



HAL
open science

Adaptation de la couche transport des systèmes de communication hybrides satellite/terrestre

Ihsane Tou

► **To cite this version:**

Ihsane Tou. Adaptation de la couche transport des systèmes de communication hybrides satellite/terrestre. Réseaux et télécommunications [cs.NI]. Université Paul Sabatier - Toulouse III, 2012. Français. NNT: . tel-00807583

HAL Id: tel-00807583

<https://theses.hal.science/tel-00807583>

Submitted on 4 Apr 2013

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 :

Université Toulouse 3 Paul Sabatier (UT3 Paul Sabatier)

Présentée et soutenue par :

Ihsane TOU

le 13 décembre 2012

Titre :

Adaptation de la couche transport des systèmes de communication hybrides
satellite/terrestre

École doctorale et discipline ou spécialité :

EDSYS : Informatique 4200018

Unité de recherche :

LAAS-CNRS

Directeur(s) de Thèse :

Thierry GAYRAUD

Pascal BERTHOU

Jury :

Laurent FRANCK - Rapporteur

Yacine GHAMRI-DOUDANE - Rapporteur

Francine KRIEF - Examineur

Michel DIAZ - Examineur

Thierry GAYRAUD - Directeur de thèse

Pascal BERTHOU - CoDirecteur de thèse

Remerciements

Les travaux présentés dans ce mémoire ont été réalisés dans le cadre d'une collaboration entre le Laboratoire d'Analyse et d'Architecture des Systèmes du Centre National de la Recherche Scientifique (LAAS-CNRS), L'institut Aéronautique et Spatial (IAS) et l'Agence Spatiale Algérienne.

Ils ont été effectués au sein du groupe OLC (Outils et Logiciels pour la Communication), devenu l'équipe SARA (Services et Architectures pour les Réseaux Avancés) du LAAS-CNRS. Je tiens donc tout d'abord à exprimer mes remerciements aux directeurs successifs du LAAS-CNRS, Messieurs Raja Chatila, le regretté Jean-Louis Sanchez Paix soit sur son âme et Jean Arlat ainsi qu'au responsable du groupe OLC, Mr Francois Vernadat, puis le directeur de l'équipe SARA Mr Khalil Drira pour le cadre de travail qu'ils m'ont offert.

Je tiens ensuite à remercier Mme Francine KRIEF qui a accepté d'assumer la fonction de président de jury de ma thèse. Je remercie aussi les professeurs Laurent FRANCK et Yacine GHAMRI-DOUDANE qui ont accepté de rapporter mon mémoire et m'ont ainsi fait profiter de leurs remarques pertinentes sur mon travail.

Je remercie également Thierry Gayraud et Pascal Berthou, mes directeurs de thèse. Ils m'ont offert l'opportunité de découvrir le monde de la recherche, tout en restant au contact de l'industrie par le biais des différents projets et m'ont témoigné une grande confiance tout au long de mes travaux en me laissant une grande autonomie. Merci à Thierry qui m'a fait profiter de son expérience et de ses critiques avisées pour me permettre de mieux mettre en valeur mes travaux de thèse et merci à Pascal pour m'avoir aidé à résoudre un certain nombre de problème et m'orienter afin de parvenir à des résultats satisfaisants. De plus, le soutien moral et psychologique apporté par mes directeurs de thèse durant mon incapacité et problèmes de santé, ainsi que la motivation qui m'apportait après cette situation délicate ont été la clé de ma réussite.

Je remercie également Mr. Didier Fériol ainsi que tous les employés de l'IAS, qui m'ont accueilli au début de ma formation et m'ont mis sur les rails de la recherche au LAAS.

Merci aussi différents partenaires, lors des projets contractuels, pour les conseils qu'ils m'ont apporté tout au long de la thèse.

Je remercie aussi mon employeur ainsi que tous ses services qui m'ont soutenu tout au long de ma période de thèse afin de parvenir à finaliser mes travaux.

Un grand merci aussi à Frédéric Nivor (Fred haaaahn), le maitre de PLATINE. Travailler dans son bureau a été une expérience très enrichissante tant au niveau professionnel que personnel. Je le remercie notamment de son aide professionnelle et ses conseils pertinents durant et après la période de sa thèse.

Je tiens à remercier Mohamad (Momo), le maitre du contrôle d'admission, pour son ouverture d'esprit et sa capacité à remonter le moral, Lionel (Mr. NS-2) pour les pauses café "scientifiques", Denis (Mr. Simulation interminable) pour ses blagues et son état à booster le moral, Akram pour sa bonne compagnie et Batak (c'est parti....).

Je remercie aussi les membres du groupe OLC et SARA, Ahmed, Johan, Guillaume 1, Guillaume2, Guillaume 3, Pedro, Rasha, Yann, Philippe Haçan, Marwa, Cedric, Jose, Jorge et tous les autres (désolé si j'en ai oublié certains). Ca a été vraiment agréable de travailler, déjeuner, discuter, ..., avec vous. Merci aussi à tous les stagiaires.

Remerciement spécial pour mes amis, frères et sœurs, les titulaires Nadia BELAID pour le soutien mutuel et les discussions constructives, son soutien permanent sur tous les plans, Amina Mekki-Mokhtar et sa bonne humeur quotidienne, Sihem Bouchareb petite sœur, Mahboubi Omar le roi du poulet tandoori, Chikh Mouss Freeman le gouteur par excellence, ainsi que Sabiha Zedek, Hamza boukaabache sans oublier bien sûr le soutien moral et psychologique apporté par ta7t à la crème. Merci pour les Aids et toutes les occasions passés ensemble, loin de nos familles, vous étiez tous ma famille (mais je me tapais la vaisselle).

Merci aussi à mes compagnons de tests réseaux, Thierry, Pakal, Momo, Batak, Fred, Florin et Rado pour les moments de détente partagés pendant la digestion ou le gouter.

Je remercie aussi les membres des différents services techniques et administratifs du LAAS-CNRS, qui m'ont permis de travailler dans d'excellentes conditions.

Je voudrais maintenant remercier ma famille sans qui je n'aurais jamais réussi à arriver jusque là. Merci à mes parents qui ont toujours été là quand j'en avais besoin et qui m'ont offert un cadre de vie de rêve. Merci à mon frère et sœurs, qui sont aussi mes amis les plus chers, avec qui j'ai partagé des moments inoubliables. Et merci à mes grands parents, oncles, tantes, cousins et cousines pour tous les moments que nous avons partagé ensemble. Avoir une famille sympa et soudée a toujours été très important pour moi et j'en suis comblé.

Enfin, je tiens bien sûr à remercier mes amis qui m'ont supporté depuis tant d'années et qui ont toujours été là pour profiter avec moi de la vie en dehors du laboratoire.

Merci aux potes de Toulouse pour toutes ces belles années qu'on a passé ensemble, pour les concerts chaabi à la kasbah, pour les bons week-ends détente.

Enfin, merci à tous mes autres amis éparpillés en Algérie et dans le monde : Ilyes binomi (el ya2ss), faycal, reda, aziz, paypes, nassim, amine, amine, baki, madjid, ace, Wolf, faten, nour, razika, sidi mohamed, Alim2005, Adel B. et à tous que je porte dans mon cœur et que ma plume a omis de porter.

Liste des acronymes

2G/3G/4G	2nd/3rd/4th Generation
3GPP	3rd Generation Partnership Project
8PSK	8 Phase-Shift Keying
16APSK	16-ary Amplitude and Phase-Shift Keying
32APSK	32-ary Amplitude and Phase-Shift Keying
AAL5	ATM Adaptation Layer 5
ACK	Acknowledgment
ADSL	Asymmetric Digital Subscriber Line
AF	Assured Forwarding
ANG	Access Network Gateway
AQM	Active Queue Management
AP	Access Point
AR	Access Router
ARC	Access Resource Controller
ARP	Address Resolution Protocol
ATM	Asynchronous Transfer Mode
BACK	Binding Acknowledgement
BBM	Break-Before-Make
BDP	Bandwidth Delay Product
BE	Best Effort
BS	Base Station
BSC	Base Station Controller
BSM	Broadband Satellite Multimedia
BSS	Base Station Subsystem
BSSID	Basic Service Set Identifier
BTS	Base Transceiver Stations

BU	Binding Update
BWA	Broadband Wireless Access
CAC	Connection Admission Control
CAPWAP	Control and Provisioning of Wireless Access Points
CBQ	Class Based Queuing
CD	Critical Data
CID	Connection Identifier
CDMA	Code Division Multiple Access
CL	Controlled-Load
CMIP	Client Mobile IP
CN	Correspondant Node
CoA	Care-of Address
COPS	Common Open Policy Service
CoT	Care-of Test
CoTi	Care-of Test Init
CQD	Classful Queuing Discipline
CRA	Continuous Rate Assignment
CTCP	Compound TCP
CTP	CAPWAP Tunneling Protocol
DAD	Duplicate Address Detection
DAMA	Demand Assignment Multiple Access
DCCP	Datagram Congestion Control Protocol
DHCP	Dynamic Host Configuration Protocol
DiffServ	Differentiated Services
DNS	Domain Name System
DS	Distribution System
DSCP	DiffServ Code Point

DSL	Digital Subscriber Line
DSMIPv6	Dual Stack Mobile IPv6
DULM	Data Unit Labelling Method
DVB	Digital Video Broadcasting
DVB-RCS	DVB Return Channel for Satellite
DVB-S	DVB for Satellite
ECN	Explicit Congestion Notification
EF	Expedited Forwarding
ER	Edge Router
ESA	European Space Agency
ESP	Encapsulating Security Payload
ESS	Extended Service Set
ESSID	Extended Service Set Identifier
ETP	Enhanced Transport Protocol
ETSI	European Telecommunications Standards Institute
EuQoS	End-to-end Quality of Service support over heterogeneous networks
FA	Foreign Agent
FAI	Fournisseur d'Accès Internet
FBACK	Fast BACK
FBSS	Fast BS Switching
FBU	Fast BU
FCA	Free Capacity Assignment
FEC	Forward Error Correction
FLAN	Flow ANalyser
FLOC	Flow Capturer
FLORE	Flow REplayer
FMIPv6	Mobile IPv6 Fast Handovers

FTP	File Transfer Protocol
FSS	Fixed Satellite Service
GGSN	Gateway GPRS Support Node
GPRS	General Packet Radio System
GMSC	Gateway MSC
GS	Guaranteed Service
GSM	Global System for Mobile
GW	Gateway
HA	Home Agent
HACK	Handover Acknowledge
HHHO	Hybrid Horizontal Hand Over
HHO	Horizontal Hand Over
HI	Handover Initiate
HIP	Host Identity Protocol
HIT	Host Identity Tag
HMIPv6	Hierarchical Mobile IPv6
HNP	Home Network Prefix
HO	Handover
HoA	Home Address
HoT	Home Test
HoTi	Home Test Init
HSDPA	High-Speed Downlink Packet Access
HSPA	High-Speed Packet Access
HSUPA	High-Speed Uplink Packet Access
HTB	Hierarchical Token Bucket
HTTP	Hyper Text Transfer Protocol
IAPP	Inter Access Point Protocol

ICMP	Internet Control Message Protocol
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
IMS	IP Multimedia Subsystem
IntServ	Integrated Services
IP	Internet Protocol
IPv4	IP version 4
IPv6	IP version 6
IPsec	IP security protocol
ISO	International Organization for Standardization
IST	Information Society Technology
ITU	International Telecommunications Union
JTG	Jugi's Traffic Generator
LAN	Local Area Network
LEDBAT	Low Extra Delay Background Transport
LBU	Local Binding Update
LCoA	On-Link CoA
LMA	Local Mobility Anchor
LWAPP	LightWeight Access Point Protocol
MAC	Medium Access Control
MAG	Mobile Access Gateway
MAP	Mobility Anchor Point
MBB	Make-Before-Break
MBWA	Mobile Broadband Wireless Access (Mobile-Fi)
MDHO	Macro Diversity HandOver
MF-TDMA	Multi-Frequency Time Division Multiple Access
MGEN	Multi-GENerator

MIH	Media Independent Handover
MIPv4/MIPv6	Mobile IP Version 4, Mobile IP Version 6
MN	Mobile Node
MOS	Mean Opinion Score
MPE	Multi Protocol Encapsulation
MPEG	Motion Picture Expert Group
MPLS	Multi-Protocol Label Switching
MPTCP	Multipath TCP
MS	Mobile Station
mSCTP	mobile SCTP
MSR	Mobile Support Router
MSS	Maximum Segment Size
MTR	Media Type Repository
MTU	Maximum Transmission Unit
NAR	New Access Router
NAT	Network Address Translation
NCC	Network Control Centre
NCoA	Next CoA
NGN	Next Generation Network
NMC	Network Management Center
NRT	Non Real Time
NSIS	Next Steps in Signaling
NSLP	NSIS Signaling Layer Protocol
NTP	Network Time Protocol
OBP	On-Board-Processing
ORENETA	One way delay REal-time NETwork Analyser
OSI	Open System Interconnection

OWD	One Way Delay
PAR	Previous Access Router
PBA	Proxy Binding Acknowledgement
PBU	Proxy Binding Update
PCoA	Previous CoA
PDP	Policy Decision Point
PEP	Policy Enforcement Point/Performance Enhancing Proxy
PHB	Per-Hop Behavior
PID	Proportionnel Intégral Dérivé
PKM	Privacy Key Management
PMIP	Proxy Mobile IP
PMP	Point-to-Multipoint
PQ	Priority Queuing
PVC	Permanent Virtual Circuit
QoS	Qualité de Service
QoS	Quality of Service
QPSK	Quadrature Phase-Shift Keying
RA	Router Advertisement
RADVD	Router ADvertisement Daemon
RAP	Resource Allocation Protocol
RBDC	Rate-Based Dynamic Allocation
RCoA	Regional CoA
RCS	Return Channel for Satellite
RCST	Return Channel Satellite Terminal
RESV	RESerVe
RE-ECN	Re-feedback ECN
RED	Random Early Detection

RFC	Request for Comments
RO	Route Optimization
RRT	Return Routability Test
RSVP	Resource reSerVation Protocol
RT	Real Time
RTP	Real Time Protocol
RTO	Retransmission TimeOut
RTT	Round Trip Time
SAP	Service Access Point
SAR	Segmentation and Reassembly
SATIP6	Satellite Broadband Multimedia System for IPv6
SCTP	Stream Control Transmission Protocol
SCPS-TP	Space Communication Protocol Specification – Transport Protocol.
SDP	Session Description Protocol
SMTP	Simple Mail Transfert Protocol
SE	Satellite Emulator
SIGMA	Seamless IP diversity based Generalized Mobility Architecture
SIP	Session Initiation Protocol
SLA	Service Level Agreement
SLAPP	Secure Light Access Point Protocol
SLS	Service Level Specification
SNMP	Simple Network Management Protocol
SP	Service Provider
SS	Subscriber Station/Security Sublayer
SST	Structured Stream Transport
ST	Satellite Terminal
TBTP	Terminal Burst Time Plan

TC	Traffic Control
TCP	Transmission Control Protocol
TFTP	Trivial File Transfert Protocol
TFRC	TCP Friendly Rate Control
TISPAN Networking	Telecommunications and Internet converged Services and Protocols for Advanced Networking
Tng	Transport next- generation
TOS	Type of Service
TS	Transport Stream
UA	User Agent
UAC	User Agent Client
UAS	User Agent Server
UDP	User Datagram Protocol
ULE	Unidirectionnal Lightweight Encapsulation
UMTS	Universal Mobile Telephone Service
UNA	Unsolicited Neighbor Advertisement
URI	Uniform Resource Identifier
UT	User Terminal
VBDC	Volume Based Dynamic Capacity
VHO	Vertical Hand Over
VoIP	Voice over IP
WFQ	Weighted Fair Queuing
WiCoP	Wireless LAN Control Protocol
WiMAX	Worldwide Interoperability for Microwave Access
WLAN	Wireless Local Area Network
WMAN	Wireless Metropolitan Area Network
WWAN	Wireless Wide Area Network
XCP	Explicit Control Protocol

XML

eXtensible Markup Language

Liste des figures

Figure 1 : Système satellite DVB-S2/RCS [4]	8
Figure 2 : cycle requête/allocation DAMA.....	11
Figure 3 : Station 802.11e [11].....	14
Figure 4 : Topologies réseaux 802.16 [12]	15
Figure 5 : Architecture de services de télémédecine [18].....	21
Figure 6 : Application des prescriptions de QoS vues par l'utilisateur [15]	22
Figure 7 : Domaine DiffServ [12]	25
Figure 8 : Les facteurs contribuant à la QoS [32]	29
Figure 9 : Composantes métadonnées [32].....	34
Figure 10 : Architecture d'intégration fortement couplée LTE [36].....	37
Figure 11 : Pile protocolaire LTE (Plan utilisateur issu des documents standards 3GPP) [36].....	38
Figure 12 : Architecture Relai LTE [36]	39
Figure 13 : architecture faiblement couplée LTE/Satellite [36].....	40
Figure 14 : La mobilité dans la pile de protocole [36]	42
Figure 15 : Correspondance entre les composantes NS et les couches réseau [44]	50
Figure 16 : Architecture PLATINE : les blocs nécessaires [47].....	53
Figure 17 : Différent élément de Platine [4]	54
Figure 18 : Interface de gestion SATEM	55
Figure 19 : façonnage du trafic en utilisant TC avec les classes DiffServ [36]	56
Figure 20 : Handover sur des réseaux hétérogènes en utilisant TC et Iptables [36].....	57
Figure 21 : système d'émulation au niveau paquet [36]	58
Figure 22 : Architecture de la plateforme de test OURSES [18]	59
Figure 23 : Couverture mondiale des satellites Inmarsat I4 [57].....	62
Figure 24 : Architecture du système aéronautique d'E-cab [58]	62
Figure 25 : Plateforme de test Inmarsat.....	64
Figure 26 : Interaction entre JTG, Gnuplot, FL3 et ORENETA	67
Figure 27 : scénario ORENETA [68]	68
Figure 28 : Schéma descriptif de la plateforme de test	70
Figure 29 : (a) Délai des paquets et (b) débit	71
Figure 30 : Evolution du rapport signal/bruit	71
Figure 31 : SNR et MODCOD Vs. Temps avec bruit.....	72
Figure 32 : Délai du flux 1 (gauche) et 2 (droite).....	74
Figure 33: Débit de deux flux (parmi les 11 lancés).....	75
Figure 34 : Le délai des paquets sur la voie aller (gauche) et retour (droite)	75
Figure 35 : Débit et délai sur la voie aller	76
Figure 36 : Débit et délai sur la voie retour	76
Figure 37 : Débit d'un flux lperf.....	77
Figure 38 : Evolution de la taille de la fenêtre de congestion.....	77
Figure 39 : Evolution du délai moyen	78
Figure 40 : Evolution du RTO (gauche) et pertes ponctuelles (droite).....	79
Figure 41 : Débit (à gauche) et pertes ponctuelles (à droite).....	80
Figure 42 : SNR et MODCOD en fonction de l'évolution du temps	80
Figure 43 : Délai aller/retour d'un Ping sur un lien satellite réel- Plateforme OURSES	86
Figure 44 : Plateforme de test utilisant SATEM.....	94
Figure 45 : Numéros de séquence, débit et RTT sur un lien 512 kbit/s.....	98
Figure 46 : numéros de séquence et occupation de la bande passante sur un lien à 2 Mbit/s	99
Figure 47 : Taille moyenne des pages web les 15 dernières années [105]	102
Figure 48 : Tests HTTP/TCP : Comparaison des débits applicatifs (512 kbit/s & 2 Mbit/s)	103

Figure 49 : Tests HTTP/TCP : comparaison de durée de téléchargement (Sat @ 2 Mbit/s)	104
Figure 50 : Tests HTTP/TCP : capacité à atteindre de bons débits	104
Figure 51 : Tests HTTP/TCP : Occupation de la bande passante (Sat @ 2 Mbit/s).....	105
Figure 52 : Durée d'envoi de 20 fois 10, 20, 50 et 100 ko.....	105
Figure 53 : Durée d'envoi de 20 fois 100, 200, 500 et 1000 Ko	106
Figure 54 : Comparaison des numéros de séquence (a) et sans New Reno (b) d'un transfert FTP sur un lien 512 kbit/s	107
Figure 55 : Comparaison des débits (a) et sans New Reno (b) d'un transfert FTP sur un lien 512 kbit/s	108
Figure 56 : Comparaison des RTTs (a) et sans New Reno (b) d'un transfert FTP sur un lien 512 kbit/s	108
Figure 57 : Tests avec TCP New Reno tests (client/server hétérogènes)	109
Figure 58 : Tests avec CTCP tests (client/server hétérogènes)	109
Figure 59 : Tests avec TCP Hybla tests (client/server hétérogènes)	110
Figure 60 : Tests avec TCP Cubic tests (client/server hétérogènes).....	110
Figure 61 : Evolution des numéros de séquence sur un lien satellite, impact des PEPs.....	112
Figure 62 : Résultats avec la première configuration ECN (relative au Tableau 10)	115
Figure 63 : Débit TFRC en condition nominale	118
Figure 64 : Occupation du media Vs. débit d'envoi autorisé	118
Figure 65 : Description de la solution	121
Figure 66 : Débit d'émission TFRC sans ACM.....	122
Figure 67 : Débit d'émission TFRC and évolution des numéros de séquence avec ACM	122
Figure 68 : Débit d'émission TFRC and évolution des numéros de séquence avec l'anticipation ACM	123
Figure 69 : Diagramme de séquence pour l'envoi en mode CCM.....	123
Figure 70 : Diagramme de séquence pour l'envoi en mode ACM sans (à gauche) et avec (à droite) anticipation	124
Figure 71 : Evolution des numéros de séquence, débit et RTT Durant 2 handover de délai.....	134
Figure 72 : Evolution de la Cwnd Durant un handover de délai	135
Figure 73 : Evolution des numéros de séquence, débits et RTT Durant un handover de bande passante	138
Figure 74 : Evolution des numéros de séquence, débits et RTT Durant un handover de la passante/délai	139
Figure 75 : Evolution des numéros de séquence durant un handover de la bande passante/délai et ECN configuré (Set 1 et 2)	141
Figure 76 : Evolution des RTTs durant un handover de la passante/délai et ECN configuré (Set 1 et 2).....	141
Figure 77 : Evolution des numéros de séquence avec une coupure de (a) 500 ms (b) 1000 ms.....	142
Figure 78 : Evolution des numéros de séquence avec une coupure de (a) 500 ms (b) 1000 ms.....	143
Figure 79 : Evolution des numéros de séquence pour 12 flux CTCP	144
Figure 80 : Evolution des numéros de séquence de la moyenne de 12 flux de chaque version de TCP.....	144

Table des matières

Introduction générale et problématique

Chapitre 1 : Introduction aux réseaux hybrides et mécanismes de mobilité

1	INTRODUCTION.....	7
2	LES RESEAUX SATELLITE ET MOBILE	7
2.1	Les réseaux satellite.....	8
2.2	Les réseaux WiFi 802.11.....	12
2.3	Les réseaux WiMax 802.16	14
2.4	Les réseaux hybrides satellite/terrestre.....	18
3	LA QUALITE DE SERVICE ET LA QUALITE D'EXPERIENCE	18
3.1	Introduction.....	18
3.2	La QoS dans la couche application	19
3.3	Les modèles de garantie de la QoS	23
3.4	La Qualité d'Expérience	28
4	LA MOBILITE ET SA GESTION	36
4.1	Les scénarios hybrides de mobilité	37
4.2	Schéma de mobilité	40
4.3	Conclusion de mobilité	43
5	CONCLUSION	44
1	INTRODUCTION.....	47
2	METHODES D'EVALUATION	48
2.1	La simulation	48
2.2	L'émulation.....	51
2.3	Système réel	58

2.4	Conclusion sur les plateformes d'évaluation	64
3	OUTILS DE MESURE POUR LES SYSTEMES EMULES OU REELS	65
3.1	Iperf.....	65
3.2	TCPTRACE	66
3.3	Wireshark	66
3.4	NTP Daemon.....	66
3.5	JTG.....	66
3.6	FL3	67
3.7	ORENETA	68
3.8	WEB100	69
4	EXEMPLE D'EVALUATION : LE PROJET OURSES	70
4.1	Tests basiques de la QoS.....	70
4.2	Tests de TCP.....	77
4.3	Les tests subjectifs	81
4.4	Limitation et problèmes rencontrés.....	82
4.5	Conclusion	82
5	CONCLUSION	83
1	INTRODUCTION.....	85
1.1	Caractéristiques des réseaux satellite GEOs.....	85
2	L'APPORT DES NOUVELLES VERSIONS TCP	88
2.1	L'évolution de TCP	88
2.2	Les options et mécanismes d'améliorations de TCP	90
3	EVALUATION DE L'IMPACT DES PROTOCOLES TCP SUR UN RESEAU SATELLITE.....	94
3.1	Plateforme de tests	94
3.2	Méthodologie de mesure/analyse	95

3.3	Résultats et analyses	95
3.4	Conclusion	116
4	PROPOSITION SUR L'AMELIORATION DU CONTROLE DE CONGESTION SUR LE DVB-S2 : LE PROTOCOLE DCCP.....	116
4.1	The TCP Friendly Rate Control	117
4.2	L'impact des réseaux satellite sur DCCP avec les flux à débit variable.....	118
4.3	L'apport de l'ACM et problématique.....	119
4.4	Solutions proposées	120
4.5	Conclusion	124
5	CONCLUSION	125
1	GESTION DE LA MOBILITE AVEC TCP DANS UN RESEAU HYBRIDE.	128
2	EVALUATION DE L'IMPACT DES PROTOCOLES TCP DANS UN CONTEXTE HYBRIDE.....	130
2.1	Comportement de TCP durant un handover	130
2.2	Résultats et analyses	132
2.3	Conclusion	144
3	CONCLUSION	145

Introduction générale

Le besoin croissant de communication et d'interconnexion entre utilisateurs entraîne l'introduction de nouvelles architectures, protocoles et technologies réseau dans le monde Internet. Les infrastructures réseau filaires ont montré jusque-là leur suprématie dans les zones urbaines. On peut citer, entre autre, la technologie DSL ou encore la fibre optique. Cependant, leurs déploiements dans les zones géographiquement reculées et faiblement peuplées seraient très coûteux. Ceci nous amène à opter pour des solutions alternatives. Nous pouvons en citer deux grandes catégories.

Nous avons tout d'abord les technologies GSM, GPRS, UMTS, 3G+ et la 4G, appelées généralement les réseaux cellulaires de télécommunications mobiles. Elles se caractérisent par leur large couverture et par le fait que tout appareil téléphonique mobile peut avoir un accès via ces réseaux. De plus, le débit agrégé qu'elles offrent est assez conséquent et peut atteindre quelques mégaoctets par seconde dans certaines zones.

La deuxième solution alternative est la technologie des réseaux par satellite. Utilisée à ses débuts pour la diffusion de la télévision, cette technologie a connu de grandes évolutions et permet actuellement un accès bidirectionnel aux services IP. De plus, pour un satellite géostationnaire, la couverture offerte peut atteindre le tiers du globe. Les grands inconvénients de cette technologie d'accès sont principalement le délai de propagation assez long ainsi que la limitation et le coût de la bande passante. Néanmoins, depuis quelques années, la technologie d'accès par satellite suscite l'intérêt des fournisseurs d'accès Internet du fait d'un coût de déploiement fixe par rapport aux autres technologies.

Pour garantir à la fois la satisfaction des clients et l'optimisation des réseaux d'accès par satellite, diverses architectures à Qualité de Service (QoS) ont vu le jour. Ces architectures gèrent la Qualité de Service à cause de la limitation et du coût des ressources afin de garantir un accès adéquat pour chaque profil utilisateur (militaire, accès en zones rurales, catastrophes naturelles...etc.) et pour chaque type de profil de données (HTTP, FTP, visioconférence, VoIP...etc.)

Le contexte actuel d'accès aux réseaux satellite est souvent associé aux réseaux de nouvelle génération (NGN). Ainsi la technologie des réseaux satellite pourrait être positionnée comme une technologie d'accès aux futures architectures réseaux intégrées ou hybrides (satellite/terrestre), en support du multiservice (type 3G+, 4G), avec ce que cela entraîne en termes d'exigences liées aux nouveaux usages. En particulier, l'accès généralisé et transparent aux services, en mobilité, partout, tout le temps et sur tous les supports. Les termes « ubiquité » ou « omniprésence » peuvent donc être utilisés pour définir ce type d'accès, traversant un ensemble de réseaux avec différentes caractéristiques (WiFi, WiMax, 3G, filaire) et en effectuant un changement de l'extrémité d'accès d'une manière transparente.

Par sa capacité de large couverture (accès usagers) ou encore par son efficacité en diffusion (point-multipoint), le satellite peut se montrer très attractif en complément des solutions cellulaires terrestres. Mais, à l'inverse, il s'accompagne de certaines contraintes physiques qui le rendent très

spécifique et assez peu transparent vis-à-vis des architectures réseaux de nouvelle génération centrées sur un cœur technologique IP. En effet, les caractéristiques techniques d'un réseau satellite (un long délai, des ressources limitées, un taux de pertes significatif...etc.) pénalisent l'ensemble du réseau, vu l'interconnexion entre les différentes portions (le réseau satellite et les réseaux terrestres filaires et non filaires).

Face aux déploiements rapides des réseaux satellite, plusieurs problèmes font surface tels que les limitations d'accès en termes de Qualité de Service ainsi que la garantie d'une mobilité accessible, performante et rapide et dans un réseau NGN.

Dans cette thèse, nous nous intéresserons à la problématique de l'accès à travers les réseaux hybrides satellite/terrestre et nous proposerons des solutions pour améliorer le comportement protocolaire et applicatif ainsi qu'une garantie de la Qualité de Service supportée par les fournisseurs et perçue par les utilisateurs.

Problématique

Depuis le positionnement des réseaux satellite comme solutions alternatives/complémentaires aux réseaux terrestres et après la démocratisation des terminaux DVB-S, puis la standardisation de la voie retour DVB-RCS, les efforts de recherche liés à cette technologie connaissent une intensification afin d'établir des architectures à QoS [1] pour minimiser les coûts d'accès tout en offrant un service correct pour le client. Cependant, l'évaluation et le paramétrage de ces architectures, en terme de Qualité de Service, devient plus que nécessaire, mais aussi très complexe.

Jusqu'à-là, il n'existe aucun processus permettant à la fois d'évaluer la capacité des architectures satellite à fournir un type de service et de les paramétrer pour supporter divers profils utilisateurs. Les tests actuels consistent à utiliser des moyens physiques d'appareillage, tels que des émulateurs physiques et des équipements de diagnostic. Ils ne considèrent que les couches basses (physique et MAC) et ne sont donc pas une référence pour des utilisations réelles. Cette méthode d'évaluation s'avère coûteuse, complexe et nécessite de gros moyens matériels. De plus, d'autres paramètres rentrent en considération dans les couches supérieures, (IP et application) et influencent le comportement applicatif, cela peut se traduire par une sous-évaluation de ces architectures.

D'un autre côté, les paramètres du terminal satellite sont, nombreux et vastes. Ceci rend le système très complexe à paramétrer et difficile à mettre en œuvre. Il faut aussi signaler qu'un test réalisé au niveau physique et qui s'est avéré satisfaisant, n'implique pas automatiquement une satisfaction de l'utilisateur, car beaucoup d'autres paramètres peuvent influencer et perturber le système satellite dans une utilisation quotidienne. Pour toutes ces raisons, les évaluations actuelles des systèmes satellites se révèlent complexes et insuffisantes. Dans ce cas, il sera plus judicieux de procéder à une approche multicouche pour l'évaluation de ce genre de système.

L'évaluation classique de la QoS dans les réseaux satellite permet de fournir des informations sur les couches basses, ces systèmes intègrent actuellement le monde Internet et implique la nécessité de la

QoS applicative. De plus, les architectures de QoS concernent toutes les couches et exigent un paramétrage complexe surtout avec l'arrivée du DVB-S2 (ACM).

Les problèmes principaux résident dans la complexité de la configuration de ces systèmes, mais aussi dans l'inexistence d'une approche pour l'évaluation applicative d'une architecture de communication par satellite.

Les tests en conditions réelles représentent la meilleure manière d'évaluer les réseaux satellite. Bien que les contraintes de temps et de disponibilité aient pénalisé considérablement le bon déroulement de tests, les résultats ont permis de mettre l'accent sur la difficulté de TCP dans les réseaux satellite et spécialement la gestion du contrôle de congestion.

La deuxième problématique concerne la couche transport dans les réseaux satellite ainsi que l'impact de ces derniers spécialement sur le protocole TCP. Nous nous intéressons particulièrement au protocole TCP - qui est le protocole de niveau transport le plus répandu actuellement- à cause du mode de transmission qu'il utilise et qui peut s'avérer "lourd" pour une transmission satellite (la fiabilité de TCP et le délai de propagation).

TCP a été conçu pour les réseaux terrestres, caractérisés généralement par de faibles délais. Durant son opération sur un réseau satellite, le long délai de propagation peut provoquer à travers les timers TCP la fausse détection de congestion, puisqu'il considère que les paquets sont perdus. De plus, les versions de TCP les plus utilisées sont basées sur les pertes (losses-based), c'est-à-dire que la gestion de la fenêtre de congestion est basée sur la mesure du niveau de pertes. Ainsi, une sous-utilisation de la bande passante peut être expérimentée.

D'un autre côté, ces dernières années, des versions TCP spécifiques ont vu le jour. En effet, ces versions prennent en compte les caractéristiques des réseaux satellite et y sont bénéfiques. Malheureusement dans un contexte de NGN, où les flux traversent un ensemble de réseaux interconnectés, les performances applicatives sur les réseaux sous-jacents se voient affectées.

Des solutions telles que les PEPs (Performance Enhancing proxy) ont été introduites pour remédier à ce problème. Les PEPs consistent essentiellement à isoler le segment satellite pour y déployer des versions adaptées au satellite. Bien que cette solution apporte des améliorations, elle présente un certain nombre d'inconvénients tels que le déploiement et le respect de la sémantique de bout en bout du protocole TCP (sécurité).

Les PEPs posent aussi de problèmes de mobilité. Beaucoup d'autres solutions qui peuvent être "rustines" ou "complexes" ont donné de bons résultats mais leurs limitations ou la complexité de déploiement pénalisent, dans certains cas, les fournisseurs et/ou les utilisateurs. Pour toutes ces raisons, nous nous sommes consacrés à trouver des solutions TCP qui se comportent aussi bien en évitant d'utiliser des solutions tel que les PEPs.

D'un autre point de vue, les réseaux par satellite ne peuvent plus être envisagés comme isolés, et doivent être interconnectés aux réseaux terrestres pour faire partie de l'Internet actuel. La difficulté réside dans la difficulté d'implémenter des solutions mixtes pour le satellite et le terrestre

L'objectif est de trouver des mécanismes implémentables dans TCP, qui fonctionnent sur les réseaux hybrides, et surtout génériques pour s'adapter aux réseaux actuels.

L'un des points forts des NGN est l'accès dans l'espace et dans le temps. L'avantage réside dans une mobilité d'accès lorsque l'un des points d'accès est inaccessible. Lors d'un changement de réseau dans un contexte hybride, deux aspects sont essentiels : la conservation de la connexion des applications et le maintien des performances à un niveau raisonnable. En effet, dans les architectures NGN, les services et les applications sont conçues indépendamment des réseaux sous-jacents (sans fil, filaire, optique, 4G...etc.) basés sur IP, et qui sont le cœur des services de téléphonie et de données. Le paradigme permanent est conçu comme une mobilité généralisée pour les utilisateurs des services, permettant un service continu de commutation pendant tout l'accès. Cela signifie que les applications (et donc les protocoles de transport sous-jacents) doivent être persistants pour le changement de réseau.

Le changement de réseau est souvent lié à une interruption de service, puisque les solutions actuelles se montrent inefficaces au niveau du transport. Nous nous intéressons principalement aux problèmes de mobilités liés à la couche transport et proposons l'utilisation de nouveaux protocoles de transport pour les réseaux hybrides satellite/terrestre.

Un des principaux problèmes liés à la mobilité est le changement des caractéristiques du réseau. En effet, lors d'un handover, la bande passante et le délai seront changés. Par exemple, lors d'une mobilité vers un réseau satellite, la bande passante sera diminuée et le délai sera plus grand. Ceci peut engendrer des problèmes pour certains types de protocoles qui utilisent les anciens états et paramètres.

L'émulateur SATEM est utilisé dans nos travaux. Cet émulateur a été développé et configuré en utilisant les résultats sur le système réel du projet OURSES, décrit ci-dessous. Dans ce manuscrit, les mesures de performances de plusieurs versions TCP sont présentées et discutées afin de répondre aux problématiques d'évaluation de la Qualité de Service dans les réseaux satellite et à l'adaptation de TCP dans les réseaux satellite et hybrides.

Les travaux effectués au cours de notre thèse ont été réalisés principalement en relation avec deux projets contractuels.

Le premier, **OURSES** (Offre d'Usage Rural de Service par Satellite) a pour objectif de mieux positionner les télécommunications par satellite sur le plan de l'efficacité technique et économique lorsque celles-ci sont utilisées en complément des technologies terrestres pour garantir le déploiement et l'usage de télé-services dans des zones ne disposant pas encore des moyens d'accès haut débit. Le projet est conduit par EADS Astrium avec les contributions des laboratoires LAAS-CNRS, IMS (Bordeaux), ENST-GET et TéSA, du CNES et des industriels Thales Alenia Space, Medesat et APX-Synstar.

Le second est un projet RT&CNES qui s'intitule **SATPERF**. Le projet a été conduit par le Centre National des Etudes Spatiales (CNES), EADS Astrium et le LAAS. L'objectif était d'améliorer les performances applicatives et protocolaires dans les réseaux satellite et hybrides –satellite/terrestre). Dans ce projet, l'émulateur SATEM a été utilisé pour effectuer les évaluations de performances.

Organisation du manuscrit

Le plan de notre manuscrit est divisé en quatre chapitres. Le premier chapitre expose un état de l'art sur les réseaux par satellite et les réseaux mobiles, une présentation des mécanismes de gestion de la Qualité de Service et les différentes recommandations proposées dans ce sens. Il se termine par une introduction à la mobilité et aux méthodes de sa gestion. Ce chapitre va permettre de recentrer les problèmes liés à l'évaluation de la Qualité de Service dans les réseaux satellite ainsi qu'à la gestion de la mobilité.

Le deuxième chapitre met en avant les différentes méthodes d'évaluation de la Qualité de Service dans les réseaux par satellite. Le troisième chapitre présente les problèmes liés aux implémentations de TCP dans un réseau satellite. Nous montrerons que les dernières évolutions du protocole TCP changent radicalement l'approche classique du problème. Nous proposons par ailleurs une solution pour améliorer un autre protocole de transport, DCCP, sur la voie aller d'un réseau satellite pour les échanges non fiables. Le dernier chapitre se consacre à l'impact de la couche transport sur les réseaux hybrides satellite/terrestre.

Chapitre 1 : Introduction aux réseaux hybrides et mécanismes de mobilité

1 Introduction

Les réseaux satellite ont vu leur déploiement s'accélérer, en tant que solutions alternatives, à cause de la rapidité et la flexibilité de leurs mises en place. Par exemple, le satellite KA-SAT va permettre de desservir jusqu'à 300.000 foyers en Internet haut débit, avec des débits montants compris entre 6 et 10 Mbits/s. L'ouverture des services commerciaux pour ces communications satellitaires est prévue pour la fin mai 2011. Ce nouveau satellite améliore la Qualité de Service disponible pour les 1,5% de foyers qui ne bénéficient pas du haut débit par ADSL en Europe et le bassin méditerranéen.

Cette mise en place est aujourd'hui envisagée en complément d'autres solutions sans fils, comme le WiMax mais surtout les réseaux cellulaires de nouvelles générations. En effet, la dernière génération des réseaux cellulaires, appelée communément la 4G, permet des accès haut débit, rapides et mobiles, et pourtant la couverture peut faire défaut. C'est la raison pour laquelle les experts se dirigent vers des solutions hybrides satellites/terrestres ou mobiles pour la gestion des accès à la donnée.

Néanmoins, le coût de ces technologies ainsi que la limitation des ressources freinent ce développement. C'est pour cela que des architectures de Qualité de Service (QoS) sont mises en œuvre, pour limiter tout d'abord les ressources mal utilisées et surtout pour garantir la satisfaction du client. En effet, chaque client a son propre profil de données à utiliser et peut donc profiter de services adaptés.

Ainsi, notre contexte de travail adresse la Qualité de Service dans les réseaux hybrides. Dans ce chapitre, nous allons présenter les différentes techniques de réseaux d'accès possibles pour les réseaux hybrides. Par la suite, nous introduisons le concept de Qualité de Service ainsi que la qualité d'expérience. Cette dernière n'est qu'une partie de la Qualité de Service mais qui a pour vocation de mesurer la satisfaction de l'utilisateur. Nous décrivons par la suite les différents protocoles utilisés, ou susceptibles d'être utilisés dans les NGNs. Nous finissons ce chapitre par une introduction à la mobilité et sa gestion.

2 Les réseaux satellite et mobile

La technologie des réseaux satellite connaît de grandes évolutions grâce à sa large utilisation dans plusieurs scénarios, la rapidité de sa mise en place et la large couverture proposée. Aujourd'hui, les réseaux satellite ne sont plus mis en place en tant que solution isolée, mais ils sont vus comme une solution complémentaire aux réseaux mobiles.

Dans le contexte de complémentarité, deux types de technologie suscitent notre intérêt, la technologie WiFi 802.11 et la technologie WiMax 802.16. De plus, le réseau cellulaire de 4ème

génération remplace son prédécesseur 3G et se voit candidat pour être accompagné par les réseaux satellite.

Avant de décrire ces technologies, nous commençons par présenter les réseaux satellite et les méthodes d'accès à ses ressources. Par la suite, nous décrivons les technologies WiFi 802.11 et WiMax 802.16.

2.1 Les réseaux satellite

2.1.1 Présentation

Le Digital Video Broadcasting over satellite (DVB-S) [2] est un standard ETSI (European Telecommunications Standards Institute) conçu en 1994 et utilisé initialement pour la diffusion (télévision par satellite). Il fournit une large couverture et minimise les infrastructures d'installations avec un simple et rapide déploiement des terminaux. La démocratisation des terminaux [2] a permis une réduction des coûts d'équipement et par la suite le développement et la standardisation d'une voie retour appelée Return Channel via Satellite (DVB-RCS) [3].

Les réseaux DVB-S/RCS (par la suite DVB-S2/RCS) ont introduit l'interactivité requise par les accès large bande dans les zones géographiques non couvertes et rurales, et fournissent un accès rapide et simple dans les zones pendant les catastrophes naturelles. Ces évolutions ont entraîné de nouvelles topologies comme l'accès Internet par satellite. La figure ci-dessous résume une topologie DVB-S2/RCS.

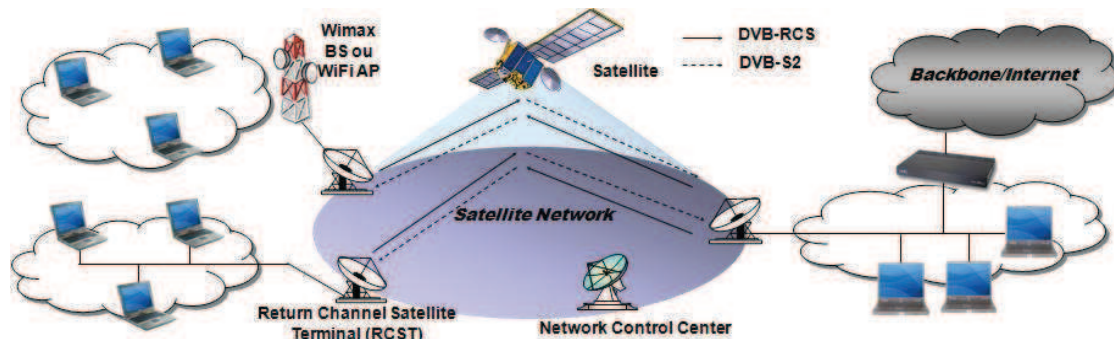


Figure 1 : Système satellite DVB-S2/RCS [4]

Les Return Channel Satellite Terminals (RCST), communément appelés Satellite Terminals (ST), ont un comportement similaire aux routeurs d'accès à la voie retour pour le trafic utilisateur. La Gateway (GW) centralise l'ensemble du trafic dans le réseau satellite et établit l'interconnexion avec les réseaux terrestres. Le Network Control Center (NCC) est le cœur du réseau satellite. Il se charge de la gestion des ressources satellite et généralement effectué au sol en couplage avec la GW.

Depuis la première génération DVB-S, la technologie de transmission via le satellite a considérablement évolué. De nouveaux schémas de codage et de modulation ont été introduits pour permettre plus de flexibilité et une meilleure utilisation du canal de transmission. En 2003, le Digital Video Broadcasting via Satellite 2ème génération DVB-S2 [5] est conçu pour remplacer le DVB-S. Il permet une meilleure utilisation du lien et un gain de capacité de 30 % par rapport à la première génération, grâce aux nouveaux schémas de modulation et à la correction d'erreur (Forward Error

Correction FEC). L'adaptation du codage et de la modulation ACM apporte une utilisation optimisée de la capacité en s'adaptant aux différentes perturbations que le réseau peut subir (perturbations météorologiques).

Nous pouvons distinguer deux types de satellites :

- Les satellites transparents : ce type de satellite permet la propagation du signal sur la voie descendante sans aucun traitement supplémentaire. La conversion du signal DVB-RCS en DVB-S2 s'effectue donc au sol dans la GW, ce qui implique un double bond pour effectuer une transmission ST à ST (ST1→satellite→GW, GW→satellite→ST2).
- Les satellites avec traitement à bord : ce type de satellite possède une charge utile capable de faire un traitement et conversion du signal à bord, donc une capacité de communication de ST à ST en un seul bond (ST1→satellite→ST2).

2.1.2 Méthodes d'accès

La GW représente dans notre cas, le seul point d'accès du réseau satellite à l'Internet. Les mécanismes de gestion de la Qualité de Service et les méthodes d'accès sont différents, selon la voie aller ou retour du système satellite.

Sur la voie aller, la GW centralise toutes les données et la signalisation, et donc occupe l'ensemble de la bande offerte par le transpondeur. La méthode d'accès sur cette voie repose sur un accès multiple à répartition dans le temps et en fréquence MF-TDMA (Multi-Frequency Time Division Multiple Access).

Depuis le DVB-S2, l'ACM apporte de meilleures performances de transmission, une souplesse totale ainsi qu'une complexité raisonnable du récepteur. Il permet une optimisation de la transmission en fonction des conditions de la voie. Différents types de modulations sont utilisés. Les plus répandues sont : QPSK, 8PSK, 16APSK, 32APSK. La correction d'erreur directe FEC (Forward Error Correction) augmente la robustesse des transmissions, généralement une combinaison de modulation et de codage (1/4, 1/3, 2/5, 1/2, 3/5, 2/3, 3/4, 4/5, 5/6, 8/9 et 9/10). Compte tenu des différents codages et modulations utilisés pour chaque lien, la bande passante attribuée à chaque terminal peut donc varier selon le taux d'erreur et donc selon une modulation et un codage donné.

Sur la voie retour, les entrées sont distribuées entre les STs à la demande (à cause des limitations des ressources), selon les besoins des terminaux et la disponibilité de la bande passante. L'algorithme DAMA (Demand Assignment Multiple Access) permet aux STs d'accéder aux ressources selon un modèle de client/serveur. Chaque client DAMA, implémenté dans le terminal satellite, établit une demande des ressources selon le besoin en capacité. Le serveur DAMA implémenté dans le NCC, fournit par la suite les ressources selon les besoins et selon les disponibilités. Les détails du mécanisme de la bande passante à la demande sont présentés dans le paragraphe suivant.

La norme DVB-RCS propose un mécanisme de résolution des contentions entre les différents STs dans le cas d'un accès simultané au lien satellite. Le Multi Frequency Time Division Multiple Access (MF-TDMA) est utilisé dans ce cas et se caractérise par une division fréquentielle correspondant aux fréquences des différentes porteuses. Chacune de ces dernières est divisée en trames appelées dans

ce cas, "supertrame" et par la suite en "timeslots" (unités temporelles de durée fixe). La transmission de données se fait en fonction des timeslots au format MPEG-2 ou ATM.

D'autres techniques existent pour la voie retour, comme le DOC-SIS [6] qui propose un modèle d'accès alternatif par rapport au DVB-RCS.

2.1.3 Mécanisme de BOD (Bandwidth On Demand)

L'allocation dynamique de la bande passante [7] permet d'utiliser les ressources du satellite d'une manière efficace, c'est pour cela que l'ETSI (European Telecommunications Standards Institute) a introduit des techniques d'allocation de la bande passante à la demande BoD (Bandwidth on Demand), dans le standard DVB-RCS. Le Demand Assignment Multiple Access (DAMA) est un protocole en mode client/serveur, qui permet au STs d'avoir régulièrement de la "capacité d'émission" auprès du NCC. Il s'agit de réservations de timeslots pendant lequel ils peuvent transmettre sur la voie retour sans de possibles contentions. Le client DAMA installé au sein du ST calcule les demandes de réservation de capacité selon l'état des files d'émission et sont transmises par des requêtes de capacités (CR) vers le serveur DAMA installé au NCC.

Le DAMA définit quatre types de requêtes pour la satisfaction des besoins des applications :

- Continuous Rate Assignment (CRA) ou requêtes de débit fixe négocié lors de l'initialisation du ST et allouée pour une connexion utilisateur.
- Rate Based Dynamic Capacity (RBDC) ou requêtes de débit variable négociées pour une supertrame.
- Volume Based Dynamic Capacity (VBDC) ou requêtes de transmission d'un volume de données cumulables sur plusieurs supertrames.
- Absolute Volume Based Dynamic Capacity (AVBDC) ou requêtes de transmission d'un volume absolu, qui est identique au VBDC sauf qu'une nouvelle requête annule la précédente.

Dans certains cas, le NCC offre une allocation Free Capacity Assignment (FCA) qui est la capacité non allouée. Cette capacité est redistribuée d'une manière équitable entre les STs lorsqu'elle est disponible.

Deux types de signalisation sont disponibles afin de transporter les requêtes de capacité vers le serveur DAMA.

- Une signalisation dans la bande : les requêtes dans ce cas sont envoyées dans les paquets de données classiques en utilisant des méthodes DULM (Data Unit Labeling Method). Des informations de contrôle et administratives peuvent aussi être envoyées avec cette méthode au NCC.
- Une signalisation hors bande : les requêtes sont envoyées via des timeslots sans contention qui sont spécialement dédiés, en début de chaque supertrame, à une transmission périodique des requêtes.

Une table de signalisation appelée TBTP est émise périodiquement par le NCC sur la voie aller pour qu'il communique aux STs les ressources affectées de la voie retour pour la prochaine supertrame.

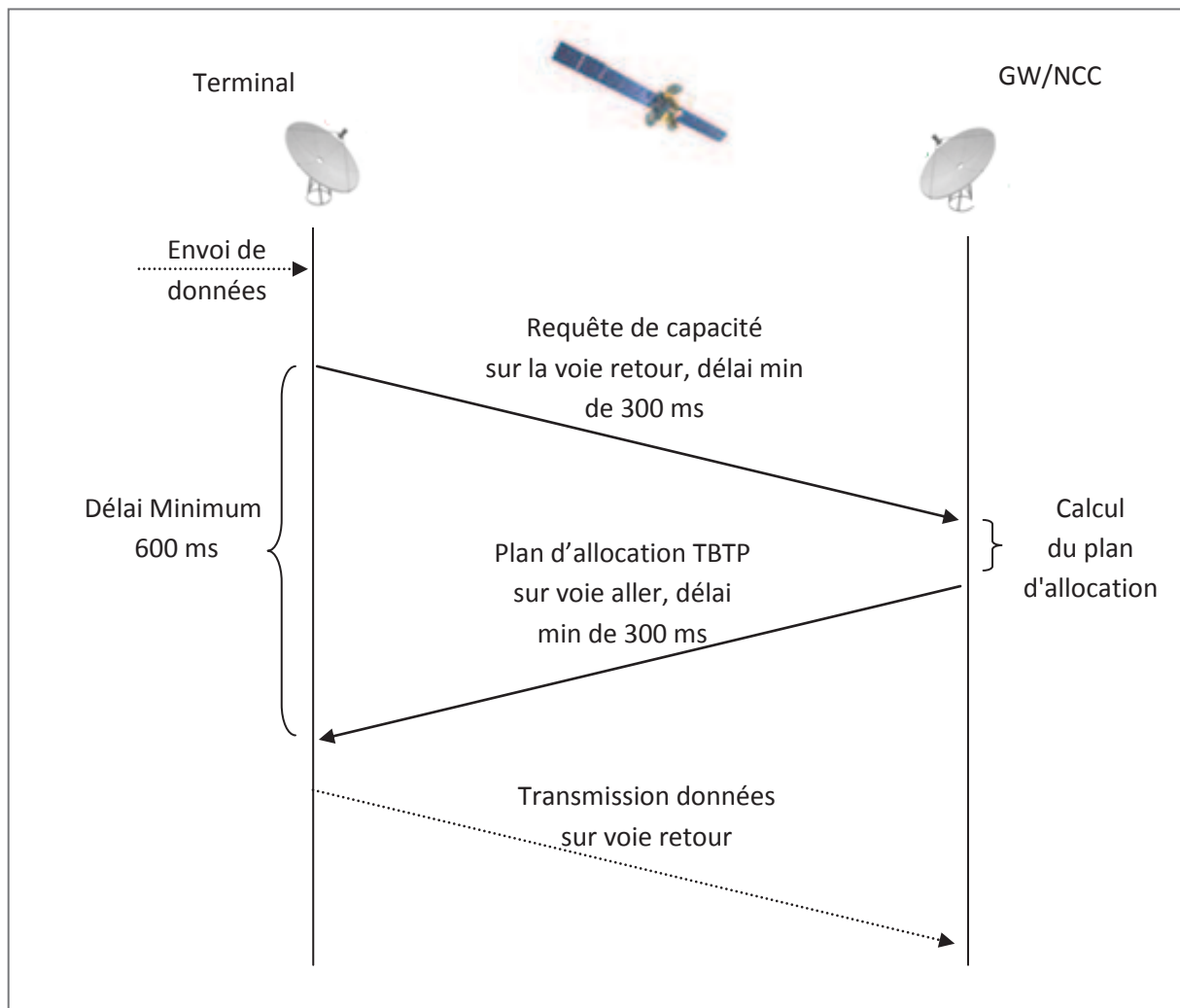


Figure 2 : cycle requête/allocation DAMA

Une durée minimum de 600 ms appelée MSL (Minimum Scheduling Latency) est exigée par chaque allocation dynamique. Cette période comporte le temps d'aller/retour dans le réseau satellite, entre le ST et le NCC, dans l'attente de la réception du plan d'allocation des ressources, ainsi que le temps nécessaire pour le calcul du plan d'allocation TBTP.

2.1.4 Exemples de réseaux satellite

Dans notre étude, nous avons eu l'occasion d'utiliser deux architectures satellites. Une architecture VSAT utilisant le système satellite Inmarsat dans le cadre du projet E-Cab et une autre DVB-S2/RCS utilisant le satellite Astra 1L dans le cadre du projet OURSES. Dans la partie suivante, nous allons décrire brièvement les deux architectures, présenter le cadre dans lequel elles ont été utilisées et voir leurs avantages et leurs inconvénients.

Le projet E-Cab : E-Cab est un projet visant à interconnecter efficacement la cabine d'avion à une liaison satellite VSAT et améliorer les services interactifs d'accès pour les passagers. La réussite de ce projet doit fournir aux aéroports et aux compagnies aériennes un environnement qui offre la possibilité d'offrir un nouveau concept d'accès aux services, en veillant à des niveaux plus élevés de satisfaction pour les passagers et à augmenter l'avantage concurrentiel pour les participants européens. Le réseau satellite utilise des liaisons Inmarsat [8].

INMARSAT (International Maritime Satellite Organization) est une compagnie de télécommunications fondée en 1979 et privatisée en avril 1999. Elle exploite 11 satellites assurant les fonctions de téléphonie et de données, via des terminaux portables ou mobiles qui communiquent avec des stations terrestres à travers ce réseau de satellites géostationnaires. Plusieurs systèmes sont disponibles, tels que le BGAN [9]. Ce dernier a été utilisé dans le cadre du projet E-Cab. Les détails concernant le terminal et la liaison seront présentés dans le chapitre 2.

Le projet OURSES : Ce projet a été initié afin d'optimiser l'utilisation des services de télémédecine via les réseaux satellite dans les zones rurales. L'objectif est de positionner les télécommunications par satellite sur le plan de l'efficacité technique et économique lorsque celles-ci sont utilisées en complément des technologies terrestres pour garantir le déploiement et l'usage de télé-services dans des zones ne disposant pas encore de moyens d'accès haut débit.

Dans le cadre de ce projet, une liaison satellite réelle a été utilisée afin de réaliser les différents tests de validation. La liaison a été fournie par SES [10]. Cette dernière (domiciliée au Luxembourg) possède un satellite multi-faisceau avec des transpondeurs Ka : Astra 1L. Sur la charge utile bande Ka ASTRA 1L, la voie montante est de 2 polarisations x 500 MHz réparties en canaux montants de 40 à 200 MHz par spot, allouable par bloc de 40 MHz. L'activation des canaux montants se fait selon un schéma fourni par SES Astra. Sur la voie descendante, les canaux montants sont concaténés par polarisation dans deux canaux descendants de 500 MHz sur la polarisation opposée.

Les détails sur les terminaux, la gateway et les fonctionnalités utilisées seront détaillés dans le chapitre 2 (Plateforme satellite réelle OURSES).

2.2 Les réseaux WiFi 802.11

Le réseau sans fil Wifi, appelé aussi WLAN (Wireless Local Area Network) est un système de transmission de données conçu par l'IEEE (Institute of Electrical and Electronics Engineers), pour l'assurance d'une liaison indépendante de l'emplacement des périphériques informatiques ; en d'autres termes, c'est la technologie qui permet un accès à une liaison de données sans l'utilisation d'un lien physique filaire (câble). Dans les entreprises, les réseaux WLAN sont implémentés comme le lien final entre le réseau câblé et le groupe d'utilisateurs clients, et qui offre un accès sans fil à l'ensemble des ressources et services du réseau de l'entreprise.

2.2.1 Architecture

Dans ce standard, une STA (Station sans fil) représente l'interface air de l'utilisateur. On peut distinguer deux modes d'utilisations des réseaux WLAN :

- Le mode infrastructure : Dans ce mode, une station de base appelée aussi AP (Access point) joue le rôle de liaison entre le réseau filaire et le réseau sans fil. Avant d'effectuer toute communication, les STAs lancent une procédure d'association avec le AP, puisque ce dernier est l'élément central d'un WLAN.
- Le mode ad hoc : Dans ce mode, les STAs communiquent sans passer par un AP. L'avantage réside dans la rapidité de connexion et dans l'automatisation de configuration et d'organisation. L'infrastructure filaire n'existe donc pas.

2.2.2 Mécanismes d'accès

Procédure d'accès DCF

Ce mécanisme est basé sur CSMA/CA (Carrier Sense Multiple/Collision Avoidance). Le canal est géré d'une manière équitable et est distribué pour les stations.

Les STAs qui veulent émettre, surveillent le canal durant une durée appelée backoff (module utilisé en cas de collision). Ce délai est choisi aléatoirement dans un intervalle temporel $[0, CW]$. Le CW (Contention Window) étant la borne haute. Le décompte backoff est interrompu dès lors que la station détecte une activité sur le canal. La première station libérée du backoff gagne l'accès au canal. Avant d'émettre sur le canal, les STA utilisent un mécanisme d'accès optionnel RTS/CTS (request to send/Clear to send) pour faire face à d'éventuels problèmes de stations dites "cachées". La STA envoie une petite trame RTS au destinataire. Si l'émission réussit sans événement de collision, le récepteur répond par un autre petit message CTS, et donc la station peut émettre sur le canal.

2.2.3 La Qualité de Service dans la technologie 802.11e

La Qualité de Service proposée actuellement dans le standard 802.11 est le "Best Effort". Le nombre d'utilisateurs donc, définit les performances du réseau d'accès.

Pour apporter des améliorations en termes de Qualité de Service, la fonction EDCF (Enhanced Distributed Coordination Function) basée sur le DCF a été donc définie par le groupe IEEE 802.11 et est baptisé **802.11e**.

Un nouveau concept a été introduit dans le nouveau standard 802.11e, l'opportunité de transmission (TXOP). Il est défini par une date de départ et par une durée pendant laquelle une station a le droit d'émettre. Plusieurs trames peuvent être donc transmises par une station dans la même TXOP, tant qu'elles ne dépassent pas le délai approprié.

Procédure d'accès distribué améliorée EDCF

Ce mécanisme propose une différenciation de service, avec quatre catégories d'accès différentes au niveau MAC (Access Category : AC). Cela se représente par des files d'émission implémentées au sein des stations. Chaque catégorie d'accès se distingue en tant qu'entité de transmission MAC indépendante, puisqu'elle a ses propres paramètres tels que la CW. Ainsi, si on fait varier la taille de l'intervalle $[0, CW[i]]$ (i : taille pour chaque file d'accès) afin d'obtenir un délai backoff, on offre un accès au canal différencié entre les STAs, mais aussi au sein des STAs. En affiliant la plus petite valeur d'attente à une catégorie, on considère que cette dernière la plus propriétaire.

La figure ci-dessous représente les files d'émission au sein d'une station. Chaque file supporte une catégorie d'accès. Le trafic qui provient de la couche supérieure doit être marqué avec une priorité adéquate pour la redirection dans la file d'attente à Qualité de Service qui lui correspond.

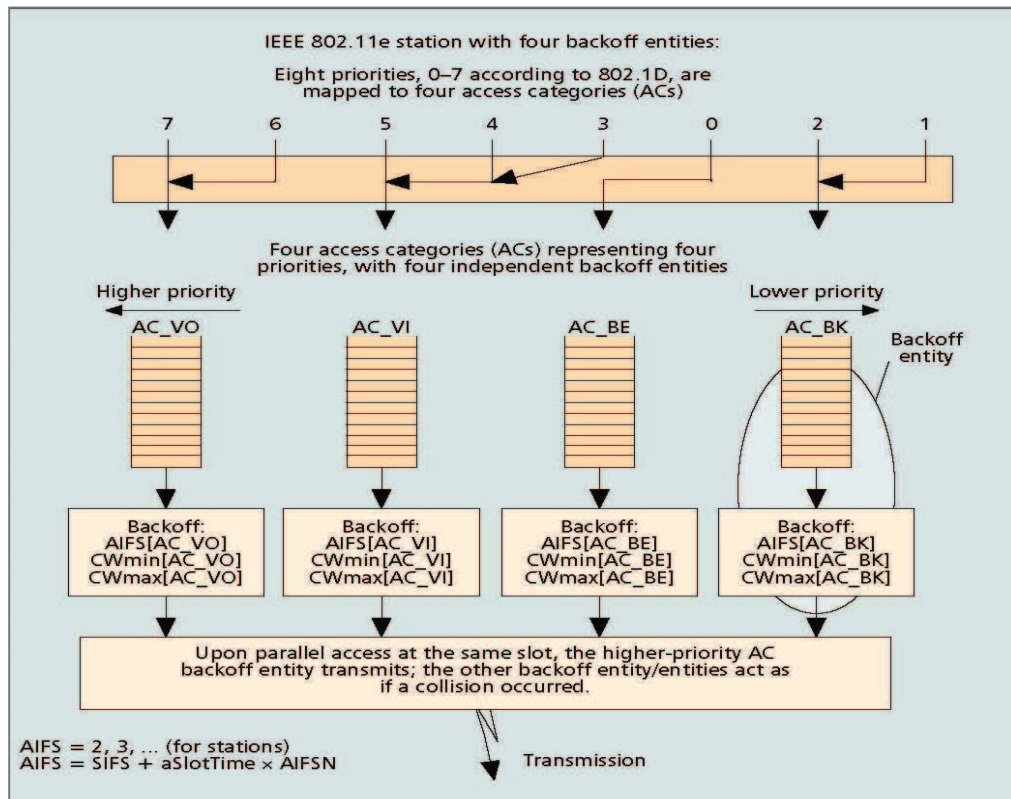


Figure 3 : Station 802.11e [11]

Procédure de structuration contrôlée HCF

L'AP dans ce mode joue le rôle de coordination hybride HC (Hybrid Coordinator), puisque les stations sont interrogées à tour de rôle par l'AP. Le TXOP est requis par les STAs auprès du HC avant l'émission. Le HC, qui est basé sur un mécanisme d'admission de contrôle, peut accepter ou refuser la requête. Dans le cas d'une acceptation, le HC programme et transmet les TXOPs aux STAs.

2.2.4 Conclusion

La technologie 802.11 b/g offre des caractéristiques, avec un débit plafonné à 54 Mbit/s, mais aussi un déploiement facile et une configuration instantanée. Le coût réduit et le gain de mobilité s'ajoutent aux avantages de cette norme. La version améliorée 802.11e apporte une meilleure utilisation à la version standard, avec des fonctions de différenciation de services du trafic, au sein du réseau mais aussi au sein des stations. Elle offre ainsi un support Qualité de Service aux exigences des applications. La nouvelle norme 802.11n permet d'apporter des améliorations par rapport aux versions précédentes en atteignant des débits jusqu'à 300 Mbit/s à courte portée.

2.3 Les réseaux WiMax 802.16

Pour pallier l'absence de réseaux sans fil à haut débit et pour permettre un support de plateforme de boucle locale radio haut débit (Broadband Wireless Access : BWA), l'organisme IEEE a proposé un nouveau standard de transmission radio : 802.16, baptisé WiMax (Worldwide Interoperability for Microwave Access).

Le WiMax peut être considéré comme un complément au standard 802.11, puisqu'il permet d'atteindre un débit théorique de 70 Mbit/s et une portée jusqu'à 50 Km. L'interconnexion entre différents points d'accès WiFi est aussi assurée par ce standard. Ainsi, cela permettra d'accroître la portée et le débit de la boucle locale radio. Avec les différentes versions du WiMax (802.16 a, c, d, e, f et dernièrement m), une large bande de fréquences comprises entre 2 et 66 GHz est assurée par ce standard. La portée et le débit du 802.16 permettent de desservir les zones géographiquement reculées.

La multiplicité des bandes de fréquences, des différents débits exploités, de l'étendue des couvertures et d'applications envisageables ouvre les portes pour sa convergence vers les réseaux mobiles de 4ème génération. Selon différents points de vue, le WiMax est tour à tour, un simple prolongement du WiFi, le cœur de réseau du WiFi, voire encore, la convergence du WiFi et du réseau cellulaire de 4e génération.

Depuis 2009, avec la norme 802.16m, les débits en nomade ou stationnaire peuvent atteindre jusqu'à 1 Gbit/s et 100 Mbit/s en mobile grande vitesse. Cette norme doit rendre possible la convergence des technologies WiMax, WiFi et 4G afin de réaliser des réseaux maillés, et fait place à la technologie MIMO (Multiple-Input Multiple-Output) pour augmenter la bande passante des transmissions.

2.3.1 Architecture

Deux types d'équipement peuvent être distingués : la station utilisateur (SS : Subscriber Station) et la station centrale (BS : Base Station). Cette dernière est chargée des fonctionnalités de gestion et de contrôle de réseau.

Comme la technologie 802.11, le 802.16 possède deux topologies. La première en étoile où les communications sont centralisées par le BS. Et la configuration maillée où les communications peuvent directement être faites entre les SSs. La figure ci-dessous illustre les deux types de topologies.

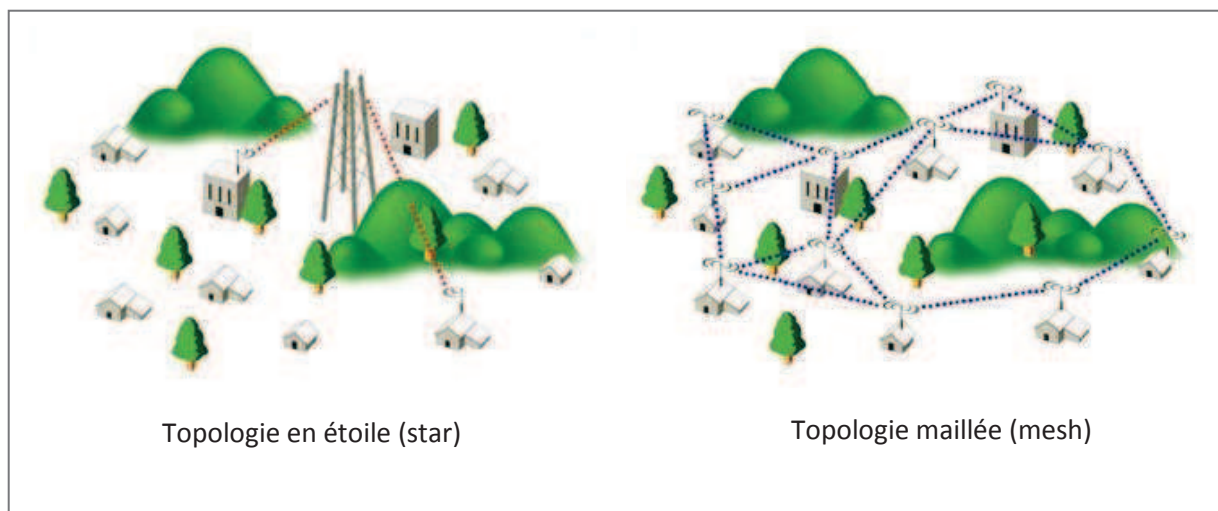


Figure 4 : Topologies réseaux 802.16 [12]

Les deux solutions présentent autant d'avantages que d'inconvénients. La topologie étoile permet d'avoir plus de débit qu'un réseau maillé. Ce dernier permet un déploiement plus rapide et subit moins les facteurs liés à la ligne de vue.

2.3.2 Caractéristiques

Des connexions sont établies entre la station de base et la station utilisateur. Ces connexions permettront la négociation et la mise à jour des paramètres de connexion telles que la bande passante, les paramètres de Qualité de Service choisis, le type de transport...etc. Les paires de connexions (voie montante et descendante) entre le BS et le SS qui sont établies sont :

- Une connexion basique utilisant l'échange de courts messages de gestion avec contraintes temporelles.
- Une deuxième connexion de gestion est utilisée pour le transfert des messages de gestion, et qui sont plus tolérants au délai.
- Une troisième connexion de gestion est créée pour échanger des messages standards DHCP (Dynamic Host Configuration Protocol), SNMP (Simple Network Management Protocol)...etc. qui sont aussi tolérants au délai.

Trois niveaux de Qualité de Service incorporent les trois connexions pour le trafic de gestion MAC entre les BS et SS.

Trois sous-couches distinctes construisent la couche MAC 802.16 :

- La couche de convergence CS (Convergence Sublayer) : La classification des unités de données SDU (Service Data Unit) vers la bonne connexion MAC est assurée par cette sous couche, et donc garantie la QoS et la bande passante allouée. Le transport des cellules ATM et les paquets IP sont aussi pris en charge par cette sous couche.
- La couche commune CPS (Common Part Sublayer) : la tâche de cette sous couche est d'offrir les fonctionnalités de base de contrôle d'accès, de l'allocation de ressources, l'établissement et le maintien de connexion.
- La couche de protection de données SS (Security Sublayer) : Cette sous couche propose des mécanismes d'authentification, d'échange de clés et de cryptage tel que l'AES (Advanced Encryption Standard). Elle se base sur un protocole nommé Privacy Key Management (PKM).

Nous nous intéressons à la sous couche commune plus précisément car elle offre un support de Qualité de Service.

2.3.3 Méthodes d'allocations de ressources

La station de base BS est responsable d'attribuer dynamiquement la bande passante pour la voie montante (SS → BS) et descendante (BS → SS) aux stations utilisateur SS. La voie descendante se base sur le mécanisme d'accès multiple à temps divisé TDMA (Time Division Multiple Access). La voie montante repose sur un accès TDMA et le DAMA. La BS contrôle aussi la taille des trames ainsi que la taille de chaque intervalle de temps. Ce dernier peut varier.

Afin d'avoir un accès au canal, les SSs demandent l'autorisation auprès de la BS. Cette dernière leur fournit des allocations de transmission sur la voie montante. Les allocations sont implémentées selon différentes fonctions :

- **Unsolicited bandwidth grants** : ou allocation de bande non réclamée.
- **Polling** ou allocation de bande par système d'interrogation des SSs : C'est la méthode par laquelle la BS alloue spécifiquement des slots temporels aux SSs. Cela va leur permettre d'effectuer des requêtes de bande passante. L'interrogation de BS aux SSs se fait à tour de rôle selon les besoins de transmission.
- **Contention procedure** ou allocation de bande selon un mécanisme de contention : Dans ce cas-là, des requêtes de demande de bande passante sont transmises par la méthode de contention en utilisant une fenêtre d'intervalle d'attente (backoff window).

Une incrémentation de requêtes de bande passante peut être utilisée, cela veut dire que la BS accumule la bande passante qui a déjà été demandée par la SS. Dans les autres cas, les requêtes sont agrégées, c'est-à-dire que la BS remplace la bande passante demandée par la SS.

Quant à l'attribution de la bande passante, elle s'effectue selon deux modes :

- GPSS (Grant Peer SS) : Elle se fait d'une manière globale pour une SS.
- GPC (Grant Peer Connection) : Elle se fait uniquement pour la dite connexion.

2.3.4 Mécanismes de Qualité de Service dans le 802.16

Le 802.16 propose des mécanismes pour garantir une Qualité de Service différenciée afin de satisfaire les exigences des applications multimédia. On survolera les méthodes d'accès utilisées ainsi que les applications appropriées pour chaque service dans ce qui suit.

- Le service UGS (Unsolicited Grant Service) :
 - *Application* : supporte les flots temps-réels qui génèrent périodiquement des paquets de données de taille fixe, tels que la voix sur IP.
 - *Méthode d'accès* : La SS n'effectue pas de requêtes de bande passante. Elle est négociée à l'initialisation de la station et lui est allouée statiquement tout au long de ce service.
- Le service rtPS (real-time Polling Service) :
 - *Application* : offre des requêtes d'opportunités temps-réel périodiques permettant aux SS de spécifier la taille de leurs données. Ce service génère plus de requêtes que l'UGS mais supporte des demandes de données de tailles variables.
 - *Méthode d'accès* : La SS reçoit les opportunités de transmission périodiques qui lui sont attribuées par polling de la BS. L'utilisation des opportunités de requêtes par contention est interdite pour les flux rtPS.
- Le service nrtPS (non-real-time Polling Service) :
 - *Application* : les flots nrtPS reçoivent le même service que les rtPS.
 - *Méthode d'accès* : est la même que pour le service rtPS. De plus, durant les congestions, les requêtes de contention sont autorisées.
- Le service best effort (BE) :
 - *Application* : le service minimal. Il est utilisé pour les flux ne nécessitant pas de Qualité de Service particulière.
 - *Méthode d'accès* : La SS est obligée d'entrer en mode de contention pour transmettre des requêtes de bande passante.

2.3.5 Conclusion

Malgré les restrictions en termes de réglementations de fréquence, qui ont pu retarder le développement de cette technologie, le standard 802.16 offre un service intéressant, avec une portée jusqu'à 50 Km et une couverture de zones géographiquement reculées. Les mécanismes de QoS avec différenciation de trafic permettent d'offrir le support nécessaire et adéquat aux applications exigeantes.

Le WiMax est vu, de plus en plus, comme un choix possible pour la convergence vers les réseaux mobiles de 4ème génération, surtout avec la norme 802.16m qui permet d'atteindre des débits de d'1 Gbps.

2.4 Les réseaux hybrides satellite/terrestre

Le besoin croissant d'être connecté partout et à n'importe quel instant entraîne le développement des moyens d'accès et l'interaction entre les différents réseaux existants. Dans cette optique, et afin d'améliorer cette intégration, de nombreuses solutions de mobilité et d'interaction entre les différents réseaux ont vu le jour pour palier ce besoin et ont donné naissance aux réseaux hybrides. Nous nous intéressons à l'hybridation entre les réseaux satellite et les réseaux terrestres. Cela représente la cohabitation d'un réseau satellite avec un autre terrestre (filaire ou sans fil), suivant un niveau d'intégration.

Parmi ces solutions, nous citons l'initiative S-WiMax (Intégration d'un réseau satellite avec le WiMax) initié par l'ATC (Ancillary Terrestrial Component).

Aux états unis, l'ATC de la commission fédérale des communications (Federal Communications Commission FCC) a approuvé [13] une technique d'utilisation des réseaux terrestres de téléphone cellulaire pour compléter les services mobiles satellites (Mobile Satellite Services MSS). Cette technique consiste à utiliser les spectres satellites pour les services terrestres intégrés. Cette attention est focalisée sur le développement d'un dispositif (terminal) bi-mode satellite /terrestre pour faciliter l'hybridation satellite/terrestre prévue par l'ATC [14]. Parmi ces réseaux hybrides on peut citer S-WiMax.

Un exemple illustrant cette interaction est présenté dans la partie de la gestion de mobilité (Partie 4 de ce chapitre).

3 La Qualité de Service et la Qualité d'Expérience

3.1 Introduction

La Qualité de Service (QoS) a été introduite pour répondre aux exigences des différents utilisateurs par rapport aux différentes applications utilisées. Au début de l'Internet, les applications n'avaient pas vraiment d'exigences spécifiques en termes d'accès, puisqu'ils s'agissaient de données textuelles non gourmandes. Avec les développements des réseaux, des applications qui y opèrent et les infrastructures d'accès, des exigences sont apparues pour faire passer simultanément de la visioconférence, des transferts de fichiers, des accès au Web et de la messagerie instantanée. Des

garanties de délai, de débit et/ou encore de gigue devaient être fournies pour les utilisateurs afin de garantir une bonne transmission des données et un bon fonctionnement du réseau.

Néanmoins, l'architecture réseau à l'époque n'a pas été conçue pour gérer tout l'ensemble des flux, puisque cela n'a été basé sur aucune méthode de différenciation des services. Le seul modèle utilisé étant le Best Effort, il ne peut garantir un bon fonctionnement de toutes les applications. La seule proposition était de surdimensionner le réseau, mais cela ne faisait que repousser les problèmes puisqu'elle est difficilement applicable sur des réseaux à faible bande passante, spécialement les réseaux sans fil. Par conséquent, il est nécessaire de gérer efficacement les ressources pour garantir aux utilisateurs une QoS adaptée à leurs attentes et besoins.

Dans cette partie, nous allons décrire les mécanismes et les fondamentaux qui permettent une bonne gestion de la QoS, au niveau application et réseau ainsi que la vision de cette dernière par rapport aux réseaux mobiles et le réseau Internet. Nous introduisons par la suite la notion de Qualité d'Expérience qui est liée à la QoS.

3.2 La QoS dans la couche application

L'UIT-T (Union internationale des télécommunications - Secteur de la normalisation des télécommunications) est un organe permanent de l'UIT (Union internationale des télécommunications) chargé de l'étude des questions techniques, d'exploitation et de tarification, et émet à ce sujet des recommandations en vue de la normalisation des télécommunications à l'échelle mondiale [15]. Elle publie des recommandations, par exemple, sur les performances des applications. Ces recommandations sont caractérisées par différents paramètres pour la quantification de la perception de l'utilisateur. Les principaux paramètres sont :

- Le délai de transfert des paquets est décrit comme le délai de bout en bout. C'est la somme des délais de propagation et de transmission, et des délais d'attente dans les files. Ce paramètre agit directement sur la satisfaction de l'utilisateur puisqu'il décrit la période de l'attente de l'utilisateur pour l'établissement du service depuis le moment où il l'a demandé.
- La variation du délai, appelée aussi gigue (ou jitter) représente la différence liée au moment d'arrivée des paquets d'information. La gigue est aussi importante que le délai de bout en bout car elle agit directement sur la perception de l'utilisateur selon l'application.
- La perte d'informations inclut essentiellement les pertes sur les bits ou paquets en cours de transmission mais aussi les erreurs causées par les dégradations introduites par le codage des médias qui améliore l'efficacité de la transmission.

Les applications utilisées au sein d'un réseau par un utilisateur final sont caractérisées par ces paramètres de performances qui sont perçus par l'utilisateur final ; Ces applications sont classées généralement en trois catégories de médias : l'audio, la vidéo et la donnée.

3.2.1 Les applications Audio

Dans cette catégorie, on distingue trois classes principales d'applications

Signaux vocaux de conversation, Voix sur IP

Les applications audio sont celles qui véhiculent de la voix ou de l'audio dans un seul sens de communication. Ces signaux vocaux sont très influencés par le délai de bout en bout unilatéral et peuvent causer deux effets principaux :

Un écho de l'ordre d'une dizaine de millisecondes qui peut créer une dégradation de la qualité de conversation et des mesures d'annulation d'écho (par exemple) doivent être mises au point pour pallier ce problème [16].

Un grand délai (de l'ordre de plusieurs centièmes de seconde) crée un effet décalage dans la conversation, c'est-à-dire, que le délai de réponse de l'une des extrémités commence à être perceptible et gênant [17].

La gigue n'est pas tolérée par l'oreille humaine et donc, des mécanismes de suppression de cette variation de délai doivent être intégrés tel que l'ajout d'un tampon de composition de gigue. Par contre, l'oreille humaine peut tolérer, à un certain degré, les distorsions du signal vocal, c'est-à-dire tolérer les pertes d'informations.

3.2.2 Les applications vidéo

Les applications vidéo sont composées de deux types de médias :

La visiophonie/vidéo interactive

Le terme visiophonie inclut les conversations bilatérales entre deux terminaux avec des flux audio et vidéo véhiculés en même temps. Les mêmes recommandations seront prescrites pour les signaux vidéo et pour les signaux vocaux, cela signifie une tolérance minimale sur les pertes, selon le type de codage de signaux vidéo et sans écho. On ajoute aussi la synchronisation audio/vidéo, c'est-à-dire que le décalage entre les deux signaux doit être limité afin de synchroniser au mieux la voix (signal vocal) avec le signal vidéo (les lèvres). L'œil humain, comme l'oreille, tolère aussi un certain nombre d'erreurs et de pertes, en fonction du codage des signaux vidéo utilisés.

Vidéo unilatérale

Comme l'audio en temps réel, les exigences dans la vidéo unilatérale, à un certain degré, ne sont pas sévères considérant qu'aucun élément conversationnel ne sera effectué. Le tableau 1 montre les objectifs de qualité pour les applications audio et vidéo.

Applications	Symétrie	Débits	Délai unilatéral	Gigue	Taux de perte d'informations
Audio conférence	Bilatéral	4-64 kbit/s	<150 ms préféré <400 ms limite	< 1ms	< 3%
Vidéo conférence	Bilatéral	16-384 kbit/s	<150 ms préféré <400 ms limite	< 1ms	< 1%
Messagerie vocale	Unilatéral	4-32 kbit/s	< 1s (reproduction) <2s (enregistrement)	< 1ms	< 3%
Streaming audio	Unilatéral	16-128 kbit/s	< 10 s	<< 1ms	< 1%

Streaming Vidéo	Unilatéral	16-384 kbit/s	< 10 s	< 1ms	< 1%
-----------------	------------	---------------	--------	-------	------

Tableau 1 : les objectifs de qualité pour les applications audio et vidéo [15]

3.2.3 Application E-Santé

Nous avons eu l'occasion d'utiliser une plateforme réelle satellite DVB-S2/RCS, lors des tests de projet OURSES [18]. En dehors de l'aspect réseau satellite, des applications d'E-santé ont été utilisées, ainsi qu'un terminal portable de télé-médecine. L'objectif était de faire ressortir l'intérêt des satellites pour la télésurveillance médicale d'un patient dans une zone blanche. La Figure 5 présente les interactions entre les différents éléments de l'architecture de télésurveillance. Cette dernière contient le système de capteurs biomédicaux BodyLan porté par le patient, le terminal portable de télé-médecine, et le segment satellite. Les capteurs corporels du BodyLan envoient les résultats au terminal par WiFi ou Zigbee [19]. Ces résultats seront analysés par le terminal puis transmis, via une liaison satellite, à un médecin dans un centre de télé-médecine. Cette architecture a été validée lors du projet sur des vrais patients au sein d'une maison de retraite, qui portaient des capteurs de signaux vitaux. Le système de capteurs doit pouvoir aider au diagnostic et détecter de potentielles anomalies dans les signaux reçus.

Les données échangées entre le terminal portable et le médecin au niveau du centre de télé-médecine sont principalement des pages WEB ou des petits fichiers PDF via un transfert FTP contenant les différentes valeurs des signes vitaux du malade. Donc, l'échange de données se fait d'une manière fiable (transmission TCP). Une petite latence est tolérée mais les pertes doivent être nulles, comme le préconise l'échange des fichiers FTP dans la Figure 6. Une session de visioconférence peut être établie entre le médecin et l'infirmier(e); dans ce cas, les recommandations sur cette session sont similaires à celles de la visiophonie.

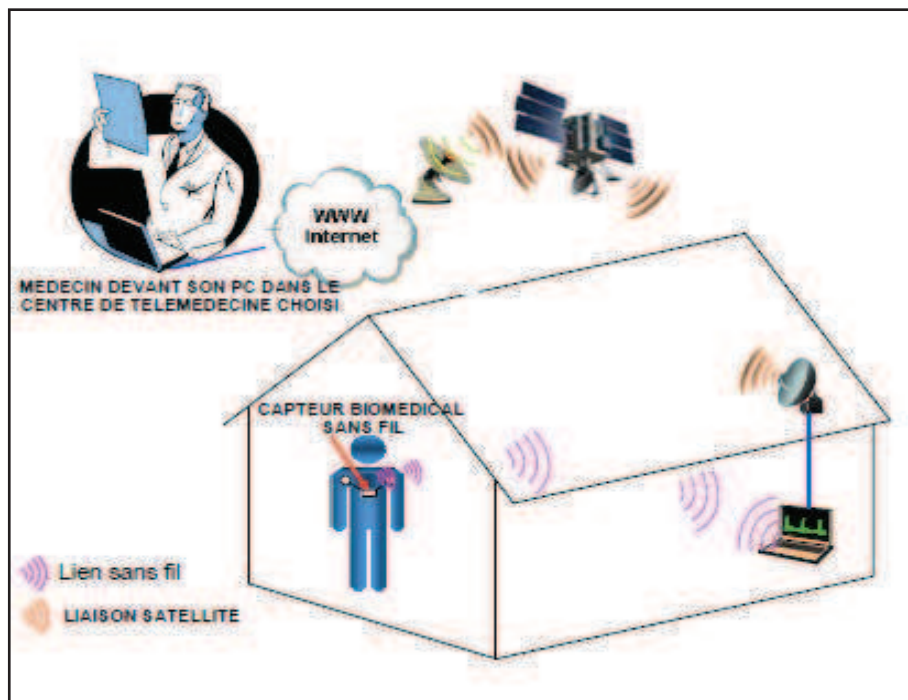


Figure 5 : Architecture de services de télé-médecine [18]

3.2.4 Classification des exigences de performances des applications en classe de QoS

La figure ci-dessous [15], explique les prescriptions de performances selon le délai unilatéral prescrit en abscisse et les pertes tolérées en ordonnée. La forme générale des cartouches donne une idée générale sur les limites.

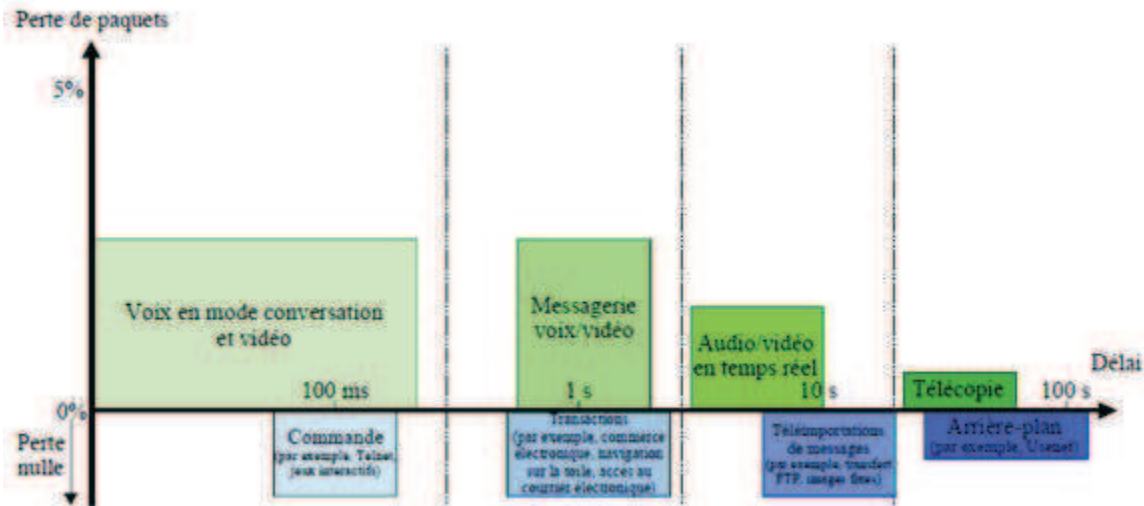


Figure 6 : Application des prescriptions de QoS vues par l'utilisateur [15]

Dans ce modèle, les applications, auparavant identifiées, peuvent être assimilées à des catégories de QoS pour un utilisateur final. Cependant, ce modèle mérite quelques éclaircissements [15]:

- La validité de ce modèle ne dépend d'aucune technique particulière, puisqu'il est basé sur la perception des dégradations de l'utilisateur. Cela implique que le modèle peut s'appliquer à n'importe quelle technique de transport sous-jacente (IP, ATM, ligne filaire, liaison hertzienne, etc.)
- Le modèle permet de savoir si un support de transport convient pour acheminer la donnée d'une application, c'est-à-dire que si le canal présente un retard unilatéral, on peut le considérer comme « non compatible » pour transporter un signal vocal, par exemple. Même si ce retard est minime (~100 ms), il peut perturber le bon fonctionnement de certaines applications vulnérables (Telnet par exemple).
- Les groupements d'applications sont délimités, en limite supérieure et inférieure, en termes de délai et de pertes, et cela permettra à l'utilisateur final de définir un service comme étant acceptable ou pas. Si la limite supérieure est dépassée, le service est considéré comme inacceptable. Cependant, si le dépassement se fait par la limite inférieure, le service reste acceptable. Cela signifie aussi qu'il y a gaspillage de ressources du système de communication.
- Comme nous l'avons cité auparavant, les applications qui ont été citées sont plus des exemples qu'une liste exhaustive. On peut insérer d'autres applications selon l'analogie avec les exemples cités.

3.2.5 Conclusion

Dans cette partie, nous avons présenté les exigences des utilisateurs finaux pour les applications utilisées. Les besoins en QoS des utilisateurs sont la tolérance aux pertes et l'acceptabilité du délai de transmission de bout en bout.

Parmi les applications identifiées, nous avons vu que les applications vidéo et audio sont les plus exigeantes en termes de délai, le plus petit possible, en gigue, quasi-nulle et en pertes minimales.

Nous allons présenter par la suite, les services offerts par les couches sous-jacentes qui proposent des mécanismes et fonctionnalités capables de répondre aux exigences d'applications multimédia.

3.3 Les modèles de garantie de la QoS

L'IETF a fait deux propositions majeures pour garantir la QoS au niveau réseau afin d'assurer un bon fonctionnement des services IP : les services intégrés (Integrated Services, IntServ) et les services différenciés (Differentiated Services, DiffServ).

3.3.1 Le modèle IntServ

En 1994, les efforts du groupe IntServ ont abouti à la mise en œuvre et définition d'une architecture de services intégrés [20]. Cette architecture comporte essentiellement deux éléments : un modèle de service étendu (IS) et un cadre d'implémentation de référence qui permet l'élaboration de ce modèle.

La base de cette architecture est la réservation de ressources par flux. Elle permet la prise en charge de la QoS sans modification du protocole IP. L'état des flux qui traverse chaque routeur doit être conservé à tout moment. Fondamentalement, le fonctionnement de l'Internet n'est pas conçu de la sorte. Celui-ci, au contraire, se basait jusqu'à présent sur la conservation des états des flux dans les terminaux utilisateurs. Quatre fonctionnalités additionnelles sont donc intégrées dans les routeurs :

- L'ordonnanceur de paquet : consiste à la mise en place d'un certain nombre de mécanismes à savoir des files d'attente et des timers pour qu'ils soient utilisés à l'acheminement des différents flux.
- Le classifieur : est responsable de la correspondance entre le paquet entrant et la classe de service associée. Le niveau de QoS de chaque classe est associé à chaque flux.
- Le contrôle d'admission : Il s'agit des mécanismes et algorithmes pour que le routeur décide d'offrir ou pas un niveau de QoS à un flux sans dégrader les garanties offertes précédemment.
- Le protocole d'établissement de réservation : le protocole choisi pour garantir la fonction de la création et le maintien de l'état des flux au sein des routeurs est RSVP (ReSerVation Protocol) [21].

Le "Best Effort" fait partie des classes de service définies. C'est le niveau de classification le plus bas et qui consiste en aucun traitement spécifique pour les paquets au niveau des routeurs. Deux nouvelles classes de service sont définies pour le traitement des flux au sein des routeurs :

- Le Guaranteed Service (GS) ou le service garanti [22]: les garanties en termes de bande passante et de temps maximal d'acheminement des paquets, c'est un service exprimé quantitativement. Le service est destiné aux applications temps réel sensibles au délai, telles

que la VoIP ou la vidéoconférence. Par contre, il n'y a aucun délai moyen garanti, et donc, l'application gère elle-même les variations de ce délai en utilisant des mécanismes de temporisation (buffers), au niveau de la réception.

- Le Controlled-Load (CL) ou le service de charge contrôlée [23]: c'est le service exprimé qualitativement en terme de capacité qui assure l'acheminement des paquets de l'utilisateur avec une QoS proche d'un réseau non congestionné.

L'architecture présentée permet d'offrir une gestion de la QoS de bout en bout d'une manière très fine, puisqu'elle garantit des niveaux de QoS non prédéfinis et contrôlés qui correspondent exactement aux besoins d'un flux applicatif. L'utilisation du protocole de signalisation RSVP permet aussi d'offrir un cadre de référence homogène d'un domaine à un autre, mais le problème principal reste le passage à grande échelle. En effet, le modèle est complètement inadéquat au cœur du réseau à cause de la concentration du trafic et des traitements qui y sont effectués.

3.3.2 Le modèle DiffServ

Bien que le modèle IntServ permette un traitement de flux par flux, et offre une architecture à QoS appréciable, dans le cas de grands réseaux, il reste difficilement applicable. Le groupe IETF s'est donc penché sur une autre solution, et propose une nouvelle approche, qui diffère du traitement flux par flux vers un traitement par des classes de services différenciés. La différenciation de service consiste à créer une agrégation de classes pour le trafic arrivant dans le réseau, selon le besoin en QoS. La granularité est donc moins fine. Mais la complexité dans les routeurs ne dépend plus du nombre de flux mais plutôt du nombre de classes de service dans les routeurs. Cela permettra aux routeurs de cœur du réseau d'être moins chargés et leurs tâches seront allégées. La complexité donc se déplace vers les routeurs d'extrémité du réseau. Le passage à l'échelle pour cette architecture est donc plus facile que celle avec les services intégrés [24].

En plus du Best Effort (BE), qui ne correspond à aucun traitement spécifique des paquets, ce modèle définit 2 classes de services appelées.

- Expedited Forwarding (EF) : défini dans [25], la classe avec la priorité maximale. Elle assure le transfert du flux tout en garantissant une bande passante avec des taux de perte, de délai et de gigue faibles.
- Assured Forwarding (AF) : défini dans [26], c'est la garantie d'acheminer d'un certain nombre de paquets en cas de congestion.

Le modèle DiffServ découpe le réseau en domaines DiffServ. Il est représenté dans la figure ci-dessous par des routeurs de cœur et des routeurs de bordure.

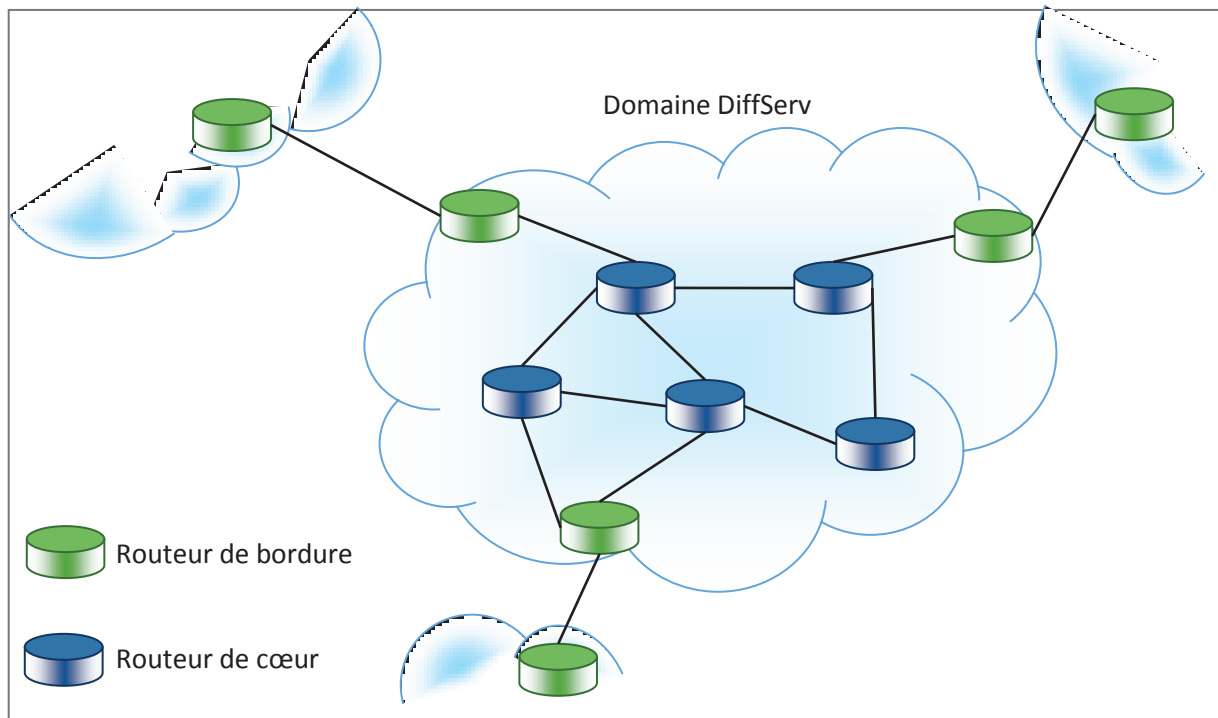


Figure 7 : Domaine DiffServ [12]

Les routeurs de bordure sont les premiers éléments actifs d'un domaine DiffServ puisque le trafic les traverse en premier. Une interconnexion est possible avec d'autres domaines DiffServ ou non DiffServ grâce à ces routeurs. Les fonctions principales effectuées par ces routeurs sont la classification des flux en classes de service et leur conditionnement. Des mécanismes de profilage et de mesure du trafic permettent la mise en forme des flux pour une correspondance au profil de trafic négocié. Dans le cas où les paquets ne respectent pas le profil, ils seront gérés par des mécanismes en les mettant dans des buffers, ou bien en leur affectant une priorité inférieure, sinon les ignorant. Par la suite, les paquets sont marqués en utilisant le paramètre DSCP (Differentiated Service Code Point) qui indique la classe de service à laquelle le flux appartient. Les champs de précedence des en-têtes IP sont utilisés à cet effet, en IPv4 : c'est le champ ToS (Type of Service), en IPv6 : c'est le champ CoS (Class of Service).

Ensuite, les paquets sont envoyés et seront identifiés par les routeurs de cœur grâce à leur marquage effectué précédemment, et dirigés vers la file de transmission correspondante qui implémente le service pour recevoir le traitement PHB correspondant. Concernant les priorités, elles sont assurées principalement par deux algorithmes, le PQ (Priority Queueing) et WRR (Weighted Round Robin), qui contrôlent l'affiliation des ressources entre les classes de services.

Cette architecture offre une solution de QoS dans les réseaux IP. La différenciation des classes permet un passage plus facile à l'échelle. En plus, des contrats peuvent être mis en œuvre entre le client et le fournisseur pour exprimer les besoins en termes de garantie de service. Ces contrats sont connus sous le nom de SLA (Service Level Agreement). La souplesse de tarification ainsi que la journalisation des prestations deviennent plus simples, mais la mise à jour des contrats reste une procédure relativement complexe et s'effectue manuellement.

Les deux modèles présentés précédemment, présentent un support important à la QoS dans les réseaux IP. Bien que l'architecture IntServ offre une QoS plus fine de bout en bout par rapport à DiffServ, avec un protocole de signalisation et de réservation assez robuste, l'intégration de services reste une solution assez lourde à implémenter dans un réseau étendu. DiffServ reste la plus utilisée comme modèle pour la facilité de déploiement ainsi que pour le contrôle des classes de services par le fournisseur. En effet, l'ETSI Broadband Satellite Multimedia BSM [27] recommande la différenciation de services pour la gestion de QoS dans les réseaux par satellite.

Dans la partie suivante, nous allons décrire les recommandations concernant la gestion de QoS pour les réseaux Internet d'une manière générale, et les réseaux mobiles type 3GPP/UMTS/GPRS.

3.3.3 La gestion de la QoS dans les réseaux large bande

Cette partie présente les classes de trafic définies par un certain nombre de groupes de normalisation. Les classes de trafic sont destinées à constituer la base des accords entre les utilisateurs finaux, les fournisseurs de services réseau et entre les prestataires de services [28].

Nous allons décrire brièvement les points majeurs des recommandations vues par le groupe ITU-T et le 3GPP. La gestion de la QoS hérite du travail déjà effectué dans le domaine sans fil et dans d'autres groupes de standardisation ETSI.

- **Vision ITU-T**

Le Tableau 2 résume l'ensemble des recommandations ITU-T Y.1541 [29]. Ces recommandations s'appliquent à tous les réseaux bout en bout à base d'IP. Elles définissent 6 classes avec 4 mécanismes principaux. L'avantage de cette classification est la mise à disposition d'une classe spéciale de signalisation qui est nécessaire pour la gestion de QoS sensible au délai. Cependant, cette classification ne permet pas de distinguer entre les applications avec une taille de paquets fixe (la voix par exemple) et variable (par exemple la vidéo et le transfert de données). Cette propriété pourrait être utilisée pour mieux préciser les ressources réseau nécessaires.

Classe de QoS	Applications	Mécanisme de base	Techniques
0	Temps réel, sensible à la gigue, grande interactivité (VoIP)	File séparée avec service préférentiel	Routage contraint
1	Temps réel, sensible à la gigue, interactivité (VoIP)		Routage moins contraint
2	Echange de données à grande interactivité (signalisations)	File séparée avec une priorité diminuée	Routage contraint
3	Echange de données interactives		Routage moins contraint
4	Applications à faible perte seulement (petit échange, vidéo streaming)	File d'une grande taille avec une priorité diminuée	N'importe quelle route
5	Applications traditionnelles dans un réseau IP par défaut	File séparée avec une basse priorité	N'importe quelle route

Tableau 2 : Les classes de QoS selon les recommandations ITU-T [29]

- **Vision 3GPP/UMTS/GPRS**

La nouvelle génération des services de téléphonie mobile utilise une QoS de bout en bout. Il en résulte un certain nombre de recommandations spécifiques [30]:

- Les attributs de QoS ne devraient pas être limités à un ou quelques mécanismes de contrôle externes de QoS mais le concept de QoS doit être capable de fournir des niveaux différents de QoS en utilisant les mécanismes spécifiques de l'UMTS (non liés aux mécanismes de QoS dans les réseaux externes).
- Tous les attributs doivent avoir un sens sans ambiguïté.
- Les mécanismes de QoS doivent autoriser l'utilisation efficace des capacités radio.
- Les mécanismes de QoS doivent autoriser une évolution indépendante du réseau cœur et d'accès, ainsi que les réseaux cellulaires.
- Les applications doivent être en mesure de définir des valeurs de QoS pour leurs transmissions de données.
- Le comportement QoS doit être dynamique, c'est à dire qu'il doit être possible de modifier les attributs de QoS au cours d'une session active.

Afin de respecter ces recommandations, des classes de trafic [Tableau 3] et des caractéristiques de QoS [Tableau 4] ont été développées. Elles se relatent aux classes de l'ITU-T du Tableau 2, où les classes de l'ITU-T 0 et 1 constituent la classe de conversation. Les deux classes 3 et 4 constituent la classe du streaming. La classe 4 s'applique également à la classe interactive. Le trafic de fond est la classe 5.

Classes de trafic	Utilisation	Exemple d'applications
Conversation	Trafic en temps réel de conversation impliquant les entités de conversation	téléphonie, téléconférence, vidéophonie et visioconférence, chat
Streaming	Trafic en temps réel en streaming impliquant l'envoi de l'information d'une entité à une autre	Diffusion audio et vidéo, surveillance
Interactive	Trafic en pseudo temps réel interactif impliquant la récupération des informations par une entité, d'une autre entité	Navigation web
Trafic background	Trafic de fond pas en temps réel impliquant la l'envoi d'informations d'une entité à une autre entité	Courier électronique et transfert de fichier

Tableau 3 : Classes de trafic pour le 3GPP/UMTS [30]

Classes de trafic	Composantes	Caractéristiques générales du trafic
Conversation	Speech Audio Vidéo Data MM	Constant Rate (CR) et Variable Rate (VR) Sensible au délai Sensible à la gigue Tolérance limitée des pertes/erreurs (selon codage)

Streaming	Speech Audio Vidéo Data MM	Variable Rate (VR) Tolérant au délai (bufferisation dans les terminaux) Sensible à la gigue (dépend des tailles des buffers dans les terminaux/gateways) Tolérance limitée des pertes/erreurs (selon codage)
Interactive	Donnée	Variable Rate (VR) Sensible au délai (mais plus tolérant que la conversation) Tolérant à la gigue Sensible aux pertes/erreurs
Trafic background	Donnée	Best Effort (BE) Pas de sensibilité au délai Tolérant à la gigue (et plus tolérant que la classe interactive) Sensible aux pertes/erreurs

Tableau 4 : Caractéristiques du trafic 3GPP/UMTS [29]

Après avoir résumé les différents éléments qui constituent la QoS ainsi que les différentes recommandations faites pour les flux au sein d'un réseau Internet satellite et cellulaire, nous présentons une notion importante qui est la qualité d'expérience QoE, qui va dans le même sens que la QoS pour améliorer le confort de l'utilisateur, mais prend ce problème d'un autre point de vue.

3.4 La Qualité d'Expérience

La Qualité d'Expérience QoE est une mesure subjective de l'expérience ou la qualité perçue par un client (end user).

Bien qu'elle soit liée à la Qualité de Service, la qualité d'expérience prend également en compte les exigences des utilisateurs finaux lors de l'utilisation des services. Par exemple, un opérateur peut offrir des services fiables de données (selon un contrat SLA par exemple), avec un bon niveau de QoS. Cependant, les utilisateurs peuvent toutefois être mécontents du rendu. Ce mécontentement est dû aux différences de perceptions entre les utilisateurs. Par exemple, un utilisateur peut trouver une vidéo, acceptable tandis que d'autres peuvent trouver cette même vidéo médiocre. De plus, des éléments externes peuvent influencer une transmission comme la qualité du matériel utilisé par l'utilisateur, et qui ne peut pas être mentionnée dans certains cas dans les SLAs. Pour toutes ces raisons, nous pouvons dire que la QoS est bonne mais la QoE est mauvaise.

Bien que la QoE soit liée à la QoS, elle est orientée utilisateur (perception utilisateur). C'est une mesure subjective de la satisfaction (ou non) de l'utilisateur du service reçu. Par exemple, La QoS applicative d'une vidéo en streaming est exprimée en délai de bout en bout, jigue ou bien BER. La QoE s'exprime dans ce cas en mesure globale de la qualité de la vidéo (fluidité, pixellisation, détails ...etc.).

La QoE est définie dans [31] comme l'acceptabilité globale d'une application ou un service, tel que perçue subjectivement par l'utilisateur final. Elle comprend les comportements complets du système de bout en bout (client, terminal, réseau, infrastructure de services, etc.) et peut être influencée par les attentes des utilisateurs et du contexte. Par conséquent, en principe, la QoE est mesurée de manière subjective par l'utilisateur final et peut différer d'un utilisateur à l'autre. Toutefois, elle est souvent estimée en utilisant des mesures objectives.

Les différences dans l'acuité perceptive et la préférence signifient que les jugements obtenus à partir de la QoE des personnes différentes peuvent varier. Par conséquent, les mesures de la QoE sont

généralement faites en utilisant les données de groupe. Lorsque les études nécessaires ont été effectuées pour calibrer la relation avec la QdE, elle peut également être estimée en utilisant des mesures objectives.

Un certain nombre de caractéristiques de performance du système contribuent à la QdE du flux de données. Par exemple, le codec et le débit d'encodage utilisé, la résolution des médias dans la source et à l'affichage, la corruption ou la perte des données, et des délais. Les interactions entre le contenu vidéo, le codec et le débit binaire utilisé et les bits spécifiques corrompus et / ou paquets perdus contribuent à une forte variabilité de la qualité perçue de la sortie vidéo.

Il existe d'autres facteurs qui peuvent influencer la réponse du spectateur. Certains de ceux-ci affectent la perception de la qualité, tels que le contexte de l'arrêt (une image particulière sera évaluée d'une façon dans le cadre de la définition TV standard (SDTV), l'autre dans le contexte de la télévision haute définition (HDTV), et encore différemment dans le contexte d'un clip vidéo sur Internet), le contexte culturel, la motivation, l'attention des facteurs connexes, état émotionnel et ainsi de suite. Les évaluations directes de la QdE sont conçues pour exclure ces facteurs, car ils ne sont généralement pas sous le contrôle d'un opérateur de réseau et donc ne contribuent pas aux exigences de l'équipement.

D'autres facteurs influencent le jugement du spectateur de l'acceptabilité. Il s'agit notamment des choses comme une expérience antérieure avec le mode de communication spécifique ou des modes connexes, le coût du service et les avantages particuliers du service fourni (la mobilité, l'indépendance de temps, la bibliothèque du programme exceptionnellement élevé) et ainsi de suite.

L'acceptabilité n'est pas équivalente à la QdE. Une image de faible résolution vidéo aura une QdE plus basse qu'une image à haute résolution, mais il peut être tout à fait acceptable pour certaines applications et de services, en fonction du dispositif d'extrémité, la taille physique de l'écran, et l'objectif pour lequel il est utilisé.

La figure suivante illustre les facteurs contribuant à la QdE. Ces facteurs sont organisés comme ceux liés à la Qualité de Service et ceux qui peuvent être classés en tant que composantes humaines.

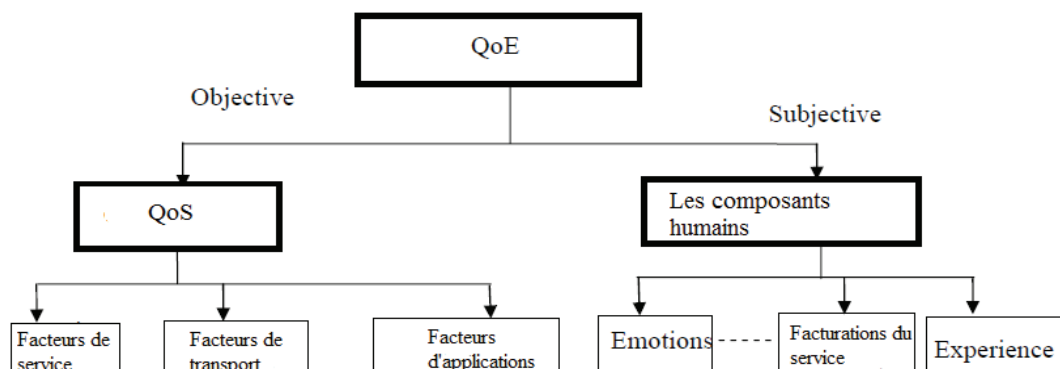


Figure 8 : Les facteurs contribuant à la QdE [32]

3.4.1 LA QdE pour la vidéo et la voix

Bien que les paramètres QoS fournissent une indication de la façon dont l'intégrité des paquets est préservée, ils ne fournissent pas un indicateur de la façon dont le contenu livré imite une expérience du monde réel. La perte de paquets représentant de la parole ou de la vidéo est plus ou moins perceptible selon l'endroit où elle se produit dans une conversation ou une image. Cette perception de réglage fin a conduit à la création d'applications spécifiques des modèles expérimentaux qui abordent la QdE. L'idée est de fournir une corrélation entre ce que le consommateur s'attend à voir et à entendre et ce que le consommateur voit réellement et entend. La première mesure de la QdE est la note d'opinion moyenne (MOS : Mean Opinion Score), initialement appliquée à la téléphonie.

Pour avoir un MOS, l'évaluateur assemble un groupe type d'individus (jury) qui évalue la qualité des échantillons de voix et/ou la vidéo choisi pour l'expérience. Les individus notent la qualité du système sous test de 1 à 5, avec 1 indiquant la qualité la plus basse et 5 indiquant la meilleure qualité. Le MOS du système testé est donc la moyenne des scores des membres du jury. L'inconvénient évident de cette méthode est la nécessité de la présence humaine (évaluateurs) pour chaque test. Pour pallier cet inconvénient, des modèles mathématiques ont été construits pour relier MOS subjectives à des paramètres mesurables, d'abord pour la parole et ensuite pour la vidéo.

Les scores MOS semblent être un bon moyen d'évaluer justement la qualité de la voix obtenue. Néanmoins, le principal inconvénient de ce modèle est qu'il est "lourd" à mettre en œuvre. En effet, pour que le score soit juste, il est nécessaire d'interroger un grand nombre de personnes sur un même échantillon. Il requiert donc un grand nombre de moyens et implique un temps d'évaluation relativement long.

Le E-modèle (défini par [33] et [32]), par exemple, utilise un ensemble de facteurs de dépréciation provenant de la gigue du réseau, des pertes, du délai des paquets, et de la performance du codec pour noter les performances du réseau sur une échelle de 0-100. La somme des facteurs de dépréciation est soustraite à partir d'un score idéal de 100 pour arriver à un indice de qualité de bout en bout. Ce modèle a été initialement conçu pour des évaluations de modèles, plutôt que les réseaux en service mais depuis, il a été appliqué à des échantillons de VoIP. D'autres modèles ont été créés pour tester la qualité de la voix dans les réseaux. La mesure de la qualité perceptive de la parole (PSQM : Perceptual Speech Quality Measurement), la perception du système de mesure d'analyse (PAMS : Perceptual Analysis Measurement System), et l'évaluation perceptuelle de la qualité vocale (PESQ : Perceptual Evaluation Of Speech Quality) produisent des scores qui peuvent également être corrélés au MOS. Le tableau ci-dessous donne la corrélation généralement utilisée par les fournisseurs d'équipements de test. Au fil du temps, cette corrélation a gagné un crédit suffisant pour être utilisée comme base pour les MOS générés par des algorithmes dans les adaptateurs de terminaux multimédia embarqués (EMTA : Embedded Multimedia Terminal Adapter). Le Tableau 5 présente les différents scores du MOS et leurs équivalences à ceux de l'E-model.

MOS Score	Niveau de qualité	Echelle d'effort	Score équivalent à l'E-Model	Exemple d'application/Codec
4.3-5.0	Très satisfaisant (MOS=5 indique un score	Relaxation complète	90-100	G.711 (MOS=4,2-4,3)

	excellent)	possible, aucun effort requis		
4.0-4.3	Satisfaisant (MOS=4 indique un bon score)	Attention nécessaire mais pas d'efforts importants	80-90	AMR (MOS=4,1)
3.6-4.0	Certains utilisateurs sont satisfaits (MOS=3 indique un score moyen)	Effort modéré requis	70-80	G.729 (MOS=4)
3.1-3.6	Plusieurs utilisateurs sont insatisfaits	Effort considérable requis	60-70	G.729 x 2 Encodings (MOS= 3.27)
2.6-31	Presque tous les utilisateurs sont insatisfaits (MOS=2 indique un score faible)	Incompréhensible	50-60	G.729 x 3 Encodings (MOS= 2.68)
1.0-2.6	Pas recommandé (MOS=1 indique un score mauvais)	non réception du signal	0-50	/

Tableau 5 : MOS et le score équivalent pour les applications de voix/vidéo (IUT-T P.800)

La corrélation entre les paramètres mesurables et les notes quantitatives a été étendue aux applications vidéo, mais la mesure de la QdE pour la vidéo est plus exigeante que la mesure audio. La perte de paquets vidéo ou le délai entraînent une série de dégradations, en fonction non seulement de ce qui est représenté par les bits perdus, mais aussi du taux de compression des séquences d'images vidéo [32].

En plus de la qualité d'image, les systèmes de mesure de la Qualité d'Expérience mettent en avant le confort d'utilisation d'un service. Par exemple la réactivité lors de la navigation dans les menus d'un portail de vidéo à la demande.

3.4.2 La QdE pour le texte et les graphiques

Le contenu média textuel

Le contenu textuel possède aussi ses recommandations de QdE. Le texte est un moyen de représentation composé de caractères formatés. Il est stocké et transmis comme une séquence de codes. Bien qu'il puisse être affiché sur le même écran que la vidéo et images fixes, il faut le décoder

dans les polices spécifiques pour la présentation à l'utilisateur, que ce soit sur l'écran ou sur papier. L'entrée se fait par un clavier. La sortie peut être présentée par une imprimante ou sur un écran.

Quatre niveaux de qualités ont été définis :

- T0 : représente la qualité minimale, l'alphabet de base et de ponctuation, pas de formatage ou de choix de police.
- T0Bis : la qualité de vidéotex, l'alphabet de base et de ponctuation, de base du jeu de caractères graphiques, pas de formatage ou de choix de police.
- T1 : la qualité de conversation texte utilisable est caractérisée par :
 - Support des polices pour la zone de langue ISO 10646 Latin-1, plus la zone de la langue ciblée pour la mise en œuvre.
 - Pas plus d'un caractère corrompu, perdu ou a marqué manquant sur 100.
 - Le délai d'entrée de caractères dans l'émetteur à l'affichage dans le récepteur doit être inférieur à 2 s.
- T2 : Bonne qualité de la conversation texte est caractérisé par :
 - Support des polices pour tous les caractères dans [b-ISO 10646].
 - Pas plus d'un caractère corrompu, perdu ou a marqué manquant par 500.
 - Le délai d'entrée de caractères dans l'émetteur à l'affichage dans le récepteur doit être inférieur à 1 s.

L'utilisateur peut avoir le contrôle sur le texte par l'édition et les fonctions de présentation. Il peut également être en mesure d'insérer des graphiques, des images fixes ou images animées dans le texte.

La qualité du texte dépend principalement de la capacité pour le formatage et l'utilisation de différents types de polices et caractères spéciaux. Quand aucune correction d'erreur n'est faite, par exemple dans la conversation, la qualité du texte est également mesurée en termes de caractères corrompus, perdus et remplacés par le marqueur de texte manquant [34].

Les caractères avec leur formatage peuvent être décodés et assemblés dans des bitmaps qui peuvent ensuite être traités comme des images fixes, par exemple, les pages de télécopies.

Le contenu média graphique

Les composants multimédia graphiques [32] permettent la capture et la représentation de l'information, son transfert depuis l'utilisateur d'origine à l'utilisateur de destination, sa présentation à l'utilisateur humain, le traitement, l'archivage et la récupération. Ce composant média permet aux images graphiques d'être capturées et transmises comme objets géométriques dont les positions, les formes et les couleurs sont codées de sorte qu'elles puissent être reproduites dans un terminal distant.

La qualité intrinsèque de l'image dépend du nombre et de la complexité des objets qui peuvent être générés, la précision de leurs dimensions et leurs positions, le nombre de couleurs possibles. La qualité globale perçue par l'utilisateur dépend aussi de la résolution de l'entrée et de sortie des systèmes.

Les objets graphiques peuvent être décodés et assemblés dans des bitmaps qui peuvent ensuite être traités comme des images fixes.

3.4.3 La QdE pour les fonctions de contrôle

QdE du Zapping

Le temps de zapping des chaînes (temps de commutation chaîne) a des liens solides avec l'expérience de l'utilisateur final de le QdS. Généralement, il est principalement déterminé par le temps nécessaire à démarrer le traitement et le décodage de la nouvelle chaîne, à travers le décodeur TV (Set To Box - STB). Les requêtes de changement ou zapping de chaîne peuvent se produire lorsque

- Il y a une demande de métadonnées dans Guide Electronique des Programmes (Electronic Program Guide - EPG) ou le Guide Interactive des Programmes (Interactive Program Guide - IPG).
- À une sélection aléatoire d'une chaîne en saisissant son numéro à l'aide de la télécommande.
- Le bouton haut/bas des chaînes est enfoncé dans la télécommande.
- Le bouton haut/bas des chaînes est enfoncé dans le panneau de la STB.
- Un canal est sélectionné dans le menu de l'application IPG.
- Le STB / TV est sous tension et réglée sur le canal initial attribué par l'IPG.

En tant que paramètre de QdE, le temps de zapping peut être représenté par trois éléments : le retard IGMP (Internet Group Management Protocol). Ce protocole est utilisé pour les requêtes pour joindre ou quitter une chaîne sur un décodeur TV, la mise en mémoire tampon et le délai de décodage.

Un des éléments clés impliqués dans la validation de la QdE en service IPTV est de savoir comment les utilisateurs peuvent changer rapidement les chaînes de télévision, qui est souvent désigné comme le temps de zapping. Cependant, la relation explicite entre le temps de zapping et la qualité perçue par l'utilisateur exprimée par un score d'opinion moyenne (MOS) est encore à l'étude [35].

Recommandations de QdE relatives à la VoD

La vidéo à la demande (VoD) fournit des fonctionnalités VCR-like (contenant ces fonctionnalités : sélection vidéo, lecture, pause, rembobinage, Fast Forward -FFW-, arrêt). Lorsqu'un abonné désire un contenu vidéo à travers la STB, l'abonné accède au contenu vidéo à partir de l'EPG (Electronic Program Guide) qui supporte le moteur de recherche du contenu pour aider à accéder au contenu de l'information. Pour garantir la flexibilité de VCR-like, ce mode offre la possibilité de gérer la pause, la lecture, le retour, l'avance rapide, et l'arrêt des entrées pour ces fonctions de contrôle.

Chaque fonction de contrôle (sélection vidéo, lecture, pause, rembobinage, FFW, arrêt) a son propre délai. Les métriques de QdE pour la qualité de la VoD de transaction sont exprimées par les indicateurs suivants :

- Le délai du processus de sélection de vidéo : c'est la période à partir du moment où le sujet est choisi jusqu'à l'instant où le contenu est affiché.

- Jouer du retard : c'est la période à partir du moment où l'entrée a été sélectionnée pour jouer jusqu'au moment où le contenu est affiché.
- Arrêter de retard : c'est la période à partir du moment où l'entrée Arrêt de la vidéo a été sélectionnée jusqu'à l'instant où le contenu est arrêté, comme indiqué par l'affichage de contenu vidéo.
- Délai Rewind : c'est la période à partir du moment où l'entrée Rewind de la vidéo a été sélectionnée jusqu'à l'instant où l'action de rembobinage est exécutée comme indiqué sur le dispositif d'affichage.
- Pause de retard : c'est la période à partir du moment où l'entrée Pause de la vidéo a été sélectionnée jusqu'à l'instant où l'action de pause est exécutée comme indiqué sur le dispositif d'affichage.
- Délai FFW : c'est la période à partir du moment où l'entrée Fast Forward de la vidéo a été sélectionnée jusqu'au moment où l'action FFW est exécutée comme indiqué sur le dispositif d'affichage.

Du point de vue de QdE, la latence des fonctions est l'un des enjeux les plus importants pour garantir la satisfaction des abonnés. Comme chaque élément de latence de fonction affecte directement la QdE, la latence doit être suffisamment faible pour répondre aux besoins des utilisateurs du point de vue des caractéristiques de fonctions VoD.

3.4.4 QdE pour les autres services de l'IPTV

QdE pour les metadata (métadonnées)

Les composantes métadonnées sont représentées dans la Figure 9 :

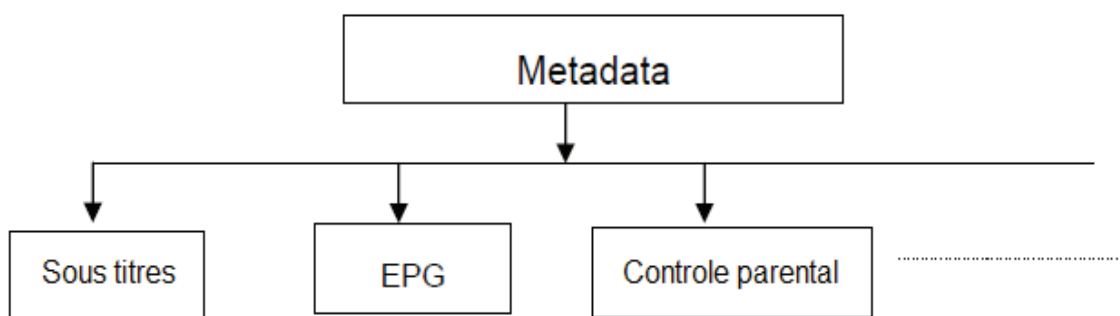


Figure 9 : Composantes métadonnées [32]

Les recommandations pour les métadonnées sont :

- Une haute disponibilité est recommandée lors de la transmission des métadonnées sur le réseau.
- La taille des données : les métadonnées sont transportées d'une manière à ce que la taille des données transportées soit suffisamment petite, par rapport à des facteurs tels que le nombre de services au total, le nombre de contenus et la bande passante réseau.
- L'exactitude : le fournisseur de services doit s'assurer que les métadonnées marquées d'un contenu particulier sont correctes.

Un exemple pour illustrer l'importance des métadonnées est la justesse de «notation» des contenus. La cote correcte sur le contenu est directement liée à ce que le client attend. Une notation parentale incorrecte, par exemple, une note «familiale» pour un contenu adulte, peut avoir de graves implications pour l'expérience client et du prestataire de service.

Les éléments suivants sont recommandés et qui peuvent être considérés comme partie de la QdE des services IPTV (EPG).

- La convivialité : l'interface utilisateur EPG doit d'être conçue pour une facilité d'utilisation.
- Le temps de réponse pour afficher la page EPG : le temps de réponse (l'intervalle de temps écoulé entre le moment de l'appui sur la touche EPG de la télécommande et le moment où la page est affichée EPG) doit être suffisamment court.

QdE pour la navigation web

Si un navigateur, est utilisé pour fournir à l'utilisateur un contenu interactif du prestataire de service, les points suivants doivent pour être pris en compte :

- Caractéristiques de la télévision : Les exigences IPTV de la QdE sur les navigateurs doivent prendre en compte le fait que les comportements et attentes des utilisateurs de télévision sont généralement différents de celles des utilisateurs de PC.
- Les écrans TV-like : Certaines fonctionnalités des écrans TV-like sont recommandées et considérées comme nécessaire pour la QdE navigateur, telles que les fonctions de superposition.
- La taille des caractères : la taille des caractères doit être suffisamment grande.
- La navigation : la fonction de navigation doit être considérée afin d'augmenter le niveau de commodité et d'opérabilité.
- Les cookies : l'utilisation des cookies doit être faite avec soin en raison de l'éventuelle limitation de la capacité de mémoire non volatile du terminal.

QdE pour le contenu de la navigation

Le contenu de navigation est défini comme des fonctions de découverte et de sélection de contenu. Ainsi, le contenu de navigation est fourni par diverses méthodes telles que la sélection de chaîne directe, l'EPG et les recommandations. Les exigences de QdE sont décrites dans les clauses suivantes en fonction des méthodes de navigation :

- La facilité d'accès au contenu de navigation par sélection directe de chaîne, en utilisant les boutons haut/bas.
- Le temps nécessaire pour découvrir et sélectionner le contenu de navigation EPG, et des évaluations subjectives de la facilité d'utilisation (par exemple, le MOS) sont recommandés pour être considéré.
- Les recommandations du contenu effectives pour les utilisateurs de son/ses amis sont également efficaces.

Pour résumer, le tableau suivant synthétise les méthodes d'évaluation subjective utilisées dans le contexte qualité d'expérience.

Application	Type d'évaluation
Audio	<ul style="list-style-type: none"> - PSQM (ITU-T P.861) - PAMS ITU-T P.800 - PESQ ITU-P.862.2 - E-Model ITU-T G.107 - MOS (ITU-T P.800, P.11)
Vidéo	<ul style="list-style-type: none"> - DCR/ACR/PC (ITU-T P. 910) - DSIS, DSCQS, SSCQE, DSCS (ITU-R BT.500) - MOS (ITU-T P.800, P.11)
Contexte média textuel et graphique	<ul style="list-style-type: none"> - Classification en 4 niveaux de qualité (comme le MOS) T0, T1, T2 et T3
QdE du Zapping/ VoD	<ul style="list-style-type: none"> - Vitesse de navigation - Vitesse d'accès aux chaînes - La maniabilité/facilité de navigation - L'Interactivité - Vitesse de changement de chaîne
l'IPTV	<ul style="list-style-type: none"> - La taille des données d'IPTV - La disponibilité - L'exactitude - La convivialité - Le temps de réponse pour afficher la page EPG - La facilité d'accès au contenu

Tableau 6 : Synthèse sur la qualité d'expérience

Comme nous l'avons déjà dit, les réseaux satellite sont utilisés comme solutions complémentaires aux réseaux terrestres. Dans ce contexte, lors d'un changement de réseau, des mécanismes de mobilité sont utilisés afin de garder la connexion active. Dans la partie suivante, nous allons présenter ces mécanismes de mobilité ainsi que les différents schémas et les scénarios de mobilité.

4 La mobilité et sa gestion

Il existe de nombreux cas d'utilisation pour lesquels les systèmes de communications hybrides satellite/terrestre pourraient s'avérer précieux.

En complément de technologies d'accès régulières terrestres, les réseaux satellite apportent de réels avantages pour les utilisateurs ou les réseaux mobiles. Le satellite fournit une large couverture tout en offrant des capacités relativement bonnes (performances, QoS, sécurité). Bien sûr, les accès par satellite n'ont pas l'intention de rivaliser avec les infrastructures terrestres et urbaines, mais peuvent être perçus comme une solution complémentaire, en apportant une alternative, qui pourrait être très utile lorsque l'infrastructure terrestre n'est pas efficace, détruite ou carrément pas disponible (en raison de la couverture).

Par conséquent, les cas typiques d'utilisation s'appliquent principalement aux domaines spécifiques comme la protection des civils (pour la gestion des catastrophes), militaire (théâtre) et aussi

le domaine des transports (soit pour le transport maritime, aéronautique, voies ferroviaire...etc.). Le but n'est pas de justifier les avantages des réseaux satellite, mais plutôt d'établir des scénarios pour l'infrastructure réseau tout entière, tout en restant sur le même niveau de l'évolution des réseaux NGN/4G (protocoles et normes).

Dans un premier temps, nous allons décrire la façon dont l'intégration peut se faire au niveau du système, puis, dans la deuxième étape nous considérons les utilisations pour élaborer les scénarios de mobilité.

4.1 Les scénarios hybrides de mobilité

L'intégration des réseaux satellite avec le réseau terrestre peut être considérée de plusieurs manières. De nombreuses solutions techniques peuvent être envisagées, selon le degré d'intégration entre le réseau satellite et le réseau terrestre. On distingue principalement trois scénarios pour les réseaux hybrides satellite/terrestre.

4.1.1 Scénario "intégration fortement couplée"

Dans le scénario, présenté dans la Figure 10, le réseau satellite est complètement fusionné avec le réseau ciblé (3G, LTE, WiMax) dans un environnement complètement transparent pour l'utilisateur mobile. L'interface d'accès radio est étendue (infrastructure et protocole) pour supporter les médias par satellite, en tant qu'accès alternatif pour l'utilisateur mobile.

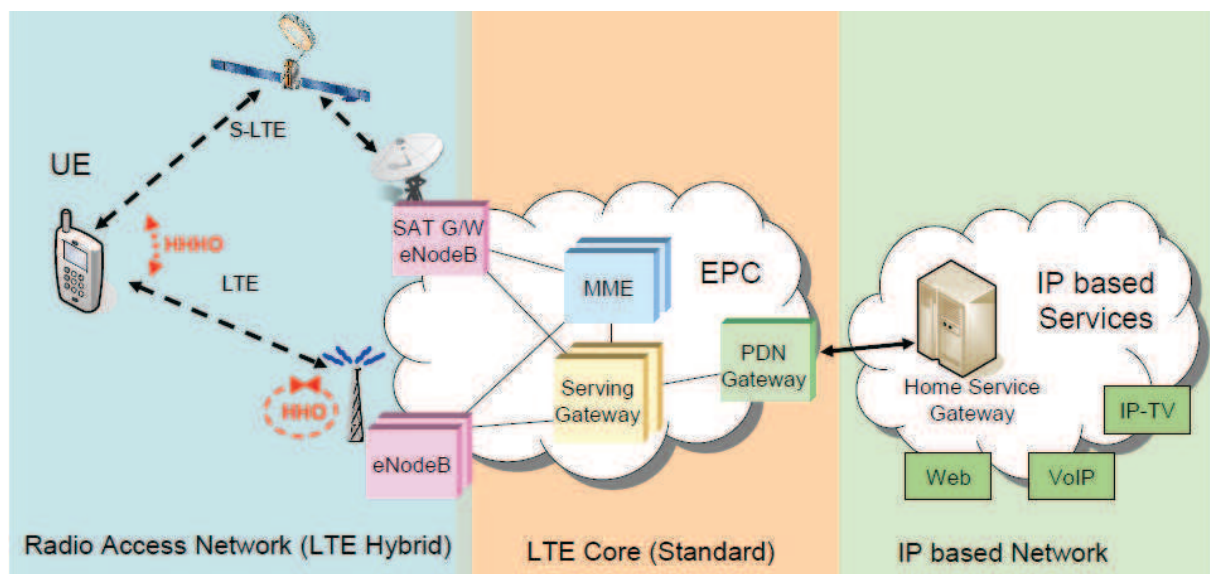


Figure 10 : Architecture d'intégration fortement couplée LTE [36]

Le Centre National des Etudes Spatiales (CNES) [37] mène des études fondées sur cette approche (notamment l'initiative SWIMAX). Prenons l'exemple des systèmes LTE, dans ce cas, le réseau satellite peut être directement intégré dans l'infrastructure de base et la passerelle satellite devient un autre eNodeB. Le terminal mobile doit s'interfacer avec la passerelle satellitaire par l'intermédiaire du satellite, basé sur le protocole terrestre (qui peut être adapté ou à l'écoute de ce média).

Cette approche est considérée comme l'étape ultime de l'intégration du satellite dans un réseau hybride. C'est parce que le système satellite est dérivé pour se conformer au protocole mobile et est

complètement intégré au noyau réseau, par l'intermédiaire d'une interface standard (eNodeB). Il est aussi le plus puissant du point de vue utilisateur. Par contre, il est le plus ambitieux, puisqu'il nécessite des fonctions d'intégration complexes, au moins au niveau du terminal, mais aussi au niveau du satellite afin de garder les équipements combinés les plus petits possible.

Le terminal est désigné comme un terminal hybride, car il peut interagir avec l'antenne satellite ou terrestre, en utilisant la même pile protocolaire (LTE, ou encore WiMax). La gestion de la mobilité est assurée par le présent protocole, à la couche accès, de la même manière que dans le réseau cellulaire terrestre.

Les caractéristiques de cette approche pour un système intégré LTE sont :

- Le protocole d'accès : LTE (extension du standard).
- Le terminal : hybride or double (terrestre/satellite intégré).
- Radio Access Network (RAN) : hybride (terrestre/satellite infrastructures).
- Satellite : MSS (Mobile Satellite Services) spécifique.
- La gateway satellite : spécifique fonctionnant comme un enodeB.
- La mobilité : assurée par LTE (par le protocole même, au niveau 2).
 - Horizontal handover (HHO) entre l'eNodeB terrestre.
 - Hybride HHO (HHHO) entre l'eNode satellite/terrestre B.

La mobilité du réseau est inutile : l'adresse IP est définie / maintenue, par un PDN-Gateway (Packet Data Network Gateway –PGW- : C'est le point d'entrée et de sortie des paquets pour l'utilisateur au sein du réseau).

La Figure 11 présente les différentes interactions entre les éléments de cette architecture.

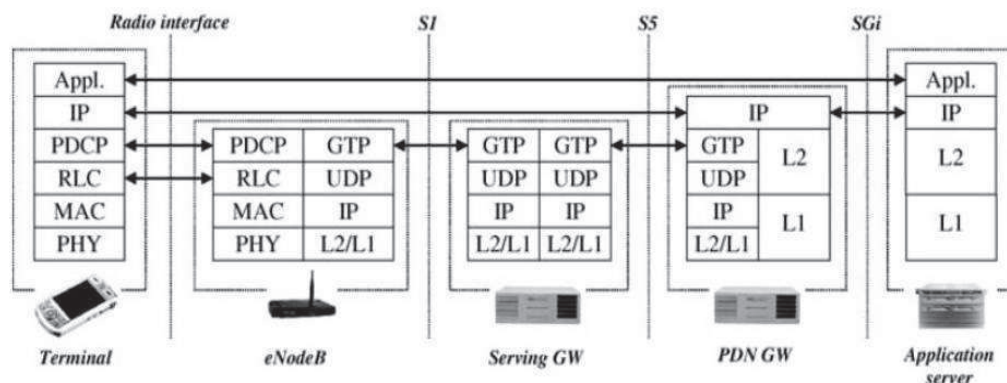


Figure 11 : Pile protocolaire LTE (Plan utilisateur issu des documents standards 3GPP) [36]

Bien que ce scénario présente l'avantage d'utiliser un seul équipement d'accès pour les deux réseaux, il présente quelques problèmes. En effet, l'un des problèmes principaux est de faire coexister deux réseaux de nature différente (satellite et terrestre) puisqu'ils n'implémentent pas les mêmes équipements d'accès. Un autre problème est celui des interférences entre les deux réseaux puisque les composants satellite et terrestre partagent les mêmes fréquences. Dans ce cas, la fréquence utilisée par le système satellite peut être identique à celle du système terrestre. C'est un moyen

d'augmenter la capacité de l'ensemble. Le problème majeur associé à cette réutilisation réside dans la gestion des interférences co-fréquence entre équipements du système satellite et équipements du système terrestre [38].

4.1.2 Scénario architecture relai

Dans le scénario architecture relai, présenté dans la Figure 12, le satellite est intégré dans l'infrastructure de réseau mobile en tant que relai. Cela signifie qu'il n'est pas positionné à l'interface radio avec le mobile, mais au-delà d'un relai satellite spécifique permettant l'accès à l'infrastructure du cœur du mobile.

Ainsi le terminal mobile est totalement standard pour le système mobile ciblé, généralement LTE (ou WiMax). Il ne sera plus un double équipement ainsi le satellite reste un FSS (Fixed Satellite Service) classique. Toutefois, il appartient à la passerelle satellite de s'interfacer avec le réseau mobile de base. Dans le modèle LTE, il pourrait agir comme un eNodeB ou un GW.

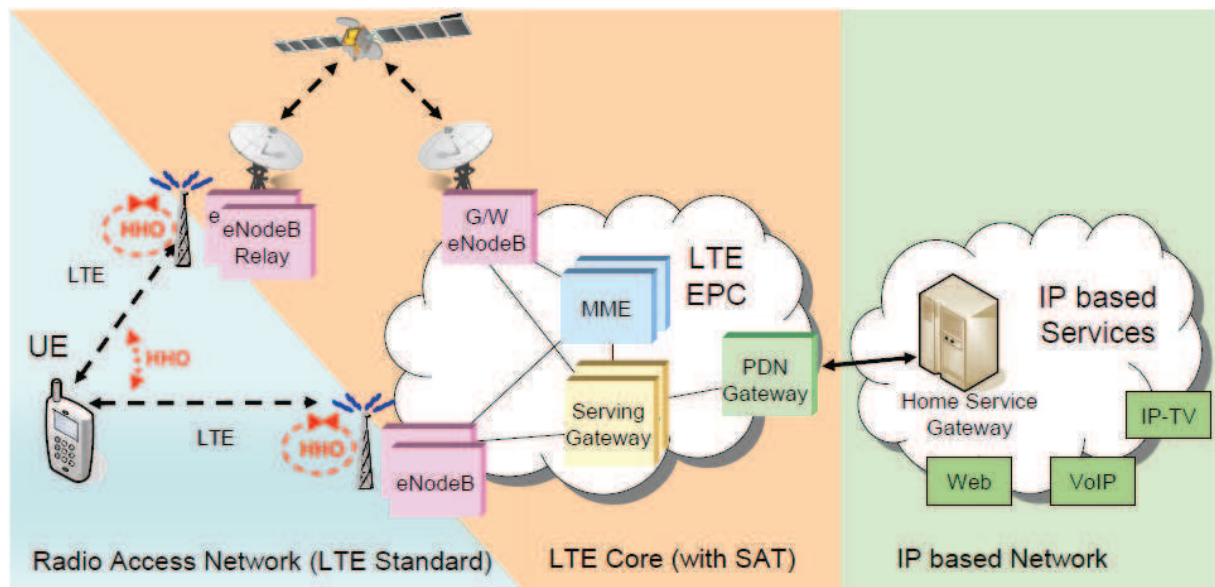


Figure 12 : Architecture Relai LTE [36]

Les caractéristiques de cette approche pour un système relai LTE sont :

- Le protocole d'accès : LTE (standard).
- Le terminal : LTE (standard).
- RAN : LTE Terrestre (avec un eNodeB qui relai le satellite à la base).
- Satellite : satellite à FSS régulier.
- La gateway Satellite : gateway standard étendu à agir comme un eNodeB ou SGW.
- Mobilité : assurée par LTE (HHO entre l'eNodeB ou le relai eNodeB).

L'avantage de cette solution est le maintien de la connexion active puisque l'adresse IP reste inchangée (définie/maintenue par un PDN-GW). La mobilité du réseau est donc inutile. Cela permet une continuité dans le service et une transparence pour l'utilisateur lors du handover.

4.1.3 Scénario faiblement couplé

La Figure 13 présente le scénario faiblement couplé. Dans ce dernier, une interface satellite spécifique est ajoutée au terminal mobile permettant la connexion au réseau de données IP, à travers un réseau d'accès basique. Il n'y a pas d'intégration avec le système mobile terrestre, contrairement aux précédents scénarios. Cela implique la mise en œuvre d'un terminal mobile multi-technologies, manipulant des interfaces pertinentes et exécutant les protocoles convenables (pour le satellite, un accès spécifique des médias comme le DVB-RCS + M est adéquat).

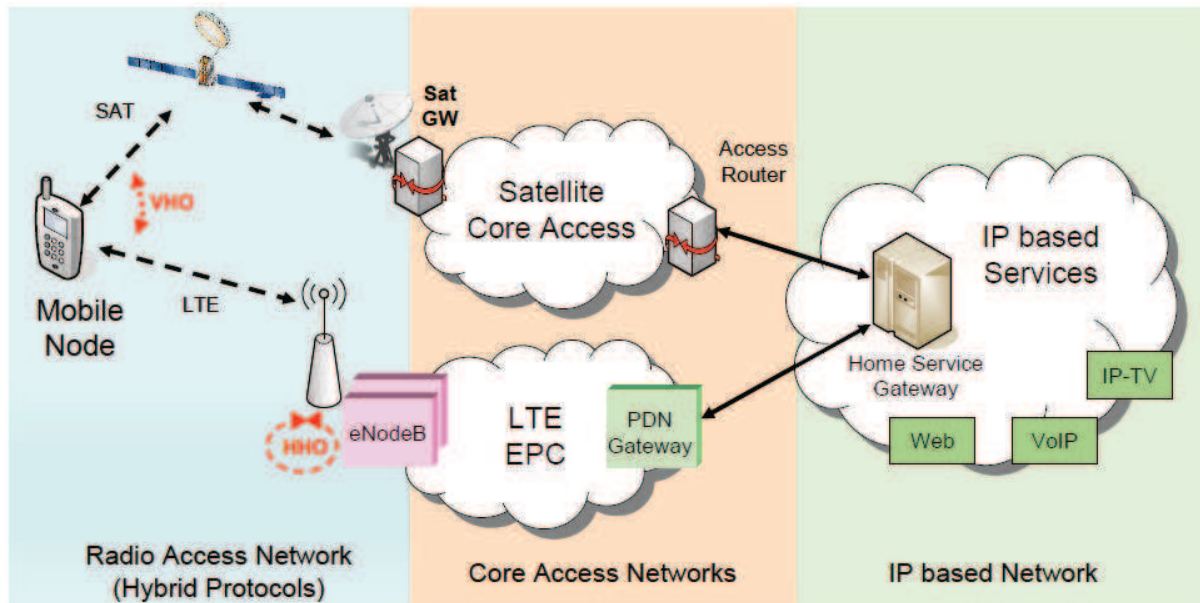


Figure 13 : architecture faiblement couplée LTE/Satellite [36]

Cette architecture peut être généralisée à toute technologie et ne se limite pas à la technologie LTE et permet à un nœud mobile d'interconnecter des services de réseau IP, par le biais des réseaux d'accès hétérogènes.

4.2 Schéma de mobilité

4.2.1 Mobilité horizontale et verticale

Deux types de mobilités peuvent être considérés :

- HHO : Horizontal Handover, où la mobilité horizontale. C'est quand la mobilité est réalisée entre différentes stations avec base dans les mêmes technologies au niveau liaison. Nous introduisons aussi la sous-classe HHHO (Hybrid Horizontal Hand Over) qui a pour but de signaler si le lien peut être changé du réseau terrestre au réseau satellite et vice versa, basé sur le même protocole (approche intégrée).
- VHO : Vertical Handover, ou la mobilité verticale. C'est quand la mobilité a lieu entre différents systèmes.

Le HHO est essentiellement géré par le protocole média, basé sur des fonctions MAC et physiques. Le VHO est aussi basé sur deux différents protocoles d'accès au média, mais nécessite des fonctions de décision de haut niveau, afin de trouver les réseaux d'accès disponibles, activer la préparation du changement de média et décider le changement de réseaux. Dans ce cas, on peut citer MIH qui permet d'effectuer un VHO et qui sera présenté dans la section suivante.

Contrairement au handover horizontal, le handover vertical peut conduire à des changements significatifs dans les débits de données pris en charge et des latences induites selon le réseau impliqué dans le transfert. En handover horizontal, la mobilité à la couche réseau est généralement effectuée après un changement de point d'accès sans fil (c'est-à-dire après l'achèvement de la mobilité des données au niveau liaison).

En revanche, au cours d'un VHO dans les réseaux de recouvrement (réseaux bâtis sur un autre réseau ou superposition de réseaux), la mobilité de couche réseau est généralement découplée de la couche de liaison et peut fonctionner indépendamment, puisque le nœud mobile est capable d'utiliser plusieurs de ses interfaces simultanément pour envoyer et recevoir des paquets.

Les deux mobilités VHO et HHO peuvent opérer en mode "break before make" BBM ou "make before break" MBB, ce qui signifie qu'il pourrait y avoir ou non interruption de la connectivité à la couche 2 pendant une certaine période lorsque la mobilité survient.

Pour le BBM, l'impact minimum peut être la perte à la volée des paquets, si l'opération de rupture du lien est trop longue pour les timeouts des couches supérieures.

Cependant, il existe des solutions standards permettant à un nœud mobile de passer en transparence entre une liaison de données et une autre, présentées brièvement dans la partie suivante.

4.2.2 Classification des schémas de mobilité

Durant les années précédentes, une multitude de schémas de gestion de la mobilité de bout en bout ont été proposés. En effet, la mobilité peut être gérée à de différents niveaux comme décrit sur la Figure 14 de la pile classique TCP/IP.

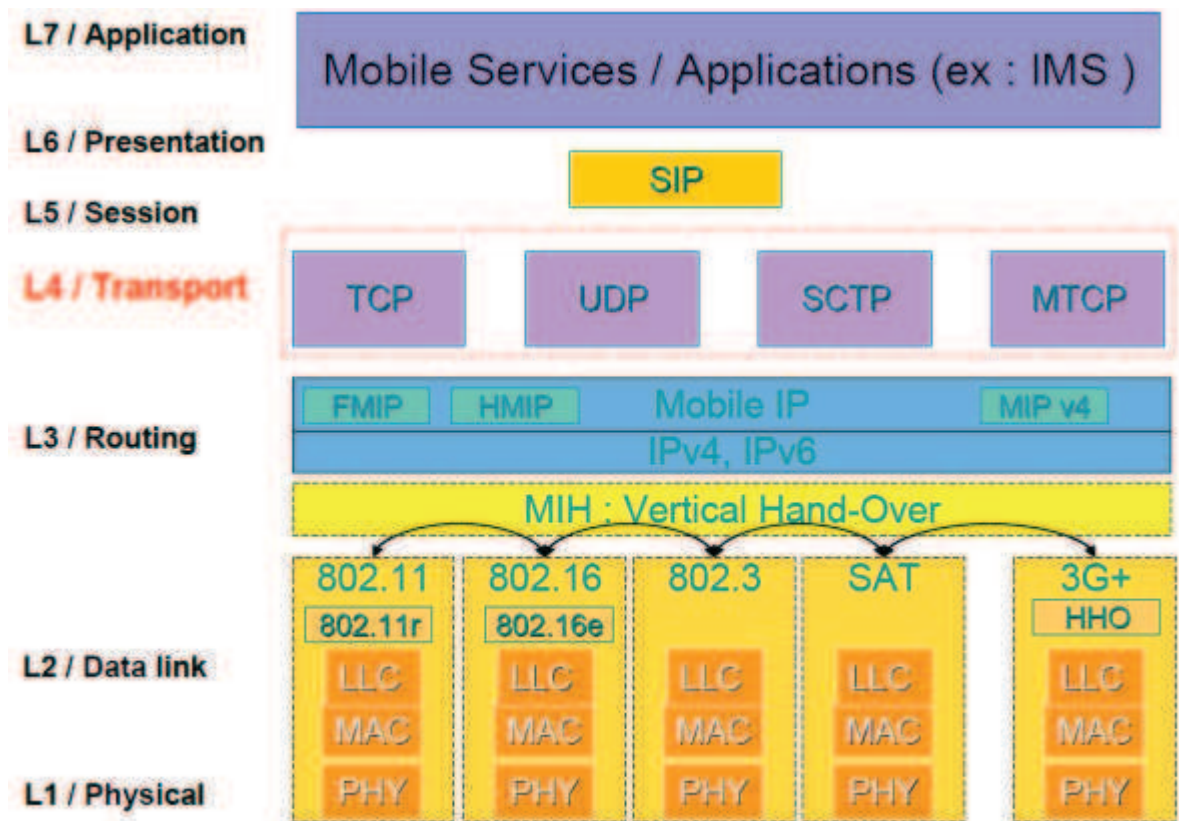


Figure 14 : La mobilité dans la pile de protocole [36]

La gestion de la mobilité peut donc être abordée de plusieurs angles de vue, mais peut essentiellement être classée en trois catégories.

- **La mobilité au niveau de la couche liaison [36]:** elle traite l'attachement physique et logique au point d'accès, d'une manière dépendante au support d'accès (l'opérateur est sous le contrôle du réseau physique et une partie du protocole MAC/PHY). Dans ce cas, on peut citer :
 - **MIH 802.21 :** Dans le but de maintenir une connexion ininterrompue lors d'une transition d'un réseau à un autre, IEEE 802.21 [39] définit ce passage indépendamment du support média (MIH media independant handover). Cette fonction opérant entre la couche 2 et la couche 3 du modèle OSI permet une mobilité entre les réseaux hétérogènes. En autorisant les équipements clients et réseaux à travailler ensemble durant ces transitions, la norme 802.21 offre des mécanismes afin d'optimiser le passage entre des réseaux de types Wifi, WiMax et mobiles. Ce standard offre une intelligence au niveau de la couche liaison de données ainsi que d'autres informations réseau adaptées dans les couches supérieures afin d'optimiser le passage (handover) entre des réseaux hétérogènes.
- **La mobilité au niveau de la couche réseau :** elle traite l'attachement à l'adresse IP (le routeur d'accès permettant le routage Internet) et l'adressage à la couche 3, où à la fois le mobile et le réseau sont concernés par les opérations.
 - **Mobile IPv6 :** C'est un protocole standard de communications de l'IETF, conçu pour permettre aux utilisateurs de se déplacer depuis un réseau IP vers un autre réseau IP

tout en maintenant les connexions actives et la même adresse IP. La mobilité IP pour IPv4 est décrite dans RFC 5944, avec des extensions décrites dans RFC 4721. La mobilité IPv6, l'implémentation de la mobilité IP pour IPv6, est définie dans RFC 3775 [40]. Mobile IPv6 permet à un nœud de changer de réseau tout en gardant d'une façon transparente ses connexions déjà établies. Cela est réalisé par le fait que chaque nœud est toujours identifié par son adresse IP d'origine. Un nœud spécial sur le réseau d'origine appelé Home Agent intercepte ainsi les paquets destinés au nœud mobile et les lui renvoie. Ce mécanisme est complètement transparent aux couches supérieures (ainsi un nœud conserve son adresse IP d'origine dans les entrées DNS par exemple). Le Home Agent peut rediriger les paquets vers le nœud car il connaît son emplacement actuel représenté par une adresse IP intitulée care-of-address. En effet le nœud informe par un message spécial son Home Agent qu'il a changé d'emplacement en lui fournissant cette nouvelle adresse.

- **HMIPv6** : Hierarchical Mobile IP, est un protocole expérimental développé par l'INRIA et Ericsson Research [41] dont le but est de réduire le taux de signalisation entre un nœud mobile, son correspondant et le Home Agent dans un contexte de micromobilité, ainsi que la prise en compte rapide d'un changement du point de rattachement du MN.
- **La mobilité haut niveau** : elle est entièrement gérée par l'hôte mobile lui-même, souvent en relation avec l'application concernée, indépendamment de la technologie du média et le réseau dans lequel il est attaché.

La différence entre ces méthodes est basée sur la distinction de la préoccupation abordée dans la problématique de la mobilité. Ainsi, chaque solution ne constitue pas une solution monolithique, utilisable indépendamment. Au contraire, elles peuvent être complémentaires pour répondre à tous les problèmes de l'application/mobilité de service.

4.3 Conclusion de mobilité

Nous avons présenté un bref aperçu sur les scénarios d'hybridation satellite/terrestre, leurs faisabilités et l'interaction qui peut être faite entre les deux réseaux. Les deux premiers scénarios cachent l'hétérogénéité du réseau avec une couche 2 dédiée et transparente pour les couches supérieures. Dans le scénario "intégration fortement couplée", le terminal peut être informé sur la présence de l'accès satellite ; dans l'architecture relai, l'hétérogénéité est intégrée au réseau. Le scénario faiblement couplé a besoin d'un troisième protocole pour assurer le changement de réseau, mobile IP [42] peut être une bonne solution, cependant les caractéristiques du réseau seront aussi cachées.

En outre, l'hétérogénéité est un problème pour la couche de transport. Un protocole de transport transparent et qui ne connaît pas les réseaux sous-jacents devrait être une meilleure solution que la connaissance du réseau sous-jacent car cela peut s'avérer une hypothèse hasardeuse.

Enfin, les principaux problèmes rencontrés lors d'un changement de réseau se situent à différents niveaux et qui sont :

- Transport

- La difficulté à exploiter la bande passante disponible (dans le cas de mobilité ascendante : vers un réseau plus large).
- Les pertes engendrées par la mobilité descendante vu la réduction de la bande passante.
- Les pertes possibles par le déclenchement des timers du fait du changement de délai.
- Les problèmes liés généralement aux contrôles de congestion et d'admission (derniers paramètres enregistrés dans l'ancien réseau).
- Les retransmissions inutiles dans certains cas.
- Application
 - La déconnexion de certaines applications qui contiennent des timers.
 - L'impossibilité de joindre le mobile à son adresse (dans le cas d'une affectation d'adresse IP), impliquant une reconnexion.
- La gestion de la QoS
 - La découverte de nouveaux services dans le nouveau réseau.
 - L'incompatibilité de la politique de gestion de la Qualité de Service de l'ancien réseau avec le nouveau.

5 Conclusion

Ce chapitre nous a permis de faire le point sur les technologies de réseaux sans fil, les différentes manières de gérer la Qualité de Service ainsi qu'une présentation des mécanismes pour la gestion de la mobilité entre ces réseaux.

Les réseaux par satellite sont une bonne alternative aux réseaux terrestres ou sans fil, lourds à déployer, dans les zones rurales et/ou non déjà couvertes. Bien que cette solution ait beaucoup d'avantages tels que la grande zone de couverture, la facilité et la rapidité d'installation, elle présente néanmoins quelques inconvénients comme le coût de la technologie, la limitation de ressources et le long délai de propagation. Le meilleur moyen de tirer profit de cette technologie et de l'utiliser en tant que solution complémentaire aux autres solutions sans fil (WiFi, WiMax et 4G). De plus, le contexte actuel de l'Internet est présenté comme un réseau hybride indissociable contenant tous les types de réseaux.

Une solution telle que S-WiMax (Satellite WiMax) peut être envisagée pour l'intégration des réseaux satellite et terrestre, avec une seule interface d'accès de type WiMax. Toutefois, ces solutions ont encore de nombreux problèmes techniques à résoudre avant leur déploiement. Dans notre étude, nous nous focalisons sur un système à deux interfaces distinctes pour accéder aux réseaux concernés

par l'hybridation. Quoiqu'il en soit, la problématique de l'hybridation, vue de la couche transport, est similaire, que le problème de la mobilité soit adressé depuis les couches 1, 2 ou 3.

La gestion de la Qualité de Service a été présentée par la suite comme un moyen d'optimiser l'utilisation de ressources pour minimiser les coûts et donner des accès paritaires selon les profils d'utilisateurs donnés. Une bonne gestion de la QoS est importante dans ce type de scénario.

Ce chapitre est clôturé par une présentation des mécanismes de gestion de la mobilité entre les réseaux satellite et les réseaux d'accès principaux. Des scénarios de mobilité ont été présentés pour dégager la faisabilité, les avantages et les inconvénients de chaque scénario.

Chapitre 2 : Evaluation de la QoS dans les réseaux satellite

1 Introduction

Le développement des systèmes de communication connaît depuis quelques années une vraie révolution. Cela a entraîné l'introduction de nouvelles architectures et de nouveaux protocoles dans le monde Internet. Bien que les réseaux terrestres présentent beaucoup d'avantages, ils restent difficiles à implanter dans les zones rurales ou dans des scénarios militaires ou de catastrophes naturelles où le déploiement doit se faire rapidement dans des zones parfois reculées. Dans ce cas, les réseaux satellite représentent une solution alternative. Ainsi, les fournisseurs d'accès s'intéressent d'avantage à ce genre d'accès et proposent de plus en plus des solutions d'accès satellitaire.

Une des étapes cruciales de la conception de ces systèmes réside dans leur dimensionnement. Pour cela il faut à priori être capable d'évaluer le service qui sera rendu. L'évaluation de ces systèmes, qui désigne l'ensemble des tests réalisés pour percevoir leur bon fonctionnement, est une bonne approche pour analyser leur comportement. L'évaluation et le paramétrage de leurs architectures, en terme de Qualité de Service, est une tâche plus que nécessaire. La variété des utilisateurs ainsi que leurs besoins rendent cette tâche encore plus cruciale et difficile. Le but étant de s'assurer que ces architectures soient en mesure de répondre aux différents besoins spécifiques des clients.

La QoS rendue par les réseaux satellite doit donc être évaluée pour estimer le bon fonctionnement de ces réseaux. L'évaluation peut être réalisée à différents niveaux, par plusieurs types de tests et pour différents protocoles et scénarios.

Il existe aujourd'hui principalement deux manières de réaliser une évaluation du système satellite.

- Evaluation des performances des couches basses : Cela concerne généralement la couche physique et liaison. Nous pouvons citer quelques tests utilisés pour cette évaluation (Performance des couches physiques et impact du canal satellitaire [18]) :
 - Mesurer l'impact des non-linéarités satellitaires sans et avec les techniques de pré-compensation.
 - Mesurer l'impact des imperfections des filtres IMUX (démultiplexeur d'entrée) et OMUX (multiplexeur de sortie) (amplitude ripple, group delay)
 - Mesurer l'impact du bruit de phase, avec et sans symboles pilotes
 - Mesurer l'impact du décalage en fréquence sur la performance du lien
- Evaluation des performances applicatives/protocollaires et impact de la QoS : tests de la Qualité de Service applicative suivant les recommandations ITU. Nous pouvons citer quelques tests tels que la mesure du délai de bout en bout ou l'évaluation de l'aptitude à gérer la bande passante, ou l'impact du satellite sur le comportement des profils d'applications...etc.

A ce jour, il n'existe aucun processus permettant à la fois d'évaluer la capacité des architectures satellite et de les paramétrer pour supporter les divers profils utilisateurs. Nous nous intéressons

spécialement aux paramétrages des couches les plus hautes (application et IP), puisque les tests réalisés par la communauté physique sont complexes, coûteux et n'offrent qu'une vision partielle des résultats que nous souhaitons compléter par cette étude.

Il existe à ce jour trois possibilités pour l'évaluation des couches hautes d'un système satellite :

- L'évaluation par simulation : consiste à l'utilisation de modèles mathématiques pour l'évaluation et la validation d'un protocole, d'une architecture ou d'un système.
- L'évaluation par émulation : consiste à imiter le comportement de systèmes réels en substituant un élément matériel par un élément souvent plus simple et disposant d'une interface de mesure et pilotage.
- L'évaluation avec un système réel : consiste à l'évaluation d'un système satellite réel complet (antenne, terminal, gateway /NCC...etc.).

Nous allons présenter plus en détail ces trois approches d'évaluation en précisant les avantages et les inconvénients de chacune d'entre elles.

2 Méthodes d'évaluation

2.1 La simulation

La simulation est la méthode la plus simple pour évaluer un réseau, un mécanisme ou un algorithme au sein d'un réseau spécifique. Il existe de nombreux simulateurs réseaux à plusieurs niveaux (couches réseaux). Nous nous intéresserons plus particulièrement à la simulation des couches hautes (IP et au-dessus). Elle peut être réalisée par un certain nombre de simulateurs tels que Qualnet, Glomosim, Opnet ou NS-2. Notre choix se tourne vers NS-2 car il est libre (open source), multiplateformes, et grandement répandu dans la recherche. De plus, sa prise en main rapide le différencie des autres simulateurs comme Opnet.

Dans la simulation, des modèles abstraits établis au préalable sont utilisés pour rejouer un scénario précis, un profil d'application/protocole et un modèle de simulation de réseau spécifique. Par exemple, on peut simuler un transfert FTP (Protocole TCP) sur la voie retour d'un réseau Satellite DVB-RCS en utilisant le modèle MFTDMA-DAMA de Secchi Raffaello [43]. L'avantage est la liberté d'effectuer des tests, de manière logicielle et de rejouer exactement le même test plusieurs fois afin d'établir des analyses comparatives et des séries statistiques.

Bien que cette méthode présente beaucoup d'avantages, en particulier pour les systèmes satellites qui nécessitent de grosses infrastructures, elle reste moins réaliste à cause de l'absence de modèle adéquat du lien physique.

Dans cette partie, nous allons décrire le simulateur NS-2, son fonctionnement ainsi qu'un exemple de modèle célèbre dans le contexte d'un réseau d'accès satellite.

2.1.1 Introduction à NS-2

Network Simulator (NS) [44] est un simulateur à événements discrets pour les réseaux informatiques. NS est né d'un projet de la Defense Advanced Research Projects Agency (DARPA), une agence du Department of Defense (DoD) des États-Unis. A sa naissance en 1989, il était considéré comme une variante de « Real Network Simulator ». Depuis, le logiciel a évolué grâce aux différents groupes qui

participent à son développement mais aussi aux apports des chercheurs qui l'exploitent dans leurs travaux de recherche. En plus de son efficacité, NS est en accès libre et peut être exécuté sur tout type de système d'exploitation. Il est donc rapidement devenu une référence dans le domaine.

Différentes versions sont disponibles. A la fin 2008, la version 3 du logiciel est parue, elle est le fruit de modifications récentes et d'une nouvelle architecture mêlant C++ et Python.

Notons que les protocoles fournis avec la version de base ont été testés par les concepteurs dans un nombre important de configurations. Néanmoins, toutes les configurations n'ont pu être testées. Cette mise en garde est donnée dans les premières pages du manuel de NS, et sur la page d'accueil du site officiel. Il faut donc garder un œil critique sur les résultats obtenus. Ainsi, il faut vérifier que ceux-ci sont cohérents par rapport aux résultats obtenus avec d'autres méthodes d'évaluation (l'émulation ou les tests de piles réelles).

2.1.2 Fonctionnement

NS-2 est un simulateur orienté objet écrit en langage C++ avec une interface de programmation en Tcl. Son architecture lui confère une grande modularité, et une réutilisation facile du code. Les utilisateurs ont ainsi la possibilité de modéliser des réseaux filaires ou sans-fil, et d'effectuer des simulations de trafics avec de nombreux protocoles, topologies et applications :

- Génération de trafic selon des lois générales (PARRETO, poisson...etc.), CBR, FTP, telnet, etc.
- Couche Transport TCP, UDP, SRM, SCTP, PLM.
- Couche Réseaux IP, routage dans les réseaux ad hoc (aodv, dsr, dsdv, tora, amodv), routage dans les réseaux filaires (link state, distance vector), les réseaux multicast, IntServ, DiffServ
- Couche MAC CSMA, CDMA, 802, X, Token ring, MPLS, liens satellite, etc.

Le simulateur supporte une hiérarchie en classes C++, et une hiérarchie semblable dans l'interface Otcl. Les deux hiérarchies sont reliées entre elles et, du point de vue utilisateur, les classes correspondent l'une à l'autre. Le lien entre les deux est fait avec un objet Tcl, dit TclObject en anglais. Lorsqu'un utilisateur crée un objet au travers de l'interface, celui-ci est instancié dans l'interface Tcl, puis relié à un objet correspondant dans les structures C++.

L'utilisation de deux langages de programmation permet d'avoir les avantages du C++ et du Tcl, sans en avoir forcément les inconvénients. En effet, le C++ est un code rapide à exécuter, mais qui peut être long à modifier (compilation, débogage). Il est donc bien adapté à la modélisation détaillée de protocoles, gérant facilement les paquets comme les variables. Le langage Otcl peut être modifié plus facilement (pas de compilation). Par contre, son exécution est plus longue, mais il est mieux adapté à la configuration des simulations, à la création des liens entre les objets, et au réglage des paramètres des différents protocoles.

En ayant une approche simpliste, on peut comparer l'architecture réseau classique avec l'architecture interne des modèles utilisés par NS (voir Figure 15). Il faut néanmoins garder en mémoire que les modifications faites par les utilisateurs sont correctes d'un point de vue réseau si et seulement si celles-ci ont été faites judicieusement et aux bons emplacements.

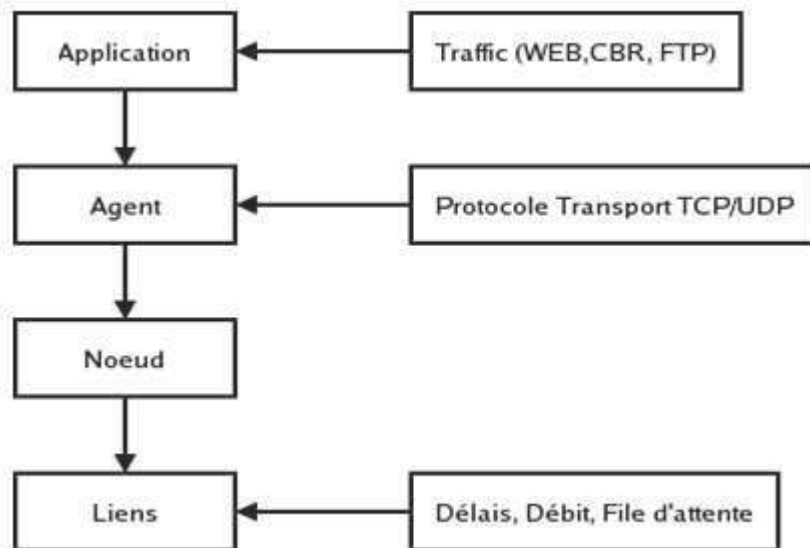


Figure 15 : Correspondance entre les composantes NS et les couches réseau [44]

Comme on peut le voir sur la Figure 15, les composants primaires d'un modèle sous NS sont les nœuds, les liens, les agents et les applications. Les différentes variables et méthodes relatives à ces composants sont définies dans des classes C++, et vont être déclarées dans le script écrit par l'utilisateur ou dans les procédures invoquées depuis ce script.

L'architecture d'un nœud est inspirée de la réalité. Lorsqu'un paquet est généré par une application, celle-ci l'envoie à l'entrée du nœud via un agent. Les paquets arrivant à l'entrée d'un nœud passent d'abord par un démultiplexeur (address classifier) qui va diriger le paquet en fonction de son adresse de destination. Si celui-ci n'est pas destiné au nœud où il se situe, il est envoyé vers un des liens dans un réseau maillé, ou vers un agent de routage dans un réseau plus complexe. Ce dernier le dirigera vers le bon lien. Si le paquet est destiné au nœud où il se situe, il passe par un second démultiplexeur (port classifier) qui va le diriger vers son port de destination qui contient un agent.

En ce qui concerne la simulation satellite, on peut ajouter tout simplement un délai au lien (i.e. 250 ms) et limiter la bande passante, sinon utiliser directement un modèle existant. Dans cette optique, plusieurs modèles NS-2 célèbres ont fait leurs preuves. Parmi ces modèles, on peut citer celui de Raffaello Secchi MF/TDMA-DAMA, qui simule l'allocation dynamique DAMA.

2.1.3 Le modèle MF/TDMA-DAMA

MF/TDMA-DAMA [43] [45] est un ensemble de modules NS-2, pour la simulation détaillée du DVB-RCS. Le Framework vise à fournir un bon niveau de détail de la couche MAC du DVB-RCS. La signalisation, à partir de requêtes de capacité et le plan d'allocation de ressources TBTP (Terminal Burst Time Plan) sont conformes au standard. Cela permet une simulation au niveau paquet dans le DVB-RCS et aux environnements hétérogènes contenant un réseau DVB-RCS.

MFTDMA-DAMA vise à être une aide importante pour le problème de dimensionnement de réseau (par exemple, la détermination du nombre de transpondeurs et leur composition pour satisfaire un SLA spécifique). Il peut aussi être utilisé pour identifier des problèmes au niveau 2, tels que l'efficacité d'encapsulation dans certains types d'applications ((Trafic Internet, VoIP...etc.).

Bien que NS-2 présente une grande flexibilité et une facilité d'utilisation, ainsi que la possibilité de simuler presque tous les types de réseau, flux et profils d'application, il présente quelques limitations, telles que l'incapacité à fragmenter les paquets. En effet, il traite ceux-ci comme des variables en les numérotant et en y accédant via des pointeurs, le réveil des composants se faisant via des handlers. Ceci va poser un problème lorsque nous voudrions modéliser le passage des protocoles ATM ou MPE vers IP. Néanmoins, la facilité que donne NS pour manipuler les paquets (changement dans l'en-tête, modification de la taille ou du type) va pallier ce défaut et ne portera pas trop préjudice à la simulation.

2.2 L'émulation

Après avoir présenté brièvement la simulation, nous introduisons l'émulation, qui est un autre moyen permettant l'évaluation des performances d'un réseau.

Le terme « émuler » signifie littéralement la recherche à imiter. Une émulation est donc une imitation du comportement physique d'un système réel. Contrairement à la simulation qui utilise une reproduction d'un modèle abstrait, créé par extrapolation de variables souvent inconnues par la simulation, l'émulation reproduit le comportement d'un modèle dont toutes les variables sont connues. Autant que pour les réseaux, l'émulation s'est répandue après son apparition dans plusieurs domaines, tels que l'émulation de machines virtuelles et l'émulation de comportement humain...etc.

L'émulation a l'avantage de pouvoir interconnecter des systèmes réels et donc d'être fidèle sur certains points. Il est par exemple possible d'émuler le réseau satellite et d'y interconnecter des applications réelles.

Ns-2 a aussi une fonction d'émulation, à savoir la capacité d'introduire le simulateur dans un réseau réel et simuler le réseau désiré entre les demandes réelles en temps réel. Des méthodes et des extensions à NS-2 ont été développées pour pouvoir combiner la simulation et l'émulation de réseaux. Par exemple, NS-2 peut émuler un réseau IEEE 802.11 entre les applications réelles. En outre, les extensions sont génériques et peuvent être appliquées à n'importe quel réseau pris en charge dans le NS-2.

Dans notre étude, nous avons fait le choix de ne pas se focaliser sur les fonctionnalités d'émulation de NS-2 et se concentrer sur deux émulateurs dédiés.

Nous allons présenter deux systèmes qui émulent le comportement d'un réseau satellite : La plateforme PLATINE devenue par la suite OpenSAND et l'émulateur SATEM. Les deux systèmes développés dans le cadre de projets de recherche auxquels nous avons participé ont été utilisés pour l'évaluation présentée dans ce manuscrit.

2.2.1 OpenSAND (ex. Platine)

Platine (PLATe-forme d'INGénierIE accès réseau) [46], devenue par la suite OpenSAND [47] est une plateforme d'émulation, initiée dans le cadre du projet SATIP6 [48] et utilisée aussi dans le projet SATSIX [4]. Elle a été créée pour émuler précisément les différentes caractéristiques d'un système DVB-S2/RCS pour quantifier l'impact des nouvelles applications IPv6 et les services liés aux NGN dans le cas où ils sont intégrés à ces réseaux. Platine se voit compatible avec l'architecture adoptée par le groupe ETSI BSM [49], et suit les recommandations SatLabs [50] en matière de QoS.

Platine fournit une modélisation d'un système DVB-RCS au niveau réseau et accès. Elle a la possibilité de s'interfacier avec des équipements externes ou avec des réseaux afin de soutenir des connexions réseau réelles.

Platine se compose de plusieurs blocs nécessaires à l'émulation DVB-S2/RCS, ces blocs sont les suivants :

- Le bloc satellite carrier : ce bloc gère l'émulation des porteuses de transmission (DVB-S2, DVB-RCS et les signalisations) en plus de l'émulation du délai satellite et les taux de pertes sur le lien.
- Le bloc DVB-RCS : afin d'appliquer les bons mécanismes de QoS, ce bloc se charge de la synchronisation et les files d'attente en fonction du bloc DAMA. Il permet aussi de remplir les trames DVB-RCS avec les paquets des couches supérieures (ATM ou MPEG2-TS) provenant du bloc encapsulation/désencapsulation.
- Le bloc DAMA : il implémente les algorithmes de DAMA responsables de l'allocation dynamique des ressources sur le réseau satellite dans la couche MAC.
- Le bloc encapsulation/désencapsulation : il utilise les mécanismes d'encapsulation AAL5 et ULE/MPE/GSE. Il est responsable des mécanismes de segmentations et réassemblage.
- Le bloc IP QoS : ce bloc est représenté par une couche intermédiaire entre le noyau du système d'exploitation Linux et Platine. Il permet de récupérer les adresses IP à travers une classification de trafic TC (Traffic Controller) puis les redirige pour être traités différemment selon leurs classes. La notion de CQD (Classful Queuing Discipline) qui est une méthode de gestion de files d'attentes, responsables de la classification des flux selon leurs classes, est à la base du bloc IP QoS. L'implémentation du CDQ est basée sur HTB (Hierarchical Token Bucket).

La Figure 16 illustre les différents blocs en interaction au niveau du système satellite à travers les trois composantes principales : ST, Satellite et la gateway/NCC.

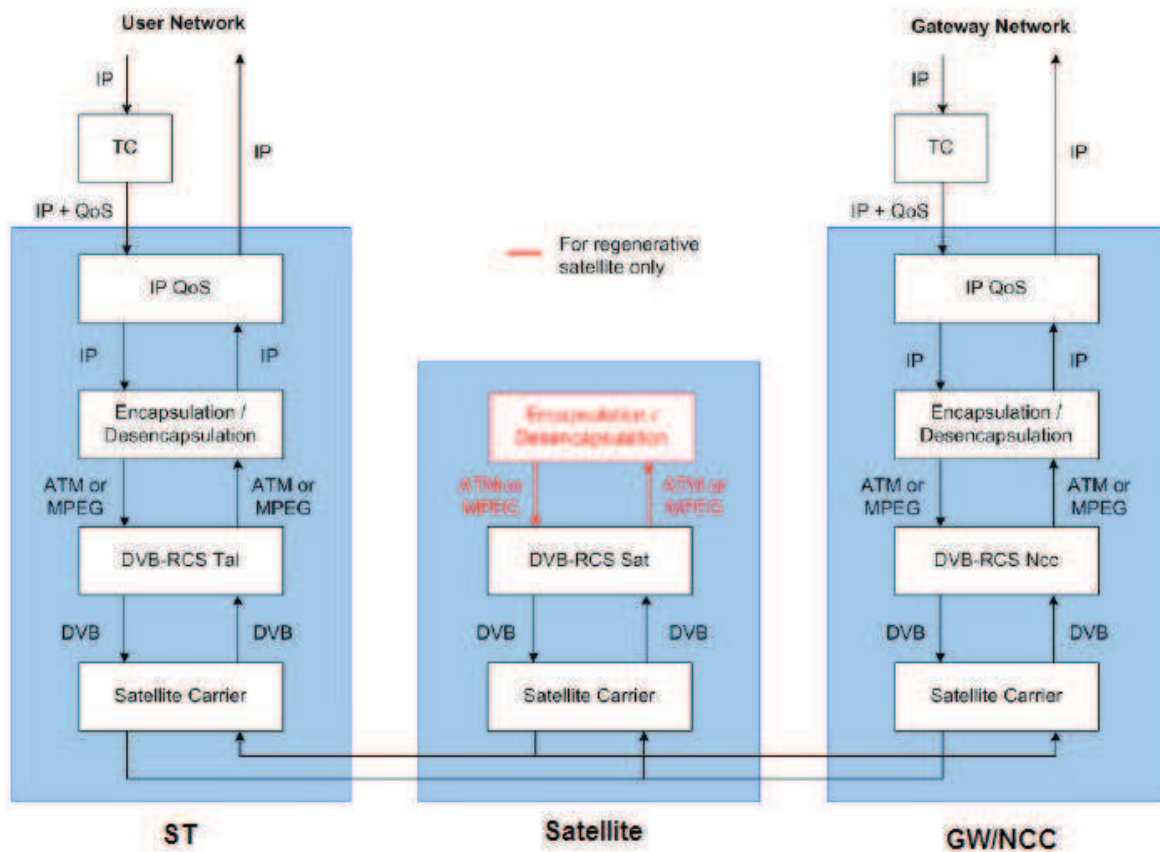


Figure 16 : Architecture PLATINE : les blocs nécessaires [47]

Plusieurs scénarios peuvent être émulés avec Platine. Chaque élément d'un réseau satellite est émulé sur un nœud dédié, comme illustré sur la Figure 17. L'émulateur Satellite (SE) permet d'émuler plusieurs types de satellite à travers une configuration spécifique de délai, de gigue, de taux de pertes de type de satellite (transparent, régénératif ou mixte) ou de type de connectivité (maillée ou étoilée). Les délais et les taux de pertes peuvent avoir des valeurs fixes ou suivre des modèles statistiques ou une distribution provenant de mesures réelles. Chaque trame DVB est encapsulée dans une trame Ethernet, cette dernière présente une grande capacité en termes de bande passante. Chaque canal satellite qui correspond à une trame et/ou à une porteuse est associé à une adresse multicast. Un spot se compose donc de quatre canaux :

- Un canal pour les données DVB-S2 sur le lien descendant.
- Un canal pour les données DVB-RCS sur lien montant.
- Deux canaux destinés à la signalisation de contrôle et de gestion (TBTP par exemple), l'un pour le lien montant et l'autre pour le descendant.

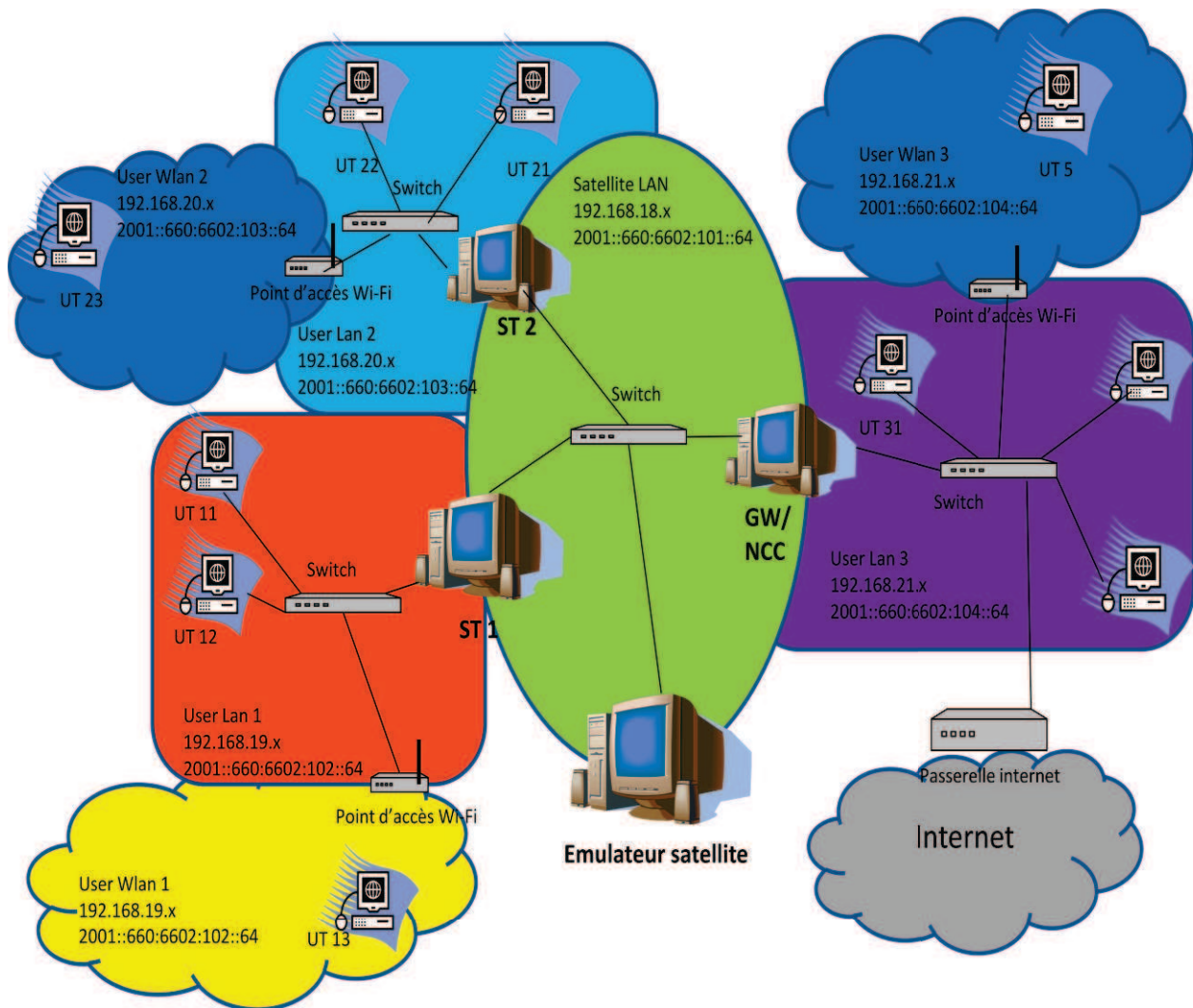


Figure 17 : Différent élément de Platine [4]

Ce genre d'émulateur permet d'avoir une grande flexibilité dans le choix de paramètres et de configurations d'un réseau satellite, ainsi qu'un grand nombre de ST. Ainsi, une interconnexion de Platine avec d'autres réseaux sous-jacent est possible.

Le réseau Ethernet, émulant le lien satellite permet alors d'interconnecter le SE avec les STs et la GW/NCC qui, eux-mêmes, sont reliés aux terminaux utilisateurs (UT) par l'intermédiaire de réseaux Ethernet et/ou d'un réseau de type WiFi par exemple. Le nombre de terminaux utilisateurs peut varier dans chaque User LAN et User WLAN selon les besoins de l'expérience et aucun système d'exploitation n'est imposé.

Par exemple, selon les scénarios choisis, plusieurs communications bidirectionnelles (HTTP, Visioconférence, FTP...etc.) entre les utilisateurs distants, derrière les STs (par exemple UT12, UT23...etc.) ou ceux derrière la GW (UT33) peuvent être effectuées.

De plus, Platine permet une configuration minutieuse de plusieurs paramètres grâce à un fichier de configuration. En effet, au niveau MAC ou IP, les débits des files AF, EF ou BE et les priorisations au niveau IP peuvent être fixés au niveau des STs et la GW. Le type de requêtes de capacités peut être choisi. Le type d'encapsulation peut aussi être choisi selon le type de satellite que nous voulons émuler. En annexe, un exemple de fichier de configuration d'un ST et une GW est présenté.

2.2.2 L'émulateur SATEM

A l'inverse de Platine, SATEM est un émulateur de niveau 3, c'est-à-dire qu'il émule un réseau en modifiant les paramètres de QoS visibles depuis la couche 3 : le délai et le taux de pertes.

SATEM a été développé par ASTRIUM (initialement conçu pour les liens DVB-S2) et s'adapte parfaitement pour nos travaux sur la couche transport.

Présentation de SATEM

Cet émulateur a été développé en 2008 par ASTRIUM et depuis, a souvent été utilisé dans différents projets. Il a été conçu pour être portable et facile d'utilisation. C'est un PC classique jouant le rôle d'un routeur, avec une interface Ethernet de gestion et deux interfaces pour les flux traversant le réseau satellite émulé. Cet émulateur offre la possibilité d'émuler des comportements très similaires aux réseaux satellite. En effet, un centre de contrôle évolué est disponible via LabView [51]. Il permet une gestion du fading du canal, la sélection du modcod et le réglage de nombreux autres paramètres spécifiques. Dans notre étude, nous connectons SATEM à une station de gestion sous Windows XP. Deux autres PC sont utilisés des deux côtés de SATEM pour générer et recevoir le trafic. La Figure 18 présente l'interface de la station de gestion Labview. SATEM est basé sur les files TC (Traffic Controller), qui sont très flexibles.

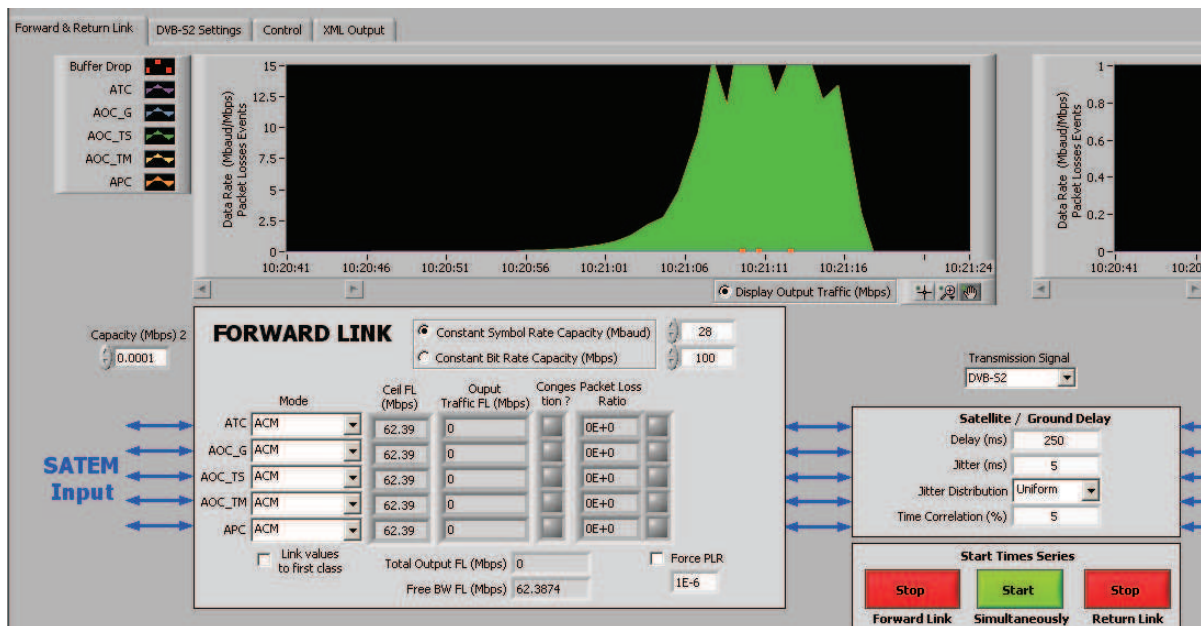


Figure 18 : Interface de gestion SATEM

Traffic Control

Les deux interfaces de SATEM sont basées sur le même mécanisme : Linux Traffic Control [52]. C'est une fonction native qui permet d'ajouter des files d'attente avec des fonctionnalités modifiables dans les systèmes Linux. TC peut réguler le flux avec un filtrage par numéro de port ou champ ToS (Type of Service) avec les capacités assurées/autorisées. Le module HTB a été choisi pour la gestion des files avec un traitement sur le débit selon la file choisie. Un autre module, nommé NETEM [53] est utilisé dans SATEM, qui n'a pas le rôle de limitation de trafic mais plutôt la mise en forme des paquets pour

reproduire le comportement du réseau satellite. C'est un outil puissant qui permet d'ajouter du délai, des paquets perdus/erronés, faire de la duplication ou dés-ordonnement des paquets.

Dans TC, la file d'attente est placée directement sur une interface mais certains modules autorisent une ou plusieurs files à avoir des liens avec d'autres modules en cascade. Le système est un arbre avec plusieurs feuilles qui jouent le rôle de files d'attente, spécifiques à un profil donné.

La Figure 19 présente une vue globale de la classification, la mise en forme et la mise en route des paquets à travers les différentes files de l'émulateur.

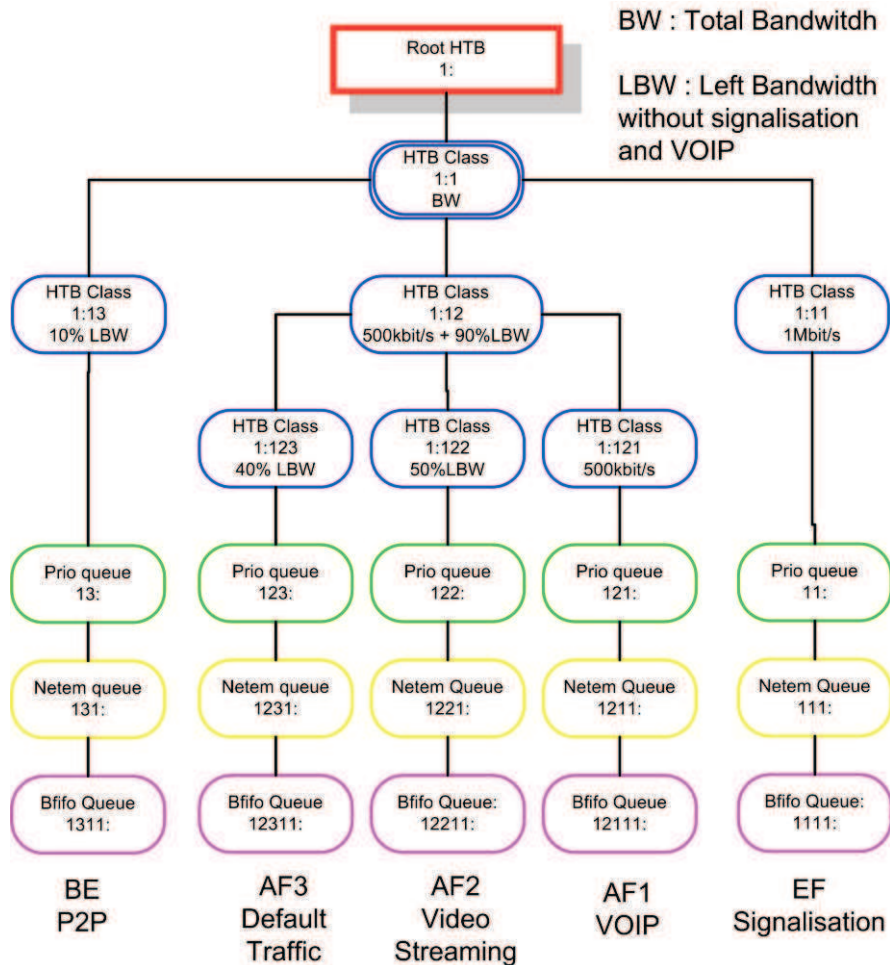


Figure 19 : façonnage du trafic en utilisant TC avec les classes DiffServ [36]

Dans la Figure 19, nous présentons l'architecture QoS des files au sein de SATEM. Par exemple, si on choisi de diriger un flux par la file EF, la classe HTB permet de mettre en forme la bande passante selon le débit choisit sur l'interface. Par la suite NETEM, en s'imbriquant dans la classe HTB, permet d'injecter un délai supplémentaire pour une configuration. Bfifo est une simple file d'attente en FIFO en octets pour la classification des paquets.

Afin de mettre en œuvre des réseaux hétérogènes, les files d'attente sont placées en parallèle avec leurs propres paramètres et un filtre ToS pour les paquets entrants est utilisé pour décider du chemin que le flux empruntera.

La mobilité ou le handover sont mis en œuvre par la modification du champ type de service (Type of Service ToS) du paquet transmis. Dans Linux, cela est possible en utilisant d'Iptables [54]. Une ligne de commande Iptables peut modifier le marquage du champ ToS d'un paquet. Ce paquet sera donc routé vers un autre réseau sans altérer les autres paquets envoyés. Le processus du handover dans SATEM est illustré dans les Figure 20 et Figure 21.

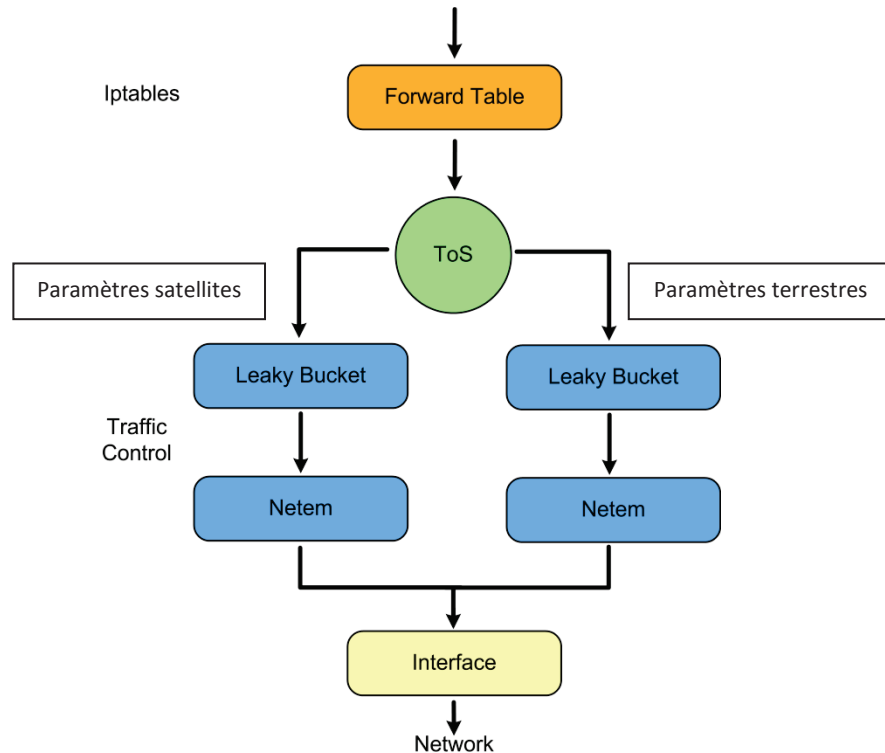


Figure 20 : Handover sur des réseaux hétérogènes en utilisant TC et Iptables [36]

Pour finir, l'émulateur doit émuler dans les deux directions. Dans notre système, une interface est liée à l'envoi et une autre à la réception, et comme TC a des fonctions limitées quand il est appliqué aux trafics entrants, le choix a été fait pour utiliser le côté émetteur pour émuler le lien retour et le côté récepteur pour assurer le lien aller.

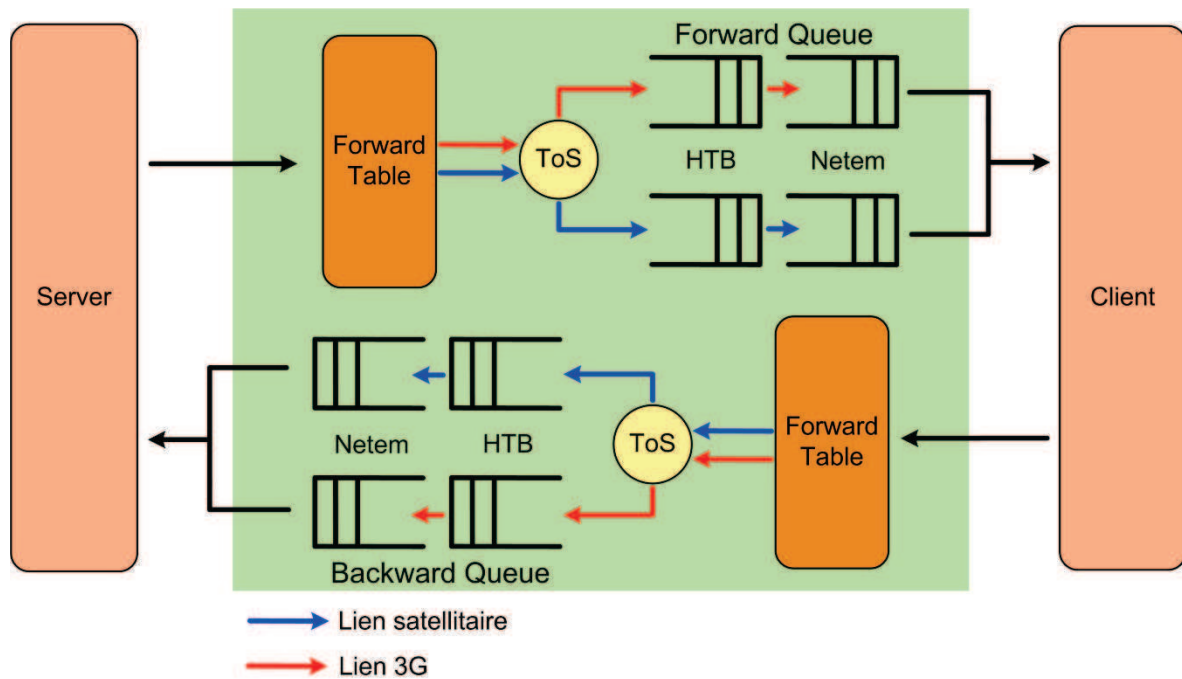


Figure 21 : système d'émulation au niveau paquet [36]

La simulation et l'émulation sont des moyens puissants pour évaluer et tester des réseaux ainsi qu'analyser les performances protocolaires et applicatives de ces réseaux. De plus, elles minimisent le matériel et le coût de la caractérisation, mais permettent juste une vision globale sans pour autant avoir le comportement de chaque élément du réseau. Le meilleur moyen pour une évaluation est l'évaluation d'un système réel puisque cela permet d'avoir des résultats reflétant un comportement réel, ainsi de voir les vrais avantages et inconvénients du système.

Dans cette optique, nous avons eu la chance de disposer de deux liaisons satellites pendant un mois. Chacune avec une technologie différente. Nous avons ainsi pu comparer les différents résultats des simulations et émulations.

2.3 Système réel

2.3.1 Architecture OURSES

La plateforme OURSES a été mise en œuvre dans le cadre du projet OURSES [18], Le projet OURSES est un des projets labellisés en 2006 par le Pôle de Compétitivité Mondial « Aéronautique, Espace et Systèmes Embarqués » (Aerospace Valley) dans le Domaine d'Activité Stratégique de la Navigation, Positionnement et Télécommunications (NPT). OURSES est cofinancé, à hauteur de 1.6 M€ pour un budget total de 3,7 M€, par la Direction Générale des Entreprises (DGE) du Ministère de l'Industrie et des Finances dans le cadre du Fond Unique mis en place pour le financement des projets des Pôles de Compétitivité français.

OURSES a pour objectif de mieux positionner les télécommunications par satellite sur le plan de l'efficacité technique et économique lorsque celles-ci sont utilisées en complément des technologies terrestres pour garantir le déploiement et l'usage de télé-services dans des zones ne disposant pas encore des moyens d'accès haut débit.

Le projet est conduit par EADS Astrium avec les contributions des laboratoires LAAS-CNRS, IMS (Bordeaux), ENST-GET et TésA, du CNES et des industriels Thales Alenia Space, Medessat et APX-Synstar.

Notre tâche principale consiste à effectuer une campagne de tests en laboratoire et par satellite en bande Ka avec système commercial basé sur du DVB-S2 et ACM sur la voie aller et DVB-RCS sur la voie retour, pour en évaluer la QoS rendue. C'est la première fois que sera testé un système DVB-S2 en bande Ka.

L'architecture de la plateforme est représentée dans la Figure 22.

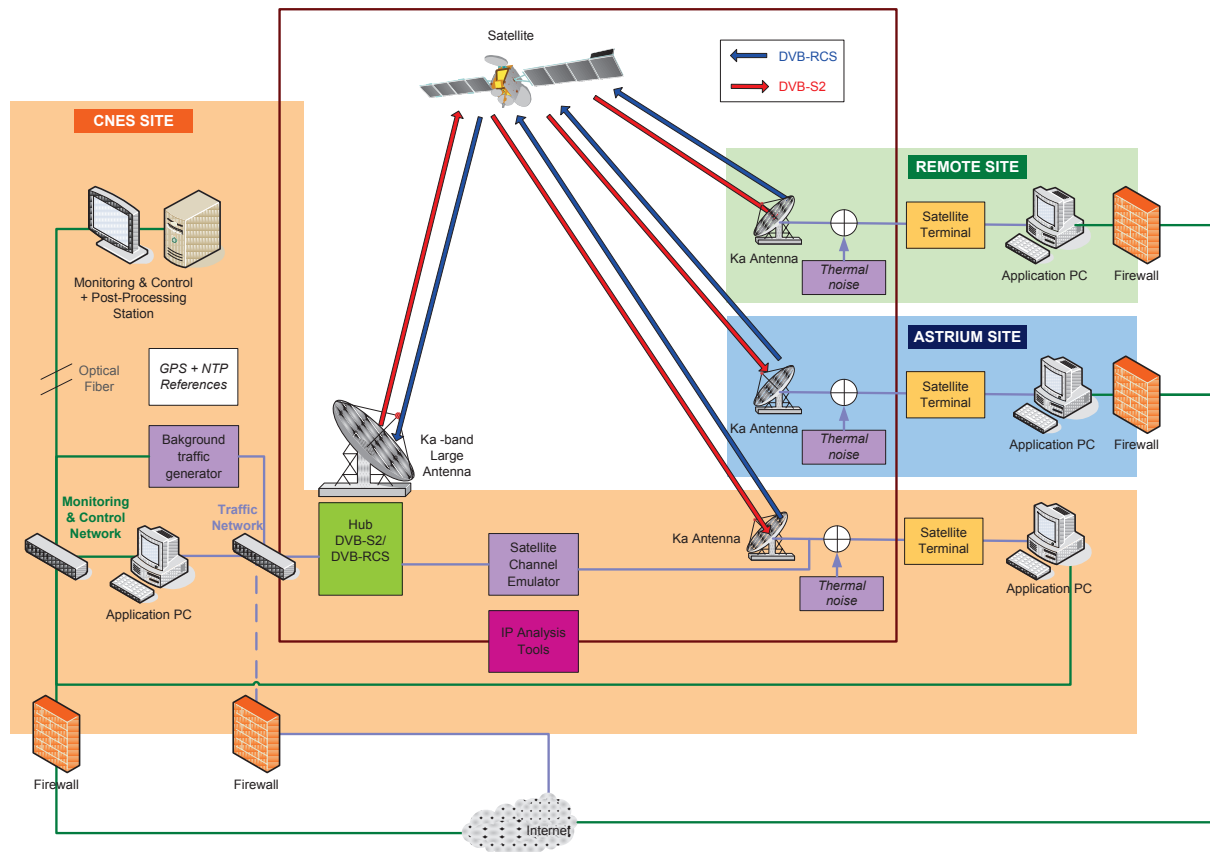


Figure 22 : Architecture de la plateforme de test OURSES [18]

L'objectif est d'effectuer une campagne de tests en bande Ka avec un système commercial basé sur du DVB-S2 et ACM sur la voie aller et DVB-RCS sur la voie retour. Pour cela, nous allons tester les performances d'applications par satellite avec DVB-S2 et ACM, avec fading réel ou émulé. Nous procédons aussi à des tests basiques sur la voie retour DVB-RCS ainsi que des tests de QoS avec des priorités différentes afin de tester leur comportement.

Nous allons présenter les différents composants de la plateforme et détailler spécialement le terminal satellite utilisé avec les fonctionnalités principales qu'il fournit.

- **Description de l'architecture et équipements utilisés**

La plateforme de test présentée dans la Figure 22 comporte les équipements suivants :

- 3 Terminaux satellite STM SatLink 1000 : un localisé au CNES, un localisé chez Astrium, et un localisé au LAAS.
- Au CNES, le banc de test (la gateway + un terminal + les équipements de test)
- Un émulateur de canal satellitaire, pour les tests en laboratoire (Doppler, bruit thermique, bruit de phase, fading, non-linéarités satellitaires ...) qui sont réalisés par le CNES et Astrium.
- Une station de supervision (M&C et post-processing) dirigée par Astrium.
- 3 routeurs VPN gérés par Astrium.
- Des stations en tant qu'utilisateurs finaux et serveur (derrière les STs et la gateway) tournant sous Microsoft Windows XP Pro, Linux Fedora et UBUNTU ont aussi été utilisés.
- Le satellite d'accès est l'Astra 1L en bande Ka.

Les tests ont été réalisés dans le banc de test du CNES. Les équipements essentiels utilisés sont le ST et la GW. Dans ce qui suit, nous allons présenter le terminal STM SatLink 1000 ainsi qu'un bref aperçu sur la Gateway Alcatel 9780 DVB-RCS.

Le STM SatLink 1000

Le terminal satellite STM Satlink 1000 [55] a été choisi pour la plateforme de test. Il fait partie des terminaux DVB-RCS certifié VSAT, supportant la norme DVB-S2 et DVB-S sur la voie aller. Il possède les fonctionnalités de gestion de la QoS, l'accélération du trafic, les VPNs et bien d'autres fonctionnalités logicielles nécessaires à la différenciation interactive des classes de données, voix et visioconférence, ainsi que des applications multicast, et surtout le support de la norme DVB-S2 implémentant l'ACM.

Le DVB-S2 apporte des améliorations sur l'adaptation de la modulation et le codage selon les conditions du lien. Dans ce sens, l'ACM sur la voie aller et l'adaptative FEC augmentent sensiblement l'efficacité de la bande passante et améliorent les marges de liaison pour atténuer les perturbations liées à la pluie. Le STM Satlink 1000 est équipé d'une carte plug-in SatLink 100 qui supporte deux types de modulations QPSK et le 8PSK. Concernant le codage, ce terminal supporte 8 différents taux de FEC en mode DVB-S2 qui sont : 1/2, 3/5, 2/3, 3/4, 4/5, 5/6, 8/9 et 9/10.

Pour la gestion de la QoS, le STM Satlink 1000 assure la différenciation de service sur la voie retour basé sur l'architecture DiffServ, et peut fonctionner dans des réseaux offrant les performances Per Hop Behaviour (PHB) comme définit dans l'architecture DiffServ. La QoS sur la voie retour est traitée principalement par le hub DVB-RCS.

Les classes PHBs (Per-Hop-Behaviour) supportées sont :

- Le Best Effort (BE) PHB
- Critical Data (CD) PHB
- Real Time Video Conferencing (RT-ViC) groupe PHB
- Real Time Voice over IP (RT-VoIP) groupe PHB

Le STM classe le trafic qui va être envoyé sur la voie retour dans différents groupes de QoS. Une ou plusieurs classes de services (CdS) vont donc être mappées sur une classe PHB/groupe PHB dans le réseau. Les CdS supportées dans le STM sont :

QoS Group ID	QoS Group Name	Maps to PHB
0	Best Effort	Best Effort (BE)
1	VoIP Audio	Real Time Voice over IP (RT-VoIP)
2	VoIP Signalling	Real Time Voice over IP (RT-VoIP)
3	ViC Video	Real Time Video Conferencing (RT-ViC)
4	ViC Audio	Real Time Video Conferencing (RT-ViC)
5	ViC Signalling	Real Time Video Conferencing (RT-ViC)
6	Critical Data	Critical Data (CD)

Tableau 7 : Les classes de services de QoS supportées par le STM SatLink 1000 [56]

L'acheminement de chaque paquet IP vers CdS correspondante est basé sur la classification dans les champs du paquet tel que le champ DSCP/TOS et d'autres champs TCP/IP (adresse et port source et destination, type du protocole).

La gateway :

Nous n'avons pas eu beaucoup de détails sur la gateway vu l'aspect "boîte noire" de cette dernière exigé par Alcatel pour des raisons de confidentialité industrielle. Toutefois, c'est une Alcatel 9780 DVB-RCS. Deux bandes de fréquence sont supportées, la Ku et Ka. La Alcatel 9780 fournit un accès supportant la technologie IP basée IPv4 ou/et le protocole d'accès PPP incluant le respect des CoS (Class of Service) requise par les clients. Elle supporte la norme ACM avec les modulations suivantes : QPSK, 8PSK, 16APSK.

2.3.2 Architecture E-Cab

La plateforme E-Cab utilise des liaisons Inmarsat [8]. Le système utilisé est le BGAN (Broadband Global Area Network) d'Inmarsat [9]. Comme nous l'avons dit au premier chapitre, ce système représente la dernière génération mise en service par Inmarsat. Il s'agit d'un réseau beaucoup plus orienté IP que les services précédents. Il fournit le même type de services que le GAN (Global Area Network), mais la partie Mobile Packet Data est très différente puisqu'au lieu d'un simple canal partagé de 64 kbit/s, BGAN propose plus de 400 kbit/s par canal, que l'on peut utiliser au choix en mode background (débit partagé entre les terminaux, facturation au volume) ou en mode streaming (débit garanti, facturation à la durée). Ce réseau s'inspire très fortement des services des réseaux GSM 3G. Le service BGAN est fourni uniquement par les satellites I-4, soit un total de 3 satellites géostationnaires.

La couverture mondiale d'Inmarsat I-4 est représentée dans Figure 23.



Figure 23 : Couverture mondiale des satellites Inmarsat I4 [57]

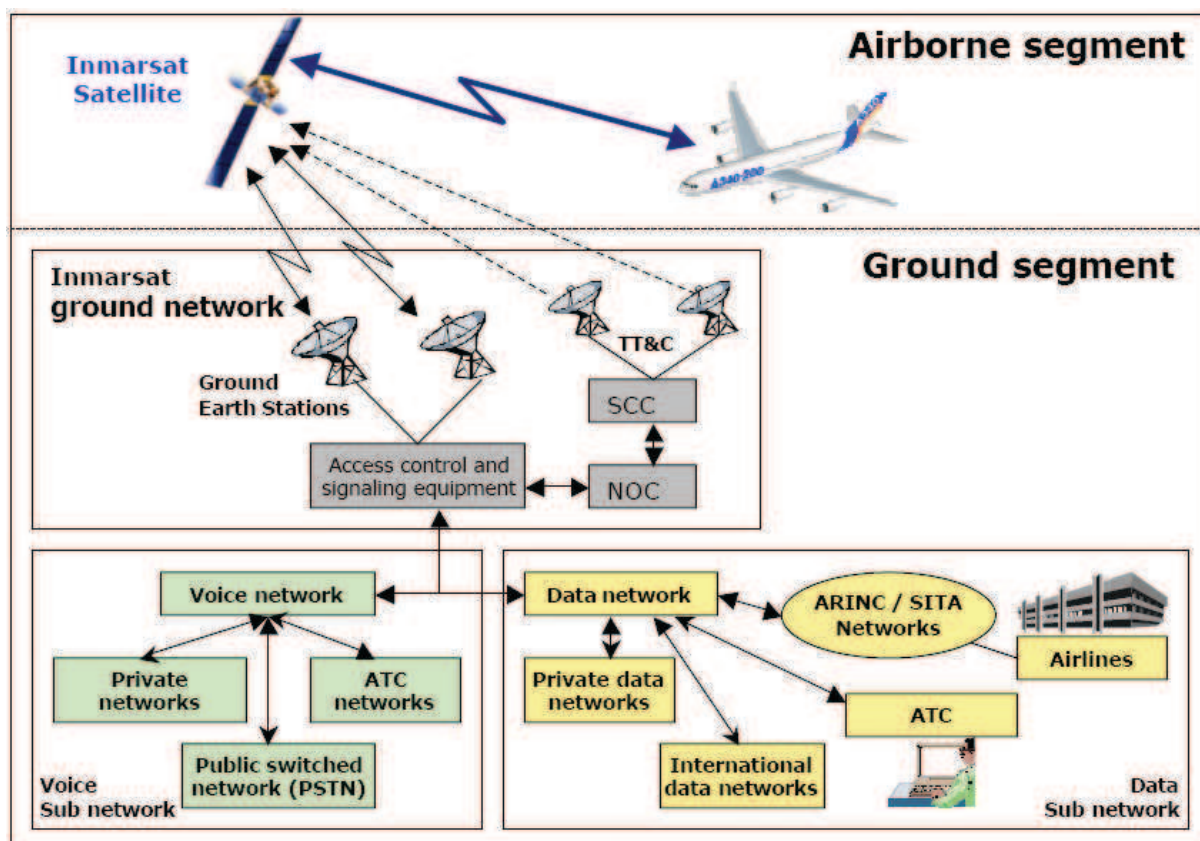


Figure 24 : Architecture du système aéronautique d'E-cab [58]

La Figure 24 présente l'architecture globale du système E-cab, qui est détaillée ci-dessous.

Le réseau sol Inmarsat (Inmarsat Ground Network) gère le système à travers les éléments suivants :

- La station TT&C ("Telemetry, Tracking and Command") assure les fonctions de télémétrie, suivi et commande.
- Le Satellite Control Center (SCC), faisant parti du TT&C, est principalement à la charge de la surveillance de l'orbite des satellites et leurs maintenances.
- Le Network Operation Center (NOC) est à la charge de l'affectation dynamique du canal pour acheminer la Qualité de Service requise.

Depuis les stations Sol (Ground Earth Stations GES), plusieurs liens opérationnels sont fournis en fonction du type de la communication (paquet ou circuit) et le service de communication (voix ou donnée)

Le segment airborne comprend trois blocs principaux :

- Un sous-système d'antenne avec un amplificateur à faible bruit et un duplexeur, une unité binaire streaming et l'antenne elle-même.
- Un satcom avionique avec l'unité satellite de données, une unité de fréquence radio et un amplificateur de haute puissance.
- Une Interface utilisateur pour le téléphone, fax, données, meeting par exemple.

BGAN supporte deux types de connexions de données IP [9] :

- Standard IP : offre un débit de données jusqu'à 492 kbit/s en Best Effort. Il est dédié aux applications classiques comme l'accès Web, l'e-mail et le transfert FTP.
- Streaming IP : optimisé pour les applications de diffusion audio et vidéo comme Windows Media et QuickTime. Deux applications peuvent être utilisées en même temps comme l'utilisation du streaming et la consultation de mail en partageant la bande passante. Dans ce contexte, le streaming est disponible avec quatre débits : 32, 64, 128 et 256 kbit/s.

Le BGAN et la QoS : Pour garantir une QoS, Inmarsat fournit une interface pour gérer et supporter la QoS sur le réseau. La Qualité de Service est négociée entre le terminal et le BGAN Core Network lorsqu'une connexion de données est ouverte. Pour assurer un débit satisfaisant pour une connexion.

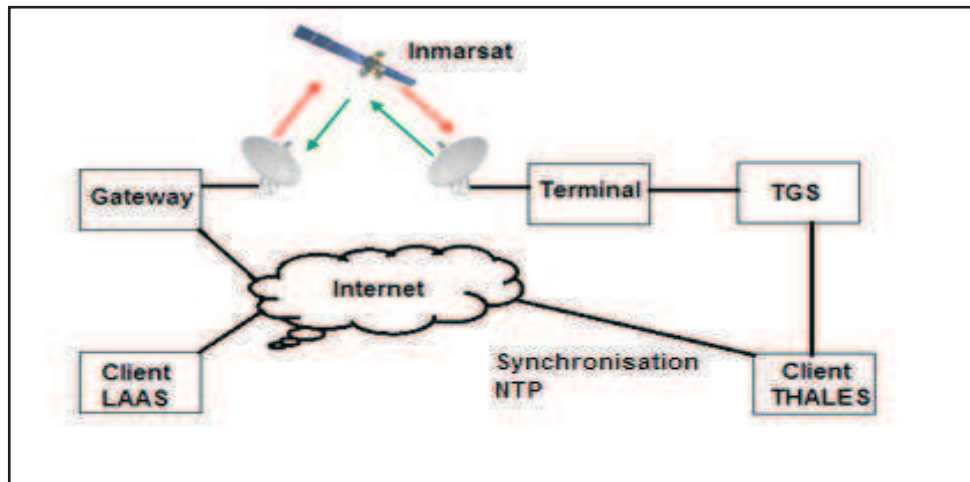


Figure 25 : Plateforme de test Inmarsat

La Figure 25 présente la plateforme de test Inmarsat. Elle se compose de :

- Client Thales : une station Dell XPS M1210 sous Fedora 9.
- Le TGS est une station de contrôle sous Debian avec deux interfaces réseaux. Une interface PPPo relie le TGS avec le terminal et une autre Ethernet le relie avec le client Thales via un routeur.
- Client LAAS : une machine se trouvant au LAAS-CNRS sous Fedora 5.
- La synchronisation NTP se fait par WiFi via l'interface Wlan0 du Client THALES.

Le terminal

Le terminal utilisé est un BGAN HNS 9201. C'est le terminal qui offre le plus de bande passante par rapport à la gamme BGAN.

Il est dédié aux utilisateurs individuels, soit en utilisant la bande passante des applications gourmandes comme la visioconférence, ou pour les petites équipes qui ont besoin de mettre en place un bureau temporaire avec une connectivité haut débit et des applications standards de données. Les débits atteints en standard IP sont 492 kbit/s (émission et réception). En streaming, une large sélection de débit pour le choix de la Qualité de Service est proposée (32 kbit/s, 64 kbit/s, 128 kbit/s et 256 kbit/s).

2.4 Conclusion sur les plateformes d'évaluation

Les deux émulateurs décrits précédemment ainsi que le simulateur NS-2 et la plateforme réelle d'OURSES présentent des avantages et inconvénients, ces derniers sont présentés dans le tableau suivant :

		Avantages	Inconvénients
Simulation	NS-2	<ul style="list-style-type: none"> - Simplicité d'utilisation 	<ul style="list-style-type: none"> - Absence du lien satellite (même émulé), se basant sur des modèles comme Raffaello Secchi par exemple - Instable et très lourd
Emulation	Platine	<ul style="list-style-type: none"> - Fidèle à DVB-RCS, il permet donc d'émuler entre autres les délais d'allocation du DAMA. - Vaste choix de paramètres configurables et mesurables 	<ul style="list-style-type: none"> - Nombreuses machines nécessaires ~10 pour un scénario conséquent - Très complexe à paramétrer - La mobilité doit être générée par un module Mobile IP
	SATEM	<ul style="list-style-type: none"> - Facile à interfacier au niveau IP. - Flexibilité lors d'un VHO (une seule interface) ainsi qu'une facilité de gestion du mécanisme Make-Before-Break. 	<ul style="list-style-type: none"> - Il n'émule pas les boucles d'allocation du DAMA
Réel	Plateforme OURSES Plateforme e-cab	<ul style="list-style-type: none"> - Résultats réels - Allocation DAMA sur la voie retour - Présence du mode ACM sur bande Ka 	<ul style="list-style-type: none"> - Paramétrage complexe des paramètres de QoS - Aspect boîte noire de la gateway
	Plateforme e-cab	<ul style="list-style-type: none"> - Résultats réels 	<ul style="list-style-type: none"> - Voie retour non implémentant le DVB-RCS

Tableau 8 : Comparaison entre les différentes méthodes d'évaluation

Ces plateformes de test nous permettent de tester les performances applicatives sur un lien satellite, selon le besoin, que ce soit en simulation, en émulation ou avec un système réel. Afin de réaliser les tests de performances, certains outils sont nécessaires. Dans la partie qui suit, nous présentons les outils qui nous ont été utiles pour réaliser nos évaluations.

3 Outils de mesure pour les systèmes émulés ou réels

Les tests sur les systèmes émulés et réels ont été réalisés à l'aide d'un certain nombre d'outils. Certains tests sont génériques (tests de fonctionnement dans de conditions normales) et d'autres de QoS. La plupart de ces outils sont disponibles en accès libre mais nous les avons complétés par des outils que nous avons implémentés et/ou modifiés.

Nous donnons ici un aperçu de ces différents outils :

3.1 Iperf

C'est un outil libre qui a été développé à l'origine par NLNR/DAST comme une alternative moderne pour mesurer les débits maximums de TCP et UDP. Iperf [59] permet de régler de nombreux paramètres TCP et UDP. Il permet d'avoir des informations sur le débit (TCP), gigue (UDP), perte de

paquets (UDP). Il est délimité par deux machines sur lesquelles il est installé. Une nouvelle version de cet outil a été proposée pour supporter les deux protocoles DCCP [60] et SCTP [61].

3.2 TCPTRACE

TCPTRACE [62] est un outil conçu par Shawn Ostermann de l'université d'Ohio, pour l'analyse de fichiers de dump TCP. Il peut prendre en entrée les fichiers produits par plusieurs sniffeurs de paquets populaires comme Tcpdump, Wireshark et WinDump (fichiers pcap). Tcptrace peut produire plusieurs types d'informations de chaque connexion, comme le temps écoulé et les segments envoyés et reçus en octets, les retransmissions, les délais d'aller-retour, les fenêtres de notification, le débit, etc. Il peut également produire un certain nombre de graphiques pour une analyse plus approfondie.

3.3 Wireshark

C'est un sniffeur/analyseur de protocole qui examine les données à partir d'un réseau en temps réel ou à partir d'une capture de fichier sur disque [63]. On peut naviguer de façon interactive sur les données capturées, visionner le résumé et l'information détaillée pour chaque ensemble. Wireshark possède quelques fonctions puissantes, incluant un affichage de langage filtré et la possibilité de visionner le flux reconstitué de la session TCP. Parmi les données que nous pouvons récupérer grâce à cet outil, l'adresse et le port source et destination utilisés, l'évolution des numéros de séquence pour une connexion TCP ou DCCP...etc.

3.4 NTP Daemon

Le daemon Network Time Protocol (NTP) [64] permet de faire des synchronisations temporelles de plusieurs machines. Un fonctionnement en client/serveur rend son utilisation très flexible et offre des possibilités d'avoir plusieurs niveaux de hiérarchie. Une exécution prolongée augmente le niveau de précision de la synchronisation.

3.5 JTG

Le Jugi's Traffic Generator (JTG) [65] est un générateur de trafic simple et puissant, développé par l'université d'Helsinki. Il permet un échange entre client et serveur des flux TCP et UDP facilement configurables. Il permet aussi de rejouer des flux déjà capturés, à l'aide de traces contenant les délais inter-paquet et la taille des paquets.

JTG permet aussi d'analyser les flux à partir des fichiers journal au format log, qui sont obtenus en récupérant les traces à la réception puis de tracer des courbes de délais, de variations de délais, de gigue ainsi que des paquets perdus, en utilisant la fonction `jtg_calc` et Gnuplot.

Ainsi, JTG peut tracer des graphes à partir de fichiers de sortie type MGEN (Multi GENerator), soit en injectant un fichier propre JTG ou un fichier ORENETA (moyennant une modification mineure). La Figure 26 résume les interactions entre les différents outils de mesure et d'analyse.

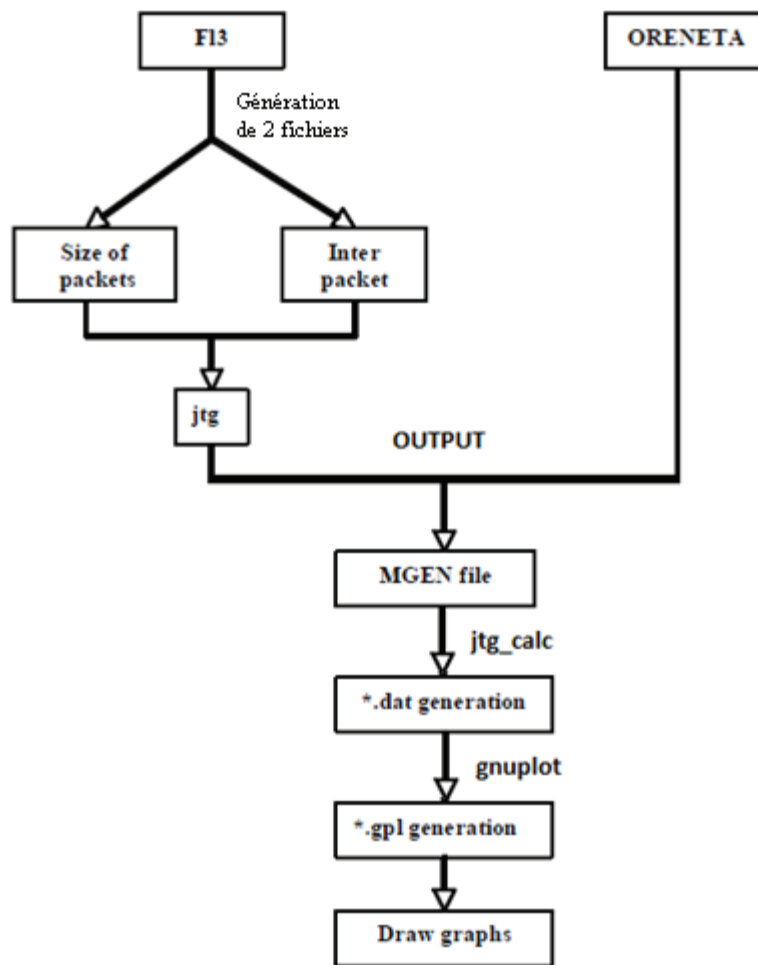


Figure 26 : Interaction entre JTG, Gnuplot, FL3 et ORENETA

Des modifications ont été apportées à cet outil pour l'adapter à nos expériences, à savoir :

- La possibilité de tracer les courbes de débit qui correspondent à chaque flux, en utilisant plusieurs méthodes de calculs de débit.
- Un changement a été apporté par [66] pour la continuité d'exécution lors d'un changement de réseau pour les tests de mobilité.

3.6 FL3

Pour un certain nombre de tests, l'expérience doit être refaite avec les mêmes profils de flux mais en utilisant des paramètres différents, c'est la raison pour laquelle, nous utilisons cet outil.

FL3 [66] est un outil développé par le LAAS-CNRS et qui comporte trois applications différentes :

- FLOW Capturer (FLOC) : permet de capturer un flux de données provenant d'une application pendant une durée déterminée. Il génère un fichier de trace "formatted_capture" contenant la date d'émission des paquets.
- FLOW REplayer (FLORE) : fonctionnant en client/serveur. Le client génère du trafic prédéfini en ayant pour entrée un fichier FLOC. La régénération de flux est composée en deux étapes. Une première phase de négociation (nombre de connexions à ouvrir et le type de protocole associé). La deuxième phase est celle de la génération. A chaque émission de paquet, le

client identifie à quelle connexion le paquet appartient, réalise un estampillage temporel, lui associe un numéro de séquence et le transmet. Le serveur crée chronologiquement un fichier de paquets reçus par connexion.

- Flow Analyser (FLAN) : permet d'avoir des statistiques comme les paquets perdus ou le débit des flux en ayant pour entrée des fichiers FLORE.

Nous utilisons aussi FL3 pour générer des fichiers log contenant les délais inter-paquet et la taille de chaque paquet. Ces fichiers seront utilisables par la suite avec JTG pour générer des fichiers type MGEN [67], pour le traçage de courbes. La Figure 26 illustre l'interaction entre les différents outils utilisés.

3.7 ORENETA

ORENETA [68] est une plateforme pour l'analyse des réseaux en temps réel, fournissant des outils pour la mesure d'une série de paramètres de Qualité de Service (les délais de bout en bout, les paquets perdus, le débit...etc.). Les données capturées peuvent être enregistrées dans des fichiers pour une analyse hors-ligne ultérieure. ORENETA a principalement été développé dans le projet européen EuQoS. Il a été utilisé dans le cadre du projet SATSIX. Cet outil permet d'analyser le trafic réel qui traverse le réseau (vidéo, audio, transfert de données...etc.)

La Figure 27 illustre le mode de fonctionnement d'un scénario ORENETA.

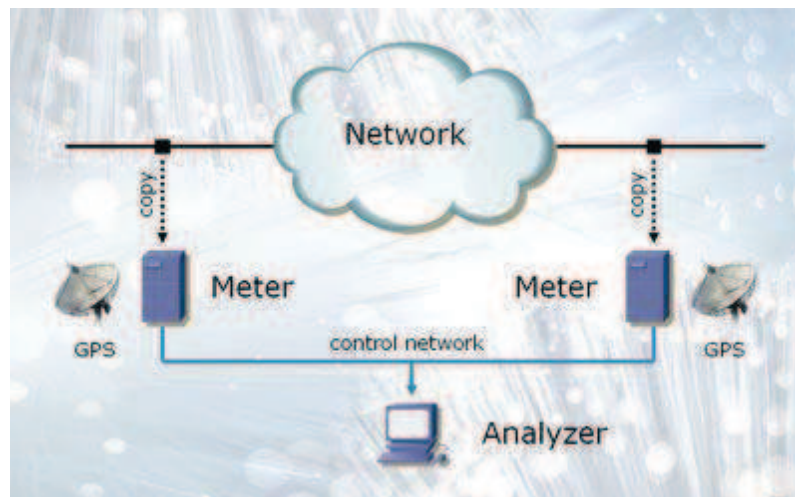


Figure 27 : scénario ORENETA [68]

On peut remarquer que l'Analyzer récolte les informations depuis deux points de mesures à l'extrémité du réseau à analyser.

Le Meter est un programme développé en C et installé sur une station linux qui écoute l'entrée du réseau et marque les paquets qu'on veut analyser. A la sortie, le deuxième Meter reconnaît les paquets marqués. Les deux Meters envoient les informations récoltées à l'Analyzer.

La première version de l'Analyzer a été développée en JAVA, incluant une interface graphique pour analyser les flux dans le réseau. Elle affiche les informations sur les flux traversant les deux Meters.

Les graphes en sortie concernent les flux vus par le meter 0 et 1, et qui sont ; le débit, le délai d'inter-paquet, le délai des paquets, la variation de délais et les paquets perdus.

La version actuelle d'ORENETA a été redéveloppée en C, avec beaucoup d'amélioration, proposées pour une utilisation robuste et plus rapide. Elle permet aussi de réaliser des mesures à partir de plusieurs points réseaux (meters) et de contrôler l'ensemble des flux ainsi que les paramètres de la QoS pour chaque segment contrôlé.

Les meters doivent être synchronisés avec NTP. Une précision supplémentaire peut être obtenue avec l'utilisation des horloges GPS. L'Analyzer n'a besoin d'aucune synchronisation parce qu'il observe les timestamps des paquets capturés par les meters.

3.8 WEB100

Le projet Web100 [69] a été créé spécialement pour développer une interface de contrôle avancée pour TCP. TCP a été instrumenté pour collecter les statistiques d'une connexion donnée. Il existe des instruments pour la capture des événements spécifiques tels que les segments envoyés et reçus. Il existe aussi de nombreux instruments plus subtils comme ceux qui caractérisent les événements protocolaires de TCP causant la réduction du débit de transmission. Ces outils sont désignés, d'une manière générale, en tant que Kernel Instrument Set (KIS), qui est le noyau du projet Web100. Les instruments sont documentés dans une base de données en ligne [70]. Les concepteurs de Web100 attendent une standardisation de l'interface Web100, pour devenir une norme sous la forme d'une base statistique étendue de gestion d'information (Management Information Base) MIB de TCP [71].

Avec l'introduction de certaines variables, l'API Web100 et la structure du noyau de données fournissent un framework convenable pour contrôler les versions non standardisées ou expérimentales de TCP.

Dans un contexte d'enseignement, Web100 a été utilisé pour fournir un instrument pour la réalisation de tests d'expérimentation, offrant aux étudiants et chercheurs des expériences pratiques. L'analyse en temps réel qu'offre Web100 pour les variables TCP, et la possibilité de modifier certains paramètres aident les chercheurs à comprendre les différentes composantes qu'impactent les flux TCP.

Web100 est actuellement présent dans des noyaux Linux en mode graphique. Tous les codes Web100 sont disponibles sur www.web100.org.

Pour plus de maniabilité de données, nous avons développé un client Web100 sous C qui permet d'utiliser le noyau Web100 pour récupérer les variables TCP dans un fichier traces afin de pouvoir réaliser des analyses plus approfondies ainsi que des comparaisons de résultats.

Les outils présentés permettent de mesurer différents paramètres de QoS tels que le débit, le délai de bout en bout ou la gigue, ainsi que des paramètres d'une communication TCP comme la taille des fenêtres de congestion ou les RTOs. D'autres outils permettent de rejouer des traces déjà capturées sur un lien émulé ou réel, cela peut servir à mettre en œuvre une analyse comparative et statistique des résultats. Dans cette optique, nous avons réalisé une série de tests d'évaluation de la plateforme réelle OURSES.

4 Exemple d'évaluation : le projet OURSES

L'objectif principal de la campagne de test est de caractériser les liaisons voies aller et retour d'un satellite opérationnel, ainsi que de tester les performances d'applications par satellite avec DVB-S2 et ACM, avec fading réel ou émulé.

Les tests réalisés dans le cadre du projet OURSES ont été divisés en trois catégories :

- Les tests basiques de la QoS
- L'analyse de TCP
- Les tests subjectifs

La Figure 28 présente une schématisation du banc de test. Un client PC2 placé derrière le terminal peut envoyer (voie retour) ou recevoir (voie aller) un flux de données vers le PC1 (serveur) qui se trouve derrière la gateway. Un générateur de bruit est placé juste entre le terminal et l'antenne satellite.

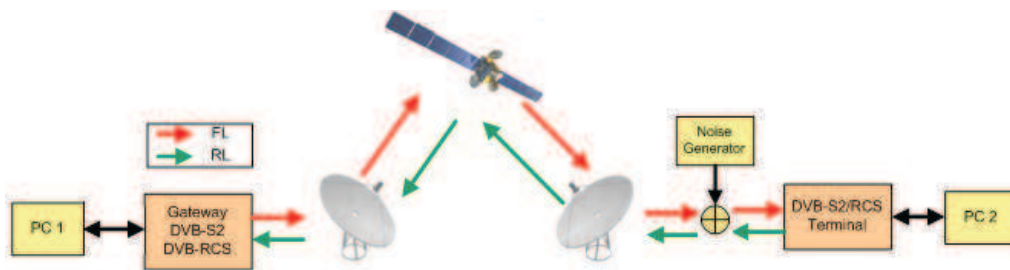


Figure 28 : Schéma descriptif de la plateforme de test

4.1 Tests basiques de la QoS

4.1.1 Les tests sur la voie Aller

Nous nous intéressons spécialement aux tests sur le lien DVB-S2. Le PC1 derrière la gateway génère du flux sur la voie aller vers le PC2. L'ACM étant mis en place sur ce lien, permet d'avoir l'impact de cette technologie sur nos tests. Le rapport signal/bruit SNR (Signal Noise Ratio) permet caractériser la qualité du lien, plus le SNR est grand plus les perturbations diminuent. Plus le SNR baisse, moins la qualité du média est bonne ce qui influence directement les pertes liées au média. Les tests sur cette voie vont permettre de déterminer la QoS rendue en termes de pertes relativement au bruit ainsi que l'apport de l'ACM. Ces tests sur la voie aller sont les premières expériences réelles ACM sur une bande Ka.

Relation SNR / BER / PER :

L'objectif est de déterminer la qualité du service rendu en termes de pertes et de mesures de la relation SNR / BER / PER. Dans ce test, nous avons procédé à :

- La génération d'un flux de 1Mbit/s avec Iperf
- L'injection de bruit aux instants 30 et 60 secondes avec un générateur de bruit (fading).
- La récupération de traces avec Oreneta (application) depuis le ST (SNR)

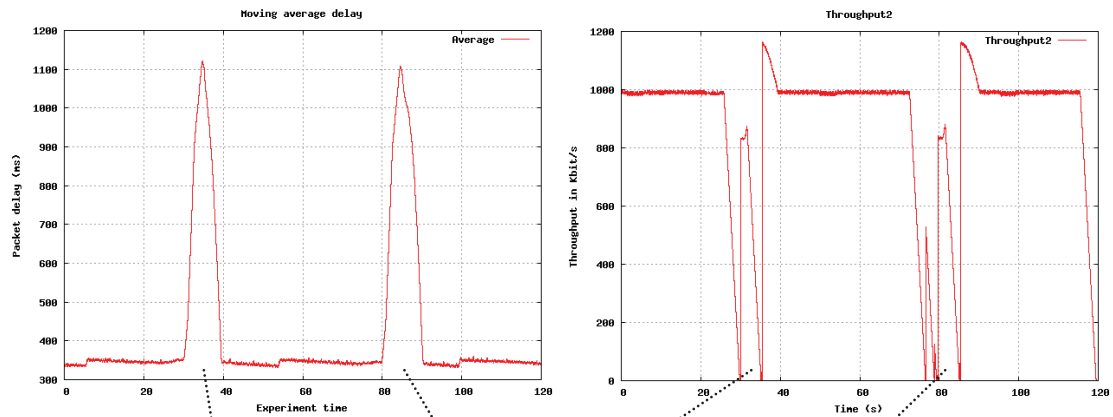


Figure 29 : (a) Délai des paquets et (b) débit

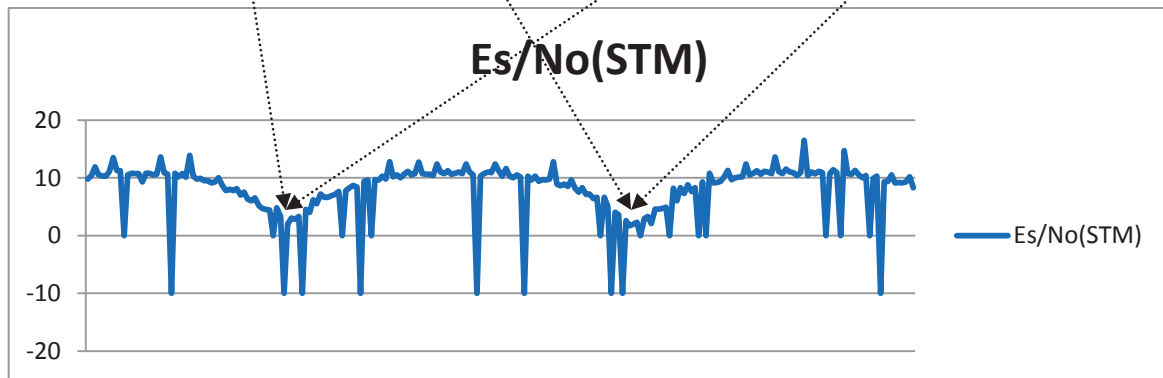


Figure 30 : Evolution du rapport signal/bruit

La Figure 29 présente, à gauche l'évolution du délai moyen de bout en bout et à droite le débit. Le délai augmente entre les temps 30- 40 secondes et 80-90 secondes. En effet, un générateur de bruit applique un phénomène de fading provoquant des pertes ce qui influence le délai de bout en bout et naturellement le débit qui devient pratiquement nul au-delà d'un certain seuil de bruit. Nous observons dans la Figure 30 la variation du SNR. A partir des traces récupérées, on remarque que la communication commence à se dégrader à la $t=25^{\text{ème}}$ seconde et devient nulle à la $34^{\text{ème}}$ seconde. Elle revient ensuite à la normale à la $39^{\text{ème}}$ seconde. Ainsi, la limite du Es/No pour qu'une communication soit encore possible est définie par :

- Dégradation de la qualité du lien à partir de 4.7 dB
- Limite de coupure si < 2 dB

Le nombre de modcod utilisés durant les phases de fading est de 3 (QPSK 1/4 puis 8PSK 2/3, QPSK 8/9, QPSK 5/6 puis à la fin QPSK ¼ au maximum du bruit). Les valeurs négatives du Es/No sont négligées.

$$PER = \frac{\text{Nombre de paquets perdus}}{\text{Nombre de paquets total}} = 0.43\%$$

Équation 1: Calcul du Packet Error Rate PER

Toutes les pertes sont concentrées dans les phases de fading. Le PER/BER (Équation 1) est inversement proportionnel au SNR. Le débit est lui proportionnel au SNR.

Mesure des durées de changements de codage

L'objectif est de déterminer la qualité du service rendu en termes de pertes et de capacité en fonction du SNR avec un changement de codage adaptatif. Dans ce test, nous avons procédé à :

- la génération du flux de 1 Mbit/s avec lperf
- l'injection de bruit aux secondes 30 et 60 avec un générateur de bruit (fading).
- La récupération de traces depuis le ST (SNR)

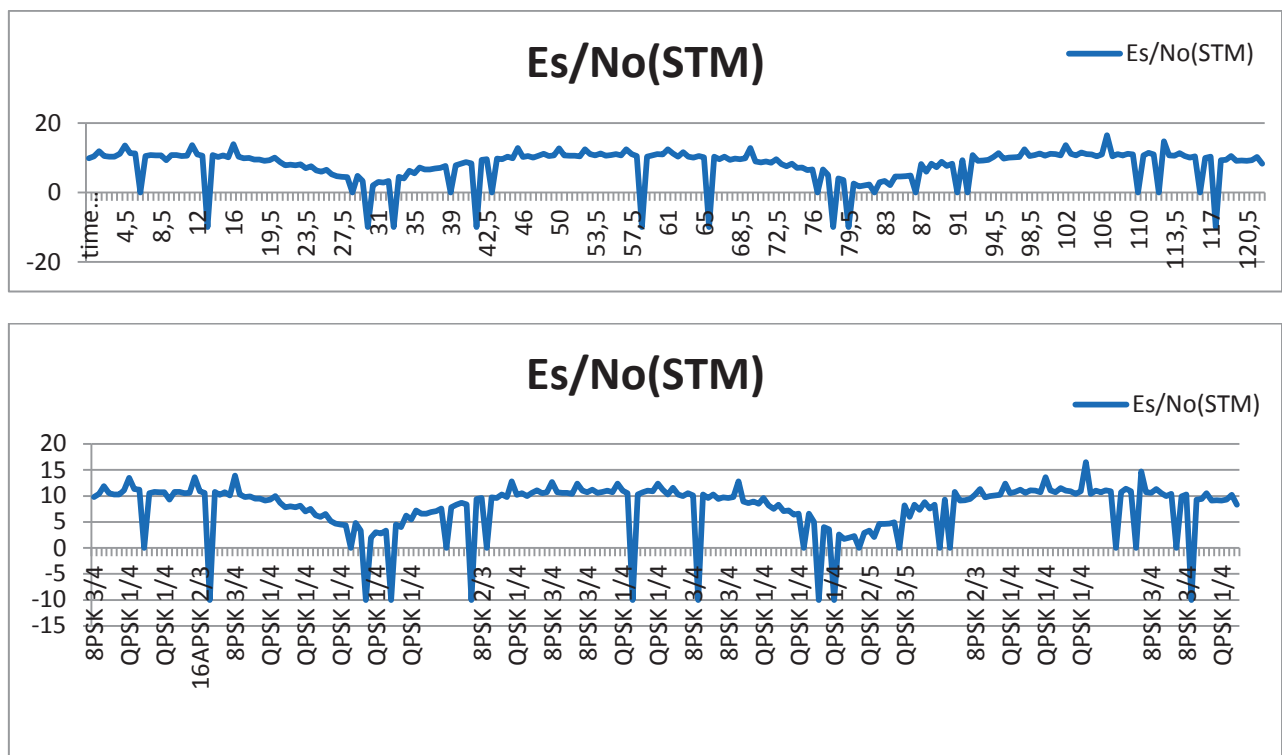


Figure 31 : SNR et MODCOD Vs. Temps avec bruit

Dans les deux graphiques de la Figure 31, nous observons une variation (changement) du modcod en fonction du SNR, plus exactement, une stabilité du modcod entre 8PSK 2/3 et 8PSK 3/4 (valeurs très proches) quand le SNR est stable (8-10) mais dès qu'il y a une atténuation du signal (automatiquement du SNR), on remarque le changement du modcod vers 8PSK 3/4, QPSK 5/6 puis QPSK 1/4. Par la suite, il se stabilise avec une priorité à QPSK 1/4. Nous pouvons donc dire que le modcod s'adapte aux signaux et affecte le débit selon le changement du SNR. Puisque le débit est proportionnel au SNR et le modocod utilisé change si le SNR vient à diminuer, le débit va lui aussi diminuer pour « s'adapter » aux conditions du signal courant affecté par le bruit. L'ordre de classement des modcods en fonction du débit est le suivant (ordre croissant du débit, c'est-à-dire, du plus mauvais modcod au meilleur) :

QPSK 1/4, QPSK 1/3, QPSK 2/5, QPSK 1/2, QPSK 3/5, QPSK 2/3, QPSK 3/4, QPSK 4/5, QPSK 5/6, QPSK 8/9, QPSK 9/10, 8PSK 3/5, 8PSK 2/3, 8PSK 3/4, 8PSK 5/6, 8PSK 8/9, 8PSK 9/10, 16 APSK 2/3.

4.1.2 Les tests sur la voie Retour

La norme DVB-RCS (Digital Video Broadcast Return Channel over Satellite) standardise une voie retour via satellite pour les terminaux satellites. Couplée avec la voie aller qui s'appuie sur la norme DVB-S/DVB-S2 (Digital Video Broadcast for Satellite/second generation), DVB-RCS introduit l'interactivité nécessaire à la définition d'une architecture d'un système multimédia large bande de nouvelle génération. En effet, la norme DVB-RCS constitue l'élément essentiel des systèmes satellites bidirectionnels et permet au ST d'interagir.

L'entité chargée de la gestion des ressources est le NCC (Network Control Center). Par le biais d'une table de signalisation (TBTP) émise périodiquement sur la voie aller, le NCC communique aux ST les affectations de ressources de la voie retour.

Afin d'allouer la bande passante de manière dynamique et donc plus efficace, le DAMA (Demand Assigned Multiple Access) avec une architecture client/serveur permet aux STs de requérir régulièrement auprès du NCC de la « capacité d'émission », c'est à dire des réservations de timeslots pendant lesquels ils pourront émettre sur la voie retour sans contention possible. Ces demandes de réservation de capacité sont calculées par client DAMA du ST selon l'état de ses files d'attente d'émission et transmises par des requêtes de capacité (CR), et qui sont :

- Des requêtes de bande passante fixe (*Continuous Rate Assignment*);
- Des requêtes de bande passante variable (*Rate Based Dynamic Capacity*);
- Des requêtes de transmission d'un volume de données (*Volume Based Dynamic Capacity*);
- Des requêtes de transmission d'un volume de données (*Absolute Volume Based Dynamic Capacity et Absolute Volume Based Capacity*).

Une allocation « bonus » FCA (Free Capacity Assignment) offerte par le NCC est redistribuée équitablement lorsqu'il reste de la bande passante non utilisée.

Par contre, entre l'émission d'une requête dynamique (RBDC ou VBDC) par le ST vers le NCC et la consommation effective de ses capacités par le ST, le cycle de Requête/Allocation (au moins un aller-retour dans le réseau Satellite) ne peut descendre en deçà d'un délai minimum incompressible. Appelé TMSL (Minimum Scheduling Latency), il correspond à environ 600ms. Cependant, selon l'état de la voie retour du réseau satellite, ce délai peut aller jusqu'à plusieurs secondes si une supertrame ne suffit pas à écouler le volume requis par l'ensemble des STs à un instant précis. Ce délai va se répercuter au niveau des applications multimédia sur son interactivité.

En plus du mécanisme d'allocation dynamique, des mécanismes de gestion de la QoS sont présents sur la voie retour. La QoS est utilisée pour fournir une différenciation de traitement des différents types de trafic IP. L'objectif est d'être capable de satisfaire les exigences des différents flux d'applications selon leur importance, tout en minimisant la bande passante requise sur la voie retour.

Comme nous l'avons présenté dans la description du terminal STM Satlink, trois niveaux de priorité sont définis et utilisés lors de nos tests.

Influence de la QoS sur un flux CBR au-delà du contrat de service

Dans ce test, nous allons nous focaliser sur l'influence de la classe de service (EF/AF/BE) ou du type de requêtes de capacité sur le délai d'un flux CBR lorsque le contrat de service est dépassé par le terminal. Dans ce test, nous avons procédé à :

- La génération d'un flux de saturation avec IPERF (service BE).
- La génération d'un trafic CBR 256 kbit/s avec IPERF.
- L'analyse du trafic CBR généré à l'aide des meters lancés par ORENETA

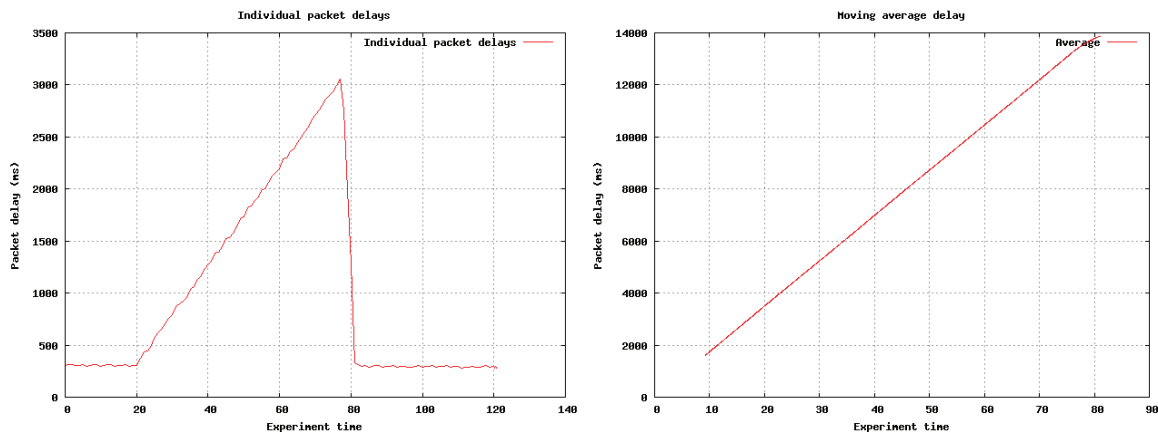


Figure 32 : Délai du flux 1 (gauche) et 2 (droite)

Sur le premier graphe (Figure 32), nous observons que le délai (avant le temps 80 secondes) ne cesse d'augmenter. Ceci est causé par la saturation de la bande passante (la somme de la bande passante des deux flux envoyés est supérieure à la bande passante de la voie retour). Dès qu'on marque le flux 1 en EF (graphe 1 à gauche), son délai revient à la normale (autour de 300 ms) mais le délai du flux 2 (graphe 2 à droite) continue à augmenter car les données sont bufférisées puisque il n'est pas marqué et est naturellement affilié à BE.

Nous concluons que la classe de service EF protège le flux prioritaire.

Mesure du partage de la capacité pour la VoIP

Dans ce test, nous allons déterminer la qualité du service rendu en termes d'équilibre de capacité lors de connexions multiples. Cela permet de tester l'équité du partage de la bande passante et vérifier le fonctionnement du mécanisme d'allocation de ressources. Dans ce test, nous procédons à :

- Le démarrage de FL3 sur les deux hôtes distants.
- La génération de 11 flux audio (UDP) avec IPERF (1 par 5 secondes).
- L'analyse le trafic généré à l'aide de FL3

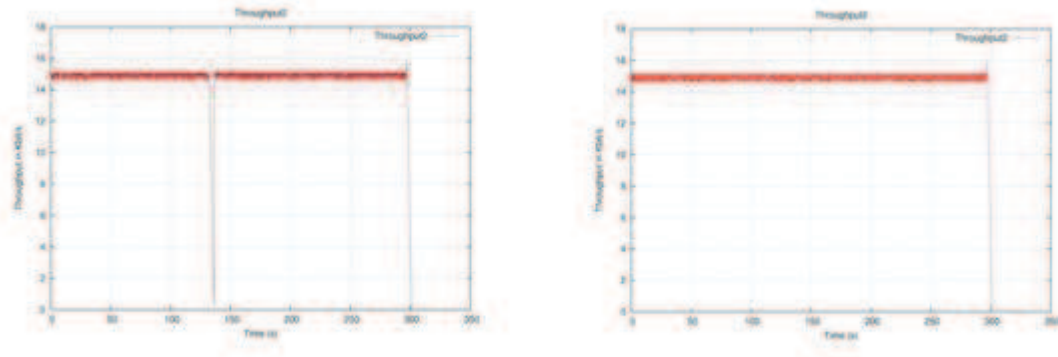


Figure 33: Débit de deux flux (parmi les 11 lancés)

Sur la Figure 33, nous observons que le débit est identique pour chaque flux. Ceci traduit un partage équitable de la bande passante entre les différents flux.

Le partage des ressources entre différents flux est assuré par le mécanisme d'allocation des ressources DAMA. Il vise une équité dans l'allocation des ressources pour les différents flux et aussi pour terminaux.

4.1.3 Les tests dans les deux sens simultanément

Mesure du délai et de la gigue

Dans ce test, nous déterminons la qualité du service rendu en termes de délai et de gigue. Dans ce test, nous procédons à :

- Le démarrage d'ORENETA sur les hôtes distants.
- La génération d'un trafic avec FL3 dans les deux sens avec un profil audio GSM.

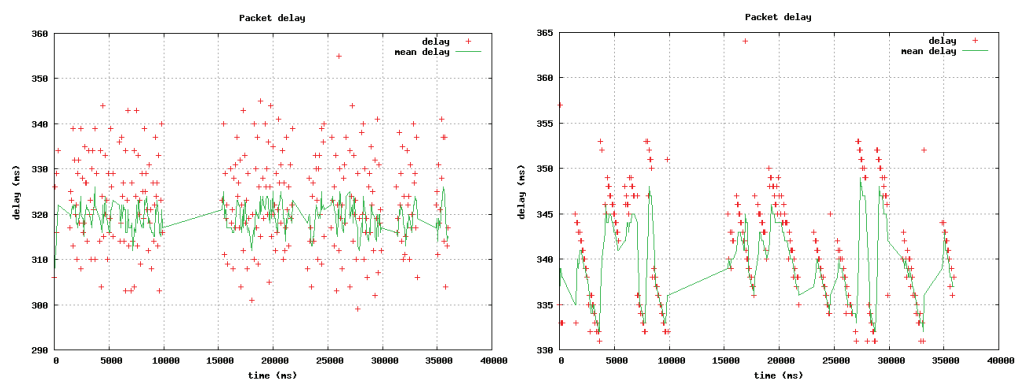


Figure 34 : Le délai des paquets sur la voie aller (gauche) et retour (droite)

Le profil GSM est très contraignant pour le DAMA car le débit est très variable du fait des suppressions de silence. Toutefois, ce premier test utilise de l'allocation statique (CRA) et donc le délai ne devrait pas être affecté par le profil.

Nous observons que sur les deux voies, le délai moyen est de 320-340 ms, signifiant un minimum de délai sur satellite. Il faut aussi signaler que le service est configuré en voie retour en FULL CRA, c'est-à-dire, qu'aucune requêtes RDBC ou VBDC n'est utilisée et que toute la bande passante est allouée en CRA.

Mesure de la capacité

Dans ce test, nous allons déterminer la qualité du service rendu en termes de capacité. Nous procédons à :

- Le démarrage d'ORENETA sur les hôtes distants.
- La génération d'un trafic avec FL3 dans les deux sens avec un profil MPEG 1.2 Mbit/s.

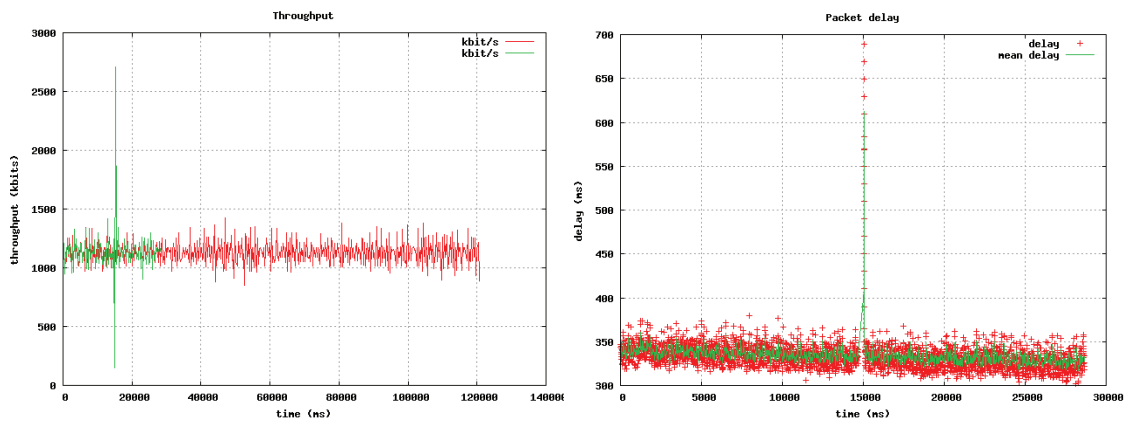


Figure 35 : Débit et délai sur la voie aller

Dans ce test, nous envoyons un flux MPEG 1.2 Mbit/s. Le débit reçu est identique au débit envoyé 1.2 Mbit/s. Le délai minimal reflète la capacité de ce flux à supporter le lien en voie aller (~350 ms). La capacité en voie aller est > 1.2 Mbit/s (voir Figure 35).

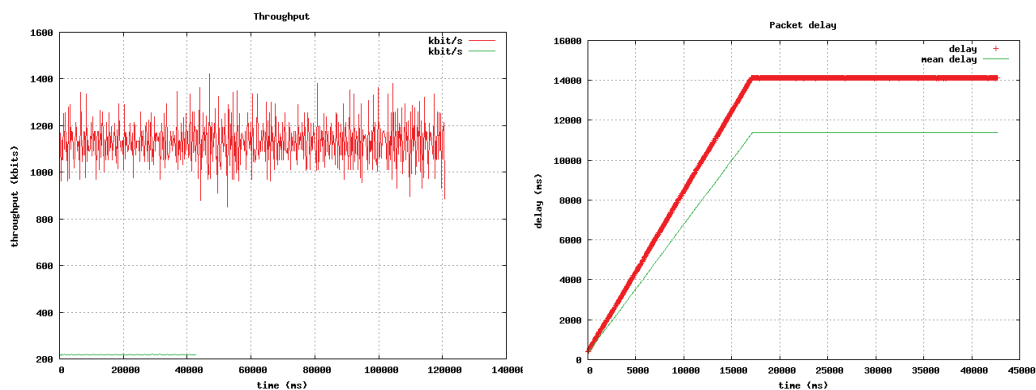


Figure 36 : Débit et délai sur la voie retour

Sur la voie retour, nous envoyons le même flux que sur la voie aller (MPEG 1.2 Mbit/s). Nous pouvons remarquer que le débit reçu dépasse de peu 200 kbit/s avec une explosion des buffers (Figure 36). La capacité en voie retour est donc de ~256 kbit/s (valeur pratique est ~230 kbit/s).

Le flux MPEG exploite très bien la voie aller et atteint son débit maximum. Cependant, MPEG a des difficultés sur la voie retour puisque son débit d'émission dépasse la taille de la bande passante disponible. Nous réalisons un autre test d'Iperf afin de mesurer la valeur réelle de la bande passante sur la voie retour. La Figure 37 montre la capacité sur cette voie. Le débit réel oscille autour de 230 kbit/s. Cela est dû aux pertes relevées sur le lien. Le taux de pertes en paquets est de l'ordre de 1,19%.

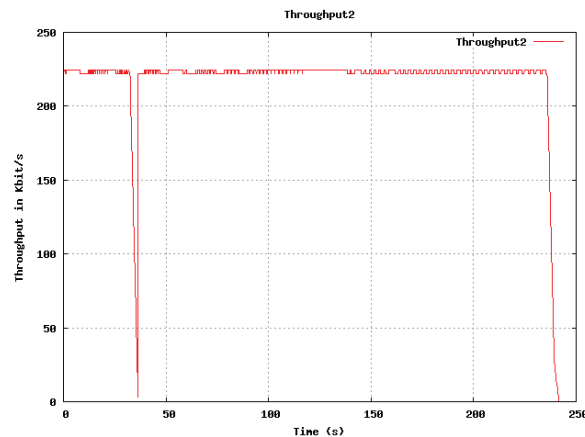


Figure 37 : Débit d'un flux Iperf

4.2 Tests de TCP

Après les tests basiques de bande passante, de délai et l'influence du SNR sur les pertes, nous nous sommes concentrés sur le protocole TCP. Nous nous sommes focalisés sur l'aspect contrôle de congestion et plus précisément, l'évolution des tailles de fenêtres et des timers de retransmissions RTOs. L'outil de mesure est Web 100. Bien qu'il présente un avantage puisqu'il récupère les paramètres TCP directement du noyau du système d'exploitation, mais il est difficile de le mettre en place et le configurer. Un autre problème a été rencontré est celui de la synchronisation de l'application de mesure Web100 avec l'application de génération de flux.

4.2.1 Evolution de la taille des fenêtres

Nous déterminons dans ce test le comportement de l'accroissement des tailles de fenêtres TCP sur la liaison satellite. Nous procédons au :

- Lancement du client IPERF TCP sur les hôtes distants.
- Lancement du client libweb100 sur l'émetteur

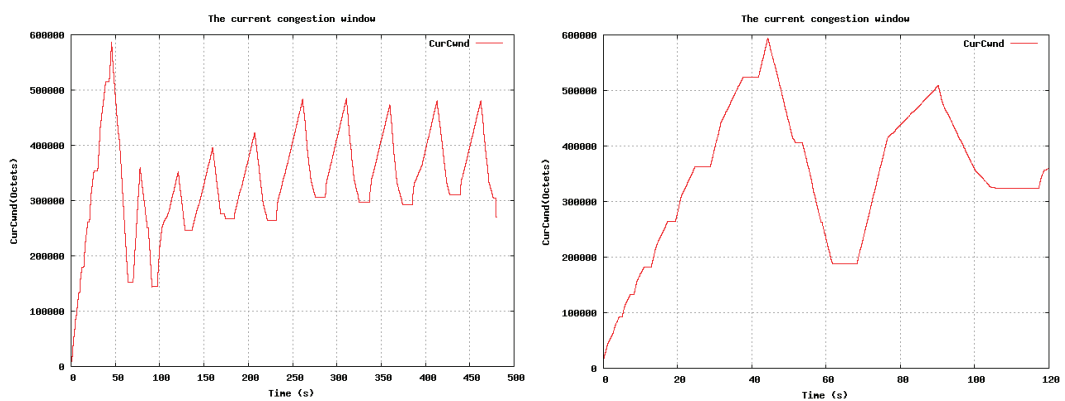


Figure 38 : Evolution de la taille de la fenêtre de congestion

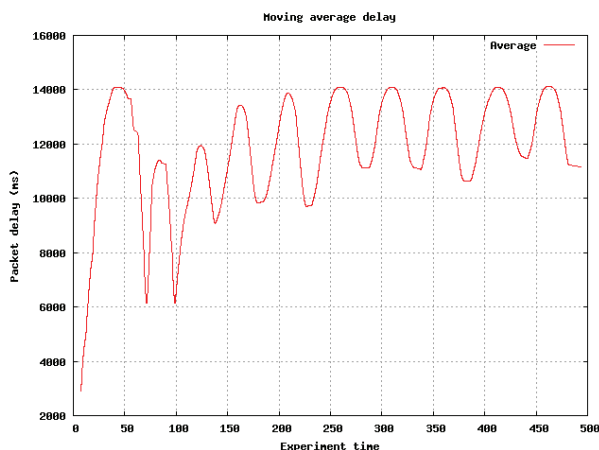


Figure 39 : Evolution du délai moyen

Dans ce test, la première remarque qu'on peut faire c'est l'évolution de la fenêtre de congestion qui ne suit pas l'évolution préconisée par l'IETF. Vu l'utilisation de machine Linux, l'auto tuning de TCP est automatiquement activée. De plus, l'implémentation de TCP sur des stations Linux [72] diffère des recommandations faites par l'IETF [73].

Dans la Figure 38 (cas 1 : temps couvert, cas 2 : temps clair), nous remarquons que TCP commence par un *Slow Start* rapide jusqu'à ce qu'il rencontre une congestion. Par la suite, il réduit la taille de sa fenêtre et entre en phase de *Congestion Avoidance*.

En termes de délai, la Figure 39 présente l'évolution des délais et montre une explosion de buffers du terminal. Cela entraîne des pertes sur liens ainsi que des fluctuations dans le débit.

4.2.2 Mesure des RTO et des fausses détections de pertes

Nous mesurons dans ce test l'évolution des timers de retransmission (RTO) de TCP sur la liaison satellite. Nous procédons comme suit :

- Lancement du client IPERF TCP sur les hôtes distants.
- Lancement du client libweb100 sur l'émetteur

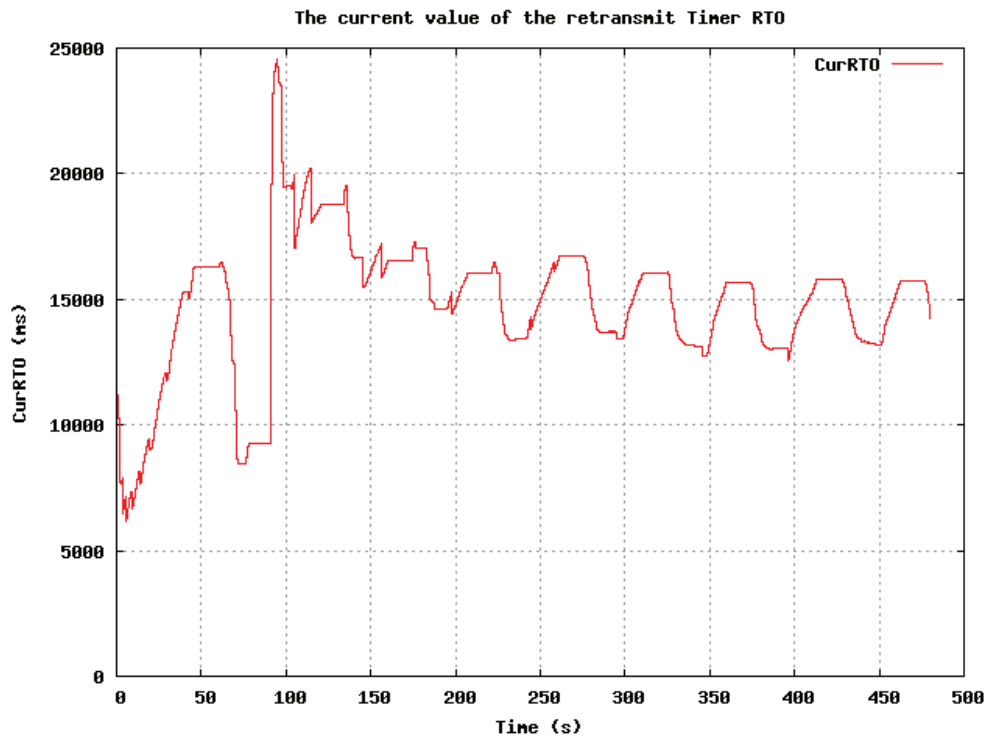


Figure 40 : Evolution du RTO (gauche) et pertes ponctuelles (droite)

Sur la Figure 40 (gauche), le RTO atteint des valeurs qui avoisinent les 15000 ms, ce qui est considérable. En effet, le RTO est calculé à travers un échantillonnage du RTT, qui lui aussi est important. Cela est dû aux grands délais de bufferisation et propagation.

En effet, un mauvais calcul du RTO génère des problèmes de contrôle de congestion puisque les timers peuvent expirer sans qu'une congestion soit présente, ou inversement avec des paquets en attente dans les buffers qui peuvent être déclarés perdus (fausse détection de pertes).

Nous avons comptabilisé un seul timeout (vu la durée du RTO) et 11 duplicate ACKs (11 utilisations du *Fast Retransmit*). Le grand délai satellite crée ce genre de problème en déclarant des paquets "supposés" en retards comme perdus (fausses congestions).

4.2.3 Influence du SNR sur TCP en mode ACM

Nous déterminons dans ce test la robustesse de TCP face à la dégradation SNR en mode ACM. Nous procédons comme suit :

- Le lancement du client IPERF TCP sur les hôtes distants.
- Le lancement du client libweb100 sur l'émetteur.
- L'utilisation d'un générateur de bruit toutes les 25 secondes.

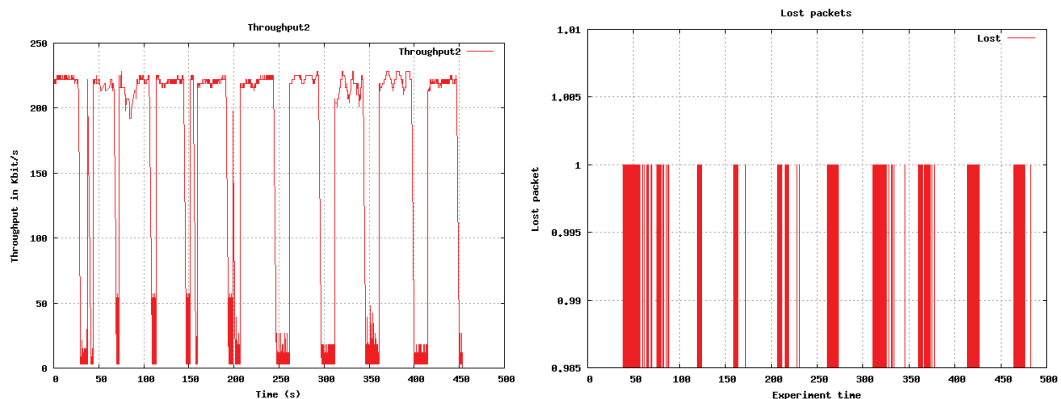


Figure 41 : Débit (à gauche) et pertes ponctuelles (à droite)

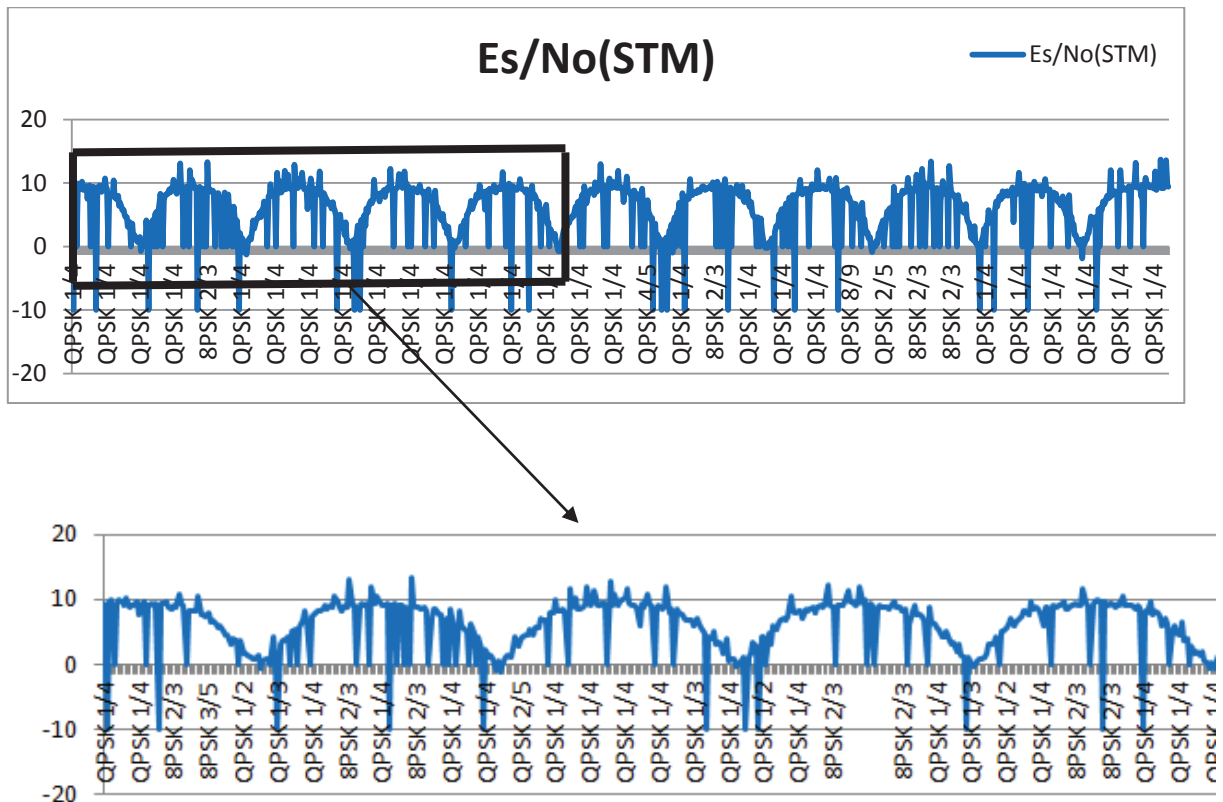


Figure 42 : SNR et MODCOD en fonction de l'évolution du temps

Dans le graphe (Figure 42), nous observons des chutes à répétition du débit dû au phénomène de fading (coupure presque totale dans le cas où l'atténuation est maximale), mais revient à la normale dès que le bruit disparaît. Les pertes correspondent aux périodes de fading. Le codage change en fonction du bruit et par conséquent en fonction du SNR. Il varie entre 8PSK 2/3 quand le bruit est faible ou nul et QPSK 1/4 quand le bruit est au maximum. TCP, comme tout protocole est sensible aux bruits mais grâce aux mécanismes d'ACM le protocole est peu perturbé et atteint, hors période de fading, un débit proche de la bande passante disponible.

4.3 Les tests subjectifs

Les tests objectifs permettent d'avoir la qualité effectivement perçue par un ensemble de population (ensemble d'utilisateurs). Cela donne un point de vue différent des caractéristiques de QoS données par les fournisseurs de service.

Dans cette partie, nous allons présenter quelques tests subjectifs réalisés. La métrique principale utilisée est la note d'opinion moyenne MOS (Mean Opinion Score).

4.3.1 Applications Audio / VoIP

Nous allons déterminer la qualité d'expérience d'une application de téléphonie IP sur réseaux par satellite en configuration nominale de ce dernier. La métrique utilisée est le MOS : qualité audio écoutée, fluidité et coupure.

Lors du lancement d'une application audio (GSM dans notre cas qui n'est pas très gourmand comme codec avec 14 kbit/s de débit), nous remarquons une bonne qualité du son écouté, avec une fluidité correcte et de rares coupures.

Le MOS moyen est égal à 3,4 sachant que le codec GSM 14 kbit/s a un MOS de 3,7 sur un réseau filaire (La compression GSM diminue considérablement la qualité initiale du média audio).

4.3.2 Applications Vidéo Streaming

L'objectif est de déterminer la qualité d'expérience d'une application de vidéo streaming sur réseaux par satellite en configuration nominale de ce dernier. La métrique utilisée est le MOS : qualité perçue, fluidité, artefacts/pertes d'images.

Lors du lancement d'une application de streaming (VLC dans notre cas), nous observons une fluidité de l'image et une belle qualité à la réception. Il faut noter qu'un décalage a été remarqué dû au délai de propagation du flux (300 ms). Quelques coupures (blocage de la vidéo ou carrés noirs) ont aussi été remarquées à cause des rares paquets perdus. Le MOS moyen est de 3,5.

4.3.3 Applications Audio / Vidéo Synchrones / Visioconférence

Nous allons déterminer la qualité d'expérience d'une application de visioconférence IP sur réseaux par satellite en configuration nominale de ce dernier. La métrique utilisée est le MOS : qualité perçue, fluidité audio et vidéo, artefacts / pertes d'images ...

Lors de l'utilisation d'une application de visioconférence, la qualité de l'audio reste pratiquement la même que le test avec l'audio seul, mais un léger blocage de l'image est remarqué avec la vidéo. Le MOS moyen est de 3,4. Le type de codec utilisé est automatique (sélection automatique de codec avec Skype).

4.3.4 Application données/Web

L'objectif est de déterminer l'aptitude du système satellite à délivrer un service Web (accès Internet classique).

Le chargement d'une page web est assez lent autour de 10 secondes (relativement à une connexion ADSL) mais assez souple vu la bande passante allouée. La qualité perçue moyenne est de 3 sur 5.

4.3.5 Application données/FTP

L'objectif est de déterminer l'aptitude du système satellite à délivrer un service FTP. La qualité perçue est mesurée grâce à la durée de téléchargement d'un fichier de 1Mo, l'intégrité du téléchargement ainsi que la durée moyenne inférieure à 1 min.

Le fichier téléchargé est intègre et complet, la durée est inférieure à 1 min (supérieure dans certains cas comme celui d'un mauvais signal).

4.3.6 Délais de synchronisation du terminal au réseau

Dans ce test, nous allons déterminer la durée de synchronisation d'un terminal au réseau.

Le délai de synchronisation du terminal (celui du CNES) est entre 40 secondes et 1 minute dans des conditions météo favorables. Mais à un certain niveau du SNR (assez bas c'est-à-dire < 3 dB) la synchronisation devient impossible (en utilisant un générateur de bruit ou mauvaises conditions météo). La valeur du SNR est inférieure à 5 dB.

4.4 Limitation et problèmes rencontrés

Nous avons rencontré plusieurs problèmes lors de nos tests, spécialement lors des tests sur la plateforme OURSES. Ces problèmes concernent la difficulté de configuration de la plateforme. En effet, la configuration du terminal surtout avec le paramétrage de la QdS reste assez complexe. De plus, le côté "boîte noire" de la gateway rend encore plus difficile la supervision de la voie retour et rend l'explication de certains comportements difficiles. Côté liaison physique, nous avons remarqué plusieurs problèmes liés aux coupures à répétitions du lien causées par les mauvaises conditions météorologiques, la mise au point de l'antenne ainsi que des problèmes liés au matériel lui-même.

Du côté ACM, le changement fréquent de modcod influe sur les résultats obtenus. En effet, un changement fréquent ne laisse pas le temps d'atteindre un débit stationnaire. Cela est dû au délai de propagation minimal dans un réseau satellite, qui affecte l'envoi des valeurs de PER, et par la suite la sélection du nouveau codage. Ce comportement crée des fluctuations dans le comportement général des applications.

Pour finir, le temps d'utilisation de la plateforme de test était partagé entre les différents partenaires, impliquant des retards sur certains tests. De plus, l'utilisation d'un seul terminal ne permettant pas de réaliser les tests en double bonde.

4.5 Conclusion

Nous avons procédé à un ensemble de tests qui ont permis d'avoir une caractérisation du lien satellite sur la voie aller et retour, mais aussi de soulever certains problèmes relatifs à des tests réels et de comportements applicatifs dans une campagne de tests sur un vrai lien satellite.

Nous avons soulevé le problème de fausses détections de pertes, puisque TCP gère la congestion avec des pertes à l'aide de certains timers à savoir le RTO. Pendant nos tests, nous avons remarqué que le RTT peut atteindre des valeurs très grandes jusqu'à 15 secondes dans certains cas, ou avoir de très petites valeurs, ceci est dû à la manière avec laquelle les RTOs sont calculés (en fonction du délai qui lui aussi est grand). Ceci pénalise les communications sur le lien satellite.

5 Conclusion

L'analyse du système réel offre la possibilité de caractériser le lien satellite. Avoir les caractéristiques du système réel aide à une éventuelle configuration d'un système émulé et s'avère le meilleur moyen pour bien paramétrer ce dernier. L'émulation par la suite se rapprochera au plus du système réel. Dans d'autres cas, le rejeu des profils de flux capturés sur le lien réel est un complément à l'analyse et permet d'effectuer des comparaisons émulées, en conditions réelles.

L'analyse du comportement des timers de retransmission, RTO pour une transmission TCP, révèle une anomalie (très grande valeur de RTO qui peut atteindre les 15 secondes dans certains cas). Ce genre d'anomalies est causé par le grand délai de bout en bout (le RTO est calculé en fonction du RTT), cela peut s'avérer problématique pour les liaisons satellites. Le grand délai de bout en bout, dû au remplissage des buffers, génère de plus en plus de pertes qui ne sont pas forcément liées à des congestions du réseau mais plutôt à la bufferisation (délai de propagation, queuing).

Les méthodes de calcul des timers TCP sont conçues pour les réseaux terrestres mais s'avèrent incompatible pour les réseaux satellite. Actuellement, dans le réseau Internet, les versions les plus utilisées sont CUBIC et NEWRENO. Cependant, TCP n'a pas été conçu pour opérer dans les réseaux à grand délai.

Les tests réalisés donnent des résultats parfois divergents qui rendent une extrapolation d'un modèle mathématique difficile. Certains modèles proposés dans la littérature et basés sur la simulation NS2 sont en contradiction avec nos résultats sur le système réel d'OURSES, le premier de son genre. Ce problème résulte vraisemblablement de l'absence de références disponibles lors de la conception de ces modèles.

En effet, la modélisation dépend essentiellement de la configuration du système pour une architecture donnée. La configuration d'une architecture dite militaire diffère d'une architecture type catastrophe naturelle ou d'un simple accès d'une zone rurale. Les types de données échangées, le débit de transfert ou la priorité de flux changent en fonction de la configuration. La direction de circulation de flux (voie aller ou retour) a de l'importance dans chacune des configurations.

Enfin, l'utilisation de l'adaptation de codage et modulation ACM ne pose pas de difficulté à TCP, qui peut exploiter le surplus de la bande passante en cas d'amélioration du lien, et résiste bien à la réduction de la bande passante en cas de fading.

Devant la complexité des systèmes satellites, la plupart des modèles proposés dans la littérature simplifie le système réel pour mettre en avant certains paramètres qui pourront être utilisés pour configurer le système en fonction de l'application visée. Les résultats obtenus sur le système réel contredisent certains des résultats obtenus à l'aide des modèles.

Pour éviter ces inconvénients, la méthodologie de mesure utilisée sur le système réel nous permet de disposer des informations pertinentes pour la conception d'émulateur ou simulateur et également de valeurs de références pour pouvoir vérifier la pertinence des résultats issus de nos travaux basés sur l'émulation ou la simulation d'un système satellite et ce à différents niveaux d'architecture réseau.

Chapitre 3 : Impact des réseaux satellite sur la couche transport TCP

1 Introduction

Les réseaux satellite ont été introduits pour compléter l'accès aux réseaux terrestres, dans les zones rurales ou zones appelées blanches. Leur utilisation s'est étendue aux diffusions de la télévision, les communications téléphoniques ainsi qu'aux échanges de données. Ces derniers sont assurés généralement par le protocole TCP, qui est le protocole le plus utilisé dans l'Internet d'aujourd'hui. Cependant, TCP rencontre de nombreux problèmes sur les réseaux satellite à cause des caractéristiques de ces derniers (long délai, limitation de la bande passante, données en burst...etc.).

Durant les dernières années, des solutions pour adapter TCP aux réseaux satellite ont été proposées tel que les optimiseurs de performances (Performance Enhancing Proxy PEP) [74]. Bien que les PEPs rapportent des améliorations aux performances applicatives, ils présentent quelques inconvénients. L'utilisation de PEP n'est pas recommandée à cause de la division de la connexion TCP du segment satellite. En effet, cette division va à l'encontre de la sémantique de bout en bout d'une connexion TCP. D'un autre côté, certaines connexions TCP cryptées ne supportent pas les PEPs pour les mêmes raisons.

De nombreuses améliorations de TCP ont été proposées comme l'utilisation de certaines options ou mécanismes d'adaptation aux satellites. De plus, de nouvelles versions de bout en bout ont été introduites. Ces nouvelles versions permettent non seulement d'adapter TCP aux réseaux satellite, d'éviter d'utiliser des solutions d'améliorations dédiées comme les PEPs et d'avoir de bonnes performances applicatives. C'est à ce titre que nous justifions dans ce chapitre de revisiter l'amélioration de TCP sur satellite par les nouvelles versions de TCP qui changent radicalement d'approche.

Ce chapitre est divisé en trois parties. Nous allons tout d'abord présenter les caractéristiques des réseaux satellite Géostationnaires (GEO). Nous présentons par la suite les options, les mécanismes d'amélioration et les nouvelles versions de TCP utilisés. Après une étude comparative des différentes solutions de transport sur les réseaux par satellite, notre objectif est de proposer une solution de transport universelle adaptée tant pour les réseaux satellite que terrestres évitant la mise en œuvre d'équipements ou de logiciels spécifiques sur le segment satellite.

1.1 Caractéristiques des réseaux satellite GEOs

Les satellites GEOs sont les plus utilisés grâce à leur position relative fixe autour de la terre. Une couverture permanente d'une portion de la planète est donc assurée. C'est pour cette raison que les GEOs sont principalement utilisés pour les télécommunications et ce malgré leur coût et leur délai de propagation important.

Le medium de communication satellite se caractérise par certains nombre de particularités qui sont présentées ci-dessus :

- **Délai de bout en bout**

Le délai de propagation dans un réseau satellite GEO est autour de 250 ms. Un délai assez important dû aux caractéristiques de la transmission et de la propagation. En ajoutant le temps de l'émission, le temps d'accès, le temps de passage dans les réseaux sous-jacents et le temps de traitement (contrôle de congestion, retransmission, codage/décodage, cryptage/décryptage...etc.), le délai de bout en bout peut atteindre des valeurs bien supérieures aux valeurs brutes. La Figure 43 présente un Ping Echo Request/ Reply sur un satellite réel (plateforme OURSES [18]). C'est l'évolution du Round Trip Time (RTT) entre un client derrière un ST et un serveur derrière la gateway. On constate un délai moyen aller/retour autour de 660 ms.

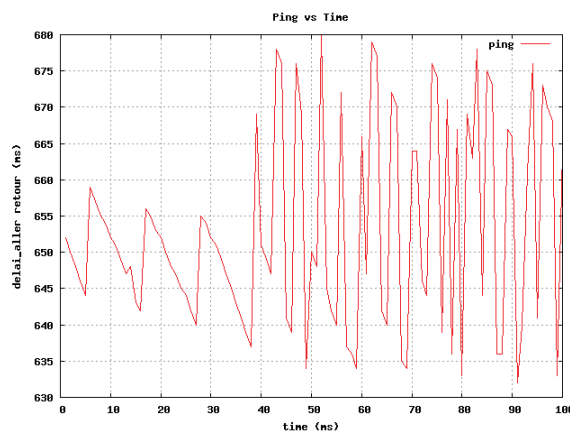


Figure 43 : Délai aller/retour d'un Ping sur un lien satellite réel- Plateforme OURSES

- **Limitation de bande passante**

Le coût de la bande passante reste assez élevé. Ceci reflète les faibles débits offerts, en particulier sur la voie retour DVB-RCS. Toutefois, avec l'exploitation de spectres peu utilisés (Ka), l'amélioration de codage et modulation (ACM) et le multi-spot, il est possible d'offrir un service haut débit.

- **Le SNR et l'atténuation du signal, impact des pertes**

Les erreurs dues au canal de transmission sont courantes sur un lien satellite. Elles sont causées par l'interface air qui demeure variable à cause des conditions de changements du signal radio.

- **L'asymétrie dans le débit entre la voie aller et retour**

Tout en fournissant des offres rentables et efficaces, le débit sur la voie aller DVB-S2 est sensiblement supérieur à celui sur la voie retour DVB-RCS. Cela est dû à la nature de la radiodiffusion satellite.

- **La variation du débit (gigue)**

Pour compenser la variation de canal (fading ou dégradation du signal), le satellite met en œuvre des mécanismes de contrôle, de codage et de modulation adaptatif (ACM) conduisant à une variation de débit pour l'utilisateur.

- **Schéma d'accès en burst**

L'accès au canal n'est pas continu, mais fondé sur une allocation en slot. Ceci produit une transmission de données en burst créant des gigue parfois importantes.

- **La bande passante à la demande**

L'allocation de la bande passante est souvent centralisée. Elle utilise des mécanismes comme le DAMA. Cela implique un délai supplémentaire pour l'obtention de la bande passante demandée. La capacité et le délai peuvent changer brusquement sur durée donnée. D'autres types de systèmes par satellite n'implémentent pas la technique de la bande passante à la demande, c'est le cas de la liaison Inmarsat décrite dans 2.3.2 du deuxième chapitre.

Cependant, certaines de ces spécificités sont communes avec les technologies sans fil terrestres, en particulier les erreurs. Ces dernières sont actuellement réduites grâce à un certain nombre d'améliorations de l'interface radio comme les algorithmes de codage ou les techniques d'ACM. Le DVB-S2 a déjà fait ses preuves et offre au satellite un canal quasiment sans erreurs.

Toutes ces caractéristiques du système satellite DVB-S2/RCS ont un impact sur le comportement de la couche transport TCP. Cependant, nous allons plus particulièrement nous intéresser à l'impact du long délai et du faible débit sur les communications TCP.

Particulièrement, pour TCP sur les réseaux satellite, les erreurs ne sont plus un problème comparées aux grands délais et les faibles débits de données. Des algorithmes standards de TCP comme le *Slow Start*, *Congestion Avoidance* ou l'utilisation des timers RTO ne sont pas conçus pour opérer sur des réseaux satellite à cause du grand délai, mais aussi à cause de l'asymétrie et la variation du débit.

Brièvement, les problèmes principaux de TCP dans un environnement satellite sont [36]:

- Un long RTT (Figure 43) conduisant à :
 - Des phases de *Slow Start* plus longues, impliquant une perte de ressource et de bande passante.
 - Un processus de détection d'erreurs plus long, puisque, par exemple, le timer de retransmission de TCP (RTO) est proportionnel au RTT (voir 4.2.2).
 - La latence introduite par les mécanismes de bande passante à la demande peut augmenter le RTT. Ceci implique une lente évolution de la taille des fenêtres de congestion durant la phase de *Slow Start* avec une possible expiration du RTO.
- Un grand produit débit-délai BDP (Bandwidth Delay Product) impose une large fenêtre de transmission pour chaque connexion TCP. Inévitablement, cela augmente la probabilité des pertes de paquets et peut déclencher des mécanismes de contrôle de congestion TCP. Dans la majorité des cas au sein des réseaux satellite, les mécanismes de contrôle de congestion TCP, interprètent les pertes comme une congestion. Cela a un impact direct sur l'utilisateur qui se traduit par une réduction importante du débit. Les grands RTTs affectent négativement le comportement TCP tant que ce n'est pas spécifié dans ses mécanismes.
- L'asymétrie du lien peut induire une faible capacité à retourner les ACKs [75], causant ainsi une désynchronisation des ACKs et limitant le débit sur la voie aller.

Le produit délai-bande passante (BDP) définit la quantité de données que l'expéditeur doit transmettre à tout instant afin d'utiliser pleinement la bande passante disponible. Le BDP est le produit de la capacité de la bande passante du lien et le (RTT).

$$BDP = bottleneck\ BW \left(\frac{bit}{s} \right) * RTT (s)$$

Équation 2 : Produit bande passante délai BDP

2 L'apport des nouvelles versions TCP

Initialement TCP n'a pas été conçu pour les réseaux satellite. De ce fait, des améliorations lui ont été apportées pour mieux s'y comporter. Plusieurs mécanismes et options ont été proposés pour mieux l'adapter au satellite tel que les PEPs. Ce dernier est devenu une fonctionnalité indissociable de TCP sur satellite.

Avec le développement des réseaux haut débit, de nouvelles évolutions de TCP ont vu le jour. En effet, ces nouvelles versions de TCP conçues pour des réseaux avec un BDP important comme les réseaux filaire, peuvent avoir un impact positif afin d'améliorer les problèmes de TCP sur satellite.

Dans cette partie, nous allons présenter brièvement les nouvelles versions TCP ainsi que les mécanismes et options d'amélioration qui ont été introduites.

2.1 L'évolution de TCP

Le principal élément qui différencie les implantations de TCP réside dans la gestion de la fenêtre de congestion. Plusieurs approches différentes ont été proposées et publiées par l'IETF, sans pour autant devenir un standard unique. Par conséquent, chaque système d'exploitation utilise une version spécifique. Nous allons présenter les différentes versions les plus utilisées et/ou prometteuses pour nos travaux.

2.1.1 Les versions plus utilisées

- **Reno [76]/New Reno [77]:** TCP Reno utilise un contrôle de congestion basé sur les pertes. Il utilise les mécanismes d'incrémenter de la Cwnd par segment chaque RTT pour chaque ACK reçu, et divise la fenêtre de congestion Cwnd (Congestion Window) chaque à événement de perte chaque RTT. Il utilise les mécanismes de control de congestion standard [76]. TCP New Reno améliore la retransmission par rapport à TCP Reno grâce à une amélioration de la phase *Fast Recovery* [77]. Le principe de base de ce dernier est la réémission du 1er paquet non acquitté puis une attente d'un acquittement. Si aucun acquittement n'est reçu, il passe en mode *Slow Start*. TCP New Reno ne quitte le mode de *Fast Recovery* que si l'ACK reçu acquitte tous les segments envoyés.
- **Cubic [78]:** dans ce cas, la fenêtre de congestion évolue selon une formule cubique en fonction du temps depuis le dernier événement de congestion. Cubic n'attend pas les ACKs pour augmenter la taille de la fenêtre de congestion. La Cwnd de Cubic dépendant seulement du dernier événement de congestion. Cubic est la version par défaut sur les systèmes d'exploitation Linux avec un noyau supérieur ou égal à 2.6.19.

- **Compound TCP [79] [80]:** CTCP a été conçu pour s'adapter rapidement à la bande passante disponible tout en ayant un comportement TCP friendly (à la TCP, non agressif). La grande particularité de sa gestion de la $Cwnd$ est d'être basée à la fois sur les pertes et les délais. CTCP gère donc une double fenêtre de congestion ; une classique comme c'est le cas pour TCP Reno et une autre basée sur le délai appelée $Dwnd$ (Delay Congestion Window). Cette dernière est utilisée uniquement durant la phase de *Congestion Avoidance*. Dans un réseau satellite par exemple, l'important délai de propagation peut fausser les timers de retransmission et causer des pertes. La fenêtre de congestion basée sur le délai permet d'estimer le délai de bout en bout et de le prendre en considération. CTCP est actuellement implémenté sur les systèmes d'exploitation Windows Vista, Seven, Server 2008. Un patch est disponible aussi pour les systèmes Windows Server 2003 et Windows XP 64 bits. Un contrôle de congestion basé sur le délai à surtout l'avantage d'anticiper les congestions par la mesure de l'évolution du délai, et donc de limiter les pertes qui sont très pénalisantes sur les réseaux satellite.

$$Window = \min (cwnd + dwnd, Awnd)$$

Équation 3 : Equation d'évolution de la fenêtre de congestion de Compound TCP

Où $Cwnd$ et la fenêtre de congestion basé sur les pertes (comme celle de TCP Reno), $Dwnd$ celle basée sur les délais et $Awnd$ et la fenêtre de réception. $Dwnd$ est calculé comme présenté ci-dessous :

$$dwnd(t+1) = \begin{cases} dwnd(t) + (\alpha \cdot win(t)^k - 1) & , \text{if } diff < \gamma \\ (dwnd(t) - \zeta \cdot diff) & , \text{if } diff \geq \gamma \\ (win(t) \cdot (1 - \beta) - cwnd / 2) & , \text{si une perte est détectée} \end{cases}$$

Équation 4 : Equation d'évolution de la fenêtre de congestion de CTCP basée sur les délais

Où $diff$ et le nombre de paquets dans les files d'attente dans tout le réseau, γ est un seuil de déclenchement, les valeurs α , β et ζ sont des constantes.

2.1.2 Les versions orientées réseaux Satellite/Sans fil

Ces versions spécialisées sans fil ou satellite sont très peu déployées (surtout côté serveur) et sont donc indissociables des solutions de PEPs.

- **Noordwijk [81] :** Ce protocole, développé par l'ESA (Agence Spatiale Européenne), a été conçu et mis en œuvre pour optimiser le transfert des petits paquets (par exemple HTTP) sur une liaison satellite, tout en assurant de performances correctes pour le transfert des gros paquets (par exemple FTP). Son scénario cible est un environnement de communication adoptant la norme DVB-RCS avec DAMA et entre deux entités PEPs (voir la description des PEPs dans 2.2.4).
- **TCP FIT [82] :** TCP FIT est basé sur cette idée d'utiliser plusieurs connexions TCP, comme certaines applications le font pour atteindre un meilleur débit. Cette version effectue une estimation de paquets en sortie et une taille moyenne de la fenêtre de la dernière période,

afin de décider d'augmenter ou diminuer le nombre de connexions virtuelles. La fenêtre de congestion évolue donc de la sorte. Les expériences [83] présentent un comportement TCP friendly de TCP FIT et montrent une amélioration de débit jusqu'au double par rapport à Cubic dans des réseaux LTE simulés.

- **Hybla [84]** : TCP Hybla est une amélioration du TCP classique qui lutte contre les détériorations des performances causées par les longs RTTs typiques aux réseaux satellite. Dans un contexte d'un réseau hétérogène contenant un segment à grande latence (satellite par exemple), Hybla vise à éliminer la pénalisation des segments à grande latence par rapport aux segments avec des petits RTTs. Il se compose d'un ensemble de procédures qui comprend : l'algorithme *Hoe* [84] pour l'estimation de la capacité du canal ; le marquage temporel timestamps; une adoption de la politique SACK et des techniques d'espacement de paquet "PACING".
TCP Hybla implique seulement une modification du côté émetteur. Cela permet une entière compatibilité avec les récepteurs standards.

2.1.3 Les versions orientées réseaux haut débit

- **Illinois** : l'idée de TCP Illinois [85] [86] est d'augmenter d'une manière importante la fenêtre quand la congestion est loin de survenir, et de réduire cette augmentation si une congestion est attendue en utilisant des fonctions du délai moyen passé en file d'attente. Il utilise une fenêtre suivant une fonction concave. Comme CTCP, la variation des délais des ACKs est utilisée pour anticiper les congestions.
- **Hamilton** : [87] C'est une modification des fonctions classiques de TCP. Il n'est pas basé sur les délais mais plutôt sur la durée écoulée depuis le dernier évènement de congestion. Dans des réseaux haut débit, les pertes sont rares. Il est donc possible d'atteindre le débit maximal très rapidement. Dans un réseau sans fil, les pertes sont, par contre, plus courantes. Il n'est donc pas intéressant d'utiliser Hamilton.

Ces nouvelles versions ont été bénéfiques pour l'amélioration des performances applicatives dans certains contextes. Cependant, une autre vision porte sur l'utilisation de mécanismes d'améliorations, indépendamment du protocole même, ou des options liées à certaines versions et à utiliser en cas de besoins spécifiques. Dans ce qui suit, nous allons présenter les mécanismes et options les plus utilisés.

2.2 Les options et mécanismes d'améliorations de TCP

Cette section résume quelques options et mécanismes qui améliorent TCP ou l'adapte pour certain type de réseau tel que les réseaux Wifi et satellite. Certains sont génériques (ex : les options TCP) contrairement à d'autres qui sont spécifiques et qui changent le comportement TCP, ou ont tendances à adapter le fonctionnement de TCP à certaines conditions ou contextes.

2.2.1 Les options TCP

Les options TCP peuvent être intéressantes à utiliser, puisqu'elles ne modifient pas la sémantique du protocole et peuvent être activées/désactivées selon l'utilisation. Elles sont normées dans l'IETF et ont un identifiant IANA [88].

- **TCP Sack** : Les acquittements de TCP sont cumulatifs. Seulement la première valeur du segment non acquitté est retournée par le récepteur. L'option SACK permet à l'émetteur de

savoir quel segment a été bien reçu. SACK [89] notifie à l'émetteur jusqu'à 4 blocs de segments reçus. Si certains segments sont perdus ou désordonnés, cela sera indiqué dans le champ option. Cela est d'autant plus utile que les liens sans fil et satellites, sont souvent l'objet de pertes corrélées. Une amélioration de l'algorithme *Fast Retransmit* est basée sur cette option [90] et certaines versions de TCP l'utilisent pour une meilleure efficacité. Cette option est négociée lors de l'initialisation de connexion.

- **TCP Time Stamp** : La mesure du RTT est importante dans la détermination du RTO, toutefois pour le traditionnel TCP, une seule valeur de RTT est calculée par fenêtre ce qui la rend potentiellement erronée. D'ailleurs il est impossible lors de la réception d'un paquet de savoir quand il a été envoyé. Ces problèmes sont gérés avec l'option Timestamps [91] qui contient deux champs : la valeur d'horloge de l'émetteur et - dans ACK seulement - la valeur reçue dans le paquet qui a été acquitté. Cette fonctionnalité peut être utilisée pour de nombreuses autres fins, notamment pour détecter la réorganisation ou pour identifier un segment reçu qui est acquitté. Cette option doit paraître dans un segment SYN afin d'être envoyée de manière fiable.
- **TCP Window Scale** : Cette option [91] permet d'augmenter la taille de la fenêtre de réception TCP au-delà de sa valeur maximale. Chaque hôte peut utiliser un facteur d'échelle différent pour sa propre fenêtre de réception. La valeur est envoyée au moment de l'initialisation de la connexion. En définissant le champ à 0, un hôte peut permettre à la paire d'utiliser l'option sans affecter sa propre fenêtre de réception. Cette option est très intéressante à utiliser dans les réseaux satellite puisqu'elle permet à la fenêtre de ne pas se limiter et une optimisation du débit.
- **TCP User Timeout** : Dans les versions TCP classiques, la connexion expire (timeout) en raison de segments non acquittés qui sont par défaut et indépendant du réseau, même si les applications peuvent définir leur propre valeur. Cette option [92] permet d'annoncer le délai d'attente timeout au pair, afin de partager la politique de temporisation timeout. Un délai d'attente prolongée permet d'avoir des périodes de silence plus longues, en raison d'économies d'énergie ou pour effectuer une mobilité, alors qu'un timeout court indique que le pair a besoin de garder le contact pour maintenir la connexion. Cela peut être utile dans le cas un serveur occupé.
Les applications peuvent choisir un timeout pour avertir, autoriser /refuser un changement des limites par les pairs.

2.2.2 Optimisations de TCP

Un objectif commun à toutes les implémentations TCP est d'être rapide pour fournir des communications efficaces et de réduire la durée d'établissement d'une connexion. Ci-dessous un ensemble de techniques qui ont été développées pour cet objectif. Il faut savoir que ces techniques ne sont pas forcément standard.

- **Initial Window** : La phase de *Slow Start* TCP est nécessaire pour atteindre un débit acceptable dans le réseau, mais ajoute également de la latence au tout début de la connexion. En 2002, la taille initiale de la fenêtre de congestion a été révisée [93] permettant une augmentation jusqu'à 4 segments. Maintenant que les sites Web sont de plus en plus complexes, il est temps de changer la norme à nouveau. Google a demandé à

l'IETF d'augmenter la fenêtre initiale à 10, car il a été observé que c'était suffisant pour télécharger 90% des pages Web dans un seul RTT [94].

Les conséquences en matière de gestion de la congestion semblent négligeables dans la plupart des environnements. Une fenêtre plus petite peut être utilisée si elle n'est pas le cas. Si ce changement est mis en œuvre dans la pile TCP, cela améliore sensiblement le comportement HTTP dans les réseaux à long délai, tels que les réseaux par satellite.

- **Quick-Start** : Les algorithmes de congestion prennent un certain temps pour s'adapter à la bonne Cwnd, ce qui peut conduire à des problèmes sur un support variable comme un réseau sans fil et en particulier en cas de mobilité verticale. Le Quick-Start [95] est un mécanisme Cross Layer entre IP et TCP permettant une demande de taux explicite vers tout le chemin d'accès.
- **Early retransmit** : [96] permet de déclencher le *Fast Retransmit*, dans certaines conditions, en particulier lorsque peu de segments sont échangés, pour réduire le nombre duplicate ACK nécessaires à une retransmission rapide (normalement 3). Cela permet à utiliser le *Fast Retransmit* pour récupérer les pertes de segments qui, autrement, nécessitent un grand délai de retransmission timeout. En d'autres termes, les connexions récupèrent depuis une perte plus rapidement, ce qui améliore la latence.
- **Limited Transmit** : Dans les environnements où le RTT est élevé, le temps perdu peut retarder significativement une connexion. Lorsqu'une réorganisation arrive, TCP attend 3 duplicate ACK pour activer le *Fast Retransmit*, mais l'arrivée des deux premiers duplicate ACK indique que le nombre de paquets en circulation a diminué (données acquittées). Il est possible pour le protocole TCP d'envoyer deux nouveaux segments si cela est permis par le Awnd (Advertised Windows) du pair.

2.2.3 Mécanismes de ré-ordonnement

Parfois, TCP ralentit en supposant qu'une perte a eu lieu mais le segment manquant peut-être enfin reçu s'il a été désordonné par le réseau. Certains mécanismes ont été proposés pour corriger ce genre d'erreur d'interprétation.

- **Le protocole Eifel** : des faux timeouts, une réorganisation ou une duplication de paquets peut conduire à un changement inapproprié dans la politique de congestion, d'un simple *Fast Retransmit* à un retour problématique dans le *Slow Start*. Ce protocole [97] [98] utilise l'option Timestamp pour se protéger contre les mécanismes erronés de contrôle de congestion.
Eifel vérifie les timestamps lors de la réception d'un ACK afin de les distinguer. Quand un ACK reçu contient un Timestamp plus ancien que celui correspondant à la modification de la politique de la congestion, l'expéditeur comprend son erreur et peut la corriger et en fonction de l'implémentation, il peut essentiellement revenir à l'état et la taille de la fenêtre utilisé auparavant.
- **DSACK** : Son objectif est similaire à Eifel, il permet la correction des mécanismes de contrôle de congestion incorrects. A la réception d'un segment dupliqué, DSACK [99] indique à l'expéditeur si un ACK a été envoyé. Il est basé sur l'option SACK qui indique des blocs de segments reçus en mettant le bloc reçu en double à la tête de la liste. De nombreux cas sont possibles, mais en utilisant à la fois l'ACK cumulatif et le premier bloc "SACKé", il

sera facile de savoir quel bloc a été acquitté. Le principal avantage de cette technique est qu'elle n'a besoin que de l'option SACK et n'ajoute qu'aucune autre information.

Toutes les options vues auparavant présentent des lacunes : des problèmes d'implémentation, de faisabilité ou d'inadaptation aux réseaux NGN. Dans un contexte de réseaux hétérogènes, le segment satellite peut pénaliser les autres segments sous-jacents. Par exemple, une valeur de la "Window Scaling" peut s'avérer intéressante pour améliorer TCP sur le segment satellite, mais cela peut engendrer des dégradations sur les segments terrestres adjacents. De ce fait, ces options peuvent apporter des améliorations au réseau satellite mais tout en pénalisant les réseaux d'interconnexion.

L'objectif est d'utiliser un mécanisme qui améliore le réseau satellite tout en évitant d'influencer les réseaux d'accès sous-jacents. La philosophie des PEPs repose sur cette idée en isolant le segment satellite du reste du réseau, pour que ses caractéristiques n'affectent pas l'ensemble du réseau.

2.2.4 PEP

Les PEPs [100] (Performance Enhancement Proxy) sont proposés de plus en plus dans les services offerts par les fournisseurs d'accès satellite. Le développement est motivé par les améliorations apportées aux performances de TCP comme l'accélération de certaines applications (HTTP et DNS par exemple).

Il y a plusieurs implémentations de PEP. Nous allons nous focaliser sur une seule solution PEP pour réaliser nos tests, à savoir PEPsal.

Le choix s'est porté sur PEPsal à cause de son implémentation open source, de la documentation fournie ainsi pour les études réalisées auparavant. PEPsal [101] est disponible pour les systèmes Linux sous la licence GNU GPL.

PEPsal est un PEP intégré basé sur les techniques de TCP-splitting (division de la connexion) [74]. La philosophie des méthodes de division de connexion est de séparer la portion satellite (puisque les protocoles classiques souffrent du grand BDP [100]) du reste du réseau. En suivant ce principe, la connexion TCP est divisée en deux différentes connexions. La première, sur le réseau terrestre depuis le client vers la machine PEPsal et adoptant un TCP standard, la deuxième depuis la machine PEPsal vers le récepteur adoptant TCP Hybla.

2.2.5 Explicit Congestion Notification

La notification explicite de congestion (dite ECN, « Explicit Congestion Notification ») est une extension TCP qui permet de signaler la congestion du réseau avant que la perte de paquets ne se produise [102].

Généralement, TCP augmente son débit tant qu'aucune perte n'a été détectée. Ce n'est que lorsqu'une perte est détectée que le débit est "modéré". ECN permet explicitement donc au routeur de signaler un début de congestion avant de commencer à perdre des paquets.

Afin d'éviter les congestions, l'émetteur peut interroger les routeurs sur son chemin [102]. Premièrement, les hôtes TCP mettent des marquages (flags) correspondants dans les segments SYN-SYN/ACK pour indiquer leur capacité à utiliser l'ECN. Ensuite, l'émetteur peut remplir le champ ECT

(ECN Capable Transport) dans son en-tête. Si un routeur sur le chemin rencontre ou attend une congestion (généralement utilisé dans la politique de RED AQM [103]), il peut mettre à jour l'ECT pour indiquer cette congestion. Le récepteur doit retourner la notification à l'émetteur ; il utilise le flag ECN-Echo dans l'en-tête TCP du segment ACK. Un émetteur recevant un tel segment réduit sa Cwnd pour limiter les congestions et met le flag Cwnd réduit (Congestion Window Reduced) dans son prochain segment.

ECN est négocié pour chaque connexion, il n'est utilisé que lorsque les deux hôtes échangeant des données signalent leur intention de l'utiliser.

Après avoir présenté toutes ces versions de TCP, les options et les mécanismes utilisés pour optimiser ces performances sur un réseau satellite, nous allons réaliser une comparaison de ces solutions en utilisant l'émulateur SATEM afin d'analyser le comportement de TCP ainsi que l'impact du réseau satellite (et ses caractéristiques).

3 Evaluation de l'impact des protocoles TCP sur un réseau satellite

Afin de comparer des mécanismes différents de TCP (par exemple le contrôle de congestion basé sur les délais et le contrôle de congestion basé sur les pertes), nous allons utiliser une approche boîte noire, c'est-à-dire mesurer de bout en bout le fonctionnement de TCP. Les mesures que nous réalisons avec cette approche permettent d'analyser le mode de fonctionnement de bout en bout TCP au sein du réseau satellite. Les tests d'évaluation sont effectués avec l'émulateur SATEM.

3.1 Plateforme de tests

Pour la réalisation des tests de performance TCP, nous avons mis en œuvre la plateforme de tests illustrée par la Figure 44.

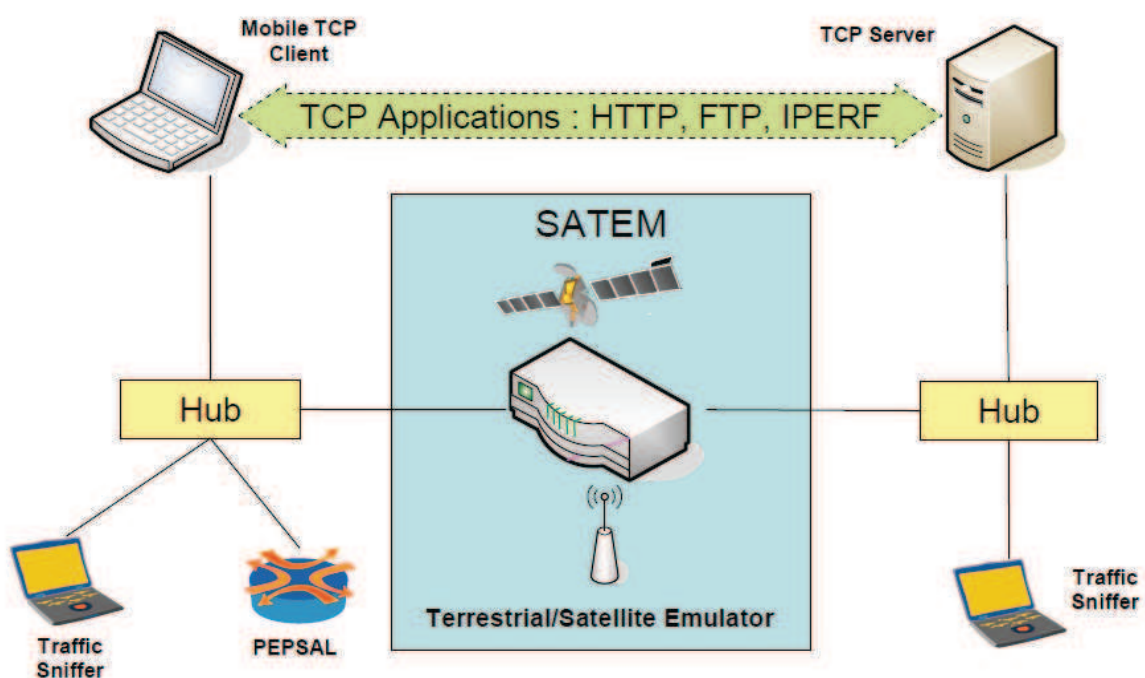


Figure 44 : Plateforme de test utilisant SATEM

La plateforme comporte :

- Un client et un serveur d'applications (HTTP, FTP, IPERF), installés sur des PC Linux (pour les versions New Reno, Cubic, Hybla et Westwood) et Windows 7 (pour Compound TCP).
- PEPsal installé sur une machine Linux.
- SATEM pour émuler le réseau satellite et/ou hybride.
- Outils de capture et TCP (Wireshark) et Web100.

Le Window Scalling est activé par défaut sur toutes les versions Linux (Cubic, Hybla, New Reno et Westwood). Cela permet de palier au problème de la limitation du débit par la taille de la fenêtre de congestion, et ainsi, voir le débit optimisé à son maximum. A travers la formule suivante illustrant le calcul de la bande passante, on peut conclure l'utilité de cette option décrite dans 2.2.

$\text{Bandwidth} = \text{Window size} / \text{RTT}$

3.2 Méthodologie de mesure/analyse

Les métriques suivantes ont été choisies pour évaluer les performances de TCP :

- L'évolution des numéros de séquence (*Sequence Number*): Cette métrique représente l'évolution des paquets transmis et est utile pour le suivi de l'efficacité et de la régularité d'une connexion TCP. Elle fournit une information sur les segments perdus, les coupures de connexion et la quantité des données transmises. La comparaison des numéros de séquence est intéressante car elle permet de comparer facilement le comportement de différentes versions TCP.
- La taille de la fenêtre de congestion *CWND* utilisée (n'est pas disponible pour CTCP sous Windows 7).
- Le délai de bout en bout : informe sur l'état des buffers.
- Le débit applicatif : définit la quantité moyenne des données transmises à l'instant t.

3.3 Résultats et analyses

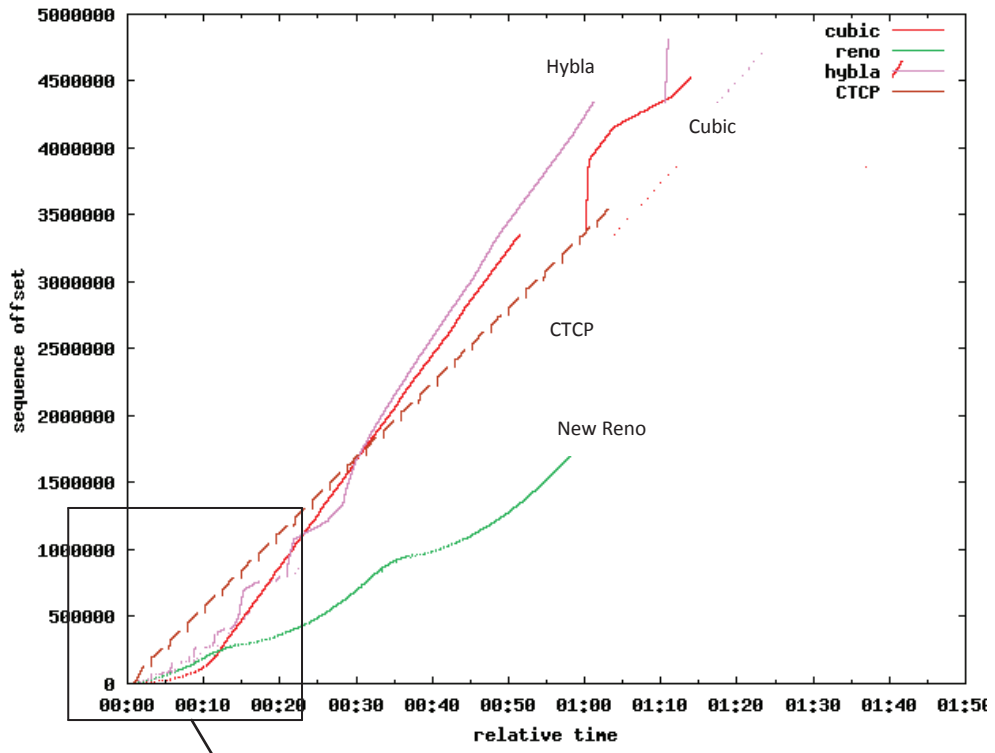
Les mesures réalisées dans cette partie ont été effectuées avec une seule connexion TCP sur un lien vide. L'objectif est d'observer précisément le comportement du contrôle de congestion de TCP sur satellite.

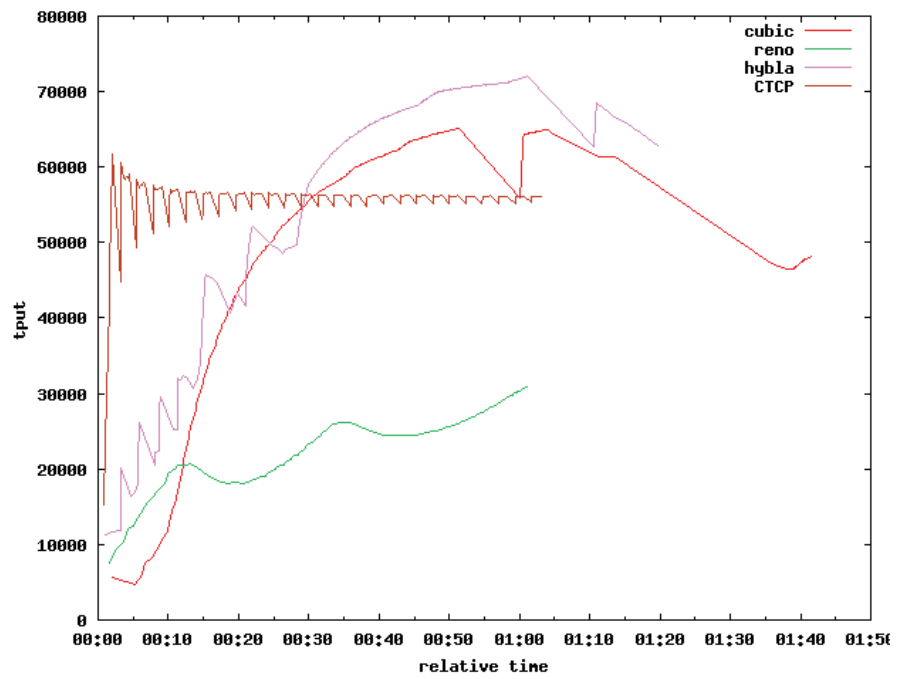
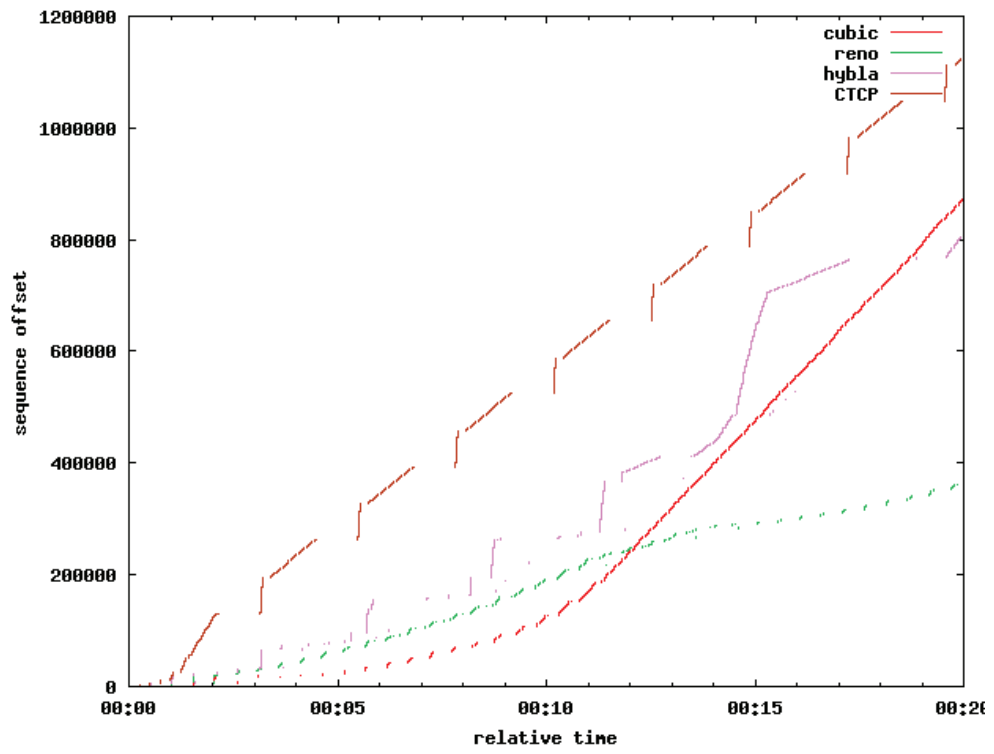
3.3.1 Tests IPERF

Dans ces tests, nous allons mesurer par l'évolution des numéros de séquence la capacité d'émission de données d'une connexion TCP. Le délai de bout en bout est mesuré d'une manière applicative, c'est-à-dire, le délai de propagation plus celui passé dans les buffers d'émission. Le débit mesuré est celui applicatif.

a) La configuration du système

Le lien satellite est émulé avec 250 ms d'OWD (One Way Delay) et 512 kbit/s ou 2 Mbit/s de bande passante selon le test. Le taux d'erreur est laissé à 0 (pas de fading). Les résultats sont présentés ci-dessous :





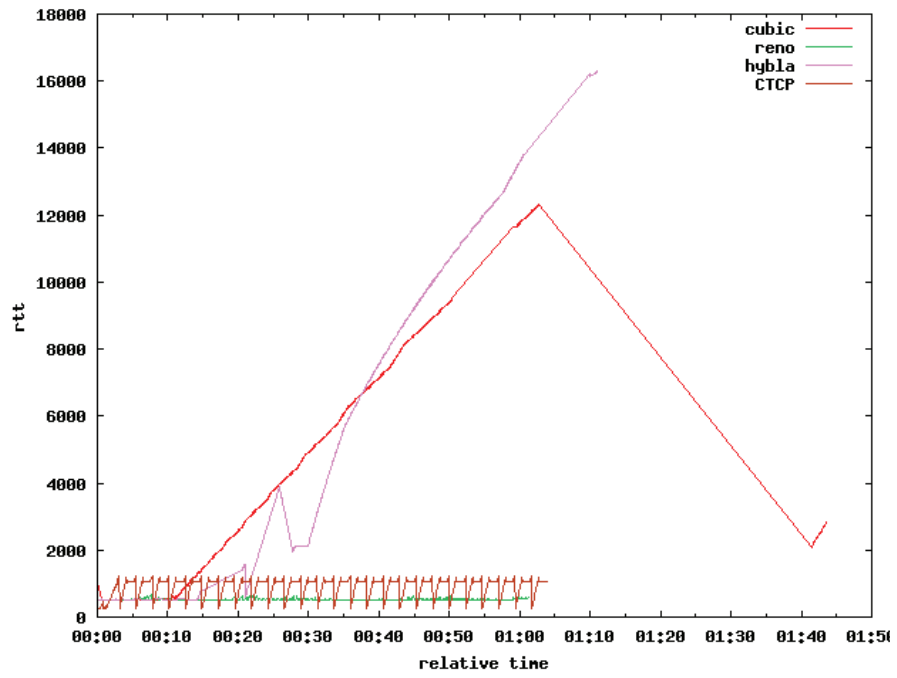


Figure 45 : Numéros de séquence, débit et RTT sur un lien 512 kbit/s

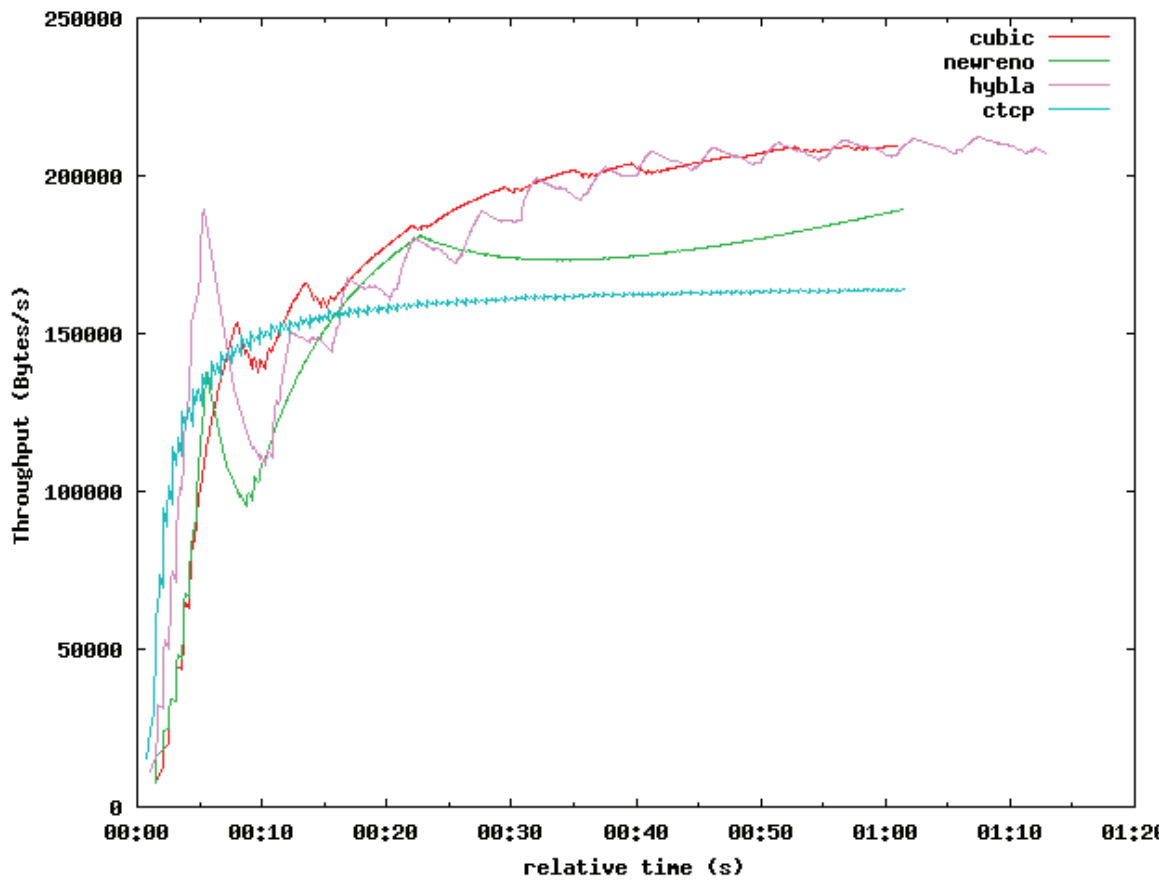
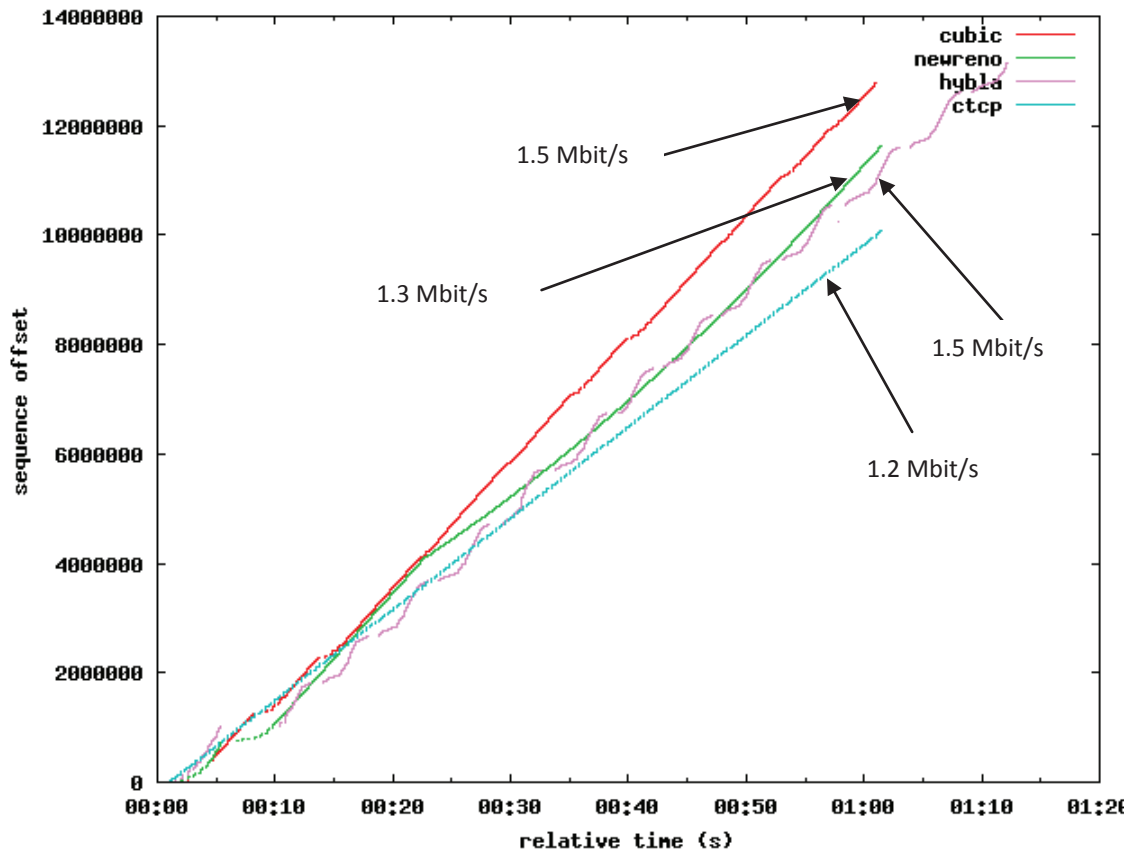


Figure 46 : numéros de séquence et occupation de la bande passante sur un lien à 2 Mbit/s

b) Analyse

Dans la Figure 45, Cubic, Hybla et CTCP montrent de meilleures performances par rapport à New Reno. CTCP présente une amélioration de plus de 50 % de débit par rapport à New Reno.

Deux périodes sont apparentes, une période de 0 à 30 s où CTCP montre une meilleure utilisation du débit disponible et une deuxième période de 30 s jusqu'à la fin où les protocoles agressifs comme Hybla et Cubic finissent par mieux occuper le lien.

CTCP est la plus stable des versions, puisque son contrôle de congestion est basé sur la gestion des délais et des pertes, ses performances sont meilleures que les autres versions, grâce à une meilleure estimation du RTT.

Les premières secondes d'Hybla semblent un peu chaotique, due à son agressivité. Par conséquent, cela déclenche les mécanismes de contrôle de congestion (Nous pouvons observer quelques trous dans la progression des numéros de séquence).

Pendant ces 30 premières secondes, le débit de CTCP présente approximativement 30 % de plus que celui de Hybla et Cubic. Ces deux derniers rattrapent CTCP à la seconde 30.

Après $t=30$ s, Hybla et Cubic poursuivent le transfert avec une intensité comparable à CTCP. Ce comportement agressif cause des problèmes 20 secondes plus tard, la transmission sera interrompue durant 10 secondes.

Par conséquent, les flux IPERF ne s'arrêteront pas à 60 s (durée du test) à cause des effets secondaires de congestion et bufferisation (timeouts, retransmissions). Par contre, CTCP montre un meilleur comportement et ne rencontre pas ce genre de problème.

En termes d'utilisation et d'occupation de la bande passante, CTCP converge rapidement vers le bon débit.

Hybla et Cubic prennent une dizaine de secondes pour atteindre le même débit, puis le dépassent grâce aux algorithmes plus agressifs. Par contre, New Reno n'est pas assez agressif, ce qui conduit à la plus mauvaise occupation du canal.

Concernant les retransmissions, Hybla retransmet plus de 2 % des paquets, contre 3 % pour Cubic, 1% pour New Reno et moins de 0.1 % pour CTCP. Le taux de retransmissions de New Reno est faible, mais cela est dû au nombre total des paquets transmis par ce dernier et qui vaut approximativement le 1/3 des paquets transmis par les autres versions. De ce fait, CTCP présente de très bonnes performances et a la meilleure utilisation de la bande passante ce qui est important dans ce contexte de ressources limitées en satellite.

La Figure 46 montre respectivement l'évolution des numéros de séquence et l'occupation de la bande passante de 4 versions différentes TCP avec les mêmes conditions de délais (250 ms d'OWD) et de 2 Mbit/s de bande passante. Tous les flux ont presque le même comportement avec un léger avantage de performance pour Cubic. Il utilise les 2 Mbit/s mieux que les autres versions, et atteint une valeur maximale approximative de 1.5 Mbit/s.

Hybla a aussi le même comportement que Cubic mais avec plus de fluctuations. Ce dernier s'adapte mieux aux réseaux large et lent (Long Fat Network) et ses performances restent bonnes avec un débit de 1.5 Mbit/s. Cependant, de nombreuses pertes sont ont été remarquées dans les premiers paquets jusqu'au débit stationnaire.

New Reno utilise aussi bien la bande passante et atteint le débit de 1.3 Mbit/s, mais avec pas mal de difficultés dans les 20 premières secondes. Il n'est pas conçu pour les réseaux satellite mais arrive tout de même à mieux exploiter les 2 Mbit/s que les 512 kbit/s.

Hybla qui est conçu pour les réseaux satellite et Cubic avec leurs agressivités dépassent CTCP. Cependant, il faut signaler l'instabilité des Hybla surtout pendant les 20 premières secondes.

Concernant CTCP, il atteint le débit de 1.2 Mbit/s. Nous pouvons dire que CTCP présente une limitation de débit quand le BDP est grand. Ceci dit, un débit de 1 Mbit/s pour une connexion TCP sur un système satellite est exceptionnel.

Pour les retransmissions, Hybla retransmet plus de 7 % des paquets, contre moins de 1 % pour Cubic, 1% pour New Reno et moins de 0.1 % pour CTCP. Hybla présente des difficultés sur les réseaux LFN (Long Fat Network). Cubic subit moins de retransmissions puisqu'il est recommandé pour les réseaux LFN [104]. Cependant, CTCP présente de très bonnes performances en termes de retransmission de paquets et a la meilleure utilisation de la bande passante.

A première vue, on peut dire que CTCP est le moins bon dans cette configuration, mais si on regarde mieux l'évolution des fenêtres et le débit dans les premières secondes, on remarque que CTCP atteint le plus rapidement le débit stationnaire. De plus, CTCP est la plus stable des versions, il concède moins de pertes et une évolution du débit quasi linéaire.

La gestion de fenêtre de congestion basée sur les pertes et les délais de CTCP lui permet d'avoir l'adaptation la plus rapide, avec un peu moins de débit mais une évolution stable.

Après avoir fait ces premières constatations, nous avons réalisé des expériences afin de mieux observer le comportement de ces versions avec des petits paquets. Ces nouveaux tests reflètent surtout le transfert de pages web, puisque ces dernières ont une taille moyenne de 10 Ko jusqu'à 1 Mo dans l'Internet. La Figure 47 présente les mesures effectuées dans l'Internet sur les 15 dernières années [105] (De 1995 jusqu'à 2011).

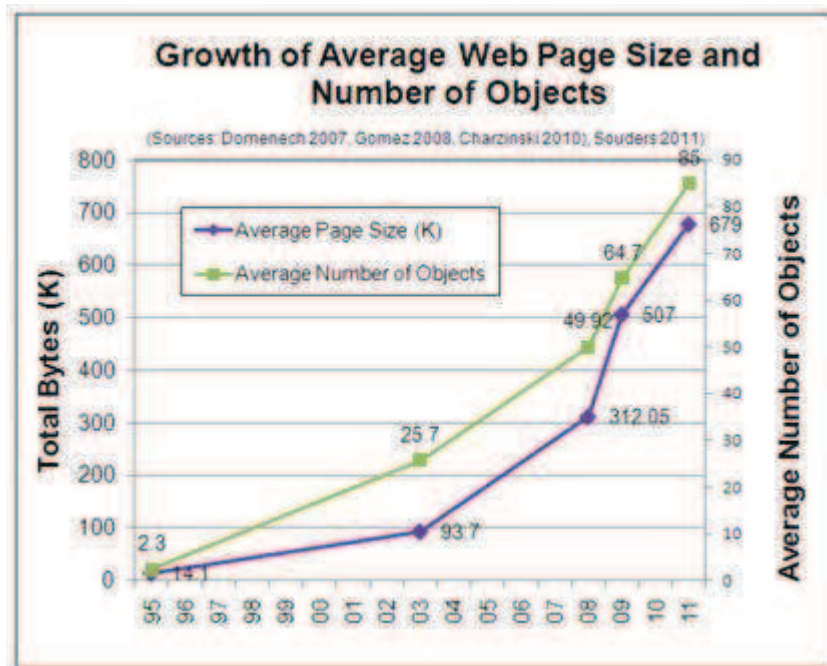


Figure 47 : Taille moyenne des pages web les 15 derniers années [105]

3.3.2 Les tests HTTP

a) La configuration

Le lien satellite est émulé avec 250 ms d'OWD (One Way Delay) et 512 kbit/s ou 2 Mbit/s de bande passante selon le test. Le taux d'erreur est laissé libre (pas de fading). Nous utilisons Wget sous Linux pour le téléchargement des pages. Les pages HTML font 320 Ko, 1 Mo et 10 Mo.

Nous utilisons 3 fichiers. 320 Ko qui représentent la taille moyenne d'une page web [105], 1 Mo et 10 Mo pour les grands éléments, qui peuvent être une photo non compressée ou une petite vidéo. Les résultats sont présentés ci-dessous :

- Comparaison du débit applicatif HTTP des versions TCP (Figure 48).
- Comparaison de la durée du téléchargement HTTP selon les versions TCP (Figure 49).
- Comparaison de la capacité des versions TCP à atteindre un bon débit (Figure 50).
- Comparaison de la capacité d'occupation du canal des différentes versions TCP (Figure 51).

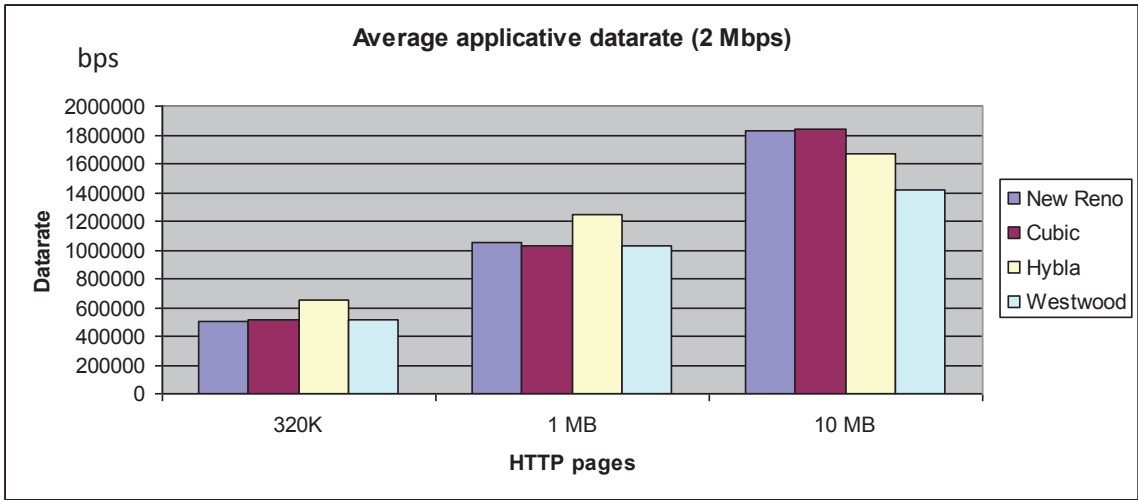
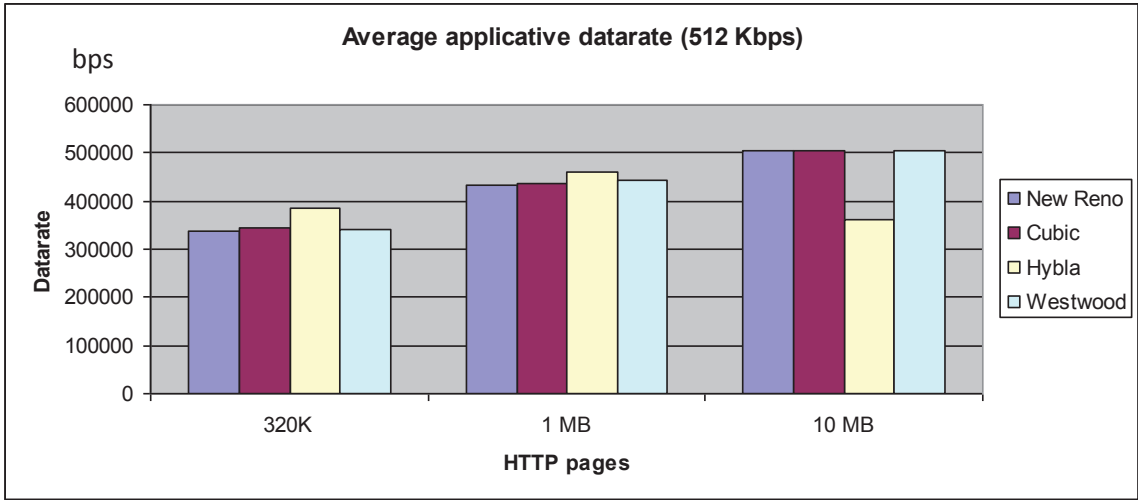


Figure 48 : Tests HTTP/TCP : Comparaison des débits applicatifs (512 kbit/s & 2 Mbit/s)

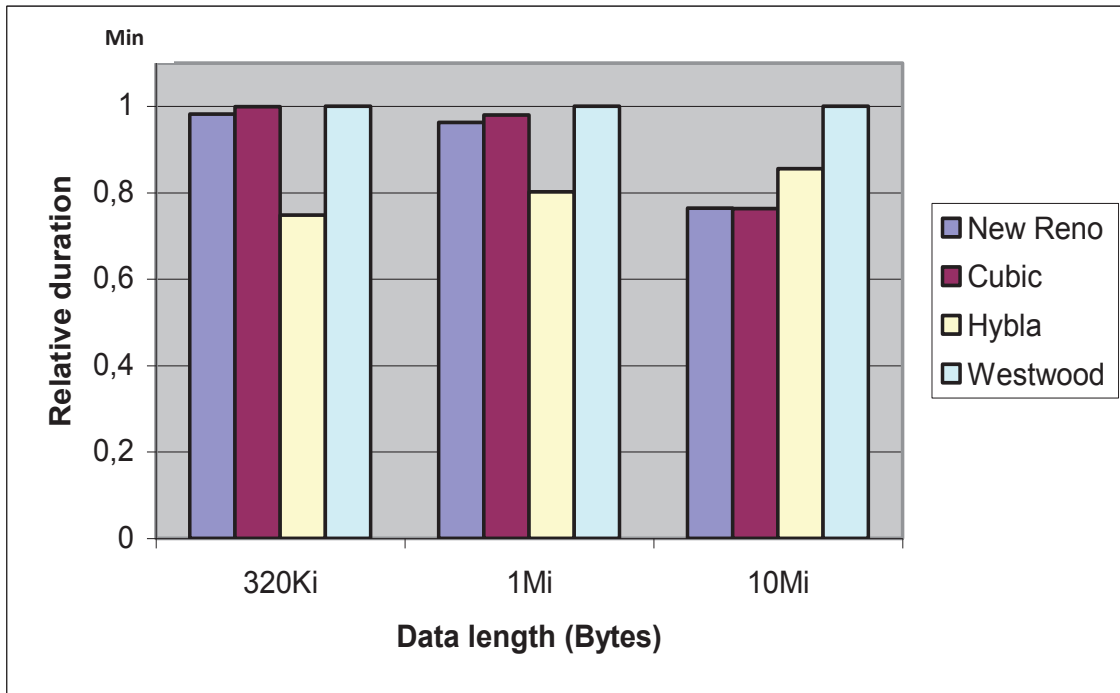


Figure 49 : Tests HTTP/TCP : comparaison de durée de téléchargement (Sat @ 2 Mbit/s)

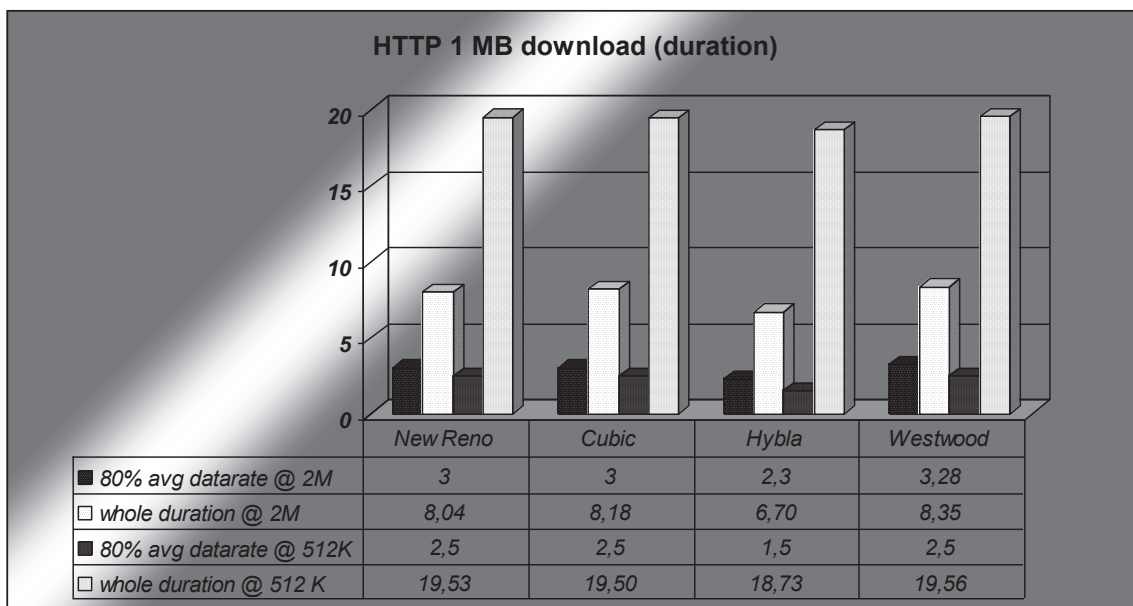


Figure 50 : Tests HTTP/TCP : capacité à atteindre de bons débits

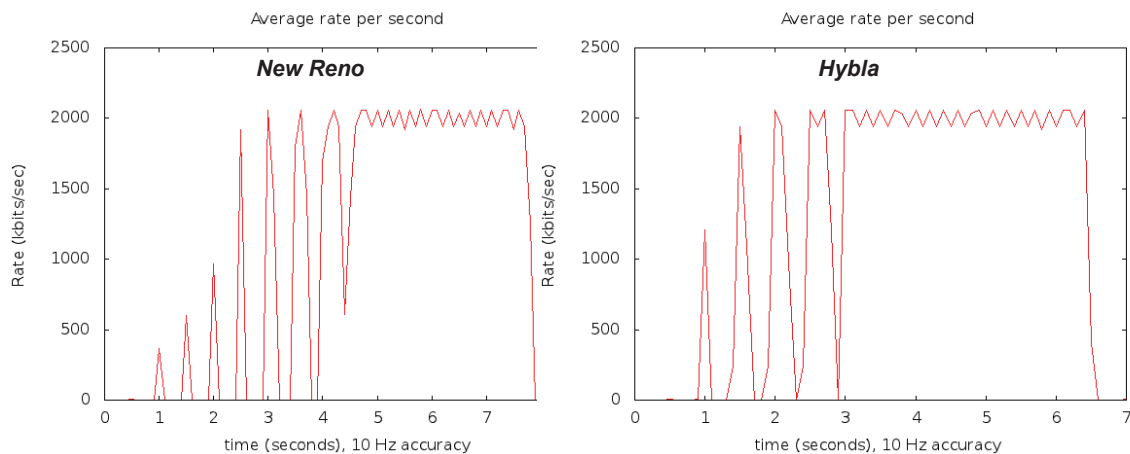


Figure 51 : Tests HTTP/TCP : Occupation de la bande passante (Sat @ 2 Mbit/s)

b) Tests http pour les petites pages Web

Afin de voir le comportement pour des petites pages, nous avons réalisé un ensemble de tests avec des pages de 10 ko jusqu'à 1 Mo. Les pages sont envoyées 20 fois. La configuration de SATEM est de 250 ms/512 kbit/s. Nous avons gardé la même configuration et utilisons un client serveur développé en python afin d'utiliser Compound TCP sous Microsoft Windows 7. Les résultats sont présentés dans les graphes suivants :

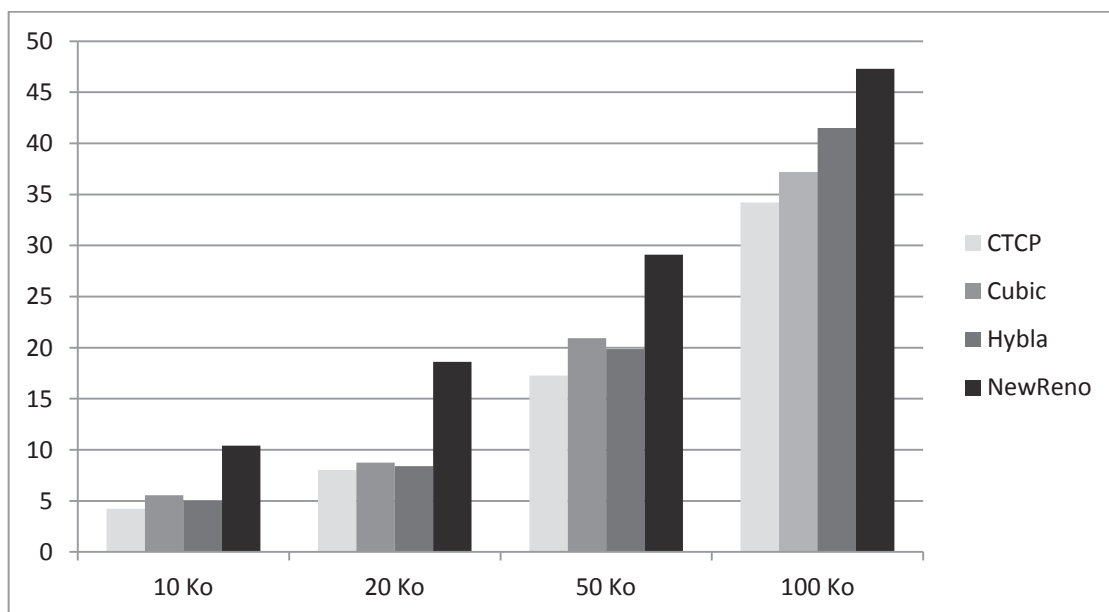


Figure 52 : Durée d'envoi de 20 fois 10, 20, 50 et 100 ko

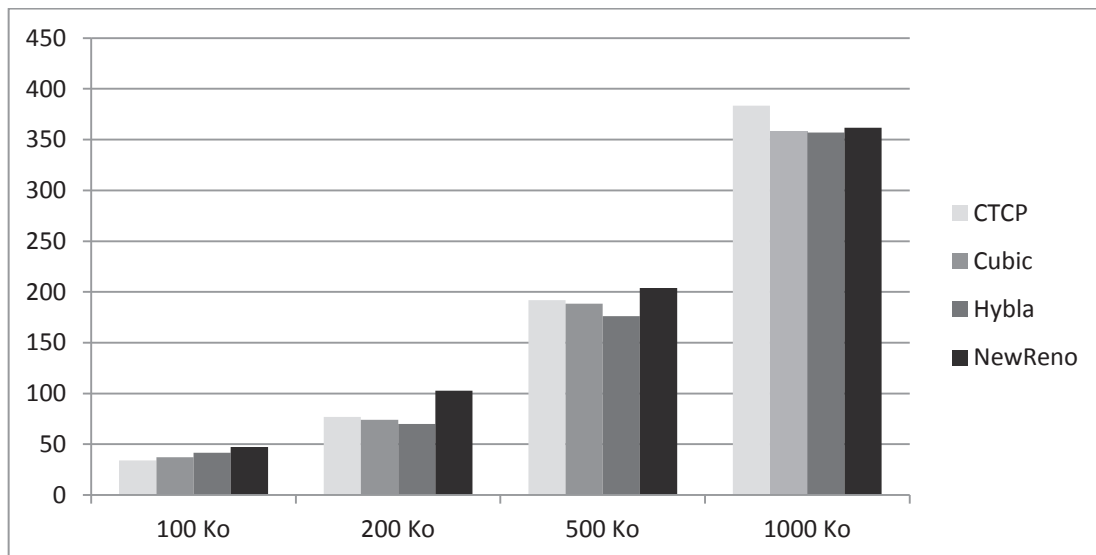


Figure 53 : Durée d'envoi de 20 fois 100, 200, 500 et 1000 Ko

c) Analyse

La Figure 48 et Figure 49 montrent les débits applicatifs ou la durée de téléchargement HTTP, on remarque que Westwood n'améliore pas le comportement de New Reno. Cubic de son côté a les meilleures performances. Dans certains cas, (page de 10 Mo @ 2 Mbit/s) Westwood devient encore moins performant, à cause de son agressivité.

Hybla atteint les meilleures performances : il est plus rapide, approximativement de 20 % de plus que les autres versions pour des pages d'une taille inférieure à 1 Mo. A partir de cette taille, ses performances baissent (à partir des pages de 3 Mo). Le pire résultat est signalé pour les pages de 10 Mo pour Hybla. Nous interprétons cela comme un effet secondaire de l'agressivité Hybla, ce qui conduit à la congestion (pertes de paquets, de timeouts et des retransmissions) et une perte de l'utilisation de la bande passante (en particulier pour de faible débit, comme illustré par 512 kbit/s), similaire à Westwood.

La Figure 50 montre la capacité à atteindre le plus grand débit, quand la bande passante est à 2 Mbit/s, Hybla est plus rapide que Cubic, New Reno ou Westwood (ce dernier est le plus mauvais). Pour les petites bandes passantes, les améliorations d'Hybla sont encore plus significatives, 40 % plus rapide que Cubic ou New Reno.

La Figure 51 illustre l'agressivité d'Hybla en le comparant à New Reno. Dans cet exemple, avec 2 Mbit/s de bande passante, l'amélioration du débit est autour de 33%, en tenant compte des critères de capacité d'occupation du canal. http/Hybla nécessite 3 s pour atteindre le maximum physique de la bande passante, contrairement à New Reno qui met 4,5 s

Ces résultats montrent qu'Hybla, comme attendu, est plus efficace que les versions traditionnelles dans le cas où la quantité de données est petite/moyenne. Pour de grande quantité de donnée (plusieurs Mo), cela est moins évident, en raison d'un manque de stabilité de l'algorithme Hybla. En outre, Hybla est d'autant plus efficace quand la bande passante est grande.

Dans la Figure 52, nous remarquons que Jusqu'à 100 Ko, c'est CTCP qui est le plus rapide et réactif. Hybla et Cubic ont un comportement similaire, avec un petit avantage pour Hybla pour les petites tailles. New Reno est moins efficace (comme prévu).

Dans la Figure 53, pour les grandes pages (à partir de 200 ko), CTCP est rattrapé par Cubic et Hybla. Cela confirme les résultats obtenus dans les premiers tests TCP de cette partie. Pour 1000 Ko, CTCP prend un grand retard du fait d'une meilleure performance d'Hybla et Cubic après la seconde 30. Une chose remarquable est le retour de New Reno, qui lui à long terme arrive à avoir une stabilité d'envoi. New Reno arrive à être stable au bout de plusieurs dizaines de secondes de transmission. Cependant, si une perte venait à être détectée (ce qui fréquent sur un réseau satellite), le redémarrage est long et cela peut s'avérer très pénalisant.

Ces résultats placent CTCP comme meilleure solution pour la navigation Web sur satellite (sauf si les pages sont très grandes). Il est réactif, rapide et atteint le débit stationnaire très rapidement.

3.3.3 Tests FTP

a) La configuration du système

Le lien satellite est émulé à 250 ms d'OWD et 512 kbit/s de bande passante, Le taux d'erreur est laissé libre (pas de fading). On transfère un fichier d'une taille moyenne de 7 Mb par FTP. Les résultats sont présentés ci-dessous. Dans chaque cas de figure, nous testons toutes les versions sur la figure à gauche. Puis nous ne gardons que Cubic, Hybla, CTCP et la solution PEP (suppression de New Reno) sur la figure à droite pour plus de clarté.

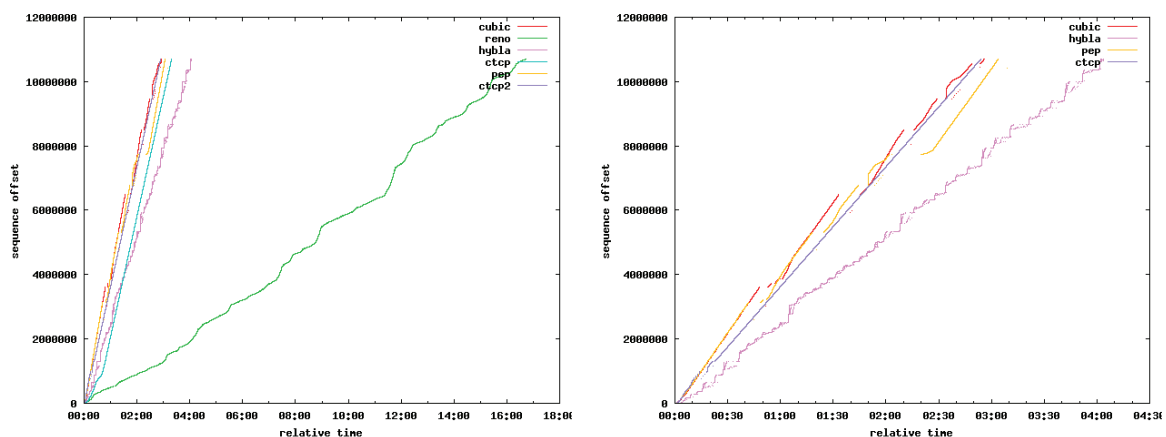


Figure 54 : Comparaison des numéros de séquence (a) et sans New Reno (b) d'un transfert FTP sur un lien 512 kbit/s

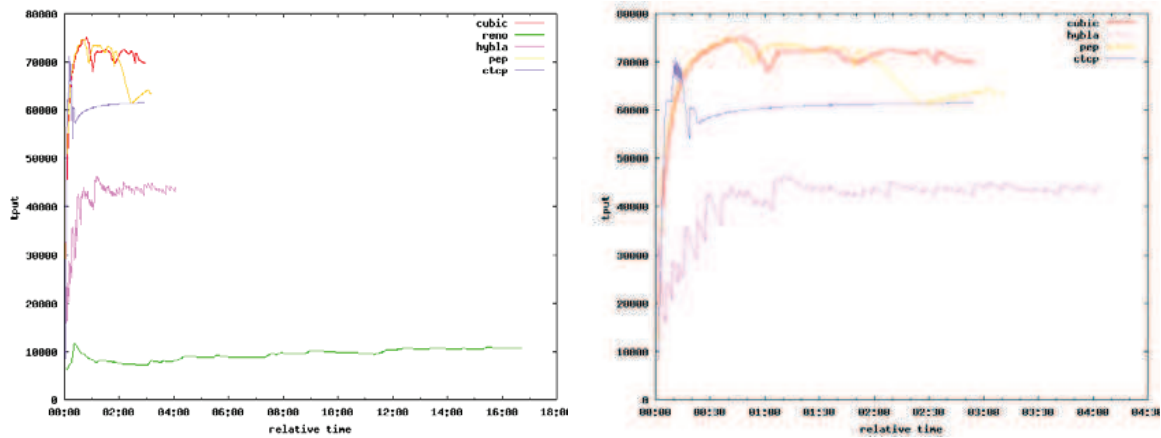


Figure 55 : Comparaison des débits (a) et sans New Reno (b) d'un transfert FTP sur un lien 512 kbit/s

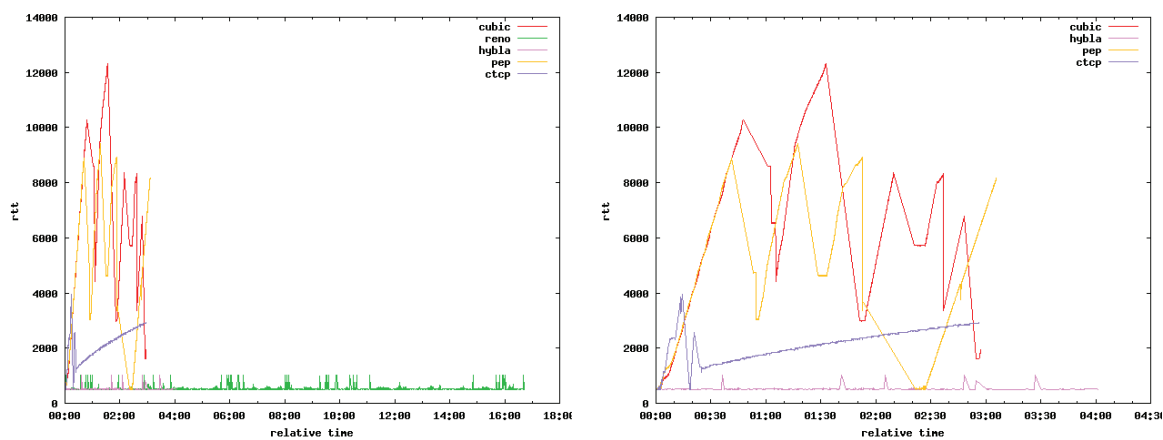


Figure 56 : Comparaison des RTTs (a) et sans New Reno (b) d'un transfert FTP sur un lien 512 kbit/s

b) Analyse

Dans cette section, Nous comparons un transfert FTP en utilisant différentes versions de TCP sur un lien satellite émulé. Les numéros de séquence dans la Figure 54 montrent un mauvais comportement de New Reno. Ce dernier n'est pas conçu pour fonctionner sur des liens satellite. Les autres versions de TCP ont de bonnes performances. Les performances sont très bonnes pour CTCP qui est meilleur que Hybla et plus stable que les solutions PEP et Cubic.

Dans la Figure 55, nous confirmons l'analyse sur l'infériorité de New Reno. CTCP a un débit plus stable comparé à Cubic et PEP. La différence de débit n'est pas très significative (0.5 Mbit/s pour Cubic et PEPsal et 0.48 pour CTCP pour CTCP avec moins de pertes et plus de stabilité).

3.3.4 Hétérogénéité des versions client/serveur

a) La configuration

Cette section compare les différentes versions quand elles sont utilisées face à face, c'est-à-dire que la version TCP du client et du serveur est différente. Cela permet de répondre à certaines questions sur le comportement de TCP, en particulier quelle version de TCP prime sur l'autre ? Et en particulier, est ce que la version du serveur a un impact sur celle du client ?

Le lien satellite est émulé à 250 ms d'OWD et 512 kbit/s de bande passante, Le taux d'erreur est laissé à 0 (pas de fading).

Les graphes ci-dessous présentent les expériences réalisées avec les différentes combinaisons possibles entre New Reno, Cubic, Hybla et CTCP. Les graphes à droites présentent les résultats en fixant l'émetteur et changeant le récepteur. Les graphes de gauche présentent le cas contraire. Chaque graphe donne la performance d'une configuration homogène du protocole (même version pour l'expéditeur et le récepteur) comme référence.

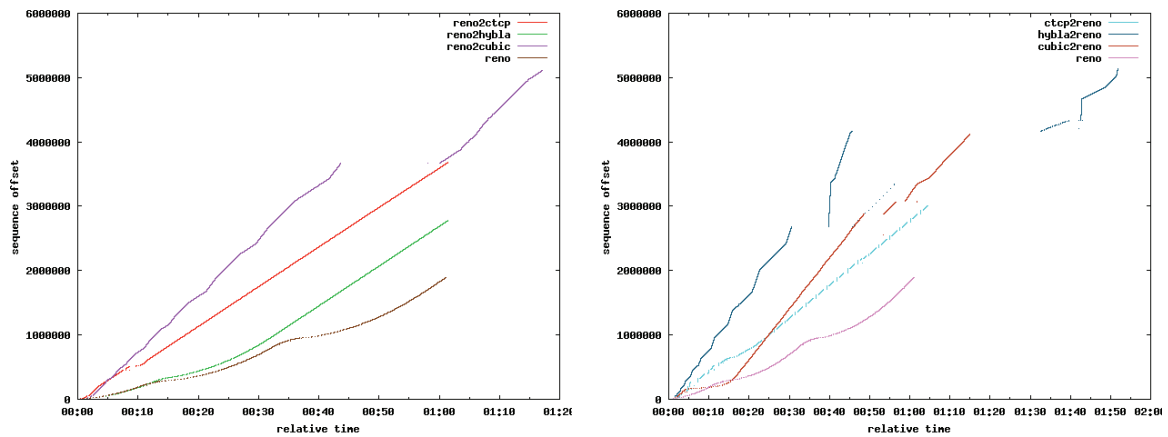


Figure 57 : Tests avec TCP New Reno tests (client/server hétérogènes)

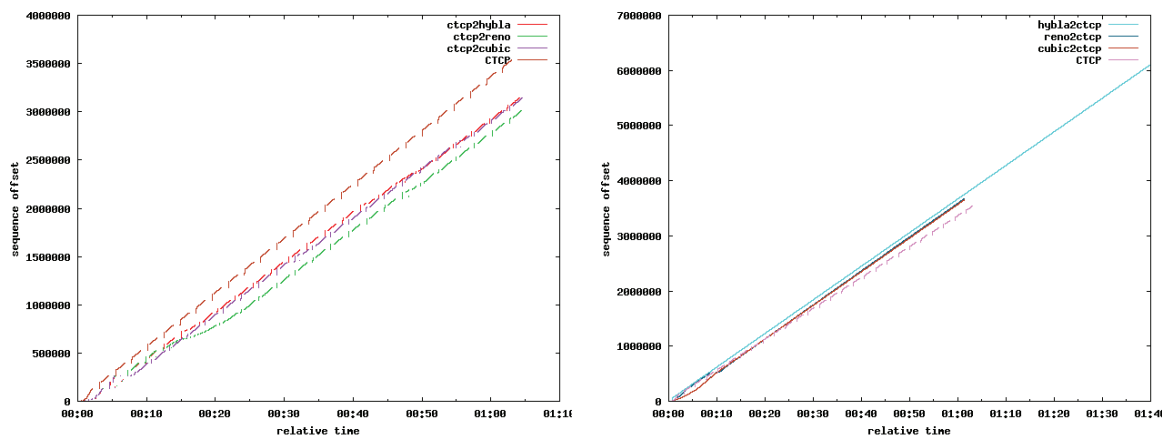


Figure 58 : Tests avec CTCP tests (client/server hétérogènes)

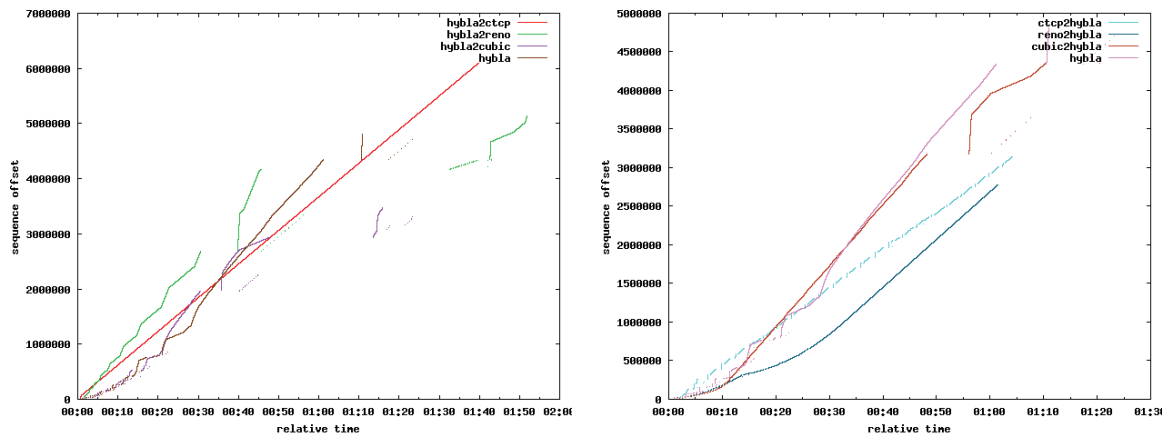


Figure 59 : Tests avec TCP Hybla tests (client/server hétérogènes)

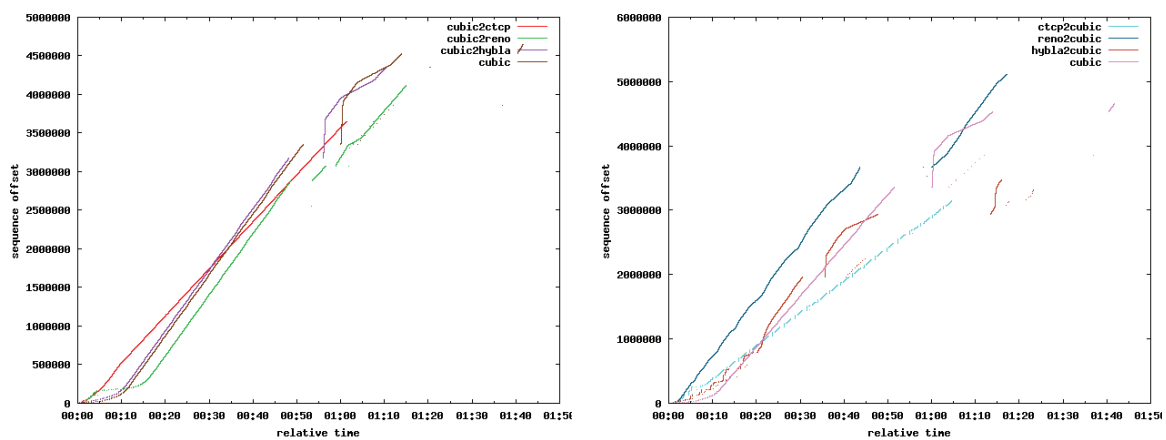


Figure 60 : Tests avec TCP Cubic tests (client/server hétérogènes)

b) Analyse

- Commentaires sur les expériences de New Reno (Figure 57) :
 - Quel que soit le côté où il est utilisé (émetteur ou récepteur), une version hétérogène de TCP (Cubic, Hybla ou CTCP) donne de meilleurs résultats qu'un New Reno bout en bout.
 - Hybla est intéressant seulement s'il est utilisé sur le côté émetteur. Utilisé sur le côté récepteur il donne plus ou moins le même résultat qu'un New Reno de bout en bout. En effet, Hybla a été conçu pour être utilisé par le côté émetteur, dans un environnement "contrôlé" pour être utilisé facilement dans un système satellite.
- Commentaires sur les expériences de CTCP (Figure 58) :
 - Le comportement de la connexion en utilisant CTCP sur l'un des côtés (émetteur ou récepteur) est similaire à une connexion CTCP.
- Commentaires sur les expériences d'Hybla (Figure 59) :
 - Quand Hybla est utilisé sur le côté émetteur, le comportement est similaire à une connexion Hybla (à l'exception avec CTCP en réception)
 - Une connexion Hybla/New Reno étonnamment produit un meilleur *Slow Start* qu'Hybla ou New Reno.
- Commentaires sur les expériences de Cubic (Figure 60) :

- Quand Cubic est utilisé sur le côté émetteur, le comportement est similaire à une connexion Cubic (à l'exception avec CTCP en réception).
- La combinaison entre Hybla en émission et Cubic en réception semble être très agressive et produit de longues coupures.

Pour résumer, le comportement des combinaisons hétérogènes client/serveur semble être principalement influencé par l'émetteur, à l'exception de CTCP qui produit toujours un comportement semblable CTCP s'il est utilisé sur un côté. New Reno profite de l'évolution du récepteur. Le Tableau 9 résume les résultats obtenus.

Receiver Sender	New Reno	Hybla	Cubic	CTCP
New Reno	x	Hybla +-	cubic +-	Ctcp +-
Hybla	Reno ++	x	cubic +-	Ctcp +-
Cubic	Cubic +-	Hybla --	x	Ctcp +-
CTCP	Ctcp +-	Ctcp +-	Ctcp +-	x
Legend : "--" worse , "+-" similar. "++" better				

Tableau 9 : Combinaison des différentes versions de TCP (client/serveur hétérogènes)

Ces résultats sont intéressants dans la mesure où les premiers tests montrent un intérêt à l'utilisation de CTCP pour la navigation Web sur satellite. De fait, la seule modification de la version des serveurs TCP permet la modification du comportement global des connexions TCP.

3.3.5 TCP versus PEP

a) La configuration

Le lien satellite est émulé à 250 ms d'OWD et 512 kbit/s de bande passante, Le taux d'erreur est laissé à 0 (pas de fading). Trois versions ont été utilisées : CTCP pour sa stabilité, New Reno en tant que référence et PEPsal. Pour rappel, PEPsal est un PEP intégré basé sur la technique de division d'une connexion (TCP-Splitting). La raison pour la division de la connexion TCP est de séparer la portion satellite du reste du réseau. Lors de la division de la connexion TCP, PEPsal adopte Hybla [74] [84] dans le segment satellite tandis que le reste du réseau adopte la version standard TCP, dans notre cas, nous avons utilisé New Reno sur le segment terrestre.

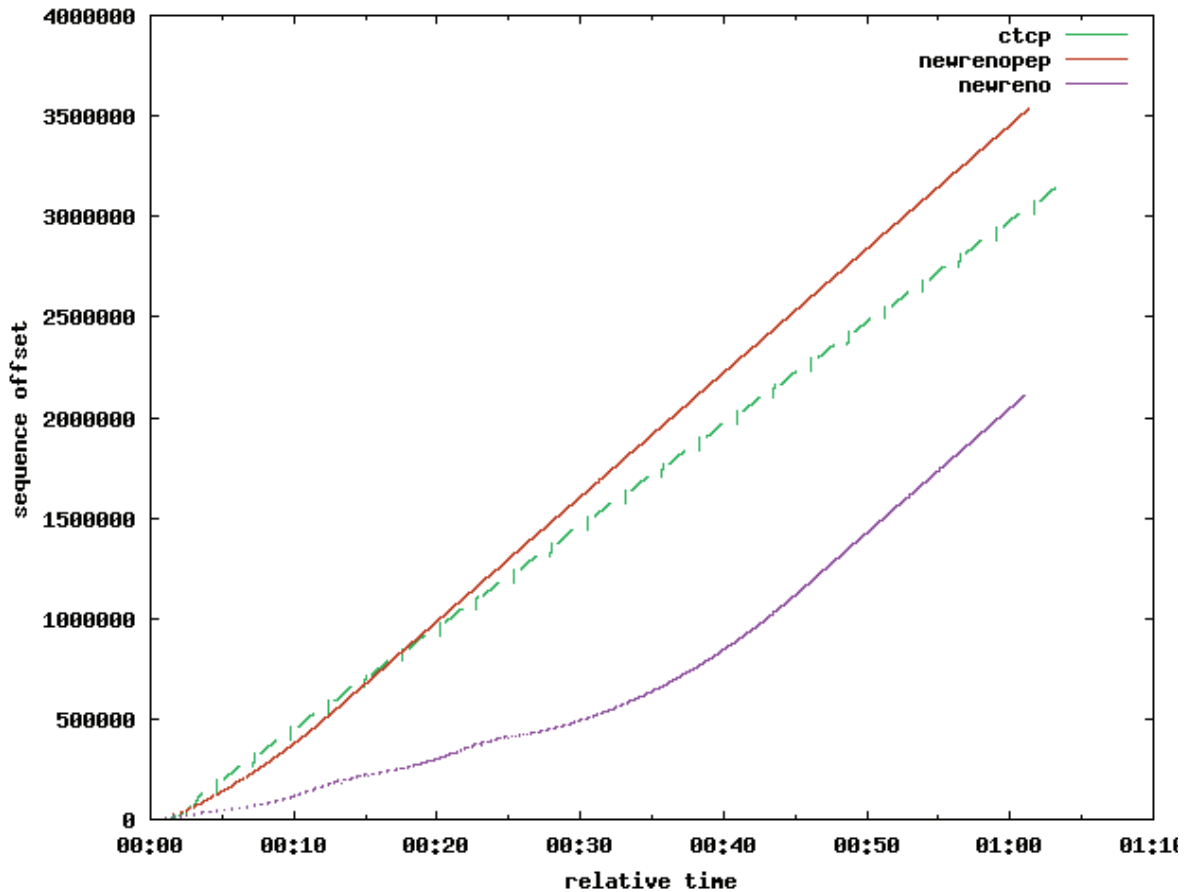


Figure 61 : Evolution des numéros de séquence sur un lien satellite, impact des PEPs

b) Analyse

Ce test nous permet d'avoir quelques éléments de réponse sur la nécessité (ou non) de l'utilisation des PEPs dans les réseaux satellite.

Compte tenu de nos résultats précédents avec les nouvelles versions TCP dans les systèmes satellite, nous essayons de montrer dans cette partie que les PEP (Solution PEP intégrée, PEPsal) ne sont plus obligatoires dans une telle configuration, en tout cas pour l'amélioration du comportement de TCP. (Les PEPs offrent des fonctionnalités supplémentaires comme la fonctionnalité d'accélération, pre-fetching HTTP...).

Dans la Figure 61, nous observons l'évolution des numéros de séquence de 3 flux de versions différentes de TCP, sous les mêmes conditions.

Il est clair que la solution PEPsal surpasse une connexion New Reno. Cependant, on note le bon comportement de CTCP, qui reste toutefois stable, un tout petit peu bas par rapport à PEPsal.

CTCP est très proche des performances de PEPsal, qui tend à confirmer que les solutions PEP ne sont plus obligatoires dans une telle configuration.

Durant les 20 premières secondes, CTCP égale les performances de PEPsal (1 Mb transférés @ 512 kbit/s). Puis PEPsal progressivement dépasse CTCP. A la seconde 60, CTCP est à 12 % au-dessous de PEPsal. New Reno bien sûr, est 42 % inférieur à PEPsal.

Les autres versions (Cubic et Hybla) ont été testées et comparées à PEPSal. Les résultats ne sont pas différents et peuvent être déduits des tests dans la partie 3.3.1. La substitution de New Reno sur la portion terrestre (entre le client et le PC PEPSal, dans notre configuration, Figure 44) par Cubic ou Hybla n'apporte pas de grandes améliorations sur le comportement général.

3.3.6 L'impact d'ECN sur les mécanismes de TCP

L'objectif principal du mécanisme ECN est de déclencher le mécanisme de contrôle de la congestion TCP avant congestion pour maintenir le niveau des buffers à une valeur raisonnable dans le réseau congestionné. Dans les systèmes satellite, le réseau congestionné sera généralement le routeur de réseau d'accès satellite, ce routeur génère généralement une restriction de bande passante sur l'accès satellite, et donc est une source potentielle de congestion.

a) La configuration

La Figure 45 et Figure 56 (respectivement, comparaison des RTTs pour un lperf et un transfert FTP) présentent le délai perçu par l'application durant une communication TCP sur un lien satellite. Ce délai est minimal avec les versions classiques de TCP (mais avec un débit limité), et avec CTCP (qui semble trouver le bon débit rapidement), mais cela prend ~14 secondes avec Cubic et Hybla. Ceci peut être expliqué par une petite surestimation de la bande passante disponible, l'agressivité du protocole est responsable de cela (une grande Cwnd), qui mène à une congestion (Figure 62). La Figure 45 compare le débit des différentes versions TCP. Si on considère que CTCP fournit la meilleure estimation de la bande passante disponible, Cubic et Hybla dépassent légèrement cette limite. Ceci est dû aussi aux buffers de l'émulateur SATEM qui sont grands, ce qui engendre un grand remplissage de données avant d'être envoyées sur le lien satellite ou perdues.

En effet, 10 secondes de délai pour une seule connexion TCP (un téléchargement par exemple) n'est pas grave, mais le grand niveau de congestion dans les routeurs impacte aussi les autres connexions TCP. Les connexions courtes ou interactives ne peuvent pas faire face à un tel délai.

ECN fournit un moyen de contrôler les niveaux des buffers, selon une configuration délicate des paramètres ECN sur le routeur d'accès satellite. Les recommandations pour la configuration ECN sont données ci-dessous [106]:

- **L'algorithme**

La taille moyenne de la file est utilisée pour déterminer la probabilité de marquage ECN. Celle-là est calculée en utilisant la moyenne EWMA (Exponential Weighted Moving Average), qui peut être plus ou moins sensible aux bursts.

Quand la taille moyenne de la file est inférieure à la valeur Min en octets, aucun paquet ne sera marqué. Quand elle dépasse la valeur Min, la probabilité augmente linéairement, jusqu'à ce que la taille moyenne de la file atteigne la valeur Max en octets. Puisque la probabilité n'est pas mise normalement à 100%, la taille de la file pourrait peut-être s'élever au-dessus du Max.

- **Les paramètres**

Min : Taille moyenne de la file au cours de laquelle le marquage devient possible.

Max : A cette taille de la file, la probabilité du marquage est à son maximum. Elle devrait être au moins le double de la valeur Min pour éviter des retransmissions synchrones.

Probability : c'est la probabilité maximum de marquage, c'est un réel entre 0.0 et 1.0. Les valeurs suggérées sont 0.01 ou 0.02 (respectivement 1 et 2%).

Limit : c'est la limite réelle de la taille de la file en octets. Les autres paquets seront supprimés. Elle doit être plus grande que la valeur de Max+Burst. Il est conseillé de la mettre plus élevée que Max.

Burst : Utilisé pour déterminer à partir de quel débit la taille moyenne de la file est influencée par la taille réelle de la file. Une grande valeur rend le calcul plus lent, permettant plus des bursts de trafic avant que le marqueur commence. Les expériences réelles s'appuient la formule suivante :

$$\text{Burst} = (\text{min} + \text{min} + \text{max}) / (3 * \text{avpkt}).$$

Avpkt : spécifié en octets. Elle est utilisée avec Burst pour déterminer la constante de temps pour le calcul de la taille moyenne de la file. 1000 est une bonne valeur (taille des paquets utilisés).

Bandwidth (optionnelle) : Ce débit est utilisé pour calculer la taille moyenne de la file après une période d'inactivité. il doit être réglé en fonction de la bande passante de votre interface.

ECN : Comme on l'a déjà dit avant, RED peut marquer ou supprimer. ECN autorise RED à notifier les hôtes finaux que leurs débits dépassent la capacité du canal. Les hôtes avec l'option Non-ECN peuvent être seulement notifiés par suppression de paquet. Si ce paramètre est spécifié, les paquets qui indiquent que leurs hôtes distinguent ECN seront seulement marqués ou supprimés, à moins que la taille de la file atteigne la valeur "Limit". Elle nécessite la commande TC avec RED.

b) Analyse

Selon nos tests et la définition du lien satellite 512 kbit/s, on propose les paramètres suivants pour les tests ECN :

ECN configuration with a 512 kbps satellite link

Parameter :	Unit:	calcul method :	set 1:	set 2:
bandwidth (bw) :	bits / second (b/s)		512000	512000
Basequeuing (BQ) :	seconde (s)		0,6	1,2
Min (min) :	octet (o)	bw * BQ	38400	76800
Max (max) :	octet (o)	max - 3*min	115200	230400
Burst (brst) :	paquet (pkt)	brst = (2*min+max)/(3*avpkt)	46	128
Limit (limit) :	octet (o)	limit = 8*max	921600	1843200
Avpkt (avpkt) :	octet (o)		1400	1000
max probability (pmax) :	pourcentage (%)		0,02 or 0,01	0,02

Tableau 10 : Configuration d'ECN dans notre plateforme

La Figure 62 présente les résultats obtenus avec la configuration Set 1 du Tableau 10. La chose remarquable est que les buffers représentés par le délai gardent des valeurs minimales (750 ms) avec ECN. Bien que nous ayons une légère perte en débit (set 2 donne un meilleur débit mais un plus grand délai) mais cela garde de bonnes valeurs pour CTCP et Hybla. A notre surprise, Cubic ne

fonctionne pas si bien avec ECN. New Reno est très affecté par l'ECN set 1 qui a été volontairement choisi pour être relativement agressif.

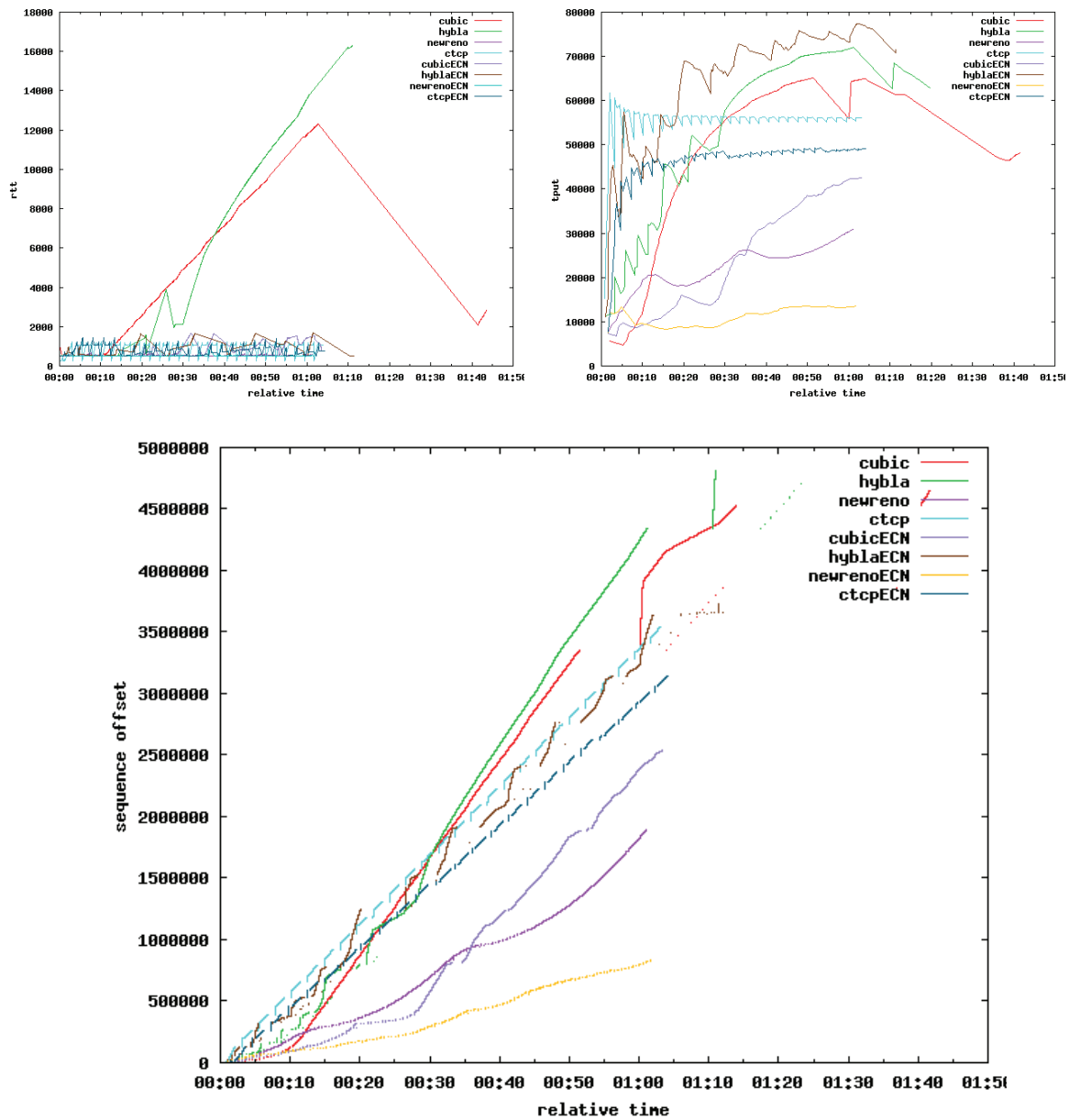


Figure 62 : Résultats avec la première configuration ECN (relative au Tableau 10)

ECN semble être une bonne solution complémentaire pour garder les buffers de routeur d'accès satellite à de bas niveau et conserve l'interactivité du système sans pénaliser les performances du système.

De plus, ECN présente une facilité de configuration puisqu'il suffit d'activer l'ECN (mettre la valeur de `ecn capability=enabled` sous windows 7 et `net.ipv4.tcp_ecn=1` sous Linux).

Au niveau du routeur, par défaut "drop tail" est utilisé en supprimant les paquets quand la file du routeur déborde, RED permet d'anticiper le débordement en supprimant des paquets avant. Avec ECN les paquets ne sont pas supprimés mais juste marqués.

3.4 Conclusion

Plusieurs versions de TCP ont été testées pour mesurer leur impact sur le réseau satellite. Les résultats préliminaires montrent la supériorité de CTCP par rapport aux autres solutions et présente un comportement presque identique une solution avec le PEP. Les nouvelles versions TCP ont de bonnes performances bien meilleures que les anciennes.

En outre, la littérature de TCP sur satellite doit être remise à jour avec les nouvelles versions que nous venons de voir dans ce chapitre.

D'autre part, les versions TCP par défaut de Linux (Cubic) et Microsoft Windows (CTCP) se répandent très rapidement hors système satellite ce qui les place comme futurs standards pour TCP.

4 Proposition sur l'amélioration du contrôle de congestion sur le DVB-S2 : le protocole DCCP

Le protocole TCP a fait ses preuves sur les réseaux terrestres, vu sa large utilisation dans les services offerts. TCP est même utilisé dans le cache du streaming live audio vidéo (youtube, dailymotion...etc).

Cependant, dans les réseaux satellite, le délai de propagation important ainsi que certains types de perturbations pénalisent TCP vu ses mécanismes de contrôle de congestion "purs" et la fiabilité qu'il exige. Cela dégrade les performances perçues. C'est pour cela, que certains protocoles sont mieux placés pour remplacer TCP sur le réseau satellite pour ce type d'application.

DCCP [60] est un protocole alternatif qui a la particularité d'être à la fois un protocole à datagramme, implémentant des mécanismes de contrôle de congestion. Il peut utiliser plusieurs mécanismes de contrôle de congestion. Un mécanisme de contrôle de congestion est formé essentiellement par un algorithme de régulation de débit mis en œuvre par l'émetteur, et un mécanisme de retour d'information (ACKs...) par le récepteur. Dans DCCP, nous identifions les méthodes de contrôle de congestion par un code d'un octet, dit CCID (*Congestion control ID*). Ce codage permettrait à DCCP de supporter environ 256 mécanismes de contrôle de congestion différents. Deux méthodes ont d'ores et déjà été normalisées ; CCID 2 [107]: un contrôle de débit « à la TCP » (*TCP-like*) ; CCID 3 [108] : un mécanisme dit *TCP-Friendly Rate Control* (TFRC). Un troisième mécanisme est en cours de développement, le *TCP-Friendly Rate Control for small packets* (TFRC-SP) [109]. Le CCID 0 est réservé et le CCID 1 correspond au contrôle de congestion basé sur l'émetteur non spécifié.

Nous avons choisi le CCID 3, qui implémente le mécanisme de TCP Friendly Rate Control (TFRC), pour effectuer nos expérimentations et notre contribution.

Dans cette partie, nous allons décrire l'impact du réseau satellite sur DCCP et proposer une amélioration sur la voie DVB-S2 pour une utilisation optimale. Nous avons choisi le CCID 3 de DCCP qui implémente TFRC pour sa "non agressivité" à traiter les phénomènes de congestion et son adaptation flexible à la bande passante disponible.

4.1 The TCP Friendly Rate Control

TCP Friendly Rate Control (TFRC) qui est utilisé par le CCID 3 fournit une compatibilité avec TCP tout en gardant l'équité entre les flux et la stabilité des transmissions. TFRC augmente son débit tout en gardant la variation moins agressive que TCP quand le taux de pertes diminue. Cependant, il évite de diviser par 2, quand les pertes surviennent, à l'exception de pertes sévères. Le débit d'émission est calculé en se basant sur les conditions du réseau. L'émetteur adapte son débit en se basant sur les messages de retour contenant le taux de pertes, le débit reçu et le délai. Comme TCP, TFRC utilise deux phases de transmission, *Slow Start* et *Congestion Avoidance* [110].

Le premier débit d'émission est calculé selon la formule ci-dessous (aucun feedback n'est encore reçu) :

$$X = \frac{\min(4.S, \max((2.S, 4380)))}{RTT} \quad \text{Équation 5}$$

Dans la phase *Slow Start*, TFRC évalue son débit selon :

$$X = \min(2.X, 2X_{recv}) \quad \text{Équation 6}$$

Quand le récepteur envoie les premiers événements de pertes, TFRC se met en *Congestion Avoidance* et utilise la formule suivante [108] :

$$X = \frac{S}{RTT \times \sqrt{\frac{2p}{3}} + \left(3 \times RTO \times \sqrt{\frac{3p}{8}} \times (p + 32p^3) \right)} \quad \text{Équation 7}$$
$$X = S.f(p, RTT)$$

Avec : X le débit d'émission en octets/s; S est la taille des paquets en octets; RTT est le délai d'aller-retour en s; p est le taux de pertes (rapports entre paquets transmis et reçus), compris entre 0 et 1; et le RTO est le timer de retransmission exprimé en s.

Le CCID 3 basé sur le TFRC, a été conçu pour être utilisé dans des réseaux génériques mais ses performances sont sensibles aux caractéristiques des réseaux, comme le RTT , les pertes et la gigue [111].

Le graphe suivant présente un flux TFRC sur un lien émulé avec une bande passante de 256 kbit/s et un RTT de 500 ms.

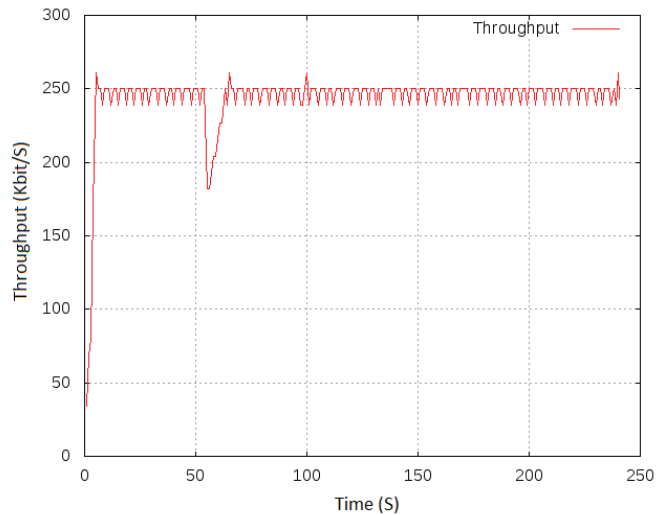


Figure 63 : Débit TFRC en condition nominale

Le Figure 63 a été obtenue avec un flux de bit rate constant. TFRC débute avec un *Slow Start* et atteint un débit stationnaire durant la transmission. TFRC, comme on l'a dit avant, a été conçu pour opérer dans les réseaux terrestres et pour des applications avec un bit rate constant et garantissant un débit d'émission constant [112]. Il n'est pas conçu pour les applications caractérisées par un bit rate variable à l'émission. Cette variation est décrite par des périodes d'émission à grand débit séparées par des périodes de transmission limitées (limited) ou non envoi (no sending). Par exemple, les applications vidéos avec un bit rate variable ou les applications modernes de VoIP supportant la suppression de silence. Les flux en burst peuvent causer des perturbations, de la même manière que des mauvaises conditions météorologiques ou le mauvais positionnement de l'antenne.

4.2 L'impact des réseaux satellite sur DCCP avec les flux à débit variable

Le graphe suivant présente le comportement de TFRC avec une application de type bit rate variable ou lors d'une perturbation [112] et [113].

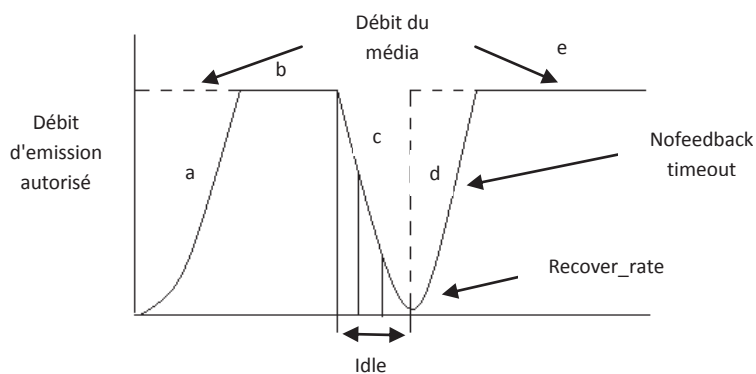


Figure 64 : Occupation du media Vs. débit d'envoi autorisé

Le graphe ci-dessus présente l'occupation du media avec le débit d'envoi autorisé. Nous observons 5 phases :

- a. Le *Slow Start* initial : il commence avec un maximum de 1 paquet par RTT, ce qui signifie que le débit de départ est inférieur au débit de l'application multimédia. Un tel comportement est accepté pour de petits RTTs, mais cela peut provoquer une dégradation des performances pour les réseaux avec une grande latence.
- b. Une période de transmission au taux des médias : il peut y avoir une situation où TFRC peut envoyer avec un taux atteignant le taux des médias, ou avec un taux plus élevé transitoire ; cette situation se produit quand un nombre de paquets se trouvent dans le buffer d'émission. S'il n'y a pas de paquets en attente dans le buffer, TFRC envoie à un taux déterminé par le taux du média.
- c. Au cours d'une période d'inactivité (Idle period) : lorsqu'un timeout de non-feedback se produit, la transmission est arrêtée pendant une période de 4 RTTs ou plus ; le TFRC réduit le débit d'envoi autorisé de moitié pour chaque expiration du timer de no-feedback (c.-à-d. 4 RTTs). Ce débit peut atteindre 2 paquets par RTT. C'est ce qu'on appelle *recover_rate* comme indiqué dans [114].
- d. Limitation de débit par le débit de réception : TFRC commence par un *Slow Start* pour se déplacer d'un *recovery_rate* vers le débit du média. Le débit d'envoi peut augmenter au plus deux fois le débit actuel du récepteur (débit reçu au cours du dernier RTT). Cela peut entraîner un retard lorsqu'il est utilisé sur des réseaux à forte latence. Un état de *data-limited* ou *application-limited* d'une application multimédia survient si elle est en veille (idle) ou n'envoie pas à un débit autorisé. Cela implique un faible débit puisque TFRC le calcule en incluant les périodes où les données n'ont pas été envoyées. Durant plusieurs RTTs, le faible débit du récepteur limite le débit d'envoi à une valeur inférieure au débit requis pour maintenir le débit multimédia.
- e. Retour au débit du média : lorsque le débit d'envoi atteint le taux de média et que les buffers d'émission sont vides, alors le débit de l'émetteur est à nouveau contrôlé par le débit du média.

Dans le cas des flux en burst, les débits en émission oscillent entre des périodes d'inactivité et d'utilisation maximale de la bande passante, et implique des oscillations entre des périodes de débit d'envoi suffisant et d'autres insuffisants. Dans notre étude, nous nous concentrons sur les perturbations causées par les conditions météorologiques, ou la position géographique. Nous supposons également que les flux utilisés sont des flux vidéo continus (CBR) en streaming sur le lien aller DVB-S2.

4.3 L'apport de l'ACM et problématique

Lors d'une perturbation météorologique, le ST mesure le taux d'erreur et le transmet à la GW via le lien retour. Par la suite, la GW change le modcod pour maintenir la fiabilité de réception en envoyant un message au STs, incluant la modulation et le codage à utiliser pour fournir la possibilité au ST à démoduler et décoder les flux. En conséquence, la bande passante disponible sera réduite. L'ACM [115] favorise la réduction de la bande passante aux coupures du lien. CCM, qui est utilisé sur la première génération DVB-S et qui garde la même modulation et un FEC constant, peut induire des coupures du lien lors de perturbations sévères.

Cette nouveauté peut produire une variation drastique et brutale de la bande passante et peut poser de nouveaux problèmes aux couches transport.

Au niveau transport, l'émetteur adapte son débit d'envoi aux perturbations en calculant le nouveau débit à chaque retour feedback, en utilisant la formule $X=f(s,p,RTT)$. Cependant, le délai de propagation et de bufferisation retardent les feedbacks au sein des réseaux à forte latence. Ce retard oblige l'émetteur à entrer dans une phase d'inactivité (Idle period). Dans ce cas, l'émetteur limite son débit d'envoi et le divise par 2 chaque 4 RTTs (no feedbacks timeout=4*RTT). Les paquets pendant cette période seront perdus et non retransmis.

En conséquent, l'émetteur retourne en *Slow Start* et le débit sera multiplié par 2 chaque RTT. Cela peut être pénalisant pour les applications multimédia.

Notre proposition concerne la récupération après une période d'inactivité ou limitée sur le lien DVB-S2.

En présence de l'ACM, la bande passante disponible est réduite par le ST suivant les nouvelles valeurs de modulation et codage FEC. Cependant, dû au grand délai du satellite, cette période de transition (réduction de bande passante) peut être longue. Pendant cette période, les ressources du satellite sont disponibles mais pas utilisées, jusqu'à l'adaptation du début avec l'ACM.

4.4 Solutions proposées

Dans cette partie, nous présentons brièvement les solutions d'amélioration de DCCP existantes puis nous présentons notre proposition relative à l'amélioration des performances de DCCP

4.4.1 Travaux relatifs

Certaines solutions ([111], [113] et [114]) ont été proposées pour améliorer le comportement de TFRC spécialement pour avoir un meilleur *Slow Start* sur satellite.

- Fast Restart (FR) : il a été proposé dans [116] pour accélérer le *Slow Start* après une période de limit sending. Fast Restart permet à l'émetteur de quadrupler le débit d'émission chaque RTT sans congestion, contrairement à la version standard du TFRC qui le double seulement (le débit d'émission). Il permet aussi de mettre le *recover_rate* à 8 paquets par RTT et ne diminue pas le débit d'envoi sous le seuil du *recover_rate* lors d'une congestion.
- Revised TFRC : IETF a révisé les spécifications TFRC pour mettre à jour la version précédente qui devient obsolète (RFC3448 vers RFC5348). La mise à jour permet d'augmenter le débit initial à 4 paquets par RTT, et par conséquent améliorer le comportement du *Slow Start* après un période Idle ou Limit sending Data. Les spécifications révisées ont été faites spécialement pour les applications multimédia contenant des périodes d'inactivité. Les détails de cette étude sont détaillés dans [113].

4.4.2 Proposition

Aucune solution mentionnée ne résout le problème de l'inutilisation des ressources pendant le changement de codage et modulation.

Nous proposons alors que l'ACM, en cas de perturbations météorologiques, soit en mesure d'avertir la couche transport DCCP du client, lorsqu'il est sur le point de changer de codage (donc de débit), afin que la couche transport DCCP puisse réagir de manière proactive, afin d'éviter de rentrer dans une phase de *Slow Start*.

Une solution de communication en Cross Layer [12] [117] peut s'avérer intéressante, en envoyant directement un message cross layer entre la couche ACM du ST à la couche transport DCCP du client (End user).

Lors du changement de codage et de modulation par l'ACM, la couche transport DCCP du client reçoit une notification du ST qui informe d'une réduction de la bande passante disponible X_{acm} . La couche transport DCCP reçoit cette notification avec le débit proposé par l'ACM et s'adapte directement sans attendre les feedbacks du récepteur, $X=X_{acm}$. Cette solution ne cherche pas à fournir une valeur de bande passante au client DCCP, mais simplement l'avertir par exemple que la bande passante est divisée par deux avec le nouveau codage. Elle ne donne qu'une valeur relative ce qui est très simple à obtenir par le ST et à utiliser par le client.

Quand l'émetteur reçoit les acquittements du récepteur (taux de perte et le calcul de RTT), le mécanisme de contrôle de congestion TFRC recalcule son débit de transmission via son équation et garde le meilleur des deux débits $\rightarrow X=\max (X_{tfr}, X_{acm})$. La Figure 65 schématise le mode d'opération de cette solution.

Cette solution limite ainsi les pertes et la réduction du délai des paquets dans les buffers. La gigue par conséquent sera aussi diminuée. Elle évite aussi à l'émetteur de rentrer en *Slow Start* et lui permet de s'aligner bien plus rapidement sur le nouveau débit réduit.

Cela évite à l'émetteur de perdre des ressources disponibles et atteindre le nouveau débit plus rapidement.

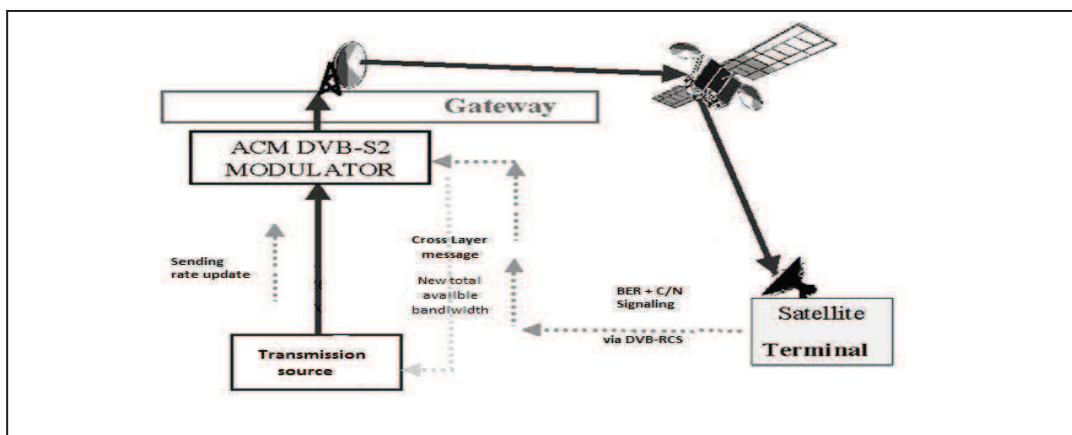


Figure 65 : Description de la solution

4.4.3 Tests et validation

Pour valider notre proposition, nous avons réalisé un ensemble de tests pour comprendre le comportement de TFRC dans notre plateforme d'émulation d'un réseau satellite SATEM.

Pour les tests, nous avons choisi les paramètres suivants :

- 512 kbit/s pour la bande passante du lien.
- 300 ms pour l'OWD.

Les perturbations sont générées par un générateur de bruit logiciel (SATEM). La génération des flux est assurée par IPERF supportant DCCP.

La Figure 66 présente le comportement de TFRC avec CCM. Le bruit à la seconde 60 réduit rapidement le débit. L'émetteur continue à envoyer pendant quelques secondes (en attendant l'expiration du timeout de no-feedback), puis l'émetteur commence à diviser son débit par deux jusqu'à être nul. Le débit redevient normal à la seconde 160 après l'arrêt du bruit.

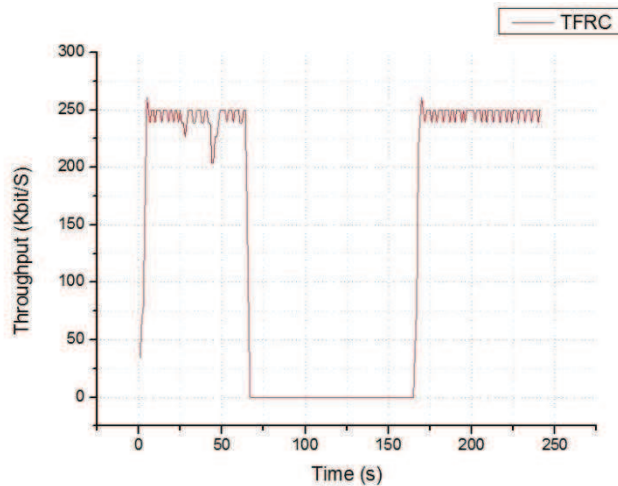


Figure 66 : Débit d'émission TFRC sans ACM

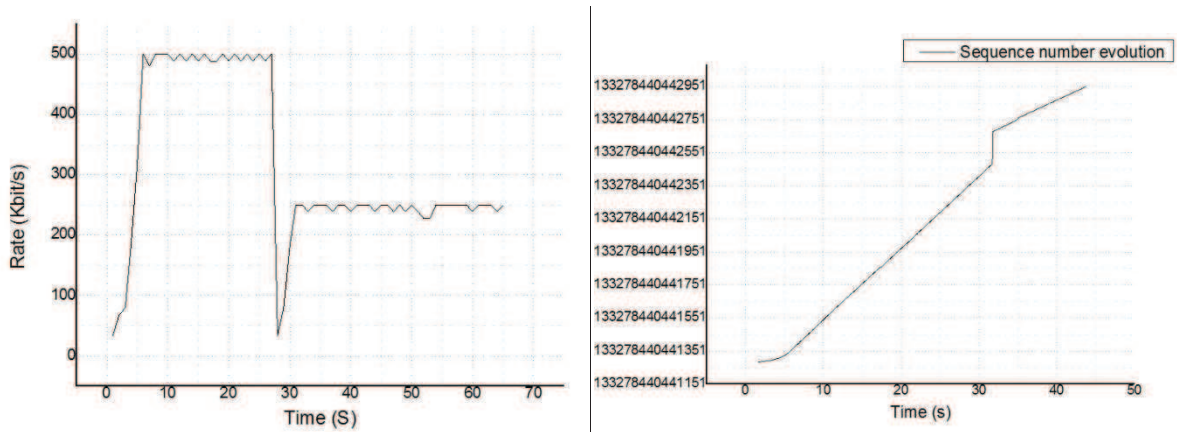


Figure 67 : Débit d'émission TFRC and évolution des numéros de séquence avec ACM

Dans la Figure 67, l'ACM adapte son débit et limite sa bande passante à 256 kbit/s après les perturbations. Une coupure de transmission est remarquée à ce moment afin d'avoir le nouveau codage et modulation. Ce délai est dû au délai de propagation satellite. Cependant, les ressources restent inutilisables puisque l'émetteur est dans la période de no data send. Il redémarre l'émission

au moment d'avoir les premiers feedbacks, de plus, l'évolution des numéros de séquence montre clairement la perte des paquets durant la période d'adaptation.

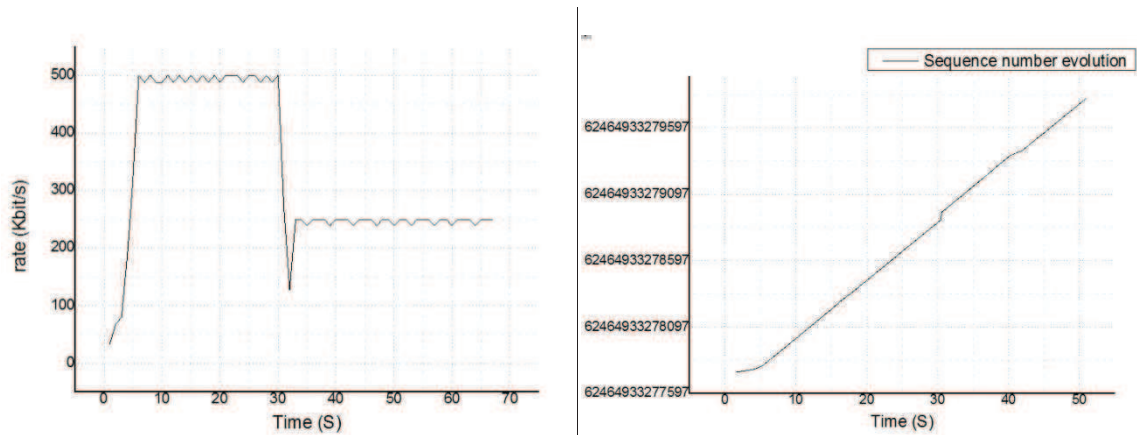


Figure 68 : Débit d'émission TFRC and évolution des numéros de séquence avec l'anticipation ACM

La Figure 68 présente le gain des ressources avec l'anticipation ACM. Quand la perturbation survient à la seconde 30, l'ACM GW envoie un message cross layer à la couche transport de l'émetteur pour l'informer de l'éventuel réduction de débit. L'émetteur n'attend pas le premier feedback mais calcule directement un nouveau débit proportionnel à la réduction imposée par l'ACM. La période de no data send est réduite considérablement. De plus, l'évolution des numéros de séquence prouve la réduction des pertes de paquets.

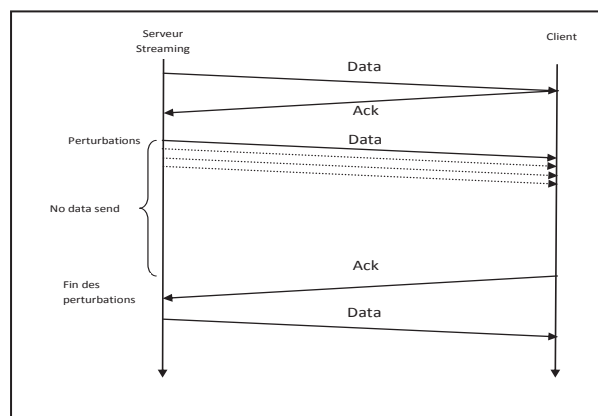


Figure 69 : Diagramme de séquence pour l'envoi en mode CCM

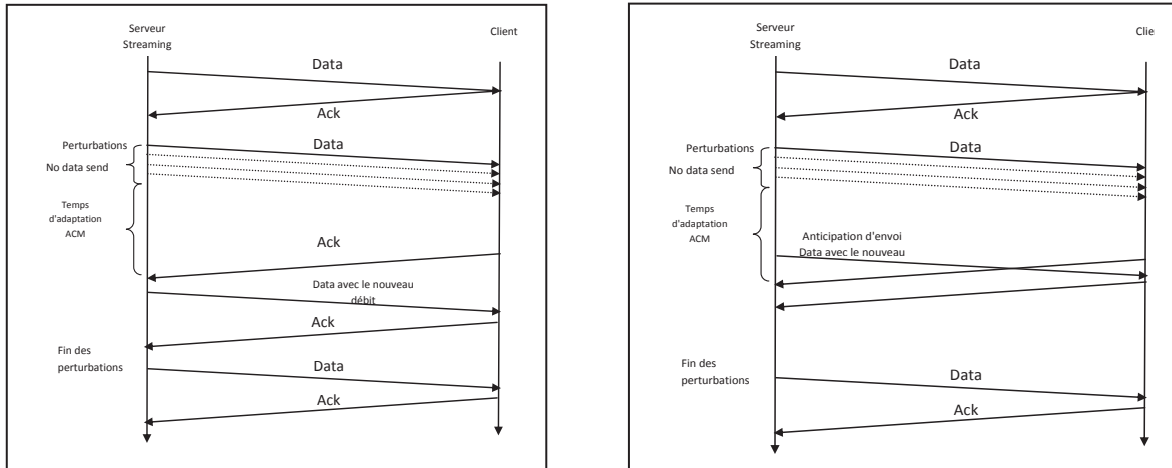


Figure 70 : Diagramme de séquence pour l'envoi en mode ACM sans (à gauche) et avec (à droite) anticipation

La Figure 69 présente le diagramme de séquence du premier cas de figure, c'est-à-dire l'envoi de données en mode CCM. Lors d'une perturbation, une coupure est à prévoir et l'émetteur entre en mode "no data send" jusqu'à la fin de la perturbation.

La Figure 70 présente le diagramme de séquence de l'envoi en mode ACM avec et sans anticipation. Le diagramme à gauche montre qu'une période est nécessaire pour le changement de codage et modulation. Pendant cette période, aucun envoi n'est effectué. Cependant, le diagramme de droite présente le cas de l'envoi de données avec ACM en utilisant le mécanisme d'anticipation. Nous remarquons que l'envoi reprend même avant la fin du délai nécessaire pour l'adaptation du codage et modulation.

Dans le cas d'une application vidéo streaming, l'émetteur a le choix de codec, et la sélection d'un nouveau codec se fera plus rapidement directement avec le message reçu de la ACM GW.

4.5 Conclusion

Le TFRC implémenté dans le CCID 3 est l'un des contrôles de congestion proposé par DCCP. Le contrôle de congestion "doux" lui permet d'adapter le débit d'émission sans pour autant être agressif. En utilisant le TFRC sur la voie aller DVB-S2, le gain de capacité en utilisant l'ACM est estimé à environ 30 % par rapport à DVB-S tout en offrant les meilleures performances de transmission, une souplesse totale et une complexité raisonnable du récepteur. Cependant, nous avons remarqué une perte de ressources due aux caractéristiques du média.

La solution proposée est prometteuse puisqu'elle optimise l'utilisation de la bande passante. L'anticipation du changement de débit en envoyant des signalisations cross layer à l'émetteur permet un gain en bande passante inutilisable (vu la latence du changement de codage et de modulation). La période d'attente de l'adaptation du débit a été réduite. Dans le cas d'une application streaming vidéo, le codec de compression peut être choisi rapidement avant l'adaptation du débit pour éviter les perturbations causées par ce changement. Malgré sa complexité (cross layer horizontal), cette solution peut tout à fait se justifier sur des serveurs de vidéo localisés derrière un accès satellite ou des serveurs d'images à haute résolution.

5 Conclusion

Les évolutions récentes de TCP, au cours des cinq dernières années, ont introduit un changement radical du contrôle de congestion. Le problème classique de l'adaptation de TCP au système satellite en est totalement modifié. La version Cubic de TCP utilise un contrôle de congestion qui rend indépendant la croissance de la fenêtre de congestion du RTT. La version Compound de TCP utilise un contrôle de congestion qui prend en compte la variation du RTT dans l'équation de la croissance de la fenêtre de congestion, ce qui de fait ne pénalise pas les réseaux long délai.

Vu les caractéristiques présentées par ces nouvelles versions de TCP, nous avons abouti à une proposition originale consistant à utiliser sur le segment satellite ces versions sans autre artifice architectural tel que les PEPs, pourtant généralisés dans les systèmes satellites aujourd'hui. Cette approche facilite de plus l'intégration d'un système satellite dans les architectures terrestres courantes. L'utilisation de PEP n'est pas recommandée à cause de la division d'une connexion TCP, qui viole les sémantiques de bout en bout basiques de TCP. De plus, rompre une connexion TCP empêche l'utilisation de certains mécanismes comme IPSec.

Nous avons testé plusieurs versions de TCP. Des tests avec des petits paquets ont été réalisés pour analyser la réactivité des versions pour un tel type de données (généralement des pages Web), CTCP est très réactif pour des petits paquets vu son démarrage très rapide.

Les résultats montrent d'une manière générale que les nouvelles versions de TCP se comportent bien mieux que les anciennes, New Reno dans notre cas. Compound TCP a donné de très bons résultats vu la double fenêtre de congestion, une basée sur le délai et une basée sur les pertes.

Les débits maximum atteignables sont de l'ordre du Mégabit (1.2 Mbit/s CTCP, 1.3 Mbit/s New Reno, 1.5 Mbit/s HYBLA, 1.5 Mbit/s CUBIC), soit un débit unitaire suffisant pour les applications classiques d'Internet. Le temps de montée est aussi tout à fait correct pour CTCP et un peu inférieur pour Hybla et Cubic. L'utilisation du PEP intégré PEPsal n'améliore pas ces résultats.

Le défaut majeur que nous voyons aux versions Cubic et Hybla est la bufferisation induite dans le routeur congestionné, vu leur agressivité.

L'ECN permet de résoudre ce problème de bufferisation et diminue le délai de bout en bout. Bien que le débit subisse une diminution, les délais de bout en bout sont réduits.

Cubic et Ctcp sont très répandus, puisque le premier est la version par défaut dans les machines Linux et le deuxième par défaut dans les machines Microsoft. Il n'y aura donc pas de problème de déploiement d'une solution particulière au satellite, contrairement à Hybla qui n'est pas répandu.

Nous avons aussi réalisé des tests en introduisant une hétérogénéité entre le client et le serveur, c'est-à-dire, par exemple, lorsqu'un client utilise Ctcp et qu'un serveur utilise une autre version, le comportement est identique à un système client/serveur avec Ctcp des deux côtés. Dans le cas inverse, cette affirmation reste vraie. Nous pouvons dire dans ce cas que Ctcp prime. L'accès à des services anciens ne pose donc pas de problème de compatibilité ou de mise à jour de protocole.

Dans le cas où Cubic est utilisé, la version de l'émetteur prime. De nombreuses versions de serveur web sont en Linux et donc utilisent par défaut Cubic.

D'autre part, pour les données n'ayant pas besoin de fiabilité, l'utilisation de DCCP sur la voie aller DVB-S2/ACM permet une meilleure gestion de certains types d'applications vidéo, mais le délai infligé par le média satellite dégrade l'adaptation du débit aux nouvelles caractéristiques du canal. Une proposition utilisant des signalisations cross layer a été faite dans ce chapitre pour minimiser le délai d'adaptation de débit et anticiper le changement de codec en émission pour des applications de type streaming live vidéo.

Finalement, l'utilisation des versions TCP pour les flux fiables (CTCP ou Cubic) ainsi que de DCCP pour les flux non fiables types vidéo streaming, comme nous l'avons proposé, nous permettent d'atteindre des performances satisfaisantes sur les réseaux satellite.

Chapitre 4 : Impact de la couche transport pour les réseaux hybrides satellite-terrestre

Actuellement, les réseaux satellite sont utilisés comme un support d'accès aux autres réseaux terrestres (WiFi, WiMax, 3G+ et filaires) et qu'on appelle communément les réseaux hybrides satellites/terrestres. Dans ce contexte, certaines exigences liées aux nouveaux usages sont apparues et spécifiquement l'accès généralisé, transparent aux services ainsi qu'à la mobilité partout et à n'importe quel instant.

Cependant, les réseaux hybrides satellites/terrestres posent une problématique que nous ne pouvons pas trouver ni dans les réseaux satellite ni dans les réseaux terrestres séparément. La problématique principale des réseaux hybrides réside dans l'adaptation de la couche transport avec les caractéristiques du nouveau réseau. En effet, lors d'un changement de réseau, plusieurs paramètres changent, et spécialement le délai de propagation, la bande passante, le taux de pertes et la gigue. Cette problématique concerne les performances applicatives sur la globalité du réseau hybride. Certaines fonctionnalités ont été conçues spécialement pour les réseaux terrestres, d'autres pour les satellites. Cela peut engendrer des dysfonctionnements si elles sont utilisées dans l'ensemble du réseau hybride. En particulier, des solutions terrestres ne sont pas adaptées aux réseaux satellite vu les caractéristiques de ces derniers (le long délai, la faible bande passante...etc.). (Voir conclusion 4.1.3).

Dans cette optique, plusieurs solutions ont été proposées pour pallier aux implications de la mobilité et de l'hybridation. Nous pouvons citer Freeze TCP conçu pour la mobilité qui "gèle" la communication pendant un handover, ou bien les PEPs (pour l'hybridation) qui permettent d'isoler la portion satellite des autres segments, afin de ne pas les pénaliser avec les dégradations subies.

Néanmoins, ces solutions d'optimisations ne sont pas adéquates dans le contexte actuel pour plusieurs raisons. D'un côté, Des problèmes de compatibilité ou d'implémentations peuvent être posés pour certaines solutions puisque nous ne pouvons pas gérer tous les éléments du réseau. Certaines solutions en client/serveur obligent l'implémentation des solutions sur les deux hôtes distants. D'autres nécessitent la manipulation des routeurs d'accès, ce qui n'est pas toujours évident. D'un autre côté, de solutions telles que les PEPs présentent quelques inconvénients. Bien que les PEPs améliorent les performances applicatives, mais cela va à l'encontre de la sémantique de bout en bout d'une connexion TCP. De plus, la division d'une connexion ne peut pas avoir lieu dans le cas des communications cryptées.

D'autres solutions peuvent être introduites dans le futur vu le développement rapide de L'Internet, de l'interconnexion de tous les réseaux et le besoin ascendant d'être connecté à tout instant et dans n'importe quel endroit.

L'objectif du chapitre est d'évaluer la pertinence des couches de transport face à la mobilité, dans un contexte hybride (satellite/terrestre). Le défi technique est tout d'abord de maintenir les applications connectées durant le changement de réseau tout en conservant leurs performances.

1 Gestion de la mobilité avec TCP dans un réseau hybride.

Notre étude vise à adapter le protocole TCP pour l'améliorer dans le contexte de la mobilité, en particulier dans les réseaux hybrides. La question principale est de faire face aux changements dans les médias d'accès. Les mécanismes présentés dans cette partie ont pour objet de traiter ce point particulier, hors du scénario hybride, mais plutôt l'Internet mobile.

- **Freeze TCP** : Il n'existe aucune fonctionnalité de TCP permettant d'indiquer à un hôte mobile qu'il va perdre la connexion et ne va plus être accessible pour un certain laps de temps. Freeze TCP [118] est un mécanisme qui demande explicitement à l'expéditeur de faire une pause en envoyant un acquittement avec une valeur de fenêtre de réception égale à 0, également appelée Zero Window Advertisement (ZWA). Lorsque le récepteur est prêt, il envoie une copie du dernier acquittement en mettant le champ Non-Zero Window afin de reprendre la communication.

Néanmoins l'expéditeur continue d'envoyer des Zero Windows Probes (ZWP) pour vérifier que la connexion TCP n'est pas perdue. Mais en utilisant des ZWPs, la réduction de la fenêtre de congestion de l'expéditeur est limitée. La connexion est réinitialisée uniquement si les ZWPs ne répondent plus après plusieurs retransmissions.

- **PETS** : Freeze TCP peut mettre en pause la communication, mais a encore besoin de répondre aux ZWPs pour indiquer à l'hôte de ne pas fermer la connexion. En longues périodes d'inactivité comme pendant les handovers, cela peut s'avérer problématique. Une solution simple, PETS [119], est d'utiliser un élément dans le réseau terrestre pour répondre aux ZWPs. Comme il doit intercepter tous les segments adressés à l'hôte mobile, il peut être placé dans un proxy ou dans l'expéditeur lui-même, au-dessous de la couche TCP.

Ce protocole est très simple parce qu'il a juste besoin de lire les segments. Quand le Zero Window Advertisement lui parvient, il répond aux ZWPs puis se met en veille quand l'hôte mobile ré-ouvre sa fenêtre de congestion. Il est important d'utiliser un autre timer pour mettre fin à la communication si le mobile est perdu. La durée de ce timer devrait être assez longue afin que le timeout de la couche application puisse expirer avant.

- **Fast Adaptation** : Cette technique est proposée par [120] et présentée ici à travers différents points qui peuvent être utilisés séparément. L'objectif est d'optimiser un handover en atténuant certains des problèmes inhérents.
 - Lors d'un handover, le mobile envoie un acquittement sur les deux réseaux, pour que le RTO de l'émetteur soit remis à zéro.
 - Un ré-ordonnancement de paquet est fort probable dans le cas d'un handover puisque les deux bandes passantes sont différentes. Il est préférable de suspendre les acquittements depuis les récepteurs pour éviter des retransmissions inutiles, cela rappelle les duplicate ACKs en retard.

- Un handover vertical conduit souvent à un changement radical du produit bande passante et délai (BDP). Il sera plus intéressant de calculer la nouvelle valeur et adapter la fenêtre de congestion. Ici, l'espacement entre les acquittements implique un goulot d'étranglement pour tout le chemin parcouru. Alors, l'envoi de paquets peut être effectué avec des espacements spécifiques. Quand les acquittements sont reçus, la taille de fenêtre de congestion à utiliser sera donc connue.

Cette technique est intéressante mais suppose que les acquittements ne sont pas altérés par le récepteur (retardés, cumulés,...etc.) et que la communication est correcte (sans ré-ordonnancement, délai, pertes,...etc.). Un contexte spécifique est à définir.

- **VA-TCP** : Vertical handoff Aware (VA-TCP) [121] a été spécialement conçu pour les besoins du handover vertical, dans les réseaux sans fil hétérogènes. C'est un protocole complet pour la gestion de la mobilité visant à optimiser les performances de TCP après un handover vertical. Il affecte l'émetteur et le récepteur de la pile TCP.

Le principe du VA-TCP est d'estimer dynamiquement la bande passante, le délai et le produit bande passante*délai (BDP) dans le nouveau réseau, lors de la mobilité.

Basé sur l'estimation du BDP, le VA-TCP fixe le *Slow Start Threshold* (sstresh), la fenêtre de congestion *Cwnd*, le timer de retransmission RTO et le RTT appropriés pour s'adapter dans le nouvel environnement.

- **TCP ACK-Pacing** : TCP ACK-Pacing [122] est une technique centrée sur le récepteur pour éviter la dégradation de débit après un handover vertical. Cela n'affectera pas l'émetteur TCP. Le principe consiste à ajuster le débit des acquittements au niveau du récepteur. Cette technique se base sur les conditions du réseau avant et après le handover, mais surtout en estimant la bande passante du nouveau réseau d'accès (RTT estimé par envoi de messages ICMP). Le produit bande passante*délai BDP peut donc être calculé ($BDP = \text{Bande Passante} * \text{RTT estimé}$).

Donc, le récepteur peut être en mesure de collecter ces données de la couche liaison du réseau correspondant.

- **Received Based Congestion Control** : Lors d'un handover, la bande passante peut changer brusquement. Un handover vers un réseau plus large rend le nouveau chemin d'accès sous-utilisé. Dans le cas contraire, cela entraînera de lourdes pertes. Comme la fenêtre d'émission est le minimum entre la fenêtre de réception *Awnd* et la fenêtre de congestion *Cwnd*, lorsque l'hôte mobile est le récepteur, il peut utiliser sa fenêtre *Awnd* pour contrôler le débit d'émission et prévenir un mauvais comportement TCP [123] [124].

De différentes méthodes peuvent être utilisées, comme le calcul de l'Advertised window en tant que Congestion Window puis l'utiliser pour l'estimation de la bande passante.. Cela permet de limiter les pertes au cours d'un handover vers un réseau moins large. Malheureusement, cela ne peut pas être utilisé pour un handover vers un réseau plus large puisque la fenêtre de congestion freinera l'envoi.

2 Evaluation de l'impact des protocoles TCP dans un contexte hybride

2.1 Comportement de TCP durant un handover

Cette section vise à caractériser la problématique de TCP face à des handovers, dans le contexte de mobilité et d'hétérogénéité de réseaux. Nous nous intéressons particulièrement sur aux handovers hybrides (horizontal ou vertical) puisque dans nos scénarios, la caractéristique du chemin de bout en bout varie fortement quand la commutation satellite/terrestre survient. Un grand nombre d'articles ont été publiés par la communauté recherche & développement sur le handover de TCP, principalement applicables aux réseaux terrestres sans fil ou cellulaires. Ces papiers ont identifié les problèmes soulevés par le handover TCP, présentés dans la prochaine sous-section. Certaines solutions ont été proposées et développées pour améliorer le comportement de TCP, et ont été présentées dans la section précédente. Malheureusement, jusqu'à maintenant, les réseaux satellite n'ont été pas traités.

Les analyses suivantes confirment les différents problèmes qui peuvent survenir au cours d'un handover.

2.1.1 La persistance des applications basées sur TCP

Le premier point pour maintenir une connexion d'une application TCP durant un handover est la connexion TCP elle-même. Le deuxième point est d'empêcher l'application de se déconnecter.

- **Maintien d'une connexion TCP**

TCP standard utilise une adresse IP comme un point final d'identification de socket (adresse IP+numéro de port), pour la destination et la source. Si un changement survient à l'un de ces quatre paramètres, cela déconnectera la socket.

Dans le cas d'un handover vertical, le réseau d'accès est commuté entre un serveur d'accès a un autre, conduisant à une réallocation d'adresse IP (en effet, le préfix sous réseau doit être changé, pour construire une adresse conforme avec la topologie IP du nouveau réseau, et pour autoriser un routage correct dans le réseau Internet). Ni TCP ou IP n'a été conçu pour ces problématiques.

Pour faire face au changement d'adresse IP, deux types de solution sont à envisager

- **A la couche transport** : Des extensions TCP ont été proposées pour éliminer la dépendance de la connexion à l'adresse IP. Par exemple, dans le protocole TCP-Migrate, Snoeren [125] a proposé d'ajouter deux nouvelles options pour le protocole TCP, et un nouvel «état» (MIGRATE_WAIT) dans la machine d'état TCP. Cela n'a pas été adopté en raison de l'impact trop profond sur le protocole. TCP-R (redirection) était une autre tentative similaire, ajoutant une couche supplémentaire pour cacher le lien rompu de la couche de transport.
- **A la couche réseau** : Il s'agit de la famille de protocoles. Mobile IP v4 est la première extension introduite au niveau réseau, pour résoudre cette question. Une amélioration ou des variantes ont été ajoutées dans la continuité

de MIPv4 : MIPv6, FMIP, HMIP, PMIP...etc. Le principe commun de Mobile IP est de cacher la mise à jour d'adresse aux couches supérieures.

- **Application timeout**

Même si la connexion est maintenue au niveau TCP, l'application peut ne pas faire face à l'interruption de communication pendant le handover. L'application peut considérer que la connexion est perdue (le silence), ce qui peut conduire à la déconnexion. Une option TCP [92], the User Timeout Option (UTO) permet d'annoncer un timeout à un pair, afin que l'application partage la politique du timeout.

L'option UTO est décrite à la section 2.2.1.

2.1.2 Flow Control adaptation

Le maintien de l'adresse IP, et par conséquent la connexion TCP n'est pas suffisant. Le changement dans les caractéristiques de la liaison comme la bande passante et le délai de propagation (par conséquent le BDP) après un handover survient et pose de gros problèmes pour la gestion de la congestion TCP. De nombreuses études ont démontré les différents problèmes qui peuvent survenir lors d'un handover.

Le contrôle de congestion TCP nécessite du temps pour sonder la capacité du nouveau réseau, ce qui peut induire à une sous-utilisation de la nouvelle liaison (en cas de handover vers le haut –réseau plus large-) ou des pertes sur le lien (en cas de handover vers le bas).

Il y a de nombreux effets secondaires qui peuvent survenir au cours du handover. Par exemple, une réorganisation de paquets, une augmentation de RTT, qui peut lui-même déclencher des opérations de contrôle de congestion TCP, et dégrader encore plus la performance. Si le contrôle de congestion TCP n'est pas en mesure de s'adapter rapidement aux conditions du nouveau réseau, après le handover, il peut en résulter des pertes de paquets multiples.

Aucun schéma de mobilité IP (méthode de la couche réseau) ne peut aider à résoudre ce problème. Il doit être géré à la couche transport. L'expéditeur TCP n'est pas au courant du changement de mobile node MN, et le handover peut être perçu comme une congestion impliquant un déclenchement de son contrôle de congestion. Il y a deux principaux critères qui induisent des effets secondaires sur le contrôle de flux TCP :

- La façon dont le handover est réalisée à la couche 2/3 : Make-Before-Break ou Break-Before-Make.
- L'évolution du BDP : un handover vers le haut (ce qui signifie une augmentation de BDP) ou vers le bas (diminution du BDP), et en particulier l'évolution du délai.

2.1.3 Impacts de TCP pour un handover type Break before Make

Le Break-Before-Make (BBM) veut dire que la couche 2 est interrompue durant le changement de réseau. La nouvelle liaison est établie seulement quand l'ancienne est fermée. Les effets du BBM sont résumés ci-dessous :

- **Risque de déconnexion** si la durée des perturbations est trop longue (applications et/ou TCP timeout)

- **La perte de paquets** durant l'interruption pourrait déclencher un mécanisme de congestion, alors que ce n'est "seulement" qu'une perte, ce qui conduit à **l'utilisation inefficace de la bande passante**.
- **Inutilisation de temps de connexion** : un RTO peut expirer pendant l'interruption, et conduire à un temps plus long pour récupérer la communication même après que le lien soit rétabli, puisque l'expiration RTO (multiples possible) empêche l'expéditeur de transmettre des données, jusqu'à ce que le prochain RTO expire (après le retour du lien). Si un RTO expire, la valeur du *sstresh* (le seuil du *Slow Start*) sera réduite et la reprise sera plus longue car elle commence avec une *Slow Start* avec des petites valeurs de *sstresh* et *cwnd*.

2.1.4 Impacts du TCP pour un handover type Make before Break

Le Make-Before-Break (MBB) signifie que la nouvelle couche 2 est préparée avant que l'ancienne soit désactivée. Deux liaisons sont disponibles durant une petite période de temps au moment du handover. Les effets sont résumés dans le tableau suivant :

	Handover ascendant	Handover descendant
Impact du Délai	<p>Retransmissions inutiles : le RTO expire parce que le RTT précédent n'est pas assez grand pour correspondre au nouveau. Le calcul du RTO ne peut pas s'adapter à cette augmentation inattendue.</p> <p>Des RTOs parasites se produisent, comme un effet secondaire</p>	<p>Retransmissions inutiles : Certains paquets "en vol" peuvent être reçus en désordre, si le nouveau lien livre les paquets plus rapidement que ceux envoyés par l'ancien lien. Ainsi, le récepteur interprète cela comme une perte de paquets, et déclenche une retransmission de déclenchement.</p> <p>-Convergence vers des RTOs plus lent</p>
Impact du BDP	<p>La bande passante inutilisée / gaspillage de ressources : TCP peut ne pas être en mesure d'utiliser la bande passante accrue du nouveau réseau.</p>	<p>une vraie congestion / perte de paquets : Le flux TCP sera trop élevé pour la bande passante plus faible, et le nouveau réseau peut ne pas être en mesure de livrer les paquets. La congestion du réseau qui se passe est vraie, les paquets peuvent être supprimés, et le contrôle de flux TCP activé.</p> <p>RTOs parasites</p>

Tableau 11 : Impact du MBB handover sur TCP

2.2 Résultats et analyses

Après une phase d'étude théorique, les résultats de test nous ont menés à des résultats incohérents. En effet, des tests réalisés sur TCP ont montré de nombreuses contradictions avec le comportement défini dans la théorie. En effet, les expériences que nous avons menées présentent des incohérences par rapport aux recommandations IETF RFC [72]. Cela dépend du système d'exploitation, de la

configuration des buffers en émission et en réception, de l'utilisation du Windows Scalling ou pas...etc. Ces manipulations sont connues sous le nom de TCP Tuning.

C'est la raison pour laquelle nous avons opté pour une approche "boite noire". Les mesures que nous réalisons avec cette approche permettent d'analyser le fonctionnement de bout en bout de TCP, dans un contexte hybride et avec une mobilité. En effet, cette étude présentera le comportement de TCP lors d'un handover depuis un réseau spécifique vers un autre (depuis un réseau satellite vers un autre terrestre et vice versa), ainsi que sa capacité d'adaptation. Des scénarios comme le "Break-Before-Make" permettent d'analyser la capacité de récupération de TCP après une coupure, ce qui est le cas généralement lors d'un handover.

La démarche des tests à suivre est incrémentale, c'est-à-dire que nous limitons la modification des paramètres (délai ou bande passante), avant d'envisager un handover plus général. Cette démarche permet de voir quel élément impacte plus que l'autre lors des handovers.

Pour analyser facilement le comportement TCP face à un handover vertical (VHO), nous avons mis en place une approche de test incrémental, où nous avons d'abord limitée la modification à un paramètre (délai ou bande passante), avant d'envisager un handover plus général, lorsque le BDP change.

Notre choix sur les versions de TCP à comparer s'est fait sur les bases précédentes parce que Cubic est la version par défaut sur Linux, Compound TCP est celle par défaut sur les machines Windows, Hybla est la versions recommandé sur les liens satellites [84], Westwood utilise un contrôle de congestion basé sur l'estimation de la bande passante et enfin New Reno en tant que référence.

2.2.1 Le handover vertical avec la variation de délai

Le handover avec uniquement le changement du délai permet d'analyser son impact sur le protocole de transport. Lors d'une baisse brusque du délai de propagation, le comportement des protocoles de transport change, les paquets et les acquittements arrivent plus rapidement et créant de possibles "déséquences", le SACK peut remédier à ce problème ou bien faire "exploser" les buffers plus rapidement. Dans le cas inverse, c'est-à-dire une augmentation brusque de délai de propagation, des retards sont à prévoir, les paquets arrivent plus lentement ainsi que les acquittements. Cela peut générer des pertes ou des fausses détections de pertes puisque les timers (et tous les paramètres de connexion) sont adaptés à la session précédente (réseau avec un faible débit). Cette expérience permet de dégager le meilleur protocole, dans le sens le plus adapté à ce genre de situation de changement brusque de délai.

a) La configuration

La bande passante a été fixée à 2 Mbit/s durant tout le test. Le délai de propagation change lors d'un handover. Le handover survient chaque 20 secondes. Les 20 premiers le RTT est fixé à 500 ms, puis passe à 50 ms dans la deuxième phase. Enfin, on remet la valeur du RTT à 500 ms pendant la dernière étape. Le taux d'erreur est à 0.

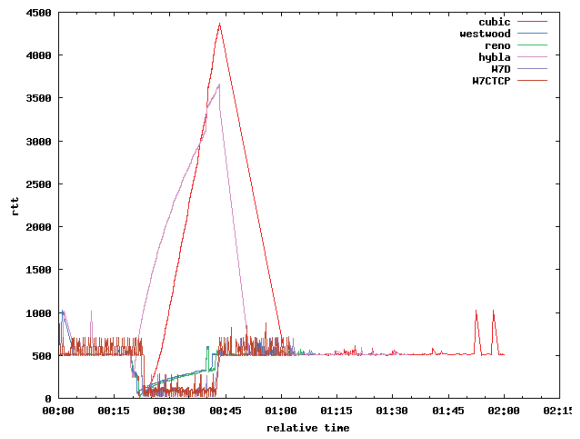
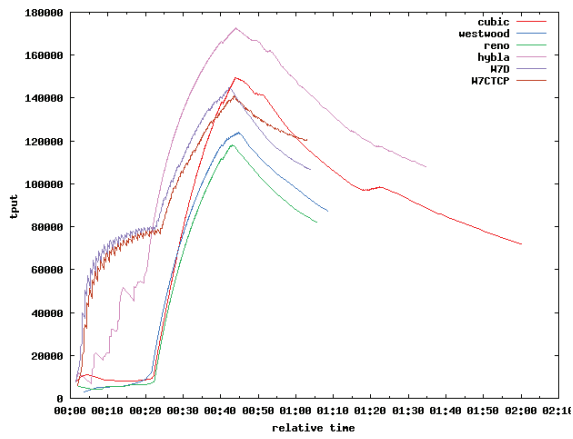
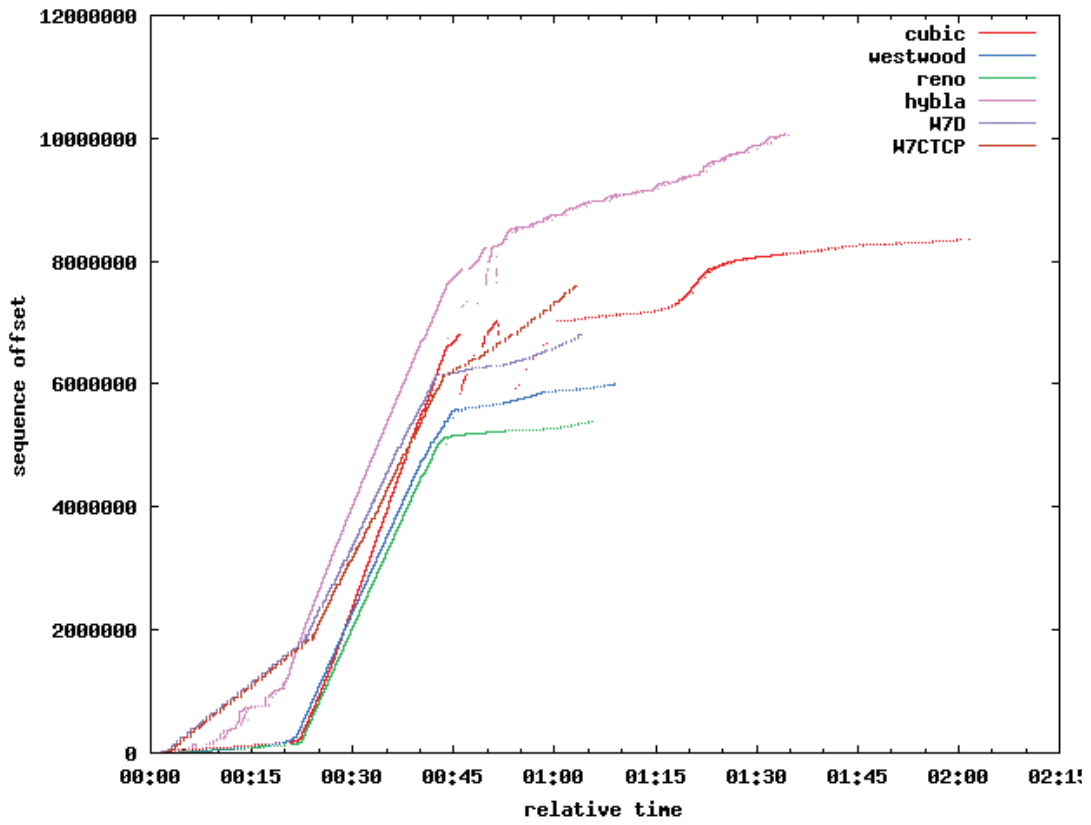


Figure 71 : Evolution des numéros de séquence, débit et RTT Durant 2 handover de délai

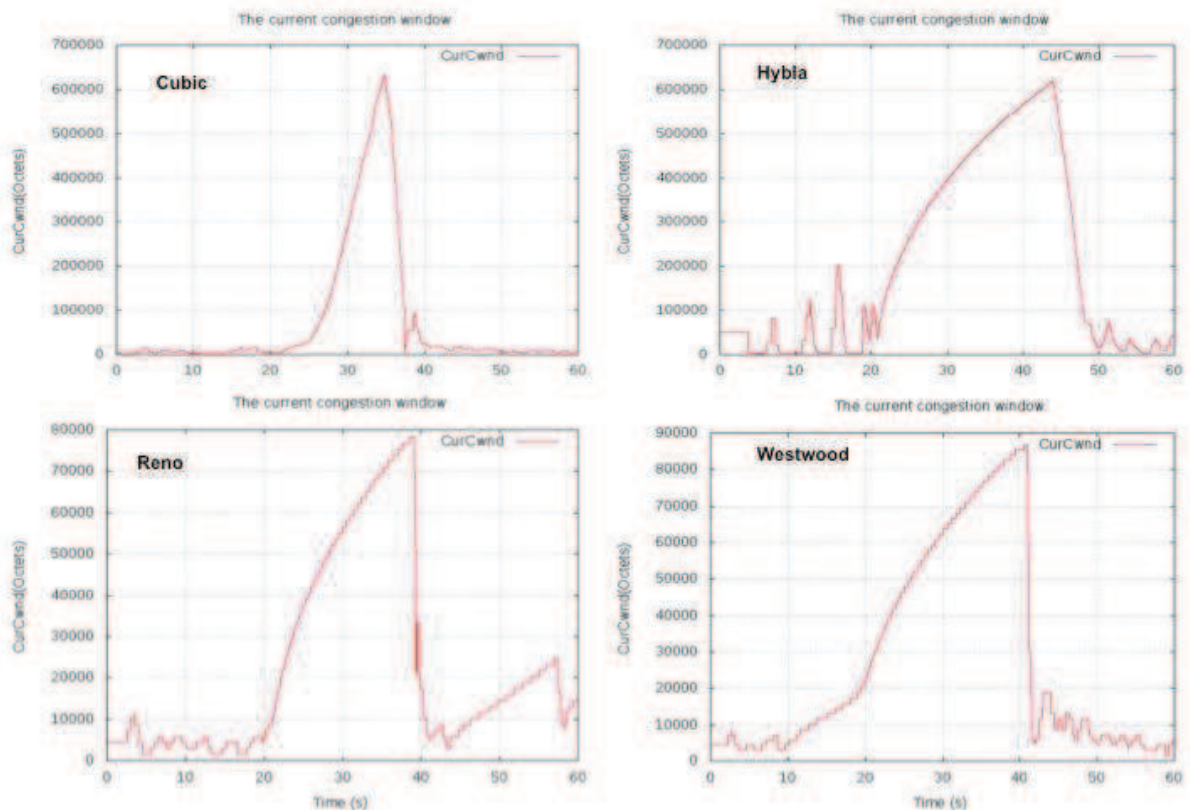


Figure 72 : Evolution de la Cwnd Durant un handover de délai

b) Analyse

La Figure 71 présente les numéros de séquence durant deux handovers. Comme nous l'avons vu dans le chapitre précédent, Hybla et CTCP ont de bonnes performances sur le lien satellite mieux que Westwood et New Reno.

Lors d'un handover d'un réseau à grand délai vers un autre à bas délai, les versions TCP sont réactives et se comportent plutôt bien. Dans un réseau à petit délai (réseau terrestre), les délais de bufferisation diminuent créant un phénomène de déblocage. En effet, les données et les acquittements arrivent plus rapidement. Cubic profite des caractéristiques actuelles pour mieux évoluer à cause de son agressivité.

Lors d'un handover vers un réseau a grand délai, Cubic et Hybla souffrent le plus de ce changement. Cubic a besoin de 10 secondes pour se stabiliser et offre une mauvaise gestion de la fenêtre de congestion, et génère des pertes. Cela est dû à sa gestion de fenêtre de congestion. Cubic évolue sa fenêtre de congestion suivant une fonction cubique dans le temps [78], depuis le dernier événement de congestion. Cette évolution se fait même sans attendre le retour des acquittements. Cela est problématique lors du changement de réseau car il se confronte brusquement à un réseau à grand délai. Son agressivité le pénalise puisque les buffers se remplissent très rapidement (grands délais) impliquant des pertes. L'équation de l'évolution de la fenêtre de congestion de Cubic est donnée ci-dessous :

$$W_{cubic}(W_0, t) = C(t - K)^3 + W_0$$

Équation 8 : Equation d'évolution de la fenêtre de congestion de Cubic

Avec W_0 est la taille de la fenêtre de congestion lors du dernier événement de congestion, t est le temps depuis le dernier événement de congestion, $K = \sqrt[3]{W_0 \beta / C}$, C et β sont des constantes spécifiques à Cubic. Quand un paquet est supprimé, la fenêtre est réduite à $(1 - \beta)W_0$.

Bien que Hybla présente plus de robustesse et résiste plutôt bien par rapport à Cubic, il génère des pertes lors de ce changement de réseau. Hybla est conçu pour des réseaux types satellites mais lors d'un changement vers un réseau plus long (grands délais), il subit des erreurs. Cela est dû à son mécanisme d'estimation Hoer [126] dans la phase précédente. Cet algorithme consiste à envoyer plusieurs paquets au même moment et obtenir une estimation de la capacité actuelle du canal à partir des délais des réceptions des acquittements.

CTCP récupère plus rapidement et offre de meilleures performances que la version de Windows 7. En effet, Compound TCP présente moins de problèmes de bufferisation, d'évolution et d'adaptation par rapport aux autres versions. Cet avantage est dû à l'utilisation de deux fenêtres de congestion pour l'évolution du débit. La fenêtre utilisée pour l'envoi des données et la somme de deux fenêtres. La première fenêtre est classique, basée sur les pertes $Cwnd$. La deuxième est basée sur les délais $Dwnd$. Cette dernière est estimée en utilisant les délais de bufferisation dans le réseau en utilisant les RTTs.

$$Window = \min (Cwnd + Dwnd, Awnd)$$

Équation 9 : Evolution de la fenêtre de congestion de CTCP

Où $Cwnd$ et la fenêtre de congestion basé sur les pertes, $Dwnd$ celle basée sur les délais et $Awnd$ et la fenêtre de réception. Les équations de la $Dwnd$ et $Cwnd$ sont présentées dans l'Équation 4.

Les débits atteints pendant les 20 premières secondes (grand délai) atteignent le triple pour CTCP par rapport aux Westwood, Cubic et New Reno et approximativement le double d'Hybla. Après le premier handover, les débits s'égalent dans l'ensemble.

En termes de retransmission, nous nous intéressons spécialement au passage vers le réseau à grand délai, c'est-à-dire le passage de la phase 2 à 3. CTCP présente le taux de pertes le plus faible avec moins de 0.01 %, contre presque 3 % pour Cubic, 2% pour Hybla, et moins de 0.1 % pour New Reno et Westwood. Ces deux dernières versions voient leur taux de retransmissions assez bas mais le nombre de paquet émis ainsi que le débit reste très faible par rapport aux autres versions (approximativement 40 % de moins) et implique une sous-utilisation de la bande passante.

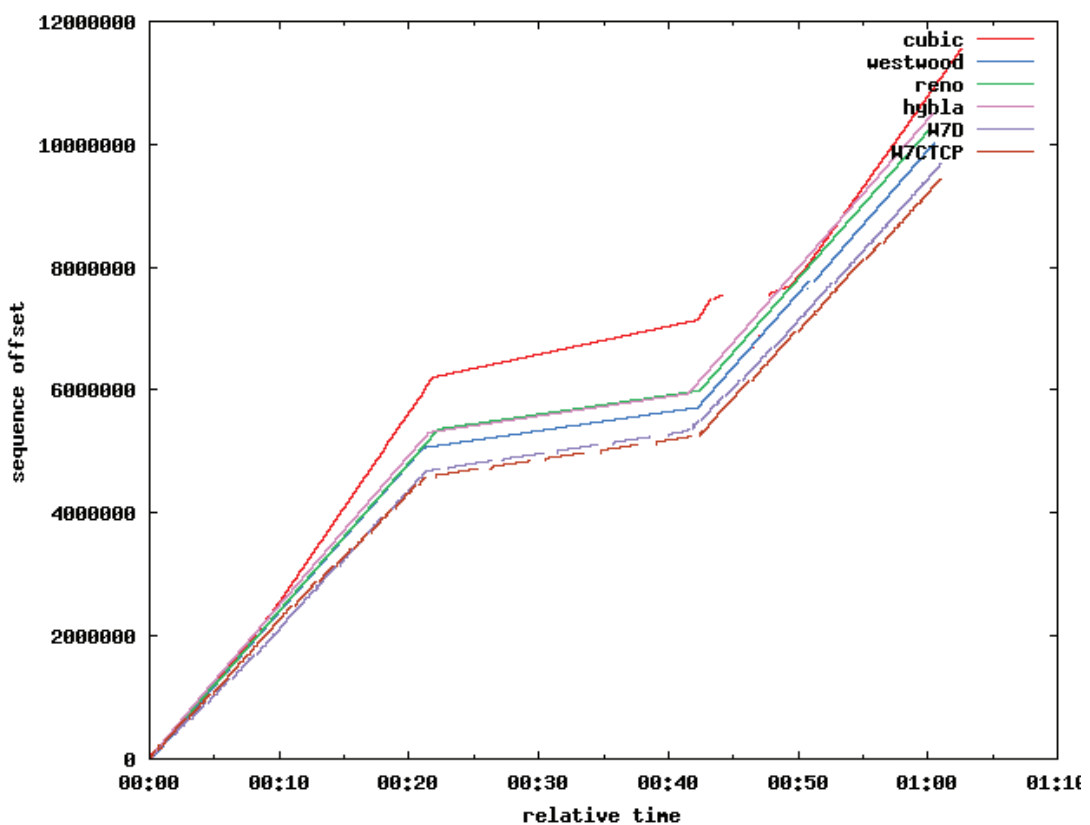
La Figure 72 présente l'évolution des fenêtres de congestion des versions TCP sous linux et confirme les performances de Cubic et Hybla. La fenêtre de congestion de Cubic est très basse et signifie qu'il n'atteint pas un grand débit sur le réseau satellite ce qui confirme l'analyse qui précède. Westwood sous-utilise la bande passante spécialement lors d'un grand délai. New Reno est testé comme référence et ne présente pas de bonnes performances sur le réseau satellite. La Figure 72 confirme les résultats sur Cubic et Hybla lors d'un handover "de délai" observé précédemment.

2.2.2 Le handover vertical avec la variation de bande passante

Le handover avec le changement de la bande passante est intéressant pour comprendre le comportement de TCP dans ces conditions et pour analyser l'impact de ce changement sur le protocole de transport. En effet, les variations (chutes ou augmentations) de bande passante influencent le comportement d'un protocole de transport TCP. Lors d'une augmentation de bande passante, le plus intéressant est de voir la réactivité des différentes versions à atteindre le nouveau débit. Dans le cas inverse, c'est-à-dire une diminution de la bande passante, l'intérêt est porté sur l'adaptation à cette "chute" de débit. En effet, lors de la baisse de débit, des pertes sont attendues ce qui va déclencher des mécanismes de récupération rapide (*Fast Recovery*), et ainsi éviter de grandes périodes sans transmission, ou un redémarrage total (*Slow Start*) impliquant une perte de temps et de ressources. Cela permet de distinguer les versions et d'en tirer "la meilleure" ou plutôt la plus adaptée aux réseaux hybrides.

a) La configuration

On fixe le RTT à 50 ms pendant toute l'expérience, et on change le débit de 512 kbit/s dans les 20 premières secondes, à 2 Mbit/s dans les 20 prochaines secondes, puis retour à 512 kbit/s dans les dernières 20 secondes. Le taux de perte est à 0.



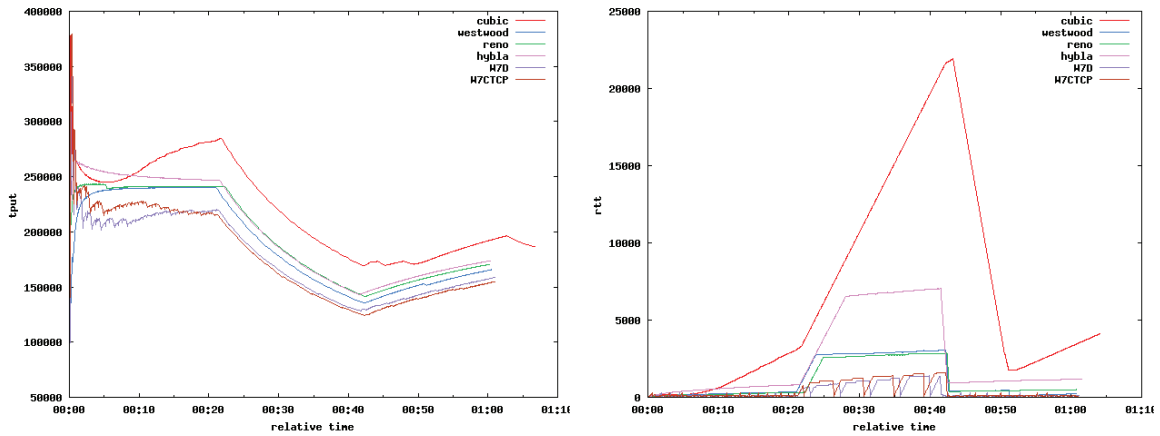


Figure 73 : Evolution des numéros de séquence, débits et RTT Durant un handover de bande passante

b) Analyse

Toutes les versions de TCP ont, généralement, un bon comportement. Cubic est affecté, comme le test précédant par le handover vers un réseau à bas débit. Tous les flux ont un comportement plus ou moins similaires, à l'exception de Cubic qui souffre à la fin de la phase 2. A ce moment, les buffers sont bien remplis et l'interruption à la seconde 43 est une conséquence de l'agressivité de Cubic, qui essaye toujours d'avoir le plus possible de bande passante.

Cubic est affecté à cause de sa gestion de la fenêtre de congestion. Le taux de retransmission de Cubic sur la phase 3 est égal à 4,3 % et reste une valeur assez élevée par rapports aux autres versions (moins de 0.1 % pour Hybla, Westwood, CTCP et New Reno).

2.2.3 Le handover vertical avec la variation simultanée de la bande passante et le délai

Le handover du produit bande passante*délai et celui qui se rapproche le plus d cas réel. Cette expérience permet de comprendre le comportement des versions TCP lors des handovers des réseaux types satellites vers d'autres types terrestres, ou vice versa. Cela permet aussi d'extraire le meilleur protocole et le plus adapté à des réseaux hybrides satellites/terrestres avec des changements apparents de délais et bande passante.

Des solutions d'améliorations et d'optimisations de communications existent (comme nous l'avons cité au début du chapitre). La plus utilisée des solutions est le PEP. Ce dernier apporte des améliorations à la transmission, en "protégeant" le segment terrestre des caractéristiques du réseau satellite, ainsi qu'avec des mécanismes d'accélération et d'optimisations de flux. Néanmoins, ils présentent plusieurs inconvénients comme l'incompatibilité avec des connexions cryptées et la rupture de la sémantique de bout en bout d'une connexion TCP. De plus, les PEPs peuvent être pénalisants lors d'une mobilité vers un réseau type terrestre.

a) La configuration

Dans cette expérience, nous allons faire varier le délai et la bande passante en même temps pour reproduire des conditions qui se rapprochent des caractéristiques d'un réseau satellite puis d'un

réseau WiFi. Nous testons aussi PEPsal, qui est un PEP intégré (*Integrated PEP*), pour mesurer l'impact des handovers sur les PEPs.

Le test est initié à 512 kbit/s/500 ms pendant les 20 premières secondes puis un handover survient à $t=20$ vers un réseau terrestre (2Mbit/s/50ms). A $t=40$ s, un autre handover survient pour retourner au premier réseau. Le taux d'erreur est à 0.

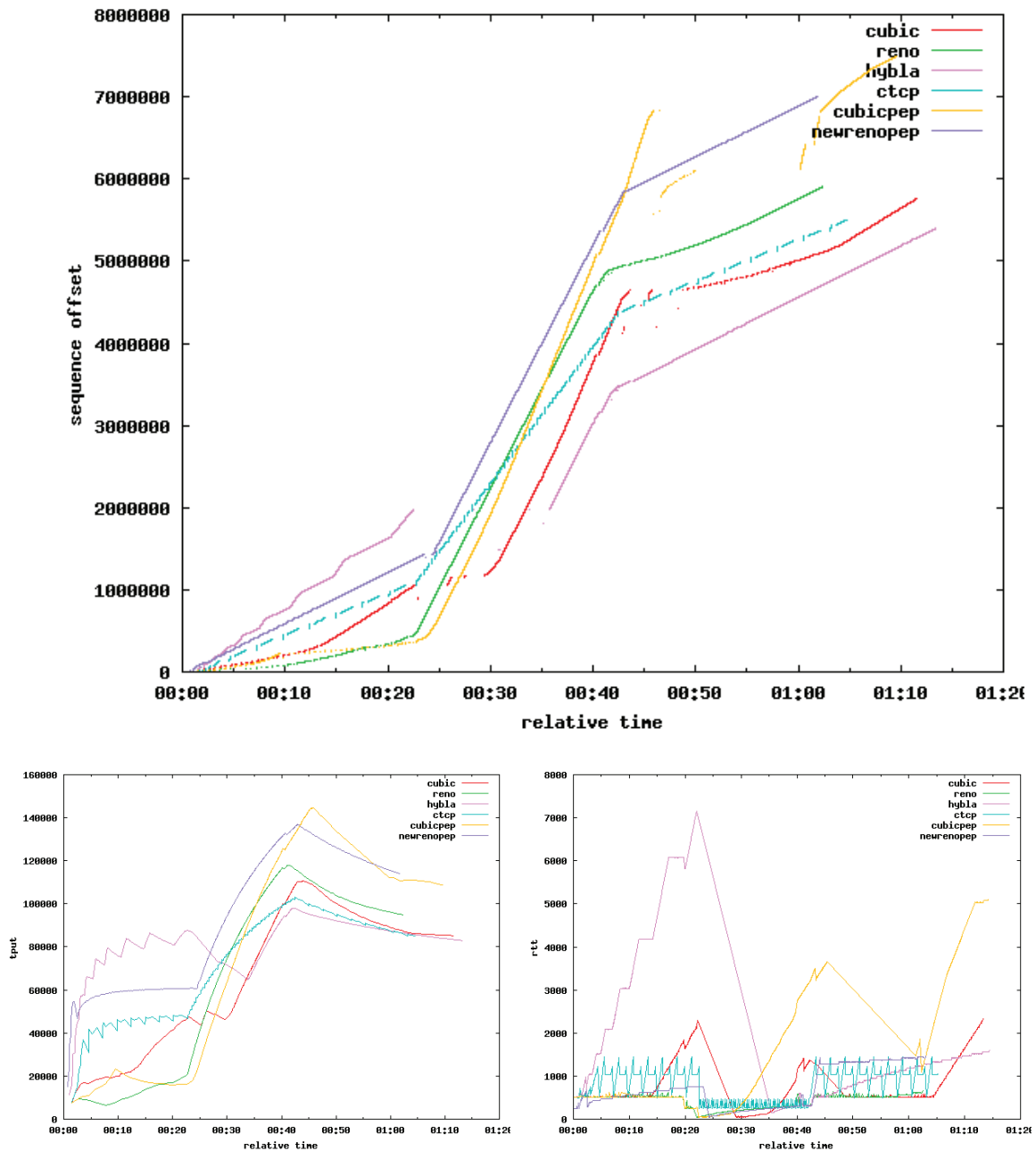


Figure 74 : Evolution des numéros de séquence, débits et RTT Durant un handover de la passante/délai

b) Analyse

Dans la Figure 74, l'évolution du New Reno avec PEPsal est très acceptable sur le réseau satellite. On observe aussi une bonne réaction d'Hybla dans cette phase, mais avec une grande coupure au premier handover. New Reno a un mauvais comportement et confirme son incompatibilité avec les réseaux satellite. CTCP est stable et offre un débit comparable aux solutions PEP. Le débit moyen d'Hybla arrive à de bons niveaux mais avec des petites variations. Le dernier graphe représente le RTT applicatif et présente les problèmes de bufferisation d'Hybla, ce qui explique la coupure à la seconde ~20.

2.2.4 Le handover vertical avec la variation simultanée de la bande passante et le délai avec ECN

En cas d'absence de PEP, les paquets arrivent très rapidement du segment terrestre sur le segment satellite signifiant une augmentation du délai de bout en bout et créant un remplissage rapide des buffers des routeurs d'accès satellite. Ce phénomène engendre des pertes de paquets et des fluctuations de débits ce qui rend la communication médiocre. Cette problématique de bufferisation affecte spécifiquement deux versions que nous avons étudiées à savoir Cubic et Hybla. D'un côté, Cubic est conçu pour les réseaux LFN (réseau à haut débit et avec un grand délai) est très agressif dans les réseaux satellite. Cette agressivité vient du fait qu'il n'attend pas les acquittements pour augmenter sa fenêtre de congestion (la fenêtre de congestion suit une fonction cubique dans le temps), il est confronté par la suite aux limitations de la bande passantes ce qui crée ces fluctuations. De l'autre côté, Hybla tient son agressivité de sa conception. Bien que cette version soit conçue pour les réseaux satellite, mais Hybla présente des difficultés puisque l'utilisation de son algorithme de base (Hoe's Bandwidth Estimation Algorithm) n'est pas toujours favorable à cause des mauvaises estimations de la capacité de la bande passante dans les réseaux hybrides.

Cependant, CTCP ne présente pas le problème de bufferisation vu l'utilisation des deux fenêtres de congestion pour l'évolution du débit. Lors du changement de réseau, la fenêtre de délai est réévaluée et si ce délai ne varie pas, il suppose que le lien n'est pas congestionné et augmente rapidement le débit. C'est la raison pour laquelle CTCP n'est pas confronté à ce problème.

ECN permet de palier aux problèmes de bufferisation parce qu'il permet de notifier explicitement les congestions depuis les routeurs d'accès. Il permet de déclencher des mécanismes de congestion TCP avant une congestion afin de maintenir le niveau buffers à une valeur raisonnable dans le réseau congestionné. Dans les réseaux hybrides, la congestion se produit généralement au niveau du routeur de réseau d'accès satellite, ce routeur génère généralement une restriction de bande passante sur l'accès satellite. La finalité est de réguler le débit d'émission, et qui implique une réduction des délais de bufferisation et une diminution de pertes.

a) La configuration

C'est la même configuration que le test précédant, sauf que l'activation d'ECN est faite dans le réseau satellite. La configuration au niveau du SATEM se fait à l'aide de la commande TC (Traffic Controller) en utilisant les paramètres Set 1 puis Set 2 (se référer au Tableau 10 : Configuration d'ECN dans notre plateforme du chapitre 3).

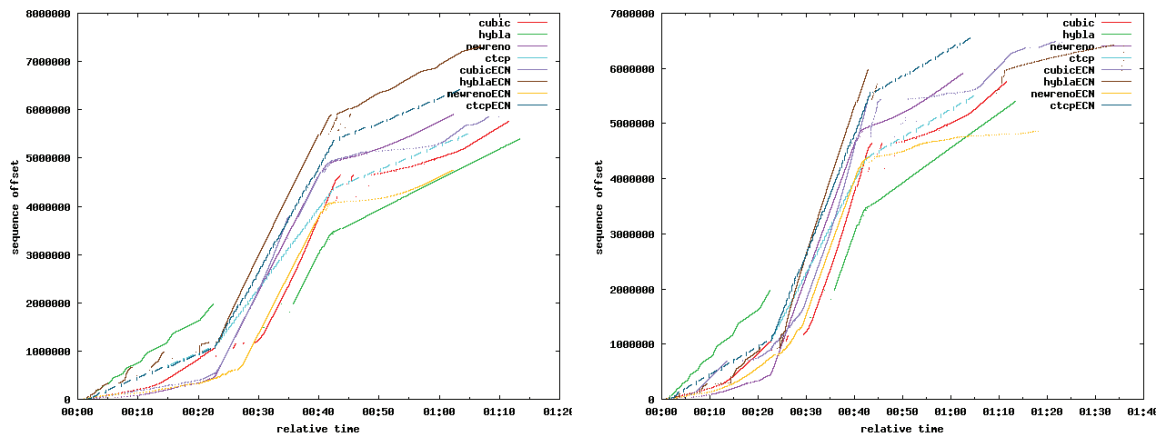


Figure 75 : Evolution des numéros de séquence durant un handover de la bande passante/délai et ECN configuré (Set 1 et 2)

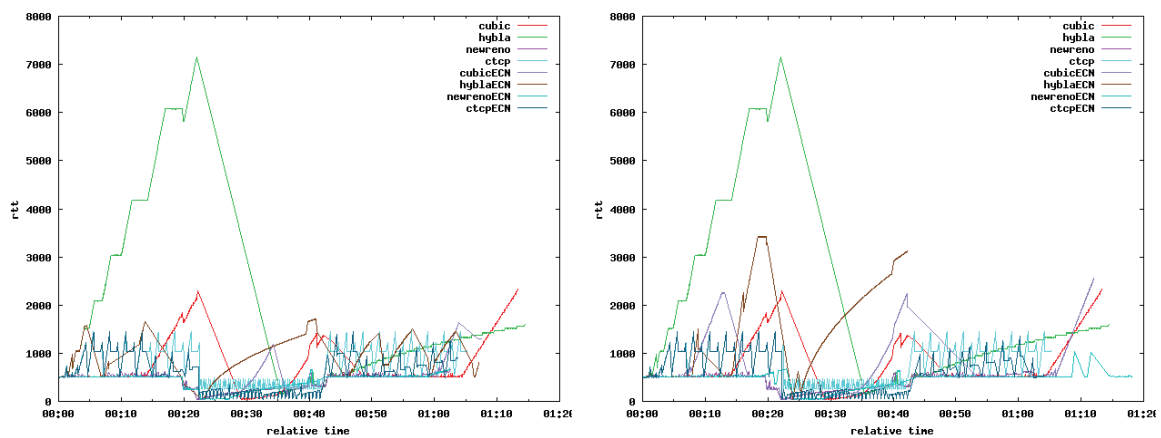


Figure 76 : Evolution des RTTs durant un handover de la bande passante/délai et ECN configuré (Set 1 et 2)

b) Analyse

Les versions utilisant l'ECN sont plus stables à cause des mécanismes de marquage et suppression volontaire avant les congestions. En outre, sur le réseau terrestre (où l'ECN est désactivé) le gain est plus significatif. Le handover avec des files vides semble avoir un bon résultat contrairement à des handovers avec des files pleines. Hybla présente des performances supérieures mais est très sensible et instable. Avec l'ECN, les délais restent contrôlés et bas. Cela préserve l'interactivité du système.

Comme nous l'avons vu dans le chapitre précédent, CTCP minimise au maximum ses pertes. Grâce aux deux fenêtres de congestions, le débit est maintenu stable et à son maximum sans subir de pertes. Cela implique la meilleure gestion de débit et une meilleure utilisation de la bande passante sans gaspillage de ressources. Ces derniers points sont très importants pour les réseaux satellite vu le coût des ressources.

2.2.5 L'impact du "Break Before Make"

Cette partie compare les versions TCP lors d'un handover qui n'est pas exempt d'erreur. La durée de l'interruption dure 500 ms puis 1000 ms.

- Break de (500/1000 ms) Before Make, et bande passante fixée à 2Mbit/s et une variation délai (500-50-500 ms RTT)

La Figure 77 (a), présente le comportement de TCP lors d'un handover de 500ms d'interruption. Quand les flux traversent le lien satellite sur la première phase, CTCP confirme son avantage par rapport aux autres versions. La même chose pour PEPsal et Hybla qui se placent en derrière en termes de performance. Après la première interruption, le redémarrage de tous les flux prend au moins 5 secondes pour Cubic, Hybla et PEPsal. CTCP réagit plus rapidement et récupère la transmission en moins de 2 secondes. Nous avons eu les mêmes conclusions à l'interruption du handover à la seconde 40.

Dans le graphe (b), quand le break prend 1 seconde, nous observons le même comportement après les handovers, mais avec plus de difficultés pour Cubic.

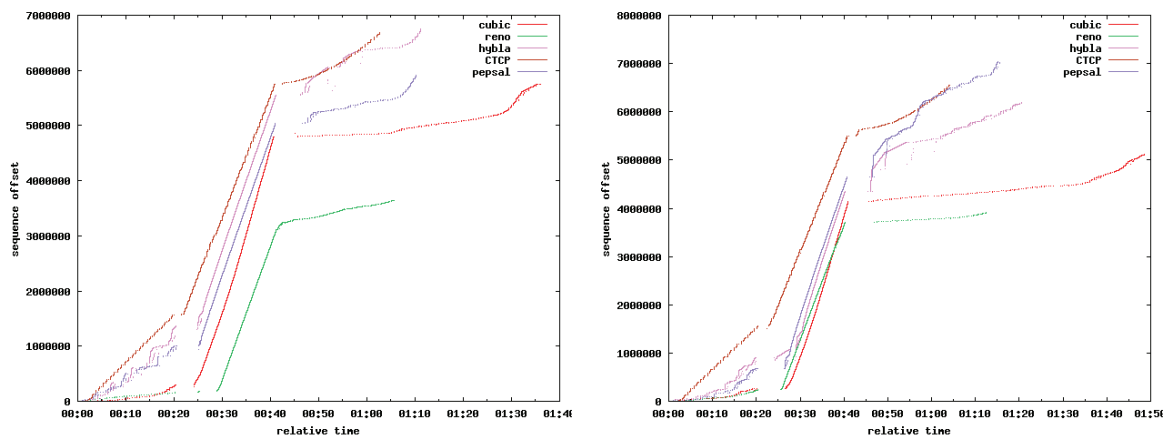


Figure 77 : Evolution des numéros de séquence avec une coupure de (a) 500 ms (b) 1000 ms

- Break (500/1000 ms) Before Make, et un délai fixe de 50 ms et une variation de bande passante (512K-2M-512K bps)

Dans la Figure 78 (a), nous observons au début de la transmission que Cubic est meilleur que les autres versions, cela est dû grâce au bas délai (mais avec une petite bande passante). Durant les deux handovers, Cubic a une grande coupure et redémarre difficilement. Hybla et PEPsal ont naturellement le même résultat et réagissent de la même façon durant les handovers. CTCP a une meilleure gestion de la coupure et récupère plus rapidement avec un temps de récupération d'environ 2 secondes.

Quand l'interruption dure 1 seconde, toutes les connexions à l'exception de CTCP, sont interrompues plus d'une seconde à cause de la rupture de la liaison sur la couche 2/3.

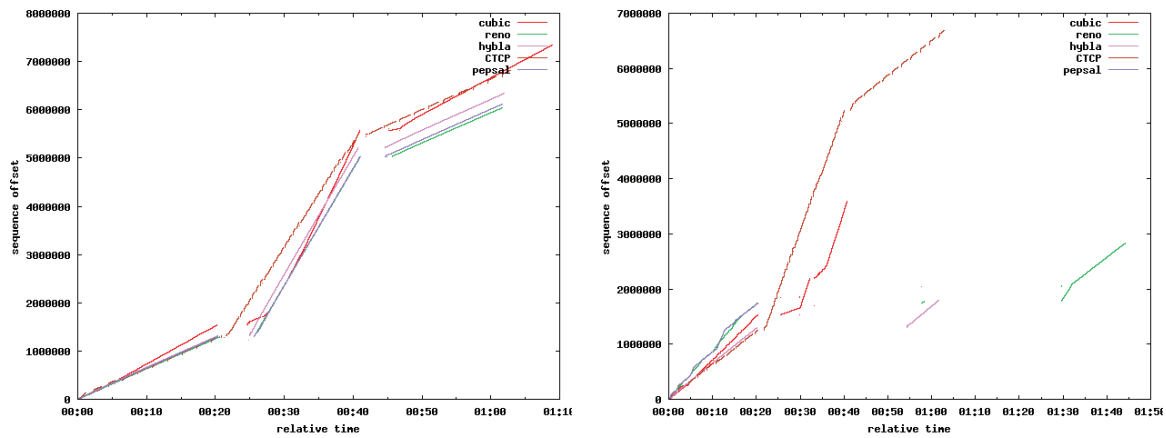


Figure 78 : Evolution des numéros de séquence avec une coupure de (a) 500 ms (b) 1000 ms

Les tests de handover BBM présentent les difficultés rencontrées par certaines versions de TCP durant les handovers et attestent que CTCP gère mieux les handovers avec ou sans interruption.

Lors d'une rupture de connexion durant un handover, le RTO expire dans la majeure partie des cas. Si l'interruption prend plus de temps (comme le cas de 1 seconde de coupure), produisant des mécanismes de congestion et impliquant des pertes. Quand un RTO expire, la valeur du Ssthreshold (seuil du *Slow Start*) est réduite. Le redémarrage en *Slow Start* sera plus long. A ce moment-là, un timeout au niveau de l'application se produit et crée une interruption totale de la transmission pour la majorité des versions.

Contrairement aux autres versions, CTCP gère mieux cette interruption. En effet, Compound TCP utilise la fenêtre de congestion basée sur le délai pour maintenir la connexion

2.2.6 Comparaison des moyennes

Afin de confirmer nos résultats, les tests ont été réalisés au minimum 12 fois. Les figures suivantes montrent la faible dispersion de mesures, ce qui explique pourquoi nous nous sommes contentés de présenter qu'une courbe par figure afin d'en améliorer la lisibilité du reste du mémoire.

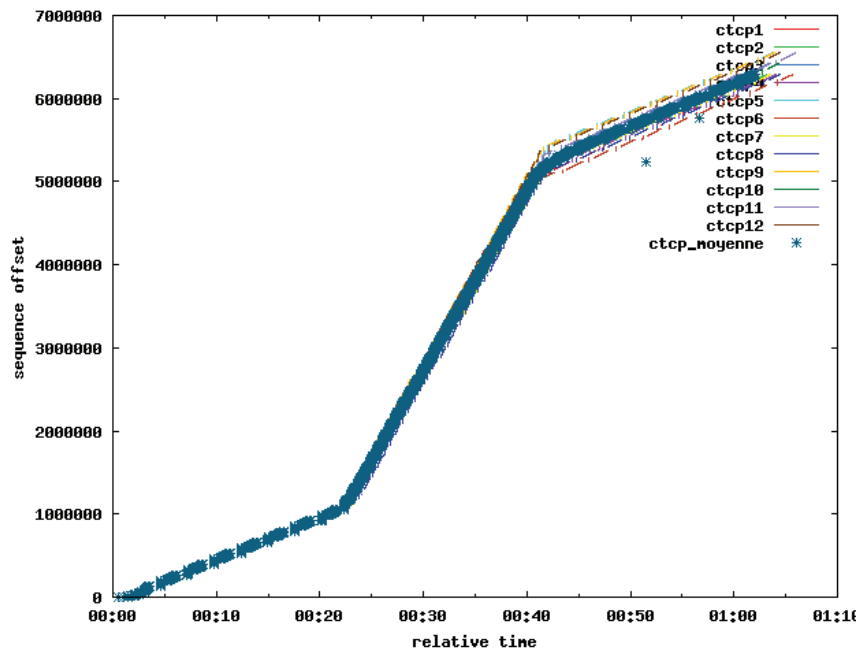


Figure 79 : Evolution des numéros de séquence pour 12 flux CTCP

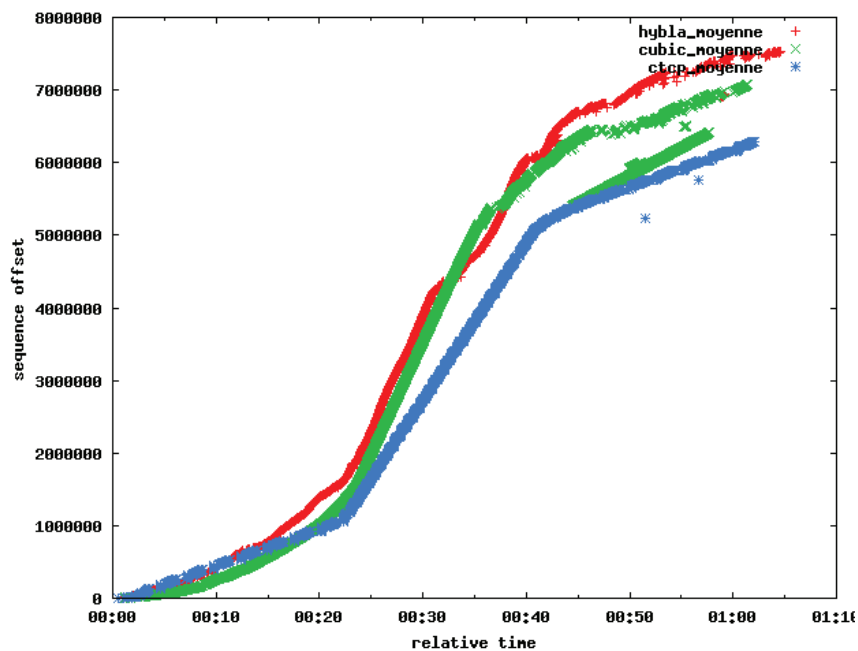


Figure 80 : Evolution des numéros de séquence de la moyenne de 12 flux de chaque version de TCP

La Figure 79 montre que CTCP a un comportement prédictible quand les conditions sont stables. Tous les flux CTCP ont un comportement presque identique.

La Figure 80 présente la comparaison des moyennes de trois versions, CTCP, Cubic et Hybla.

2.3 Conclusion

Les tests de changement de réseaux montrent que CTCP a les meilleures performances. Cubic et Hybla présentent aussi de bons résultats mais sont agressifs surtout lors d'un passage vers un réseau plus long (augmentation de délai) ou moins large (diminution de bande passante). De ce fait, une

grande bufferisation est effectuée par ces deux versions, causant de grands délais d'attente et ainsi des pertes potentielles. ECN permet de remédier à ce problème en stabilisant les délais à cause des mécanismes de marquage et de suppressions volontaires avant les congestions.

De plus, sur le réseau terrestre (où l'ECN est désactivé) le gain est plus significatif. Avec l'ECN, les délais restent contrôlés et bas. Cela préserve l'interactivité du système déjà réduite par le délai du lien satellite.

3 Conclusion

Lors d'un handover du réseau satellite vers le réseau Wifi, nous n'avons pas relevé de problème majeur, du moment que la version de TCP utilisée est efficace sur satellite. Toutes les versions de TCP ont la capacité d'exploiter rapidement le surplus de bande passante.

Dans l'autre sens, c.-à-d. un handover du réseau Wifi vers le réseau satellite, la réduction du débit ne pose pas vraiment de problème. Par contre, l'augmentation du délai de propagation cause de nombreux problèmes, qui s'expliquent par un soudain surplus de bufferisation sur le routeur d'accès. La réduction de débit et conjointement l'augmentation du délai génèrent de nombreuses pertes qui sont pénalisantes pour les protocoles trop agressifs. Ctcp et Hybla sont peu perturbés par le handover, contrairement à Cubic. L'utilisation de PEP peut améliorer la situation, du fait de l'utilisation de Hybla (Hybla PEPsal). Toutefois, nous avons montré que l'utilisation de Ctcp permet de se passer d'une solution PEP.

De plus, nous avons montré que Ctcp s'adapte bien au changement de délai, puisque c'est le seul protocole à intégrer la variation du délai pour son contrôle de congestion. Un contrôle de congestion basé à la fois sur les pertes et sur les délais est donc la solution pour résoudre le problème de l'hybridation.

D'autre part, l'ECN permet de résoudre les problèmes d'agressivité de certaines versions comme Cubic et Hybla. La bufferisation est gérée au mieux lors du changement de réseau vers les réseaux satellite.

Des scénarios de type "Break-Before-Make" ont été étudiés. Quand la coupure de connexion au moment du changement de réseau dure 500 ms, Ctcp confirme sa supériorité sur les autres versions. En effet, Ctcp est plus réactif au redémarrage que Cubic et Hybla. Quand la coupure de connexion au moment du changement de réseau dure 1000 ms, la difficulté s'amplifie pour Cubic, conséquence de son agressivité.

A la vue de ce qui précède, notre approche qui s'appuie sur la supériorité générale du comportement de Ctcp constitue donc une solution de niveau Transport performante pour les réseaux hybrides. Non seulement, nous avons démontré dans le chapitre précédent son adaptation aux réseaux satellite mais nous démontrons aussi ici sa capacité à supporter les changements de réseaux incontournables dans un réseau hybride satellite/terrestre.

Conclusion générale

Nos travaux ont concerné l'évaluation de la Qualité de Service dans les réseaux satellite. Après une première partie concernant des mesures faites sur des systèmes réels, nous avons focalisé nos propositions sur l'adaptation de la couche transport dans les réseaux hybrides satellite/terrestre. Ces derniers représentent la solution idéale pour un accès Internet mobile, permanent et facile à mettre en œuvre, puisque les réseaux satellite ont une grande zone de couverture. Dans le cas où le réseau satellite fait défaut (tunnel, zone non couverte...etc.), le réseau mobile de 3ème ou 4ème génération est la solution idéale pour une complémentarité parfaite.

Dans notre démarche, et afin de proposer la solution d'adaptation, nous avons réalisé un certain nombre de test sur l'émulateur satellite SATEM. Les paramètres utilisés ont été déduits d'une campagne de test faite auparavant sur un lien réel.

A partir des résultats obtenus sur le vrai lien satellite, nous avons pu tester de nombreuses versions de TCP, en émulation et en utilisant SATEM. Les premiers résultats ont montré l'apport des nouvelles versions à savoir Hybla, Cubic et Compound TCP. Par contre, New Reno, présenté comme la solution standard dans les réseaux terrestres, n'est pas adapté pour les réseaux satellite.

Le grand défaut relevé sur les versions Cubic et Hybla est la bufferisation provoquée dans le routeur congestionné, par leur agressivité et sanctionnant les performances de TCP. Pour régler ce problème de bufferisation, l'ECN permet de diminuer le délai de bout en bout ainsi que d'améliorer les performances des communications. Bien que le débit subisse une diminution, les délais de bout en bout sont significativement diminués.

Par la suite, nous avons comparé les nouvelles versions de TCP avec des solutions PEPs. Nous avons conclu que les PEPs ne sont plus tellement un avantage par rapport aux nouvelles versions de TCP. Il n'est donc plus forcément nécessaire d'utiliser des solutions PEPs dans les réseaux satellite du moment que CTCP est utilisé, éventuellement Cubic.

Concernant le cas d'un réseau hybride, c'est-à-dire, lors d'un handover du réseau satellite vers le réseau Wifi, nous n'avons pas relevé de problème majeur, du moment que la version de TCP utilisée est efficace sur satellite. Toutes les versions de TCP ont la capacité d'exploiter rapidement le surplus de bande passante.

Dans le cas inverse, c'est-à-dire un handover du réseau Wifi vers le réseau satellite, la réduction du débit ne pose pas vraiment de problème. Par contre, l'augmentation du délai cause de gros problèmes, qui peuvent s'expliquer par un soudain surplus de bufferisation au niveau des buffers. La réduction de débit et conjointement l'augmentation du délai génère de nombreuses pertes qui peuvent être pénalisantes pour les protocoles trop agressifs. CTCP et Hybla sont peut perturbés par le handover, contrairement à Cubic qui supporte moins cette transition.

Compound TCP s'adapte bien au changement de délai, puisqu'il utilise un contrôle de congestion supplémentaire basé sur le délai. La combinaison d'un contrôle de congestion basé sur les pertes et sur les délais semble donc idéale pour résoudre le problème de l'hybridation.

En outre, l'ECN permet de résoudre les problèmes d'agressivité et la bufferisation pour certaines versions comme Cubic et Hybla. Bien qu'une baisse de débit soit introduite, la bufferisation est mieux gérée lors du changement de réseau vers les réseaux satellite.

Généralement, lors d'un handover, un délai est nécessaire au redémarrage des flux TCP. Deux scénarios ont été étudiés le BBM et le MBB. Les résultats ont montré une bonne adaptation de CTCP relativement à Hybla ou Cubic. Ces deux derniers souffrent lors d'un changement de réseaux à cause de leur agressivité. CTCP confirme son bon comportement et gère mieux le redémarrage après la coupure.

En complément à TCP, DCCP a été étudié en tant que nouveau protocole pour les échanges non fiables. L'utilisation de DCCP est prometteuse et peut apporter un plus dans les réseaux satellite puisque c'est un protocole à datagramme utilisant des contrôles de congestion et non un flux d'octets comme TCP. L'utilisation de DCCP sur la voie aller DVB-S2/ACM permet une meilleure gestion de certains types d'application vidéo, spécialement de type streaming. Une proposition utilisant des signalisations cross layer a été faite qui minimise le délai d'adaptation de débit lors des changements de codage de l'ACM et anticipe le changement de codec en émission pour des applications de type streaming.

Nous avons analysé le comportement de plusieurs versions TCP sur un réseau satellite et hybride satellite/terrestre pour dégager les différents problèmes liés à la couche transport dans les réseaux satellite et hybrides. Elle permet aussi d'apporter des solutions pour une utilisation optimale de la couche transport dans les réseaux satellite.

Au final, notre proposition en s'appuyant soit sur Compound TCP qui est la version par défaut sur les systèmes Windows et soit sur Cubic qui est celle par défaut sur les systèmes Linux facilite le passage à l'échelle des solutions proposées.

Nous avons ainsi proposé des solutions qui répondent aux nombreuses problématiques posées en s'appuyant sur de nouvelles architectures capables de fonctionner sur des réseaux hybrides satellites/terrestres.

Perspectives

Les travaux précédents ouvrent un certain nombre de perspectives.

Les études des protocoles de transport que nous avons réalisé démontrent que malgré tous les travaux réalisés sur ce sujet, un certain nombre de problèmes se pose encore et pas seulement dans le cas d'un réseau par satellite.

Par exemple, certains phénomènes observés sur les implémentations de Cubic que nous avons utilisé sont à approfondir comme par exemple la gestion des retransmissions et les timers. D'une manière générale, nous avons remarqué que les implémentations TCP diffèrent plus au moins des références théoriques. En effet, Linux gère TCP avec quelques libertés par rapport au standard, donnant de bons résultats mais un paramétrage complexe est à prévoir (tailles des buffers en émission, en réception, le windows scalling, min RTO...etc.). Il sera donc intéressant de dégager la configuration adéquate pour les réseaux satellite et les réseaux hybrides satellite/terrestre.

Concernant les réseaux hybrides satellites/terrestres, l'étude s'est concentrée sur l'émulation avec SATEM. Bien que cette méthode ait donné suite à de bons résultats, un nouvel outil de simulation satellite (OpenSand [47]) aujourd'hui disponible en open source permettrait d'affiner les résultats. En effet, notre émulateur SATEM réalise une émulation au niveau 3, qu'OpenSand complète par une simulation de mécanismes de niveau 2 dont en particulier la boucle d'allocation DAMA de DVB-RCS.

La mobilité est gérée dans notre approche en tenant compte des spécificités du réseau hybride. Il serait intéressant de traiter celle-ci comme elle se fait aujourd'hui dans Internet au niveau 3 avec des solutions telles que Mobile IPv6 ou MIH pour une gestion universelle des handovers. D'autre part, on pourrait également utiliser des solutions telles que mSCTP pour gérer cette mobilité directement au niveau Transport.

Un autre point important concerne le cas de transmission entre deux utilisateurs mobiles. Un changement de réseau pour les deux utilisateurs est possible et impliquerait des répercussions sur les réseaux.

Le cas de mobilité entre 3 réseaux (Satellite, WiFi et 3G) n'a pas été étudié. Ce cas sera intéressant puisque les réseaux de nouvelles générations incluent ces trois réseaux. L'utilisation de tous les supports de transmission permet un accès quasi permanent à Internet.

Référence

- [1] O. Alphand, *Architecture à qualité de service pour systèmes satellites DVB-S/RCS dans un contexte NGN.*: Rapport LAAS N°05672, Décembre 2005.
- [2] ETSI: EN 300 421 V1.1.2 (1997-08) Digital Video Broadcasting (DVB) and channel coding and modulation for the 11/12 GHz satellite services. Framing structure,.
- [3] ETSI: ETSI EN 301 790 V1.4.1 (2005-04) Digital Video Broadcasting (DVB) and Interaction channel for satellite distribution systems,.,
- [4] IST SATSIX project. [Online]. <http://www.ist-satsix.org/>
- [5] Digital Video Broadcasting (DVB) ETSI EN 302 307 V1.2.1 (2009-08) and channel coding and modulation systems for Broadcasting, Interactive Services, News Gathering and other broadband satellite applications (DVB-S2) Second generation framing structure,.
- [6] Le Tooway. [Online]. <http://www.internetparsatellite.net/index.php/t/15/n/337/tooway.html>
- [7] A. Pietrabissa F. Delli Priscoli, "Design of a bandwidth-on-demand protocol for satellite networks modeled as timedelay systems," *Automatica*, vol. 40, no. 5, Mai 2004.
- [8] The mobile satellite company. [Online]. <http://www.inmarsat.com/>
- [9] *BGAN and IP data connections, Version 01.*, 2006.
- [10] ASTRA 1L SES. [Online]. <http://www.ses.com/4628656/astra-1l>
- [11] S. Choi, G. R. Hiertz, O. Klein and B. Walke S. Mangold, "Analysis of IEEE 802.11e for QoS Support in Wireless Lans," *IEEE Wireless Communications*, Décembre 2003.
- [12] Frédéric Nivor, *Architecture de communication pour les applications multimédia interactives dans les réseaux sans fil.*, 2009.
- [13] Ancillary Terrestrial Component (ATC). [Online]. <http://transition.fcc.gov/ib/sd/ssr/atc.html>
- [14] Arif Ansari Santanu Dutta and Michael Tseytlin, "S-WiMAX: Adaptation of IEEE 802.16e for Mobile Satellite Services," *IEEE Communications Magazine*, June 2009.
- [15] ITU-T, *Transmission systems and media, Digital systems and networks - Quality of service and performance - End-user multimedia QoS categories, Recommendation G.1010.*, Novembre 2001.
- [16] STEINMETZ R., "Human Perception of Jitter and Media Synchronisation," *IEEE Journal on Selected Areas in Communications*, vol. 14, no. 1, Janvier 1996.

- [17] SHNEIDERMAN B., "Response Time and Display Rate in Human Performance with Computers," *ACM Computing Surveys (CSUR)*, vol. 16, no. 3, Septembre 1984.
- [18] Projet OURSES "Aerospace Valley". [Online]. <http://www.ourses-project.fr/>
- [19] ZigBee. [Online]. <http://www.zigbee.org/>
- [20] D. Clark and S. Shenker R. Braden, *RFC 1633, Integrated Services in the Internet Architecture : an Overview, IETF, June 1994.*
- [21] L. Zhang, S. Berson, S. Herzog and S. Jamin R. Braden, *RFC 2205, Resource ReSerVation Protocol (RSVP) – Version 1 Functional Specification, IETF, September 1997.*
- [22] C Partridge and R. Guerin S. Shenker, *RFC 2212, Specification of Guaranteed Quality of Service, IETF, September 1997.*
- [23] J. Wroclawski, *RFC 2211: Specification of the Controlled-Load Network Element Service, IETF, September 1997.*
- [24] Black D., Carlson M. Blake S., *RFC 2475, An Architecture for Differentiated Services, IETF, Décembre 1998.*
- [25] Poduri K. Jacobson V., *RFC 2598, An Expedited Forwarding PHB, IETF, Juin 1999.*
- [26] Baker F., Weiss W. Heinanen J., *RFC 2597, Assured Forwarding PHB Group, IETF, Juin 1999.*
- [27] ETSI Broadband Satellite Multimedia. [Online]. <http://www.etsi.org/website/technologies/broadbandsatmultimedia.aspx>
- [28] *ETSI TR 102 157 V1.1.1 (2003-07), Satellite Earth Stations and Systems (SES); Broadband Satellite Multimedia;IP Interworking over satellite;Performance, Availability and Quality of Service.*
- [29] *ITU-T Y.1541 Series Y: Global Information Infrastructure, Internet protocol aspects and next generation networks.*, 2006.
- [30] 3GPP Specification Change Request records for: 23.107. [Online]. <http://www.3gpp.org/ftp/Specs/html-info/23107-CRs.htm>
- [31] *UIT-T P.10/Amd.1, Vocabulary for performance and quality of service.*
- [32] ITU-T, *ITU-T G.1080, Series G: Transmission systems and media digital, Digital systems and networks. Multimedia quality of service and performance – Generic and user-related aspects.*
- [33] ITU-T, *G.107 : The E-model: a computational model for use in transmission planning.*

- [34] UIT-T T.140, *Protocol for multimedia application text conversation*.
- [35] R., Ahmed, K. and Brunnstrom K. Kooij, *Perceived quality of channel zapping, fifth IASTED International Conference Spain, pp. 155-158., 2006*.
- [36] Thierry Gayraud, Ihsane TOU, Lionel Bertaux, Michel Diaz, Fabrice Planchou, Valentin KRETZSCHMAR, Gelard Patrick, Emmauel Dubois Pascal Berthou, "RT&CNES, Satperf Project," LAAS-CNRS, CNES, EADS ASTRIUM,.
- [37] Centre National des Etudes Spatiales. [Online]. <http://www.cnes.fr>
- [38] Vinent DESLANDES, *Partage de spectre dans les systèmes intégrés.*, Juin 2012.
- [39] Vivek Gupta and Vijay Raman Ajoy Singh, *IEEE 802.21 Media Independent Handover Services, http://www.ieee802.org/21, January 2006*.
- [40] Perkins C., Arkko J. Johnson D., *RFC 3775, Mobility Support in IPv6, IETF, Juin 2004*.
- [41] C. Castelluccia, K. El Malki and L. Bellier H. Soliman, *RFC 4140, Hierarchical Mobile IPv6 Mobility Management (HMIPv6), IETF, August 2005*.
- [42] P. Calhoun, J. Bharatia C. Perkins, *RFC 4721, Mobile IPv4 Challenge/Response Extensions (Revised), IETF, Janvier 2007*.
- [43] Francesco Potorti, Raffaello Secchi Alberto Gotta, "Simulating Dynamic Bandwidth Allocation on Satellite Links. University of Genoa," *WNS2 '06, 2006*.
- [44] The Network Simulator - ns-2. [Online]. <http://www.isi.edu/nsnam/ns/>
- [45] The simulator manual. [Online]. <http://www.abdn.ac.uk/~eng863/simulator-manual.pdf>
- [46] P. Berthou, T. Gayraud, F. Nivor, B. Jacquemin, D. Barvaux and J. Nicol C. Baudoin, "PLATINE: DVB-S2/RCS enhanced testbed for next generation satellite networks," in *IP Networking over Next-Generation Satellite Systems, International Workshop, Budapest.*, July 2007.
- [47] OpenSAND, satellite emulation platform. [Online]. <https://launchpad.net/opensand>
- [48] *IST SATIP6 project, IST-2001-34344 contract*.
- [49] *TR102157 v1.1.1. ETSI TC SES; Broadband Satellite Multimedia; IP Interworking over satellite: Performance, Availability and Quality of Service.*, July 2003.
- [50] *SatLabs System Recommendations – Quality of Service Specifications.*, June 2008.
- [51] LabView software for metering and processing data from laboratory equipments. [Online]. <http://www.ni.com/labview/whatis/>

- [52] Traffic Control short explanation. [Online]. <http://tldp.org/HOWTO/Traffic-Control-HOWTO/software.html#s-iproute2>
- [53] Netem – Network Emulation tool for Traffic Control. [Online]. <http://www.linuxfoundation.org/collaborate/workgroups/networking/netem>
- [54] Iptables – Linux tools for firewall configuration. [Online]. <http://en.wikipedia.org/wiki/Iptables>
- [55] STM GROUP, INC. [Online]. <http://www.stmi.com>
- [56] STM, *SatLink VSAT User Guide*.
- [57] France Satellite: <http://www.francesatellite.com>.
- [58] "Projet E-Cab ,".
- [59] IPERF, Générateur de flux. [Online]. <http://iperf.fr/>
- [60] M. Handley, S. Floyd E. Kohler, *RFC 4340, Datagram Congestion Control Protocol (DCCP), IETF, Mars 2006*.
- [61] Q. Xie, K. Morneault, C. Sharp, H. Schwarzbauer, T. Taylor, I. Rytina, M. Kalla, L. Zhang, V. Paxson R. Stewart, *RFC 2960, Stream Control Transmission Protocol, IETF, October 2000*.
- [62] TCPTRACE. [Online]. <http://www.tcptrace.org/>
- [63] Wireshark. [Online]. <http://www.wireshark.org/>
- [64] D. L. Mills, *RFC 1305, Network Time Protocol (Version 3) Specification, Implementation and Analysis, IETF, March 1992*.
- [65] Jugi's Traffic Generator (JTG). [Online]. <http://www.cs.helsinki.fi/u/jmanner/software/jtg/>
- [66] B. Jacquemin, *Architecture pour la mobilité et la qualité de service dans les systèmes satellites DVB-S2/RCS.*, Juin 2010.
- [67] The Multi-Generator (MGEN). [Online]. http://pf.itd.nrl.navy.mil/mgen/mgen.html#_MGEN_Log_File
- [68] OreNETa (One Way Delay Real-time Network Analyser). [Online]. <http://cba.upc.edu/orenet>
- [69] Matt Mathis John Heffner Raghu Reddy, *Web100: Extended TCP Instrumentation for Research, Education and Diagnosis, ACM SIGCOMM Computer Communications Review, Volume 33, Number 3., July 2003*.
- [70] Web100. Kernel Instrument Set, 2002. [Online]. <http://www.web100.org/download/kernel/alpha2.0/tcpkis.txt>

- [71] J. Heffner, R. Reddy, and J. Saperia M. Mathis, *TCP Extended Statistics MIB. Work in progress: IETF Internet-Draft, November 2002. Status page: <http://www.web100.org/mib>.*
- [72] P. Sarolahti and A. Kuznetsov, *Congestion Control in Linux TCP. In Proceedings of Usenix 2002/Freenix Track, pp. 49--62. Monterey, CA, USA., June 2002.*
- [73] Paxson V., Stevens W. Allman M., *RFC 2581: TCP Congestion Control, IETF Network Working Group, 1999.*
- [74] R. Firrincieli, D. Lacamera C. Caini, "A Performance Enhancing Proxy for TCP satellite connections, in: Interworking and Resource Management in Satellite Systems Special Series," *IEEE Aerospace and Electronic Systems Magazine* , vol. 22, 2007.
- [75] Padmanabhan V, Fairhurst G, Sooriyabandara Balakrishnan H, *RFC 3449, M. TCP performance implications of network path asymmetry, BCP 69. IETF, December 2002.*
- [76] V. Paxson, E. Blanton M. Allman, *RFC 5681, TCP Congestion Control, IETF, 2009.*
- [77] T. Henderson, A. Gurtov S. Floyd, *RFC 3782, The NewReno Modification to TCP's Fast Recovery Algorithm, IETF, 2004.*
- [78] I. Rhee, L. Xu S. Ha, "CUBIC: a new TCP-friendly high-speed TCP variant," *ACM SIGOPS Operating Systems Review*, 2008.
- [79] Q. Zhang, M. Sridharan K.T. Jingmin Song, "A Compound TCP approach for High-speed and Long Distance Networks," *Infocom*, 2006.
- [80] Q. Zhang, M. Sridharan K.T. Jingmin Song, "Compound TCP: A Scalable and TCP-Friendly Congestion Control for High-speed Networks - Protocols for Fast Long-Distance Networks," *PFLDNet*, 2006.
- [81] E. Kristiansen C. Roseti, *TCP Noordwijk: optimize TCP-based transport over DAMA in satellite networks, International Communications Satellite Systems Conference., 2008.*
- [82] J. Wen, J. Zhang, Y. Han – J. Wang, *TCP-FIT: An Improved TCP Congestion Control Algorithm and its Performance, Infocom'2011., October 2010.*
- [83] J. Wen, J. Zhang, Y. Han J. Wang, *A Demonstration of a new TCP Congestion Control Algorithm over LTE and other Challenging Networks, Mobicom., 2010.*
- [84] R. Firrincieli C. Caini, "TCP Hybla: a TCP enhancement for heterogeneous networks," *International Journal of Satellite Communications and Networking*, 2004.
- [85] T. Başar, R. Srikant S. Liu, *TCP-Illinois: A Loss and Delay-Based Congestion Control Algorithm for High-Speed Networks, VALUETOOLS., 2006.*

- [86] A. F. Harris, C. J. Hughes S. V. Adve, *The Illinois GRACE Project: Global Resource Adaptation through CoopEration.*: Proceedings of the Workshop on SHAMAN, Juin 2002.
- [87] R. Shorten D. Leith, *H-TCP: TCP for high-speed and long-distance networks*, Proc. PFLDnet., 2004.
- [88] IANA – TCP options assignment. [Online]. <http://www.iana.org/assignments/tcp-parameters/tcp-parameters.xml>
- [89] J. Mahdavi, S. Floyd and A. Romanow M. Mathis, *RFC 2018 : TCP Selective Acknowledgment Options*, IETF Standard Track, 1996.
- [90] M. Allman, K. Fall and L. Wang E. Blanton, *RFC 3517 : A Conservative Selective Acknowledgment (SACK)-based Loss Recovery Algorithm for TCP* IETF Standard Track, 2003.
- [91] R. Braden and D. Borman V. Jacobson, *RFC 1323 : TCP Extensions for High Performance* IETF Standard Track, IETF, 1992.
- [92] L. Eggert and F. Gont, *RFC 5482 : TCP User Timeout Option*, IETF Standard Track, 2009.
- [93] S. Floyd and C. Partridge M. Allman, *RFC 3390 : Increasing TCP's Initial Window*, IETF Standard Track, 2002.
- [94] N. Dukkipati, Y. Cheng and M. Mathis J. Chu, "draft-ietf-tcpm-initcwnd-01 : Increasing TCP's Initial Window," *IETF Internet Draft*, 2011.
- [95] M. Allman, A. Jain and P. Sarolahti S. Floyd, *RFC 4782 : Quick-Start for TCP and IP*, IETF Experimental, 2007.
- [96] K. Avrachenkov, U. Ayesta, J. Blanton and P. Hurtig M. Allman, *RFC 5827 : Early Retransmit for TCP and Stream Control Transmission Protocol (SCTP)*, IETF Experimental, 2010.
- [97] R. Ludwig and M. Meyer, *RFC 3522 : The Eifel Detection Algorithm for TCP*, IETF Experimental, 2003.
- [98] R. Ludwig and A. Gurtov, *RFC 4015 : The Eifel Response Algorithm for TCP*, IETF Standard Track, 2005.
- [99] E. Blanton and M. Allman, *RFC 3708 : Using TCP Duplicate Selective Acknowledgement (DSACKs) and Stream Control Transmission Protocol (SCTP) Duplicate Transmission Sequence Numbers (TSNs) to Detect Spurious Retransmissions*, IETF Experimental, 2004.
- [100] M. Kojo, J. Griner, G. Montenegro, Z. Shelby, J. Border, *RFC 3135, Performance Enhancing Proxies Intended to Mitigate Link-Related Degradations*, IETF, June 2001.
- [101] C. Caini et al., "Analysis of TCP live experiments on a real GEO satellite testbed," *Performance*

Evaluation, vol. 66, 2009.

- [102] S. Floyd, D. Black K. Ramakrishnan, *RFC 3168, The Addition of Explicit Congestion Notification (ECN) to IP*, IETF, Septembre 2001.
- [103] V. Jacobson S. Floyd, *Random Early Detection Gateways for Congestion Avoidance*, *IEEE/ACM Transactions on Networking.*, 1993.
- [104] T. Yamamoto, "Estimation of the advancedTCP/IPalgorithms for longdistancecollaboration," *Fusion Engineering and Design*, vol. 83, no. 2-3, pp. Pages 516–519, April 2008.
- [105] Average Web Page Size Septuples Since 2003. [Online]. <http://www.websiteoptimization.com/speed/tweak/average-web-page/>
- [106] Random Early Detection Description. [Online]. <http://linux.die.net/man/8/tc-red>
- [107] E. Kohler S. Floyd, *RFC 4341, Profile for Datagram Congestion Control Protocol (DCCP), Congestion Control ID 2: TCP-like Congestion Control*, IETF, March 2006.
- [108] E. Kohler, J. Padhye S. Floyd, *RFC 4342, Profile for Datagram Congestion Control Protocol (DCCP) Congestion Control ID 3: TCP-Friendly Rate Control (TFRC)*, IETF, March 2006.
- [109] E. Kohler S. Floyd, *Profile for Datagram Congestion Control Protocol (DCCP) Congestion ID 4: TCP-Friendly Rate Control for Small Packets (TFRC-SP).*, August 2009.
- [110] R. Boreli and E. Lochin. G. Sarwar, *Performance of VoIP with DCCP for Satellite links*. ICC., 2009.
- [111] G. Fairhurst A. Sathiaseelan, *Performance of VoIP using DCCP over DVB-RCS Satellite Network*, ICC 2007.
- [112] S. Floyd, J. Padhye and J. Widmer M. Handley, *RFC 3448, TCP Friendly Rate Control (TFRC):Protocol Specification*, IETF, Jan. 2003.
- [113] G. Fairhurst A.Sathiaseelan, *TCP-Friendly Rate Control (TFRC) for bursty media flows*, *Comput. Commun.*(2011).
- [114] S. Floyd, J. Padhye and J. Widmer E. Kohler, *RFC 5348, TFRC TCP Friendly Rate Control (TFRC): Protocol Specification*, IETF, Sep. 2008.
- [115] ETSI: *TS 102 441 V1.1.1 (10/05) DVB-S2 Adaptive Coding and Modulation for Broadband Hybrid Satellite Dialup Applications*.
- [116] S. Floyd and A. Sathiaseelan E. Kohler, *Faster Restart for TCP Friendly Rate Control (TFRC)*, IETF Draft 06, Jul. 2008.
- [117] M. Gineste, C. Baudoin, P. Berthou, T. Gayraud F. Nivor, *Cross-layer anticipation of resource allocation for multimedia applications based on SIP signaling over DVB-RCS Satellite System*.

Budapest, Hongrie: IP Networking over Next-generation Satellite Systems Workshop, INNSS 2007, Juillet 2007.

- [118] T. Goff, J. Moronski, D.S. Phatak, and V. Gupta, "Freeze-TCP: A true end-to-end TCP enhancement mechanism for mobile environments," *INFOCOM*, 2000.
- [119] R. Jain, G. Dommety C. So-In, *PETS: PEristent TCP using Simple freeze, Future Information Networks.*, 2009.
- [120] K. Sleurs, E. Van Lil, A. Van de Capelle D. Li, *A fast adaptation mechanism for TCP vertical handover, Advanced Technologies for Communications.*, 2008.
- [121] H. P. Chang Y. C. Lin, *VA-TCP: a vertical handoff-aware TCP- Proceeding Applied Computing.*, 2007.
- [122] M. Takahiro, Y. Miki M. Yosuke, *TCP Congestion Control with ACK-Pacing for Vertical Handover, IEICE Transactions on Communications.*, 2007.
- [123] H Kawahara, Y. Oie H. Kaga, *TCP Flow Control Using Link Layer Information in Mobile Networks - SPIE proceedings series.*, 2002.
- [124] D. Pearce, P. Mitchell Y. Gou, *A Receiver-based Vertical Handover Mechanism for TCP Congestion Control - IEEE transactions on wireless communications.*, 2006.
- [125] Snoeren. Draft TCP Migrate. [Online]. <http://nms.lcs.mit.edu/publications/draft-snoeren-tcp-migrate-00.txt>
- [126] J. C. Hoe, "Improving the start-up behavior of a congestion control sheme for TCP," *ACM SIGCOMM*, 1996.
- [127] BGAN France. [Online]. <http://www.bgan-france.com>