

TABLE DES MATIERES

DEDICACES.....	ii
REMERCIEMENTS.....	iii
RESUME.....	iv
ABSTRACT.....	v
TABLE DES MATIERES.....	vi
LISTE DES FIGURES.....	vii
LISTE DES TABLES.....	viii
LISTE DES ACRONYMES.....	ix
PUBLICATIONS ET CONFERENCES.....	x

INTRODUCTION GENERALE.....	1
1. CONTEXTE ET PROBLEMATIQUE DE LA THESE.....	1
2. CONTRIBUTION ET STRUCTURE DE LA THESE.....	3

CHAPITRE I : CONCEPTS DE BASE

1. GENERALITES SUR LES RESEAUX SANS FIL.....	6
1.1. Introduction.....	6
1.2. Définition.....	6
1.3. Les différentes technologies des réseaux sans fil.....	7
1.4. Différents normes de réseaux sans fil.....	8
1.5. Spécificités des réseaux sans fil.....	10
1.5.1. Spécificité physique.....	10
1.5.2. Erreur du canal.....	10
1.5.3. Contention du Médium et collision.....	11

1.5.4. Mobilité.....	12
1.5.5. Spécificité du routage.....	13
1.5.6. Congestion.....	13
1.5.7. Considérations énergétiques.....	14
1.6. Les réseaux sans fils ad hoc mobiles.....	14
1.7. Caractéristiques des réseaux ad hoc.....	15
2. PROTOCOLE DE TRANSPORT.....	17
2.1. Introduction.....	17
2.2. Le protocole TCP.....	18
2.2.1. Caractéristiques et fonctionnement général.....	18
3.FONCTIONS DE CONTROLE DE CONGESTION	
3.1. La phase slow-start (démarrage lent).....	22
3.2. Congestion avoidance (évitement de congestion).....	23
3.3. L’algorithme Additive Increase and Multiplicative Decrease (AIMD).....	23
3.4. La reprise sur erreur.....	23
3.4.1. Fast retransmit (retransmission rapide).....	24
3.4.2. Fast-recovery (recouvrement rapide).....	24
3.4.3. Selective Acknowledgment (SACK).....	24
4.VARIANTES DE TCP.....	25
4.1. TCP Tahoe.....	25
4.2. TCP Reno.....	25
4.3. TCP New Reno.....	26
4.4. TCP Vegas.....	26
4.5. TCP Westwood+.....	26
4.6. TCP SACK.....	27
5.TCP ET LES RESEAUX SANS FIL.....	27
5.1. Problèmes de TCP dans les réseaux ad hoc mobiles.....	28
6.LE PROTOCOLE HTTP.....	30
7.LES SERVICES DE VIDEO STREAMING.....	32
7.1. Introduction.....	32

7.2. Le streaming vidéo.....	32
7.3. Le système de streaming multimédia.....	33
8.LES DIFFERENTES TECHNOLOGIES DE STREAMING.....	34
8.1. Le streaming en direct et le streaming stocké.....	34
8.1.1. Streaming adaptatif.....	35
8.1.2. Le streaming non adaptatif.....	35
9. Conclusion.....	36

CHAPITRE II : ETAT DE L'ART

1.INTRODUCTION.....	37
2. LA QUALITE DE SERVICE.....	37
2.1. Définition.....	38
3. LA QUALITE D'EXPERIENCE.....	39
3.1. Définition.....	39
3.2. Les approches de la QoE.....	40
3.2.1. Les approches objectives.....	40
3.2.2.Modèles de planification réseau.....	41
3.2.2. Approche subjective.....	41
4.L'APPROCHE CROSS-LAYER.....	42
4.1. La conception cross-layer.....	43
4.2. Approches du cross-layer dans les réseaux sans fil.....	43
4.3. Les types d'architecture cross-layer.....	44
4.3.1. Architecture cross-layer à base de communication directe.....	44
4.3.2. Architecture Cross-layer à base de communication indirecte.....	44
4.3.3. Architecture Cross-layer à base de nouvelles abstractions.....	44
4.4. Avantages et inconvénients du Cross-layer [108,109].....	45
5.APPRENTISSAGE DANS LE STREAMING ADAPTATIF.....	45
5.1. Introduction.....	45

5.2. Apprentissage par renforcement.....	46
5.3. Processus de Décision de Markov.....	46
5.4. Résolution d'un PDM dans l'incertain.....	47
5.4.1. La stratégie ϵ -greedy.....	48
5.4.2. La stratégie de Boltzmann.....	49
6.PERFORMANCES DU PROTOCOLE TCP DANS LES RESEAUX SANS FIL AD HOC.....	49
6.1. Introduction.....	49
6.2. Différentes causes de perte de paquets dans les réseaux sans fil.....	50
6.2.1. Problème lié à la puissance du signal.....	50
6.2.2. Problème lié aux interférences.....	50
6.2.3. Autres causes de perte de paquets.....	51
6.3. TCP et le problème de perte de paquets dans les réseaux sans fil (impact et performance)....	52
6.3.1. La congestion et la perte de paquet dans les réseaux sans fil.....	52
6.4. TCP et la qualité de service dans les réseaux sans fil.....	53
6.5. Etude comparative des travaux existants.....	54
6.5.1. Les solutions inter-couches (Cross-Layer).....	54
6.5.2. Les solutions basées sur la couche Transport.....	60
6.6. Conclusion.....	62
7. IMPACT DU PROTOCOLE TCP SUR LES SERVICES VIDEO STREAMING ADAPTATIFS.....	63
7.1. TCP et les services de vidéo streaming adaptatif.....	63
7.2. Protocoles de transport dédiés au streaming.....	63
7.3. Impact des paramètres de TCP sur la qualité de service des services de vidéo streaming adaptatifs.....	65
7.4. L'enjeu de la qualité d'expérience sur les services de vidéo streaming adaptatif.....	65
7.5. Technique d'adaptation du réseau.....	66
7.5.1. Contrôle de débit basé source.....	66
7.5.2. Contrôle de débit basé récepteur.....	71
7.5.3. Contrôle de débit hybride.....	76
8.CONCLUSION ET CRITIQUES.....	78

CHAPITRE III : APPROCHES POUR L'AMELIORATION DE TCP DANS UN ENVIRONNEMENT SANS FIL

1.INTRODUCTION	80
2. CONTRIBUTIONS	81
2.1. Première contribution : Proposition d'un mécanisme inter-couches de différenciation de perte pour TCP dans les réseaux sans fils ad hoc.....	81
2.1.1. Positionnement du problème.....	81
2.1.2. Description de la solution.....	82
2.1.3. Formulation de la solution proposée.....	83
2.1.4. Récupération de la valeur du RSSI minimal et du bruit maximal du chemin.....	84
2.1.5. Organigramme et Algorithme.....	86
2.1.6. Evaluation par simulations.....	87
2.1.7. Conclusion.....	94
2.2. Deuxième contribution : TCP et l'amélioration des services de vidéo streaming adaptatifs...	94
2.2.1. Introduction.....	94
2.2.2. Positionnement du problème.....	95
2.2.3. Description de la solution.....	98
2.2.4. Emulation.....	106
2.2.5. Conclusion.....	113
Conclusion générale et perspectives	114
Références Bibliographiques.	

Liste de figures

Figure1 : Catégories de réseaux sans fil.....	07
Figure 2 : Principales normes de réseaux sans fil.....	08
Figure3 : Interférence de nœud dans une topologie en chaine.....	12
Figure 4 : Réseau ad hoc.....	15
Figure 5 : Extension de couverture par un réseau adhoc.....	17
Figure 6 : Mécanisme de l’acquittement.....	19
Figure7 : Etablissement d’une connexion.....	20
Figure8 : Diagramme d’états simplifié de TCP.....	21
Figure 9 : Démarrage lent de TCP.....	22
Figure 10 : Variation du débit de TCP avec la longueur du chemin.....	29
Figure 11 : Média streaming.....	32
Figure12 : Système général de streaming multimédia.....	34
Figure 13 : Flux direct adaptatif.....	35
Figure 14 : QoS/QoE domaines d'application inspirés de [126].....	39
Figure 15 : QoS/QoE couches dans le modèle OSI.....	40
Figure 16 : Quelques types d’interférence.....	51
Figure 17 : Comparaison des valeurs du débit filaire et sans fils avec déplacement.....	53
Figure 18 : Comparaison des valeurs du débit filaire et sans fils sous l’effet des interférences.....	54
Figure 19 : Classification des solutions pour l’amélioration de TCP dans les réseaux sans fil.....	55
Figure20 : TCP-Split.....	59
Figure 21 : Inter-couches (Cross-Layer) réseau et physique.....	59
Figure 22 : Architecture 3GPP-DASH.....	74
Figure 23 : Structure d’un segment TCP.....	82
Figure 24 : Organigramme de l’architecture proposée.....	86
Figure 25 : Topologie de la simulation.....	89

Figure 26 : Résultats de premier scénario.....	90
Figure 27 : Résultat du deuxième scénario.....	91
Figure 28 : Résultat du troisième scénario.....	92
Figure 29 : Résultats du quatrième scénario.....	93
Figure 30 : Vidéo streaming adaptatif à travers HTTP.....	96
Figure 31 : Services de vidéo streaming adaptatif les plus connus.....	97
Figure 32 : Accès aux services de vidéo streaming adaptatif à travers différents types de technologies avec différents type de terminaux.....	98
Figure 33 : Datagramme d'un service de streaming adaptatif.....	109
Figure 34 : Scénario de l'émulation.....	109
Figure 35 : Consommation du forfait Internet.....	110
Figure 36 : Consommation de la batterie.....	111
Figure 37 : Consommation de CPU.....	111
Figure 38 : Le MOS global des utilisateurs.....	112
Figure 39 : Evolution de la bande passante disponible dans le temps.....	112
Figure 40 : Valeur du MOS (MOS*2) par rapport au temps.....	113
Figure 41 : Valeur du MOS (MOS*2) par rapport au temps.....	1
Figure 42 : Valeur du MOS (MOS*2) par rapport au temps.....	2
Figure 43 : Valeur du MOS (MOS*2) par rapport au temps.....	4

LISTE DES TABLES

Table 1 : Comparaison générale des approches.....	62
Table 2 Comparaison des différentes approches de vidéo streaming.....	78
Table 3 : Valeur de FR (facteur de résolution).....	99
Table 4 : Valeur de FS (facteurs d'écran).....	99
Table 5 : Valeur de BC (facteur de batterie).....	99
Table 6 : Les valeurs facteur utilisateur avec les qualités vidéo.....	100
Table 7 : Les qualités vidéo avec les débits minimaux.....	101
Table 8 : Pourcentage de récompense par rapport au MOS.....	104
Table 9 : Paramètres des terminaux utilisés dans l'émulation.....	107
Table 10 : Paramètres de l'émulation.....	109

LISTE DES ACRONYMES

Wpan : Wireless Personal Area Network	DCCP : Datagram Congestion Control Protocol
Wlan : Wireless Local Area Network	SCTP : Stream Control Transmission Protocol
Wman : Wireless Metropolitan Area Network	RTO : Retransmit Time Out
Wran : Les réseaux régionaux	RTT : Round trip time
Wwan : Wireless Wide Area Network	RTTVAR : Variation du RTT
BLR : Boucle locale radio	Syn : Paquet de synchronisation
WIMAX : Worldwide Interoperability for Microwave Access	ACK : accusé de réception
GSM : Global System for Mobile Communication	CWND : congestion window
GPRS : General Packet Radio Service	HTTP : Hypertext Transfer Protocol
DVB-S : Digital Video Broadcasting-Satellite	AIMD : Additive Increase and Multiplicative Decrease
WIFI : Wireless Fidelity	DUPACK : ACK dupliqué
UWB : Ultra-Wide Band	SACK : Selective Acknowledgment
OFDM : Orthogonal Frequency Division Multiplex	MPEG : <i>Moving Picture Experts Group</i>
MAC : Media Access Control	QoS : Qualité de service
AP : Access point	QoE : Qualité d'expérience
IP : Internet protocol	IPTD : IP Packet Transfer Delay
TCP : Transport control protocol	IPDV : IP packet delay variation
UDP : User Datagram Protocol	IPLR : IP Loss Ratio
	IPER : IP Packet Error Rate

IPPT : IP Packet Throughput

MOS : **Mean opinion score**

OSI : Open Systems Interconnection

RCSF : Réseaux de Capteurs Sans Fils

PDM : processus de décision de Markov

RSSI : Puissance du signal reçu

DCF : Distributed Coordination Function

MANET : Mobile AD-HOC network

ICMP : Internet Control Message Protocol

RRN : Notification de la restauration de la route

RFN : Notification d'erreur de route

BER : Erreurs de bits

ECN : Explicit Congestion Notification

NDER : Notification-de-Déconnection-Explicite-de-Route

NP : Nœud-Pivotant

RL : Requête localisée

NRER : Notification-Réussie-Explicite-de-Route

PRR : Phase de la reconstruction de Route

LACK : Acquiescement local

COPAS : Contention-based Path Selection

DSR : Un protocole de routage

OOO : Livraison Out-Of-Order

TPSN : TCP Packet-Sequence-Number

ADSN : ACK de duplication de numéro de séquence

RTP : Le protocole de transmission temps réel

RTSP : Real Time Streaming Protocol

NAT : *Network address translation*

RTCP : Real-time Transport Control Protocol

MIMD : l'augmentation multiplicative et la diminution multiplicative

MTP : Multimedia transport protocol

VTP : Protocole de transport vidéo

LDA : L'algorithme de discrimination de perte

AR : le taux d'estimation atteint

SAMM : Source-Adaptive Multilayered Multicast Algorithms

TFRC : Contrôle de débit avec TCP Friendly

RLM : Receiver-driven Layered Multicast

LVMR : Layered Video Multicast Retransmission

3G-DASH : Dynamic Adaptive Streaming over HTTP

PVR : Réseau d'enregistrement vidéo personnel

CDN : *Content delivery network*

AEC : Approche de l'environnement contrôlé

RNN : Réseau de neurone aléatoire

GOP : Un groupe d'images successives

SSM : La puissance du signal minimale de la route

NMX : La puissance du bruit maximal de la route

PNMX : La puissance du bruit maximal de la route prédite

PSSM : La puissance du signal minimale de la route prédite

NS3 : Network simulator

AODV : Ad hoc On Demand Distance Vector (protocole de routage)

CSMA/CA : Carrier Sense Multiple Access with Collision Avoidance

UF : le facteur d'utilisateur

TR : la résolution du terminal

FR : le facteur de résolution

FB : Le facteur de la batterie

BC : La batterie disponible

FS : Le facteur de la taille d'écran

SS : La taille de l'écran

MDP : un processus de décision markovien

BW : La bande passante

CAS : L'approche YOUTUBE classique

UAS : L'approche proposée

Publications et conférences

- 1) Yassine DOUGA, Malika BOURENANE, Yassine HADJADJ-AOUL, Abdelhamid MELLOUK: « TCP based-user-parameters control for adaptive video streaming». Dans le journal international MULTIMEDIA TOOLS AND APPLICATIONS Springer journal, ISSN: 1380-7501, DOI 10.1007/s11042-015-2857-1,(aout 2015).
- 2) Yassine DOUGA, Malika BOURENANE: « Simulation of an Adaptive video streaming based-user control using TCP parameters ». Dans la Revue Méditerranéenne des Télécommunications, Vol. 5, No 2 (2015).
- 3) Yassine DOUGA, Malika BOURENANE: « New adaptation method of TCP for mobile ad hoc networks ». Dans le proceeding de IEEE WCCAIS 2014, Computer Applications and Information Systems World Congress, Tunisie, (janvier 2014).
- 4) Yassine DOUGA, Malika BOURENANE: « Improve Performance of TCP over Mobile AdHoc Network using a cross-layer solution». Dans le journal IJSER, International Journal of Scientific & Engineering Research, Volume 5, Issue 7, ISSN 2229-5518, , (July-2014).
- 5) Yassine DOUGA, Malika BOURENANE, Abdelhamid MELLOUK: « Adaptive Video Streaming Using TCP Factors Control with User Parameters ». Dans le proceeding de Elsevier FNC'14, The 9th International Conference on Future Networks and Communication, Volume 34, 2014, Pages 526–531, Niagara Falls, Canada, (aout 2014).
- 6) Yassine DOUGA, Malika BOURENANE: « A cross layer solution to improve TCP performances in Ad Hoc wireless networks». Dans le proceeding de IEEE SaCoNeT 2013, International Conference of Smart Communications in Network Technologies France, Paris, (juin 2013).

INTRODUCTION GENERALE

Introduction générale

1. Contexte et problématique de la thèse

De nos jours, certains protocoles supportant l'Internet sont les mêmes qu'à l'origine, ils n'ont presque pas changé [78]. Suivant une structure en couches abstraites, fournissant chacune une fonctionnalité bien définie [79], la pile protocolaire TCP/IP a réussi à bien s'adapter aux évolutions des technologies sous-jacentes. Cependant ces nouvelles technologies ont graduellement étendu le contexte dans lequel ces protocoles doivent fonctionner.

La réduction en taille et le prix des équipements informatiques a largement facilité l'apparition de l'informatique mobile et l'accès simultané à plusieurs réseaux Figure 41. Ces équipements informatiques tels que smartphones, tablettes, ordinateurs portables ...etc, sont transportés au rythme des déplacements de leur utilisateur. Plus récemment, ces terminaux se sont aussi vus pourvoir de plus d'interfaces réseau supplémentaires Figure 42. Ainsi, ils peuvent s'adapter aux mouvements de leur porteur, établir une ou plusieurs connexions à de multiples réseaux afin d'être « toujours connectés au mieux » [80].

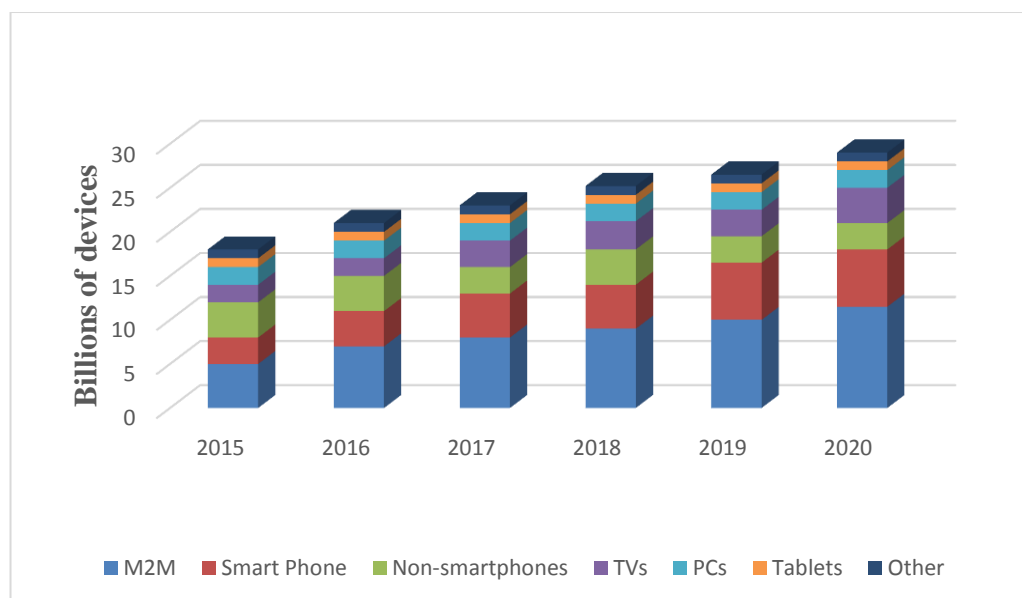


Figure 41 Différents types de terminaux qui accèdent de nos jours à Internet

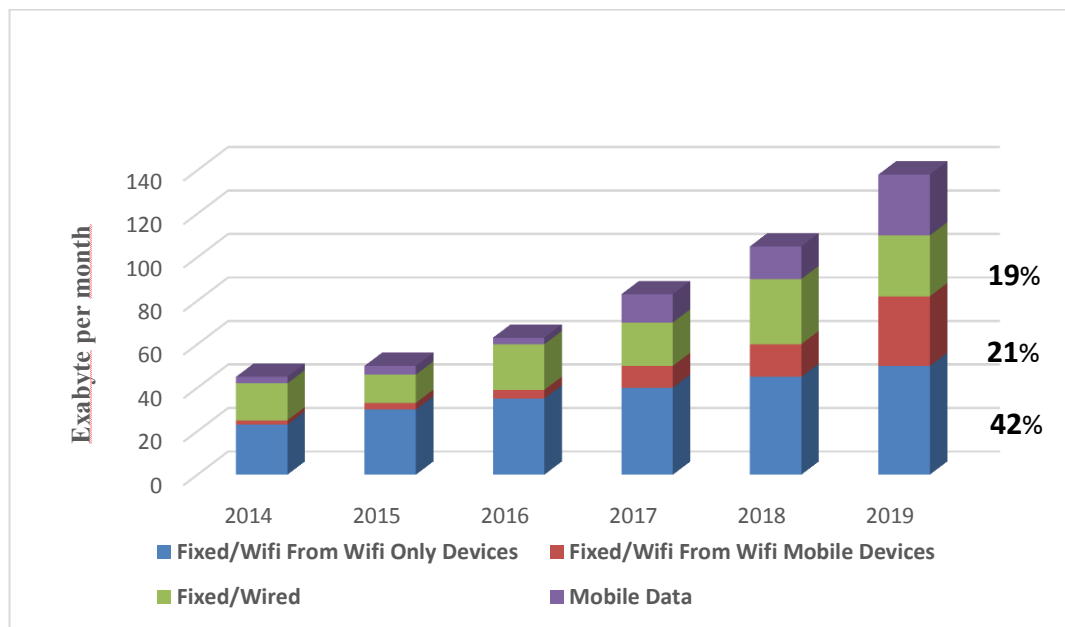


Figure 42 Classification du flux de données sur Internet par technologie d'accès

L'environnement sans fil fondé sur l'utilisation des liaisons radios est caractérisé par deux modes de communication, à savoir le mode avec infrastructure où la communication entre deux points ou nœuds quelconques du réseau passe nécessairement par une station de base. Ainsi que les réseaux mobiles sans infrastructure (appelés réseaux ad hoc) où chaque point ou nœud du réseau peut communiquer directement avec ses voisins, sans station de base intermédiaire.

Les réseaux ad hoc sont une catégorie de réseaux sans fil caractérisés entre autres par une topologie dynamique [16], une bande passante limitée, des contraintes de consommation d'énergie. Le lien sans fil a la caractéristique essentielle de varier en fonction du temps et de l'espace. Il est aussi caractérisé par une mémoire à courte échelle due au multi trajets pouvant causer une rafale d'erreurs qui occasionne l'impossibilité de transmettre correctement les paquets sur le lien. Le changement de l'état du lien sans fil de "bon" à "mauvais" et vice-versa peut intervenir de façon asynchrone pendant des laps de temps très courts. En plus de la variation du canal à petite échelle, il y a une variation à grande échelle qui implique le conditionnement de l'état moyen du canal par la position de l'utilisateur et le niveau des interférences possibles.

Ainsi, la récente prévalence des réseaux d'accès sans-fil en général et des réseaux ad hoc en particulier présente de nouveaux problèmes à des protocoles dont les hypothèses de conception sont basées sur l'usage d'un média fiable. Par exemple le protocole de transport TCP classique, souffre d'une mauvaise interprétation des pertes de paquets dues aux conditions du medium radio et non à des congestions sur le réseau. Il ne peut cependant

pas distinguer la cause de ces pertes en raison de la séparation en couches. Cette erreur entraîne des baisses de performances lorsque des réseaux sans fil sont traversés [83].

Le passage des systèmes reposant sur les réseaux filaires vers les environnements sans fil et la différence notoire qui existe entre ces deux environnements ont conduit à l'émergence des systèmes multi-niveaux ou Cross-Layer. Ces systèmes visent à répondre aux besoins de l'amélioration des performances qui s'imposent au bénéfice des systèmes sans fil dont par exemple les réseaux ad hoc. Le concept de « Cross-Layer » introduit une technique d'adaptation des protocoles au contexte sans fil à travers le partage d'information entre les couches. Cette technique permet d'obtenir des gains de performance divers comme démontré dans plusieurs études [19] [20][21] [23] [24] [25] [26] [27], ce qui justifie pleinement l'importance des modèles réseaux « Cross-Layer ».

La technique « Cross-Layer » s'applique à tous les protocoles de divers niveaux du modèle en couches, tant qu'il existe des interactions pour lesquelles les performances globales du système peuvent être améliorées. Un exemple simple relatif au multimédia illustre l'importance de cette conception novatrice. La gestion de ressources, l'adaptation et les stratégies de protection disponibles dans les couches inférieures de la pile du modèle OSI (PHY, MAC, Réseau, Transport) sont optimisées sans considérer explicitement les caractéristiques spécifiques des applications multimédias par exemple, les algorithmes de flux et de compression ne considèrent pas les mécanismes fournis par les couches inférieures pour la protection d'erreur, l'ordonnancement, la gestion de ressource, ainsi de suite.

De nos jours, la plupart des applications sur internet utilisent le protocole http. Le protocole http est un protocole de couche application, il utilise le protocole TCP comme protocole de transport de données. Parmi les applications les plus utilisés et répondu sur internet on trouve les applications de vidéo streaming adaptative basé http figure 43.

Les applications multimédia interactives fournissent des environnements de communication très complets pouvant être utilisés dans le cadre du travail collaboratif synchrone ou encore dans un contexte ludique et familial. Cela va de la communication audio et/ou vidéo jusqu'aux applications partagées...etc. Ces applications, par leur aspect interactif, requièrent un service de qualité du système de communication sous-jacent afin de fonctionner correctement. Au niveau réseau, ce service, varie selon les applications et s'exprime par exemple en termes de garanties sur le délai de bout en bout, le débit de transmission ou encore le taux de perte d'information.

Jusqu'à-là, les applications multimédia ont été largement déployées et supportées sur les infrastructures filaires telles que la fibre optique comme lien en cœur de réseau, ou encore la technologie xDSL (Digital Subscriber Line) pour la distribution finale. De nos jours, le large déploiement et les avantages que les réseaux sans fil présentent a fait que la plupart des utilisateurs des applications multimédia accèdent à ce genre d'application via réseau sans fil. En déployant les applications multimédia interactives sur de tels réseaux,

de nombreux problèmes apparaissent, tel que la dégradation de la qualité de la vidéo reçue, les interruptions de la vidéo, la réception d'une qualité de vidéo qui n'est pas adéquate avec les exigences des utilisateurs ou avec les conditions du réseau.

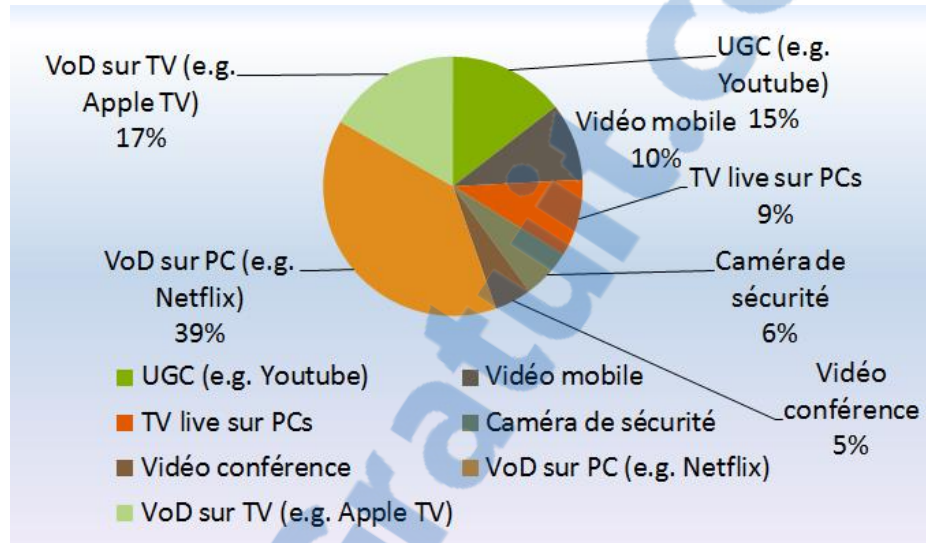


Figure 43 Portion de téléchargement des données de vidéo streaming sur Internet par services

2. Contribution et structure de la thèse

Dans cette thèse, nous proposons une nouvelle architecture inter-couche du protocole TCP. Cette architecture a pour but d'améliorer les performances du protocole TCP dans les réseaux sans fil en général et dans les réseaux ad hoc en particulier. L'idée de cette approche est de faire une interaction entre les deux couches du modèle OSI. Ces deux couches sont, la couche physique et la couche transport. Cette interaction va nous permettre d'améliorer le mécanisme de perte de paquet et de contrôle de congestion du protocole TCP en se basant sur deux grandeurs de la couche physique qui sont la puissance de signal et le bruit. A l'aide de ces deux valeurs, la nouvelle architecture de TCP saura faire la distinction entre les pertes de paquets dû à la congestion et ceux qui sont dû à l'environnement sans fil. D'après les résultats obtenus ceci va garantir un bon fonctionnement du protocole TCP dans les réseaux sans fil sans perdre en performance. Cette architecture a été testée sous NS3.

Suite aux travaux sur notre nouvelle architecture et afin de mieux la perfectionner, nous avons opté pour une nouvelle adaptation qui vise à améliorer la qualité de service (QoS) du réseau sans fil et la qualité d'expérience QoE des utilisateurs pour des applications de vidéo streaming en utilisant les paramètres de la nouvelle architecture du protocole TCP qu'on a développé auparavant. Cette amélioration vise à adapter les

paramètres du réseau qui dépendent directement du protocole TCP comme la bande passante, la gigue, le délai, le taux de pertes, avec les différentes caractéristiques du terminal de l'utilisateur et de son feedback. Les tests ont été fait avec des émulations, les résultats sont représentés sous forme de MOS (Mean Opinion Score) ou le ressenti global des utilisateurs.

Le reste de la thèse est structuré comme suite

Le premier chapitre, fourni des définitions et des notions de base sur tout ce qui est en relation avec les réseaux sans fil, le protocole de transport TCP et les services de vidéo streaming adaptatifs, afin de servir de point de repère pour les lecteurs dans ce qui suit dans les chapitres suivants.

Le deuxième chapitre, l'état de l'art, décrit, compare et critique les travaux existants des deux approches. Le but de ce chapitre est de pointer sur les limites des travaux précédents et leurs inconvénients.

Dans le troisième chapitre, nous décrivons avec détails les deux solutions proposées, les formules et les méthodes utilisées, en plus des algorithmes explicatifs. Ensuite, nous introduirons l'environnement et les conditions de test (simulation et émulation) des deux approches, les résultats obtenus et les critiques qui se rapportent aux résultats.

Enfin nous terminons par une conclusion qui situe nos travaux et qui fixe nos perspectives.

CHAPITRE I : CONCEPTS DE BASE

CHAPITRE I**CONCEPTS DE BASE**

Sommaire

1. Généralités sur les réseaux sans fil.....	6
2. Protocole de transport.....	17
3. Fonctions de contrôle de congestion.....	21
4. Variantes de TCP	25
5. TCP et les réseaux sans fil	27
6. Le protocole http.....	30
7. Les services de vidéo streaming.....	32
8. Les différentes technologies de streaming.....	34
9. Conclusion.....	36

1. Généralités sur les réseaux sans fil**1.1. Introduction**

Les réseaux sans fil sont en plein développement. Conduit par l'émergence des appareils de communication tels que les téléphones cellulaires, les ordinateurs portables et par la flexibilité de leur interface, ce qui permet à un utilisateur de changer de place tout en restant connecté. Ainsi l'utilisateur transite d'un environnement restreint vers un environnement ubiquitaire où les traitements se font à travers plusieurs infrastructures.

1.2. Définition

Les réseaux sans fil sont des réseaux informatiques ou numériques dont les nœuds sont connectés par radio. La mobilité et la flexibilité comptent parmi les principaux avantages des réseaux sans fil. En effet, un utilisateur peut se connecter à un réseau existant et se déplacer librement tant qu'il reste sous la couverture radio du réseau. La flexibilité se traduit par la rapidité de déploiement. Il existe différents types de réseaux

sans fil, tels que les réseaux de téléphonie mobile, les réseaux ad hoc et les réseaux de capteurs.

1.3. Les différentes technologies des réseaux sans fil

Aujourd'hui, divers équipements sont accessibles aux entreprises et au grand public permettant des connexions entre appareils sans fil. Plusieurs communautés collaborent afin de standardiser les technologies de ces réseaux, tantôt concurrentes, tantôt complémentaires. La figure 1 montre une classification des différentes technologies selon leur portée.

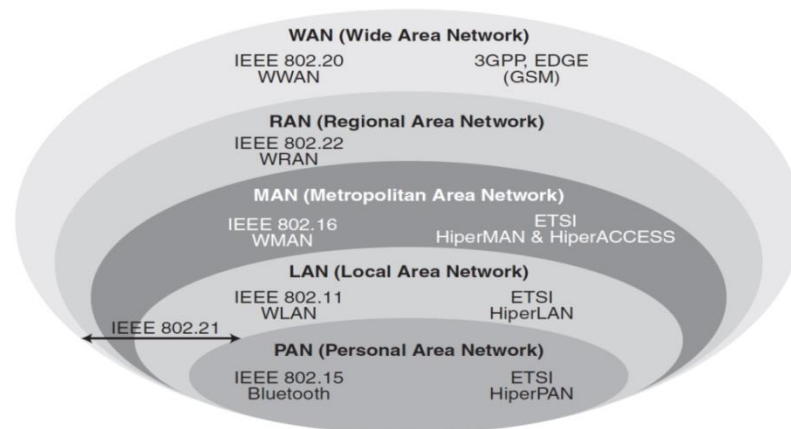


Figure 1 : Catégories de réseaux sans fil [84]

Où

- **WPAN** : Les réseaux sans fil personnels (Wireless Personal Area Network), sont des réseaux sans fil à très faible portée (de l'ordre d'une dizaine de mètres). Ils concernent l'environnement immédiat d'une personne. La principale technologie pour mettre en œuvre de tels réseaux est le Bluetooth (proposée par Ericsson en 1994) pour une portée maximale d'une trentaine de mètres, elle fournit un taux de transmission radio théorique de 1 Mbit/s.

- **WLAN** : Les réseaux locaux sans fil (Wireless Local Area Network) concernent un environnement plus étendu que les réseaux personnels comme une maison, une entreprise ou un campus. Ils sont principalement basés sur la technologie IEEE 802.11 ou sur les technologies HiperLan 1 et Hiperlan 2 qui offre des taux de transmission radio théoriques pouvant atteindre 54 Mbit/s. Ces technologies ont une portée plus importante : 300 mètres en extérieur et 100 mètres à l'intérieur des bâtiments pour IEEE 802.11b et d'une centaine de mètres pour Hiperlan 2.

- **WMAN** : Les réseaux métropolitains sans fil (Wireless Metropolitan Area Network) sont connus sous le nom de boucle locale radio (BLR). Ils sont basés sur la

norme IEEE 802.16 et visent à couvrir une région étendue comme une ville (plusieurs kilomètres). La norme la plus connue est le WiMAX, permettant d'obtenir un taux de transmission radio théorique de l'ordre de 74Mbit/s sur un rayon de dizaines de kilomètres.

- WRAN : Les réseaux régionaux sont étudiés par l'IEEE 802.22. Le rayon de la cellule peut atteindre 50 kilomètres pour les gammes de fréquences en dessous de 1 GHz. La distance potentielle du terminal étant importante, le débit montant est assez limité. En revanche, sur la bande descendante, 4 Mbit/s sont disponibles. L'application de base est la télévision interactive ou les jeux vidéo interactifs.

- WWAN : Les réseaux sans fil étendus (Wireless Wide Area Network) tirent profit de l'infrastructure réseau cellulaire pour fournir à l'utilisateur une connectivité réseau même en déplacement. Les WWANs regroupent notamment les différents réseaux téléphoniques de première et deuxième génération mais également les réseaux satellitaires. Parmi les technologies des réseaux cellulaires téléphoniques : GSM (Global System for Mobile Communication) et GPRS (General Packet Radio Service). Les réseaux satellites s'appuient sur des normes comme DVB-S (Digital Video Broadcasting-Satellite).

1.4. Différents normes de réseaux sans fil

Les principales normes sont IEEE 802.15, pour les petits réseaux personnels d'une dizaine de mètres de portée, IEEE 802.11, ou Wi-Fi, pour les réseaux WLAN, IEEE 802.16, pour les réseaux WMAN atteignant plus de dix kilomètres, IEEE 802.22 pour les WRAN (Wireless Regional Area Network), et IEEE 802.20, pour les réseaux WWAN qui correspondent aux solutions cellulaires permettant de recouvrir un pays.

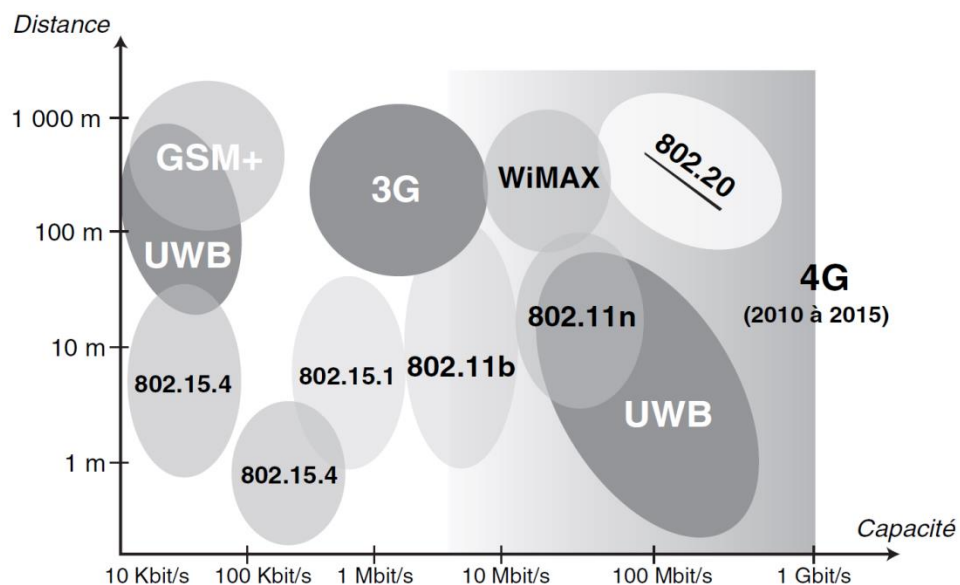


Figure 2 : Principales normes de réseaux sans fil

Dans le groupe IEEE 802.15, trois sous-groupes normalisent des gammes de produits en parallèle :

- IEEE 802.15.1, le plus connu, prend en charge la norme Bluetooth, aujourd'hui largement commercialisée. La version 3.0 utilise l'interface radio décrite dans IEEE 802.15.3, ce qui procure à Bluetooth une nouvelle jeunesse, avec un débit de 480 Mbit/s.

- IEEE 802.15.3 définit la norme UWB (Ultra-Wide Band), qui met en œuvre une technologie très spéciale, caractérisée par une puissance d'émission extrêmement faible, sous le bruit ambiant, mais sur pratiquement l'ensemble du spectre radio (entre 3,1 et 10,6 GHz). Le débit est de 480 Mbit/s sur une portée de 3 m et décroît à environ 120 Mbit/s à une dizaine de mètres.

- IEEE 802.15.4 s'occupe de la norme ZigBee, qui a pour objectif de promouvoir une puce offrant un débit relativement faible mais à un coût très bas. ZigBee est avant tout normalisé pour le passage des commandes plutôt que des données.

- 802.11 est une norme établie par l'IEEE. Elle décrit les couches physiques et MAC d'interfaces Réseau radio et infra-rouge. Les débits possibles varient entre 1 et 54 Mbit/s suivant les techniques et les éventuelles extensions de la norme employées. Les portées prévues sont variantes entre quelques dizaines et quelques centaines de mètres en fonction de la vitesse choisie et de l'environnement. Cette norme cible deux contextes d'utilisation, le mode infrastructure et le mode appelé ad hoc. La norme originelle de 802.11 date de 1997 et décrit les couches physiques et MAC pour un débit allant jusqu'à 2 Mbit/s en radio, dans la bande dès 900 MHz. Des extensions ont été publiées depuis qui viennent lui ajouter des améliorations et des modes de fonctionnement plus performants. Les principales extensions sont les suivants :

- 802.11b ajoute la description d'une couche physique améliorée proposant des débits de 5.5 et 11 Mbit/s en plus de ceux déjà supportés.

- 802.11a ajoute des modes encore plus rapides (jusqu'à 54 Mbit/s) en travaillant dans la bande dès 5 GHz, mais en utilisant des techniques OFDM d'accès au canal.

- 802.11g utilise des techniques OFDM similaires à la 802.11a, mais en restant dans la bande ISM à 2.4 GHz. Les débits possibles atteignent également les 54 Mbit/s tout en gardant la compatibilité avec les équipements 802.11b.

- 802.11n augmente la vitesse des réseaux sans fil local (WLAN), elle améliore la fiabilité et elle étend la zone de couverture sans fil.

- 802.11e cherche à améliorer 802.11 de façon à pouvoir donner des garanties de qualité de service.

1.5. Spécificités des réseaux sans fil

L'intégration des technologies sans fil avec les protocoles classiques des réseaux fixes ne va pas sans poser des difficultés notamment au niveau de la couche Transport.

Les réseaux sans fil arborent des spécificités qui les caractérisent par rapport aux réseaux filaires. Ils sont basés sur une liaison utilisant des ondes radioélectriques (radio et infrarouges) au lieu des câbles habituels. Par nature, les communications par ce type de liaison entraînent un certain nombre de problèmes n'ayant pas d'équivalent dans le monde filaire.

La propagation du signal radio est un élément clé de la qualité d'une transmission mais le nombre de fréquences et de canaux disponibles pour la communication est limité. Ainsi, lorsque deux nœuds émettent simultanément sur des bandes de fréquence ou des canaux proches, des interférences (bruit) seront ressenties sur les communications. De plus, la distance et l'environnement en général, affectent la qualité d'une transmission radio en perturbant le signal. Celui-ci s'atténue en fonction de la distance mais aussi en fonction des contraintes de l'environnement sans fil.

Les spécificités des réseaux sans fil représentent des sources d'imperfection qui contribuent à déformer le signal jusqu'à le rendre inutilisable. Les sections suivantes s'intéressent aux spécificités des réseaux sans fil ayant des répercussions sur le protocole de transport TCP.

1.5.1. Spécificité physique

Le principal problème qui se manifeste au niveau physique concerne la dégradation du signal due à certaines causes. En effet, sur un support sans fil, les perturbations issues des interférences, de l'environnement et du mouvement des nœuds viennent dégrader le signal original.

Aussi, les caractéristiques du support de communication tel que débit, délai ou taux de perte dépendent fortement de l'environnement radio et sont sujettes aux interférences et aux bruits. Le taux d'erreur de niveau physique peut aller au-delà de trois fois celui d'un support filaire [94].

1.5.2. Erreur du canal

Les erreurs de bit par rafales peuvent corrompre les paquets en transmission, conduisant ainsi à la perte des paquets de données TCP ou des accusés de réception (ACKs). Si l'émetteur TCP ne peut pas recevoir l'accusé de réception dans le délai de retransmission (RTO), il réduit immédiatement sa fenêtre de congestion, régresse sa retransmission de façon exponentielle, et retransmet le paquet perdu. Les erreurs de canal

intermittentes peuvent donc provoquer la réduction de la taille de la fenêtre de congestion de l'émetteur, ce qui entraîne un faible débit.

1.5.3. Contention du Médium et collision

Les protocoles d'accès au canal (de niveau 2 dans le modèle OSI et l'équivalent de la couche liaison dans le modèle TCP/IP) décrivent l'ensemble des méthodes permettant de coordonner les accès au support partagé entre plusieurs stations. Plus spécifiquement, la couche MAC décrit les règles permettant à une station de transmettre et d'écouter. Ces règles sont écrites de façon à remplir un ensemble de services tels que l'équité (dans l'accès au canal ou dans les ressources utilisées), la fiabilité, le passage à l'échelle, la qualité de service et ce en maximisant les débits. Ces protocoles sont critiques dans un environnement où les stations ne peuvent pas recevoir en même temps qu'elles émettent et dans lequel le support est partagé et en diffusion par nature.

Les mécanismes de contrôle d'accès au support basés sur la contention, tels que le protocole MAC IEEE 802.11 [114], ont été largement étudiés et incorporés dans de nombreux bancs d'essai pour réseaux sans fil multi-sauts ad hoc, où les nœuds voisins se disputent le canal sans fil partagé avant de transmettre. Ces protocoles se sont avérés affecter significativement les performances de TCP [115, 116, 117]. Lorsque TCP fonctionne sur MAC 802.11, comme il a été souligné dans [118], le problème de l'instabilité devient très dramatique. Il est montré que les collisions et le problème de la station exposée sont deux raisons majeures qui peuvent empêcher un nœud d'atteindre un autre lorsque les deux nœuds sont à portée de transmission.

Si un nœud ne peut pas atteindre son nœud adjacent après plusieurs tentatives, il déclenche alors une défaillance de la route, ce qui, à son tour provoquera le démarrage de la découverte de route par le nœud source. Aucun paquet de données n'est envoyé avant qu'une nouvelle route ne soit trouvée. Pendant ce processus, TCP invoque les algorithmes de contrôle de congestion s'il observe un délai d'expiration. Ainsi, de sérieuses oscillations dans le débit TCP seront donc observées. En outre, le mécanisme du backoff BEB (Binary Exponential Backoff) utilisé dans la couche MAC exacerbe cette situation [116]. En effet, un paquet de données volumineux qui occupe le canal partagé, diminue la chance d'accès au médium pour un nœud intermédiaire. Ce dernier marque une attente d'une période de temps aléatoire et essaye à nouveau. Après plusieurs essais négatifs, un échec de route est signalé.

TCP peut également rencontrer de sérieux problèmes d'iniquité [115, 116, 118] pour les raisons indiquées ci-dessous:

- La topologie peut provoquer une iniquité en raison de l'inégalité des chances d'accès au canal pour les différents nœuds. En effet, comme indiquée dans la figure 3 où le cercle en bleu dénote une portée de transmission valide d'un nœud et le cercle en rouge

une portée d'interférence d'un nœud, les nœuds dans cette topologie en chaîne de 7 nœuds subissent différents degrés de compétitions. Soient deux flux TCP, dont TCP flow 1 est envoyé du nœud 0 vers le nœud 1 et TCP flow 2 du nœud 6 vers le nœud 2. La transmission du flux 1 subit l'interférence de 3 nœuds (les nœuds 1, 2 et 3) alors que la transmission du nœud 3 vers le nœud 2 subit l'interférence de 5 nœuds (les nœuds 0, 1, 2, 4 et 5). Ainsi, le flux 1 obtient un débit beaucoup plus élevé que le flux 2 en raison de l'inégalité des chances d'accès au canal.

- Le mécanisme du backoff peut conduire à une situation d'iniquité car il favorise toujours le dernier nœud qui a émis avec succès.

- La longueur des flux TCP influence l'iniquité. En effet, Des flux plus grands conduisent à un temps aller-retour (RTT) plus long et une probabilité de rejet des paquets plus grande entraînant ainsi une baisse et une fluctuation du débit TCP de bout en bout. L'iniquité est amplifiée à travers cette réaction en chaîne. Ainsi, un haut débit sera plus élevé alors qu'un débit faible subira encore une baisse.

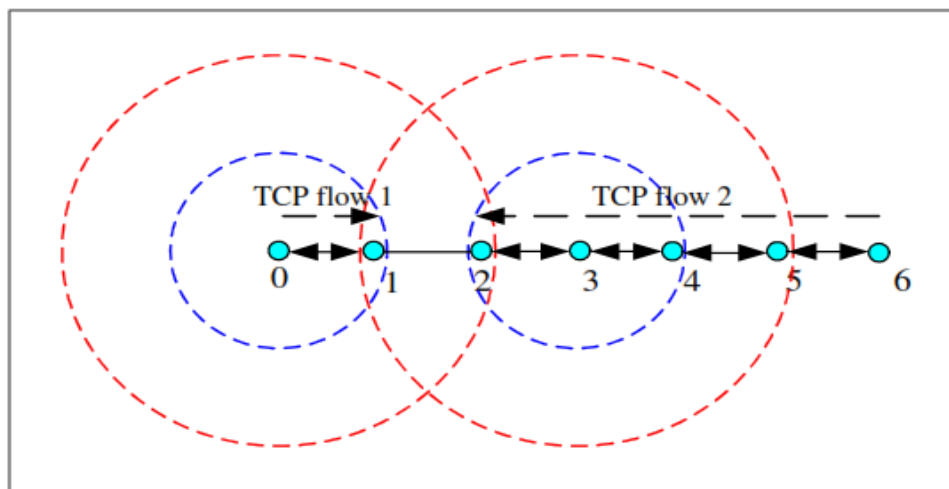


Figure 3 : Interférence de nœud dans une topologie en chaîne

1.5.4. Mobilité

La mobilité peut provoquer la rupture de lien et l'échec de l'itinéraire entre deux nœuds voisins, ce qui peut entraîner alternativement des pertes de paquets. Comme TCP ne distingue pas entre les pertes de paquets causées par les défaillances de routes et celles en raison d'une congestion, il déclenche les mécanismes de contrôle de congestion qui réagissent négativement. En effet, si la découverte d'un nouvel itinéraire prend plus de temps qu'un RTO, l'émetteur TCP invoque un contrôle de congestion et le débit déjà

réduit en raison des pertes va encore diminuer. Il pourrait être encore pire lorsque la source et la destination d'une connexion TCP se situent dans différentes partitions du réseau. Dans un tel cas, plusieurs expirations RTO consécutives conduisent à une inactivité durant une ou deux minutes, même si l'expéditeur et le récepteur se reconnectent finalement.

Les auteurs Fu et al. ont effectué des simulations en tenant compte de la mobilité, de l'erreur du canal, et de la contention du média partagé [116]. Ils ont indiqué que les déconnexions et reconnexions réseau induites par la mobilité ont un impact plus significatif sur la performance de TCP comparées aux erreurs du canal et aux contentions du média partagé. Lorsque la mobilité augmente, par rapport à un TCP de référence, TCP New Reno souffre d'une chute relative de débit allant de presque 0% dans un cas statique à 90% dans un cas fortement mobile (lorsque la vitesse de déplacement est de 20m/s). En revanche, une congestion et une légère erreur de canal (soit 1%) ont un effet moins visible sur TCP (avec moins de 10% de baisse de performance par rapport au TCP de référence).

1.5.5. Spécificité du routage

Le routage est un élément essentiel pour la communication entre les nœuds éloignés. Cependant, la mobilité et la dynamique de la topologie rendent sa réalisation complexe. En effet, cette mobilité va augmenter les problèmes de propagation du signal dans la mesure où un nœud mobile va voir évoluer les conditions de propagation du milieu qui l'entoure en fonction de sa position. Le signal s'atténue donc avec l'éloignement et les nœuds ne peuvent communiquer qu'avec leurs voisins proches.

Les routes sont de courte durée en raison des ruptures fréquentes de liens. Pour réduire les retards dus à la découverte d'une nouvelle route, certains protocoles de routage tels que l'algorithme de routage temporellement ordonné (Temporally-Ordered Routing Algorithm : TORA) [119, 120] maintiennent plusieurs itinéraires entre une paire émetteur-récepteur et utilisent un routage multi-chemin pour transmettre les paquets. Dans un tel cas, les paquets provenant de chemins différents peuvent ne pas arriver dans l'ordre au niveau du récepteur. Ignorant le routage multi-chemin, le récepteur TCP interprétera cette situation comme une congestion. Le récepteur va donc générer des ACKs dupliqués qui amènent l'expéditeur à invoquer des algorithmes de contrôle de congestion tels que la retransmission rapide ou fast retransmission (dès la réception de 3 ACK dupliqués).

1.5.6. Congestion

Il est connu que TCP est un protocole de la couche transport agressif. Ses tentatives d'utiliser pleinement la bande passante entraînent les réseaux ad hoc facilement vers la congestion. En outre, en raison de nombreux facteurs tels que le changement de route et le délai variable imprévisible de MAC, la relation entre la taille de la fenêtre de congestion et

le débit de données tolérable pour une route n'est plus maintenue dans les réseaux ad hoc. La taille de la fenêtre de congestion calculée pour l'ancienne route peut être trop grande pour la route nouvellement trouvée, entraînant ainsi une congestion du réseau si l'émetteur transmet toujours à taux plein, autorisé par l'ancienne taille de la fenêtre de congestion.

La congestion et la surcharge peuvent donner lieu à un débordement du buffer et à une augmentation de contention de lien, ce qui dégrade les performances de TCP.

1.5.7. Considérations énergétiques

Malgré l'essor dans l'équipement à connexion sans fil de plus en plus miniaturisé, la capacité finie de la batterie est toujours l'une des plus grandes limitations de ces appareils. Dans les réseaux sans fil, la consommation de l'énergie est observée au niveau communication, dans l'envoi et le contrôle des données et au niveau traitement qui concerne les aspects de traitement de protocole.

Comme la puissance des nœuds mobiles est limitée, tout système efficace doit être conçu pour être économe en énergie. Dans certains scénarios où la recharge de la batterie n'est pas autorisée, l'efficacité énergétique est critique pour prolonger la durée de vie du réseau. La recherche sur TCP dans les réseaux ad hoc a besoin de trouver un équilibre entre la consommation d'énergie et une session à haut débit. Pour contourner cette limitation énergétique, des mesures d'optimisation doivent être prises au niveau matériel et logiciel.

Dans la suite, nous allons mettre l'accent sur quelques éléments essentiels caractérisant les réseaux locaux sans fil en particulier les réseaux ad hoc mobiles.

Dans ces réseaux, les communications entre équipements terminaux peuvent s'effectuer par le biais de stations de base ou directement. Dans le premier mode, appelé mode infrastructure, le réseau s'organise autour d'une topologie en étoile où le point central appelé point d'accès (AP : Access Point) coordonne l'utilisation du support. Le point d'accès joue souvent le rôle de passerelle vers un réseau filaire, voire vers Internet. Dans le second mode d'utilisation appelé ad hoc, les nœuds ne s'appuient plus sur un nœud central pour communiquer. Toutes communications entre ces nœuds se fait directement sans intermédiaire. L'intérêt de ce mode réside dans la possibilité de communiquer en l'absence d'infrastructure.

1.6. Les réseaux sans fils ad hoc mobiles

Une autre grande catégorie de réseaux sans fil est constituée par les réseaux ad hoc, dans lesquels l'infrastructure n'est composée que des stations elles-mêmes. Ces dernières acceptent de jouer le rôle de routeur pour permettre le passage de l'information d'un

terminal vers un autre, sans que ces terminaux soient reliés directement. Les réseaux ad hoc peuvent être déployés dans tout environnement difficile où le déploiement d'une infrastructure réseau filaire est très contraignant, soit parce qu'il est difficile à mettre en place, soit parce que l'existence du réseau ne dure pas assez longtemps pour justifier une installation par câblage. De tels réseaux, dépourvus de toute infrastructure, trouvent de nombreux champs d'applications : lors d'une campagne militaire, d'une situation de catastrophe (tremblement de terre, désastre naturel, etc), d'une urgence, d'une salle de conférence, ou encore dans le cadre des réseaux de senseurs/capteurs (*sensor networks*). La figure 4 illustre la notion de réseau ad-hoc:

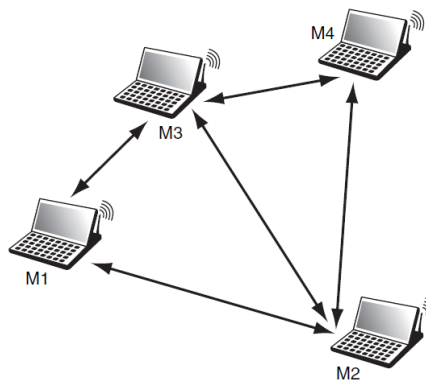


Figure 4 : Réseau ad hoc

1.7. Caractéristiques des réseaux ad hoc

Les réseaux ad hoc sont caractérisés par des propriétés particulières. Chaque propriété est considérée dans la littérature comme étant une problématique en soi. Nous présentons dans ce qui suit ces propriétés :

- **L'absence de l'infrastructure** : les réseaux ad hoc se distinguent des autres réseaux mobiles par l'absence de l'infrastructure fixe et par leurs contrôles décentralisés.
- **Les topologies dynamiques** : les nœuds sont libres de se déplacer aléatoirement, alors la topologie du réseau précisément le multi-saut peut changer d'une façon brusque et rapide, et peut être constituée de liaisons unidirectionnelles et bidirectionnelles en même temps.
- **La bande passante limitée et des liens à débits variables** : les liens sans fil posséderont toujours une capacité inférieure à leurs homologues câblés. L'utilisation des



méthodes de partage du canal radio (accès multiple) influence directement la bande passante réservée à un terminal ad hoc.

- **Utilisation limitée de l'énergie** : l'alimentation des nœuds se reposent sur des batteries ou d'autres sources d'énergie limitées. Cette alimentation limitée demande de prendre en considération l'optimisation de la conservation de l'énergie.
- **Sécurité physique limitée** : les réseaux sans fil mobiles sont généralement plus vulnérables à des attaques que les réseaux câblés fixes. En effet, la sécurité est nécessaire à la démocratisation des réseaux MANET.
- **Qualité des liaisons variables** : à cause du bruit et des interférences entre les nœuds, la qualité des liaisons peut varier.

La solution développée pour les réseaux ad-hoc prend pour fondement l'environnement IP. Les mobiles qui jouent le rôle de passerelles (le plus souvent l'ensemble des mobiles) implémentent un routeur dans leurs circuits, de telle sorte que les problèmes posés reviennent essentiellement à des problèmes de routage dans Internet, la mobilité étant gérée par le protocole IP Mobile. MANET (*Mobile Ad-hoc NETWORK*) [95] est un groupe de travail de l'IETF (*Internet Engineering Task Force*) [96] qui se préoccupe de la spécification et de la normalisation des protocoles de routage pour les réseaux ad-hoc au niveau IP. Ce groupe s'est appuyé sur les protocoles classiques d'Internet et les a perfectionné pour qu'ils puissent fonctionner avec des routeurs mobiles.

Les réseaux ad-hoc posent de nombreux problèmes du fait de la mobilité des nœuds. Ceci résulte en une topologie changeante dans le temps d'une manière imprévisible. D'autre part, la portée des transmissions radio étant limitée les nœuds formant le réseau ad hoc doivent se relayer par un protocole de routage dédié pour retransmettre les messages d'une source vers une destination.

Les protocoles de routage doivent prendre en compte les caractéristiques de ces réseaux mobiles, multi-sauts et sans fil. Les difficultés de conception sont principalement liées (i) aux caractéristiques spécifiques des liens sans fil et (ii) aux changements fréquents de topologie dus à la mobilité.

Les protocoles de routage des réseaux ad hoc s'appuient sur trois modèles de fonctionnement : les protocoles proactifs, les protocoles réactifs et les protocoles hybrides. On peut les différencier par la méthode utilisée pour découvrir le chemin entre le nœud source et le nœud destination.

Les protocoles proactifs établissent et mettent à jour les routes pour tous les nœuds du réseau en se basant sur l'échange périodique d'information de routage. Dans le cas des protocoles réactifs les routes sont établies uniquement à la demande. En effet, une procédure de découverte de route est déclenchée lorsqu'un nœud souhaite envoyer des paquets vers un destinataire dont la route est inconnue. Les protocoles hybrides quant à

eux, combinent les techniques des protocoles proactifs et réactifs. Ils utilisent un protocole proactif, pour apprendre le proche voisinage. Au-delà de cette zone prédéfinie, le protocole hybride fait appel aux techniques des protocoles réactifs pour la recherche des routes.

Les réseaux ad-hoc sont utiles dans de nombreux cas de figure. Ils permettent de mettre en place des réseaux dans un laps de temps restreint, en cas, par exemple, de tremblement de terre ou pour un meeting avec un très grand nombre de participants. Une autre possibilité est d'étendre l'accès à une cellule d'un réseau sans fil comme Wi-Fi. Comme illustré à la figure 5, un terminal situé hors d'une cellule peut se connecter à une machine d'un autre utilisateur se trouvant dans la zone de couverture de la cellule. Ce dernier sert de routeur intermédiaire pour accéder à l'antenne de la cellule.

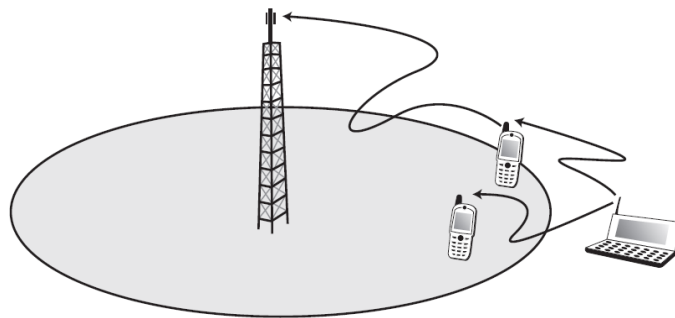


Figure 5 : Extension de couverture par un réseau ad hoc

2. Protocole de transport

2.1. Introduction

La fonctionnalité principale de la couche transport est le transfert de l'émetteur vers le récepteur, des données de bout-en bout de manière fiable et économique. Elle permet d'isoler les niveaux supérieurs des variations technologiques et des imperfections des couches inférieures. Pour cela, elle emploie des protocoles de transport qui utilisent les services de la couche réseau (IP) pour assurer l'acheminement des données. Les protocoles largement déployés dans les réseaux IP sont : TCP (Transmission Control Protocol), UDP (User Datagram Protocol), les protocoles DCCP (Datagram Congestion Control Protocol) et SCTP (Stream Control Transmission Protocol).

Etant donné que notre intérêt porte sur l'amélioration de TCP dans les réseaux ad hoc, nous nous focalisons sur sa description en donnant ces caractéristiques dans les réseaux filaires pour lesquels il a été conçu et nous étudions son comportement dans les réseaux sans fil.

2.2. Le protocole TCP

Le protocole de contrôle de transmission TCP a été défini pour fournir un service de transfert de données fiable entre deux applications sur des stations distantes raccordées par un réseau à commutation de paquets utilisant le protocole IP (Internet Protocol). La fiabilité du transfert est obtenue par différents mécanismes tels que l'établissement de connexion, la gestion de timers de retransmissions ou encore le contrôle de la fenêtre de retransmissions. La spécification initiale, définie dans le Request For Comments RFC 793 de 1981 [97] a fait l'objet de nombreux travaux qui ont conduit à des améliorations de la spécification initiale.

Dans la pratique, la plupart des déploiements du protocole TCP ont été soigneusement conçus dans le contexte des réseaux câblés. Dans un environnement ad hoc l'implémentation de TCP peut conduire à des performances médiocres dues aux propriétés intrinsèques des réseaux sans fil. Des études récentes de métrologie rapportent que TCP est non seulement utilisé pour le transport de données, mais est également utilisé pour le transport des flux "streaming" [98].

2.2.1. Caractéristiques et fonctionnement général

TCP est un protocole de transport de bout-en-bout, fonctionnant en mode orienté connexion dont l'objectif est de transmettre de manière fiable des données applicatives entre deux utilisateurs. Le terme orienté connexion signifie que deux machines souhaitant communiquer via TCP doivent commencer par établir une connexion avant de pouvoir dialoguer. Une fois la connexion établie, les deux extrémités de la connexion peuvent communiquer de manière bidirectionnelle (full-duplex).

Outre le mode orienté connexion et la fiabilité, les principales fonctionnalités de TCP sont :

Ordre : TCP assure, par utilisation des numéros de séquence de paquets et des tampons de réception, que les paquets arrivent dans le même ordre que celui de leur émission.

Contrôle de congestion : des mécanismes de contrôle de congestion (slow-start, Congestion Avoidance, Fast Retransmit, Fast Recovery [21]) permettent d'éviter l'engorgement du réseau lorsque ce dernier est saturé et d'augmenter le débit de transmission lorsque le réseau est faiblement chargé. En fait, le contrôle de congestion assure une corrélation entre le débit d'émission et le débit disponible sur le réseau tout en garantissant une certaine équité pour le partage de ce débit entre plusieurs flux TCP.

Contrôle de flux : TCP utilise un mécanisme de fenêtre d'émission glissante dynamique pour contrôler le débit de transmission de l'émetteur. Le débit d'une connexion est donc directement lié à la taille de sa fenêtre, cette dernière représentant la quantité d'information pouvant être émise. Ce mécanisme est également utilisé pour le

contrôle de congestion. N'ayant pas ou peu d'informations sur la teneur du réseau emprunté, notamment en termes de type de liens, de débit disponible ou de congestion éventuelle, TCP ne peut compter que sur les événements qu'il interprète comme des notifications de congestion pour établir le débit de la connexion et assurer aux utilisateurs le meilleur débit instantané possible.

Transfert de flux d'octets de données : TCP découpe les flux d'octets de données de l'application en segments dont la taille est en fonction de la fenêtre glissante. Les segments seront ensuite réassemblés par le récepteur afin de reformer un flux d'octets qu'il fournira à l'application destinataire.

Contrôle d'erreur sur bit : TCP intègre un mécanisme de checksum sur 16 bits pour couvrir les erreurs sur bit de l'entête et des données.

2.2.1.1. Mécanisme de l'acquittement

Les acquittements sont utilisés pour fiabiliser la communication. En effet, chaque segment émis dans un flux TCP sera acquitté par le récepteur. Dans certaines conditions, les acquittements ne parviennent pas à l'émetteur. Dans ces cas et après un délai d'attente fixé, chaque segment émis mais non acquittés et donc considérés comme perdu est alors réémis. Le délai d'attente ou RTO (Retransmit Time Out) est enclenché à l'établissement de la connexion et remis à zéro chaque fois que la transmission d'un segment est correctement validée. Il est calibré en fonction du délai d'aller-retour entre l'émetteur et le récepteur noté RTT (Round Trip Time). Le RTT est lui-même mesuré grâce aux temps de réception des acquittements. La figure 6 représente le mécanisme d'acquittement.

Sachant que la perte d'un segment est généralement due à une congestion sur le réseau, les acquittements servent également à réguler le débit d'émission des sources TCP.

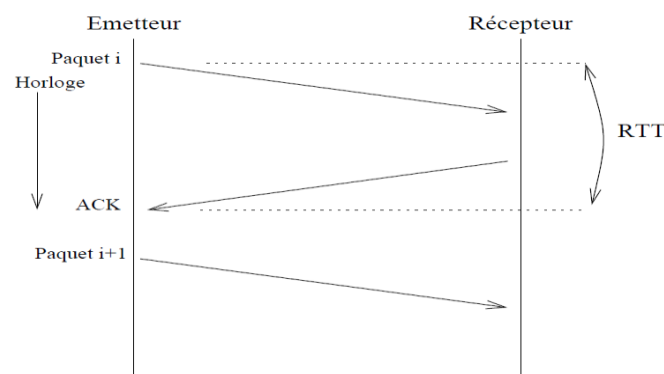


Figure 6 : Mécanisme de l'acquittement

Le délai d'aller-retour (RTT) est mesuré grâce aux temps d'émission des segments et de réception de leur acquittement. Le TCP utilise le RTT pour calculer le RTO en

utilisant l'algorithme de V. Jacobson [98]. Cet algorithme fournit une valeur lissée du RTT en effectuant une moyenne pondérée de la valeur actuelle du RTT lissé et de la dernière mesure du RTT instantané. Il permet également de calculer un indice de variation du RTT (RTTVAR). Chaque évaluation du RTT et de sa variation donne lieu à la mise à jour du RTO [99].

2.2.1.2. La connexion

Entre l'établissement et la fermeture de la connexion, le mode opératoire de TCP est séparé en deux états principaux permettant d'assurer le contrôle de flux et de congestion :

- Le début de connexion.
- L'état stable (le TCP fonctionne correctement).

L'établissement d'une connexion TCP s'effectue en trois phases (figure 7) :

- L'établissement de la connexion
- Le transfert de données
- La fin de la connexion

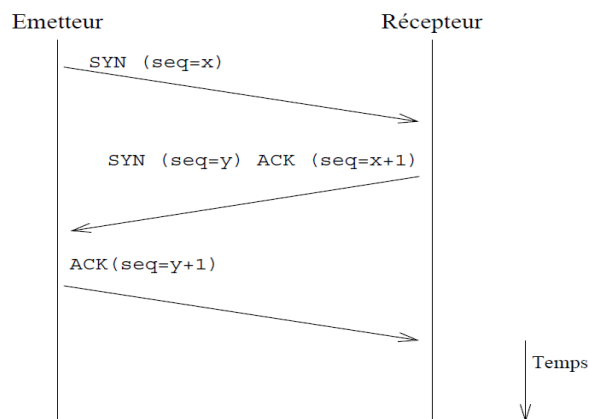


Figure 7 : Etablissement d'une connexion

Où :

- SYN est un paquet de demande de **s**ynchronisation ou établissement de connexion
- ACK (**a**cknowledgment) signale que le paquet est bien reçu
- SYN-ACK (**s**ynchronize, **a**cknowledge) paquet envoyé par le récepteur.

La Figure 8 est une représentation simplifiée du diagramme d'états de TCP insistant sur les changements d'états qui ont lieu entre l'ouverture et la fermeture d'une connexion.

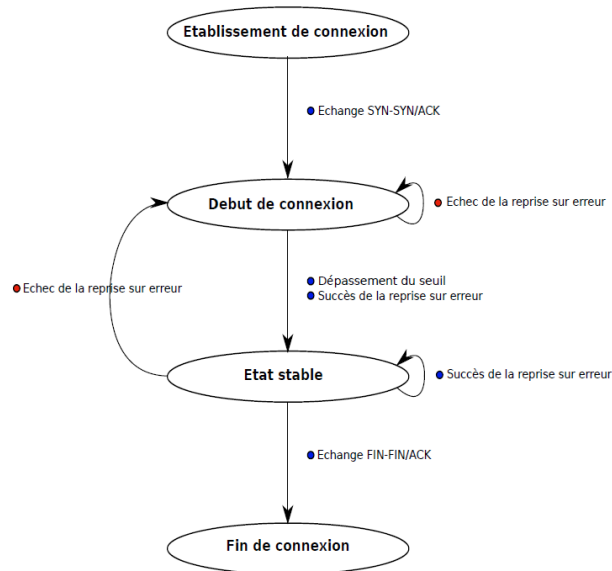


Figure 8 : Diagramme d'états simplifié de TCP

3. Fonctions de contrôle de congestion

Le contrôle de congestion est un mécanisme qui permet de régir le débit d'un ou de plusieurs flux. Ce mécanisme vise généralement à répondre aux trois exigences suivantes :

- Maximiser l'utilisation de la bande passante.
- Limiter le nombre de paquets perdus, c'est-à-dire ne pas dépasser les capacités des ressources du réseau.
- Partager équitablement la bande passante entre les différents flux parcourant le ou les liens les plus lents.

Ce contrôle est implémenté à l'aide d'une variable (cwnd) nommée « congestion window » ou fenêtre de congestion. Concrètement, le nombre maximum de segments de données que l'émetteur peut envoyer avant d'en recevoir le premier acquittement est le minimum entre cette variable (cwnd) et la taille de la fenêtre annoncée par le récepteur à chaque acquittement de paquet. Le contenu de cette variable est piloté par les algorithmes de départ lent (slow start) et d'évitement de congestion (congestion avoidance). La limite de croissance de la variable cwnd est la taille de la fenêtre annoncée par le récepteur. Une fois la capacité de débit maximale atteinte, si un paquet est perdu, l'algorithme d'évitement

de congestion diminue linéairement la valeur de $cwnd$ (contrairement au slow start qui l'augmente exponentiellement).

Dans ce paragraphe nous allons décrire le mécanisme sur lequel repose le contrôle de congestion des différentes versions de TCP.

3.1. La phase slow-start (démarrage lent)

La phase de démarrage lent, ou slow-start est utilisée quand le TCP ignore quelle est la bande passante disponible au démarrage de la session, ou lorsque les conditions du réseau semblent avoir grandement changé forçant à redécouvrir la nouvelle bande passante disponible. TCP utilise ce mécanisme pour sonder le réseau : il commence par émettre une petite quantité de données puis augmente rapidement sa capacité d'émission jusqu'à la détection d'un événement de congestion ou au dépassement d'une variable appelée $ssthresh$, qui détermine si la connexion doit continuer en slow-start ou passer à l'état stable.

Durant le slow-start, l'émetteur commencera par émettre le contenu de la fenêtre de congestion $cwnd$, dont la taille est communément fixée à 3 segments par la RFC 3390 [29] et attend l'acquittement. Une fois l'acquittement reçu, l'émetteur peut alors doubler le nombre de segments et attendre les acquittements. Ce comportement se traduit par une croissance exponentielle de la fenêtre de congestion qui double pour chaque délai aller-retour, ou RTT jusqu'à trouver la limite de congestion du réseau. Une fois cette limite atteinte, des segments commencent à se perdre et l'émission reprend avec une fenêtre de congestion à 1. TCP quitte alors la phase de slow-start pour passer en mode d'évitement de congestion et divise par deux la fenêtre de congestion. La figure suivante (figure 9) illustre la phase du slow-start:

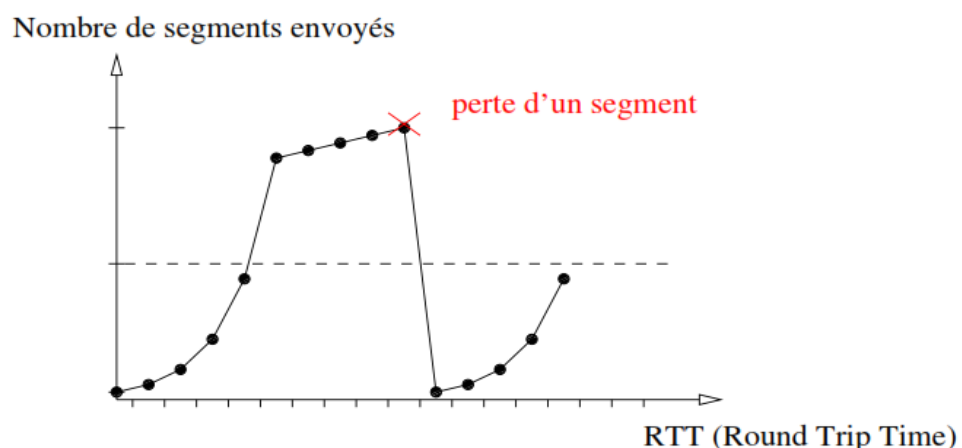


Figure 9 : Démarrage lent de TCP

Le slow-start permet ainsi d'éviter une surcharge du réseau qui serait préjudiciable non seulement à la performance individuelle de la connexion mais également à la

performance collective du réseau. Cependant, il n'en reste pas moins trop conservateur dans le cas des transmissions de connexions courtes et inadapté aux réseaux à long RTT de par sa dépendance directe à ce paramètre. Par exemple, la transmission d'une connexion de 10 segments, représentative des objets HTTP, nécessitera au minimum 3 RTTs et donc plus d'1.5 secondes dans un environnement satellitaire.

3.2. Congestion avoidance (évitement de congestion)

La phase d'évitement de congestion est utilisée quand le flux connaît approximativement quel est le débit disponible. Celui-ci est estimé à la valeur du débit utilisé lors de la dernière perte. Il s'agit pendant cette phase de se rapprocher lentement du débit disponible pour éviter de provoquer une congestion trop importante. Quand une perte est détectée (par trois ACK identiques ou par l'expiration du délai de retransmission (RTO)), la fenêtre de congestion est diminuée en fonction d'un paramètre β .

3.3. L'algorithme Additive Increase and Multiplicative Decrease (AIMD)

TCP utilise l'algorithme distribué Additive Increase and Multiplicative Decrease (AIMD) permettant de faire converger les différents flux partageant les mêmes conditions réseaux vers le même débit équitable. Cet algorithme est basé sur une augmentation additive assurée par la phase d'évitement de congestion et une réduction multiplicative de la fenêtre cwnd réalisée à chaque congestion où cwnd est réduite de moitié.

Lorsque cwnd atteint la taille correspondant à un partage équitable de la bande passante, le même débit est obtenu par chaque flux TCP de même RTT traversant le lien congestionné. Dans ce cas, chaque flux subit les cycles d'une perte suivie d'une phase d'évitement de congestion [100].

Pour sortir d'une congestion d'autres solutions ont été proposées où la fenêtre de congestion est réduite différemment. Les objectifs à garder en vue pour ces algorithmes sont d'être efficaces, équitables et qu'ils présentent une convergence vers une solution adéquate à partir d'un état quelconque et avec une vitesse assez rapide. Ainsi Z.Wang et J. Crowcroft [101] proposent l'algorithme DUAL dans lequel la fenêtre de congestion CW est réduite d'un facteur 8 en fonction des RTT mesurés tous les round-trip. De même que R. Jain [102] propose une solution TRI-S dans laquelle la fenêtre est également réduite d'un facteur 8.

3.4. La reprise sur erreur

L'expiration d'une temporisation sur la non-réception d'un ACK a longtemps été le seul moyen de détecter une perte. Cette temporisation appelée Retransmission Time Out (RTO) est enclenchée à l'établissement de la connexion et remise à zéro chaque fois que la

transmission d'un segment est correctement validée. Sa durée importante, fixée à 3 secondes par la RFC 2988 [110], permet d'être certain que le segment attendu est bien perdu mais également de s'assurer que le réseau a eu le temps nécessaire pour sortir d'un état potentiel de congestion. L'expiration du Timer entraîne alors irrémédiablement le retour dans l'état précédent avec une CW égale à 1.

Avec ce seul mécanisme de détection, aucune distinction n'est par exemple possible entre les pertes dues à des erreurs sur le support et les pertes dues à une congestion majeure du réseau. Toute perte résulte en une même limitation du débit de la connexion.

Depuis certaines versions, la reprise sur erreur de TCP a été affinée en ajoutant des mécanismes qui lui permettent de distinguer les raisons d'une perte afin de la traiter plus efficacement. Ces mécanismes reposent sur un des principes de TCP qui veut que le récepteur envoie à chaque réception de segment un ACK du dernier segment reçu de façon ordonnée. Ainsi, la réception d'un segment ne faisant pas numériquement suite au précédent entraîne l'émission d'un DupACK, c'est-à-dire un ACK en tout point équivalent au dernier envoyé.

3.4.1. Fast retransmit (retransmission rapide)

Fast Retransmit utilise la réception de ces DupACKs pour statuer sur la raison de la perte. La réception de 3 DupACKs, nombre jugé suffisant pour considérer que les segments n'arriveront pas de façon désordonnée et que la perte n'était pas due à une congestion majeure du réseau mais à un événement isolé, entraîne la retransmission instantanée du prochain segment attendu par le récepteur et la transition vers l'état stable.

3.4.2. Fast-recovery (recouvrement rapide).

Ce passage à l'état stable se fait via une réduction significative de cwnd afin de prévenir une potentielle surcharge du réseau. Fast-Recovery [110, 111] est alors utilisé pour diminuer l'effet de cette réduction de la fenêtre : chacun des DupACKs reçu incrémentera la taille de cwnd d'une unité et ce jusqu'à la réception de l'accusé du segment retransmis.

3.4.3. Selective Acknowledgment (SACK)

Il n'est pas rare que TCP subisse des pertes multiples dans une même fenêtre, ce qui a pour conséquence de détériorer de façon importante sa performance. En effet, sans mécanisme approprié, la réception des DupACKs ne permet à TCP de récupérer qu'une perte par RTT. Le mécanisme SACK [103] associé à une politique de retransmission sélective permet de récupérer des pertes multiples. Le récepteur TCP ne se contente plus de renvoyer un DupACK accusant réception du dernier segment arrivé dans l'ordre, mais envoie également à l'émetteur les segments qu'il a correctement reçus dans un champ

SACK. L'émetteur peut alors retransmettre les données manquantes dès réception des DupACKs signalant le ou les mêmes segments manquants.

4. Variantes de TCP

Depuis l'apparition du protocole de transport TCP, plusieurs algorithmes qui visent à améliorer le mécanisme de contrôle de congestion ont été proposés. Tous ces algorithmes utilisent le même principe de transmission que TCP de base, par contre chaque algorithme propose un nouveau mécanisme de détection et d'évitement de congestion. Ces différentes variantes ont été proposées dans le but de permettre à TCP de réagir au mieux aux pertes de paquets. Dans ce qui suit, nous allons donner un aperçu sur le fonctionnement des principales variantes de TCP.

4.1. TCP Tahoe

La première version de TCP basée sur un système de fenêtrage variable est apparue pour répondre aux situations de congestion dans les réseaux longue distance. Cette version est baptisée TCP Tahoe. Elle utilise un système de slow-start, avec une valeur initiale de cwnd à 1 et une valeur de cwnd maximum de ssthresh (avec une valeur par défaut de 65535). Au début TCP Tahoe effectue un slow start, une fois qu'une perte de paquet se produise, le nouveau ssthresh prend la valeur de cwnd courante et celle-ci est remise à la valeur 1. Une fois le seuil ssthresh atteint, TCP exécute une Congestion Avoidance. À partir de là la valeur de cwnd augmente de façon linéaire et donc plus lentement qu'en slow-start. A la réception de trois fois le même ACK, une congestion est déclarée et le paquet perdu est renvoyé sans attendre l'expiration du timeout (fast retransmit).

Le problème majeur de Tahoe est le temps dépensé dans la détection de pertes. En effet, les temporisateurs sont généralement et même plus souvent d'une résolution de 500 ms, pendant que les délais d'aller retour tendent de plus en plus à se raccourcir (câbles torsadé, fibre optique,...). La détection d'une perte peut alors prendre au moins 500 ms ce qui constitue plusieurs délais d'aller retour.

4.2. TCP Reno

La différence avec Tahoe est qu'il utilise le Fast-Recovery. A la réception de trois ACK dupliqués il réduit de moitié la valeur de cwnd, ensuite il fixe le seuil de ssthresh à la taille de cwnd. Il entre ensuite en phase de retransmission rapide et en recouvrement rapide. S'il a un timeout il entre à nouveau en phase démarrage lent comme avec TCP Tahoe mais avec une fenêtre de congestion à 1MSS (maximum segment size).

Les principales innovations dans TCP Reno par rapport à TCP Tahoe sont le facteur de réduction de la fenêtre, dans le cas où une perte est détectée par acquittements dupliqués, et les actions à entreprendre à ce moment. En effet, dans Reno lors de la détection d'une perte par acquittements dupliqués, seul le paquet supposé être perdu, mais pas les suivants, est retransmis, et la phase de congestion au lieu du slow start est exécutée à nouveau. Cependant, les performances de TCP Reno se dégradent lorsque plusieurs pertes successives ont lieu dans la même fenêtre de données.

4.3. TCP New Reno

L'algorithme New Reno n'est qu'une modification légère mais significative apportée à l'algorithme Reno. En effet, cette modification, permet à TCP New Reno d'éviter de devoir attendre pour qu'un déclenchement de retransmission se produise lorsque plusieurs paquets sont perdus au sein d'une même fenêtre de congestion.

Le changement s'opère au niveau de la phase de recouvrement rapide (Fast recovery) : Tant que l'émetteur n'a pas reçu les ACK de tous les paquets perdus (réception d'un acquittement partiel pour seulement une partie des segments perdus), TCP New Reno reste dans cette phase. Un accusé partiel permet donc de se rendre compte que parmi les paquets retransmis une première fois, certains ont à nouveau été perdus. Dès lors, cette version n'attend pas un Time Out pour retransmettre ces paquets perdus. TCP New Reno ne quitte pas l'algorithme de Fast Recovery contrairement à Reno qui sort de ce mode dès la réception d'un ACK non dupliqué.

La version TCP New Reno apporte des réponses aux pertes successives, cependant elle souffre du fait qu'elle utilise un RTT pour détecter chaque perte de paquet. Néanmoins elle est parmi les versions les plus utilisées de TCP actuellement.

4.4. TCP Vegas

Plutôt que d'attendre une perte de paquet, TCP Vegas prend en compte le temps de réponse du destinataire (le RTT) afin d'en déduire le ratio auquel envoyer les paquets. En fonction du temps de réponse, on est capable de supposer l'état des buffers des routeurs intermédiaires. Vegas modifie pour cela plusieurs algorithmes vus jusqu'ici (slow-start, Congestion Avoidance, Fast Retransmit). Grâce à cette technique, Vegas a de meilleurs débits et moins de pertes de paquets que Reno. Cet algorithme permet d'atteindre un partage équitable des ressources. De même, il y a assez peu de pertes avec Vegas puisqu'à l'état stable, le débit correspond de près à la capacité du lien [33].

TCP Vegas permet de s'adapter plus rapidement aux changements de disponibilité des liens. Cependant, les performances se dégradent lorsqu'il est utilisé avec d'autres protocoles qui ne prennent pas en compte l'état des liens avant une perte de paquet. Son inconvénient est qu'il nécessite une modification de la pile TCP au niveau de l'émetteur et du récepteur.

4.5. TCP Westwood+

TCP Westwood est une variante modifiée côté émetteur de l'algorithme de TCP New Reno. Cette modification a pour but d'apporter une meilleure gestion des LFN (long fat network). TCP Westwood modifie les valeurs de ssthresh et de cwnd en fonction des estimations qu'il fait sur la bande passante disponible. La première version faisant de mauvaises estimations, elle a été corrigée par la suite, d'où le nom Westwood+. Ces estimations se basent, comme pour Vegas sur le temps de réponse du destinataire (RTT), sauf que ces estimations sont utilisées différemment.

TCP Westwood possède un mécanisme qui détecte qu'il n'y a pas de congestion (Persistent Non Congestion Detection, PNCD), ce qui lui permet de rapidement utiliser la bande passante disponible. Il modifie le slow-start (devient moins agressif) en y ajoutant une phase : « Agile Probing ». Cette modification lui permet (avec les estimations via le RTT) d'avoir une phase d'accroissement de cwnd [34] quasi exponentielle puis de se stabiliser, d'accroître, puis de se stabiliser, etc...

L'algorithme gagne beaucoup en efficacité, mais perd en équité puisqu'en cas de pertes de paquets il ne s'adapte pas directement en divisant la fenêtre cwnd par deux.

4.6. TCP SACK

Dans cette version les algorithmes de slow-start et de Congestion Avoidance sont les mêmes que pour TCP Reno. Lors d'une perte de paquets, Selective ACK TCP ou TCP SACK permet de déduire que le récepteur a reçu les paquets jusqu'à la séquence numéro N, mais aussi de préciser à l'émetteur que le récepteur a reçu certains des paquets suivants. L'émetteur n'a plus à renvoyer des segments déjà reçus. SACK est une option qui se négocie à la connexion. Cette option permet de gagner un peu de performances, lors de pertes de segments, mais il est nécessaire qu'il soit implémenté sur les deux parties (émetteur et récepteur).

5. TCP et les réseaux sans fil

Initialement le protocole TCP a été conçu pour faire face aux pertes de paquets liés à la congestion dans les réseaux filaires. Cependant, les études menées sur TCP en environnement sans fil [121], indiquent une dégradation de performances, par rapport à son fonctionnement en environnement filaire. La dégradation de performance s'explique par différents facteurs :

- **Limitation de la bande passante**

Contrairement à un réseau filaire, la largeur de bande disponible est réduite : des débits de 11 à 50 Mbps pour les réseaux locaux sans fil de type 802.11 à comparer avec

des débits de 100 Mbps à 1 Gbps pour les réseaux locaux filaires type IEEE 802.3 (Ethernet), et ceux de l'ordre du 10 Gbps pour les fibres optiques des réseaux longue distance.

- **Taux d'Erreur Lien**

Le taux d'erreurs bits (Bit Error Rate, BER) sur des liaisons sans fil est de l'ordre de 10^{-2} jusqu'à 10^{-3} alors que celui sur des liens filaires se situe entre 10^{-6} et 10^{-12} . Ces erreurs provoquent des destructions de paquets qui empêchent l'émetteur de recevoir un acquittement avant l'expiration du timer de retransmission; celui-ci doit alors, conformément aux procédures de contrôle de congestion, diminuer sa fenêtre et retransmettre car TCP assimile l'altération temporaire à une congestion. Ces altérations sur les liens sans fil peuvent survenir par burst (en rafales) en provoquant plusieurs pertes de paquet sur une période de temps inférieure au temps pour émettre une donnée et recevoir l'acquiescement (temps aller/retour).

Notons que dans le cas de réseaux sans fil utilisés en accès d'un réseau longue distance filaire, la perte de paquet au dernier saut sur la liaison sans fil provoque une retransmission de bout en bout, génératrice de trafic supplémentaire sur le lien sans fil, et donc encombrement et source d'erreurs additionnelles.

- **Mobilité de l'utilisateur**

Quand un mobile se déplace d'une cellule dans une autre, un ensemble d'informations doit être transporté de son ancienne station de base à sa nouvelle station. Cette procédure, nommée Handover, peut provoquer de courtes interruptions de la connectivité. Le protocole de transport TCP assimile ces périodes de déconnexion à des phases de congestion et déclenche inutilement la procédure de contrôle de congestion qui diminue le débit du flux. L'interruption momentanée de la communication peut également être liée à un mobile qui sort de sa zone de portée de transmission ou bien des signaux radio bloqués par des obstacles.

- **Fluctuation des valeurs de RTT**

En plus des pertes de paquets dues à l'inefficacité des canaux sans fil, de fortes fluctuations de valeurs RTT à travers la connexion peuvent entraîner l'expiration du TCP Retransmission Time-Out. Cette situation conduit au déclenchement de l'algorithme de contrôle de congestion et à la retransmission des paquets, bien que ces derniers ne sont pas réellement perdus mais simplement retardés. Ainsi, le protocole TCP diminue sa fenêtre de congestion au minimum et sous-utilise la bande passante disponible inutilement, en plus de sa consommation d'énergie pour retransmettre les paquets supposés perdus.

5.1. Problèmes de TCP dans les réseaux ad hoc mobiles

Les principales raisons qui entraînent une dégradation de performance dans les réseaux ad hoc proviennent, comme dans les réseaux sans fil, de la qualité du lien sans fil,

mais également de la qualité du chemin. Une raison à cette dégradation provient du comportement erroné de TCP qui déduit à tort les pertes de données comme une congestion et diminue inutilement son débit d'émission.

- Pertes de données dans les réseaux ad hoc

Les réseaux ad hoc présentent un taux d'erreurs plus important que celui des réseaux traditionnels en raison du taux d'erreurs (BER) des liaisons sans fil, mais également des pertes liées aux collisions de la couche d'accès au médium, ou encore de problèmes des nœuds cachés.

- Coupures de chemin dans les réseaux ad hoc mobiles

La mobilité des nœuds du réseau provoque des modifications de topologie gérées par la couche réseau. Le protocole de routage est en charge du rétablissement du chemin en cas de rupture du chemin courant, en exécutant un processus de rétablissement dont le délai est non négligeable. Celui-ci est fonction du nombre de nœuds dans le réseau, des types de transmission des nœuds, de la topologie courante du réseau, de la bande passante disponible et de la nature du protocole de routage.

Si ce temps de rétablissement est plus grand que la valeur du temps de retransmission, l'émetteur suppose l'apparition d'une congestion dans le réseau et retransmet les paquets perdus. Ces retransmissions peuvent, si le chemin n'a toujours pas été rétabli, mener à un gaspillage de bande passante et de batterie ainsi qu'une augmentation dans le délai d'acheminement de bout en bout. De plus de par le contrôle de congestion, lorsqu'un nouveau chemin aura été mis en place, le débit sera inutilement faible.

- Impact de la longueur du chemin

La possibilité d'une coupure de chemin augmente avec la longueur de celui-ci et lorsque la longueur augmente, le débit de TCP se dégrade comme nous le montrons sur la Figure 10 :

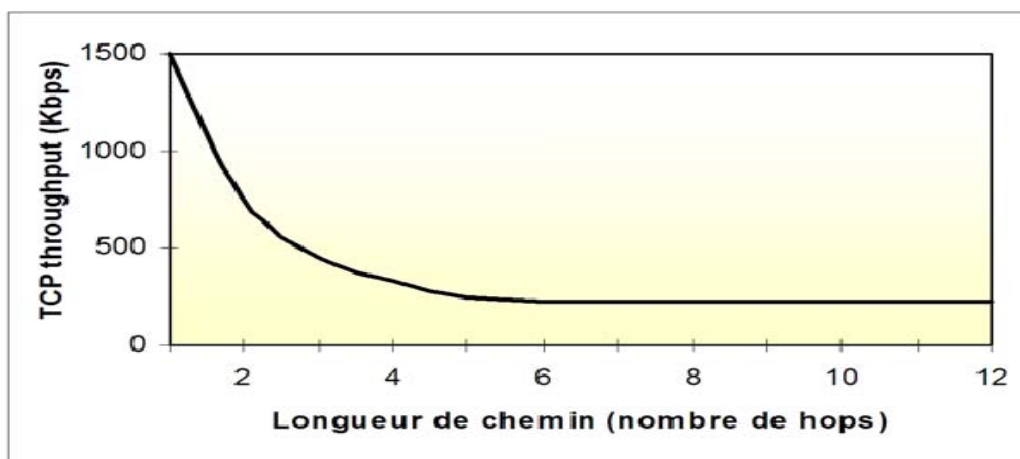


Figure 10 : Variation du débit de TCP avec la longueur du chemin

- **Comportement asymétrique de liens et de chemins**

Les propriétés du canal radio utilisé dans les réseaux ad hoc peuvent mener à l'existence de liens asymétriques entre nœuds. En effet, un paquet est correctement délivré à un nœud dans le sens aller, mais l'acquiescement, peut ne pas être reçu dans le sens inverse. Cette impossibilité de réception peut être temporaire, le temps que le canal radio devienne à nouveau bidirectionnel.

Dans les réseaux ad hoc utilisant le protocole d'accès IEEE 802.11, chaque acquiescement ACK peut nécessiter un échange de trames RTS-CTS (Request To Send – Clear To Send) qui peut mener à une consommation significative de la bande passante sur le chemin inverse. Si ce dernier est symétrique du chemin aller, ceci entraînera une réduction du débit sur le chemin aller. A noter qu'une coupure de chemin sur un chemin d'acquiescement affectera inutilement le protocole.

- **Pic de signalisation et recherche de route**

La construction des tables de routage par la couche réseau nécessite l'envoi de nombreux paquets de signalisation. Dans le cas des protocoles réactifs, le médium va leur être dédié durant une période assez longue. Le fort coût d'accès au médium en temps et en espace, fait que cette période peut-être suffisamment longue pour produire des expirations du timer de retransmission sur certaines connexions. Il n'y a pas réellement de congestion dans le réseau mais seulement un pic de charge de signalisation. Celui-ci peut se produire à l'établissement du chemin mais surtout à la restitution d'une route en cas de rupture.

- **Contrôle de flux par fenêtres glissantes**

TCP emploie une fenêtre coulissante pour contrôler le flux et éviter des engorgements du récepteur. Dans les réseaux ad hoc, le mécanisme de fenêtre coulissante peut ne pas être adapté, lorsque le protocole d'accès n'est pas équitable. Par exemple, les protocoles d'accès tels que le protocole CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance) montrent une iniquité d'accès à court terme, car un nœud qui a capturé le canal a une probabilité plus élevée de le capturer à nouveau. Dans [122], il a été montré que cette iniquité peut mener à la réception d'un groupe de paquets d'acquiescement, qui provoquera un effet de burst sur l'émission de trafic.

6. Le protocole HTTP

HTTP est le protocole le plus utilisé de la couche Application [104] pour la transmission de médias sur TCP. Il définit ses propres formats de message et aussi comment le client et le serveur s'échangent les messages. HTTP est un protocole sans état, car il ne conserve aucune information d'état sur le client et ses requêtes. Il prend en charge deux modes de connexion, le mode de connexion persistante qui correspond à l'utilisation de la même connexion TCP pour servir plusieurs requêtes provenant du même client au

cours d'une période de temps et le mode de connexion non persistante qui correspond à l'utilisation de connexion séparée pour servir chaque requête.

HTTP définit deux types de messages pour son fonctionnement, les messages de requête et les messages de réponse. Un message type de requête HTTP peut être illustré par l'exemple suivant :

```
GET / somedir / page .html HTTP /1.1
Host : www.nt.uni - saarland .de
Connection : close
User - agent : Firefox
Accept - language : En
```

Les messages HTTP sont des messages texte ASCII et se composent de plusieurs lignes. La première ligne correspond à la ligne de demande et les lignes suivantes sont appelées lignes d'en-tête.

- La ligne de requête comporte trois champs, la méthode, l'URL et le champ de version HTTP.
- La méthode correspond à l'action à prendre et les options possibles comprennent GET (permettant d'accéder à l'objet ou au fichier sur le serveur), POST, HEAD, PUT et DELETE.

Le message réponse HTTP peut être illustré par l'exemple suivant :

```
HTTP /1.1. 200 OK
Connection : close
Date : Wed Dec 26 09:17:28 CET 2012
Server : Apache /2.2.3 ( Debian )
Transfer - Encoding : chunked
Content - Type : text / html
( data data data ..... data )
```

Le message de réponse comprend trois sous-sections, à savoir: la ligne d'état, la ligne d'en-tête et les données. La ligne d'état indique l'état du serveur et inclut la version de protocole, le code d'état et le message d'état. Chaque code d'état informe le client sur le résultat de la demande. Par exemple, le code «200 OK» indique que la requête a aboutie (succès). De même, le statut «404 Not Found» est envoyé si le document demandé n'existe pas au niveau du serveur.

7. Les services de vidéo streaming

7.1. Introduction

Au début, l'Internet était utilisé pour partager et échanger des informations textuelles. Dans les années quatre-vingt et avec l'apparition de machines de plus en plus puissantes on commençait à l'utiliser aussi pour transmettre des contenus multimédias et pour des transmissions en temps réel.

L'infrastructure des réseaux était construite pour des transmissions en paquets indépendant du temps. La diffusion en temps réel nécessite un contrôle de flux qui prend en compte les dépendances temporelles des paquets. Ces fonctions n'étaient pas fournies par les anciennes techniques.

7.2. Le streaming vidéo

Le Streaming consiste à découper les données en paquets dont la taille est adaptée à la bande passante disponible entre le client et le serveur. Quand le client a reçu suffisamment de paquets (buffering), l'application cliente commence à jouer un paquet, décompresse un autre et reçoit un troisième. Ainsi la technologie du streaming permet de commencer à visualiser un contenu multimédia sans être obligé de télécharger la totalité des données (*cf.* Figure 11) [105]. Les flux transmis et affichés sont du type multimédia (vidéo, audio ou les deux). D'autres types de médias (photos, livres électroniques, applications, etc) n'ont pas besoin de cette méthode de transfert, car n'ayant aucune contrainte de séquentialité, ils sont téléchargés entièrement avant d'être traités, exécutés ou affichés.

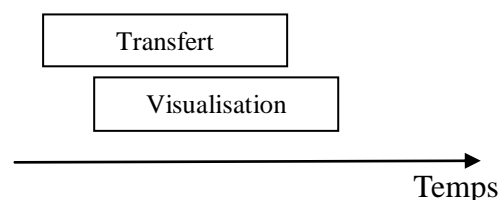


Figure 11 : Média streaming

Les flux audio et vidéo sont généralement volumineux, ce qui suppose un délai important de transmission. Cette latence peut ne pas être acceptable pour l'utilisateur. De plus, l'objet doit être stocké dans le disque de l'équipement récepteur, qui peut parfois être un problème à cause de la taille.

Par ailleurs, les flux multimédia ne sont pas souvent lus dans leur totalité. Les usagers visionnent généralement le début d'une vidéo ou d'une chanson pour en connaître le contenu avant de l'arrêter, puis l'arrêter au milieu. Dans cette situation il est inutile de télécharger tout le fichier audio ou vidéo et ainsi de gaspiller la bande passante du réseau. Les événements dits en direct, audio ou vidéo, créés et transmis en temps réel, doivent être transmis au fur et à mesure de leurs productions. Dans ce cas, il est impossible d'enregistrer préalablement le contenu à transmettre.

Le streaming est considéré comme un défi technologique, de part ses contraintes de temps réel : une sorte de synchronisation des données doit être fournie par la source de l'information et correctement traitée à la réception afin d'afficher les données (vidéo ou audio) de façon adéquate. Le Multimédia streaming emprunte l'Internet comme un moyen de transmission économique. Ce streaming peut être envoyé de deux manières différentes : unicast ou multicast. L'unicast implique l'envoi d'un flux multimédia séparément à chaque destinataire, alors qu'en multicast, le même flux est envoyé dans le réseau, et est dupliqué proportionnellement au nombre d'utilisateurs.

7.3. Le système de streaming multimédia

Le streaming multimédia diffère du transfert de données textuelles. Il peut tolérer une certaine quantité de perte de données tant dit qu'il nécessite une livraison en temps réel (contrainte de délai strict) du contenu. En plus de la diffusion habituelle entre un émetteur et un récepteur unique (unicast), il existe des scénarios où plusieurs clients ont besoin d'être pris en charge au même moment (multicast). Pour assurer ceci, différents protocoles de transport multimédia, codages et format de représentation et de transmission ont été définis et mis au point au fil du temps. La diversité de ces formats a créé de différents types de systèmes de médias streaming (y'a pas de solution standard). En général, les systèmes de diffusion de médias peuvent être classés en fonction du type de contenu en cours de transfert, de la méthode d'accès, des algorithmes de contrôle (d'adaptation) et des protocoles de réseau sous-jacents.

Les principaux composants d'un système de streaming multimédia sont : l'encodeur, le serveur streaming, le client, le protocole de transfert des médias et le réseau physique sous-jacent de streaming. Un tel système de transmission (en continu) est représenté à la figure 12.

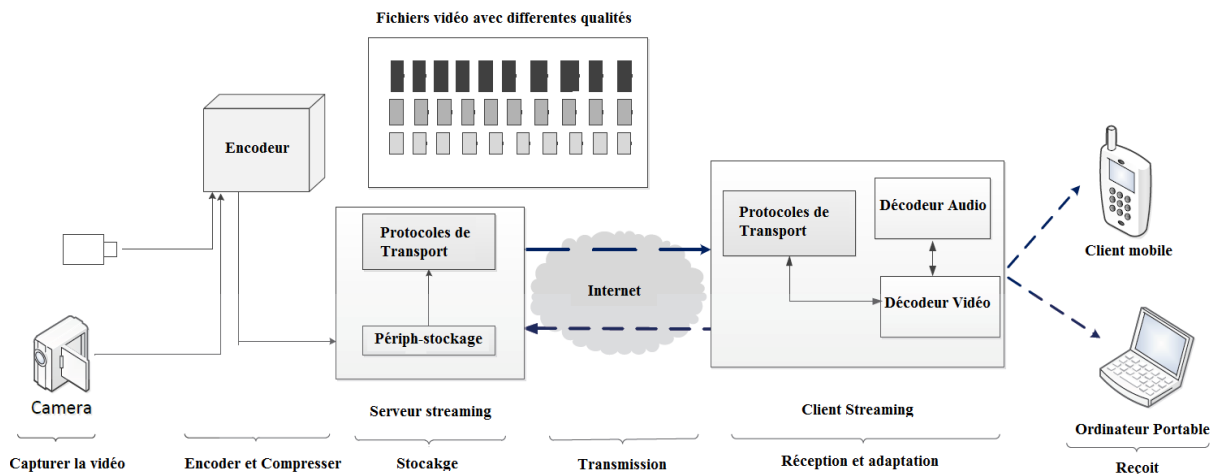


Figure 12 : Système général de streaming multimédia

Les fonctions généralement effectuées par un système de streaming multimédia sont la capture, l'encodage, le stockage, la transmission, la réception, le décodage et le rendu. Initialement, la caméra capture des événements du monde réel et produit le contenu multimédia soit sous forme d'images fixes ou d'une séquence d'images (vidéo). Dans certains cas, la sortie de l'appareil est en format de support brut sans aucune compression. Ce média brut ne peut être transféré sur le réseau car il nécessite une très grande bande passante (en raison de sa grande taille). Pour transmettre le média, il doit être compressé en utilisant des normes de compression appropriée (par exemple, MPEG) à l'extrémité émettrice. Les données compressées sont du côté serveur, soit comme un fichier unique ou un ensemble de fichiers.

La partie qui envoie est elle aussi titulaire des fichiers de description multimédia. Ces fichiers peuvent contenir diverses métadonnées telles que des informations sur le lieu et le moment des fichiers multimédias disponibles sur le serveur. A la demande du récepteur, l'expéditeur lui transmet les fichiers de description et les fichiers multimédias. Du côté récepteur (client), le fichier multimédia est reçu sous la forme de paquets, par la suite ces paquets vont être rassemblés pour donner le fichier multimédia compressé d'origine. Le flux compressé entre dans un décodeur, qui décode les médias et les transmet par la suite au moteur de rendu pour l'affichage.

8. Les différentes technologies de streaming

8.1. Le streaming en direct et le streaming stocké

La diffusion peut être directe ou stockée. Le streaming en direct doit respecter une contrainte de temps très stricte. Les opérations de capture, d'encodage, de transmission, de

décodage et d'affichage doivent être effectués en temps réel et sans délai supplémentaire pour les feedbacks et la retransmission. Dans un environnement avec de telles contraintes de temporisation strictes, si les données sont retardées ou corrompues, aucune mesure corrective ne pourrait être effectuée. Mais en cas de streaming stocké, les données sont stockées chez l'émetteur (serveur). Ils peuvent être corrigés ou retransmis en cas de perte ou de dégradation.

Basé sur la présence (coté client et / ou serveur) ou l'absence d'un mécanisme d'adaptation, le streaming vidéo peut être classé comme adaptatif ou non adaptatif.

8.1.1. Streaming adaptatif

Le streaming adaptatif est un processus qui ajuste la qualité d'une vidéo transmise en fonction des conditions du réseau variables telles que la bande passante disponible, la résolution d'affichage, la puissance du processeur, l'état tampon, etc., afin de garantir la meilleure expérience d'affichage possible. La figure 13 suivante décrit un flux direct adaptatif :

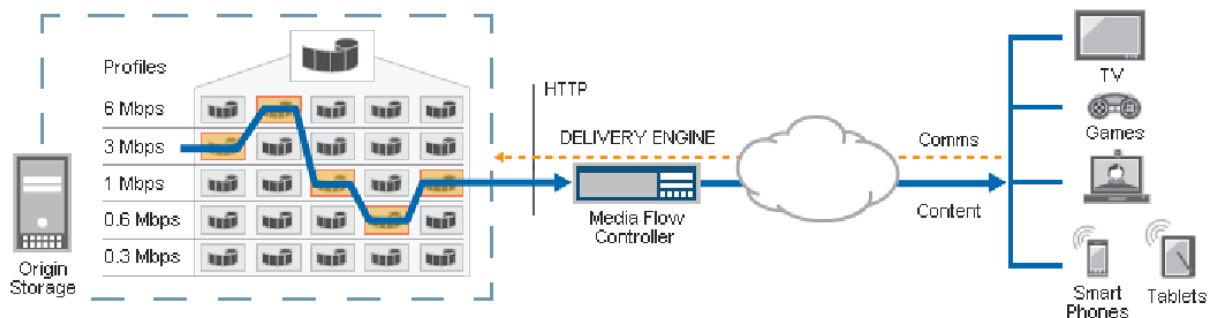


Figure 13 : Flux direct adaptatif

Le streaming adaptatif est devenu très populaire parce qu'il améliore largement l'expérience de l'utilisateur final : la qualité de la vidéo est ajustée dynamiquement aux conditions du réseau de l'utilisateur afin d'offrir la meilleure qualité possible. Il réduit aussi la mémoire tampon et optimise la livraison à travers une large gamme d'appareils.

8.1.2. Le streaming non adaptatif

Dans ce mode, une seule version codée de la vidéo est utilisée. Par conséquent, aucune flexibilité n'est disponible en choisissant automatiquement entre les différentes versions des médias. Ce qui fait que la teneur en continu ne peut être adaptée aux conditions disponibles. Par conséquent, si le dispositif de l'utilisateur ne dispose pas suffisamment de bandes passantes, des ressources du processeur ou de la mémoire,

l'affichage de la vidéo chez l'utilisateur va souffrir d'interruptions (playout). Cependant, dans certains systèmes non adaptatifs, l'utilisateur dispose de multiples qualités et peut choisir manuellement la bonne qualité vidéo qui va avec ces ressources.

9. Conclusion

Dans ce chapitre nous avons défini tout ce qui est pré requis à notre travail. Vu que notre environnement de travail est principalement basé sur les réseaux sans fil en général et les réseaux Ad Hoc mobile en particulier, nous avons commencé par la description de cet environnement en citant les caractéristiques et contraintes spécifiques. Par la suite, comme notre recherche porte sur l'amélioration des performances du protocole de transport TCP, nous avons défini ce protocole ainsi que son fonctionnement et ses différents mécanismes et versions. Enfin, pour le deuxième axe, nous avons défini notion de vidéo streaming adaptative, son évolution, ses différents mécanismes et protocoles afin de les utiliser dans les chapitres suivant comme base de recherche.

CHAPITRE II : ETAT DE L'ART

CHAPITRE II

ÉTAT DE L'ART

Sommaire

1. Introduction.....	35
2. La qualité de service.....	35
3. La qualité d'expérience.....	37
4. L'approche cross-layer.....	40
5. Apprentissage dans le streaming adaptatif.....	43
6. Performances du protocole TCP dans les réseaux sans fil ad hoc.....	47
7. Impact du protocole TCP sur les services vidéo streaming adaptatifs.....	61
8. Conclusion et critiques.....	76

1. Introduction

Dans les sections suivantes de ce chapitre nous passons en revue les éléments utiles qui entrent dans la conception de nos approches.

2. La qualité de service

Le développement considérable de l'Internet et l'arrivée sur le marché d'offres de connexions à haut débit, ont ouvert la voie au développement de nombreuses applications multimédias. Ces applications possèdent des contraintes qui ne s'adaptent plus au service best-effort (service au mieux) fourni nativement par l'Internet.

Ainsi, les exigences du trafic liées à la nature variable du flux multimédia ont rapidement fait sentir le besoin en matière de débit. La diversité de l'information transmise a motivé l'apparition de nouveaux mécanismes afin d'apporter davantage de robustesse dans le fonctionnement du réseau. Ces exigences ont été regroupées sous le concept de Qualité de Service (QoS). Au niveau architecture ce concept est variable pour chaque couche du réseau. La QoS est traitée différemment en fonction du niveau de l'information (message, paquet, trame, bit).

Il existe différentes définitions du concept de QoS. Dans [124], Crawley la définit comme un ensemble de nécessités que doit satisfaire un réseau en transportant un trafic

donné. Ces nécessités sont traduites par des métriques qui font référence à la capacité du réseau à transporter les informations en terme de disponibilité, au délai maximal que doit expérimenter les paquets, au taux de perte de paquets minimal que doit garantir le réseau, etc. [125]. Pour assurer que ces métriques aient des valeurs permettant de satisfaire les besoins, différentes stratégies ou mécanismes sont utilisés par les opérateurs et administrateurs de réseaux. Ces techniques incluent le routage, la régulation de trafic, l'ordonnancement de paquets, le contrôle du trafic, le contrôle d'admission, etc.

2.1 Définition

L'ISO 8402, définit la qualité de service (QoS) comme : l'ensemble des caractéristiques d'un produit ou d'un service qui confère à celui-ci la possibilité de satisfaire aux exigences énoncées. Tant dit que l'ISO 9000, définit cette notion comme l'aptitude d'un ensemble de caractéristiques intrinsèques à satisfaire des exigences.

Dans les réseaux informatiques le terme QoS fait référence à une large collection de technologies et de techniques qui visent à offrir aux applications et à leurs utilisateurs des priorités non fonctionnelles (dites de QoS, telles qu'un débit garanti ou un délai borné par exemple) sur le transfert de leurs données de bout-en-bout.

On considère généralement qu'il n'est pas nécessaire pour l'application de savoir comment l'information est acheminée de bout en bout par le réseau ; cependant, l'utilisateur final est sensible à certaines propriétés qui ont un impact direct sur sa satisfaction vis-à-vis du réseau. On considère généralement que son ressenti peut se résumer par quatre paramètres pertinents qui sont :

- La latence, c'est-à-dire le délai instantané qui impacte la transmission de bout en bout,
- La gigue, qui est la différence de latence entre les paquets (certains vont très vite, d'autres plus lentement),
- Le débit, qui correspond au volume d'information transporté par unité de temps,
- La perte de paquets, c'est-à-dire le taux de paquets qui n'arrivent pas à leur destination.

Suivant le type d'application visé, les quatre paramètres n'ont pas la même importance. Par exemple, dans le cadre d'un transfert de fichier, la latence et la gigue ont peu d'impact sur l'utilisateur, l'essentiel étant que le fichier soit transféré rapidement et sans erreur. Dans le cadre d'une communication téléphonique (transport multimédia interactif), le débit importe peu, tant qu'il suffit à transporter les données vocales numérisées ; cependant les paramètres temporels (latence, gigue) et le taux de paquets perdus ne doivent pas dépasser un certain seuil pour le confort des participants à la conversation téléphonique.

On peut alors définir deux paramètres pour évaluer les besoins d'une application communicante :

- L'élasticité, c'est-à-dire la faculté d'une application réseau à s'adapter [127] à un changement brusque de qualité de connexion (typiquement, une application de transfert de fichier ou une application de streaming vidéo avec adaptation automatique du CODEC selon le débit de la ligne est fortement élastique),
- L'interactivité, c'est-à-dire le niveau d'interaction ou de réactivité nécessaire pour un bon fonctionnement de l'application entre les deux points communicants (typiquement, les applications de type temps réel sont fortement interactives).

3. La qualité d'expérience

3.1. Définition

La QoE désigne, suivant la définition de l'European Telecommunications Standards Institute (ETSI), une mesure de performance du point de vue de l'utilisateur d'un système de communication basée sur des mesures psychologiques objectives et subjectives [123]. Elle prend en compte des paramètres techniques tels que les paramètres de QoS et est influencée par le contexte dans lequel la communication se fait et les attentes de l'utilisateur. Dans le contexte des services de communication, la QoE peut être influencée par de nombreux facteurs tels que le type de service ou le service lui-même, le contenu, le réseau, le matériel de diffusion, l'application utilisée ainsi que le contexte d'utilisation. La QoE est une mesure importante de la performance globale d'un système de communication qui prend en compte ce qui se passe en amont et en aval du réseau jusqu'aux terminaux des utilisateurs (voir Figure 14) :

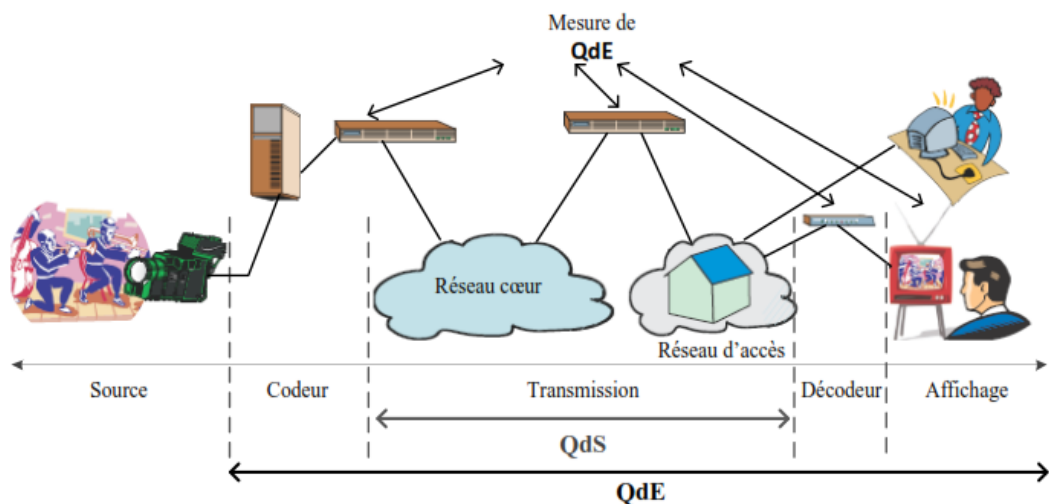


Figure 14 : QoS/QoE domaines d'application inspirés de [126]

Il est important de noter la différence entre QoS et QoE. Considérant le modèle de référence, OSI, la QoS concerne particulièrement les couches inférieures jusqu'à la couche transport et la QoE les couches supérieures (Figure 15).

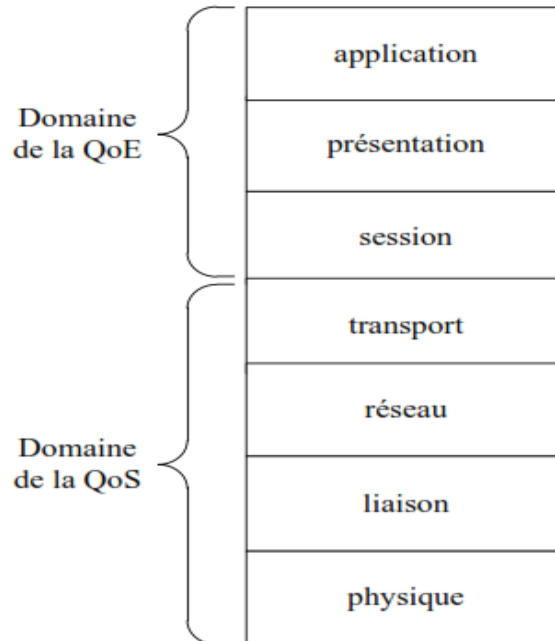


Figure 15 : QoS/QoE couches dans le modèle OSI

3.2. Les approches de la QoE

Pour mesurer la QoE, différentes approches ont été identifiées. Ces approches dépendent du type de l'application et sont catégorisées en méthodes subjectives, objectives et modèles de planification réseau. Les méthodes subjectives se basent sur l'opinion des usagers sur un service multimédia. Suivant ces méthodes, il est demandé aux usagers d'indiquer par des notes variant de 1 à 5 leur niveau de satisfaction et la moyenne donne le MOS. Bien qu'elles soient considérées comme les méthodes les plus fiables pour évaluer la QoE, elles comportent, néanmoins, certains inconvénients. En effet, ces méthodes sont très fastidieuses, consomment beaucoup de temps et ne peuvent être utilisées en temps réel. Pour pallier à ces limitations, des méthodes objectives ont été élaborées et standardisées. Elles permettent d'évaluer la QoE en se basant sur un ensemble de paramètres liés à un service particulier ou sur des paramètres du signal à la sortie pour estimer la QoE. Ces méthodes sont :

3.2.1. Les approches objectives

Par définition, les approches objectives sont basées sur des techniques mathématiques ou/et comparatives qui génèrent des mesures quantitatives de la qualité d'un service proposé à un utilisateur. Ce type d'approche est utile pour la surveillance de

la qualité d'un service en cours d'utilisation, pour la conception du réseau des terminaux utilisés, ainsi que dans l'optimisation de la sélection des codecs à utiliser. Il existe deux types de méthodes objectives:

- **Les méthodes intrusives :**

Les méthodes intrusives sont basées sur les signaux, tandis que les méthodes non intrusives sont basées sur les paramètres du réseau ou de l'application. En général, les méthodes intrusives sont exactes, mais ne sont pas utilisables pour la supervision du trafic en temps réel. Ceci à cause de la nécessité d'avoir à disposition la source du service comme modèle de référence pour la mesure de la qualité. En effet, ces méthodes comparent le signal à la source, considéré comme une référence, au signal déformé à la sortie pour donner une estimation de la QoE. Dans certains cas, les caractéristiques du signal d'entrée peuvent être inconnues.

- **Les méthodes non intrusives :**

À l'inverse, l'avantage des méthodes non intrusives est qu'elles ne nécessitent pas d'acquérir l'original du service comme référence. Elles évaluent le signal à la sortie et une estimation de la QoE est déduite. Elles peuvent donc être utilisées pour une mesure de la qualité de l'expérience en direct.

Les approches objectives ne prennent généralement pas en compte le contexte d'utilisation. Dans le cas d'une diffusion multimédia, ces éléments peuvent être le type de contenu multimédia diffusé ainsi que la façon dont l'utilisateur reçoit ce contenu (type de terminal). Il met aussi de côté la façon dont le contenu est perçu par le système visuel humain (HVS) [71]. C'est pourquoi plusieurs travaux [72, 83] ont montré que les évaluations objectives de la qualité de l'expérience avaient une corrélation faible avec la perception subjective d'un utilisateur humain.

3.2.2. Modèles de planification réseau

Les modèles de planification réseau sont considérés comme une sous-catégorie des méthodes objectives car ils ne se basent pas sur les opinions des usagers. Cependant, ils ne procèdent pas à une analyse du signal réel mais ils utilisent une fonction qui fait le lien entre la QoE et des paramètres intrinsèques du réseau notamment ceux de la QoS.

3.2.3. Approche subjective

Les approches subjectives sont généralement basées sur des tests d'utilisateurs devant évaluer différentes configurations d'un service. De nombreuses publications utilisant ces méthodes ont été réalisées [13, 48, 49]. Mais ceux-ci sont généralement basés sur la recherche d'un seuil d'acceptabilité général permettant de contenter la majorité des utilisateurs. Le principal objectif de ces études est de définir les différents seuils de qualité pour la diffusion multimédia.

3.2.3.1. Score d'opinions moyen (MOS)

La méthode la plus utilisée pour les tests subjectifs est « le score d'opinions moyen » (MOS). Le MOS est une méthode standardisée dans une recommandation de l'ITU T [35].

L'échelle du « score d'opinions moyen » comprend cinq niveaux. Chaque niveau est censé refléter le jugement des utilisateurs concernant la qualité d'un service vidéo. Ce type de test est très souvent utilisé, car il est le seul à prendre en considération la subjectivité inhérente aux ressentis individuels de chaque utilisateur.

3.2.3.2. Défauts du MOS

Certains défauts lui sont souvent reprochés comme :

- Son coût de mise en place important,
- Une mise en place et une réalisation très chronophages,
- Le fait qu'il ne peut pas être utilisé en temps réel,
- Un manque de répétabilité.

Néanmoins, le MOS est un outil indispensable lors de la mise au point d'un nouveau type de codec (dispositif capable de compresser et/ou décompresser un signal numérique). Les algorithmes complexes auxquels un codec moderne fait appel sont souvent assortis d'un certain nombre de paramètres qui le rendent plus ou moins propre à tel ou tel type d'application. Le choix de la valeur des paramètres est très difficile (parfois impossible) à faire de façon rationnelle. Dans ce cas, le MOS vient au secours des chercheurs en apportant une réponse quantitative basée sur une expérience perceptuelle réelle.

4. L'approche cross-layer

L'architecture actuelle des protocoles réseaux, que ce soit le modèle OSI ou le modèle TCP/IP, repose sur un ensemble de modules devant chacun fournir un service précis et ce de façon autonome. Ces modules ont été organisés hiérarchiquement dans une pile, chaque module se trouvant au-dessus du service dont il a besoin pour fournir le sien.

Une telle conception a l'avantage de fournir un niveau d'abstraction permettant de découper le problème en sous-problèmes, plus facile à aborder, et par là même d'améliorer la compréhension que l'on a du problème entier. Le succès de TCP/IP réside certainement plus dans la conception de son architecture que dans les algorithmes qu'il met en œuvre. Cependant, dès l'apparition d'influences entre deux couches, l'envie est forte que ces couches puissent communiquer entre elles un ensemble de variables, de statistiques, qui aideront à la résolution d'un problème. C'est ce que proposent les architectures inter-couches (cross-layer) notamment à travers des retours d'informations (feedbacks).

La conception cross layer met l'accent sur l'optimisation de la performance du réseau en permettant aux différentes couches de la pile de communication à partager des informations d'état ou de coordonner leurs actions en vue d'optimiser conjointement les performances du réseau.

Cette approche consiste à concevoir des protocoles à travers l'exploitation des dépendances entre différentes couches, dont le but est d'obtenir un gain en performance. La conception Cross-layer est aussi définie par l'exploitation de l'interaction et l'exécution d'optimisation conjointe sur plusieurs couches, sous contraintes prédéterminées de ressources [106]. Cette approche particulièrement souhaitable pour les réseaux ad hoc et les réseaux de capteurs, permet d'atténuer relativement l'effet de la limitation des ressources des nœuds et la surcharge significative générée par l'utilisation des protocoles mono couche.

La multiplication des mécanismes de QoS et des techniques d'adaptation sur les différentes couches du modèle TCP/IP a engendré plusieurs problématiques causées principalement par l'isolation des couches. Afin d'apporter une solution à ces problématiques et d'optimiser les performances des systèmes communicants, le concept *cross-layer* a été envisagé pour permettre d'améliorer les performances de transmission dans les réseaux sans fil et d'assurer une meilleure QoS pour les services multimédia.

4.1. La conception cross-layer

Le concept cross-layer a été proposé en premier lieu pour les réseaux sans fil basés sur une architecture TCP/IP. Il a été introduit pour combler essentiellement la perte de performance générée par l'utilisation de l'architecture TCP/IP, conçue spécialement pour les réseaux filaires. En effet, il y a une interdépendance étroite entre les couches protocolaires des réseaux sans fil. L'application de la conception cross-layer peut aider à exploiter cette interdépendance, et favoriser l'adaptabilité de celle-ci sur la base des informations échangées. Cependant, un tel processus de conception doit être soigneusement coordonné, puisqu'il est difficile de caractériser les interactions entre les différentes couches de protocoles. De plus, l'optimisation conjointe des couches peut conduire à des algorithmes complexes, qui engendreront en conséquence des problèmes de mise en œuvre, de débogage, de mise à jour et de standardisation. En effet, il n'existe pas de standard pour la normalisation de la modélisation protocolaire et la méthodologie d'échanges entre les couches. Cependant, plusieurs plans de gestion ont été proposés afin de faciliter la conception des protocoles cross-layer.

4.2. Approches du cross-layer dans les réseaux sans fil

Il existe plusieurs techniques cross-layer pour améliorer les performances des applications utilisant principalement les réseaux sans fil. Historiquement, ces mécanismes étaient limités à des interactions simples entre la couche physique et la couche liaison de

données. De plus, les mécanismes proposés étaient indépendants et visaient essentiellement à l'amélioration d'une imperfection précise. Plusieurs travaux sont arrivés par la suite, proposant de faire collaborer plusieurs couches en prenant en charge plusieurs paramètres, pour une optimisation globale. Trois grandes familles d'approches existent :

- L'approche ascendante (Bottom-up) : dans cette approche, les couches supérieures optimisent leurs mécanismes en fonctions des paramètres (conditions) des couches inférieures. Par exemple, le débit vidéo à la sortie d'un serveur peut varier en fonction des taux de pertes réseaux.
- L'approche descendante (Top-down): dans cette approche, les couches supérieures fournissent des paramètres de configuration aux couches inférieures. De la même manière, les couches inférieures peuvent considérer certaines spécificités de niveau applicatif pour exécuter leurs traitements.
- L'approche mixte (Integrated) : Cette approche exploite les deux approches précédentes dans une même architecture afin de trouver la meilleure configuration et optimisation possible pour un fonctionnement optimal du système.

4.3. Les types d'architecture cross-layer

Il existe trois catégories d'architectures cross-layer [106, 107]:

4.3.1. Architecture cross-layer à base de communication directe

La conception cross-layer à base de communication directe consiste à permettre la communication directe (*sans intermédiaire*) entre les protocoles au niveau des couches adjacentes et non adjacentes. Ainsi, les protocoles mono couche proposés pour les architectures en couches doivent être redéfinis, et de nouvelles routines sont à intégrer. Cette redéfinition permet de pouvoir manipuler les données cross-layer échangées entre les différentes couches.

4.3.2. Architecture cross-layer à base de communication indirecte

Dans cette architecture, une entité intermédiaire se charge des communications entre les différentes couches protocolaires. Par conséquence, le fonctionnement normal de la pile protocolaire est conservé, ce qui permet de ne pas redéfinir les protocoles existants (*architecture en couches*). La dénomination et les fonctionnalités de l'entité intermédiaire varient selon l'architecture cross-layer.

4.3.3. Architecture cross-layer à base de nouvelles abstractions

La troisième catégorie d'architectures Cross-layer est particulièrement distincte des deux autres car elle présente des abstractions complètement nouvelles. Dans cette

approche le concept d'architecture en couches est totalement abandonné, puisque plusieurs couches peuvent être couplées ensembles afin de construire une sorte de super couche, qui peut gérer différentes tâches dans le réseau. Cette nouvelle approche Cross-layer permet d'offrir une forte flexibilité avec un minimum de problèmes.

4.4. Avantages et inconvénients du cross-layer [108,109]

La conception cross-layer a prouvé son efficacité par rapport à l'approche classique en couches. L'application de ce nouveau concept dans les RCSFs a engendré plusieurs améliorations, que ce soit au niveau de la gestion efficace des ressources (*économie d'énergie*) ou de la sécurité. En effet, l'exploitation de l'interaction entre plusieurs couches protocolaires permet d'éliminer toute forme de redondance et de concevoir des protocoles robustes qui traitent le problème d'économie d'énergie et de sécurité, en prenant en considération différentes couches du modèle OSI. Ce type de protocoles cross-layer est plus que nécessaire pour les RCSFs, étant donné qu'il permet de remédier à leurs limitations. L'inconvénient de la conception cross-layer est qu'elle ne peut pas être implémentée dans une expérimentation réelle ou dans une émulation à cause de limites imposées par le matériel qui fonctionne suivant les modèles OSI ou TCP/IP. De ce fait, actuellement, toutes les approches cross-layer attendent une solution qui permettra de surpasser cette impasse.

5. Apprentissage dans le streaming adaptatif

5.1. Introduction

Dans la communauté multimédia beaucoup de recherches visent l'accès ubiquitaire aux contenus en ligne [84]. Le but est d'offrir des services quelque soit le moment, le lieu ou le type du terminal. De nombreux mécanismes d'adaptation multimédia proposés, font face aux contraintes et difficultés qui sont généralement liées à la diversité des documents et contenus multimédia, à l'hétérogénéité des réseaux d'accès et à la variété des terminaux.

Dans ce cadre, les hypothèses spécifiques aux services mobiles doivent être intégrées pour créer des techniques d'adaptation pertinentes. L'approche que nous proposons est basée sur l'usage d'un agent d'apprentissage qui prend en charge la dynamique du contexte à laquelle il doit s'y adapter. Cet agent doit permettre à l'utilisateur d'adapter son comportement en interagissant avec l'environnement du réseau.

A cet effet, il doit percevoir les états successifs du contexte grâce à des observations et effectuer des actions d'adaptation. Ces actions ont un effet sur le contexte en faisant varier les ressources disponibles ou en conditionnant les comportements des utilisateurs. Elles ont donc une influence sur la dynamique stochastique du contexte. Si l'on dispose d'un critère de performance, cette dynamique peut alors être utilisée pour choisir de bonnes politiques d'adaptation. Pour atteindre cet objectif, nous formalisons la politique d'adaptation grâce à un processus de décision de Markov (PDM).

5.2. Apprentissage par renforcement

L'apprentissage par renforcement s'intéresse à l'acquisition automatisée de capacités pour la prise de décisions en environnement complexe et incertain. Il s'agit d'apprendre une stratégie d'action optimale par l'expérience ou par essais-erreurs qui associe à l'état actuel la prochaine action à exécuter de manière à obtenir la meilleure récompense à long terme.

Quatre composantes principales interviennent dans l'apprentissage par renforcement, à savoir :

1- **L'état** résume la situation de l'agent et de l'environnement à chaque instant. Sa dynamique résulte des actions de l'agent sur l'environnement, de l'évolution aléatoire de l'environnement dans le temps et de la dynamique interne de l'agent.

2- **L'action** est choisie et exécutée par l'agent à chaque instant. Suite à cela, il reçoit une récompense instantanée, et perçoit le nouvel état courant.

3- **La récompense** est réelle, positive ou négative. C'est un indicateur immédiat de la qualité d'une action. On cherchera à maximiser les récompenses ou minimiser les pertes (récompenses négatives) pour améliorer la stratégie ou la politique de l'agent.

4- **La politique** modélise le comportement décisionnel de l'agent. Il s'agit dans le cas général d'une fonction, déterministe ou aléatoire, associant des actions à effectuer aux états observés de l'environnement.

L'utilisation du modèle d'apprentissage par renforcement constitue un point important dans notre proposition. Dans la partie qui suit nous allons formaliser les concepts introduits par la théorie de l'apprentissage par renforcement, en particulier nous introduisons le cadre formel standard qui est celui des Processus Décisionnels de Markov (PDM).

Les PDM permettent de modéliser l'évolution d'un système, en particulier de l'environnement dans lequel évolue un agent en fonction des actions qu'il effectue : à chaque pas de temps, l'agent observe l'état courant, choisit une action qu'il exécute pour transiter dans un état et recevoir une récompense.

5.3. Processus de Décision de Markov

Les PDMs font appel à des concepts issus de la théorie des probabilités et de la théorie de la décision [85]. Ils mettent en place un cadre théorique qui permet de définir une stratégie optimale pour un système en interaction avec son environnement dont l'évolution dans le temps n'est pas déterministe. Ils permettent la prise en compte de l'incertitude sur l'effet d'une prise de décision.

Un Processus de Décision de Markov est défini formellement par le quadruplet $\langle S, A, T, R \rangle$ tel que :

- S est l'ensemble fini d'états ;
- A est l'ensemble fini d'actions ;
- $T : S \times A \times S \rightarrow [0, 1]$ est la fonction de transition où $T(s, a, s')$ est la probabilité pour un agent de se trouver dans l'état $s' \in S$ après qu'il a effectué l'action a dans l'état s .
- $R : S \times A \times S \rightarrow \mathfrak{R}$ est la fonction récompense associée à chaque transition. $R(s, a, s') = E\{r_{t+1} \mid s_t=s, a_t=a, s_{t+1}=s'\}$ est l'espérance mathématique de la fonction de renforcement (récompense ou sanction) pour la transition d'un état s vers un état s' .

Les fonctions R et T sont souvent stochastiques et vérifient la propriété de Markov. Un PDM vérifie la propriété de Markov [86], dans le sens où la fonction de transition de l'environnement ne dépend que de l'état précédent de l'agent ainsi que de l'action que celui-ci vient d'exécuter. Cette propriété se traduit par :

$\forall s' \in S, P\{s_{t+1}=s' \mid s_t, a_t\} = P\{s_{t+1}=s' \mid s_t, a_t, s_{t-1}, a_{t-1}, \dots, s_0, a_0\}$, ce qui garantit que la fonction de transition est indépendante de l'ensemble des états et des actions passés de l'agent.

5.4. Résolution d'un PDM dans l'incertain

On appelle politique déterministe une fonction $\pi : S \rightarrow A$ qui associe à chaque état observé, une action $\pi(s)$ à effectuer. Cette politique détermine le comportement de l'agent. D'une manière générale, on utilise des politiques de Markov non déterministes, qui associent à chaque état s et à chaque action a , la probabilité $\pi(s, a)$ que l'agent choisisse l'action a suite à l'observation s . La politique est donc définie par $\pi : S \times A \rightarrow [0;1]$ (avec $\sum_a \pi(s, a) = 1$).

La résolution d'un PDM consiste à trouver la politique optimale π qui maximise la récompense. Lorsque les fonctions T et R sont connues, il est possible d'utiliser les techniques de la programmation dynamique pour résoudre le PDM [87], [88]. Mais dans le cas général, l'agent évolue au sein d'un environnement dynamique et aléatoire et n'a donc pas accès à ces fonctions. La résolution du PDM se situe alors dans le cadre de l'apprentissage par renforcement [89].

L'une des techniques probablement les plus répandues pour résoudre un PDM est le Q-learning [90]. Il s'agit d'apprendre itérativement une fonction $Q : S \times A \rightarrow \mathfrak{R}$ qui approxime la fonction de récompense R du PDM. Les Q-valeurs $Q(s, a)$ sont utilisées pour mesurer la qualité de choisir une action spécifique a dans un certain état s en fonction des récompenses perçues. Initialement, $\forall s \in S, \forall a \in A, Q(s, a) = 0$ et la politique de l'agent est pseudo-aléatoire. À chaque pas de temps, l'agent effectue une action $a = \pi(s)$ suivant la politique π et modifie la valeur de Q par la règle de mise à jour suivante :

$$Q(s, a)_{t+1} = Q(s, a)_t + \alpha_s \cdot (R(s, a)_t + \gamma \cdot \max_a \{Q(s', a)\} - Q(s, a)_t) \quad 2.1$$

Le taux d'apprentissage α_s détermine dans quelle mesure la nouvelle information doit être prise en compte (si $\alpha_s = 0$, l'agent n'apprend rien ; si $\alpha_s = 1$, $Q(s, a)$ ne dépend que de la dernière récompense obtenue). Il détermine ainsi la vitesse de convergence et la précision.

Le paramètre γ joue le rôle d'un taux d'intérêt, on désigne parfois ce paramètre par le terme *coefficient d'amortissement* (discount) qui détermine l'importance des récompenses futures : une récompense reçue k unités de temps plus tard vaut seulement γ^k de ce qu'elle vaudrait au temps courant. Il pondère donc les récompenses immédiates par rapport aux récompenses retardées.

L'importance que l'on accorde aux récompenses futures varie selon la valeur de γ :

- Avec $\gamma \rightarrow 0$, l'agent maximise les récompenses immédiates, et si $\gamma = 0$ l'agent apprend à choisir a_t afin de maximiser seulement la récompense suivante r_{t+1}
- Avec $\gamma \rightarrow 1$, l'agent a un horizon de plus en plus lointain, les récompenses futures ont plus d'importance.

Dans l'apprentissage par renforcement l'agent doit choisir la stratégie à suivre: il peut explorer l'environnement ou exploiter les états et les récompenses déjà connues. Un agent qui apprend par renforcement ne connaît pas explicitement les actions qu'il doit entreprendre. Il doit donc expérimenter toutes les actions à sa disposition afin d'évaluer leurs qualités respectives. Il explore alors activement son environnement de façon à découvrir une politique optimale. Cette situation donne lieu au problème du compromis exploration/exploitation, où il faut en même temps chercher la politique optimale et avoir le meilleur gain possible [90], [91].

5.4.1. La stratégies-greedy

Une approche souvent utilisée pour équilibrer le dilemme exploration/exploitation est la méthode ε -greedy [90] qui définit des proportions fixes de temps pour explorer ou exploiter. Elle utilise une action dite gloutonne (qui est la meilleure action évaluée par la politique courante), une distribution uniforme de probabilités et un $\varepsilon \in [0, 1]$. Cette stratégie choisit l'action gloutonne avec une probabilité $1-\varepsilon$, et une autre action est aléatoirement choisie (en utilisant une distribution uniforme) avec une probabilité ε . Le choix de l'action prochaine a_t est donc :

$$a_t = \begin{cases} \operatorname{argmax}_a Q(s, a) & \text{avec probabilité } 1-\varepsilon \\ \text{action aléatoire} & \text{avec probabilité } \varepsilon \end{cases} \quad 2.2$$

5.4.2. La stratégie de Boltzmann

La prochaine action est choisie selon la distribution de Boltzmann, donnée par la formule suivante :

$$P(a_t) = \frac{e^{Q(s,a)/T}}{\sum_{b \in A} e^{Q(s,b)/T}} \quad 2.3$$

Où T est la température associée à la distribution. Quand T est élevé, la distribution est presque uniforme. Plus la température diminue et $T \rightarrow 0$, plus la probabilité de choisir l'action a dépend de $Q(s, a)$: pour un état s donné, les actions pour lesquelles $Q(s, a)$ est élevé ont plus de chance d'être élues. On se rapproche de la stratégie 0-greedy. En pratique, on fait décroître la température, ce qui permet de moduler exploration et exploitation sans distinguer explicitement ces deux phases.

Il a été montré que si $\gamma < 1$ et si chaque couple (s, a) est visité un nombre infini de fois avec α_s tendant vers 0, alors les valeurs de Q convergent vers la politique optimale π^* [92] (l'action a qui a la meilleure valeur $Q(s, a)$ pour un état s donné correspond à $\pi^*(s)$).

Soit $Q^*(s, a)$ la fonction Q-valeur optimale définie par :

$$Q^*(s, a) = \max_{\pi} Q^\pi(s, a), \forall s \in S, a \in A \quad 2.4$$

Alors:

$$\pi(s) = \operatorname{argmax}_{a' \in A} Q(s, a') \quad 2.5$$

6. Performances du protocole TCP dans les réseaux sans fil ad hoc

6.1. Introduction

Les réseaux ad-hoc sont des systèmes distribués complexes. Ils se composent de nœuds qui s'auto-organisent dynamiquement, pour former temporairement une topologie de réseau aléatoire (sans aucune infrastructure). L'introduction de nouveaux protocoles tels que Bluetooth [15] ou IEEE 802.11 [16] a permis le déploiement commercial des réseaux ad hoc pour des fins commerciales. A l'origine, le protocole de contrôle de transmission (TCP) a été conçu pour fournir et garantir une transmission fiable de bout en bout des données sous forme de paquets. Le fonctionnement du protocole TCP doit théoriquement être indépendant de la technologie utilisée pour la transmission des données, par exemple, si le protocole Internet utilise un support de transmission filaire ou sans fil, ceci ne doit pas avoir un impact sur le comportement ou le bon fonctionnement du protocole TCP. La plupart des améliorations qui ont été déjà faites sur le protocole TCP (validé, approuvé et utilisé) ont été conçus pour un bon fonctionnement et une bonne stabilité dans les réseaux

filaires. Cependant ces travaux ne traitent pas le problème de la dégradation des performances du protocole TCP dans les transmissions sans fil.

6.2. Différentes causes de perte de paquets dans les réseaux sans fil

Contrairement au réseau filaire où la perte de paquets est principalement due à la congestion, la perte de paquets dans les réseaux sans fil peut être causée par plusieurs facteurs. Dans ce qui suit, nous allons décrire en plus des deux principaux facteurs qui sont la puissance du signal reçu et les interférences, d'autres facteurs moins importants.

6.2.1. Problème lié à la puissance du signal :

La puissance du signal reçu ou le RSSI joue un rôle très important dans la bonne transmission des paquets dans les réseaux sans fil ; plus la valeur du RSSI est petite plus le risque de perte de paquets augmente à cause des erreurs de transmission. La valeur du RSSI est très variable sur le temps, elle change en fonction de plusieurs facteurs, tels que le type de la technologie sans fil utilisée, la puissance attribuée au dispositif de transmission sans fil et les obstacles entre l'émetteur et le récepteur. Le facteur qui a le plus d'impact sur la valeur du RSSI est la mobilité des nœuds, plus un nœud sans fil est loin de la zone de couverture sans fil plus sa puissance de signal diminue jusqu'à ce qu'elle atteigne un certain seuil où la transmission s'arrête complètement. Par contre, plus un nœud sans fil se rapproche du centre de la zone de couverture sans fil, plus la puissance de son signal augmente et donc la probabilité d'une perte de paquets due à la puissance du signal diminue.

6.2.2. Problème lié aux interférences

Dans un contexte filaire comme un bus Ethernet, tous les nœuds partagent la même vision de l'état du canal de transmission (libre ou occupé) et des ressources à disposition pour un éventuel échange. Cette vision unifiée n'est plus possible dans les réseaux sans fil du fait que le canal partagé n'est plus un simple support matériel partagé mais un médium immatériel tel que l'air. D'autre part, la vision de chaque nœud dépend de sa portée de transmission et de sa zone d'interférence. Plusieurs problèmes connus sont issus de cette vision limitée et qui sont liés au bruit résultant des interférences tel que le problème du nœud caché.

Etant donné qu'il est très difficile de séparer complètement en fréquence les transmissions simultanées dans les réseaux sans fil, certaines transmissions seront produites en même temps et dans la même bande de fréquences. Par conséquent, la bande passante consommée par les flux de données et les ressources disponibles pour un nœud ne sont plus des concepts locaux, ils dépendent des nœuds voisins partageant le canal de transmission ce qui engendre un bruit qui va augmenter les risques de perte de paquets.

D'autre part il y a un autre type d'interférences qui sont dues à des facteurs externes tels que les ondes dégagées par les canaux électriques de hautes fréquences ou bien des vibrations produites par un bruit très fort (Figure 16). Soit l'exemple d'un chantier ou d'un moteur puissant, qui produit un bruit. L'effet de ce bruit va dégrader la qualité de transmission sans fil et engendrer de temps en temps des erreurs de transmission. Ces erreurs causeront par la suite des pertes de paquets.



Figure 16: Quelques types d'interférence

6.2.3. Autres causes de perte de paquets

- **Effet doppler** : En raison des débits relatifs à l'émetteur et au récepteur.
- **Les stations cachées et exposées [28]** : Dans les réseaux ad hoc, les stations peuvent s'appuyer sur un mécanisme de détection de porteuse physique afin qu'elles déterminent si le canal est inactif, tel que la norme IEEE 802.11 DCF (Distributed Coordination Function), ceci peut engendrer des erreurs de transmission.
- **Les chemins asymétriques** : Les chemins asymétriques dans les réseaux ad hoc peuvent apparaître sous plusieurs formes comme la bande passante asymétrique, le taux d'erreurs asymétriques et le chemin asymétrique.
- **Les contraintes d'énergie** : Puisque les batteries associées à chaque nœud mobile ont une alimentation limitée, la puissance de traitement est limitée. Ceci est un problème majeur dans les réseaux ad hoc, parce que chaque nœud agit comme un système d'extrémité et un routeur en même temps ; ceci implique que de l'énergie supplémentaire est nécessaire pour acheminer et traiter les paquets, le protocole de transport doit pouvoir gérer cette ressource d'une manière efficace.

6.3. TCP et le problème de perte de paquets dans les réseaux sans fil (impact et performance)

Le comportement du protocole TCP vis-à-vis d'une perte de paquet sur une transmission dans certaines conditions (interférences, mobilité,...) dégrade les performances du réseau en matière de bande passante, de délaisetc.

6.3.1. La congestion et la perte de paquet dans les réseaux sans fil

Dans l'algorithme de Van Jacobson [97] (détection de congestion pour TCP), on constate que le mécanisme de contrôle de congestion est insensible à la perte de paquets endommagés. Le taux de perte le plus élevé dû à un paquet endommagé par une fenêtre dégrade le débit du protocole TCP jusqu'à 60% [97].

6.3.1.1. Le délai périodique

Lors de l'utilisation du protocole TCP, le délai périodique représente les déconnexions fréquentes de l'émetteur TCP. Ceci se produit lorsque l'horloge de retransmission double de valeur pour chaque tentative de retransmission non réussite, afin de réduire le taux de transmission, En suite, à la reconnexion du nœud mobile, ces déconnexions font que TCP consomme beaucoup plus de temps pour récupérer une telle réduction, en plus les données ne seront pas transmises durant cette période de temps.

6.3.1.2. Collision des acquittements (ACK) et les paquets de données

Le mécanisme d'évitement de collision IEEE 802.11b élimine toutes les collisions. Puisque le trafic du protocole TCP est bidirectionnel (avec les paquets de données dans un sens et les paquets de d'acquittement dans l'autre sens), il peut y avoir une collision de paquets de données et les paquets d'acquittement. Ces collisions causent une retransmission au niveau de la couche MAC ou au niveau de la couche TCP lorsque la récupération d'erreur n'est pas utilisée dans la couche liaison. Ici, Jacobson a testé le taux de retransmission des paquets UDP et TCP dans un environnement peu susceptible aux interférences. Dans UDP, les retransmissions sont relativement lentes (presque 1%), mais quand il a utilisé TCP, les retransmissions ont augmenté de 5% [97]. Il a expliqué cette augmentation par les collisions des paquets de données et d'acquittements. La réduction des performances n'est pas importante, mais les performances sont encore plus faibles si la récupération d'erreur n'est pas utilisée par la couche liaison. Le débit du TCP sans retransmission dans la couche MAC est inférieur à 23% par rapport à celui obtenu par la retransmission dans la couche MAC [97]. Les performances de l'UDP, même sans retransmission dans la couche MAC, sont un peu plus élevées que celles du TCP avec retransmission dans la couche MAC.

6.4. TCP et la qualité de service dans les réseaux sans fil

La QoS regroupe un ensemble de technologies mises en œuvre pour assurer des débits suffisants et constants sur les réseaux en général et Internet en particulier. Dans les réseaux filaires, le protocole TCP est considéré parmi les moyens les plus efficaces pour garantir une bonne transmission de données à cause de sa fiabilité ainsi que ses mécanismes de contrôle de congestion, de multiplexage qui favorisent le type de données par rapport à un autre type afin de garantir un bon fonctionnement du service. Cependant dans les réseaux sans fil et dans certaines conditions, au lieu d'améliorer la QoS, les mécanismes du protocole TCP font que la QoS se dégrade.

Les graphes 17 et 18 comparent les performances et la qualité de service fourni par le protocole TCP en matière de débit, afin de montrer l'impact du protocole TCP sur la qualité de service des réseaux sans fils sous les contraintes de mobilité et des interférences.

Dans ces figures nous constatons deux faits : quand il n'y a ni mobilité ni interférence, le débit est assez bon avec presque les mêmes valeurs pour les cas filaire et sans-fil. Cependant, dès que les nœuds de la topologie commence à être mobile ou subissent des effets d'interférences, leurs débit se dégradent d'une façon importante dans le réseau sans fil, ce qui impacte négativement la qualité de service du réseau. Par contre sur le réseau filaire ceci n'aura presque aucun impact sur la qualité de service du réseau (impact négligé).

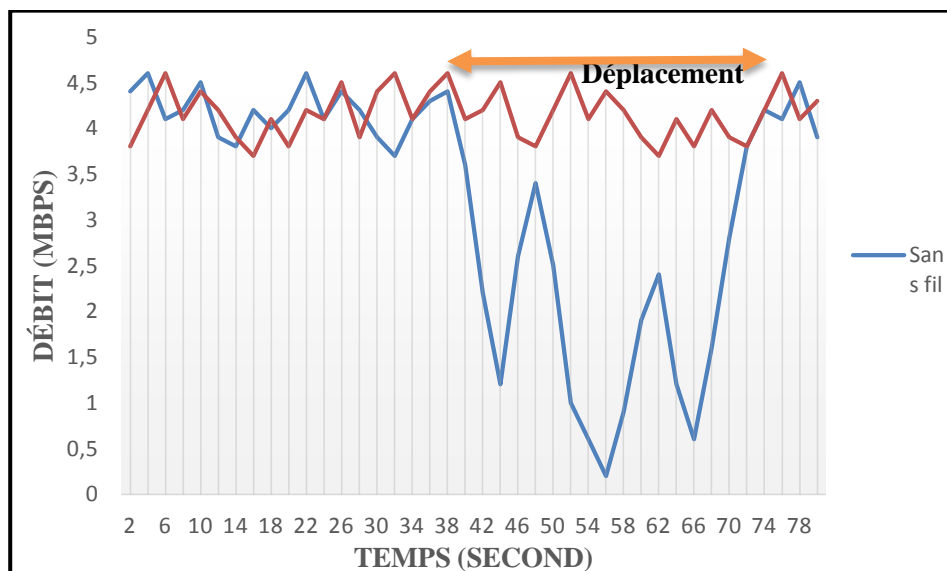


Figure 17 : Comparaison des valeurs du débit filaire et sans fils avec déplacement

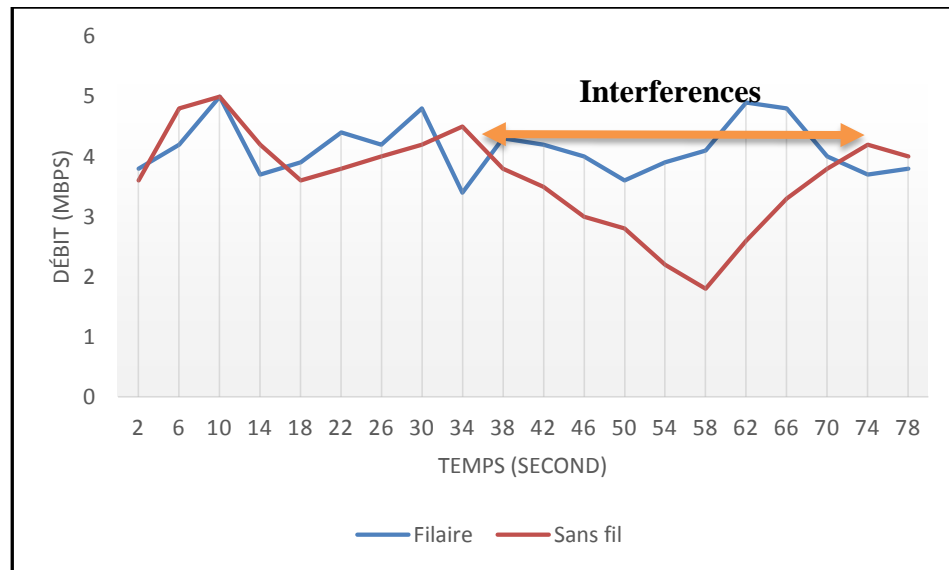


Figure 18 : Comparaison des valeurs du débit filaire et sans fils sous l'effet des interférences.

6.5. Etude comparative des travaux existants

Dans le cadre des améliorations du protocole TCP dans les réseaux sans fils en général et les réseaux ad hoc en particulier, plusieurs approches ont été proposées. La plupart de ces approches visent à trouver une solution qui permet au protocole TCP de faire la différence entre une perte de paquet due à une congestion du lien et une perte de paquet qui est due aux erreurs du lien.

Les solutions proposées jusqu'à présent peuvent être classées en deux catégories, les solutions inter-couches et les solutions basées sur la couche de transport. Ces approches sont résumées dans la figure 19.

6.5.1. Les solutions inter-couches

6.5.1.1. TCP-F(TCP Feedback) [18]

TCP-F est une approche basée sur une réponse de retour (rétroaction) afin d'éviter les erreurs de chemin dans les réseaux MANET. Cette approche permet à l'expéditeur TCP de faire la distinction entre les pertes dues à la congestion du réseau et celles dues à d'autres facteurs. Ainsi, lorsque l'agent de routage d'un nœud détecte l'interruption d'une route, il envoie explicitement un paquet de notification d'erreur de route (RFN) à la source du paquet. En recevant le RFN, la source passe dans un état de sommeil (snooze state). L'expéditeur TCP qui est en état de sommeil arrête l'envoi de paquets et sauvegarde les valeurs de toute variable comme le délai et la taille de la fenêtre de congestion. L'expéditeur TCP reste dans cet état de sommeil jusqu'à ce qu'il reçoit un paquet notification de la restauration de la route (RRN). En recevant le RRN, l'expéditeur TCP va

sortir de l'état de sommeil et va reprendre la transmission en se basant sur les valeurs déjà sauvegardées de la fenêtre de congestion et du délai.

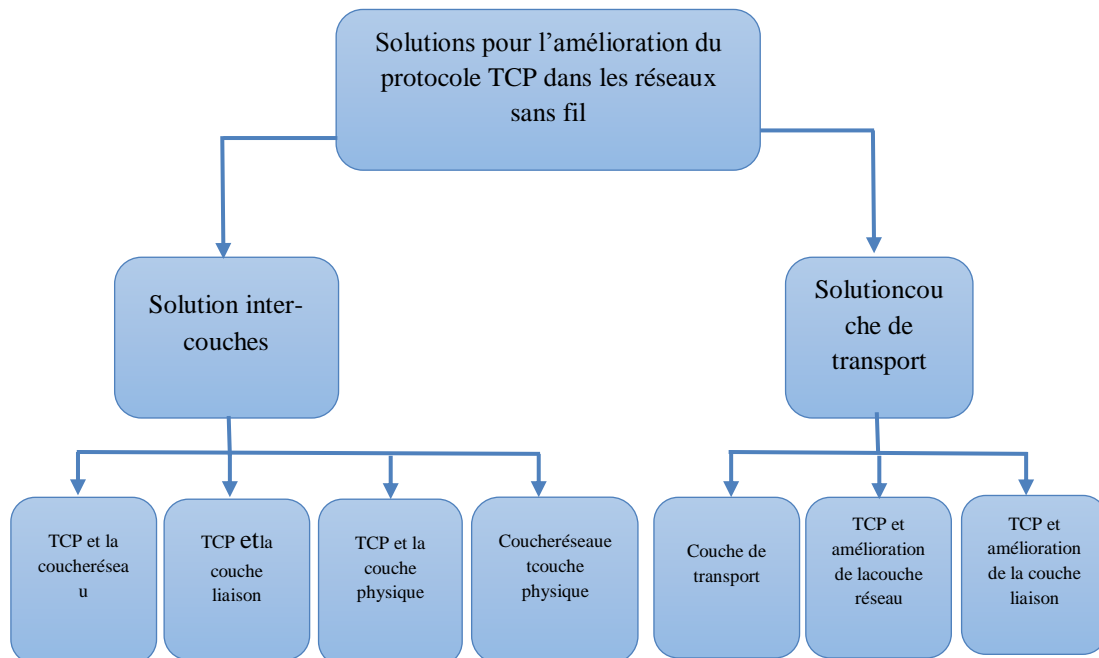


Figure 19 : Classification des solutions pour l'amélioration de TCP dans les réseaux sans fil

Afin d'éviter les scénarios de blocage dans un état de somnolence sur la réception d'une notification RFN, l'expéditeur TCP déclenche une horloge d'échec de route. Lorsque ce délai expire l'algorithme de contrôle de congestion est exécuté normalement.

Dans [18], les auteurs rapportent une amélioration en utilisant TCP-F sur TCP. Le scénario de simulation est basique et ne repose pas sur un réseau ad hoc. Au lieu de cela, ils imitent le comportement d'un réseau ad hoc au niveau de la couche transport.

6.5.1.2. ELFN-based technique [19]

La technique de notification d'erreur de lien explicite ou ELFN-based technique est similaire à TCP-F. Cependant, contrairement à celle-ci, l'évaluation de la proposition est basée sur une interaction réelle entre TCP et le protocole de routage. Cette interaction vise à informer TCP sur les échecs de la route quand ils se produisent. Les auteurs utilisent un message ELFN, qui est envoyé par le protocole de routage à l'expéditeur. Le message

ELFN est comme un message «hôte inaccessible» du protocole ICMP, qui contient les adresses et les ports de l'expéditeur et du récepteur, ainsi que le numéro de séquence de paquets TCP. Sur la réception du message ELFN, la source répond en désactivant son horloge de retransmission et entre dans un mode "veille". Pendant la période d'attente, l'expéditeur TCP sonde le réseau pour vérifier si la route est rétablie, si l'accusé de réception du paquet de sonde est reçu, l'expéditeur TCP quitte le mode veille, reprend son horloge de retransmission et continue les opérations normales.

Cette technique a apporté des améliorations significatives sur le TCP standard, mais d'autres évaluations sont encore nécessaires. En cas de forte charge le protocole ELFN donne de plus mauvais résultats que le TCP standard, ceci est dû au fait que le protocole ELFN est basée sur le sondage régulier du réseau afin de détecter les rétablissements de routes.

6.5.1.3. ATCP [20]

ATCP or TCP ad hoc utilise lui aussi une réponse de retour de la couche réseau. En plus des échecs de route, ATCP a pour but de régler le problème des taux d'erreur élevés de bits (BER). L'expéditeur TCP peut être mis dans plusieurs états, en état de conservation, en état de contrôle de congestion ou en état de retransmission. Sur les nœuds source de TCP, une couche appelée ATCP est inséré entre les couches TCP et IP. ATCP, écoute les informations d'état de réseau fourni par les messages ECN (Explicit Congestion Notification) et par le protocole ICMP "Destination inaccessible", puis ATCP met l'agent de TCP dans l'état approprié selon la réponse reçue. Sur réception d'un message "Destination inaccessible", l'agent TCP entre dans un état de conservation ; lors de cet état l'agent TCP est gelé et aucun paquet n'est envoyé jusqu'à ce qu'une nouvelle route soit trouvée durant le sondage du réseau. Le message ECN est utilisé comme un mécanisme de notification qui indique à l'expéditeur les éventuelles congestions du réseau sur la route utilisée. Lors de la réception du message ECN, le mécanisme de contrôle de congestion du TCP est invoqué normalement sans attente. Pour détecter les pertes de paquets dues aux erreurs de route, ATCP surveille les accusés de réception reçus. Lorsque ATCP reçoit trois ACK dupliqués, il ne transmet pas la troisième ACK en double, mais met TCP dans l'état de conservation et retransmet rapidement le paquet perdu de la mémoire tampon TCP. Après avoir reçu le prochain ACK, ATCP permet à TCP de se mettre dans un état normal.

ATCP a été implémenté, testé et évalué selon différents scénarios, tels que la congestion et les liens perdus. Dans tous les cas, le transfert d'un fichier donné en utilisant ATCP donne de meilleures performances que le TCP standard. Toutefois, le scénario utilisé est un peu spécial, puisque ni les liaisons sans fil, ni les protocoles de routage ad hoc n'ont été considérés, les auteurs ont utilisé un banc d'essai expérimental composé de cinq ordinateurs équipés de cartes Ethernet. Avec ces ordinateurs, les auteurs ont formé un réseau de quatre sauts.

6.5.1.4. TCP-Bus [21]

TCP-Bus utilise les capacités du TCP standard, la mémoire tampon et le numéro de séquence. D'un autre côté, tout comme les approches précédentes TCP-bus se basent aussi sur une réponse de retour ; il utilise les évaluations du réseau pour détecter les échecs de la route afin de réagir convenablement à cet événement. L'apport que propose cette approche est le fait d'introduire la capacité du 'tampon mémoire' dans les nœuds mobiles. Les auteurs sélectionnent un protocole de routage de type initialisation-à-la-demande-de-la-source ou ABR (Routage basé associativité) [22] initié à la demande. Les améliorations suivantes sont proposées :

- **La notification explicite** : deux messages de contrôle sont utilisés pour informer la source de l'échec de la route et du rétablissement de route. Ces messages sont appelés la Notification-de-Déconnection-Explicite-de-Route (NDER) et la Notification-Réussie-Explicite-de-Route (NRER). A la réception de la (NDER) de la part du nœud qui a détecté l'échec de route, ce nœud est appelé Nœud-Pivotant (NP), la source cesse d'envoyer des paquets. De même après le rétablissement de route par le PN en utilisant une requête localisée (RL), le PN transmettra le NRER à la source. A la réception de la NRER, la source reprend la transmission de données.

- **Extension des valeurs du timeout** : pendant la phase de la reconstruction de Route (PRR), tout au long de la route (de la source à la NP) les paquets sont insérés dans la mémoire tampon. Pour éviter les timeout durant la phase PRR, l'horloge de retransmission de paquet est sauvegardée sur le tampon mémoire puis doublée par la suite.

- **Demande de retransmission sélective** : Puisque la valeur de l'horloge de retransmission est doublée, les paquets perdus le long de la route depuis la source vers NP, ne seront pas retransmis jusqu'à l'expiration de l'horloge de retransmission ajustée. Pour résoudre ce problème, une indication est faite à la source de manière à pouvoir retransmettre les paquets perdus d'une manière sélective.

- **Evitement des demandes inutiles de retransmission rapide** : Au moment où la route est rétablie, la destination informe la source des paquets perdus le long de la route (de la NP à la destination). A la réception de cette notification, la source retransmet simplement ces paquets perdus. Le problème est que les paquets mis en mémoire tampon le long du trajet depuis la source vers l'unité de raccordement peuvent arriver à destination plutôt que les paquets retransmis, ainsi la destination répondra par des ACK dupliqués, et les paquets de requêtes inutiles pour la retransmission rapide sont mis à l'écart.

- **La retransmission fiable du message de contrôle** : Afin de garantir avec exactitude le bon fonctionnement de TCP-bus, les auteurs proposent de transmettre de manière fiable les messages de contrôle de routage NDER et NRER. La transmission se fait d'une manière fiable en restant à l'écoute sur le canal après avoir transmis les messages de contrôle. Si un nœud a envoyé un message de contrôle et que ce dernier n'a

pas reçu ce message relayé pendant un timeout, il conclut que le message de commande est perdu, donc il retransmet ce message.

Cette approche (TCP-Bus) offre de nombreuses nouvelles techniques pour l'amélioration de TCP. Les nouvelles contributions de cet article sont les techniques de mise en mémoire tampon et la transmission fiable de messages de contrôle. Dans leur évaluation, les auteurs ont constaté que le protocole TCP-Bus surpasse le TCP standard et le TCP-F dans des conditions différentes. L'évaluation est basée uniquement sur le protocole de routage ABR, d'autres protocoles de routage doivent être pris en compte. Aussi, TCP-Bus n'a pas tenu compte du fait que le pivotement de nœud peut empêcher le calcul d'un nouvel itinéraire partiel à la destination.

6.5.1.5. Split TCP [25]

Les connexions TCP avec un grand nombre de sauts souffrent de défaillances de routes fréquentes en raison de la mobilité. Pour améliorer le débit de ces connexions et régler le problème de partage de débit, le régime de Split TCP a été introduit pour diviser les longues connexions TCP en segments plus courts localisées (Cf. figure 20). Le nœud interface entre deux segments localisés est appelé un proxy. L'agent de routage décide s'il attribue le rôle de proxy à son nœud selon le paramètre de distance inter-proxy. Le proxy intercepte les paquets TCP, les insère dans la mémoire tampon, et accuse leur réception à la source (ou au proxy précédent) en envoyant un acquittement local (LACK). En outre, un proxy est chargé de délivrer les paquets, à un taux approprié, au prochain segment local. Lors de la réception d'un LACK (de la part du prochain proxy ou de la destination finale), le proxy vide sa mémoire tampon des paquets. Afin d'assurer la fiabilité de la source à la destination, un ACK est envoyé par la destination à la source de manière similaire au protocole TCP standard.

Ce schéma divise les fonctionnalités de la couche Transport sur le contrôle de bout-en-bout ainsi que sur le contrôle de congestion. Cela se fait à l'aide de deux fenêtres de transmission à la source, la fenêtre de congestion et la fenêtre de bout-en-bout. La fenêtre de congestion est une sous-fenêtre de la fenêtre de bout en bout. Bien que le changement de la taille de la fenêtre de congestion se fasse en fonction du taux d'arrivée des paquets LACK de la part du prochain proxy, la taille de la fenêtre de bout-en-bout va changer en fonction du débit d'arrivée de bout-en-bout des accusés de réception ACK de la destination. A chaque proxy, il y aura une fenêtre de congestion qui gère le taux d'envoi entre les proxys.

Les résultats des simulations indiquent qu'une distance inter-proxy entre 3 et 5 a un bon impact à la fois sur le débit et la répartition. Les auteurs rapportent qu'une amélioration allant jusqu'à 30% peut être obtenue dans le débit total à l'aide de Split TCP. Les inconvénients sont des grands tampons et une surcharge du réseau. En outre, cette proposition rend le rôle de nœuds proxy plus complexes car ils ont à contrôler la livraison des paquets des proxys suivants pour chaque session TCP.

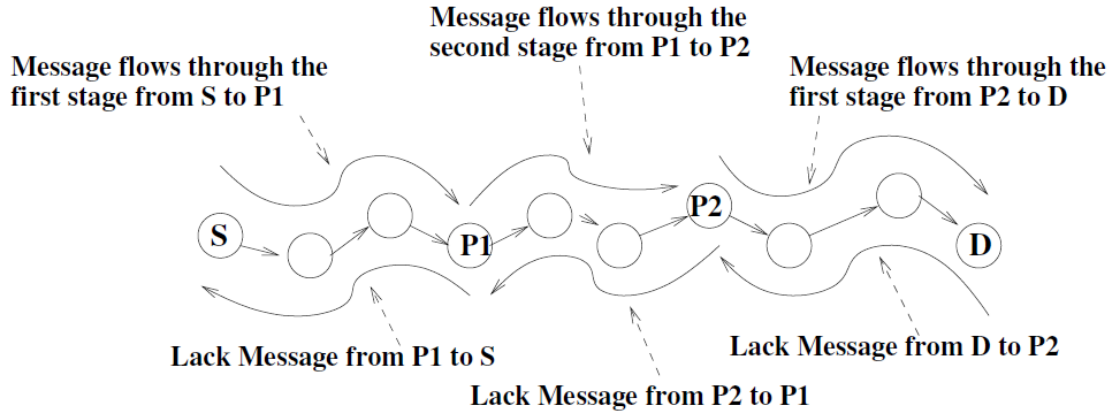


Figure 20 : TCP-Split

6.5.1.6. Gestion de lien basé sur la puissance du signal [26]

Dans cet algorithme chaque nœud conserve un enregistrement des puissances de signaux reçus de nœuds voisins 1-hop. En utilisant ces enregistrements, le protocole de routage prédit immédiatement les échecs de route ; cette prédiction est appelé Proactive Link Management. Lors de la détection de cet événement, l'agent de routage de la source est averti par un message Going-Down (cf. Figure 21) à la réception de ce message de l'agent de routage de la source arrête d'envoyer des paquets et initie une procédure de recherche de route. La nouveauté de cette proposition est le mécanisme réactif du Link Management. Ce mécanisme augmente la puissance de transmission recherchant de nouvelles routes à chaque fois qu'une route subit des erreurs de route. Les mécanismes de gestion de liens réactifs et proactifs peuvent être couplés de la manière suivante : Suite à la prévision sur la panne du lien, l'agent de routage du nœud informe la source pour arrêter l'envoi, ce nœud augmente sa puissance d'émission pour gérer les paquets en transit qui utilisent ce lien.

Les auteurs utilisent des simulations sur des scénarios de réseaux faiblement chargés. Les résultats montrent que leur approche donne 45% d'amélioration sur les performances du protocole TCP.

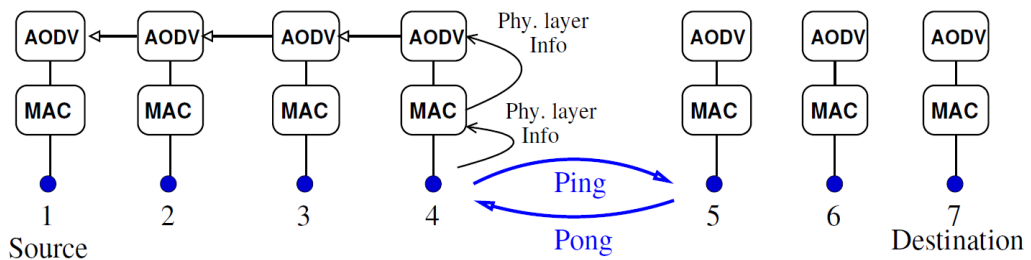


Figure 21 : Inter-couches réseau et physique.

6.5.1.7. COPAS [63]

L'approche **C**ontention-based **P**ath **S**election aborde le problème de baisse de la performance de TCP en raison des concurrences d'accès sur un canal sans fil. Elle met en œuvre deux techniques : la première est la transmission disjointe et la deuxième est la route retour, qui consiste à choisir des itinéraires disjoints pour les données TCP et les paquets TCP ACK. Le second est l'équilibrage-contentieux dynamique, qui consiste à mettre à jour dynamiquement les itinéraires disjoints. Une fois la contention d'une route dépasse un certain seuil, appelé seuil de temporisation, une nouvelle route moins contentieuse est choisie pour remplacer la route contentieuse.

Dans cette proposition, la contention sur le canal sans fil est mesurée en fonction du nombre de fois qu'un nœud a reculé au cours de chaque intervalle de temps. Aussi à tout moment une route peut être brisée, en plus de lancer une procédure de rétablissement d'itinéraire, COPAS redirige les paquets TCP en utilisant le deuxième itinéraire alternatif. En comparant COPAS et DSR, les auteurs ont constaté que COPAS surpasse DSR en termes de débit TCP et des frais généraux de routage (ressources). L'amélioration du débit TCP peut atteindre jusqu'à 90%. Cependant, l'utilisation de COPAS, tel que rapportée par les auteurs, est limitée aux réseaux statiques ou des réseaux à faible mobilité. Quand les nœuds se déplacent plus rapidement en utilisant une transmission disjointe et des routes retours, ceci augmente la probabilité de défaillance de route rencontrée par les connexions TCP.

6.5.2. Les solutions basées sur la couche Transport

6.5.2.1. Fixed RTO [23]

Cette technique est une technique basée expéditeur, elle ne se base pas sur les réponses retour d'état du réseau. Les auteurs utilisent une heuristique afin de faire la différence entre les échecs de route et la congestion du réseau. Lorsque deux timeouts qui se suivent, ce qui correspond à la situation dans laquelle l'ACK manquant n'est pas reçu avant l'expiration du deuxième RTO, l'expéditeur conclut qu'un échec de route s'est produit. Le paquet de données est retransmis à nouveau mais sans doublé la valeur du RTO (reste la même valeur que celle de l'envoi précédent). Cette approche fonctionne avec le protocole TCP standard, où un algorithme de réduction de puissance «exponentielle» est utilisé. Le RTO reste fixe jusqu'à ce que la route soit rétablie et que la réception du paquet retransmis est accusée.

Les auteurs ont évalué cette proposition en utilisant différents protocoles de routage ainsi que différents scénarios TCP sélectifs et l'accusé de réception retardé. Ils signalent que des améliorations significatives ont été obtenues lors de l'utilisation fixe-RTO avec les protocoles de routage à la demande. Néanmoins, comme indiqué par les auteurs eux-mêmes, cette proposition est limitée à des réseaux sans fil uniquement, une limitation sérieuse depuis l'interopérabilité avec les réseaux câblés est nécessaire. En outre, la

supposition que deux timeout consécutifs sont les résultats exclusifs de défaillances d'itinéraire a besoin de plus de l'analyse, en particulier en cas de congestion.

6.5.2.2. TCP DOOR [24]

Cette approche est la solution de bout en bout qui ne nécessite pas la coopération de nœuds intermédiaires ; elle est basée sur des événements Out-Of-Order (OOO). Les événements (OOO) sont interprétés comme une indication d'échec de route. La détection d'un événement OOO est effectuée soit par un mécanisme basé sur expéditeur ou bien d'un mécanisme basé sur le récepteur. Le mécanisme basé expéditeur utilise la propriété de fonction croissante du numéro de séquence ACK pour détecter les événements OOO. Dans le cas de paquets ACK dupliqués, ces ACKs auront le même numéro de séquence, mais l'émetteur a besoin d'informations supplémentaires pour détecter les cas d'OOO. Cette information est une option d'un octet ajouté aux ACKs appelé ACK de duplication de numéro de séquence (ADSN). L'ADSN est incrémenté et transmis avec chaque ACK dupliqué. Cependant, le mécanisme basé sur le récepteur a besoin d'une option TCP supplémentaire sur deux octets pour détecter les événements OOO, appelé TCP Packet-Sequence-Number (TPSN). Le TPSN est incrémenté et transmis avec chaque paquet TCP y compris les paquets retransmis. Si le récepteur détecte un événement OOO, il devrait en informer l'expéditeur en fixant un bit d'option spécifique, appelé bit de OOO, dans l'en-tête de paquet ACK.

Une fois que l'expéditeur TCP détecte un événement OOO, il prend les deux mesures d'actions suivantes : désactiver temporairement le contrôle de congestion et le recouvrement rapide lors de l'évitement de congestion. Dans le premier cas, l'expéditeur TCP désactive l'algorithme de congestion pour une période de temps spécifique (T1), la prochaine action est la suivante : si l'algorithme de contrôle de congestion a été invoqué au cours de la période de temps passé (T2), l'expéditeur TCP doit revenir immédiatement à l'état d'avant l'invocation du contrôle de congestion. Les auteurs font des périodes T1 et T2 comme fonction du RTT (round trip time).

Durant la simulation de cette approche, différents scénarios ont été envisagés en combinant tous les mécanismes et les actions mentionnées ci-dessus. Leurs résultats montrent que les mécanismes basés expéditeur/récepteur se comportent de façon similaire. Ainsi, ils recommandent l'utilisation du mécanisme de détection de l'expéditeur car il ne nécessite pas de notification de la part de l'expéditeur pour le destinataire. En ce qui concerne les actions, ils ont comme rôle de désactiver temporairement le contrôle de congestion et le recouvrement rapide lors de l'évitement de congestion et le fait qu'ils sont utilisés dans la détection d'événement OOO, ils ont constaté que les deux actions conduisent à une amélioration significative. En général, le protocole TCP-DOOR améliore le rendement de 50%. Néanmoins, la supposition que les événements OOO sont forcément les résultats de l'échec de route, mérite beaucoup plus d'analyse. Les protocoles de routage

multi-route tels que TORA peuvent produire des événements OOO qui ne sont pas liés à des défaillances de route.

Le tableau suivant récapitule les approches vues en mentionnant une comparaison entre ces approches.

Approches	Type de solution	Degrée de complexité	Débit	Type de réseau	Nombre de connexions	Evaluation
TCP-F	Inter-couches	Moyenne	Faible	Mobile aléatoire	Une seule	Simulation
ELFN	Inter-couches	Moyenne	Faible	Mobile aléatoire	Une seule	Simulation
ATCP	Inter-couches	Haute	Faible	Mobile aléatoire	Une seule	Emulation
TCP-Bus	Inter-couches	Haute	Moyen	Mobile aléatoire	Multiple	Simulation
Split-TCP	Inter-couches	Moyenne	Moyen	Mobile aléatoire	Une seule	Simulation
TCP avec puissance de signal	Inter-couches	Haute	Faible	Mobile aléatoire	Deux	Simulation
COPAS	Couches réseau	Moyenne	Haut	Statique	Multiple	Simulation
Fixed-RTO	Couches TCP	Faible	Faible	Mobile aléatoire	Une seule	Simulation
TCP DOOR	Couches TCP	Faible	Moyen	Mobile aléatoire	Une seule	Simulation

Table 1 : Comparaison générale des approches

6.6. Conclusion

Après étude et analyse des différents approches qui essaient d'améliorer le comportement et les performances du protocole TCP dans les réseaux sans fil, nous avons constaté qu'il y a plusieurs façons et techniques qui peuvent être utilisées et qui donnent de bons résultats et performances. Cependant la plupart de ces solutions sont des solutions inter-couches, ceci limite l'implémentation de ces solutions sur des simulations. Il est impossible pour le moment d'émuler n'importe quelle solution cross-layer dans une expérimentation sur un cas réel. De plus, la plupart des approches proposées donnent de bons résultats dans des scénarios bien précis (faible mobilité, débit faible,) alors que le cas réel fait référence à des contraintes qui sont beaucoup plus rigides.

7. Impact du protocole TCP sur les services vidéo streaming adaptatifs

7.1. TCP et les services de vidéo streaming adaptatif

Les services de vidéo streaming adaptatifs se basent sur un algorithme d'adaptations vidéo/débit ; c'est une méthode qui change la qualité de la vidéo reçue en fonction des conditions du réseau. Généralement, les services de vidéo streaming adaptatifs sur Internet fonctionnent sur les réseaux non gérés. Les technologies de vidéo streaming envoient le contenu vidéo du serveur vers le client en utilisant le protocole HTTP standard à travers le protocole de transport (TCP). Le protocole HTTP présente quelques avantages qui permettent un accès universel, le partage de connexion pour de nombreux appareils, la fiabilité, la convergence mobile-fixe, la robustesse...etc. Le flux streaming est géré par un agent adaptatif. Cet agent se base sur des informations vitales du réseau telles que les conditions du réseau, les paramètres TCP...etc.

7.2. Protocoles de transport dédiés au streaming

De nombreux protocoles ont évolué pour assurer la transmission multimédia sur Internet. Ils appartiennent à différentes couches du modèle OSI de la pile de protocoles et fournissent différents types de services aux applications.

Dans les réseaux basés sur IP, deux protocoles de transfert de données sont principalement utilisés : TCP et UDP. À l'opposé du protocole UDP, qui introduit des pertes ou une arrivée désordonnée des paquets, le protocole TCP garantit la livraison des données en les expédiant à nouveau si nécessaire et effectue un contrôle de congestion lors de la transmission des données.

Un contenu audiovisuel peut tolérer la perte d'un certain pourcentage de ses données, sans pour autant affecter la qualité perçue par le spectateur, ceci sous réserve que les données perdues ne soient pas consécutives. Ainsi, la fiabilité de livraison des données dynamiques, à l'opposé des données statiques, est souvent inutile. D'ailleurs, il ne sert à rien de retransmettre un paquet perdu si le paquet retransmis arrive lorsque la séquence à laquelle il appartient a déjà été utilisée. Non seulement la retransmission est inutile mais elle consomme des ressources, au risque de retarder d'autres paquets.

Cependant, un inconvénient majeur du protocole UDP est qu'il n'effectue pas de contrôle de congestion d'une manière compatible avec TCP. En effet, il continue à augmenter le débit de transfert jusqu'à saturation du réseau. Il ne partage pas équitablement la bande passante avec les trafics bâtis sur TCP. Ceci conduit à l'effondrement du réseau en privant toutes les autres applications de bande passante (tout protocole confondu).

Le protocole de transmission temps réel RTP est un protocole optimisé pour le transfert de données dynamiques sur un réseau point à point tout en respectant leurs contraintes de temps réel. Il est également utilisé pour la multidiffusion. Ce protocole

effectue un découpage intelligent des données pour que chaque paquet soit décodable indépendamment des autres. Les paquets RTP sont marqués temporellement de manière à être réorganisés, en cas de réception désordonnée ou tardive, afin d'afficher le stream de manière cohérente et synchronisée chez l'utilisateur. Ces marques temporelles permettent également la détection d'éventuelles pertes de paquets. Cependant, RTP ne gère pas la réservation de ressources réseau et ne garantit pas de qualité du service en temps réel. On est lui associe le protocole RTSP (couche session), pour effectuer le contrôle des données.

Dans le cas du streaming, le contrôle de la congestion revient à superviser le débit. Le contrôle du débit (rate control) peut être géré au niveau de l'émetteur (serveur), du récepteur (client), ou des deux (hybride). Un trafic est considéré compatible avec TCP, ou « TCP-friendly », s'il ne réduit pas, à long terme, le débit d'autres trafics plus que « sa version TCP » ne le réduirait dans les mêmes conditions.

Malgré les propositions de protocoles faites au niveau applicatif, la majorité des applications utilisent toujours TCP ou UDP. L'explication revient au fait que la couche transport est implémentée au niveau des systèmes d'exploitation qui se sont limités, dans le passé, aux protocoles TCP et UDP.

Deux combinaisons de protocoles les plus largement utilisés sont HyperText Transfer Protocol / Transmission Control Protocol (HTTP / TCP) et Real Time Protocol / User Datagram Protocol (RTP / UDP). Chacune de ces combinaisons fournit différents types de services de streaming.

1) Le streaming basé sur RTP / UDP utilise le protocole RTP / RTSP et UDP, le protocole de la couche Transport qui propose un contrôle physique du réseau disponible [93]. RTSP est utilisé en extension de HTTP pour supporter le déplacement fluide dans le flux pendant sa lecture.

La combinaison RTP / UDP prend en charge les transmissions multicast et broadcast. Même si elle fonctionne bien dans les réseaux managés, elle a cependant ses propres inconvénients, à savoir :

- RTP exige beaucoup de ressources car il gère chaque session de streaming sur l'expéditeur.
- les paquets RTP sont bloqués par les NAT [113] et ne peuvent pas passer à travers l'Internet actuel.

2) Le streaming sur HTTP via TCP permet de traverser les pare-feu du réseau et d'exploiter l'utilisation commune et l'implémentation du protocole HTTP.

Dans l'Internet d'aujourd'hui, RTP / RTCP est limitée par rapport au déploiement à grande échelle. En revanche, le streaming basé sur HTTP / TCP est devenu populaire grâce déjà à sa facilité de déploiement dans l'Internet actuel. Bien qu'il ne supporte que la transmission unicast et les médias ne sont délivrés que dans de grands segments, le

streaming sur HTTP / TCP offre néanmoins de nombreux avantages par rapport à RTP / RTCP

Il simplifie la connectivité, puisqu'un flux / TCP HTTP peut traverser les pare-feu de réseau et Network Address Translators (NAT) [93]. Aussi, la combinaison de protocoles HTTP / TCP en streaming peut être gérée sans la nécessité de maintenir un état de session sur le serveur, améliorant ainsi l'évolutivité du système. De cette manière, un déploiement à grande échelle d'un système de diffusion est rendue possible par l'utilisation de l'infrastructure Web déjà disponibles, y compris les serveurs HTTP, les proxys et les caches.

7.3. Impact des paramètres de TCP sur la qualité de service des services de vidéo streaming adaptatifs

Les paramètres TCP ont un impact significatif sur la communication entre le client et le serveur, en particulier dans le transport des données de vidéo streaming. L'analyse de la vidéo en streaming basé-TCP montre que pour avoir un bon transport de données le débit TCP devrait être le double par rapport au débit de la vidéo, et ce pour garantir de bonnes performances du service vidéo Streaming. Les services de vidéo streaming adaptive essaie de surmonter ce problème, ils adaptent la qualité vidéo en fonction de la bande passante réseau disponible. La bande passante du réseau influence directement le processus de sélection de la qualité de la vidéo. A noter aussi que la valeur de la bande passante du réseau calculée dépend des paramètres du TCP, la formule 2.6 montre que la bande passante théorique calculée en fonction de la taille de la fenêtre coulissante du TCP et le temps aller-retour d'un paquet RTT (round trip time). Ces deux variables font partie des paramètres du TCP et n'importe quel changement sur ces paramètres de TCP implique un changement sur la bande passante du réseau, ce qui influence par la suite, dans le cas des services de vidéo streaming adaptatifs, la qualité vidéo reçu par le client de la part du serveur.

$$\text{Bande-passante (théorique)} = \text{Taille de la fenêtre TCP} / \text{RTT} \quad 2.6$$

7.4. L'enjeu de la qualité d'expérience sur les services de vidéo streaming adaptatif

De nos jours, des sessions de vidéo streaming contribuent à améliorer notre expérience de vie et seront encore plus présents dans nos activités professionnelles et personnelles dans les futures générations de réseau. La distribution de ces sessions pour les utilisateurs fixes et mobiles avec des supports de qualité de service (QoS) et de Qualité d'Expérience (QoE) est une question clé pour attirer et garder les clients, tout en augmentant les profits des fournisseurs et en optimisant les ressources du réseau. Du point de vue du réseau, ce défi est principalement due à l'absence de techniques de contrôle de distribution de paquets efficaces et le comportement dynamique des ressources filaires et

sans fil partagés. Du point de vue de l'utilisateur, des sessions de vidéo streaming équipé d'un soutien de QoE doivent être accessibles n'importe où et à tout moment.

Les systèmes traditionnels basés QoS, tels que ceux à services différenciés fondés sur l'architecture DiffServ [RFC2998] et les services intégrés qui s'appuient sur l'architecture IntServ [RFC2998], visent à assurer un bon niveau de qualité de sessions vidéo dans un environnement réseau où cette qualité est basée sur un ensemble d'opérations de mesure et de contrôle réseau. Actuellement il existe des approches qui fournissent des garanties de qualité de service pour les sessions de vidéo streaming en fonction des mesures basées sur des métriques réseau/paquets. Des exemples de ces mesures sont le débit, la perte de paquets, le délai et la gigue, sauf que ces mesures n'indiquent pas l'impact réel de la qualité vidéo reçu du point de vue de l'utilisateur. Par conséquent, on conclut que les paramètres de qualité de service existants ne parviennent pas à capturer les aspects subjectifs associés avec le système visuel humain (HVS).

Afin d'optimiser l'utilisation des ressources du réseau et augmenter le niveau de qualité de la vidéo, les systèmes de contrôle de qualité de service QoS doivent tenir compte des conditions actuelles du réseau, les caractéristiques de la vidéo et l'apport en QoE.

7.5. Technique d'adaptation du réseau

La conception des techniques d'adaptation de réseau pour la transmission de la vidéo sur Internet pose de nombreux défis. Dans cette section, nous présentons des solutions basées sur la couche de transport. En particulier, nous résumons les travaux de recherche existants en les classant selon leurs types.

Trois catégories de contrôle de débit existent : Contrôle basé sur la source, contrôle basé sur le récepteur et contrôle hybride. Nous terminons cette section en résumant, évaluant et comparant ces techniques.

7.5.1. Contrôle de débit basé source

Dans cette approche, la source (expéditeur) est chargée d'adapter le débit de transmission vidéo. Des mécanismes de contrôle de débit basées sur la source fournissent des informations sur l'état du réseau pour l'expéditeur afin qu'il puisse régler et adapter le débit du flux vidéo. Le contrôle du débit basé sur la source peut être implicite pour les deux approches unicast et multicast.

En ce qui concerne la transmission unicast, elle concerne deux approches existantes, l'une basée sur le sondage et l'autre basée sur équation. La première approche est basée sur le sondage d'expériences. Le débit d'envoi est régulé par les stratégies d'augmentation additive et de diminution multiplicative (AIMD) ou par l'augmentation multiplicative et la diminution multiplicative (MIMD). L'augmentation ou la diminution du débit dépend du

seuil du taux de perte de paquets. Sur l'approche basée équation, le débit d'émission est calculé par une équation de débit qui prend en considération l'état de la route de transmission dans le réseau.

En ce qui concerne la transmission multicast, l'expéditeur utilise un seul canal multidiffusion où seul le contrôle de débit en fonction de sonde peut être utilisé. La multidiffusion monocanal est efficace puisque tous les récepteurs partagent le même et seul canal. Si la vidéo multicast devait être livrée par le biais des flux unicast individuelles, l'efficacité de la bande passante serait faible, mais les services pourraient être différenciés puisque chaque récepteur peut négocier avec la source les paramètres des services.

7.5.1.1. Contrôle de débit avec TCP

Le mécanisme de contrôle de congestion TCP est en lui-même l'approche de contrôle de débit basé-sondage la plus répandue.

7.5.1.2. Contrôle de débit avec le protocole de transport multimédia (MTP) [2]

Le protocole de transport MTP est une modification du protocole TCP. C'est une solution basée-sondage qui désactive gracieusement les retransmissions tout en préservant les horloges de transmission ainsi que les caractéristiques du mécanisme de congestion du TCP. Tout comme le protocole TCP, MTP effectue le démarrage lent, l'évitement de congestion, la retransmission rapide et le recouvrement rapide. Il offre cependant une API transparente UDP-like qui permet aux applications média streaming de prendre des décisions de mise à l'échelle des médias informés et fournit une sémantique de livraison de paquets UDP. En outre MTP n'offre pas une garantie de livraison des paquets dans le bon ordre.

A la base, le protocole MTP conserve les mêmes mécanismes de détection de perte de paquets et de recouvrement de pertes que ceux du protocole TCP, mais réduit les caractéristiques de grand délais et gigue. Quand l'émetteur MTP reçoit des accusés de réception dupliqués, il lance le mécanisme d'évitement de congestion, de retransmission rapide et de recouvrement rapide de la même façon que le protocole TCP, et donne un mouvement de la fenêtre de congestion ainsi qu'une horloge de transmission de paquets identiques à ceux du protocole TCP. A la réception d'un troisième accusé de réception ACK dupliqué, au lieu d'entrer en phase de retransmission, le protocole MTP active l'expéditeur MTP qui augmente sa fenêtre de transmission et envoie un nouveau paquet. Quand il reçoit un accusé de réception pour ce nouveau paquet, il diminue sa fenêtre de transmission. Ceci est fait afin de ne pas considérer le remplacement/retransmission de paquets en tant que nouvelle transmission au moment de la prochaine décision de transmission de paquets, tout en conservant le même comportement de transmission que celui de TCP au niveau de la couche réseau.

Dans le cas d'un timeout de transmission, l'expéditeur MTP réagit de la même manière qu'un expéditeur TCP en effectuant un démarrage lent sauf que, contrairement à TCP qui redémarre la transmission depuis le dernier numéro de séquence de paquet ACK reçu, MTP redémarre en transmettant un nouveau paquet directement.

Les résultats de la simulation en [2], ont montré que les flux vidéo MTP héritent des bonnes caractéristiques du flux TCP et UDP. Les résultats montrent aussi que les flux MTP ressemblent également aux flux de TCP-Friendly en plus du fait qu'ils peuvent adapter le débit de TCP-Friendly.

7.5.1.3. Protocole de transport vidéo (VTP) [3]

Le protocole de transport VTP est basé sur le sondage qui offre un nouveau mécanisme de contrôle de débit de bout en bout. Ce mécanisme a été conçu spécifiquement pour le streaming en temps réel dans les réseaux sans fil. Son premier but était de bien fonctionner dans l'Internet par câble, d'être robuste aux erreurs aléatoires, afin de fournir une bonne convivialité avec le protocole TCP ainsi que d'être déployé avec succès sur les liaisons sans fil.

VTP combine l'estimation du taux avec des techniques de discrimination de perte. Il se compose de deux éléments importants, à savoir le taux d'estimation atteint (AR) et l'algorithme de discrimination de perte (LDA). Le taux AR est le débit avec lequel l'expéditeur a envoyé ces paquets avec succès à travers le goulot d'étranglement. Plus spécifiquement, c'est le débit que le récepteur peut mesurer, ajouté à la fraction correspondante à la perte de paquets à la sortie du goulot d'étranglement dû aux erreurs aléatoires. Le LDA permet au protocole VTP de distinguer les pertes de congestion des pertes d'erreurs de route. Le protocole VTP mesure l'AR et adapte son débit d'émission lorsqu'une congestion est détectée par le LDA.

Le contrôle de débit avec VTP est basé sur l'analyse du débit d'envoi instantanée du protocole TCP. Comme TCP, VTP augmente linéairement la bande passante disponible jusqu'à ce qu'une congestion soit détectée. Au cours d'une phase de congestion et en imitant le comportement de TCP sans impliquer une décroissance multiplicative, l'expéditeur VTP réduit son débit d'envoi et le taux d'encodage vidéo jusqu'à ce que la congestion s'arrête. Cela permet au protocole VTP de transmettre une plus grande partie de flux vidéo.

Le protocole VTP est un protocole de bout-en-bout qui fonctionne bien dans l'environnement sans fil sans le soutien des couches inférieures ainsi que les mécanismes AQM (analyse quantifiée de la marche) [3]. Il vise à être adaptable et flexible en faisant des hypothèses minimales sur le réseau et en utilisant des évaluations de réseau comme un indicateur approximatif. Ces caractéristiques sont les facteurs de motivation de la conception du protocole VTP.

7.5.1.4. Source-Adaptive Multilayered Multicast Algorithms (SAMM) [4]

Dans les algorithmes SAMM qui ont été proposés pour la transmission multicast de vidéos, la source ajuste les paramètres de codage, en incluant le nombre de vidéos et leurs débits respectifs dans une réponse retour (feedback) de congestion sur un flux continu de la part des récepteurs. Les algorithmes SAMM peuvent être basés sur le concept de bout-en-bout ou basé-réseau. Sur les algorithmes basés-réseau, la congestion est surveillée et détectée par les nœuds intermédiaires de réseau. D'autre part, sur les algorithmes de bout-en-bout, le contrôle de congestion se fait exclusivement chez la source et chez les récepteurs.

Afin de renforcer les algorithmes SAMM, une architecture de réseau doit être définie. Quatre éléments de base sont nécessaires pour la mise en œuvre d'un algorithme vidéo SAMM qui sont : une source vidéo adaptative de plusieurs niveaux, des récepteurs vidéo de plusieurs niveaux, des routeurs multicast et des nœuds avec la capacité de renvoyer des réponses retours (feedback). La source vidéo adaptative a la capacité de générer des données vidéo de plusieurs niveaux de qualité. Le récepteur fait référence à la qualité qu'il veut recevoir. Les routeurs multicast doivent être capables d'effectuer le transfert multicast, le routage, la diminution de priorité, l'isolation de flux et le contrôle de congestion. Pour éviter l'implosion, une agrégation des réponses retour est faite par la fusion et l'organisation des réponses retour distribuées dans le réseau superposé. Après la réception d'un nombre donné de paquets vidéo, le récepteur envoie une réponse retour au nœud de fusion le plus proche. Celui-ci transmettra à la source la liste de débits demandés par les récepteurs.

7.5.1.5. RTP/RTCP (Real Time Protocol) [5]

Tous les protocoles et les algorithmes ci-dessus se réfèrent à la couche Transport, à part Real Time Protocol (RTP) qui est un protocole Internet standard se trouvant sur la couche Application. Il est utilisé pour le transport des données en temps réel de bout en bout, y compris l'audio et la vidéo. Il peut aussi être utilisé pour les services de média à la demande ainsi que les services interactifs comme la téléphonie sur Internet et les communications unicast.

Le protocole RTP fonctionne habituellement sur le dessus du protocole UDP afin de faire usage de ses fonctions de multiplexage et de contrôle. RTP ne garantit pas une livraison rapide de paquets et ne fournit pas les paquets dans leur ordre. Le protocole RTP donne la responsabilité de récupérer les segments perdus et le séquençage des paquets à la couche Application. Par conséquent, il ne propose aucune formule de fiabilité ou de contrôle de débit/congestion. Il offre un service d'horodatage ainsi que des numéros de séquence pour assurer la fiabilité et le contrôle débit/congestion, mais leur mise en œuvre est totalement attribuée à l'application.

Le Real Time Control Protocol (RTCP) accompagne le protocole RTP. Il sert à faire des statistiques sur les connexions et les informations multimédias telles que le nombre d'octets envoyés, le nombre de paquets envoyés, le nombre de paquets perdus, la gigue et les délais aller-retour. Les expéditeurs, les récepteurs et les moniteurs tiers peuvent utiliser ces informations pour juger la qualité de leurs connexions afin de procéder aux ajustements nécessaires telles que le passage d'un faible codec de compression à un codec de compression plus élevé. Les gestionnaires de réseau peuvent utiliser des informations contenues dans les paquets RTCP pour évaluer la performance de leurs réseaux dans la diffusion multicast. RTCP est utilisé pour les conférences en temps réel (grands groupes) utilisant le réseau Internet, y compris l'identification de la source, le soutien pour les passerelles et les traducteurs multicast-à-unicast. Il propose également des évaluations de qualité de service à partir de récepteurs au groupe de multi diffusion. Aussi il sert comme un support pour la synchronisation des différents flux de médias. Les données RTP / RTCP peuvent être cryptées afin d'améliorer la sécurité.

Le protocole RTP / RTCP n'est pas responsable des tâches de niveau supérieur comme l'assemblage ou la synchronisation qui doivent être fait dans la couche Application. Il fournit des mécanismes et des fonctionnalités de contrôle nécessaires à l'acheminement du contenu en temps réel. RTP est un framework de protocole qui n'est délibérément pas complet. Il est ouvert pour être utilisé sur de nouveaux formats de données ainsi que pour de nouveaux logiciels multimédia. En ajoutant de nouveaux profils et de nouvelles spécifications de formats de données, on peut adapter le protocole RTP à de nouveaux formats de données et à de nouvelles applications.

7.5.1.6. Contrôle de débit avec TCP Friendly (TFRC) [6]

Le mécanisme de contrôle de débit de TCP friendly est l'un des algorithmes de streaming de bout-en-bout basés équation les plus répandus et souvent utilisé comme référence. TFRC a pour rôle de servir comme 'framework' de contrôle de congestion pour toutes les applications qui ne nécessitent pas le fonctionnement à 100% du mécanisme de fiabilité du protocole TCP et qui profitent de la faible variation du débit d'émission.

Dans le protocole TFRC, l'émetteur envoie un flux de paquets de données vers le récepteur avec une vitesse contrôlée par la formule suivante :

$$X = \frac{s}{R * \sqrt{\frac{2*b*p}{3}} + (t_RTO * (3 * \sqrt{\frac{3*b*p}{8}}) * p * (1 + 32 * p^2))} \quad 2.7$$

Où X est le débit de transmission en octets / seconde, s est la taille du paquet en octets, R est le temps d'aller-retour (RTT) en secondes, p est la probabilité de perte de paquets (entre 0 et 1), t_RTO est la valeur du retransmission timeout de TCP en secondes et b est le nombre de paquets notifiés par un seul accusé de réception TCP.

Quand un paquet feedback est reçu par le récepteur, l'expéditeur change son taux d'envoi sur la base des informations contenues dans la réponse retour. Si l'expéditeur ne reçoit pas de feedback durant deux temps aller-retour (RTT), il abaisse son taux d'envoi de moitié. Ce résultat est obtenu au moyen d'une minuterie qui est appelé la minuterie de non-feedback. Quand un paquet feedback est reçu, les actions suivantes doivent être effectuées : calculer un nouvel échantillon du temps d'aller-retour, mettre à jour la valeur estimée du temps aller-retour, mettre à jour l'intervalle de temporisation, mettre à jour le débit d'émission et réinitialiser la minuterie de non-feedback.

Le récepteur envoie périodiquement des messages de réponse retour à l'expéditeur. Les paquets feedback devraient normalement être envoyés au moins une fois par RTT. Ils devraient également être envoyés à chaque fois qu'un nouvel événement de perte est détecté sans attendre la fin d'un RTT, et à chaque fois qu'un paquet out-of-order qui supprime un événement de perte de l'historique des pertes est reçu. Si l'expéditeur transmet de nombreux paquets par RTT, il peut y avoir certains avantages si l'envoi de messages feedback se feront plus d'une fois par RTT car cela permet d'avoir une réponse plus rapide qui servira à l'évolution des mesures de RTT. Cependant, l'envoi d'un grand nombre de paquets feedback par RTT pourra surcharger le réseau.

Le protocole TFRC a été conçu pour répondre à un événement de perte paquet. Cependant, avec la popularité croissante de terminaux Internet sans fil et la grande demande qui concerne le flux streaming vidéo par les utilisateurs mobiles, il est important pour les protocoles de streaming de travailler d'une façon efficace sur les liens sans fil en supportant les erreurs aléatoires élevées du sans fil.

7.5.2. Contrôle de débit basé récepteur

La technique d'adaptation de débit basée source souffre en matière de performance en raison des différents besoins des récepteurs en matière de bande passante. L'hétérogénéité de l'environnement réseau empêche la source de transmettre les données avec un seul débit. En dehors de cela il n'y a pas de techniques vraiment acceptables pour déterminer la capacité du réseau et la façon qui permet de trouver le débit approprié. Différentes approches utilisant le contrôle de débit basé récepteur ont été proposées. La plus populaire de ces approches est celle qui combine un algorithme de compression multicouche avec un système de transmission multicast. Les récepteurs définissent implicitement les arbres de distribution de multidiffusion simplement en exprimant leur intérêt à recevoir des flux bien définie. Ainsi, il n'y a aucune signalisation explicite entre les récepteurs et les routeurs ou entre les récepteurs et la source.

Comme dans le contrôle de débit basé source, nous classons les protocoles et les mécanismes de contrôle de débit basés récepteur existants en deux catégories : les approches basées sur le sondage et les approches basées sur des équations. Dans l'approche basée sonde le récepteur choisit le niveau optimal de souscription et ajoute des couches en joignant les groupes de multidiffusion correspondants jusqu'à ce que la

congestion se produise. Ensuite, il recule d'un point de fonctionnement afin de soulager le goulot d'étranglement. D'autre part, dans l'approche basée équation du récepteur, il estime la bande passante disponible puis décide d'adhérer ou de disjoindre un groupe de multidiffusion.

Les solutions basées récepteur peuvent être classées elles-mêmes en deux catégories : Des solutions agnostiques et non agnostiques. Du côté récepteur, les solutions de streaming adaptatif qui adaptent le débit peuvent soit interagir avec les utilisateurs et donc entrer dans la classe agnostique soit fonctionner dépendamment des utilisateurs et entrer dans la classe non agnostique.

7.5.2.1. Techniques non agnostiques

7.5.2.1.1. Receiver-driven Layered Multicast (RLM) [7]

Le protocole RLM fonctionne dans le modèle IP actuel. Il utilise les techniques du meilleur effort (best-effort) de livraison multipoints de paquets, l'efficacité de la livraison multidiffusion d'IP et de la communication orientée groupe. Dans le protocole RLM, la source encode son signal en plusieurs couches et transmet chaque couche sur un groupe de multidiffusion distincte. Les récepteurs peuvent rejoindre ou quitter un groupe en fonction de sa capacité : si la congestion se produit alors déposer une couche ou si une capacité de réserve supplémentaire existe alors ajoutez une couche. Le récepteur recherche le niveau optimal de souscription et selon ce schéma il ajoute des couches jusqu'à ce que la congestion se produise.

Le récepteur doit déterminer son niveau d'abonnement actuel. Si une congestion se produit alors l'abonnement est trop élevé. Cela est facile à détecter car la congestion est exprimée explicitement dans le flux de données par la perte de paquets ou par la dégradation de la qualité de transmission. D'autre part, lorsque l'abonnement est trop bas, il n'y a pas de signal équivalent. La façon de régler ce problème est d'ajouter des couches spontanément à des moments bien choisis. Si l'ajout de couches provoque la congestion du récepteur, il diminue rapidement les couches de placement. Si aucune congestion n'est détectée alors le récepteur s'approche d'un pas supplémentaire vers le point de fonctionnement optimal.

Les chercheurs ont évalué le protocole RLM sur des scénarios simples et ils n'ont considéré que l'interaction inter-RLM. Ils ont constaté que RLM peut entraîner une forte iniquité inter-RLM.

7.5.2.1.2. Layered Video Multicast Retransmission (LVMR) [8]

Le protocole LVMR s'appuie sur un système de distribution de vidéo qui utilise un codage en couches sur Internet. Il aborde les problèmes de la congestion et de l'hétérogénéité du réseau en utilisant des techniques de codage en couches de la vidéo.

Dans ce contexte, il permet à chaque récepteur de souscrire à un sous-réseau des couches de vidéo en fonction de sa puissance de traitement et de la disponibilité de la bande passante réseau. Les deux contributions clés du protocole sont les suivants :

(i) l'amélioration de la qualité de réception dans chaque couche en retransmettant les paquets perdus, détermination d'une limite supérieure de temps de récupération et l'ajout d'un point de restauration qui permettra au système de faire un maximum de retransmissions avec succès,

(ii) l'adaptation à la congestion ainsi qu'à l'hétérogénéité du réseau en utilisant un mécanisme de contrôle de débit hiérarchique.

Contrairement aux techniques de contrôle de débit existantes basées-expéditeur et basées-récepteur, pour lesquelles l'ensemble des informations sur la congestion du réseau est soit disponible à l'expéditeur (en approche basée expéditeur) ou reproduit au niveau des récepteurs (dans l'approche basée récepteur), le mécanisme hiérarchique de contrôle de débit distribue l'information entre l'expéditeur, les récepteurs et les agents du réseau. Par exemple, les récepteurs peuvent demander les paquets perdus aux voisins (récepteurs désignés DR), de telle sorte que chaque entité conserve uniquement les informations pertinentes pour elle-même. En plus de cela, l'approche hiérarchique permet de prendre des décisions intelligentes en termes de réalisation d'expériences simultanées et de choisir l'une des nombreuses expériences possibles à tout instant basée sur un minimum d'information à propos des agents du réseau.

Le protocole LVMR peut être déployé immédiatement sur le réseau, en raison du fait qu'il ne dépend d'aucun mécanisme de qualité de service ou d'autres composants dans le réseau. D'autre part le protocole LVMR montre une évolutivité limitée. Les voisins ne sont pas nécessairement tout le temps disponibles pour partager les informations entre les récepteurs vidéo.

7.5.2.1.3. 3GP Dynamic Adaptive Streaming over HTTP (3GP-DASH) [9]

Depuis la version 10, le service Adaptive 3GPP streaming HTTP est appelé 3GP-DASH après une activité de normalisation 3GPP SA4. La figure 22 montre le principe de la spécification 3GP-DASH. Cette spécification fournit :

– Une définition normative de la présentation des médias. Avec cette définition la présentation des médias est définie comme un ensemble structuré de données qui est accessible au client DASH à travers la description des présentations des médias.

– Une définition normative des formats d'un segment. Un segment est défini comme une unité de données intégrant d'une présentation multimédia qui peut être référencé uniquement par une URL HTTP (probablement restreinte par une plage d'octets).

- Une définition normative du protocole de livraison utilisé pour la livraison des segments, à savoir HTTP / 1.1.
- Une description informative sur la façon dont un client DASH peut utiliser les informations fournies pour établir un service de streaming pour l'utilisateur.

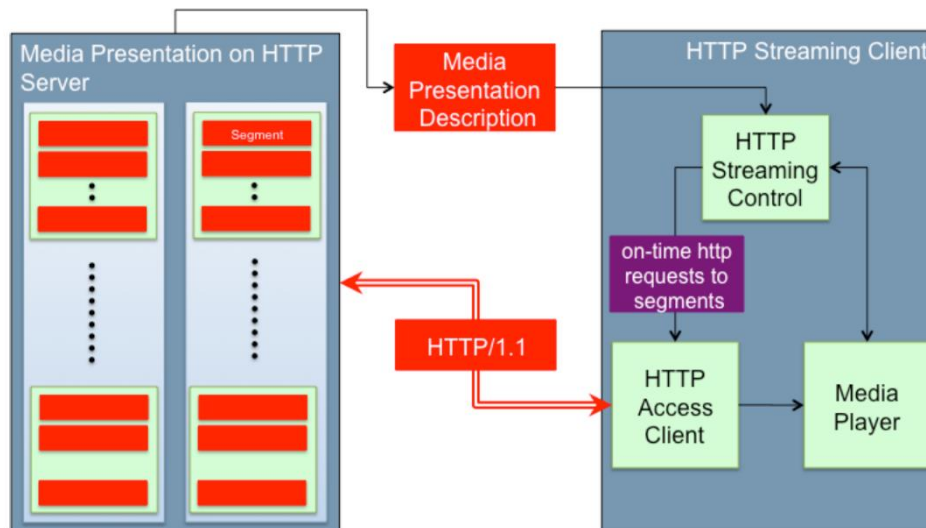


Figure 22 : Architecture 3GPP-DASH

3GPP-DASH est défini sur deux niveaux :

- a) un cadre générique pour les services streaming adaptatifs dynamiques indépendant du format d'encapsulation de données pour les segments des médias.
- b) Une instantiation spécifique de ce framework avec le format de fichier multimédia de base 3GPP / ISO en spécifiant les formats de segment, en partie en se référant aux formats TS 26,244.

Cette approche rend le framework défini par le 3GPP extensible pour tous les autres formats de segments, des codecs et des solutions DRM.

3GP-DASH prend en charge plusieurs services, entre autres :

- Le streaming à la demande.
- La télévision linéaire, y compris la diffusion de médias en direct.
- L'option « Time shift visualisation » avec les fonctionnalités réseau d'enregistrement vidéo personnel (PVR).

Des traitements spécifiques ont été ajoutés durant la conception de 3GPP-DASH, tel que le fait que le réseau peut être déployé sur des serveurs HTTP standard et que les

distributions peuvent être fournies par des infrastructures Web réguliers comme les CDN basées HTTP. La spécification laisse aussi place pour les différentes options de déploiement serveur/côté-réseau ainsi que pour les implémentations client optimisé.

La spécification définit également des dispositions pour soutenir des fonctionnalités telles que :

- La sélection initiale du client et/ou la représentation du contenu pour un utilisateur-spécifique.
- L'adaptation dynamique du contenu déjà lu afin de réagir aux changements environnementaux tels que la bande passante ou la puissance de traitement.
- Des modes, tels que la recherche, l'avance rapide ou le rembobinage.
- L'insertion de la publicité pré-codé ou d'autres contenus d'une manière simple dans les services à-la-demande et les services de streaming en direct.
- La délivrance efficace de plusieurs langues et pistes audio.
- La protection du contenu et la sécurité de transports.

7.5.2.2. Techniques agnostiques

7.5.2.2.1. Approche de l'environnement contrôlé (AEC) [10]

L'approche de l'environnement contrôlé est basée sur l'idée de l'environnement de test en laboratoire, qui est spécialement conçu pour fixer les facteurs environnementaux pouvant influencer les expériences de visualisations des utilisateurs. Télécommunications Internationale Union-télécommunications (UIT-T) [3] définit les recommandations qu'il faut tenir en compte pour lancer un bon test de laboratoire, elle décrit aussi les critères de sélection des participants qui effectuent le test. La recommandation UIT-T [11] a fourni les lignes directrices pour effectuer des essais subjectifs dans un environnement contrôlé, y compris la sélection des participants qui représentent les utilisateurs d'un service. En effet, pour obtenir une notation subjective selon la recommandation ITU-T [12], les participants devraient être des non-experts, dans le sens où ils ne devraient pas être directement concernés par l'image ou la qualité de la vidéo dans le cadre de leur travail normal. Cette approche présente les avantages suivants :

- L'environnement de test est totalement sous contrôle.
- La facilité d'observer l'influence d'un paramètre.
- La liberté de choisir des participants qui appartiennent à différentes professions, groupe d'âge, etc.

L'approche de l'environnement contrôlé a aussi quelques limitations pour évaluer la performance de la QoE, à savoir :

- Le critère de temps.
- Le nombre de participants qui sont prêts à passer du temps pour le test dans le laboratoire et exprimer leur perception de la qualité pour le service vidéo est limitée.
- Acheter les équipements spéciaux et les appareils qui permettent d'effectuer le test rend cette approche très coûteuse.
- Il est difficile de trouver une configuration de l'environnement de test laboratoire qui ressemble à l'environnement du monde réel.

7.5.3. Contrôle de débit hybride

Selon les solutions de contrôle de débit hybride, les récepteurs ajustent le débit de réception de la vidéo en ajoutant/enlevant des canaux/couches tandis que l'expéditeur en même temps ajuste le débit du canal en fonction des réponses retours (feedback) transmises par le récepteur. Ce système combine les avantages des approches basées récepteur et approches basées expéditeur afin de donner naissance à un nouveau système plus efficace, mais avec des mécanismes d'adaptation plus compliqués.

7.5.3.1. QoE-based Network Selection for Multimedia Users in IEEE 802.11 wireless networks (NSFMU) [13]

Dans l'approche citée en [13], les auteurs décrivent un schéma de sélection de réseau dans les réseaux IEEE 802.11. Les réseaux disponibles sont évalués selon la méthode PSQA [13]. Les paramètres objectifs de QoS tels que la bande passante, la gigue et la perte de paquets sont combinés avec les paramètres subjectifs de la QoE, tels que « la qualité de la vidéo est bonne ». Les paramètres de la QoE sont estimés en utilisant la technique du MOS (Mean Opinion Score ou le ressenti global). Exceptionnellement, un réseau de neurone aléatoire (RNN) est utilisé pour l'apprentissage statistique des valeurs du MOS [13] et fournit une évaluation automatisé de la QoE pour chaque réseau.

Selon les auteurs, les étapes préalables pour permettre l'application de la méthode de PSQA sont les suivantes :

Étape 1 : des facteurs de la qualité-affective et une base de données des vidéo déformés : les facteurs de la QoS sont sélectionnés, ensuite une plage de valeurs pour chaque facteur est défini. L'ensemble des facteurs ainsi que leurs valeurs spécifiques est appelé « configuration ». A la base de cela, une base de données vidéo qui contient de nombreuses configurations est créée.

Étape 2 : Une évaluation subjective de la qualité : selon la méthode MOS, une évaluation subjective des configurations qui sont contenues dans la base de données vidéo est effectuée. Deux bases de données similaires qui contiennent les configurations et les

scores relatifs au MOS sont créées. Le premier est désigné comme « base de données de formation » et le second comme « base de données de validation ».

Étape 3 : Un apprentissage du comportement de la qualité avec RNN : Le résultat de l'étape 2 est une évaluation qualitative du contenu des bases de données vidéo qui ont été créés à l'étape 1. Le RNN reçoit en entrée la base de données de formation et tente de faire un auto-apprentissage en fonction de ses données. Une fonction f vérifie si la connaissance acquise par le RNN est convergente par rapport à la base de données de validation. Si la connaissance acquise par le RNN est relativement proche de la base de données de validation, le RNN est positivement évalué, sinon un examen des données divergentes est effectué, jusqu'à l'obtention d'un bon RNN.

Quand un utilisateur a besoin de faire une sélection de réseau, chaque point d'accès l'informe à propos de sa QoE estimée. Cette information provient des expériences des précédents utilisateurs connectés au point d'accès, elle est incluse dans les balises et les trames de réponse de la sonde. L'utilisateur définit ses exigences minimales en matière de QoE en utilisant une échelle qui représente le MOS. Le terminal de l'utilisateur se connecte aux réseaux qui répondent aux exigences minimales de la QoE et effectue la sélection parmi ces réseaux.

7.5.3.2. QoE-assessment for Multiple Description video Coding [14]

Les auteurs en [14] proposent l'approche Multiple Description Coding (MDC) comme une solution pour réduire la dégradation de la qualité de la vidéo due à la perte de paquets, aux erreurs de bits et aux erreurs dues à la transmission en rafale. L'idée du MDC est de diviser le flux vidéo en deux ou plusieurs descriptions corrélées. Les descriptions contiennent des données complémentaires, de sorte que plus le nombre de description qu'on reçoit est grand, meilleure est la qualité perçue. Ainsi, la probabilité de recevoir une bonne qualité de vidéo (utile) est améliorée.

Dans cette approche (MDC), les auteurs proposent deux descriptions basées sur des groupes pairs et impairs d'images (GOP). Dans le codage H.264 [14], GOP est un groupe d'images successives. Chaque GOP contient plusieurs types de trames : I-trame (intra-image codée), P-trame (image codée prédictive) et B-trame (bidirectionnel prédictive d'image codée). Habituellement, un GOP commence avec un I-trame et sera suivi par plusieurs P-trames séparés par certaines distances de trame.

Ces distances seront occupées par les B-trames. Les I-trames contiennent une image complète et ne nécessitent aucune information supplémentaire pour la reconstruire. Par conséquent, toute erreur dans la structure GOP est corrigée par la prochaine I-trame et chaque GOP peut être décodé indépendamment sans affecter les autres GOPS. Les auteurs se basent sur ce fait pour la création des deux descriptions de flux vidéo en utilisant les GOP pairs et impairs, de façon à ce que lorsqu'un client reçoit les deux descriptions, il sera en mesure de reconstituer l'intégralité de la vidéo. En cas de réception d'une seule

description, une méthode de division est proposée afin de garantir qu'au moins, la moitié des I-frames qui sont importants soient livrés. Ainsi, on aura au moins une vidéo certes avec une faible qualité, mais toujours lisible par un observateur humain.

Le tableau suivant donne une comparaison des différentes approches de vidéo streaming :

	Agnostique/ non agnostique	Solution	Basée sonde/ équation	Basée débit/fenêtre	Unicast/ multicast	Couche transport/ application	Support sans fil	Avis client
Basée source	Non agnostique	TCP	Sonde	Fenêtre	Unicast	Transport	non	Non
		UDP	-	-	Unicast	Transport	non	Non
		MTP	Sonde	Fenêtre	Unicast	Transport	non	Non
		VTP	Sonde	Fenêtre	Unicast	Transport	oui	Non
		SAMM	-	Débit	Multicast	-	non	Non
		RTP/RTCP	-	-	Uni/multicast	Application	non	Non
		TFRC	Equation	Débit	Unicast	Application	non	Non
Basée récepteur	Non agnostique	RLM	Sonde	-	Multicast	App/trans	non	Non
		LVMR	-	-	Multicast	App/trans	non	Non
		3GP-DASH	Sonde	Débit	Unicast	Application	non	Non
	agnostique	AEC	Sonde	Fenêtre	Unicast	App/trans	non	Oui
Hybride	Non agnostique	NSMU	-	-	Unicast	App/trans	non	Non
		ADVC	-	-	Unicast	App/trans	non	Non

Table 2 : Comparaison des différentes approches de vidéo streaming.

8. Conclusion et critiques

Notre première constatation faite après les travaux présentés dans ce chapitre est que la plupart des protocoles et mécanismes arborés sont utilisés pour le streaming en temps réel. Ils ne traitent pas le problème des pertes de paquets dans les réseaux sans fil car ils ont été conçus principalement pour les réseaux câblés qui souffrent du manque d'un mécanisme de contrôle de débit pouvant gérer efficacement les pertes de paquets dans les réseaux sans fil.

Aussi, après étude et analyse des différents approches qui essaient d'adapter le débit pour améliorer la qualité de service et la qualité d'expérience dans les services de vidéo streaming, nous remarquons que plusieurs façons et techniques peuvent être utilisées et qui donnent de bons résultats et de bonnes performances. Cependant la plupart de ces solutions ne considèrent pas l'avis du client à propos du service et de la qualité de

l'adaptation apportée. Néanmoins en dépit de l'amélioration que ces approches fournissent en matière d'adaptation débit/qualité, si le client n'est pas satisfait de cette adaptation, l'approche perdra de sa valeur.

**CHAPITRE III : APPROCHES POUR
L'AMELIORATION DE TCP DANS UN
ENVIRONNEMENT SANS FIL**

CHAPITRE III : APPROCHES POUR L'AMELIORATION DE TCP DANS UN ENVIRONNEMENT SANS FIL

Sommaire

1. Introduction.....	80
2. Contributions.....	81
2.1 Contribution 1: Proposition d'un mécanisme inter-couches de différenciation.....	81
de perte pour TCP dans les réseaux sans fils ad hoc	
Conclusion 1.....	94
2.2 Contribution 2 : TCP et l'amélioration des services de vidéo streaming.....	94
adaptatifs	
Conclusion 2.....	113

1. Introduction

L'objectif principal de notre thèse est l'amélioration du protocole de contrôle de transmission (TCP) pour le transfert des flux de données en général et les flux multimédia en particulier dans un environnement sans fil. Les travaux de recherche se sont déroulés en deux étapes :

- (i) La phase une, « Proposition d'un mécanisme inter-couches pour la différenciation de perte pour TCP dans les réseaux sans fils ad hoc », consiste à apporter des améliorations au protocole TCP afin d'augmenter sa compatibilité avec les différents types de réseaux tel que les réseaux sans fils mobile (MANET) ainsi que d'améliorer ces performances en matière de bande passante optimale, de fréquence de perte de paquets et de fiabilité.
- (ii) La phase deux intitulée « TCP et l'amélioration des services de vidéo streaming adaptatifs » dans laquelle il s'agit de proposer une nouvelle méthode d'adaptation qui vise à améliorer la QoE des utilisateurs des services de vidéo streaming adaptatif. Cette amélioration est réalisée en intégrant les paramètres des terminaux et leurs feedback dans le processus d'adaptation de la qualité reçue.

Nous présentons dans ce qui suit, chacune de ces contributions en mettant l'accent sur l'amélioration apportée au niveau de TCP.

2. Contributions

2.1. Première contribution : Proposition d'un mécanisme inter-couches de différenciation de perte pour TCP dans les réseaux sans fils ad hoc

2.1.1. Positionnement du problème

TCP est le protocole de transport dominant dans les communications qui nécessitent de la fiabilité sur Internet. Il assure l'ordonnancement des paquets à leur arrivée, la retransmission des paquets perdus ainsi qu'un contrôle de congestion. Dans les réseaux sans fil, les atténuations du signal et les interférences peuvent dégrader provisoirement la qualité des liens radios. Alors que TCP a été conçu pour s'adapter aux problèmes de congestion dans les réseaux filaires, il s'agit à présent pour lui de s'adapter aux contraintes des réseaux sans fil en utilisant au mieux les ressources limitées du système.

En effet, l'hypothèse implicite de TCP que toute perte de paquets est due à une congestion n'est plus valide dans les réseaux sans fil en particulier les réseaux mobiles ad hoc, où les erreurs du canal sans fil (interférence, perte du signal, bruit,...), la contention des liens, la mobilité et le routage multi-chemin peuvent considérablement corrompre ou troubler la livraison des paquets.

Sur des liaisons filaires, la majorité des pertes de paquets sont principalement due à des problèmes de congestion dans le réseau. En plus de retransmettre les paquets perdus, TCP met naturellement en œuvre un système pour réduire la charge dans le réseau en réduisant son débit d'émission et ainsi résister à la congestion. Lorsque le réseau comporte un lien radio, des paquets peuvent être perdus du fait de la faible fiabilité de la liaison sans fil. A la suite de cette perte qui peut être due à une cause autre que la congestion, TCP déclenche de façon inopportune son système de contrôle de congestion ce qui a pour effet de réduire le débit de la liaison inutilement. Les performances de la liaison s'en trouvent de ce fait fortement dégradées.

Plusieurs tentatives ont été faites pour améliorer les performances de TCP dans les réseaux sans fil, mais aucune de ces approches n'a été normalisée. De plus, des recherches ont été effectuées et qui s'intéressent aux performances de TCP dans les réseaux mobiles ad hoc multi-sauts mais qui sont encore actives dans ce domaine et de nombreux problèmes sont encore ouverts.

Le travail réalisé dans la première partie des travaux de cette thèse, a comme principal objectif la proposition d'une approche collaborative basée sur une conception Cross-Layer pour améliorer la performance de TCP de bout en bout dans les réseaux sans fil. L'approche proposée vise à résoudre la cause principale de la dégradation de la performance de TCP à travers des liens sans fil, à savoir l'incapacité de TCP à déterminer si un paquet a été perdu en raison d'erreurs de transmission ou par suite de congestion de réseau. Comme une congestion crée déjà un ralentissement global du réseau puisqu'il faut retransmettre les paquets perdus, la solution proposée permet donc de minimiser l'impact des retransmissions pour revenir le plus rapidement possible à un état général stable.

2.1.2. Description de la solution

Le principe général de l'approche proposée est d'apprendre au mécanisme de perte de paquets et de contrôle de congestion du standard TCP à faire la distinction entre les différentes raisons de perte de paquets. Si les pertes de paquets sont dues à la congestion, dans ce cas TCP réduit sa fenêtre de flux jusqu'au soulagement de la congestion. Sinon, si la perte de paquets est due à une autre raison que la congestion, TCP garde la même taille de la fenêtre de flux et lance une procédure qui tente de régler le problème selon la raison de perte détecté.

Notre solution est basée sur une architecture cross-layer et a été testée sur la base du standard TCP Reno. Pour évaluer les performances de la nouvelle architecture nous avons utilisé une simulation sous NS3. Les scénarios et les résultats sont discutés dans ce chapitre.

L'approche cross-layer ou inter-couches s'appuie sur les interactions entre deux niveaux de l'architecture OSI (Open System Interconnection). Dans cette approche, les échanges d'informations se font entre des couches qui ne sont pas forcément adjacentes : la performance globale est optimisée en adaptant chaque niveau en fonction de l'information disponible. Dans notre solution nous avons utilisé la communication directe entre la couche physique (PHY) et la couche transport (TCP).

La communication directe entre les différentes couches du modèle OSI pose un problème de complexité de gestion des espaces de mémoire partagée entre les couches, qui peut conduire à un problème d'implantation dès lors que les ressources des équipements sont limitées. De plus, ce type d'architecture pose aussi des problèmes de maintenance dans la mesure où l'ajout, le retrait ou l'évolution d'éléments, influe sur de nombreuses interactions. Cependant, le choix de ce type de communication dans notre travail, se justifie par l'utilisation du champ « Réserve » qui se trouve dans l'en-tête de segment TCP (figure 23). A l'origine ce champ a été réservé pour un usage futur.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Port Source 2 octets										Port destination 2 octets																					
Numéro de séquence																															
Numéro d'acquittement																															
Taille de l'en-tête		Réserve	ECN / NS	CWR	ECE	URG	ACK	PSH	RST	SYN	FIN	Fenêtre																			
Somme de contrôle																Pointeur de données urgentes															
Options																								Remplissage							
Données																															

Figure 23 : Structure d'un segment TCP

2.1.3. Formulation de la solution proposée

Afin d'adapter le mécanisme de perte de paquets et de contrôle de congestion de TCP dans les environnements sans fil, nous avons ajouté au protocole TCP de nouvelles

fonctions et conditions. Le but de ces fonctions et conditions est de permettre à TCP de faire la distinction entre les pertes dues à la congestion, à la mobilité et aux interférences.

L'idée est de prédire la valeur du prochain RTT en se basant sur les « 5 » dernières valeurs de RTT, la valeur de la puissance du signal et de bruit en utilisant une fonction « f ». C'est Sur la base de cette valeur estimée de RTT (PRTT) que le protocole TCP calculera la valeur du RTO. Pour la prochaine séquence d'envoi de paquets, TCP utilisera la valeur du RTO calculé à la base du RTT estimé comme le temps à attendre avant de décider que les paquets envoyés seront considérer comme perdus ou pas.

$$\mathbf{RTO = \min [UBOUND, \max [LBOUND (BETA * RTT)]]} \quad \mathbf{3.1}$$

Où :

UBOUND est la valeur maximale de RTO (e.g. 1 minute).

LBOUND est la valeur minimale du RTO (e.g. 1 seconde).

BETA est le facteur de variance du délai (par exemple 1,3-2,0)

Dans le cas où TCP détecte une perte de paquets (à la base du RTO estimé), il lancera le nouveau mécanisme qu'on lui a ajouté afin de trouver la vraie raison de perte de paquets à l'aide de formules probabilistes. Ces formules ont comme entrée la puissance du signal minimale de la route 'SSM' (tous les sauts), et la puissance du bruit maximal de la route 'NMX' (tous les sauts) des 5 dernières séquences. Apres que TCP prédit les valeurs de PSSM et de PNMX de la prochaines séquence de paquets a envoyé, il commence à procéder par élimination.

$$PNMX = \frac{\sum_{n=4}^{n=0} (|NMX_n|)}{5} \quad \mathbf{3.2}$$

$$PSSM = \frac{\sum_{n=4}^{n=0} (|SSM_n|)}{5} \quad \mathbf{3.3}$$

$$PRTT = f (RTT, SSM, NMX)$$

$$PRTT = \left(\sum_{n=4}^{n=0} \frac{(RTT_n)}{5} \right) * \left(\frac{|PNMX|}{\frac{\sum_{n=4}^{n=0} (|NMX_n|)}{5}} \right) * \left(\frac{\frac{\sum_{n=4}^{n=0} (|SSM_n|)}{5}}{|PSSM|} \right) \quad \mathbf{3.4}$$

Pour estimer la valeur du RTT prédit, Nous avons proposé une formule calcule la moyenne des 5 dernières valeurs du RTT (des séquences précédentes), puis il multiplie cette moyenne par deux facteurs. Le premier facteur représente la variation de la puissance de signal actuelle par rapport aux 5 dernières valeurs de puissance de signal. Le deuxième

facteur représente la variation de la valeur du bruit actuelle par rapport aux 5 dernières valeurs de bruit. A la fin, TCP va estimer une valeur de RTT basé sur les anciennes variations de RTT et en fonction du changement des valeurs de puissance de signal et de bruit.

D'abord, TCP commence avec la valeur absolue de PSSM, si elle est inférieure ou égale à la valeur du seuil de la puissance du signal minimal des cartes sans fil utilisé, TCP déduira que la perte est due à la puissance du signal qui n'est pas assez puissante. Ceci peut être dû à la mobilité ou à l'emplacement des nœuds. Dans ce cas TCP fait appel au protocole de routage afin de trouver une nouvelle route (avec une meilleure valeur de puissance du signal) toute en désactivant le mécanisme de réduction de fenêtre de flux de TCP temporairement, ceci va permettre de garder un débit élevé.

Sinon, si la valeur de PSSM est supérieure à la valeur de seuil de puissance du signal minimal de la carte sans fil utilisé, dans ce cas, TCP testera la valeur absolue de PNMX. Si elle supérieur ou égale à la valeur de seuil de bruit maximal de la carte sans fil utilisé, alors TCP déduira que la perte de paquets est due au bruit. Afin de résoudre ce problème le nouveau mécanisme de TCP va tenter de changer le canal sans fil utilisé afin de réduire les interférences toute en désactivant le mécanisme de réduction de fenêtre de flux de TCP temporairement, ceci va permettre de garder un débit élevé.

Comme dernière étape. Après avoir obtenu des résultats négatifs suite aux tests effectués par le nouveau mécanisme sur les valeurs de PSSM et de PNMX, TCP conclura que la perte de paquets a été provoquée par une congestion du réseau. Pour résoudre le problème, le nouveau mécanisme n'a qu'à s'assurer que le mécanisme de réduction de fenêtre de flux de TCP est bien activé.

2.1.4. Récupération de la valeur du RSSI minimal et du bruit maximal du chemin

Dans les réseaux ad hoc la transmission des paquets peut se faire en mono-saut ou en multi-saut. Dans le mono-saut l'émetteur et le récepteur partage une liaison unique. Cette liaison est caractérisée par une puissance du signal unique et un bruit unique, ce qui facilite le processus de récupération de ces deux valeurs. Contrairement au mono-saut, les transmissions multi-saut se font à travers plusieurs liaisons entre plusieurs nœuds, chaque deux nœuds de la route utilisée, partage une liaison différente avec une puissance du signal et du bruit différente de ceux qui les succèdent.

Afin de récupérer la puissance du signal minimale et la puissance du bruit maximale de la route, nous avons fait appel au champ 'RESERVED' des paquets TCP ACK qui représente le retour du récepteur vers l'émetteur (voir la section Pré-requis). Sur ce champ, nous allons sauvegarder, la puissance du signal minimal et la puissance de bruit maximale du chemin de la sorte :

- Les parties du champ qui sont réservés à la puissance du signal et du bruit seront initialisés avec la puissance du signal et le bruit du récepteur, respectivement.
- Après chaque saut vers un nouveau nœud de la route, le mécanisme compare les deux valeurs contenues dans le paquet ACK, si la puissance du signal du nœud est inférieure de celle de l'ACK alors il remplace celle de l'ACK avec celle du nœud, sinon il ne fait rien. De même pour le bruit, si la valeur du bruit du nœud courant est plus puissante que celle sauvegardée sur l'ACK alors il remplace celle de l'ACK avec celle du nœud, sinon il ne fait rien.
- Ainsi de suite jusqu'à ce que le paquet ACK arrive vers l'émetteur, là il va faire la même chose que les nœuds intermédiaires, puis il récupère les deux valeurs sauvegardées sur le champ 'RESERVED' de l'ACK. Ainsi, le récepteur, recevra la valeur courante minimale de la puissance du signal et la valeur maximale du bruit.

Algorithme de récupération des valeurs de SSM et de NMX

Début

// Quand un nœud 'n' reçoit un paquet ACK

Si (nœud(n) <>nœud_émetteur) **alors**

 CSS = Get(nœud(n)_signal_strength);

 CN = Get(nœud(n)_noise);

Si (ACK_RESERVED(SS)>CSS) **alors**

 ACK_RESERVED(SS)=CSS;

Si (ACK_RESERVED(N)<CN)**alors**

 ACK_RESERVED(N)=CN;

n=n+1;

Sinon

n=n+1;

Fin ;

2.1.5. Organigramme de la solution proposée

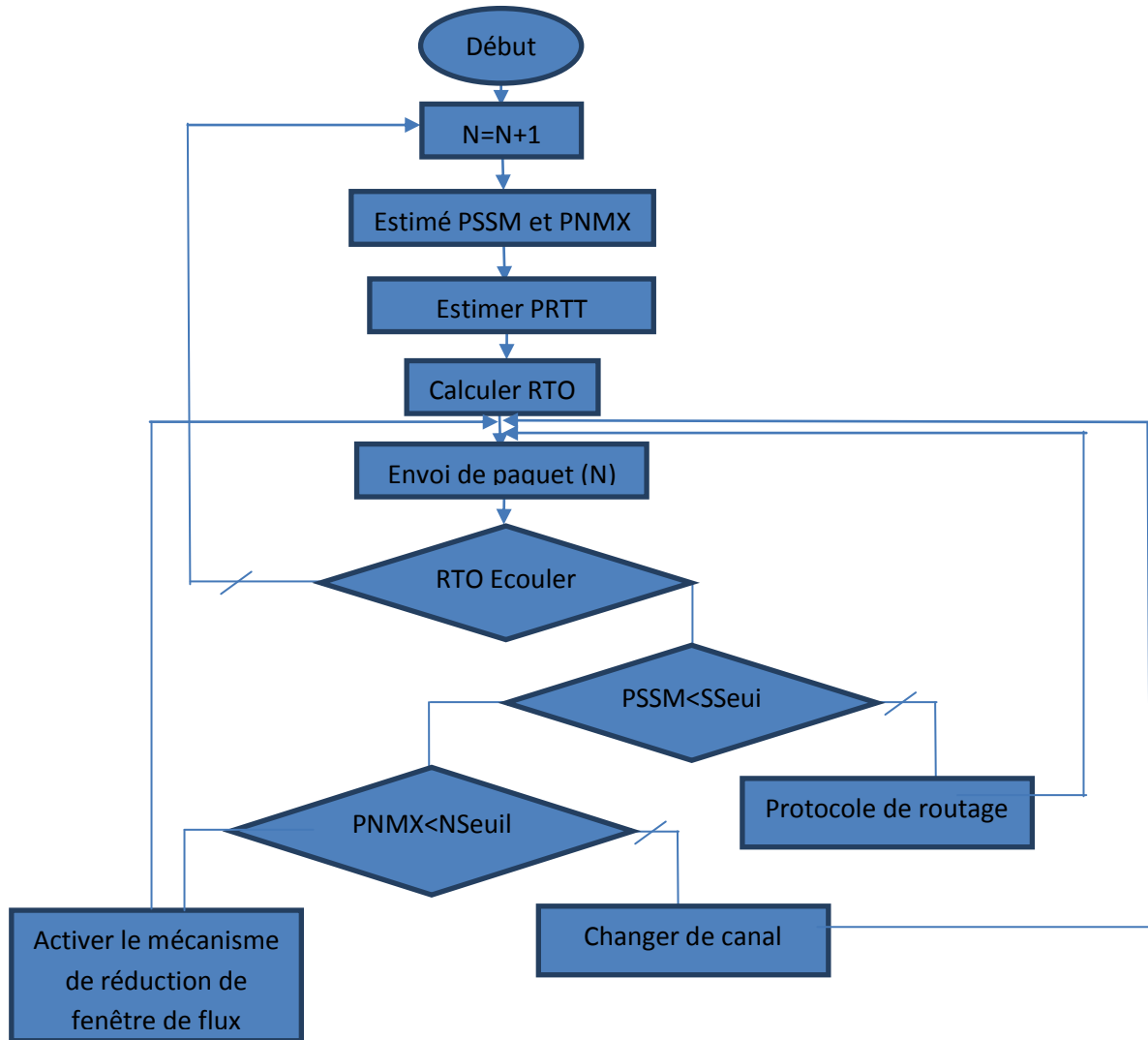


Figure 24 : Organigramme de la solution proposée

Algorithme descriptif du mécanisme proposé

Début

// A la réception de l'ACK (n-1) de la dernière séquence envoyé.
 SMM = recup(SS, ACK (N-1)) ; //SS est la puissance du signal
 Enfiler(SSM, TSSM);
 PSSM = somme (TSSM[0]+TSSM[1]+TSSM[2]+TSSM[3]+TSSM[4])/5;
 PRTT = f (RTT, SSM, NMX); //Estimation de PRTT
 PRTO = min (UBOUND, max (LBOUND (BETA * RTT))); // Calcul de PRTO
 Seuilssm = get_sseuil(eth0); //Récupération de la valeur de seuil signal de la carte
 Seuilnmx = get_nseuil(eth0); //Récupération de la valeur de seuil bruit de la carte
 Envoi (Seq (n)); //Envoi de la séquence de paquet en cours
 Si(RTO_ecouler(PRTO)=faux)**alors**
 N=N+1 ; //Passer à la séquence de paquets suivante

Sinon

Si (PSSM >= Seuilssm)**alors**
 Exécuter (AODV) ; // Essayer de résoudre le problème de Signal bas
 Envoi (Seq(n)) ; //Vérifier si le problème est résolu

Sinon

Si (PNMX < Seuilnmx)**alors**
 Exécuter (Changer le canal sans fil) ; //Résoudre le problème
 Envoi (Seq(n)) ; //Vérifier si le problème est résolu ou pas

Sinon

Exécuter (Activer le mécanisme de réduction de fenêtre TCP) ;
 //Afin de soulager la congestion
 Envoi (Seq(n)) ; //Vérifier si le problème est résolu

Fin

2.1.6. Evaluation par simulations
2.1.6.1. Spécification du contexte d'évaluation

Pour pallier principalement aux difficultés de la mise en œuvre de l'expérimentation en environnement réel (manque de contrôle au niveau des paramètres, la difficulté de reproduire une même expérience, etc.), il existe un moyen classique permettant l'évaluation du protocole, ce moyen est la simulation de réseau.

La simulation est une solution souvent plébiscitée par la communauté scientifique. Elle possède en effet l'avantage de reposer sur des modèles validés ou au moins reconnus. De nombreuses approches et solutions informatiques sont actuellement modélisées implémentées et reconnues par la communauté scientifique par le biais de la simulation. De ce fait, la simulation est souvent utilisée par les chercheurs pour mener leurs expérimentations.

Classiquement, une simulation consiste à reproduire le comportement du système complet dans un environnement synthétique où le temps est simulé par une horloge à événements discrets. Elle demande une modélisation complète, allant des applications au réseau physique, ce qui peut se révéler à la fois complexe et coûteux en terme de temps d'exécution et d'utilisation mémoire dans le cas de réseaux de taille importante (100 nœuds et plus). La simulation à événements discrets est le modèle le plus utilisée par la communauté scientifique.

Les simulateurs de réseau sont des outils logiciels généralement utilisés pour le développement, le test ou encore le "debugage" d'applications ou de protocoles réseau. Un simulateur à événements discrets se caractérise par le fait que les changements d'états dans le réseau simulé (événements) se produisent à des instants (sans durée) répartis de manière discrète sur l'axe des temps. Dans le cas des réseaux complexes, ceci permet de traiter les événements sans être soumis à la contrainte du temps réel.

2.1.6.2. Environnement et scenario

L'implémentation et la validation de notre approche a été simulé avec le logiciel de simulation réseau NS3 (network simulator 3). Celui-ci est un logiciel libre de simulation à événements discrets très largement utilisé dans la recherche académique et dans l'industrie. Il est considéré par beaucoup de spécialistes des télécommunications comme le meilleur logiciel de simulation à événements discrets, en raison de son modèle libre, permettant l'ajout très rapide de modèles correspondant à des technologies émergentes.

Durant la simulation, plusieurs scénarios ont été considérés. Le but était de comparer l'approche que nous avons proposée avec deux autres versions de TCP, qui sont : TCP Reno et TCP solution avec puissance du signal afin de vérifier la performance de notre approche par rapport aux approches étudiées.

Notre modèle de simulation est basé essentiellement sur les principes suivants :

- Une topologie de réseau ad hoc composée de 23 nœuds homogènes
- Tous les nœuds utilisent le même protocole de routage « AODV ».
- Le réseau est représenté par un graphe aléatoire. Dans ce graphe, chaque nœud est placé initialement d'une manière aléatoire et uniforme dans une région déterminée au préalable.
- Tous les nœuds sont identiques (les mêmes caractéristiques) mais ils fonctionnent de manière indépendante ;
- Chaque nœud décide seul de ses déplacements ;
- L'accès au support de transmission se fait par le protocole CSMA/CA;

La topologie du réseau est générée par une distribution aléatoire des nœuds dans une région fixée préalablement par la simulation. Elle est donnée par la figure 25.

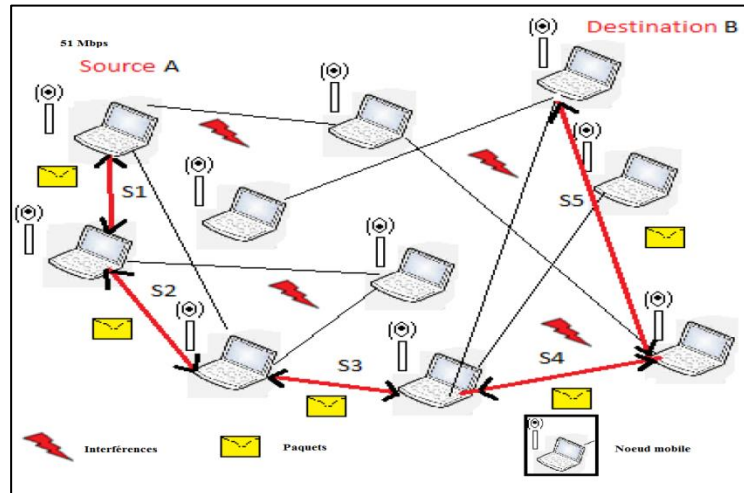


Figure 25 : Topologie de la simulation

Le transfert de paquets se fait en multi-saut. Le débit de transfert maximal est de 50 Mbps. Les données utilisées sont des données multimédia (séquences vidéo). Enfin, les interférences et la mobilité des nœuds (représentées par des variables) changent de scénario à un autre.

Les différentes classes TCP (NS3) que nous avons utilisé pour implémenter notre approche TCP sont :

TOPOLOGIE-TEST.cc/ NODE.cc/ MOBILITY.cc/ TCP-Reno.cc/ TCP-SOCKET-BASE.cc/ SET-RTT.cc/ RTT-ESTIMATOR.cc/PACKET.cc/ YANS-WIFI-PHY.cc/ (get signal, get noise).

Plusieurs simulations sous différents scénarios ont été effectuées pour étudier le comportement de notre approche. Nous avons considéré 4 scénarios qui sont décrits ci-dessous :

Scénario 1 : un transport de données sur un réseau ad hoc multi-saut, les nœuds de la topologie ne sont ni mobiles ni sous l'effet des interférences.

Scénario 2 : un des nœuds de la topologie (faisant partie de la route utilisée) est mobile. Situé au centre de la topologie, il se déplace avec le temps vers l'extérieur de la topologie. Le routage est statique, tous les paquets passent par le nœud mobile. Dans ce scénario les interférences sont négligées.

Scénario 3 : les nœuds de la topologie ne sont pas mobiles, mais ils sont sous l'effet des interférences qui augmente avec le temps. Les interférences sont appliquées sur le canal par défaut avec un faible effet sur les canaux du voisinage. La fonctionnalité de changement de canal de notre solution est désactivée.

Scénario 4 : un des nœuds de la topologie (faisant partie de la route utilisée) est mobile. Il se déplace du centre vers l'extérieur de la topologie. Le routage est statique, tous les paquets passent par le nœud mobile. Dans ce scénario les interférences sont négligées. Le même nœud subit un effet d'interférence croissant. Les interférences appliquées sur le canal de ce nœud mobile ont un faible effet sur les canaux du voisinage. La fonctionnalité de changement de canal de notre solution reste désactivée.

2.1.6.3. Résultats obtenus et discussion

Les résultats de chaque scénario sont représentés par des graphes, où chaque graphe permet de comparer les résultats obtenus des trois versions de TCP étudiées. Dans cette section nous résumons les résultats les plus concluants.

a) Résultat du scénario 1

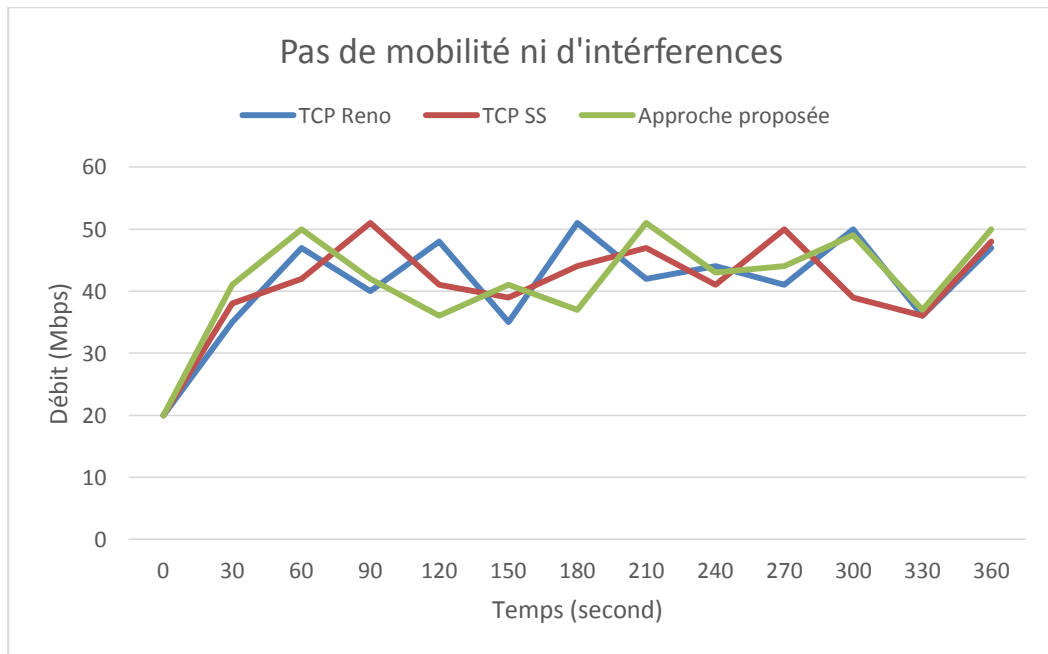


Figure 26 : Résultat du premier scénario

Interprétation

L'analyse du graphe (figure 26) comparatif du premier scénario de simulation montre que : Si on est dans un environnement sans fil et que les nœuds de ce dernier ne sont ni mobiles ni sous l'effet des interférences, les trois versions testées (dotés d'un mécanisme de contrôle de congestion) donnent presque les mêmes résultats en matière de débit fourni. L'explication est que dans un cas considéré comme idéal, un environnement sans fil peut être considéré comme un environnement filaire (où les pertes de paquets sont principalement dues à la congestion).

b) Résultat du scénario 2

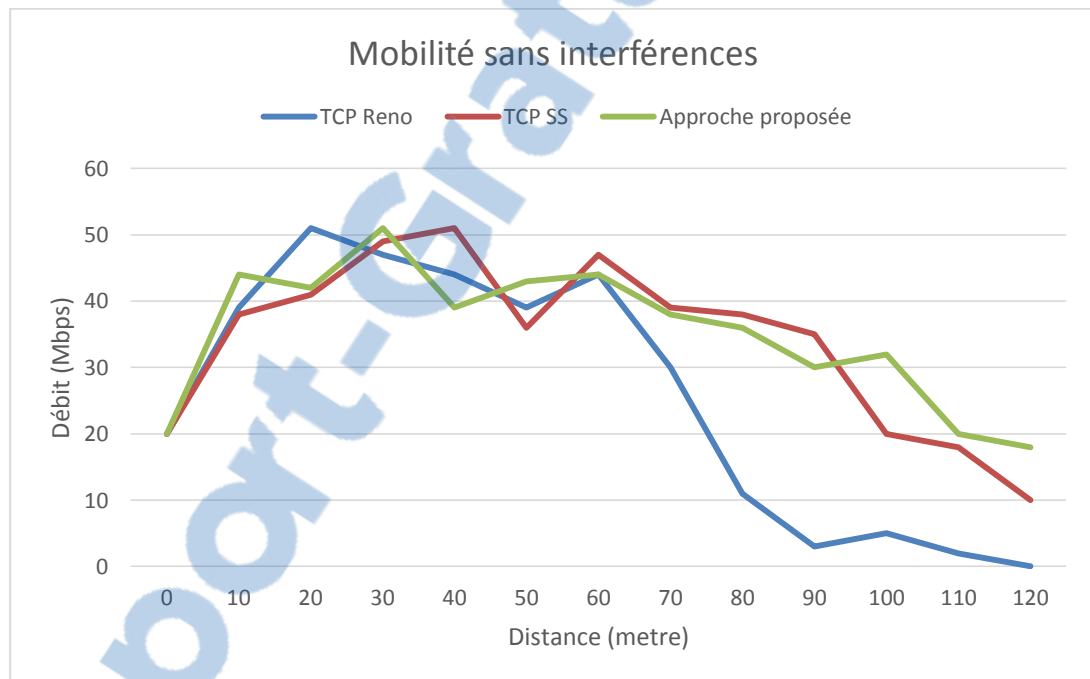


Figure 27 : Résultats du deuxième scénario

Interprétation

Dans ce scénario où nous avons introduit la mobilité, nous remarquons que l'éloignement du nœud de la zone de couverture de l'environnement sans fil impacte négativement la puissance du signal de ce nœud. Cet effet se répercute sur le mécanisme de perte de paquets de TCP.

Pour TCP Reno, qui ne dispose pas de mécanisme lui permettant de s'adapter à la perte de paquets, tente d'alléger une congestion qui n'existe pas. En effet, une fois que la puissance du signal atteint un certain seuil où les paquets commencent à se perdre suite à la mobilité, le mécanisme qui réduit la taille de la fenêtre de flux de TCP Reno se déclenche

afin de limiter le débit. Les répercussions de cette tentative sont négatives sur les performances de TCP Reno comme le montre la figure 27 à partir de la 60ème seconde.

Contrairement à TCP Reno, TCP avec la solution de la puissance du signal (TCP SS), est équipée d'un mécanisme qui lui permet de faire la distinction entre les pertes qui sont dues à la congestion et les pertes dues à la puissance du signal (mobilité). Les résultats montrent que TCP SS a un meilleur comportement que TCP Reno lorsque la puissance du signal du nœud en mouvement commence à s'affaiblir à fur et à mesure qu'il s'éloigne de la zone de couverture sans fil. Ainsi tant que la puissance du signal est suffisamment élevée pour faire une transmission, TCP SS essaye de garder une valeur maximale de débit.

Comme notre approche se base aussi sur la puissance du signal, les résultats sont assez proches de ceux de TCP SS. La raison est qu'une fois que la puissance du signal commence à s'affaiblir et qu'elle cause quelques pertes de paquets, notre approche désactive le mécanisme de réduction de la fenêtre de flux TCP, ce qui conduit à la continuation du transfert (nouveaux paquets ou paquets perdu) avec un haut débit.

c) Résultat du scénario 3

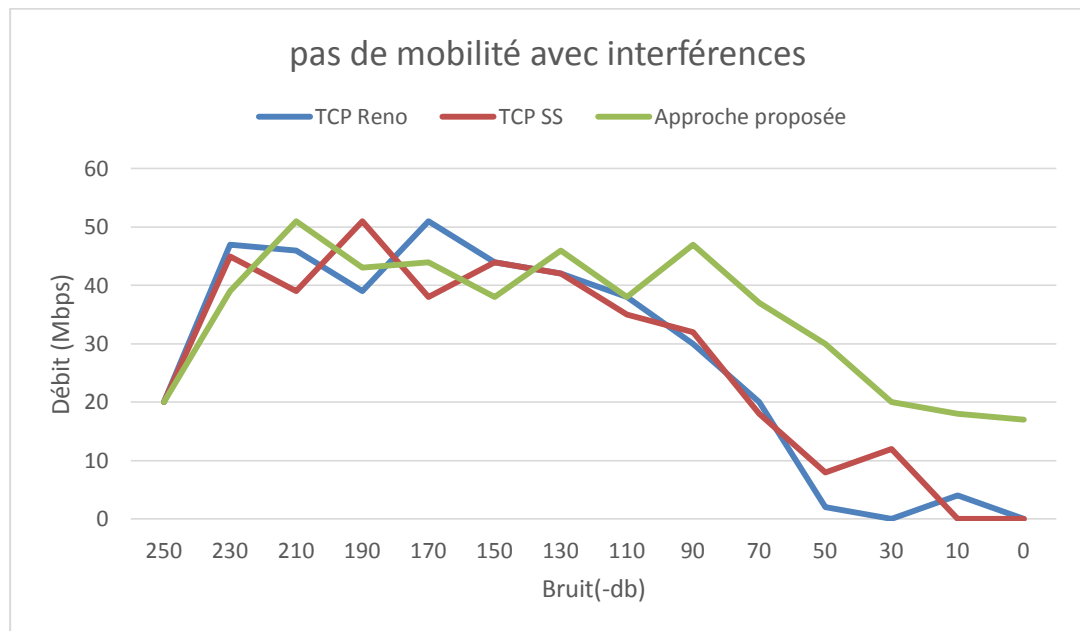


Figure 28 : Résultat du troisième scénario

Interprétation

Dans le scénario illustré par la figure 28, les nœuds ne sont pas mobiles mais sont soumis à des interférences. Après une certaine intensité du bruit, les interférences commencent à provoquer des pertes de paquets. Comme TCP Reno et TCP SS ne disposent

pas de mécanisme qui détecte les pertes dues au bruit, ils considèrent les pertes subies comme des cas de congestion. Ce phénomène peut avoir pour effet de dégrader les performances des flots de ces protocoles en réduisant abusivement leur débit.

En revanche, dans notre approche TCP considère les intensités du bruit et change de canal pour avoir de meilleures performances lorsque le bruit est moyen. Lorsque l'intensité de celui-ci atteint une valeur assez importante, TCP désactive le mécanisme de réduction de flux et continue à émettre avec le même débit. Ceci justifie les bons résultats de notre approche comme le montre la figure 28.

d) Résultat du scénario 4

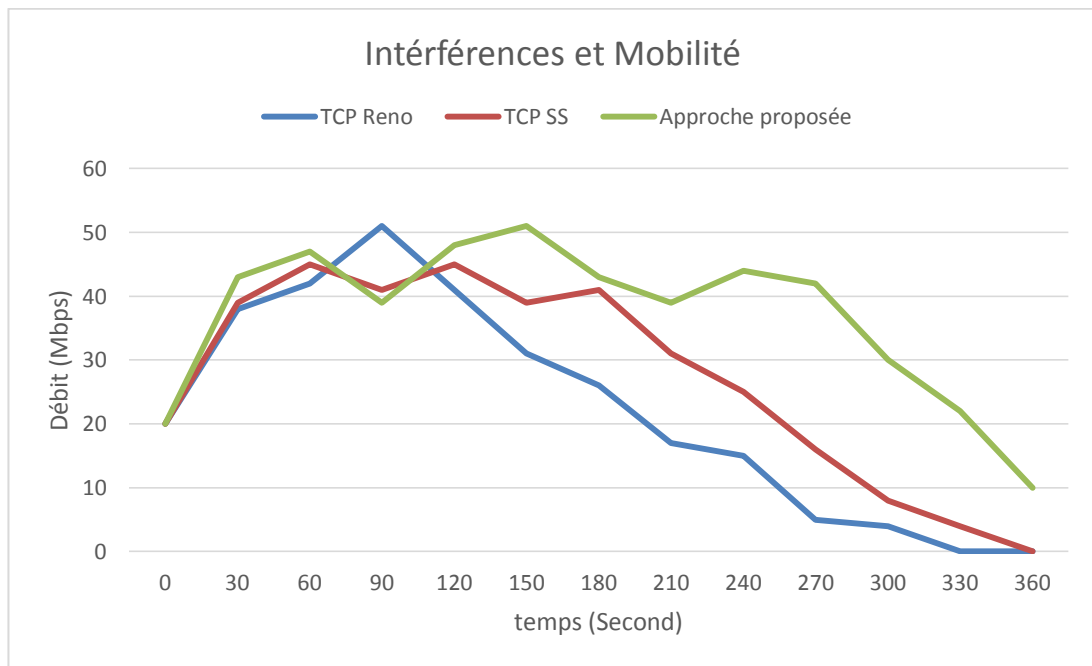


Figure 29 : Résultat du quatrième scénario

Interprétation

Les conditions de cette simulation vont augmenter la probabilité de perte de paquets dues à l'environnement sans fil par rapport à ceux dues à la congestion (mobilité et interférence). Les résultats (figure 29) montrent que TCP Reno perd considérablement en performance après une certaine valeur d'interférences et après que le nœud mobile atteinte une distance du centre de la topologie. Cela est dû au fait que le nombre de perte dû à l'environnement sans fil augmente considérablement et que TCP Reno les considère comme cas de congestion. TCP SS réagit de la même façon que TCP Reno, il perd en performance après une certaine valeur d'interférence, et considère que les pertes dues au bruit sont dues à la congestion. Il réduit ainsi le débit mais gère le problème de puissance du signal. Notre approche offre une meilleure performance en matière de débit. Cette

performance découle du traitement séparé des pertes de paquets dues à la mobilité et aux interférences par rapport aux pertes dues à la congestion.

2.1.7. Conclusion

Notre première contribution vise à améliorer les performances du protocole TCP dans les environnements sans fil en général et dans les réseaux ad hoc en particulier. Elle est basée sur une solution Cross-Layer qui fait intervenir les couches Physique et Transport. L'idée est d'adapter le protocole TCP Reno avec les caractéristiques des réseaux ad hoc en introduisant deux nouvelles grandeurs qui sont la puissance du signal et le bruit. Ces deux grandeurs vont aider le mécanisme de perte de paquets de TCP à faire la distinction entre les pertes dues à la congestion et ceux dues à l'environnement sans fil.

L'approche a été testée dans une simulation et les résultats obtenus démontrent sa performance comparée à celle de TCP Reno classique et de la version TCP solution avec puissance du signal qui représente une tentative d'amélioration du protocole TCP dans les réseaux sans fil.

2.2. Deuxième contribution : TCP et l'amélioration des services de vidéo streaming adaptatifs

2.2.1. Introduction

Le téléchargement progressif ou pseudo streaming permet à l'utilisateur de lire un média alors que le fichier est encore en cours de téléchargement. Il se différencie du téléchargement " normal " pour lequel l'utilisateur doit attendre jusqu'à ce que le fichier soit complètement téléchargé pour pouvoir ensuite le lire depuis son disque dur. Avec le pseudo streaming, une copie du fichier est conservée sur le disque de l'utilisateur, ce qui lui permettra de relire le media ultérieurement. Le téléchargement progressif est également appelé streaming HTTP car les logiciels serveur Web utilisant des protocoles standard (serveurs HTTP) peuvent transmettre des fichiers à téléchargement progressif ; il se différencie du vrai streaming, qui tire parti des protocoles spéciaux utilisés par les logiciels serveur de médias pour adapter la transmission à la bande passante.

Lorsque le protocole HTTP est utilisé pour le téléchargement progressif, la fiabilité en elle-même du protocole peut poser des problèmes, étant donné que la retransmission des paquets perdus ou endommagés peut interrompre la lecture du pseudo-streaming. Le protocole HTTP n'offre pas de fonction permettant d'adapter la vitesse de transfert des données au débit audio ou vidéo, c'est-à-dire la vitesse de lecture des données qui constituent le contenu multimédia. À l'aide du protocole HTTP, le téléchargement d'un film d'une minute pourrait prendre une seconde, une minute ou même une heure, selon la taille du fichier et la vitesse de la connexion. Si la vitesse de connexion est inférieure au débit de données du contenu multimédia, celui-ci arrivera quand même au client, mais le téléchargement progressif ne pourra pas s'effectuer sans difficulté. Le protocole HTTP

fonctionne sur TCP, ce qui lui permet de transmettre des contenus audio ou vidéo sans perte ni endommagement. Les paquets perdus ou endommagés sont simplement renvoyés.

Cependant, le streaming, avec son assortiment de protocoles propriétaires, s'est retrouvé de lui-même à ne plus réussir à répondre à la demande croissante des utilisateurs. C'est en 2007 que la société « Move Networks » introduit un nouveau concept qui fit grandement évoluer le streaming au niveau international : le streaming adaptatif basé sur HTTP.

Ainsi, l'utilisation de l'application de lecture vidéo présente sur le terminal du client permet de superviser la vitesse de téléchargement et demander des paquets de qualités variables en s'adaptant automatiquement aux conditions du réseau disponible. L'impact de ce choix est énorme, car il a permis aux vidéos d'être distribuées un peu partout à l'aide des réseaux standards et d'être mis en cache pour le passage à l'échelle.

L'arrivée des protocoles de streaming adaptatif sur HTTP a révolutionné la diffusion de multimédia sur Internet. Cette technologie est principalement basée sur quelques paramètres de la QoS. Cependant, le ressenti de l'utilisateur n'est pas pris en compte et rien ne permet aux fournisseurs de services d'être informés sur la satisfaction des utilisateurs par rapport au contenu. Ainsi, jusqu'à présent la QoE n'a pas été considérée comme une entrée dans le processus d'adaptation du flux de diffusion distribué à travers HTTP fonctionnant sur TCP.

2.2.2. Positionnement du problème

Les services de vidéo streaming à travers le protocole HTTP (*cf.* figure 30) utilisent le protocole de transport TCP pour le transport de données entre le serveur vidéo streaming adaptatif et le client. Le principe de ces services est de fournir au client une qualité vidéo relative par rapport à sa configuration réseau dans le temps. L'un des paramètres qui a le plus d'impact dans le processus de choix de la qualité vidéo à transmettre est le débit de la connexion du client. Plus le client a un meilleur débit, plus la qualité vidéo qu'il recevra du serveur sera meilleure. Comme il a été déjà mentionné dans le chapitre I, le débit a un rapport direct avec les paramètres du protocole TCP, que ça soit la taille de la fenêtre TCP, le délai ou le RTT. Donc tout changement dans les paramètres de TCP va impérativement changer le débit qui impactera à son tour la qualité vidéo reçue à partir du serveur.

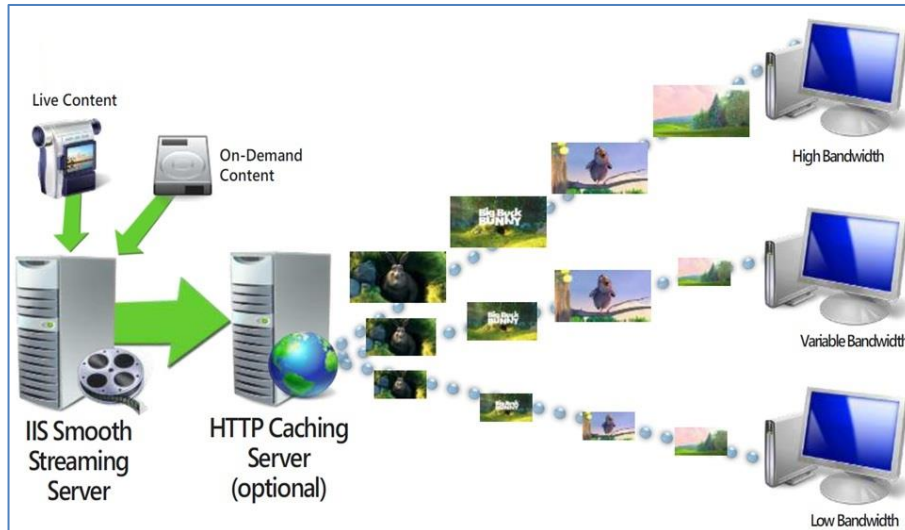


Figure 30 : Vidéo streaming adaptatif à travers HTTP

De nos jours, les services de vidéo streaming (cf. figure 31) sont parmi les services les plus utilisés sur Internet. Ceci a créé une grande concurrence entre les fournisseurs de cette catégorie de service. En effet, chaque fournisseur essaye d'attirer le plus de clients possible pour des fins commerciales, en proposant des services de vidéo streaming qui répondent au maximum aux attentes des clients. Ce fait a conduit à l'introduction du concept de la QoE dans les services de vidéo streaming adaptatifs. Le but dans ce type de service est d'apporter une meilleure satisfaction aux clients en dépit des contraintes matériel et réseau (QoS) afin d'attirer le maximum possible de client vers leurs marchés.



Figure 31 : Services de vidéo streaming adaptatif les plus connus

Les utilisateurs utilisent différents types de terminaux et de technologies afin (cf. figure 32) d'accéder aux services de vidéo streaming adaptatif tel que : les smart phones, les tablettes, les ordinateurs portables, les ordinateurs de bureau, les télévisions...etc et même sur des montres intelligentes. Chaque type de terminal est caractérisé par des paramètres comme : son CPU, sa batterie, sa taille d'écran, sa résolution d'affichage, sa mémoire vive....etc. D'autre part, l'accès à ces services se fait aussi à travers différents types de technologies de connexion Internet comme : l'ADSL (Sans fil et câblé), la 3G, la 4G...etc. Ces différentes technologies offrent parfois un accès illimité à Internet comme l'ADSL mais parfois limité tel que la 3G ou la 4G.

Comme dans certains cas où la connexion à Internet est limitée, les utilisateurs préfèrent recevoir une diffusion de qualité suffisamment bonne relativement avec leur terminal plutôt qu'une distribution couteuse en matière de forfait Internet. Par exemple, lorsqu'il s'agit d'un smart phone connecté à la 4G, le débit est très bon et le client recevra donc un contenu d'une qualité de 1080p (si elle est disponible sur le serveur), mais comme le client est sur un smart phone avec un petit écran et une batterie assez limitée, le mieux serait de lui transmettre une qualité moins importante (480p ou 720). De ce fait, le client économisera son forfait 4G ainsi que la consommation de sa batterie (une meilleure qualité vidéo consomme plus de batterie qu'une autre qualité plus faible).

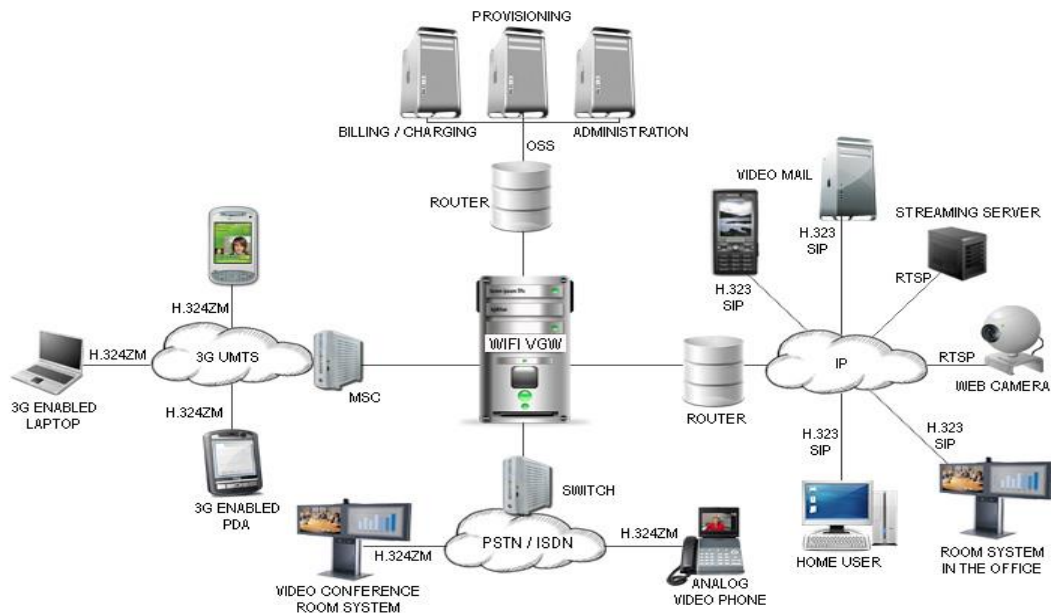


Figure 32 : Accès aux services de vidéo streaming adaptatif à travers différents types de technologies avec différents type de terminaux

Dans la suite de ce chapitre nous allons décrire une nouvelle approche qui vise à améliorer la qualité d'expérience des utilisateurs de service de vidéo streaming adaptatifs en utilisant les paramètres du protocole de transport TCP. L'idée est d'estimer une valeur de débit en fonction des paramètres du terminal de l'utilisateur et de son feedback en utilisant des formules d'apprentissage par renforcement. Nous appliquerons ensuite cette valeur de débit en utilisant le paramétrage de TCP et le routage interne afin de recevoir une qualité de vidéo qui satisfait les utilisateurs.

2.2.3. Description de la solution

2.2.3.1. Principe de la solution.

Le principe de l'approche proposée repose sur l'introduction d'un facteur utilisateur global UF. Ce facteur est calculé sur la base d'une pondération des différents paramètres du terminal de l'utilisateur et sera mise à jour en fonction du feedback de l'utilisateur. Pour cette mise à jour, nous avons utilisé l'apprentissage par renforcement pour la prise en compte du caractère évolutif des différents paramètres.

Chaque intervalle de valeurs de l'UF correspond à une qualité vidéo bien précise, qui à son tour, correspond à une valeur de débit estimé. Donc, l'idée est de calculer un facteur d'utilisateur UF, lui attribuer une qualité vidéo, déduire le débit correspondant et finalement appliquer ce débit sur l'interface liée à Internet (interface virtuelle).

2.2.3.2. Les étapes de la solution

Comme première étape de la solution, nous devons récupérer la valeur possible des différents paramètres du terminal de l'utilisateur. Dans notre approche nous nous sommes focalisé sur 3 paramètres qui sont : la **résolution de l'écran**, la **taille de l'écran** et la **batterie** disponible. En plus de la dépendance qui existe entre eux, ces facteurs sont les plus pertinents. En effet, un affichage en haute résolution va demander davantage de puissance de traitement et consommera plus de batterie (*ex.* le texte devient illisible à basse résolution et une haute résolution demandera plus de batterie).

Etape 1 : Les différents paramètres sont :

- FR → le facteur de résolution d'affichage, calculé sur la base de la table 3.
- FS → le facteur de taille de l'écran, calculé sur la base de la table 4.
- FB → le facteur de batterie disponible, calculé sur la base de la table 5.

Résolution de l'affichage (TR)	Facteur de résolution (FR)
$2160 \leq TR$	7
$1440 \leq TR < 2160$	6
$1080 \leq TR < 1440$	5
$720 \leq TR < 1080$	4
$480 \leq TR < 720$	3
$240 \leq TR < 480$	2

Table 3 : Valeurs de FR (facteur de résolution)

Taille de l'écran (SS)	Facteur de la taille d'écran (FS)
$55 \leq SS$	7
$42 \leq SS < 55$	6
$32 \leq SS < 42$	5
$15 \leq SS < 32$	4
$7 \leq SS < 15$	3
$SS < 7$	2

Table 4 : Valeurs de FS (facteur d'écran)

Batterie disponible (BC %)	Facteur de batterie (FB)
$40 \leq BC$	6
$20 \leq BC < 40$	4
$0 \leq BC < 20$	2

Table 5 : Valeurs de BC (facteur de batterie)

La pondération de chaque paramètre va aider à estimer la valeur du facteur global UF. En effet UF est égal à la somme des 3 sous-facteurs FR, FB, FS.

$$UF = FR + FB + FS \quad 3.5$$

Le facteur utilisateur est une note qui va de 8 comme valeur minimale jusqu'à 20.

Etape 2 : Une fois le facteur utilisateur estimé, nous entamons l'étape d'estimation de débit. La table 6 attribue à chaque facteur d'utilisateur UF une qualité de vidéo à recevoir :

Facteur utilisateur (UF)	Qualité vidéo (pixel)
6	144p
7	144p
8	144p
9	240p
10	240p
11	360p
12	360p
13	360p
14	480p
15	480p
16	720p
17	720p
18	1080p
19	1080p
20	1440p

Table 6 : Les valeurs facteur utilisateur avec les qualités vidéo

Dans les services de vidéo streaming adaptatif, chaque qualité vidéo requière un débit minimal approximatif pour bien fonctionner sans interruption. La table 7 attribue à chaque qualité vidéo une valeur minimale de débit. Cette table va nous permettre d'estimer le débit nécessaire correspondant au facteur UF.

Après l'attribution d'une qualité vidéo au facteur d'utilisateur, nous attribuons par la suite à cette qualité vidéo, un débit Internet en se basant sur la table 7. Par conséquent, à la fin nous obtenons un débit attribué au facteur d'utilisateur que nous avons calculé au début. Cette valeur de débit sera utilisée pour limiter le flux vidéo de l'interface Internet.

Qualité vidéo (pixel)	Débit minimal requis (Kbps)
144p	32
240p	57
360p	128
480p	245
720p	430
1080p	720
1440p	1057

Table 7 : Les qualités vidéo avec les débits minimaux

Le processus d'adaptation est exécuté toutes les 5 minutes, ceci est dû au fait que les facteurs d'utilisateur peuvent changer dans le temps. Par exemple : la batterie qui s'affaiblit ou bien même l'utilisateur change d'écran ou de terminal (le cas de la vidéo à la demande).

Afin de donner une meilleure satisfaction à l'utilisateur nous avons doté notre approche d'un mécanisme d'apprentissage par renforcement. Ce mécanisme permet de mettre à jour les valeurs de la qualité vidéo attribuées à chaque facteur d'utilisateur UF en fonction du feedback ou de la note donnée au service par l'utilisateur. Cette note est comprise entre 1 et 5 et elle représente le ressenti global de l'utilisateur ou le MOS (Mean Opinion Score) vis à vis du service après chaque visualisation.

2.2.3.3. Formalisme du modèle proposé

Dans cette section, nous allons modéliser le contexte dynamique de la phase d'adaptation des valeurs de la qualité vidéo avec celles du facteur utilisateur. L'objectif est de caractériser cette phase d'adaptation par un processus décisionnel de type PDM (Processus de Décision de Markov).

Le mécanisme de mise à jour (table 6) fonctionne comme suit : Après que le serveur vidéo transmet au client une description de la vidéo (suite à sa demande), l'utilisateur demande un segment sur la base de mesures de débit locales. Cette demande ne sert que de suggestion préliminaire et donc une optimisation locale correspondante est effectuée. Nous utilisons une interface virtuelle et un opérateur mobile qui agit à son niveau pour adapter son débit. L'opérateur mobile reformule la demande sous la forme d'un PDM fini afin de prendre en compte le problème de d'adaptation et de le transmettre au serveur vidéo.

Dans cette modélisation, nous travaillons dans un contexte observable et introduisons un PDM qui le modélise. Le but est de pouvoir caractériser des politiques d'adaptation

vérifiant ainsi la satisfaction de l'utilisateur par rapport au contenu reçu lorsque la bande disponible bw décroît et N (le nombre d'utilisateurs) augmente.

Pour cela, nous estimons que pour chacun des N utilisateurs il y a un ensemble de facteurs Utilisateur indexés par $l = 1, 2, \dots, L$. Nous appelons cet ensemble $U = \{UF_l^i / i = 1, 2, \dots, N \ \& \ l = 1, 2, \dots, L\}$. Soit V , l'ensemble feedbacks des utilisateurs indexés par $j = 1, \dots, M$ où $V = \{V_j^i / i = 1, 2, \dots, N \ \& \ j = 1, 2, \dots, M\}$.

Le cadre formel est construit sur le processus de révision périodique de la valeur du débit attribuée à chaque facteur d'utilisateur en se basant sur la note donnée par le feedback de l'utilisateur. Nous considérons que chaque utilisateur a 'K' états de débit possible et chaque 'état' a un facteur d'utilisateur global correspondant, à savoir $B = \{bw_k / k = 1, 2, \dots, K\}$ où bw_k est le débit requis à l'étape k pour un facteur d'utilisateur. La valeur du débit mesuré (réellement affecté qui peut ne pas être égale à la valeur de bw_k) est mise à jour périodiquement dans la table 7, afin de servir comme entrée pour le processus d'adaptation.

2.2.3.4. Formulation du processus de décision de Markov

Un PDM et un ensemble composé de quatre éléments ($S ; A ; P ; r$) qui interviennent dans le processus d'apprentissage par renforcement. Ces composants sont S , l'ensemble des états du système, A un ensemble d'action, $P(s)$ la probabilité de transition de l'état s_t à l'instant t vers l'état s_{t+1} à l'instant $t+1$, lorsqu'une action a_t est appliquée. La fonction de récompense $r(s)$ indique le retour (positif ou négatif) immédiat lorsque l'action a_t est appliquée à l'état s_t . Au niveau de chaque état du système, un agent calcule le gain obtenu. L'objectif de cet agent est d'apprendre qu'elle action choisir pour un état s_t afin de maximiser la fonction de récompense cumulative.

Les quatre composants de notre PDM sont :

a) Les états du système

Un état (observable) du système à l'instant t est défini par : $s_t = (VQ_t ; UF_t^i ; BW_t)$ où :

- VQ_t représente la qualité vidéo à l'instant t ,
- UF le facteur utilisateur correspondant pour VQ à l'instant t pour l'utilisateur i
- et BW est le débit correspondant à la qualité VQ à l'instant t pour l'utilisateur i .

En outre, l'ensemble des états du système $S = \{s_1 ; s_2 ; \dots ; s_T\}$ représente l'évolution du service de vidéo streaming durant une session T . L'instant 't' représente la durée de la transmission de la vidéo. Etant donné que l'état du système ne dépend que de son état précédent (le plus récent), la propriété de Markov est préservée.

Nous définissons l'espace des états S comme : $S = U^N \times B^N$.

b) Les actions du système

Nous définissons les actions séquentielles par : $\{a_t\}$ pour $(t=0, 2, 3..)$ où l'action a_t est une décision prise à l'instant 't'. L'ensemble des actions pour un état donné est $A(S) = \{A_{sbw}, A_{ubw}\}$ ou l'action A_{sbw} sert à sélectionner une valeur de débit par rapport à la qualité vidéo VQ_t relativement au facteur d'utilisateur UF_t^i . L'action A_{ubw} sert à remplacer et mettre à jour la dernière valeur de débit (par rapport au feedback de l'utilisateur) en fonction des feedbacks (table 6) afin d'être considérée dans les prochaines adaptations.

c) La transition des états

La transition des états entre s_t à s_{t+1} est déterminée par rapport au facteur UF_t^i et au débit disponible à l'instant t.

La probabilité de transition peut être obtenue à l'aide de la formule suivante :

$$\begin{aligned}
 P_{at}(S_t, S_{t+1}) &= \{S_{t+1} | S_t, a_t\} \\
 &= \{(UF_{t+1}^l, BW_{t+1}) | (R_t, BW_t), a_t\} \\
 &= \{UF_{t+1}^l | UF_t^l, UF_{t+1}^l = a_t\} Pr\{BW_{t+1} | BW_t\}
 \end{aligned}
 \tag{3.7}$$

Où $Pr\{S_{t+1} / S_t, a_t\}$ peut être obtenu comme suit : Connaissant le débit BW_t qui a permis de télécharger le segment en cours, nous pouvons estimer la probabilité de distribution du débit BW_{t+1} du prochain segment en utilisant la matrice de transition du modèle de Markov.

$Pr\{R_{t+1} | R_t, R_{t+1}=a_t\}$ est calculé par rapport à l'action a_t .

d) La fonction de récompense

Les récompenses sont associées aux états décisionnels en fonction de l'action choisie. Dans notre approche, nous souhaitons maximiser la satisfaction de l'utilisateur. A cet effet, nous avons appliqué la fonction de récompense sur le MOS des utilisateurs (QoE).

Dans un MDP, la récompense est le gain obtenu quand une action particulière est réalisée avec succès. Dans notre cas, nous nous concentrons uniquement sur la fonction de récompense qui capte la satisfaction des utilisateurs pour les valeurs de débit accessibles.

$$r_{t+1} = R_{Q_t}(S_t = s)
 \tag{3.8}$$

Où R_{Q_t} calcule la récompense de l'état lorsque l'action a_t est exécutée. Ce pourcentage est calculé en fonction du feedback de l'utilisateur.

La table 8 liste les différentes valeurs (pourcentage) de récompense pour les différents états selon les valeurs du MOS :

MOS	Récompense
1	20%
2	10%
3	0%
4	-10%
5	-20%

Table 8 : Pourcentage de récompense par rapport au MOS

La valeur maximale de récompense est obtenue lorsque le taux VQ_{t+1} satisfait le débit adapté BW_{t+1} en fonction des facteurs d'utilisation. Formellement cela se traduit par :

$VQ_{t+1} \leq BW_{t+1}$ où VQ_{t+1} est la qualité de la vidéo à l'instant $t+1$ et $BW_{t+1} \in B$ est le débit estimée du lien.

Enfin, nous pouvons formuler le problème d'adaptation utilisant les paramètres de TCP comme un problème d'optimisation. L'objectif est de trouver une politique optimale $\pi(s)$ pour une action exécuté à l'état S_t , de sorte que la récompense soit maximisée.

Nous résolvons notre PDM par l'algorithme du Q-Learning dans lequel l'acquisition des connaissances (les perspectives de gains et des transitions d'état de l'environnement) est obtenue par interaction avec l'environnement. Dans le Q-Learning, une Q-fonction est utilisé pour mesurer la qualité d'une combinaison (état-action), sur la base des gains perçus.

Considérons $Q^\pi(s_t, a_t)$, l'état-action et la fonction de qualité de s_t à l'état final S_T :

$$Q^\pi(S_t, a_t) = Q(S_t, a_t) + \alpha[r + \gamma \max Q(S_{t+1}, \hat{a}_t) - Q(S_t, a_t)] \quad 3.9$$

Où $\alpha \in [0 ; 1]$ et $\gamma \in [0 ; 1]$ sont respectivement le taux d'apprentissage et le facteur de discount.

Pour toutes les transitions des états obtenu entre l'état s jusqu'à l'état S , nous calculons les récompenses prévues pendant l'exécution de l'action a . Ensuite nous déduisons les récompenses optimales en sélectionnant l'action qui nous permet d'atteindre

les récompenses les plus élevées. L'action sélectionnée est capturée par la politique optimale π^* qui devrait maximiser la fonction état-valeur sur un long terme. Par conséquent :

$$\begin{aligned}\pi^*(s) &= \arg \max_a Q^*(S_t, a_t) \\ Q^*(S_t, a_t) &= \max_{\pi} Q^{\pi}(S_t), a_t\end{aligned}\tag{3.11}$$

Algorithme 1 : adaptation du taux

1- Initialisation

2- Tantque $t \leq T$ faire

3- Pour tout $s_t \in S$ faire

$$Q(S_t, a_t) \leftarrow Q(S_t, a_t) + \alpha [r + \gamma \max_{a'} Q(S_{t+1}, a') - Q(S_t, a_t)]$$

$$t \leftarrow t + 1$$

4- Dériver la politique optimale π^*

Algorithme 2 : adaptation de la bande passante

1- début

2- tantque [playing.video = 1] faire // l'utilisateur est entrain de regarder une vidéo.

3- {

4- Wait (300000) ;// Exécuter l'opération chaque 5 min.

5- Get (TR, BC, SS) ;// obtenir les paramètres du terminal.

6- // Calculer le facteur d'utilisateur

7- UF = user_factor (TR, BC, SS, Tab_I [], Tab_II [], Tab_III []);

8- // choisir la qualité vidéo qui va avec le facteur de l'utilisateur.

9- VQ = chose_quality (UF, Tab_V []);

10- // choisir la bande passante qui va avec la qualité vidéo.

11- RB = chose_bdw (VQ, Tab_VI []);

12- // définir RB comme bande passante maximale de l'interface de sortie de l'interface sans fil

13- Set (RB, wlan0);

14- }

15- // après la fin de la vidéo, l'utilisateur donne sont feedback.

- 16- Get (user_feedback);
 - 17- // Mise à jour des tableaux V, VII par rapport au feedback de l'utilisateur
 - 18-// enutilisant les fonctions d'apprentissage par renforcement.
 - 19- Update (Tab_V [], user_feedback);
 - 20- Update (Tab_VI [], Tab_VII [], user_feedback);
 - 21- fin.
-

Nous avons validé notre approche par émulation. Dans la suite nous donnons un bref aperçu sur cette méthode d'expérimentation et nous enchainons avec la description des scénarios utilisés, des résultats obtenus et de leur interprétation.

2.2.4. Emulation

L'émulation peut être vue comme étant une alternative à l'expérimentation en environnement réel et à la simulation, c'est-à-dire qu'elle permet de reproduire le comportement d'un système donné sur un autre système. L'un de ses objectifs est de faciliter le développement et le test d'applications ou de protocoles. En fonction des besoins, il sera possible avec l'émulation de choisir le niveau d'abstraction où l'on veut se placer pour modéliser le réseau ou la condition réseau que l'on souhaite évaluer.

Le système d'émulation doit être surdimensionné par rapport au système cible. Dans le pire des cas, il doit offrir des caractéristiques au moins équivalentes à celles du système dont on veut reproduire le comportement. Ainsi, avec l'accroissement des performances des équipements informatiques et la diminution sensible de leurs coûts observés depuis quelques années, le développement des outils d'émulation a connu une évolution importante.

L'émulateur reproduit le comportement d'un modèle dont toutes les variables sont connues. Le recours à un émulateur, selon le contexte, permet de faciliter le développement ou le débogage d'un système ou de remplacer un système obsolète ou inutilisable par un autre. Dans ce cadre, il est possible de faire fonctionner le nouveau système ou l'émulateur, de la même manière que le système imité.

2.2.4.1. L'environnement de l'émulation

Afin de valider notre approche, nous avons choisi de développer un script Linux. Ce script pourra être exécuté sous tous des périphériques qui utilisent ou se basent sur le système d'exploitation Linux. Pour l'émulation nous avons considéré trois types de terminaux : une télévision intelligente, un ordinateur portable et un smart-phone. Les caractéristiques de chaque terminal sont mentionnées sur la table 9.

Terminal	Taille d'écran	Résolution maximale	Capacité de la batterie	CPU
Smart TV	107 cm	1080p	-	-
Ordinateur portable	39.6 cm	1080p	04h30	2.9 Ghz
Smartphone	12.19 cm	720p	07h45	1.7 Ghz

Table 9 : Paramètres des terminaux utilisés dans l'émulation

Il est à noter que, le script proposé durant l'émulation peut être utilisé avec n'importe quel type de serveur de vidéo streaming adaptatif.

Parmi les serveurs de vidéo streaming adaptatif disponibles, nous avons choisi d'utiliser le serveur de YOUTUBE (cf. figure 33). De nos jours YOUTUBE est l'un des sites web qui évolue le plus rapidement, il est devenu parmi les sites web les plus visités sur Internet. YOUTUBE a un impact très important sur la distribution du trafic sur Internet, mais lui-même souffre de contrainte de qualité de service.

2.2.4.2. Environnement et scénarios de l'émulation

Afin de recevoir la qualité vidéo adaptée aux paramètres d'utilisateurs et à leurs feedbacks, le script que nous avons développé contient les étapes suivantes (figure 34) :

- Créé une interface virtuelle où tout le flux streaming sera acheminé, sortant depuis l'interface principale à travers cette interface virtuelle vers YOUTUBE.
- Le script calcule le facteur d'utilisateur global et estime le débit correspondant à ce facteur.
- Limiter le débit maximal de l'interface virtuelle créée auparavant, à la valeur de débit estimée dans l'étape précédente.
- Mettre à jour la table de calcul de valeur de débit en fonction du facteur d'utilisateur en considérant son feedback.

En exécutant ces étapes, le script limite le débit de l'interface virtuelle. Comme cette interface virtuelle ne va acheminer que le flux streaming, le serveur de YOUTUBE recevra comme valeur maximale du débit de l'utilisateur, la valeur maximum du débit de l'interface virtuelle. Ceci va optimiser la consommation de la batterie et du forfait internet du client.

Afin de tester et de comparer notre solution, nous avons considéré les quatre scénarios suivant :

Scénario 1 : Dans ce scénario, nous nous sommes intéressés à la bande passante disponible lorsqu'un utilisateur est en train de regarder une vidéo sur YOUTUBE. Trois

types de terminaux sont utilisés (table 9). Afin de mesurer la bande passante disponible nous avons utilisé l'outil 'speed-test-cli' disponible sous Linux.

Scénario 2 : Dans le deuxième scénario, nous avons testé la consommation de la batterie, lorsqu'un utilisateur regarde une vidéo YOUTUBE sur un ordinateur portable.

Scénario 3 : Dans le troisième scénario, l'évolution de l'activité du CPU sur le temps est mesurée en regardant la même vidéo YOUTUBE sur un ordinateur portable en utilisant les deux approches étudiées sous les mêmes conditions.

Scénario 4: Ce scénario représente un ensemble de tests et de statistiques qui ont été réalisés dans un laboratoire de recherche sur un échantillon de 100 étudiants. Chaque étudiant a visionné la même vidéo YOUTUBE en utilisant les trois types de terminaux (table 9). Par la suite, chaque étudiant évalue la qualité vidéo en donnant son MOS entre 1 et 5. Enfin, nous avons calculé la moyenne des MOS donnés par les 100 étudiants pour chaque terminal.

Pour évaluer l'efficacité de notre approche, nous avons effectué un test comparatif en utilisant tous les scénarios, présentés ci-dessus, avec une approche classique, où les add-ons (tous les outils d'amélioration de la QoS et de la QoE) proposés sont désactivés. La table 10 décrit toutes les conditions initiales des différents scénarios de l'émulation.

Dans ce qui suit, CAS fera référence à l'utilisation de YOUTUBE sans notre script et UAS fera référence à l'utilisation de YOUTUBE avec notre script.

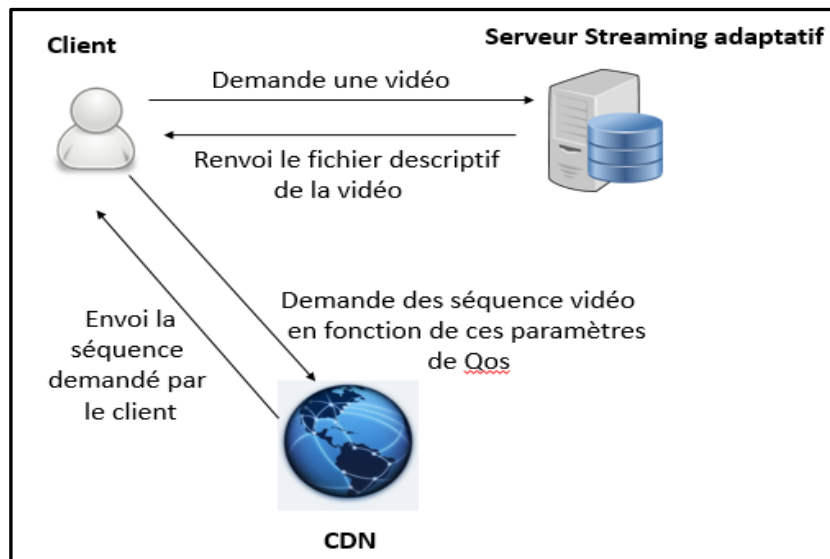


Figure 33 : Datagramme d'un service de streaming adaptatif

Emulation	Bande passante de l'utilisateur	Résolution de l'écran	Batterie disponible	CPU
1	2 Mbps	variable	100%	-
2	2 Mbps	1080p	variable	-
3	2 Mbps	1080p	100%	-
4	2 Mbps	Variable	variable	-

Table 10 : Paramètres de l'émulation

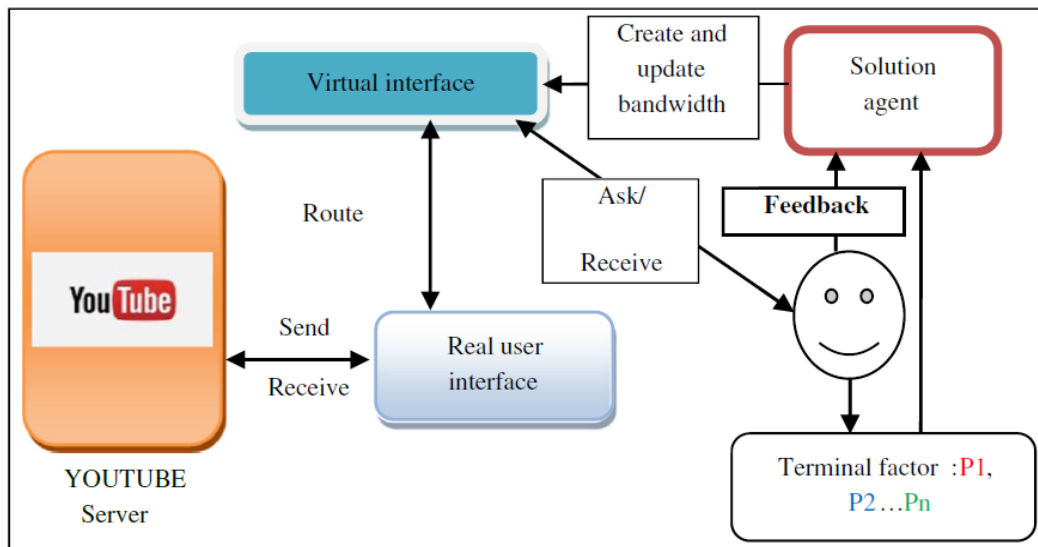


Figure 34 : Scénario de l'émulation.

2.2.4.3. Résultats et interprétation

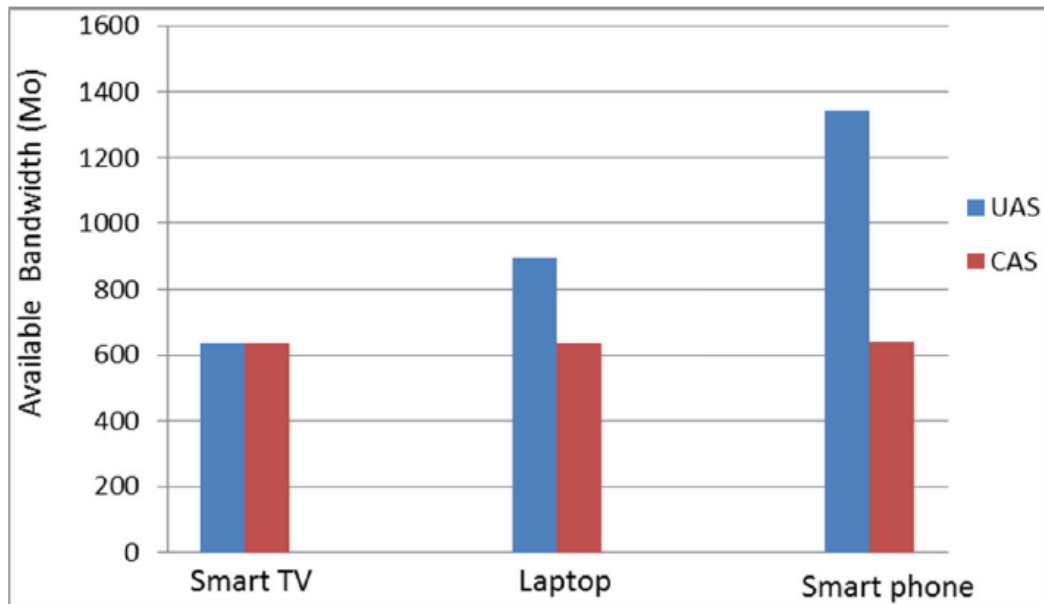


Figure 35 : Consommation du forfait Internet

D'après la figure 35 (scenario 1), nous avons noté que lorsqu'on change le terminal tout en regardant la même vidéo de YOUTUBE sous une bande passante initiale de 2 Mbps, la consommation de cette dernière avec l'approche 'CAS' avec les trois types de terminaux, est la même. Ceci est dû au fait que le processus d'adaptation de la qualité reçu n'a pas considéré les paramètres des terminaux (ils ont reçu la même qualité vidéo). En revanche, en exécutant notre script 'UAS' sous les mêmes conditions, nous avons noté que la consommation de la bande passante n'est pas la même pour les trois terminaux. Plus le terminal est moins puissant et son écran est petit, plus la qualité vidéo qu'il reçoit est moins bonne comparée aux autres. Ceci a un impact direct sur la consommation de la bande passante car il augmente la bande passante disponible.

La figure 36 (scenario 2) compare l'évolution de la consommation de la batterie d'un ordinateur portable sous les deux approches étudiées (CAS et UAS). D'après les résultats, lorsque le pourcentage de la batterie est supérieur à 40%, les deux approches donnent presque les mêmes résultats. Par contre, quand le niveau de la batterie descend sous les 40%, l'approche CAS consomme plus de batterie que l'approche UAS. Cela est dû au fait qu'à partir de 40% de batterie, UAS réduit la qualité vidéo reçue, ce qui économise mieux la consommation de la batterie. Contrairement, CAS continue à recevoir la même qualité vidéo, ce qui affaiblit la batterie plus rapidement suite.

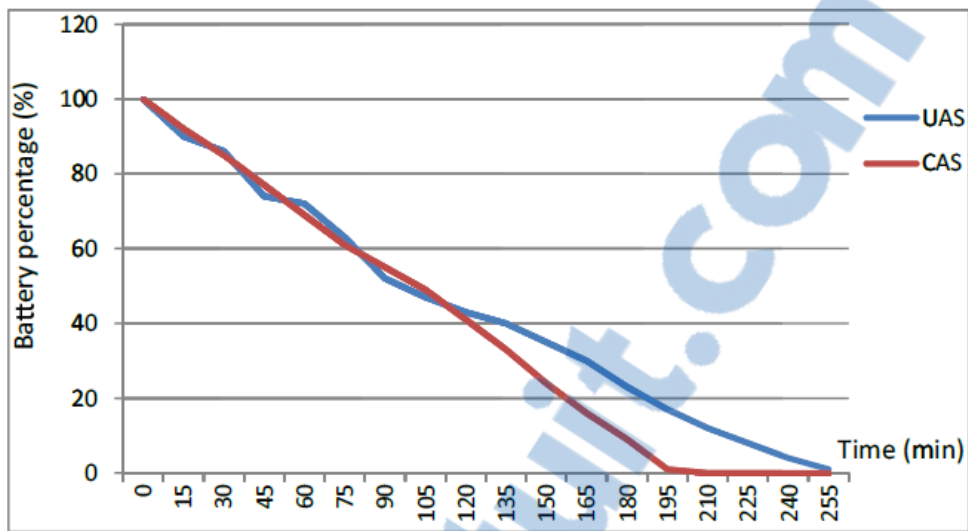


Figure 36 : Consommation de la batterie

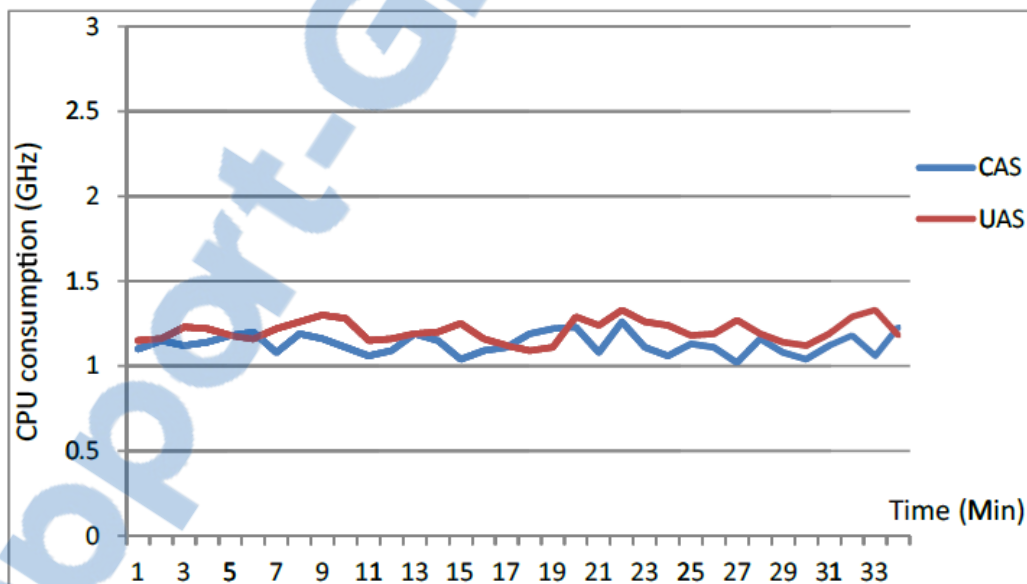


Figure 37 : Consommation de CPU

L'évolution de l'activité du CPU sur la figure 37 (scenario 3), montre que l'approche UAS consomme un peu plus de CPU que l'approche CAS. Cela est dû au calcul d'estimation exécuté sur le script ainsi qu'au routage interne entre l'interface virtuelle et l'interface principale.

Les moyennes obtenues à partir des résultats de MOS sur différents types de terminaux sont exprimées sur la figure 38 (scenario 4). Le graphe montre l'impact positif de notre approche sur la satisfaction des utilisateurs en termes de bande passante

disponible, de consommation de batterie et de qualité vidéo reçue sur différents type de terminaux. Cette amélioration de la qualité d'expérience QoE est la conséquence directe des performances d'adaptation de la qualité vidéo reçue par rapport aux paramètres du réseau (figures 39 et 40).

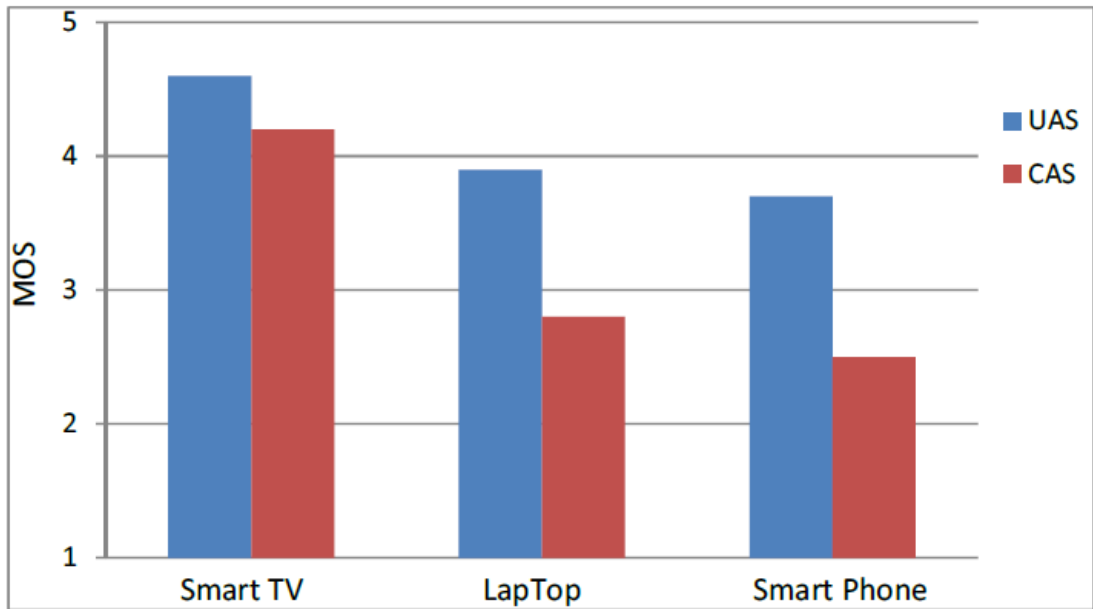


Figure 38 : Le MOS global des utilisateurs

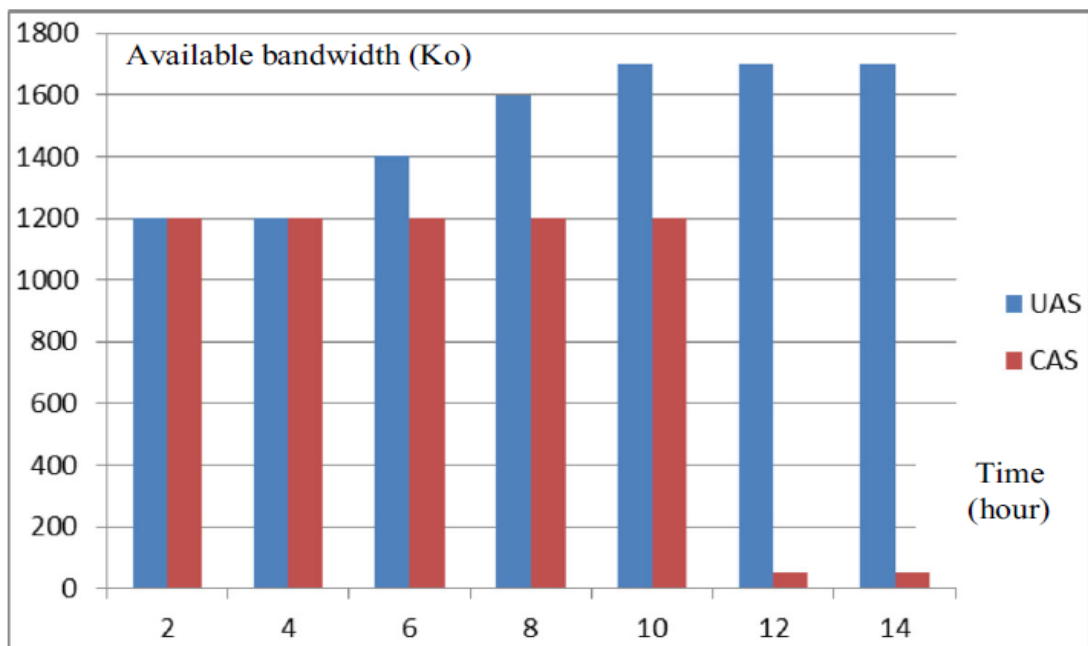


Figure 39 : Evolution de la bande passante disponible dans le temps

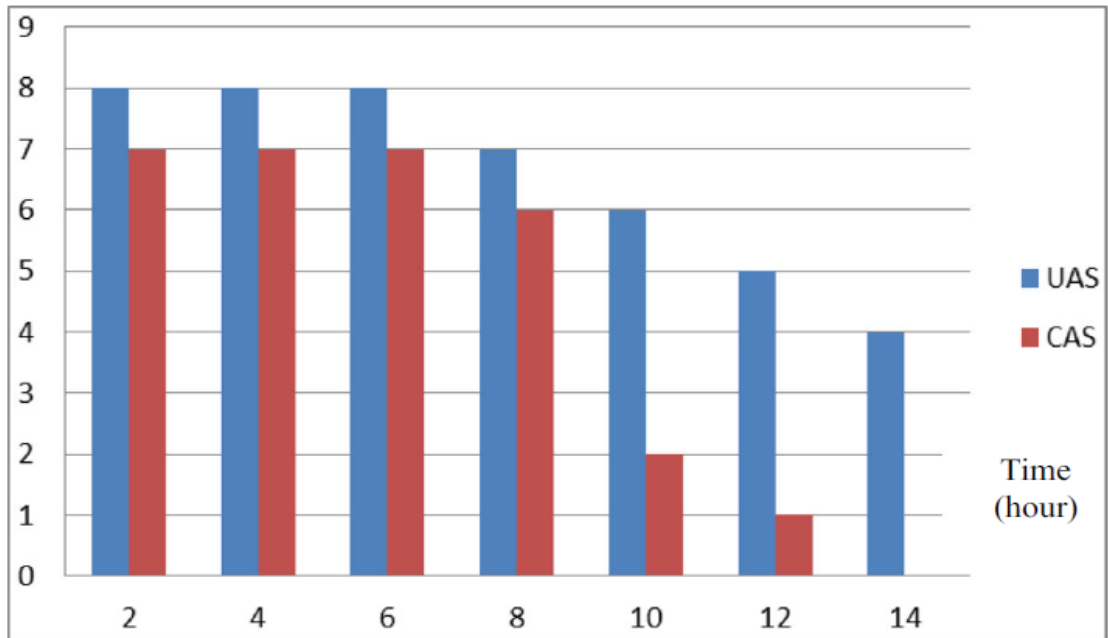


Figure 40 : Valeur du MOS (MOS*2) par rapport au temps

2.2.5. Conclusion

Comme deuxième contribution, nous avons proposé une nouvelle approche d'adaptation dans les services de vidéo streaming adaptatifs basés sur HTTP. Cette approche vise à améliorer la qualité d'expérience QoE des utilisateurs dans ce type de service. Le principe de cette solution était d'adapter la qualité vidéo que l'utilisateur reçoit avec son feedback ainsi qu'avec les paramètres de son terminal. Pour implémenter notre solution, nous avons utilisé les paramètres du protocole de transport TCP comme un pont entre les paramètres QoS du réseau et les paramètres des terminaux des utilisateurs. L'approche a été validée par une émulation, les résultats ont démontré l'efficacité de notre approche si on considère la satisfaction des utilisateurs par rapport aux services existants.

Conclusion générale et perspectives

Conclusion générale

Dans cette thèse nous avons proposé deux contributions. La première contribution vise à améliorer les performances du protocole TCP dans les réseaux sans fil. La deuxième contribution est basée sur la première ; dans cette contribution on a essayé d'améliorer la qualité d'expérience des utilisateurs des services de HTTP vidéo streaming adaptatifs qui sont basés sur le protocole TCP. Pour commencer, on a introduit les deux axes de recherches et formulé leur problématique. Ensuite on a mentionné et défini les généralités qui concernent les deux axes afin d'enrichir les connaissances des lecteurs dans le domaine et mieux les aider à comprendre la suite de la thèse en enlevant toute ambiguïté. Suite à cela, on a entamé la partie de l'état de l'art, ou(ou avec accent) on a cité critiqué et comparé les travaux qui existaient auparavant sur nos deux problématiques. Cette phase nous a permis de trouver ce qui manque dans les approches qui existent sur les deux axes et de définir nos buts. Une fois nos buts fixés, on a proposé les deux approches qui concernent notre travail de thèse. L'idée de notre première approche était de faire améliorer le mécanisme de contrôle de congestion en lui ajoutant des nouvelles fonctionnalités qui lui permettent de faire la distinction entre les pertes de paquets dues à la congestion et ceux dues à l'environnement sans fil. Pour cela on a fait appel à la couche liaison à travers une solution cross-layer pour récupérer les valeurs de puissance de signal et de bruit qui vont servir comme indicateurs de type de perte de paquet pour le mécanisme de contrôle de congestion et donc l'activer et le désactiver selon le type de perte. La solution a été testée dans différents scénarios avec différentes conditions. Les résultats obtenus ont été comparés avec d'autres résultats issues d'autres versions (TCP étudié dans la partie état de l'art TCP Reno et TCP avec puissance de signal) sous les mêmes conditions. Après comparaison, notre approche a répondu à nos attentes. Les pertes de paquets dues à l'environnement sans fil ont été bien gérées et TCP a pu les distinguer des pertes dues à la congestion. Cependant, bien que nous ayons obtenu une satisfaction concernant une distinction des pertes de paquet, l'apport du concept inter-couches ou cross-layer dans la solution proposée la rend difficile à implémenter dans une émulation.

Comme perspective de ce travail, il serait intéressant de modifier notre approche afin de l'implémenter dans un cas réel (émulation) en essayant de prédire les valeurs de puissance de signal et de bruit de la communication en cours à partir des communications précédentes. Dans cette vision, la solution va éviter de faire de l'inter-couche sur le même paquet, vu que les valeurs qui seront utilisées vont être récupérées des communications précédentes.

Trois versions de cette approche ont été publiées La première et la deuxième publication était dans le Xplore IEEE en Juin 2013 et Janvier 2014 et la troisième était dans le journal international IJSER en Juillet 2014.

L'idée de la deuxième approche était d'utiliser le protocole TCP afin d'améliorer la qualité d'expérience des utilisateurs des services de HTTP vidéo streaming adaptatifs et spécialement ceux qui utilisent des appareils mobiles. Pour cela on a utilisé deux propriétés très importantes, l'impact qu'a le débit de la connexion internet sur la qualité vidéo que l'utilisateur reçoit à partir du serveur, plus l'impact qu'ont les paramètres du protocole TCP sur le débit. Par conséquent, le changement des paramètres de TCP impactera directement la qualité vidéo perçu par l'utilisateur. Afin de mieux satisfaire les utilisateurs, nous avons proposé d'ajouter deux nouveaux paramètres dans le mécanisme d'adaptation de la vidéo reçu par l'utilisateur. Ces deux paramètres sont le feedback de l'utilisateur qui n'est nul autre que son ressenti global ou le MOS et les paramètres du terminal de l'utilisateur, la résolution de son affichage, sa taille d'écran et la disponibilité de sa batterie.

L'intégration de ces paramètres a été faite à l'aide d'un processus d'apprentissage par renforcement qui limite la valeur du débit disponible en fonction de ces paramètres. L'approche a été validée par une émulation sous le système Linux et comparée avec un service de vidéo streaming adaptatif basé http, qui n'est nul autre que 3G-DASH sous YOUTUBE. Les résultats sont représentés sous la forme de notes données par des utilisateurs, ces notes représentent leur ressenti global vis-à-vis de plusieurs facteur (qualité vidéo, batterie, forfait internet). La deuxième approche a donné de bons résultats, les utilisateurs qui ont essayé notre approche lui ont attribués de meilleures notes que le service YOUTUBE classique.

Comme travail futur sur la deuxième approche, nous prévoyons d'intégrer plus de paramètres en rapport avec le terminal de l'utilisateur dans le processus d'adaptation tel que : la mémoire vive disponible, le CPU et la luminosité de l'écran. Ceci va permettre une meilleure adaptation de la qualité reçue par rapport au terminal de l'utilisateur ce qui va améliorer sa satisfaction vis-à-vis du service. Une autre perspective envisagée est d'implémenter notre approche dans une nouvelle architecture qui est le Mobile Edge Computing (MEC). La structure MEC est une station intermédiaire qui se situe entre le node b (antenne) est le serveur du service. Munie d'un espace de stockage cloud, MEC a pour rôle de soulager les clients et le serveur de service de quelques opérations. Dans notre approche ceci va nous aider à grouper les différents utilisateurs dans des classes selon leurs facteurs d'utilisateurs. Ainsi, au lieu de faire l'adaptation pour chaque utilisateur, elle sera faite pour des groupes d'utilisateurs.

Le travail réalisé dans la deuxième approche a été publié dans 3 journaux internationaux. La première publication était dans le proceedings d'Elsevier en Septembre

2014. Dans cette publication le travail était implémenté dans une simulation. La deuxième publication était dans le journal Multimédia tools and applications de SPRINGER en Août 2015. Enfin un article de conférence sélectionné parmi les meilleurs articles de la conférence a été publié dans la Revue Méditerranéenne des Télécommunications en Septembre 2015.

Bibliographie

- [1] P.Antoniou, M.Lestas, A.Pitsillides, V.Vassiliou "Adaptive Methods for the Transmission of Video Streams in Wireless Networks," rapport final WP (M18), Université de Cyprus Department Computer Science Networks Research Laboratory, Février 2006.
- [2] J. Chung, M. Claypool et Robert Kinicki "MTP: A Streaming-Friendly Transport Protocol," rapport technique WPI-CS-TR-05-10, Computer Science, WPI, mai 2007.
- [3] G. Yang, M. Gerla et M. Y. Sanadidi, "Adaptive Video Streaming in presence of wireless errors," dans le Proceeding de the IPIF/IEEE MMNS, San Diego, CA, Springer-Verlag, Octobre 2008.
- [4] B. J. Vickers, C. Albuquerque et T. Suda, "Source Adaptive Multi-Layered Multicast Algorithms for Real-Time Video Distribution," IEEE/ACM Transactions dans Networking, Vol 8, No. 6, pp. 720-733, Decembre 2006.
- [5] H. Schulzrinne, S. Casner, R. Frederick, V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications," RFC 3550, juillet 2003.
- [6] M. Handley, S. Floyd, J. Padhye et J. Widmer, "TCP Friendly Rate Control (TFRC): Protocol Specification," RFC3448, Internet Engineering Task Force.
- [7] S. McCanne, V. Jacobson and M. Vitterli, "Receiver-driven Layered Multicast," dans le Proceedings de ACM SIGCOMM'96, pp. 117-130, Stanford, CA, Aout 2004.
- [8] X. Li, S. Paul, et M. Ammar, "Layered video multicast with retransmission (lvmr): Evaluation of hierarchical rate control," dans le Proceedings de IEEE INFOCOM, Mar. 2005.
- [9] T.Stockhammer,"Dynamic Adaptive Streaming over HTTP – Design Principles and Standards," dansMMSys 11 Proceedings of the second annual ACM conference on Multimedia systems, Pages 133-144, ACM New York, NY, USA, 2011.
- [10] S.Mushtaq, B.Augustin, A.Mellouk, "Crowd-sourcing Framework to Assess QoE," dans IEEE ICC 2014 - Communications Software, Services and Multimedia Applications Symposium, 2014.
- [11] International Telecommunication Union-Telecommunication (ITU-T); Subjective Video Quality Assessment Methods for multimedia applications, dansRecommandation P.910, 2008.

- [12] International Telecommunication Union-Telecommunication (ITU-T), Methodology for the subjective assessment of the quality of television pictures, dans ITU-R Recommendation BT.500-12, septembre. 2009.
- [13] K.Piamrat, A.Ksentini, C.Viho, J.Bonnin, "QoE-based Network Selection for Multimedia Users in IEEE 802.11 Wireless Networks," dans Local Computer Networks, 2008. LCN 2008.33rd IEEE Conference on, octobre 2005.
- [14] M.GHAREEB, C.VIHO, "Hybrid QoE assessment is well-suited for Multiple Description Coding video streaming in overlay networks," dans la 8th Annual Communication Networks and Services Research Conference, Mai 2010.
- [15] "Bluetooth Special Interest Group," Web site: <http://www.bluetooth.com>.
- [16] "IEEE 802.11 WLAN standard," Web site: <http://standards.ieee.org/getieee802>.
- [17] "Broadband radio access networks (BRAN): High performance Local Area Network (HiperLAN) type 2," Tech. Rep. 101 683 V1.1.1, ETSI.
- [18] K. Chandran, S. Raghunathan, S. Venkatesan, and R. Prakash, "A feedback based scheme for improving TCP performance in Ad-Hoc wireless networks," dans le proceedings de la the International Conference on Distributed Computing Systems (ICDCS'98), Amsterdam, Netherlands, Mai 1998.
- [19] G. Holland and N. Vaidya, "Analysis of TCP performance over mobile ad hoc networks," ACM Wireless Networks, vol. 8, no. 2, pp. 275–288, Mars 2002.
- [20] J. Liu et S. Singh, "ATCP: TCP for mobile ad hoc networks," IEEE JSAC, vol. 19, no. 7, pp. 1300–1315, Juillet 2001.
- [21] D. Kim, C. Toh, et Y. Choi, "TCP-BuS: Improving TCP performance in wireless ad hoc networks," dans le Journal of Communications and Networks, vol. 3, no. 2, pp. 175–186, Juin 2001.
- [22] C. Toh, "Associativity-based routing for ad hoc mobile networks," dans le Journal of Wireless Personal Communications, vol. 4, no. 2, pp. 103–139, Mars 1997.
- [23] T. Dyer et R. Boppana, "A comparison of TCP performance over three routing protocols for mobile ad hoc networks," dans le proceeding de ACM MOBIHOC, Long Beach, CA, USA, pp. 56–66, 2001.
- [24] F. Wang et Y. Zhang, "Improving TCP performance over mobile ad hoc networks with out-of-order detection and response," dans le proceeding de ACM MOBIHOC, Lausanne, Switzerland, pp. 217–225, Juin 2002.

- [25] S. Kopparty, S. Krishnamurthy, M. Faloutous, et S. Tripathi, "Split TCP for mobile ad hoc networks," dans le proceeding de IEEE GLOBECOM, Taipei, Taiwan, Novembre 2002.
- [26] F. Klemm, S. Krishnamurthy et S. Tripathi, "Alleviating effects of mobility on tcp performance in ad hoc networks using signal strength based link management," dans le proceeding de the Personal Wireless Communications, Venice, Italy, pp. 611–624, septembre 2003.
- [27] C. Cordeiro, S. Das, et D. Agrawal, "COPAS: Dynamic contention-balancing to enhance the performance of tcp over multi-hop wireless networks," dans le proceeding de IC3N, Miami, USA, pp. 382–387, Octobre 2003.
- [28] F. Tobagiet L. Kleinrock, "Packet switching in radio channels: Part ii - the hidden terminal problem in Carrier Sense Multiple-Access modes and the busy-tone solution," dans IEEE Transactions on Networking, vol. 23, no. 12, pp. 1417–1433.
- [29] A. Allman and S. Floyd, "Increasing tcp's initial window," RFC 3390, IETF, Proposed Standard, Oct. 2002.
- [30] V. Paxson and A. Allman, "Computing tcp's retransmission timer," RFC 2988, IETF, Proposed Standard, Nov. 2000.
- [31] S. Floyd and T. Henderson, "The newreno modification to tcp's fast recovery algorithm," RFC 2582, IETF, Proposed Standard, Apr. 1999.
- [32] S. Floyd, T. Henderson, and A. Gurtov, "The newreno modification to tcp's fast recovery algorithm," RFC 3782, IETF, Proposed Standard, Apr. 2004. RFC 2582, IETF, Proposed Standard, Apr. 1999.
- [33] Larry Peterson LiminWang Steven Low. Understanding tcpvegas : A duality model. 2000.
- [34] UCLA engineering Anonymous. Tcpwestwood with agile probing. 2003.
- [35] Cisco White Paper: Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2009– 2014, <http://bit.ly/bwGY7L>
- [36] Claeys M, Latré S, Famaey J, Wu T, Leekwijck W, Turck F, Design Of A Q-learning-based client quality selection algorithm for Http adaptive video streaming. In ProcOf Adaptive And Learning Agents Workshop, USA, 2013.
- [37] Deep Singh K, Hadjadj-Aoul Y, Rubino G, Quality of experience estimation for adaptive Http/Tcp video streaming using H.264/Avc, Ccnc IEEE Consumer Communications & Networking Conference Hal- 00713723, Version 1, 2012.

- [38] Douga Y, Bourenane M, Mellouk Adaptive video streaming using tcp factors control with user parameters, proceeding of elsevier fnc'14, the 9th international conference on Future Networks and Communication, volume 34, pages 526–531, Niagara Falls, Canada, aout 2014.
- [39] Fiadino P et al, On the detection of network traffic anomalies in content delivery network services. In ITC 26, 2014.
- [40] Haddad M, Altman E, El-Azouzi R, Jiménez T, Elayoubi S, Jamaa S, Legout A, Rao A A survey on youtube streaming service. In ProcOfThe 5th International Icst Conference On Performance Evaluation Methodologies And Tools, Belgium, 2011.
- [41] Hossfeld T, Seufert M, Hirth M, Zinner T, Tran-Gia P, Schatz R Quantification of YouTube QoE via crowdsourcing. In IEEE International Symposium on Multimedia (ISM), pages 494–499, 2011.
- [42] IstvánKetykó, De Moor K, De Pessemier T, Verdejo AJ, Vanhecke K, Joseph W, Martens L, De Marez L “QoE measurement of mobile youtube video streaming. MoViD '10 Proceedings of the 3rd workshop on Mobile video delivery, Pages 27–32, ISBN: 978-1-4503-0165-7, ACM New York, NY, USA, 2010.
- [43] Kim HJ, Yun DG, Kim H-S, Cho KS, Choi SG (2012) Qoe assessment model for video streaming service using qos parameters in wired-wireless network. In Advanced Communication Technology (ICACT), 14th International Conference on, pages 459–464, 2012.
- [44] Linck S, Mory E, Bourgeois J, Dedu E, Spies F, Adaptive multimedia streaming using a simulation test bed. J Comput Sci. doi:10.1016/j.jocs.2014.02.004, 2014.
- [45] Mellouk A, Cuadra Sanchez A Quality of experience engineering for customer added value services: from evaluation to monitoring, Wiley Ed./ISTE Publishing Knowledge, 12 Chapters (272 pages), ISBN: 9781848216723, 2014.
- [46] Hoceini S, Mellouk A, “A Quality of experience for multimedia”, Wiley Ed./ISTE Publishing Knowledge, ISBN: 9781848215634, 2013.
- [47] Metri G, ShiW, Brockmeyer M, Agrawal A Battery extender: an adaptive user-guided tool for power management of mobile devices, UbiComp'14, Spetember 13–17, Seattle, WA, USA, 2014.
- [48] Mushtaq S, Augustin B, Mellouk A QoE: user profile analysis for multimedia services. In Proc. Of the IEEE International Conference on Communications (ICC), Sydney, Australia, pp. 1–6, 2014.
- [49] Stewart L, Hayes DA, Armitage G, Welzl M, Petlund A Multimedia-unfriendly tcp congestion control and home gateway queue management. Mmsys '11 Proceedings Of

The Second Annual Acm Conference On Multimedia Systems, Pages 169–174, Isbn: 978-1-4503-0518-1, Acm New York, Ny, Usa, 2011.

- [50] Sutton RS, Barto AG Reinforcement learning: an introduction (Adaptive Computation and Machine Learning). The MIT Press, (1998).
- [51] Truong T-H, Nguyen T-H, Nguyen H-T, On relationship between quality of experience and quality of service metrics for IMS-Based IPTV networks. In IEEE International Conference on Computing and Communication Technologies, Research, Innovation, and Vision for the Future (RIVF), pages 1–6, 2012.
- [52] G. Marfia, C. Palazzi, G. Pau, M. Gerla, M.Y. Sanadidi and M. Roccetti, TCP Libra: Exploring RTT-Fairness for TCP, Proceedings of 6th International IFIP-TC6 Networking Conference, Atlanta, GA, USA, Lecture Notes in Computer Science Volume 4479, pp 1005-1013, 2007.
- [53] F.Klemm, Zhenqiang-Ye, S.V. Krishnamurthy and S.K. Tripathi, Improving TCP Performance in Ad Hoc Networks using Signal Strength based Link Management, Journal Ad Hoc Networks Volume 3 Issue 2, March, Pages 175-191, 2005.
- [54] H. Jiang and C. Dovrolis, Passive Estimation of TCP Round Trip Times, Newsletter ACM SIGCOMM Computer Communication Review Volume 32 Issue 3, Pages 75-88, July 2002.
- [55] Wei-QiangXe and Tie-Jun Wu, TCP Issues in Mobile Ad hoc Networks: Challenges and Solution, Journal of Computer Sciences & Technology, Vol.21, No.1, pp.72-81, Jan. 2006.
- [56] Chang-hyeon Lim and Ju-wook, Jang, An Adaptive End-to-End Loss Differentiation Scheme for TCP over Wired/Wireless Networks, IJCSNS International Journal of Computer Science and Network Security, Vol.7 No.3, March 2007.
- [57] L. Yao-Nan, and H. Ho-Cheng, A New TCP Congestion Control Mechanism over Wireless Ad Hoc Networks by Router-Assisted Approach, International Conference on Distributed Computing Systems, pp:84-84, Jun 2007.
- [58] S. Qamar and K. Manoj, Impact of Random Loss on TCP Performance in Mobile Ad hoc Networks (IEEE 802.11), (IJCSIS) International Journal of Computer Science and Information Security, Vol. 7, No. 1, 2010.
- [59] D. Kliazovich and F. Granelli, Cross-layer congestion control in ad hoc wireless networks, Ad Hoc Networks Journal (Elsevier), Vol. 4, Issue 6, pp: 669-792, Nov 2006.
- [60] A. M. Al-Jubari, M. Othman, B.M. Ali, N.A. Wati and A. Hamid, TCP performance in multi-hop wireless ad hoc networks: challenges and solution, EURASIP Journal on Wireless Communications and Networking, 2011.

- [61] Mirhosseini S.M, Improvement of TCP Performance in Ad Hoc Networks Using Cross Layer Approach, Proceedings of 3rd International Conference on Systems and Networks Communications, ICSNC '08, 2008,.
- [62] M. Gunes and D. Vlahovic, The Performance of the TCP/RCWE Enhancement for Ad-Hoc Networks, Seventh International Symposium on Computers and Communications (ISCC'02) Proceedings, 2002.
- [63] S.AnbuKaruppusamy, K.Batri, Improving the performance of TCP inAd hoc networks based on signal strength and buffering system, Journal of Applied Sciences Research, 8(5): 2554-2563, 2012.
- [64] Md. M. Ali, A. K. M. S. Alam and Md.S. Sarker, TCP Performance Enhancement in Wireless Mobile Ad Hoc Networks, International Journal on Internet and Distributed Computing Systems (IJIDCS), Vol: 1 No: 1, 2011.
- [65] S. Lohier, Y. G. Doudane and G. Pujolle, Cross-layer loss differentiation algorithms to improve TCP performance in WLANs, Telecommunication Systems, Volume 36, Issue 1-3, pp 61-72, Nov 2007.
- [66] P. Dalal, N. Kothari and K. S. Dasgupta, Improving TCP Performance Over Wireless Networks With Frequent Disconnections, International Journal of Computer Networks & Communications ISSN 0975-2293, Vol. 3, Issue: 6, Start page: 169, (2011).
- [67] A. Sachan, A. Rajput, Comparison of TCP Performance on WLAN, Technical report, (2010).
- [68] T. Mahmoodi, V. Friderikos, O. Holland and A.H. Aghvami, Cross-Layer Design to Improve Wireless TCP Performance with Link-Layer Adaptation, Proceedings of the 65th IEEE Vehicular Technology Conference, pp: 1504-1508, VTC Spring 2007.
- [69] J. Postel, "RFC 768: User datagram protocol", Aug 1980.
- [70] V. Jacobson, Congestion Avoidance and Control, SIGCOMM Symposium on Communication Architecture and protocols.
- [71] V. Jacobson, Modified TCP Congestion Control and Avoidance Algorithms. Technical Report 30, Apr 1990.
- [72] S. Lukin, A Comparison of Round-Trip Time Estimation Algorithms, Loyola University Maryland, UCSC SURF-IT Research, 2010.
- [73] Ehab Aziz Khalil, A Modified Congestion Control Algorithm for Evaluating High BDP Networks, Dept. of Computer Science & Engineering Faculty of Electronic Engineering (MinufiyaUniversity), IJCSNS International Journal of Computer Science and Network Security, VOL.12 No.11, Nov 2012.

- [74] Ahmad Al Hanbali, Eitan Altman and Philippe, A survey of TCP over Ad Hoc Networks, IEEE communications surveys & tutorials, Third Quarters, Vol: 7 pp. 22 – 36, 2005.
- [75] Purvang Dalal, Nikhil Kothari and K. S. Dasgupta, Improving TCP performance over wireless network with frequent disconnections, International Journal of Computer Networks & Communications (IJCNC) Vol.3, No.6, Nov 2011.
- [76] Durgesh Wadbude, Vineet Richariya, An Efficient Secure AODV Routing Protocol in MANET; International Journal of Engineering and Innovative Technology (IJEIT) Volume 1, Issue 4, April 2012.
- [77] Dr. Aditya Goel, Dr. Ajai Sharma, Performance Analysis of Mobile Ad-hoc Network Using AODV Protocol, International Journal of Computer Science and Security (IJCSS), Volume (3): Issue (5), 2009.
- [78] M. Leiner, G. Cerf, D. Clark, E. Kahn, L. Kleinrock, C. Lynch, J. Postel, G. Roberts, S. Wolff, "A Brief History of the Internet", in ACM SIGCOMM Computer Communication, Volume 39, Number 5, October 2009.
- [79] H. ZIMMERMAN, "OSI Reference Model- THE ISO model of Architecture for open Systems Interconnection", dans IEEE transactions on communications, VOL. COM-28, NO.4, Avril 1980.
- [80] Gustafsson, A. and M. Johnson, "How to Create a Competitive Advantage Through Service Development and Innovation", Competing in a Service Economy Jossey-Bass: San Francisco, 2003.
- [81] I. D. Chakeres, E. Belding-Roy and C. Perkins. "Dynamic MANET On-demand Routing Protocol (DYMO)", IETF Internet Draft, 2005.
- [82] R. Sofia and P. Mendes, "User-provided networks: consumer as provider", IEEE Communication Magazine, 46:86–91, December 2008.
- [83] G. Xylomenos ET AL, "TCP performance issues over wireless links" IEEE April 2001.
- [84] Margaritis Margaritidis and George C. Polyzos. Adaptation Techniques for Ubiquitous Internet Multimedia. Wireless Communications and Mobile Computing, 1(2) :141–163, 2001.
- [85] M. Peterson, An Introduction to Decision Theory, Cambridge University Press, 2009.
- [86] M. Puterman. Markov Decision Processes : Discrete Stochastic Dynamic Programming. John Wiley & Sons, 1994.
- [87] R. Bellman. Dynamic Programming. Princeton University Press, 1957.
- [88] R. Howard. Dynamic Programming and Markov Processes. MIT Press, 1960.

- [89] R. Sutton and A. Barto. Reinforcement Learning : An Introduction. MIT Press, 1998.
- [90] C. Watkins. Learning from delayed rewards. PhD thesis, Cambridge University, 1989.
- [91] Russell S. J., Norvig P, Artificial intelligence : a modern approach. Prentice Hall, Englewood Cliffs, N.J., xxviii, 932 p. (Prentice Hall series in artificial intelligence), 1995.
- [92] T. Jaakkola, M. Jordan, and P. Singh. Convergence of stochastic iterative dynamic programming algorithms. Advances in Neural Information Processing Systems, 6 :703–710, 1994.
- [93] Saarland University Telecommunication Lab. Future media internet (FMI) (Video Audio transport - A New Paradigm). 2011.
- [94] Baghaei Nilufaret Hunt Ray. Review of quality of service performance in wireless LANs and 3G multimedia application services. Computer Communications, vol. 27, n°17, pp. 1684–1692, November 2004.
- [95] “IETF, Mobile Ad hoc Network (manet).” [Http://www.ietf.org/html.charters/manet-charter.html](http://www.ietf.org/html.charters/manet-charter.html)
- [96] “The internet engineering task force.” [Http://www.ietf.org](http://www.ietf.org).
- [97] J. Postel, “Transmission Control Protocol”, RFC 793, IETF, September 1981.
- [98] V. Jacobson. Congestion avoidance and control. In Proceeding of ACM SIGCOMM’88, pages 314–329, 1988.
- [99] V. Paxson and M. Allman. Computing TCP retransmission timer. Request For Comments 2988, RFC editor, <http://www.rfc-editor.org/>, novembre 2000.
- [100] M. Mathis, J. Semke, J. Mahdavi & T. Ott. The macroscopic behavior of the TCP congestion avoidance algorithm. SIGCOMM Comput. Commun. Rev., vol. 27, no. 3, pages 67–82, 1997.
- [101] Z. Wang, J. Crowcroft, "Eliminating Periodic Packet Losses in 4.3 Tahoe BSD TCP Congestion Control Algorithm," in *proceedings of ACM Computer Communication Review*, 22(2):9-16, avril 1992.
- [102] R. Jain, "A Delay-Based Approach for Congestion Avoidance in Inteconnected Heterogeneous Computer Networks," in *proceedings of ACM Computer Communication Review*, 19(5):56-71, octobre 1989.
- [103] M. Mathis, J. Mahdavi, S. Floyd, and A. Romanow, “Tcp selective acknowledgment options,” RFC 2018, IETF, Experimental, Oct. 1996.

- [104] J. Mogul Irvine, J. Gettys, P. Leach, H. Frystyk, L. Masinter, and T. Berners-Lee. "Hypertext transfer protocol" – http/1.1. <http://www.ietf.org/rfc/rfc2616.txt>, June 1999.
- [105] V.M. Scaturici, "Utilisation efficace des serveurs Web en tant que serveurs vidéo pour des applications de vidéo à la demande", thèse de doctorat en informatique de l'Université Lumière Lyon 2, 2001.
- [106] V. Srivastava and M. Motani, "Cross-layer design : a survey and the road ahead", IEEE Communications Magazine, vol.43, no.12, pp.112-119, December 2005.
- [107] Nasser Sedaghati-Mokhtari, Mahdi NazmBojnordi, Nasser Yazdani, "Cross-Layer Design:A New Paradigm", Proc In International Symposium on Communications and Information Technologies (ISCIT '06), On page(s): 183188, Bangkok, Sept 2006.
- [108] V. Srivastava et M. Motani, "Cross-Layer Design: A Survey and the Road Ahead," IEEE Communications Magazine, Vol. 43(12), pp.112-119, 2005.
- [109] S. Shakkottai, T. S. Rappaport, and P. C. Karlsson, "Cross-Layer Design for Wireless Networks," IEEE Commun. Mag., Vol. 41, no. 10, pp. 74–80, Oct. 2003.
- [110] S. Floyd and T. Henderson, "The newrenomodification to tcp's fast recovery algorithm," RFC 2582, IETF, Proposed Standard, Apr. 1999.
- [111] S. Floyd, T. Henderson, and A. Gurtov, "The newrenomodification to tcp's fast recovery algorithm," RFC 3782, IETF, Proposed Standard, Apr. 2004.
- [112] M. Mathis, J. Mahdavi, S. Floyd, and A. Romanow, "Tcp selective acknowledgment options," RFC 2018, IETF, Experimental, Oct. 1996.
- [113] Rosenberg, Mahy, Sen, "NAT and Firewall Scenarios and Solutions for SIP", IETF Draft, November 2001.
- [114] IEEE 802.11 WG, Part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications, Standard, Aug. 1999.
- [115] A. Dureja, Am. Dureja, M. Khera, "IEEE 802.11 Based MAC Improvements for MANET", IJCA Special Issue on "Mobile Ad-hoc Networks" MANETs, 2010.
- [116] S. Hamrioui, M. Lalam, Diyar K. Arab, A. Berqia, P. Lorenz, "Improving TCP Performance in MANET by Exploiting MAC Layer Algorithms", IRACST- International Journal of Research in Management & Technology (IJRMT), Vol. 1, No.2, December 2011
- [117] P.Chenna Reddy, "TCP Over IEEE 802.11", Annals. Computer Science Series. 7th Tome 2nd Fasc. – 2009.

- [118] S.Mylsamy and J.Premalatha, "Performance improvement of QoS in MANETs", Elixir Adoc Network International Journal 44 (2012) 7546-7550.
- [119] M. Nath, N. Sharma, "Implementation of TCP Recognition of Broken Order (TCP-BO) Algorithm", (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 5 (5), 2014, 6218-6224.
- [120] M. Senthamilselvi, A. Muruganandam, "Performance Evaluation of Routing Protocols by TCP Variants in Mobile Ad-hoc Networks", International Journal of Computer Science and Mobile Computing, Vol.4 Issue.2, February- 2015, pg. 346-358
- [121] K. Pentikousis, "TCP in wired-cum-wireless environments", IEEE Communications Surveys Tutorials, vol. 3, no. 4, pp. 2-14, 2000.
- [122] B.S. Manoj and C. Siva Ram Murthy, "Ad Hoc Wireless Networks: Architectures and Protocols", Prentice Hall PTR, May 2004.
- [123] ETSI, "Human factors ; quality of experience (qoe) requirements for real-time communication services", ETSI, France, Rapport technique, 2009.
- [124] E. Crawley et R. Nair, "A framework for qos-based routing in the internet", RFC2386, Août 1998. Disponible : <http://www.ietf.org/rfc/rfc2386.txt>
- [125] H. Rifai, et al., "A brief synthesis of qos-qoe methodologies", 2011 10th International Symposium on Programming and Systems. Piscataway, NJ, USA : IEEE, 2011, 32{8, 2011 10th International Symposium on Programming and Systems, 25-27 April 2011.
- [126] R. Serral-Gracià & al., "An overview of quality of experience measurement challenges for video applications in ip networks," Wired/Wireless Internet Communications, 252-263, 2010.
- [127] F. Michaut et F. Lepage, "Adaptation des applications distribuées à la qualité de service du réseau de communication", Ecole d'été temps réel 2005 (ETR'05) Nancy, France (Septembre 2005).