On concurrent multipath transfer in SCTP-based handover scenarios”. In : *Communications, 2009. ICC'09. IEEE International Conference on*. IEEE. 2009, p. 1-6.
- [21] Matt CALDER et al. “Analyzing the Performance of an Anycast CDN”. In : *Proceedings of the 2015 Internet Measurement Conference. IMC '15*. Tokyo, Japan : ACM, 2015, p. 531-537. ISBN : 978-1-4503-3848-6. DOI : 10.1145/2815675.2815717.
- [22] Yu CAO, Mingwei XU et Xiaoming FU. “Delay-based congestion control for multipath TCP”. In : *2012 20th IEEE International Conference on Network Protocols (ICNP)*. Oct. 2012, p. 1-10. DOI : 10.1109/ICNP.2012.6459978.
- [23] K. CHEBROLU et R. R. RAO. “Bandwidth aggregation for real-time applications in heterogeneous wireless networks”. In : *IEEE Transactions on Mobile Computing* 5.4 (avr. 2006), p. 388-403. ISSN : 1536-1233. DOI : 10.1109/TMC.2006.1599407.
- [24] Kameswari CHEBROLU, Bhaskaran RAMAN et Ramesh R. RAO. “A Network Layer Approach to Enable TCP over Multiple Interfaces”. In : *Wireless Networks* 11.5 (sept. 2005), p. 637-650. ISSN : 1572-8196. DOI : 10.1007/s11276-005-3518-5.
- [25] Ludmila CHERKASOVA et Gianfranco CIARDO. “Role of Aging, Frequency, and Size in Web Cache Replacement Policies”. In : *Proceedings of the 9th International Conference on High-Performance Computing and Networking. HPCN Europe 2001*. Berlin, Heidelberg : Springer-Verlag, 2001, p. 114-123. ISBN : 3-540-42293-5. URL : <http://dl.acm.org/citation.cfm?id=645565.658748>.
- [26] Dah-Ming CHIU et Raj JAIN. “Analysis of the increase and decrease algorithms for congestion avoidance in computer networks”. In : *Computer Networks and ISDN systems* 17.1 (1989), p. 1-14.
- [27] J. CHU et al. *Increasing TCP's Initial Window*. RFC 6928. RFC Editor, avr. 2013. URL : <http://www.rfc-editor.org/rfc/rfc6928.txt>.
- [28] CISCO. *Cisco Visual Networking Index : Forecast and Methodology, Global content delivery network internet traffic, 2016-2021*. Rapp. tech. 2017. URL : [https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/complete-white-paper-c11-481360.html#\\_Toc484813991](https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/complete-white-paper-c11-481360.html#_Toc484813991).
- [29] CISCO. *The Zettabyte Era : Trends and Analysis*. URL : <https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/vni-hyperconnectivity-wp.html>.



- [30] Jake COBB et Hala ELAARAG. “Web proxy cache replacement scheme based on back-propagation neural network”. In : *Journal of Systems and Software* 81.9 (2008), p. 1539-1558.
- [31] Quentin De CONINCK et Olivier BONAVENTURE. *Multipath Extension for QUIC*. Internet-Draft draft-deconinck-quic-multipath-00. Work in Progress. Internet Engineering Task Force, mar. 2018. 28 p. URL : <https://datatracker.ietf.org/doc/html/draft-deconinck-quic-multipath-00>.
- [32] Laurent COUSTET. *MLVPN - MultiLink Virtual Public Network*. URL : <https://zhome.github.io/MLVPN/>.
- [33] *D3.3.2 Multi Access Networking Architecture*. BATS Project, 2014. URL : [https://web.archive.org/web/20160810045407/http://batsproject.eu/index.php?option=com\\_content&view=article&id=83&Itemid=488](https://web.archive.org/web/20160810045407/http://batsproject.eu/index.php?option=com_content&view=article&id=83&Itemid=488).
- [34] Quentin DE CONINCK et Olivier BONAVENTURE. “Multipath QUIC : Design and Evaluation”. In : *13th International Conference on emerging Networking EXperiments and Technologies (CoNEXT 2017)*. 2017. URL : <http://multipath-quic.org>.
- [35] Danny DE VLEESCHAUWER et Dave C ROBINSON. “Optimum caching strategies for a telco CDN”. In : *Bell Labs Technical Journal* 16.2 (2011), p. 115-132.
- [36] Stephen E. DEERING et Robert M. HINDEN. *Internet Protocol, Version 6 (IPv6) Specification*. RFC 2460. RFC Editor, déc. 1998. URL : <http://www.rfc-editor.org/rfc/rfc2460.txt>.
- [37] A. DEMERS, S. KESHAV et S. SHENKER. “Analysis and Simulation of a Fair Queueing Algorithm”. In : *Symposium Proceedings on Communications Architectures & Protocols*. SIGCOMM '89. Austin, Texas, USA : ACM, 1989, p. 1-12. ISBN : 0-89791-332-9. DOI : 10.1145/75246.75248.
- [38] Thomas DREIBHOLZ et al. “On the use of concurrent multipath transfer over asymmetric paths”. In : *Global Telecommunications Conference (GLOBECOM 2010), 2010 IEEE*. IEEE. 2010, p. 1-6.
- [39] Nandita DUKKIPATI et Nick MCKEOWN. “Why Flow-completion Time is the Right Metric for Congestion Control”. In : *SIGCOMM Comput. Commun. Rev.* 36.1 (jan. 2006), p. 59-62. ISSN : 0146-4833. DOI : 10.1145/1111322.1111336.
- [40] E. DUROS et al. *A Link-Layer Tunneling Mechanism for Unidirectional Links*. RFC 3077. RFC Editor, mar. 2001. URL : <http://www.rfc-editor.org/rfc/rfc3077.txt>.

- [41] Lars EGGERT, John HEIDEMANN et Joe TOUCH. “Effects of ensemble-TCP”. In : *ACM SIGCOMM Computer Communication Review* 30.1 (2000), p. 15-29.
- [42] ETSI. *Satellite Earth Stations and Systems (SES) ; Hybrid FSS satellite/terrestrial network architecture for high speed broadband access*. Rapp. tech. European Telecommunications Standards Institute (ETSI), 2015.
- [43] ETSI. *Satellite Earth Stations and Systems (SES) ; Multi-link routing scheme in hybrid access network with heterogeneous links*. Rapp. tech. European Telecommunications Standards Institute (ETSI), 2017.
- [44] EUTELSAT. *Lancement réussi du satellite KA-SAT d’Eutelsat par une fusée Proton d’Ils depuis le cosmodrome de Baïkonour*. 2010. URL : <http://www.eutelsat.com/news/compress/fr/2010/html/CP5910-KA-SAT-success/CP5910-KA-SAT-success.html>.
- [45] Dino FARINACCI et al. *The Locator/ID Separation Protocol (LISP)*. RFC 6830. Jan. 2013. URL : <https://rfc-editor.org/rfc/rfc6830.txt>.
- [46] F. Le FAUCHEUR et al. *Content Distribution Network Interconnection (CDNI) Logging Interface*. RFC 7937. RFC Editor, août 2016. URL : <http://www.rfc-editor.org/rfc/rfc7937.txt>.
- [47] Adrienne Porter FELT et al. “Measuring HTTPS adoption on the web”. In : *26th USENIX Security Symposium*. 2017, p. 1323-1338.
- [48] Frederic FIEAU, Stephan EMILE et Sanjay MISHRA. *CDNI extensions for HTTPS delegation*. Internet-Draft draft-fieau-cdni-interfaces-https-delegation-03. IETF Secretariat, jan. 2018. URL : <http://www.ietf.org/internet-drafts/draft-fieau-cdni-interfaces-https-delegation-03.txt>.
- [49] R. FIELDING, Y. LAFON et J. RESCHKE. *Hypertext Transfer Protocol (HTTP/1.1) : Range Requests*. RFC 7233. RFC Editor, juin 2014. URL : <http://www.rfc-editor.org/rfc/rfc7233.txt>.
- [50] R. FIELDING, M. NOTTINGHAM et J. RESCHKE. *Hypertext Transfer Protocol (HTTP/1.1) : Caching*. RFC 7234. RFC Editor, juin 2014. URL : <http://www.rfc-editor.org/rfc/rfc7234.txt>.
- [51] Roy T. FIELDING et al. *Hypertext Transfer Protocol – HTTP/1.1*. RFC 2616. RFC Editor, juin 1999. URL : <http://www.rfc-editor.org/rfc/rfc2616.txt>.
- [52] Marco FIORE, Claudio CASSETTI et Giulio GALANTE. “Concurrent multipath communication for real-time traffic”. In : *Computer Communications* 30.17 (2007), p. 3307-3320.

- [53] A. FORD et al. *TCP Extensions for Multipath Operation with Multiple Addresses*. RFC 6824. RFC Editor, jan. 2013. URL : <http://www.rfc-editor.org/rfc/rfc6824.txt>.
- [54] Roberta FRACCHIA et al. “WiSE : Best-path selection in wireless multihoming environments”. In : *IEEE Transactions on Mobile Computing* 6.10 (2007).
- [55] Vladimir FRIEDMAN. “VDSL Technology Issues—An Overview”. In : *About Analog Dialogue* (2000), p. 50.
- [56] H. FUJINOKI. “Multi-path BGP (MBGP) : A solution for improving network bandwidth utilization and defense against link failures in inter-domain routing”. In : *2008 16th IEEE International Conference on Networks*. Déc. 2008, p. 1-6. DOI : 10.1109/ICON.2008.4772612.
- [57] S Jamaloddin GOLESTANI. “A self-clocked fair queueing scheme for broadband applications”. In : *INFOCOM’94. Networking for Global Communications., 13th Proceedings IEEE*. IEEE. 1994, p. 636-646.
- [58] R. GONZALEZ et al. “Hybrid delay-based congestion control for multipath TCP”. In : *2016 18th Mediterranean Electrotechnical Conference (MELECON)*. Avr. 2016, p. 1-6. DOI : 10.1109/MELCON.2016.7495389.
- [59] GOOGLE. *Google IPv6 Statistics*. Rapp. tech. Google, jan. 2018. URL : <https://www.google.com/intl/en/ipv6/statistics.html>.
- [60] S. GUNDAVELLI et al. *Proxy Mobile IPv6*. RFC 5213. RFC Editor, août 2008. URL : <http://www.rfc-editor.org/rfc/rfc5213.txt>.
- [61] Andrei GURTOV et Tatiana POLISHCHUK. “Secure multipath transport for legacy Internet applications”. In : *Broadband Communications, Networks, and Systems, 2009. BROADNETS 2009. Sixth International Conference on*. IEEE. 2009, p. 1-8.
- [62] Michio HONDA et al. “Multipath congestion control for shared bottleneck”. In : *Proc. PFLDNeT workshop*. 2009, p. 19-24.
- [63] C. HOPPS. *Analysis of an Equal-Cost Multi-Path Algorithm*. RFC 2992. RFC Editor, nov. 2000.
- [64] Hung-Yun HSIEH et Raghupathy SIVAKUMAR. “pTCP : An end-to-end transport layer protocol for striped connections”. In : *Network Protocols, 2002. Proceedings. 10th IEEE International Conference on*. IEEE. 2002, p. 24-33.
- [65] Chung-Ming HUANG et Ching-Hsien TSAI. “WiMP-SCTP : Multipath transmission using stream control transmission protocol (SCTP) in wireless networks”. In : *Advanced Information Networking and Applications Workshops, 2007, AINAW’07. 21st International Conference on*. T. 1. IEEE. 2007, p. 209-214.

- [66] *HughesNet Gen5 : Déjà plus de 100 000 abonnés Internet par satellite*. Internet par satellite, 2017. URL : <https://www.internetparsatellite.net/index.php/t/2/news/2460/hughesnet-gen5-deja-plus-abonnes-internet.html>.
- [67] ICANN. *Available Pool of Unallocated IPv4 Internet Addresses Now Completely Emptied*. URL : <https://www.icann.org/en/system/files/press-materials/release-03feb11-en.pdf>.
- [68] *Internet par satellite : Eutelsat refuse les nouveaux clients dans 28 départements*. Next Impact, 2015. URL : <https://www.nextinpact.com/news/95929-satellite-eutelsat-refuse-nouveaux-clients-sur-certains-departements.htm>.
- [69] Jana IYENGAR et Martin THOMSON. *QUIC : A UDP-Based Multiplexed and Secure Transport*. Internet-Draft draft-ietf-quic-transport-11. IETF Secretariat, avr. 2018. URL : <http://www.ietf.org/internet-drafts/draft-ietf-quic-transport-11.txt>.
- [70] Janardhan R IYENGAR, Paul D AMER et Randall STEWART. “Concurrent multipath transfer using SCTP multihoming over independent end-to-end paths”. In : *IEEE/ACM Transactions on networking* 14.5 (2006), p. 951-964.
- [71] M. JIA et al. “Broadband Hybrid Satellite-Terrestrial Communication Systems Based on Cognitive Radio toward 5G”. In : *IEEE Wireless Communications* 23.6 (déc. 2016), p. 96-106. ISSN : 1536-1284. DOI : 10.1109/MWC.2016.1500108WC.
- [72] JOURNAL OFFICIEL DE L’UNION EUROPÉENNE. *Lignes directrices de l’UE pour l’application des règles relatives aux aides d’État dans le cadre du déploiement rapide des réseaux de communication à haut débit*. Union européenne, 2013. URL : <https://eur-lex.europa.eu/legal-content/FR/TXT/?uri=CELEX%3A52013XC0126%2801%29>.
- [73] Srikanth KANDULA et al. “Dynamic load balancing without packet reordering”. In : *ACM SIGCOMM Computer Communication Review* 37.2 (2007), p. 51-62.
- [74] Jussi KANGASHARJU, James ROBERTS et Keith W ROSS. “Object replication strategies in content distribution networks”. In : *Computer Communications* 25.4 (2002), p. 376-383.
- [75] Frank P KELLY, Aman K MAULLOO et David KH TAN. “Rate control for communication networks : shadow prices, proportional fairness and stability”. In : *Journal of the Operational Research society* 49.3 (1998), p. 237-252.

- [76] Frank KELLY et Thomas VOICE. “Stability of End-to-end Algorithms for Joint Routing and Rate Control”. In : *ACM SIGCOMM Computer Communication Review* 35.2 (avr. 2005), p. 5-12. ISSN : 0146-4833. DOI : 10.1145/1064413.1064415.
- [77] Ramin KHALILI et al. *Opportunistic Linked-Increases Congestion Control Algorithm for MPTCP*. Internet-Draft draft-khalili-mptcp-congestion-control-05. IETF Secretariat, juil. 2014. URL : <https://www.ietf.org/archive/id/draft-khalili-mptcp-congestion-control-05.txt>.
- [78] Han Ah KIM, Bong-hwan OH et Jaiyong LEE. “Improvement of MPTCP Performance in heterogeneous network using packet scheduling mechanism”. In : *2012 18th Asia-Pacific Conference on Communications (APCC)*. Oct. 2012, p. 842-847. DOI : 10.1109/APCC.2012.6388200.
- [79] Kyu-Han KIM et Kang G SHIN. “PRISM : Improving the performance of inverse-multiplexed TCP in wireless networks”. In : *IEEE Transactions on Mobile Computing* 6.12 (2007), p. 1297-1312.
- [80] Bernd KLASSEN. “Socially Augmented Content Distribution in Hybrid Networks”. Thèse de doct. University of Luxembourg, Luxembourg, Luxembourg, 2013.
- [81] Charles KRASIC et Jonathan WALPOLE. “Priority-progress streaming for quality-adaptive multimedia”. In : *Proceedings of the ninth ACM international conference on Multimedia*. ACM. 2001, p. 463-464.
- [82] K. LAHEY. *TCP Problems with Path MTU Discovery*. RFC 2923. RFC Editor, sept. 2000. URL : <http://www.rfc-editor.org/rfc/rfc2923.txt>.
- [83] Jason LEE et al. “Applied techniques for high bandwidth data transfers across wide area networks”. In : *Lawrence Berkeley National Laboratory* (2001).
- [84] Ka-Cheong LEUNG, Victor OK LI et Daiqin YANG. “An overview of packet reordering in transmission control protocol (TCP) : problems, solutions, and challenges”. In : *IEEE transactions on parallel and distributed systems* 18.4 (2007).
- [85] Ka-Cheong LEUNG et Changming MA. “Enhancing TCP performance to persistent packet reordering”. In : *Journal of Communications and Networks* 7.3 (2005), p. 385-393.
- [86] Bo LI et al. “On the optimal placement of web proxies in the internet”. In : *INFOCOM’99. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*. T. 3. IEEE. 1999, p. 1282-1290.

- [87] Dr. Tony LI et Curtis VILLAMIZAR. *IS-IS Optimized Multipath (ISIS-OMP)*. Internet-Draft draft-ietf-isis-omp-01. Work in Progress. Internet Engineering Task Force, fév. 1999. 8 p. URL : <https://datatracker.ietf.org/doc/html/draft-ietf-isis-omp-01>.
- [88] M. LI et al. “Multipath Transmission for the Internet : A Survey”. In : *IEEE Communications Surveys Tutorials* 18.4 (Fourthquarter 2016), p. 2887-2925. ISSN : 1553-877X. DOI : 10.1109/COMST.2016.2586112.
- [89] Qiao LI et al. “A novel cache replacement policy for ISP merged CDN”. In : *Parallel and Distributed Systems (ICPADS), 2012 IEEE 18th International Conference on*. IEEE. 2012, p. 708-709.
- [90] Jianxin LIAO, Jingyu WANG et Xiaomin ZHU. “cmpSCTP : An extension of SCTP to support concurrent multi-path transfer”. In : *Communications, 2008. ICC’08. IEEE International Conference on*. IEEE. 2008, p. 5762-5766.
- [91] Yeon-sup LIM et al. “Cross-layer path management in multi-path transport protocol for mobile devices”. In : *INFOCOM, 2014 Proceedings IEEE*. IEEE. 2014, p. 1815-1823.
- [92] Reiner LUDWIG et Randy H KATZ. “The Eifel algorithm : making TCP robust against spurious retransmissions”. In : *ACM SIGCOMM Computer Communication Review* 30.1 (2000), p. 30-36.
- [93] G. MALKIN. *Nortel Networks Multi-link Multi-node PPP Bundle Discovery Protocol*. RFC 2701. RFC Editor, sept. 1999. URL : <http://www.rfc-editor.org/rfc/rfc2701.txt>.
- [94] Shiwen MAO et al. “MRTP : a multiflow real-time transport protocol for ad hoc networks”. In : *IEEE Transactions on Multimedia* 8.2 (2006), p. 356-369.
- [95] M. MATHIS et al. *TCP Selective Acknowledgment Options*. RFC 2018. RFC Editor, oct. 1996. URL : <http://www.rfc-editor.org/rfc/rfc2018.txt>.
- [96] P. E. MCKENNEY. “Stochastic fairness queueing”. In : *INFOCOM ’90, Ninth Annual Joint Conference of the IEEE Computer and Communication Societies. The Multiple Facets of Integration. Proceedings, IEEE*. Juin 1990, 733-740 vol.2. DOI : 10.1109/INFCOM.1990.91316.
- [97] Michael MITZENMACHER. “A Brief History of Generative Models for Power Law and Lognormal Distributions”. In : *Internet Mathematics* 1.2 (2004), p. 226-251. DOI : 10.1080/15427951.2004.10129088.
- [98] Michael MITZENMACHER. “Dynamic models for file sizes and double pareto distributions”. In : *Internet Mathematics* 1.3 (2004), p. 305-333.

- [99] Michael MITZENMACHER. “The power of two choices in randomized load balancing”. In : *IEEE Transactions on Parallel and Distributed Systems* 12.10 (2001), p. 1094-1104.
- [100] Robert MOSKOWITZ et al. *Host Identity Protocol Version 2 (HIPv2)*. RFC 7401. Avr. 2015. DOI : 10.17487/RFC7401. URL : <https://rfc-editor.org/rfc/rfc7401.txt>.
- [101] David MURRAY et al. “An analysis of changing enterprise network traffic characteristics”. In : *Communications (APCC), 2017 23rd Asia-Pacific Conference on*. IEEE. 2017, p. 1-6.
- [102] R. MURRAY et B. NIVEN-JENKINS. *Content Delivery Network Interconnection (CDNI) Control Interface / Triggers*. RFC 8007. RFC Editor, déc. 2016. URL : <http://www.rfc-editor.org/rfc/rfc8007.txt>.
- [103] J. NAGLE. *On Packet Switches With Infinite Storage*. RFC 970. RFC Editor, déc. 1985. URL : <http://www.rfc-editor.org/rfc/rfc970.txt>.
- [104] *Netflix Open Connect Appliance Deployment Guide, Fill window*. Netflix, 2018. URL : <https://openconnect.netflix.com/deploymentguide.pdf>.
- [105] B. NIVEN-JENKINS et R. van BRANDENBURG. *Request Routing Redirection Interface for Content Delivery Network (CDN) Interconnection*. RFC 7975. RFC Editor, oct. 2016. URL : <http://www.rfc-editor.org/rfc/rfc7975.txt>.
- [106] B. NIVEN-JENKINS et al. *Content Delivery Network Interconnection (CDNI) Metadata*. RFC 8006. RFC Editor, déc. 2016. URL : <http://www.rfc-editor.org/rfc/rfc8006.txt>.
- [107] E. NORDMARK et M. BAGNULO. *Shim6 : Level 3 Multihoming Shim Protocol for IPv6*. RFC 5533. RFC Editor, juin 2009. URL : <http://www.rfc-editor.org/rfc/rfc5533.txt>.
- [108] *NordNet suspend la vente d’offres Eutelsat dans la moitié de la France*. Next Impact, 2015. URL : <https://www.nextinpact.com/news/95939-nordnet-suspend-vente-doffres-eutelsat-dans-moitie-france.htm>.
- [109] Thiago Queiroz de OLIVEIRA et Marcial P FERNANDEZ. “FuzzyCDN : fuzzy redirection algorithm”. In : *Advanced Information Networking and Applications (AINA), 2013 IEEE 27th International Conference on*. IEEE. 2013, p. 437-444.
- [110] *OpenSAND*. URL : <http://opensand.org/content/home.php>.

- [111] Al-Mukaddim Khan PATHAN et Rajkumar BUYYA. “A taxonomy and survey of content delivery networks”. In : *Grid Computing and Distributed Systems Laboratory, University of Melbourne, Technical Report 4* (2007).
- [112] Gang PENG. “CDN : Content distribution network”. In : *arXiv preprint cs/0411069* (2004).
- [113] Q. PENG et al. “Multipath TCP : Analysis, Design, and Implementation”. In : *IEEE/ACM Transactions on Networking* 24.1 (fév. 2016), p. 596-609. ISSN : 1063-6692. DOI : 10.1109/TNET.2014.2379698.
- [114] *Percentage of Web Pages Loaded by Firefox Using HTTPS*. Let’s Encrypt Stats. URL : <https://letsencrypt.org/stats/#percent-pageloads>.
- [115] C. PERKINS. *IP Mobility Support for IPv4, Revised*. RFC 5944. RFC Editor, nov. 2010. URL : <http://www.rfc-editor.org/rfc/rfc5944.txt>.
- [116] C. PERKINS, D. JOHNSON et J. ARKKO. *Mobility Support in IPv6*. RFC 6275. RFC Editor, juil. 2011. URL : <http://www.rfc-editor.org/rfc/rfc6275.txt>.
- [117] Federico PEROTTO, Claudio CASSETTI et Giulio GALANTE. “SCTP-based transport protocols for concurrent multipath transfer”. In : *Wireless Communications and Networking Conference, 2007. WCNC 2007. IEEE*. IEEE. 2007, p. 2969-2974.
- [118] Graham PETERS et al. “The bats project”. In : *Future Network and Mobile Summit (FutureNetworkSummit), 2013*. IEEE. 2013, p. 1-8.
- [119] Dhananjay S PHATAK et Tom GOFF. “A novel mechanism for data streaming across multiple IP links for improving throughput and reliability in mobile environments”. In : *INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*. T. 2. IEEE. 2002, p. 773-781.
- [120] *Plan France Très Haut Débit*. République Française, 2015. URL : <http://francethd.fr/>.
- [121] J. POSTEL et J. REYNOLDS. *File Transfer Protocol*. STD 9. RFC Editor, oct. 1985. URL : <http://www.rfc-editor.org/rfc/rfc959.txt>.
- [122] J. QADIR et al. “Exploiting the Power of Multiplicity : A Holistic Survey of Network-Layer Multipath”. In : *IEEE Communications Surveys Tutorials* 17.4 (Fourthquarter 2015), p. 2176-2213. ISSN : 1553-877X. DOI : 10.1109/COMST.2015.2453941.



- [123] Lili QIU, Venkata N PADMANABHAN et Geoffrey M VOELKER. “On the placement of web server replicas”. In : *INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*. T. 3. IEEE. 2001, p. 1587-1596.
- [124] William J REED et Murray JORGENSEN. “The double Pareto-lognormal distribution—a new parametric model for size distributions”. In : *Communications in Statistics-Theory and Methods* 33.8 (2004), p. 1733-1753.
- [125] *Report : State of the Web, HTTPS Requests*. HTTP Archive. URL : <https://httparchive.org/reports/state-of-the-web#pctHttps>.
- [126] Julian RESCHKE et Salvatore LORETO. ‘Out-Of-Band’ Content Coding for HTTP. Internet-Draft draft-reschke-http-oob-encoding-12. IETF Secretariat, juin 2017. URL : <http://www.ietf.org/internet-drafts/draft-reschke-http-oob-encoding-12.txt>.
- [127] P. SAROLAHTI et al. *Forward RTO-Recovery (F-RTO) : An Algorithm for Detecting Spurious Retransmission Timeouts with TCP*. RFC 5682. RFC Editor, sept. 2009.
- [128] T. SAVOLAINEN, J. KATO et T. LEMON. *Improved Recursive DNS Server Selection for Multi-Interfaced Nodes*. RFC 6731. RFC Editor, déc. 2012. URL : <http://www.rfc-editor.org/rfc/rfc6731.txt>.
- [129] Fabian SCHNEIDER et al. “Pitfalls in HTTP traffic measurements and analysis”. In : *International Conference on Passive and Active Network Measurement*. Springer. 2012, p. 242-251.
- [130] J. SEEDORF et al. *Content Delivery Network Interconnection (CDNI) Request Routing : Footprint and Capabilities Semantics*. RFC 8008. RFC Editor, déc. 2016. URL : <http://www.rfc-editor.org/rfc/rfc8008.txt>.
- [131] Kenneth C SEVCIK et Isi MITRANI. “The distribution of queuing network states at input and output instants”. In : *Journal of the ACM (JACM)* 28.2 (1981), p. 358-371.
- [132] Yaron SHEFFER et al. *Use of Short-Term, Automatically-Renewed (STAR) Certificates to Delegate Authority over Web Sites*. Internet-Draft draft-ietf-acme-star-00. IETF Secretariat, juin 2017. URL : <http://www.ietf.org/internet-drafts/draft-ietf-acme-star-00.txt>.
- [133] Manpreet SINGH, Prashant PRADHAN et Paul FRANCIS. “MPAT : aggregate TCP congestion management as a building block for Internet QoS”. In : *Network Protocols, 2004. ICNP 2004. Proceedings of the 12th IEEE International Conference on*. IEEE. 2004, p. 129-138.

- [134] K. SKLOWER et al. *The PPP Multilink Protocol (MP)*. RFC 1717. RFC Editor, nov. 1994. URL : <http://www.rfc-editor.org/rfc/rfc1717.txt>.
- [135] K. SKLOWER et al. *The PPP Multilink Protocol (MP)*. RFC 1990. RFC Editor, août 1996. URL : <http://www.rfc-editor.org/rfc/rfc1990.txt>.
- [136] Donald R SMITH. “A new proof of the optimality of the shortest remaining processing time discipline”. In : *Operations Research* 26.1 (1978), p. 197-199.
- [137] D. STEBILA et N. SULLIVAN. “An Analysis of TLS Handshake Proxying”. In : *2015 IEEE Trustcom/BigDataSE/ISPA*. T. 1. Août 2015, p. 279-286. DOI : 10.1109/Trustcom.2015.385.
- [138] Shaler STIDHAM JR. “A last word on  $L = \lambda W$ ”. In : *Operations Research* 22.2 (1974), p. 417-421.
- [139] M. SZYMANIAK, G. PIERRE et M. van STEEN. “Latency-driven replica placement”. In : *The 2005 Symposium on Applications and the Internet*. Jan. 2005, p. 399-405. DOI : 10.1109/SAINT.2005.37.
- [140] John TADROUS, Atila ERYILMAZ et Hesham EL GAMAL. “Proactive content distribution for dynamic content”. In : *Information Theory Proceedings (ISIT), 2013 IEEE International Symposium on*. IEEE. 2013, p. 1232-1236.
- [141] *THD-Sat*. CNES. URL : <https://thdsat.cnes.fr/en/THDSAT/index.htm>.
- [142] B. TRAMMELL et M. KUEHLEWIND. *Report from the IAB Workshop on Stack Evolution in a Middlebox Internet (SEMI)*. RFC 7663. RFC Editor, oct. 2015. URL : <http://www.rfc-editor.org/rfc/rfc7663.txt>.
- [143] Cheng-Lin TSAO et Raghupathy SIVAKUMAR. “On effectively exploiting multiple wireless interfaces in mobile hosts”. In : *Proceedings of the 5th international conference on Emerging networking experiments and technologies*. ACM. 2009, p. 337-348.
- [144] Curtis VILLAMIZAR. *MPLS Optimized Multipath (MPLS-OMP)*. Internet-Draft draft-villamizar-mpls-omp-01. Work in Progress. Internet Engineering Task Force, fév. 1999. 26 p. URL : <https://datatracker.ietf.org/doc/html/draft-villamizar-mpls-omp-01>.
- [145] Curtis VILLAMIZAR. *OSPF Optimized Multipath (OSPF-OMP)*. Internet-Draft draft-ietf-ospf-omp-02. Work in Progress. Internet Engineering Task Force, fév. 1999. 38 p. URL : <https://datatracker.ietf.org/doc/html/draft-ietf-ospf-omp-02>.

- [146] Anwar WALID et al. *Balanced Linked Adaptation Congestion Control Algorithm for MPTCP*. Internet-Draft draft-walid-mptcp-congestion-control-04. IETF Secretariat, jan. 2016. URL : <http://www.ietf.org/internet-drafts/draft-walid-mptcp-congestion-control-04.txt>.
- [147] Bing WANG et al. "Multipath live streaming via TCP : scheme, performance and benefits". In : *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)* 5.3 (2009), p. 25.
- [148] Limin WANG, Vivek PAI et Larry PETERSON. "The Effectiveness of Request Redirection on CDN Robustness". In : *ACM SIGOPS Operating Systems Review* 36.SI (déc. 2002), p. 345-360. ISSN : 0163-5980. DOI : 10.1145/844128.844160.
- [149] Tim WAUTERS et al. "Load balancing through efficient distributed content placement". In : *Next Generation Internet Networks, 2005*. IEEE. 2005, p. 99-105.
- [150] Patrick WENDELL et Michael J FREEDMAN. "Going viral : flash crowds in an open CDN". In : *Proceedings of the 2011 ACM SIGCOMM conference on Internet measurement conference*. ACM. 2011, p. 549-558.
- [151] Stan ZACHARY. "A note on insensitivity in stochastic networks". In : *Journal of applied probability* 44.1 (2007), p. 238-248.
- [152] David ZATS et al. "DeTail : Reducing the Flow Completion Time Tail in Datacenter Networks". In : *Proceedings of the ACM SIGCOMM 2012 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*. SIGCOMM '12. Helsinki, Finland : ACM, 2012, p. 139-150. ISBN : 978-1-4503-1419-0. DOI : 10.1145/2342356.2342390.
- [153] Ming ZHANG et al. "A Transport Layer Approach for Improving End-to-End Performance and Robustness Using Redundant Paths." In : *USENIX Annual Technical Conference, General Track*. 2004, p. 99-112.
- [154] Ming ZHANG et al. "RR-TCP : a reordering-robust TCP with DSA-CK". In : *Network Protocols, 2003. Proceedings. 11th IEEE International Conference on*. IEEE. 2003, p. 95-106.
- [155] Yiping ZHANG, Shilong MA et Jun HUANG. "A simple approach of improving dns based cdn video sharing system". In : *Information Networking, 2008. ICOIN 2008. International Conference on*. IEEE. 2008, p. 1-5.

# Liste des communications

## Conférences internationales avec comité de lecture

- [1] E. BOUTTIER et al. “Analysis of Content Size Based Routing Schemes in Hybrid Satellite & Terrestrial Networks”. In : *2016 IEEE Global Communications Conference (GLOBECOM)*. Déc. 2016. DOI : 10 . 1109/GLOCOM.2016.7841601
- [2] E. BOUTTIER et al. “Heterogeneous Multipath Networks : Flow vs. Packet Based Routing in a Size-Aware Context”. In : *2017 IEEE Global Communications Conference (GLOBECOM)*. Déc. 2017. DOI : 10 . 1109/GLOCOM.2017.8255095
- [3] E. BOUTTIER et al. “Improving content delivery with size-aware routing in hybrid satellite / terrestrial networks”. In : *2018 IEEE International Conference on Communications (ICC)*. (à paraître). Mai 2018

## Présentations

- [1] Exposé « Analysis of size-aware routing schemes in hybrid terrestrial / satellite networks », Journée NetSatDay, Toulouse, juin 2016
- [2] Session poster « Livraison de contenus sur réseau hybride satellite / terrestre », Journées CNES Jeunes Chercheurs, Toulouse, octobre 2016
- [3] Exposé « Improving content delivery with size-aware routing in hybrid satellite / terrestrial networks », Journée NetSatDay, Toulouse, juin 2018



---

## Résumé

---

Alors que les zones urbaines connaissent actuellement le déploiement de la fibre optique, les zones rurales, moins rentables, sont délaissées. On constate ainsi une augmentation de la fracture numérique amenant les utilisateurs des zones rurales à se tourner vers d'autres moyens d'accès tels que les accès Internet par satellite.

Cependant, ces derniers souffrent d'une limitation qui est le délai important induit par le temps de propagation du signal entre la terre et l'orbite géostationnaire. Pour contourner cette limitation, nous nous intéressons dans cette thèse à l'utilisation simultanée d'un réseau d'accès terrestre et d'un réseau d'accès satellite. Nous proposerons ainsi un mécanisme de routage intelligent choisissant le réseau le plus adapté pour la livraison des contenus suivant leur taille.

**Mots clés :** distribution de contenus, réseau hybride satellite / terrestre

---

## Abstract

---

Whereas urban areas are currently experiencing the deployment of optical fiber, rural areas, less profitable, are left behind. There is thus an increase in the digital divide leading users in rural areas to turn to other means of access such as satellite Internet access.

However, satellite Internet access suffers from a limitation which is the significant delay induced by the propagation time of the signal between the earth and the geostationary orbit. To overcome this limitation, we studied in this thesis the simultaneous use of a terrestrial access network and a satellite access network. We thus proposed an intelligent routing mechanism choosing the most suitable network for the content delivery based on its size.

**Keywords :** content delivery, satellite / terrestrial hybrid network