

République Algérienne Démocratique et Populaire
Ministère de L'Enseignement Supérieure et de la Recherche Scientifique
Université Saad Dahlab Blida 1

N°D'ordre :



Faculté des sciences

Département d'informatique

Mémoire Présenté par :

TELDJOUNE AZEDDINE

BOUSSETTA YAHIA

En vue d'obtenir le diplôme de master

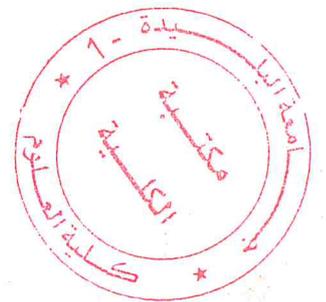
Domaine : Mathématique et informatique

Filière : Informatique

Option : système informatique et réseaux

Sujet :

Amélioration de la qualité d'expérience des services de vidéo
streaming adaptatif dans une architecture MEC



Devant le jury :

Mr. YKLEF

Président

Mr. BENAISSI

Examineur

Mr. DOUGUA Yassine

Promoteur

Promotion 2017 /2018

Remerciements

Nous tenons tout d'abord à remercier le bon dieu, qui, grâce à lui on a pu accomplir ce modeste travail.

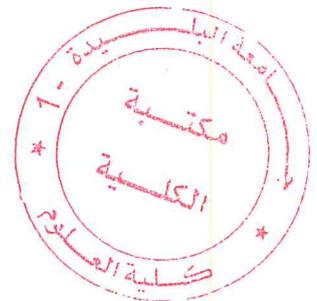
Ensuite, ce mémoire n'aurait pas été réalisé sans le soutien moral et les prières de nos parents, c'est à eux que nous dédions notre travail.

Sans oublier de témoigner nos reconnaissances, gratitude et remerciements à notre promoteur Mr Y.DOUGUA pour son orientation, son soutien et surtout sa disponibilité et le temps qu'il a consacré au bon déroulement du travail. Merci !

Egalement nous remercions nos collègues, tous les étudiants de deuxième année Master Informatique (SIR) promotion 2017_2018, nos frères et sœurs et ainsi nos amis(es) pour leurs encouragements.

Nos remerciements vont aussi aux membres de jury pour l'intérêt qu'ils ont porté à notre recherche, pour l'examiner et l'enrichir avec leurs précieuses propositions.

Enfin, merci à toutes les personnes qui ont contribué de près ou de loin à la réalisation de notre travail.



Dédicace

Je dédie ce travail à tous les membres de ma très chère famille, mes amis et mes collègues, en particulier :

A Mon cher père qui m'a appris la confiance en soi et l'autonomie que Dieu le tout puissant te préserve et t'accorde santé, A ma mère qui était toujours à mes cotés, tu m'as comblé avec ta tendresse, tes prières ont toujours guidé mes pas tout au long de mon parcours que le bon Dieu te donne une longue vie.

Mes sœurs Imane, Rachida et fatima

Mes frères Noureddine, mahfoud, mohamed et abdelouahab

A tous les membres de la famille Feldjouné et Sadari

A tous mes amis salah, mohamed lebahi, kheireddine

Et à mon metre Himeur Boualam qui je vien de perdre

Dédicace

Je dédie ce travail à tous les membres de ma très chère famille, mes amis et mes collègues, en particulier :

A Mon cher père et ma chère mère.

Mes sœurs et Mes frères

A tous les membres de la famille Boussetta

A tous mes amis salah, sedik, kheireddine, lotfi

ملخص:

أدى ظهور خدمات الوسائط المتعددة ونموها السريع في الشبكات اللاسلكية إلى ظهور تحديات لمقدمي الخدمات و مشغلي الشبكات الذين يستمدون من نطاق جودة الخدمة (QoS) المعلومات التقنية لشبكاتهم. لقد تم بذل الكثير من الجهود في إعداد مقاييس جديدة تعكس دقة جودة الخدمة المقدمة , يسمى هذا القياس بجودة التجربة (QoE), لذلك عادة ما تستخدم أدوات جودة التجربة بالتوازي مع أدوات جودة الخدمة, إنها تجعل من الممكن التحسين من تدهور الخدمة من وجهة نظر المستخدم, وبالتالي إطلاق تحليلات بهدف يؤدي إلى إجراء تحسينات في الخدمة. وبالمثل فإنها تسمح للمهندسين (خدمات التلفزيون, الصوت...) أو المدربين الفنيين أو التجاريين بتوجيه أبحاثهم مسبقا. وبالتالي التقليل من الوقت اللازم لحل مشكلة ما

في عملنا اخترنا أن نطبق تطبيقنا على خدمات تدفق الفيديو المتوائم (HAS) في بنية شبكة MEC لتحسين تجربة المستخدم (QoE). يعد تدفق HTTP المتلائم (HAS) تقنية بث فيديو يستخدم على نطاق واسع عبر الإنترنت, لكل من طلب الفيديو (VoD) أو خدمات البث المباشر, ويستخدم بروتوكول TCP كبروتوكول للنقل, يتم تقسيم الفيديو الأصلي في server إلى عدة أجزاء ذات نفس المدة, يتم تحويل هذه الأجزاء إلى عدة مستويات من الجودة. الفكرة الأساسية ل (MEC) هي تشغيل تطبيقات وتنفيذ مهام المعالجة أقرب إلى المستخدم, نتيجة لذلك, يتم التقليل من ازدحام الشبكة والتطبيقات تعمل بشكل أفضل. في هذه المذكرة اقترحنا حلا لتحسين جودة التجربة في خدمات البث الفيديوي المتركز على MEC والذي يضيف التحكم في الازدحام إلى عملية تحسين جودة الفيديو.

كلمات مفتاحيه: جودة التجربة (QoE) ، بث الفيديو ، الحوسبة المتنقلة (MEC) ، الازدحام ، احتمال الاحتقان (PC) ، الخادم.

Résumé :

L'émergence et la croissance rapide des services multimédias dans les réseaux sans fils ont créé de nouveaux défis pour les fournisseurs des services et les opérateurs réseaux, qui, au-delà de la Qualité de Service (QoS) issue des paramètres techniques de leurs réseaux, beaucoup d'efforts ont été déployés pour mettre en place une nouvelle métrique qui reflète, de façon plus précise la qualité du service offert. Cette mesure s'appelle la qualité d'expérience (Quality of Experience). Donc, Les outils de Qualité d'Expérience sont en général utilisés en parallèle des outils de Qualité de Service. Ils permettent de mettre en avant une dégradation du service d'un point de vue d'utilisateur et donc, de déclencher des analyses avec l'objectif d'aboutir à des actions de remédiation. De même, ils permettent aux ingénieurs (services TV, voix...), ou aux responsables techniques ou métiers (Applicatifs en entreprise) de pré-orienter leurs recherches et donc de réduire le temps de résolution d'un problème.

Dans notre travail, nous avons choisi d'appliquer notre contribution sur les services de vidéo streaming adaptatif HTTP (HAS) dans une architecture réseau MEC afin d'améliorer la qualité d'expérience QoE des utilisateurs. Le streaming adaptatif HTTP (HAS) est un type de service de vidéo streaming largement utilisé sur internet que ça soit pour la vidéo à la demande (VoD) ou pour les services de streaming en direct. Elle utilise le protocole TCP en tant que protocole de transport. Dans ce genre de service, la vidéo d'origine est divisée dans le serveur en plusieurs segments de même durée. Ces segments sont transcodés en plusieurs niveaux de qualité. L'idée de base de l'architecture MEC est d'exécuter des tâches du serveur plus près de l'utilisateur sur une station MEC. La congestion du réseau est l'une des raisons de dégradation des performances du réseau ainsi que la QoE. Dans ce mémoire, nous avons proposé une solution pour l'amélioration de la qualité d'expérience dans les services de vidéo streaming basée sur l'architecture MEC, qui ajoute le contrôle de congestion au processus d'adaptation de la qualité vidéo.

Mots clés : Qualité d'expérience(QoE), vidéo Streaming, mobile edge computing (MEC), Congestion, Probabilité de Congestion(PC), Serveur.

Abstract:

The emergence and the fast growth of multimedia services in wireless networks created new challenges for the service providers and network operators who, derive from the quality of service (QoS) the technical parameters of their network, a lot of efforts have been made to develop a new metrics which reflect the accuracy of the quality of the service provided. This measure is called the quality of experience. Therefore, the tools of quality of experience are usually used in parallel with quality of service tools. They allow improving the service degradation according to user's opinion, thus, launching analysis in order to achieve remedial actions. Also, they allow the engineers (TV services, voices...) or the technical managers or businesses (company applications) to lead their searches and then to reduce the time of problem resolutions.

In our work, we choose to apply our contribution to adaptive video streaming services HTTP (HAS) in MEC network architecture in order to improve the users' quality of experience QoE. The adaptive streaming HTTP (HAS) is a technique of video streaming widely used on the internet whether it was for video on demand (VoD) or for live streaming services. It uses the TCP protocol as a transport protocol. In this kind of service, the original video is divided in the server into several segments of the same duration. These segments are converted in several levels of quality. The main idea of the MEC architecture is to run applications and to perform processing tasks more closely to the user. Therefore, the network congestion is reduced and the applications work better. In this thesis, we proposed a solution to improve the quality of experience in the streaming video services based on the MEC architecture, which adds the congestion control to the process of improving video quality.

Keywords: Quality of Experience (QoE), Streaming Video, Mobile Edge Computing (MEC), Congestion, Probability of Congestion (PC), Server.

Table des matières

Remerciement.....	i
Dédicace.....	ii
ملخص.....	iii
Résumé.....	v
Abstract.....	vi
Table des matières.....	vii
Liste des figures.....	x
Liste des tableaux.....	xi
Chapitre01 : Notion de base	
Introduction générale :	1
1. Modèle OSI :.....	4
1.1. Définition :	4
1.2 Pourquoi l'utilisation de couches ?.....	5
1.3 Les couche du modèle OSI :.....	6
1.3.1 la couche physique :.....	6
1.3.2 La couche liaison de données :	6
1.3.3 La couche réseau :	6
1.3.4 La couche transport :.....	7
1.3.5 La couche session :	8
1.3.6 La couche présentation :	8
1.3.7 La couche application :	8
2. Le protocole TCP :.....	8
2.1 Définition.....	8
2.2 Le format des données sous TCP :	9
2.3 Caractéristiques de protocole TCP :.....	11
2.4 Fiabilité des transferts :.....	12
2.5 Etablissement d'une connexion :.....	14
2.5.1 L'ouverture d'une connexion TCP :	14
2.5.2 La fermeture d'une connexion TCP :	15
3. Le protocole UDP :	17
3.1 Définition :	17
3.2 Le format des données sous UDP :	17

4. Comparaison entre le protocole TCP et le Protocole UDP :.....	18
5. Le réseau sans file :	19
5.1 Définition :	19
5.2 Les catégories des réseaux sans fils :.....	19
5.2.1 Selon la zone de couverture :	20
5.2.2 Selon l'infrastructure :.....	24
5.2.3 Comparaison entre les réseaux ad hoc et les réseaux cellulaires :	25
6. Le vidéo streaming :	26
6.1 Introduction:	26
6.2 TCP et le vidéo streaming sure internet :.....	27
6.3 HTTP Live streaming.....	27
7. La congestion :	28
8. Le protocole CROSS-LAYER(Multicouches)	28
8.1 Définition :	28
8.2 La communication dans les architectures Cross-layer :.....	29
8.2.1 Communication direct entre les couches :.....	30
8.2.2 Communication via une base de données partagée :.....	31
8.3 Les approches du Cross-Layer dans le réseau sans fil :	31
9. L'architecture MEC (mobile edge computer).....	32
9.1. Introduction :	32
9.2 Normes MEC par le MEC ISG:.....	33
Chapitre02: Etat de L'art	
1. déférentes approches de vidéo streaming :.....	36
1.1 Dash (Dynamic Adaptative streaming over http).....	36
1.2. HLS (http Live streaming)	38
1.3 HDS (http dynamic streaming)	40
1.4 Etude comparative des approches.....	41
1.5 Etude critique :	41
2. L'architecture MEC (Mobile Edge Computing):	42
2.1 Introduction.....	42
2.2 Solution basé sur l'architecture MEC :	44
2.2.1 Introduction :	44
3. différences approches de la gestion de congestion dans les réseaux sans files :	48

3.1 TCP Reno :.....	48
3.2 Solution inter-couche pour le contrôle de Congestion de TCP dans les réseaux sans files :	49
3.2.1 Formulation de la solution:	50
3.3 Comparaison générale :	54
4. La congestion dans les enodeBs :.....	54
5. Etude critique :.....	55
6. Problématique :.....	55
Chapitre03:Conception	
1. Introduction	57
2. Architecture :	57
3. Solution et algorithme :	58
3.1 Description de la solution :.....	58
3.2 Algorithme de la solution :.....	60
Chapitre04: Implémentation et Résultats	
1. Introduction :	64
2. Environnement de test :	64
2.1 System d'exploitation :.....	64
2.2 Topologie d'émulation :	64
3. Scenario de teste :.....	68
3.1 Scenario 01 :.....	69
3.1.2 Discussion des résultats:.....	69
3.2 Scenario02 :	70
3.2.1 Discussion des résultats:.....	70
3.3 Scenario 03 :	70
3.3.1 Discussion des résultats:.....	71
3.4 Scenario 04 :	72
3.4.1 Discussion des résultats:.....	72
Conclusion générale & perspectives :.....	73
Référence	74

Liste des figures :

Figure 1: Le modèle OSI.....	5
Figure 2: Format d'un segment TCP.....	9
Figure 3: fiabilité de transfert.....	13
Figure 4: Perd des segments.....	13
Figure 5: Ouverture de connexion TCP [6].....	15
Figure 6: Fermeture de connexion TCP.....	16
Figure 7: Format de segment UDP.....	17
Figure 8: classification des réseaux sans fils	20
Figure 9: Réseau en mode ad hoc [15]	25
Figure 10: Les modèles de communication Cross-Layer	30
Figure 11: Les approches de Cross-Layer	32
Figure 12: Schéma du fonctionnement de DASH.....	37
Figure 13: Diagramme de flux pour les algorithmes d'adaptation pour DASH ..	38
Figure 14 : schéma du fonctionnement de HLS	39
Figure 15: Placement de MEC.....	43
Figure 16: Une architecture pour le streaming adaptatif mobile avec MEC Assistance.....	45
Figure 17: Séquence de message (mode de redirection).....	47
Figure 18: Séquence de message (mode proxy)	47
Figure 19: TCP Reno	49
Figure 20: structure d'un segment TCP.....	50
Figure 21: Organigramme de la solution	52
Figure 22: Placement de MEC.....	57
Figure 23: Schema de la solution proposée	58
Figure 24: Topologie d'émulation.....	64
Figure 25: Récupération de la bande passante	68
Figure 26: Calcule de PC	68
Figure 27: Stockage de fichier MPD.....	68
Figure 28: Application de modification dans la réponse	68
Figure 29: Résultat de scenario 01.....	69
Figure 30: Changement de qualité selon PC	70
Figure 31: Nombre des coupures par rapport à PC.....	71
Figure 31: calcule de MOS.....	72

Introduction générale :

Le vidéo streaming devient l'un des services les plus populaires pour les consommateurs mobiles. En effet, le trafic vidéo mobile, qui représentait 55% du trafic mobile total en 2015, représentera plus de 70% en 2020 [1]. La technologie de streaming n'est pas utilisée exclusivement pour les live streams, elle peut également être utilisée à la place de la méthode traditionnelle de téléchargement progressif, pour diffuser des fichiers audio et vidéo à la demande des utilisateurs.

Les réseaux sans fil se sont multipliés à un rythme effarant, ce qui donne à un nombre croissant des utilisateurs de l'ordinateur la possibilité de se connecter à internet n'importe où et à tout moment, les réseaux sans fil sont faciles à utiliser, et moins coûteux à mettre en œuvre que les réseaux câblés. A cause du nombre croissant des utilisateurs de réseau sans fil, une surcharge est marquée, c'est le problème de congestion. La congestion est l'un des problèmes majeurs posés ces derniers temps. Différentes méthodes et plusieurs solutions ont été suggérées pour faire diminuer les effets de ce problème, cependant, ça représente toujours un obstacle dans les réseaux sans fil. Dans notre mémoire, nous avons basé nos recherches sur la congestion.

Ces dernières années, une nouvelle architecture est apparue, c'est l'architecture MEC (Mobile Edge Computing) qui est une nouvelle option pour les fournisseurs de service et les opérateurs du réseau qui ont du mal à répondre aux demandes des utilisateurs en termes de consommation de débit et de la bande passante. L'idée de cette architecture est de réduire la congestion et d'améliorer les services clients en effectuant les tâches du traitement associé plus près de l'utilisateur final afin qu'il soit mieux satisfait. Plusieurs stratégies peuvent être utilisées, tel que les outils de la qualité d'expériences qui sont en général utilisés en parallèle des outils de qualité de services qui permettent aux utilisateurs de pré-orienter leurs recherches et donc de réduire le temps de résolution d'un problème (problème donné sur le réseau).

Dans ce mémoire intitulé : l'amélioration de la qualité d'expériences des services de vidéo streaming adaptatif basé sur l'architecture MEC, notre travail se résume en quatre chapitres :

Dans le premier chapitre, on a cité quelques notions de base. Tout d'abord, on a parlé du model OSI, on a expliqué les différentes couches avec leurs fonctionnements, puis, on a cité des protocoles du couche application (TCP et UDP) avec des caractéristiques, avantages et inconvénients de chaque protocole et la comparaison entre les deux. Ensuite, on a introduit le réseau sans fil et ses catégories, après, on a défini le vidéo streaming et la congestion. Finalement, on a parlé de l'architecture MEC et certaines de ses normes par le groupe de spécification de l'industrie(ISG).

Le deuxième chapitre (état de l'art) est divisé en deux parties, dans la première partie, on a cité les approches de vidéo streaming adaptatif les plus utilisées comme le Dash et le HLS, ensuite, on a cité les avantages et les inconvénients de chaque approche et on a terminé par une comparaison entre les approches et une étude critique. Dans la deuxième partie, on a défini l'architecture MEC (Mobile Edge Computing) en citant quelques exemples de son utilisation notamment le contrôle de la congestion. Ensuite, on a cité quelques approches de contrôle de congestion dans les réseaux sans fils avec une comparaison entre les approches, après, on a parlé de la congestion sur les enodeBs .Ce chapitre se termine par une problématique.

Dans le troisième chapitre, on a exposé notre solution d'amélioration de la qualité d'expériences des services de vidéo streaming adaptatif basé sure l'architecture MEC, on a mis l'architecture d'émulation de notre solution, ensuite, on a posé un schéma de la solution proposée avec l'explication, puis, on a créé un algorithme pour la solution. Enfin, on a discuté la solution proposée et l'algorithme.

Dans le quatrième chapitre, on a présenté l'environnement d'émulation de notre solution avec une présentation de système d'exploitation et la topologie d'émulation, ensuite, on a cité les outils et les logiciels utilisés dans l'émulation, après, on a cité les scénarios du test et on discuté les résultats de notre émulation.

Enfin, on a terminé par une conclusion générale et une perspective.

CHAPITRE 01 :

NOTION DE BASE

1. Modèle OSI :

1.1. Définition :

Le modèle OSI (Open Systems Interconnections model) a été créé en 1978 par l'organisation internationale de normalisation (ISO).

C'est un standard universel pour l'échange d'informations au sein d'un réseau, entre réseaux et des frontières géographiques [2]

Ce standard d'architecture réseau est un modèle à sept couches pour l'interconnexion des systèmes ouverts (OSI). Il permet d'homogénéiser la conception des réseaux et de maîtriser le traitement distribué. Les communications entre ces systèmes et réseaux distribués exigent une conception normalisée. Pour cela, les relations et les intersections entre les divers services et fonctions du réseau doivent être définies à l'aide d'interfaces et de protocoles communs.

Dans le modèle OSI, l'architecture réseau définit une hiérarchie de couches indépendantes, chacune contenant des modules chargés de réaliser des fonctions précises. Cela se traduit par un ensemble de règles définissant la façon dont les nœuds réseau doivent interagir pour communiquer et échanger des informations. Le modèle OSI définit des relations standard entre le matériel et les logiciels au sein des systèmes informatiques complexes. Chaque couche du modèle OSI fournit des services spécifiques qui contribuent au fonctionnement global du réseau.

Ces couches sont présentées sur la figure suivante [figure 1] [3]. On peut effectuer une subdivision entre les couches en les classifiant entre les couches basses (les couches 1, 2, 3 et 4) et les couches hautes (les couches 5, 6 et 7).

A chaque couche du modèle, la mise en forme des données doit être identique pour chaque niveau pour l'émetteur et le récepteur.

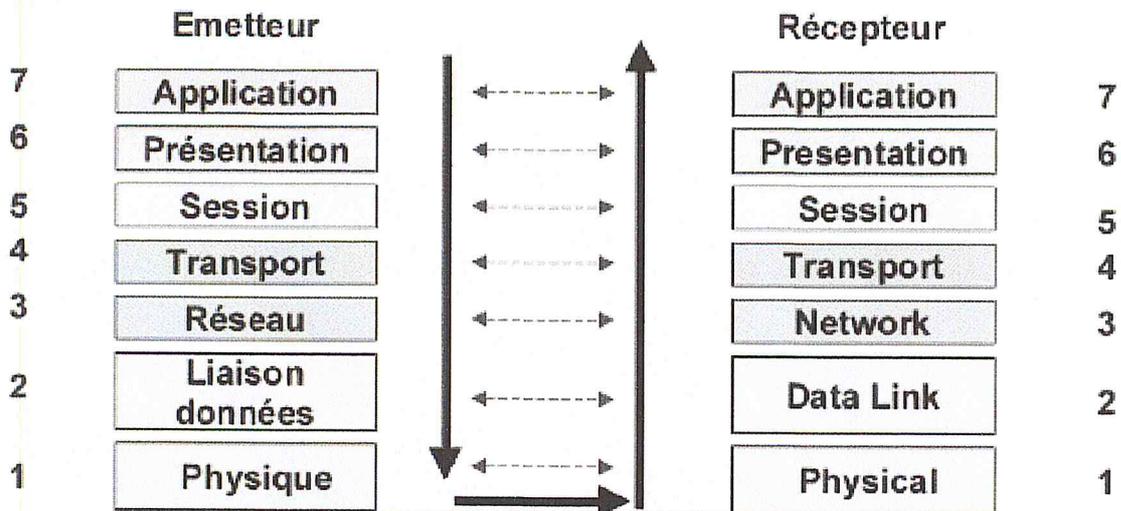


Figure 1: Le modèle OSI

1.2 Pourquoi l'utilisation de couches ?

Le plus simple pour comprendre l'intérêt des couches du modèle OSI, est de prendre la conversation entre deux personnes. Lors d'une conversation de ce genre, chaque personne a tout d'abord une idée, réfléchit ensuite à la formulation de cette idée, puis de la façon dont elle va l'énoncer (en parlant, criant, chantant, etc....). Il en résulte un son émis par la personne et qui transporte son idée. C'est un peu ce concept que l'on retrouve dans les couches du modèle OSI. [2]

Ces couches déterminent comment les informations doivent être transmises d'un ordinateur source à un ordinateur distant. En effet, pour qu'une communication puisse se dérouler correctement, il faut d'une part que le transport des informations se fasse correctement, mais pas uniquement. Il faut aussi, par exemple, que la communication se fasse dans la même langue pour l'émetteur et le destinataire. Dans le cas contraire, ils ne se comprendraient pas.

En langage réseau, ce sont les protocoles qui assurent que la communication se fasse bien dans le même langage, ou traduit le langage si ce n'est pas le cas.

1.3 Les couche du modèle OSI :

1.3.1 la couche physique :

Elle définit les caractéristiques électriques et mécaniques des interconnexions entre les machines, le but de cette couche c'est la transmission des bits à l'état brut sur un support physique (canal de communication).

Cette couche assurer qu'un bit à 1 envoyé sur une extrémité arrive aussi à 1 de l'autre côté, et non à 0, concerne le voltage pour représenter les états 0 et 1. [4]

1.3.2 La couche liaison de données :

Offre un transfert de trames de données sans erreurs à partir d'un nœud à un autre sur la couche physique, permettant ainsi aux couches au-dessus de transmettre sur la liaison sans aucune erreur.

Cette couche fournit un lien d'établissement et de libération des liens de connexions entre deux nœuds adjacents du réseau.

Fournit aussi l'accusé de réception de trames (reçoit des accusés de réception de trame). Détecte et récupère les erreurs qui se produisent dans la couche physique en retransmettant les trames non reconnues et gère les doublons de trames. [4]

1.3.3 La couche réseau :

La couche réseau (NETWORK LAYER) se charge de l'adressage des messages. Elle fournit un schéma d'adressage, et traduit les adresses logiques (les adresses IP) en adresses physiques (les adresses MAC des cartes réseaux). [3]

Les fonctions de la couche réseau :

- La traduction des adresses et des noms logiques en adresses physiques
- Le routage des messages en fonction de leur priorité et de l'état du réseau.
- La gestion du trafic sur le réseau.
- La commutation des paquets.
- Le contrôle de l'encombrement des messages sur le réseau.
- Le découpage ou le réassemblage des messages en fonction de la capacité de la carte réseau (et de celle de son correspondant)

1.3.4 La couche transport :

La couche transport s'assure que les paquets ont été reçus dans l'ordre, sans erreurs, sans pertes, ni duplication.

La couche de transport fournit :

- Segmentation des messages : accepte un message de la couche (session) au-dessus d'elle, divise le message en unités plus petites (si elles ne sont pas déjà assez petites) et transmet les plus petites unités à la couche réseau. La couche de transport réassemble le message au niveau de la station de destination.
- Message d'accusé de réception : assure la distribution fiable de messages de bout en bout avec les accusés de réception.
- Le contrôle du trafic de messages : indique à la station de transmission de « recul » lorsqu'aucun tampon de message n'est disponible.
- Multiplexage de session : multiplexe plusieurs flux de message ou de sessions sur un seul lien logique et garde en mémoire quels messages appartiennent à quelles sessions (voir la couche de session). [4]

En règle générale, la couche de transport peut accepter des messages relativement volumineux, mais il y a des limites de taille de message strictes, imposées par la couche réseau (ou inférieure). Par conséquent, la couche de transport doit décomposer les messages en unités plus petite. Les protocoles les plus connus de la couche transport sont TCP (Transmission Control Protocol), UDP (User Datagram Protocol).

Le TCP basé sur la réception d'accusé de réception par l'émetteur, contrairement à l'UDP qui ne permet pas à l'émetteur de vérifier si les données sont effectivement reçues en recevant un accusé de réception.

1.3.5 La couche session :

La couche session est maintenir un contexte de communication entre la source et la destination, sa fonction est de faire l'ouverture et la fermeture de la connexion (session), la reconnaissance des noms, et le contrôle de dialogue entre les processus communicants (qui transmet, à qui, à quelle moment, pour combien de temps, etc ...). [5]

1.3.6 La couche présentation :

La couche présentation met en forme les données à présenter à la couche application. On peut la voir comme un traducteur pour le réseau. Cette couche peut convertir les données d'un format utilisé par la couche application en un format courant au niveau de la station émettrice, puis convertir le format commun en un format connu pour la couche d'application au niveau de la réception.

La couche de présentation assure la traduction de codage de caractère, assure aussi la conversion et la compression des données (réduit le nombre de bits qui doivent être transmis sur le réseau). [4][5]

1.3.7 La couche application :

La couche application c'est la dernière couche du modèle OSI .Elle regroupe les services qui traitent des aspects sémantique de l'application dans un système réparti.

C'est l'interface entre les processus utilisateurs et le réseau (permet de l'accès au réseau). [5]

Il exit de nombreux protocoles de la couche application (FTP, SMTP, HTTP).

2. Le protocole TCP :

2.1 Définition

TCP (Transmission Control Protocol) est un des principaux protocoles de la couche transport du modèle OSI. Il permet, au niveau des applications, de gérer les données en provenance (ou à destination) de la couche inférieure du modèle OSI. [6]

TCP est un protocole orienté connexion, c'est-à-dire qu'il permet à deux machines qui communiquent de contrôler l'état de la transmission. Les caractéristiques principales du protocole TCP sont les suivantes[6] : TCP permet de remettre en ordre

les datagrammes en provenance du protocole IP, TCP permet de vérifier le flot de données afin d'éviter une saturation du réseau, TCP permet de formater les données en segments de longueur variable afin de les "remettre", TCP permet de multiplexer les données, c'est-à-dire de faire circuler simultanément des informations provenant de sources (applications par exemple) distinctes sur une même ligne TCP permet enfin l'initialisation et la fin d'une communication de manière courtoise.

2.2 Le format des données sous TCP :

La figure suivante donne la structure d'un segment TCP [6]

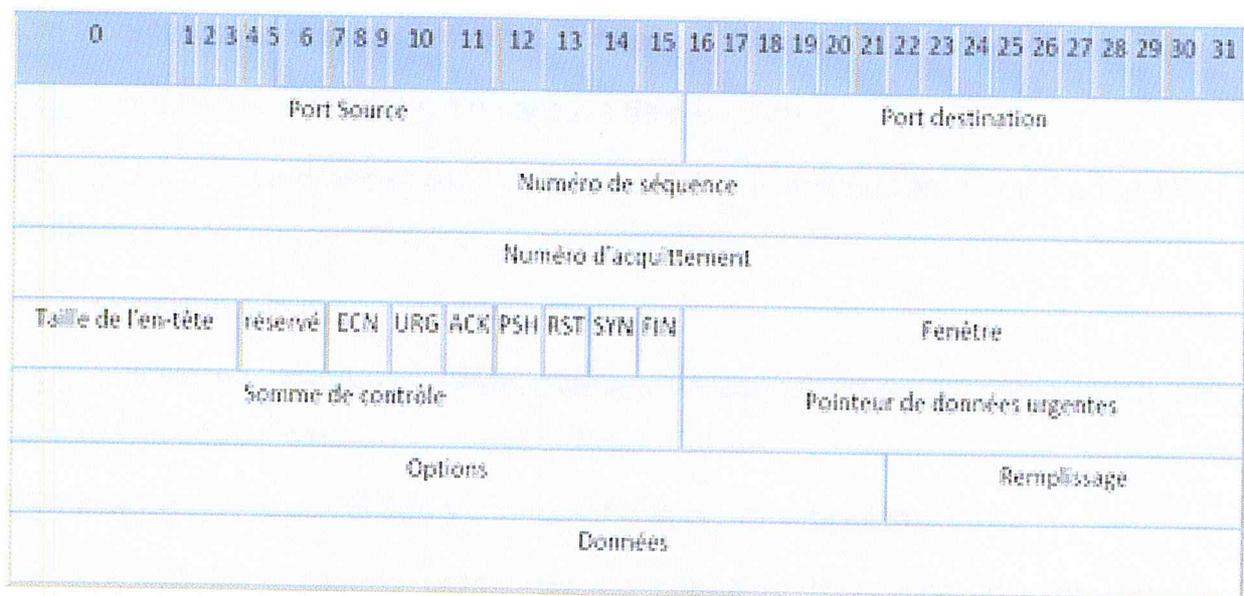


Figure 2: Format d'un segment TCP

La Signification des différents champs :

Port Source (16 bits): Port relatif à l'application en cours sur la machine source

Port Destination (16 bits): Port relatif à l'application en cours sur la machine de destination

Numéro de séquence (32 bits): Lorsque le drapeau SYN est à 0, le numéro de séquence est celui du premier mot du segment en cours. Lorsque SYN est à 1, le numéro de séquence est égal au numéro d'ordre initial utilisé pour synchroniser les numéros de séquence (ISN)

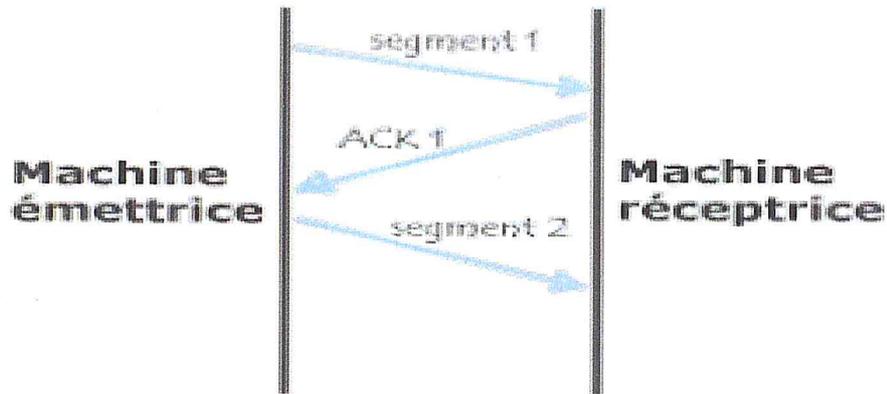


Figure 3: fiabilité de transfert

De plus, grâce à une minuterie déclenchée dès réception d'un segment au niveau de la machine émettrice, le segment est réexpédié dès que le temps imparti est écoulé, car dans ce cas la machine émettrice considère que le segment est perdu...

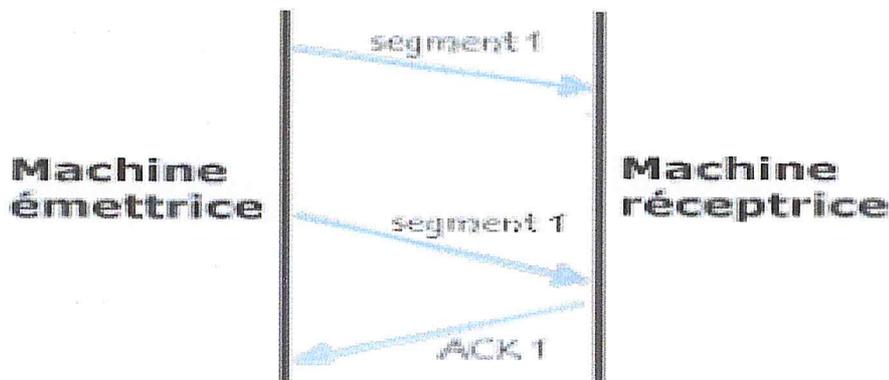


Figure 4: Perd des segments

Toutefois, si le segment n'est pas perdu et qu'il arrive tout de même à destination, la machine réceptrice saura grâce au numéro d'ordre qu'il s'agit d'un doublon et ne conservera que le dernier segment arrivé à destination...

2.5 Etablissement d'une connexion :

2.5.1 L'ouverture d'une connexion TCP :

Etant donné que ce processus de communication, qui se fait grâce à une émission de données et d'un accusé de réception, est basé sur un numéro d'ordre (appelé généralement numéro de séquence), il faut que les machines émettrices et réceptrices (client et serveur) connaissent le numéro d'ordre initial de l'autre machine. [6]

L'établissement de la connexion entre deux applications se fait souvent selon le schéma suivant :

- ✓ Les ports TCP doivent être ouverts
- ✓ L'application sur le serveur est passive, c'est-à-dire que l'application est à l'écoute, en attente d'une connexion
- ✓ L'application sur le client fait une requête de connexion sur le serveur dont l'application est en ouverture passive. L'application du client est dite "en ouverture active"

Les deux machines doivent donc synchroniser leurs séquences grâce à un mécanisme communément appelé three way handshake(poignée de main en trois temps), que l'on retrouve aussi lors de la clôture de session.

Ce dialogue permet d'initier la communication, il se déroule en trois temps, comme sa dénomination l'indique :

- Dans un premier temps la machine émettrice (le client) transmet un segment dont le drapeau SYN est à 1 (pour signaler qu'il s'agit d'un segment de synchronisation), avec un numéro d'ordre N, que l'on appelle numéro d'ordre initial du client
- Dans un second temps la machine réceptrice (le serveur) reçoit le segment initial provenant du client, puis lui envoie un accusé de réception, c'est-à-dire un segment dont le drapeau ACK est à 1 et le drapeau SYN est à 1 (car il s'agit là encore d'une synchronisation). Ce segment contient le numéro d'ordre de cette machine (du serveur) qui est le numéro d'ordre initial du

client. Le champ le plus important de ce segment est le champ accusé de réception qui contient le numéro d'ordre initial du client, incrémenté de 1

- Enfin, le client transmet au serveur un accusé de réception, c'est-à-dire un segment dont le drapeau ACK est à 1, dont le drapeau SYN est à zéro (il ne s'agit plus d'un segment de synchronisation). Son numéro d'ordre est incrémenté et le numéro d'accusé de réception représente le numéro d'ordre initial du serveur incrémenté de 1

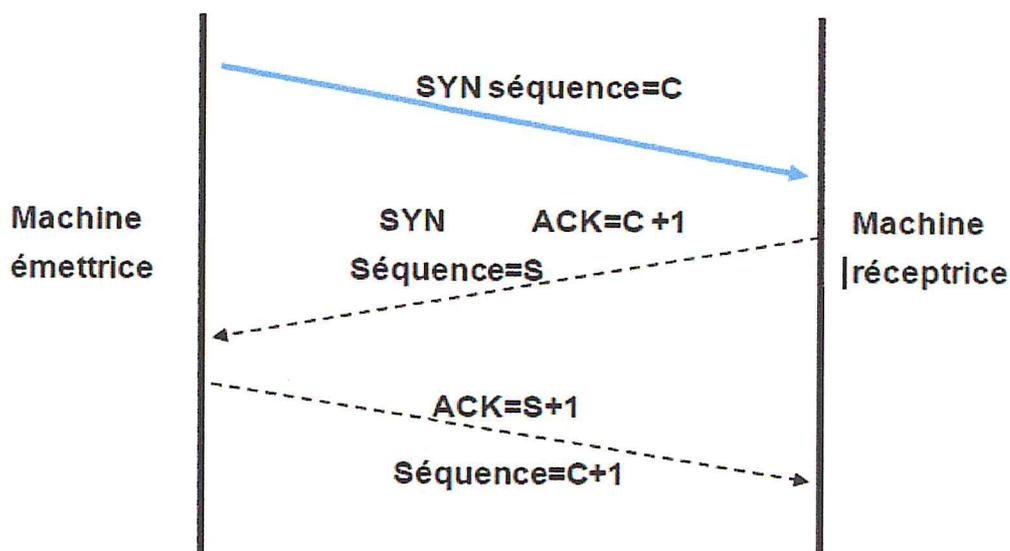


Figure 5: Ouverture de connexion TCP [6]

2.5.2 La fermeture d'une connexion TCP :

La figure suivante décrit le mécanisme de fermeture d'une connexion TCP [6]

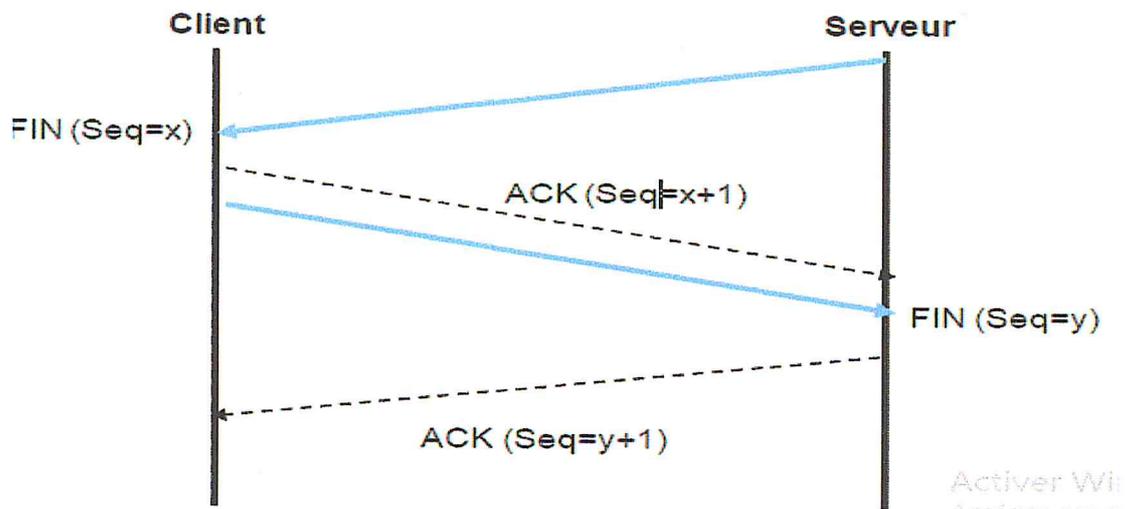


Figure 6: Fermeture de connexion TCP

La terminaison d'une connexion TCP nécessite quatre échanges de segment TCP.

Comme une connexion TCP est bidirectionnelle (full duplex) la rupture de celle-ci doit être faite dans les deux sens (Client/serveur) de la communication. Le client comme le serveur peuvent envoyer un segment avec l'indicateur FIN présent pour signifier leur fin d'envoi de données. La réception d'un segment avec le FIN indique que l'autre extrémité n'enverra plus de données.

Le terme utilisé alors est (**half close**), la connexion est à demi fermée.

Classiquement, c'est le client qui génère l'envoi du segment avec un indicateur FIN, il réalise une fermeture active (**active close**), le serveur à réception de ce segment réalise alors une fermeture passive (**passive close**).

Le serveur acquitte le FIN par un ACK, informe l'application pour la libérer de cette connexion et envoie ensuite un segment FIN au client qui à son tour, l'acquitte par un ACK.

3. Le protocole UDP :

3.1 Définition :

Le protocole UDP (User Datagram Protocol) est un protocole non orienté connexion de la couche transport du model OSI, Contrairement au protocole TCP : il n'existe pas de procédure de connexion préalable à l'envoi des données, et il n'y a pas de garantie de bonne livraison d'un datagramme à sa destination, l'ordre d'arrivés des datagrammes peut différer de l'ordre d'envoi .il est également possible que des datagrammes soient dupliqués. Les fonctions assurant la retransmission et le ré-ordonnancement doivent être assurées par les protocoles de la couche supérieure si elles sont souhaitées. [9]

Ce protocole est très simple étant donné qu'il ne fournit pas de contrôle d'erreurs .le rôle de protocole UDP est de permettre la transmission de données de manière très simple entre deux entités, chacun étant définie par une adresse IP et un numéro de port.

Le protocole UDP est utilisé dans le cas de communication multicast, le multicast permet l'envoi de paquet d'une source vers plusieurs destinataires.

3.2 Le format des données sous UDP :

La figure suivante décrit le format de message UDP : [9]

Port UDP source	Port UDP destination
Langueur msg UDP	Checksum
Données...	

Figure 7: Format de segment UDP

- **Port Source:** il s'agit du numéro de port correspondant à l'application émettrice du segment UDP. Ce champ représente une adresse de réponse pour le

destinataire. Ainsi, ce champ est optionnel, cela signifie que si l'on ne précise pas le port source, les 16 bits de ce champ seront mis à zéro, auquel cas le destinataire ne pourra pas répondre (cela n'est pas forcément nécessaire, notamment pour des messages unidirectionnels).

- **Port Destination:** Ce champ contient le port correspondant à l'application de la machine destinataire à laquelle on s'adresse.
- **Longueur:** Ce champ précise la longueur totale du segment, en-tête comprise, or l'en-tête a une longueur de 4 x 16 bits (soient 8 x 8 bits) donc le champ longueur est nécessairement supérieur ou égal à 8 octets.
- **Somme de contrôle:** Il s'agit d'une somme de contrôle réalisée de telle façon à pouvoir contrôler l'intégrité du segment.

4. Comparaison entre le protocole TCP et le Protocole UDP :

Dans le tableau suivant en va résumer quelques différences entre le protocole TCP et le protocole UDP [8][9]

Le protocole TCP	Le protocole UDP
<ul style="list-style-type: none"> - Pour envoyer des données sans perd des segments - Utilisée l'accusé de réception pour l'acheminement des segments - Fonctionne uniquement en point à point - Elle est très complexe que le protocole UDP 	<ul style="list-style-type: none"> -N'utilise pas l'accusé de réception -Gestion simple de communication -Communication multicast (le transfert es données est très rapide) -C'est pourquoi UDP est utilisée quand on a des flux (Streaming) audio ou vidéo à transfert -Très simple de mettre en œuvre

Tableau 1: Différence entre TCP et UDP

5. Le réseau sans fil :

5.1 Définition :

Un réseau sans fil (Wireless network) [10] est, comme son nom l'indique, un réseau dans lequel au moins deux terminaux peuvent communiquer sans liaison filaire. Grâce aux réseaux sans fils, un utilisateur a la possibilité de rester connecté tout en se déplaçant dans un périmètre géographique plus ou moins étendu, c'est la raison pour laquelle on entend parfois parler de "mobilité".

Les réseaux sans fils sont basés sur une liaison utilisant des ondes radioélectriques (radio et infrarouges) en lieu et place des câbles habituels. Il existe plusieurs technologies se distinguant d'une part par la fréquence d'émission utilisée ainsi que le débit et la portée des transmissions.

Les réseaux sans fils permettent de relier très facilement des équipements distants d'une dizaine de mètres à quelques kilomètres. De plus l'installation de tels réseaux ne demande pas de lourds aménagements des infrastructures existantes comme c'est le cas avec les réseaux filaires. En contrepartie se pose le problème de la réglementation relatives aux transmissions radio-électriques. De plus les ondes hertziennes sont difficiles à confiner dans une surface géographique restreinte, il est donc facile pour un pirate d'écouter le réseau si les informations circulent en clair. Il est donc nécessaire de mettre en place les dispositions nécessaires de telle manière à assurer une confidentialité des données circulant sur les réseaux sans fils.[11]

5.2 Les catégories des réseaux sans fils :

Un réseau sans fil (Wireless network) est un réseau où au moins deux nœuds peuvent avoir une communication sans liaison filaire. Ces réseaux de communications permettent aux utilisateurs de profiter de tous les services traditionnels des réseaux indépendamment de leurs positions géographiques.

Les réseaux sans fil peuvent avoir une classification selon deux critères. Le premier est la zone de couverture du réseau.

Au vu de ce critère il existe quatre catégories : les réseaux personnels, les réseaux locaux, le réseau métropolitain et les réseaux étendus. Le second critère est l'infrastructure ainsi que le modèle adopté. Par rapport à ce critère on peut diviser les

réseaux sans fils en : réseaux avec infrastructures et réseaux sans infrastructure, comme on le voit dans l'illustration de la figure suivante [12] :

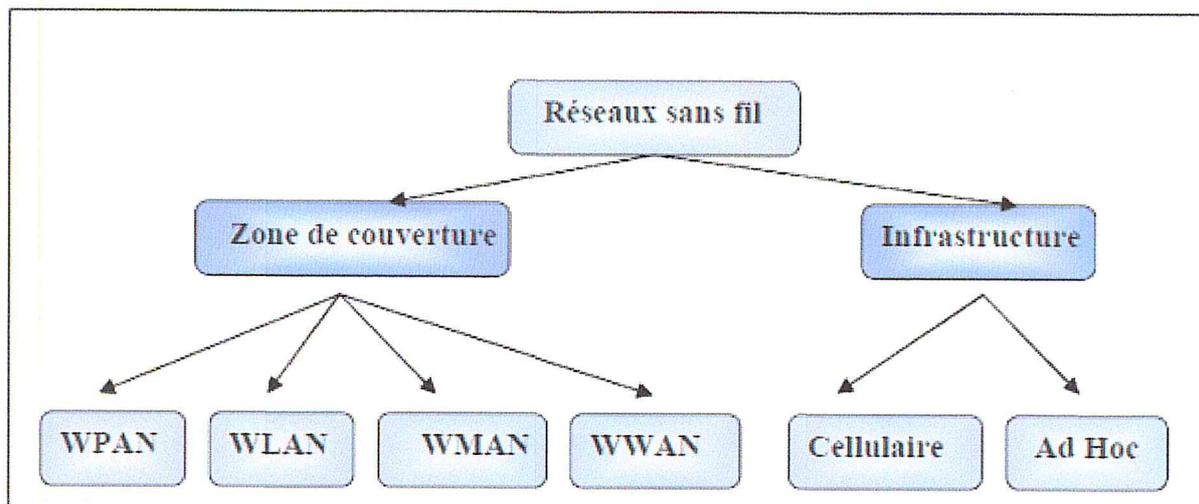


Figure 8: Classification des réseaux sans fils

5.2.1 Selon la zone de couverture :

a. Réseaux personnels sans fil (WPAN) :

Les réseaux personnels sans fil ou Wireless Personal Area Network (WPAN), sont des réseaux sans fil à très faible portée, de l'ordre d'une dizaine de mètres. Ils sont le plus souvent utilisés à faire communiquer entre eux des matériels présents sur une personne (par exemple une oreillette et un téléphone portable). Ils sont également utilisés pour relier des équipements informatiques entre eux sans liaison filaire : par exemple pour relier une imprimante ou un PDA (Personal Digital Assistant) à un ordinateur de bureau ou faire communiquer deux machines très peu distantes [11].

Il existe plusieurs technologies permettant la mise en œuvre de tels réseaux qui sont :

- **Bluetooth :**

La norme Bluetooth (pris en charge par IEEE 802.15.1) est une technologie de moyen débit, elle permet d'atteindre un débit maximal théorique de 1Mbps (environ 720Kbps effectif) à basse consommation énergétique.

Bluetooth utilise la bande de fréquence 2.4GHz avec une couverture entre 10 et 30 mètres. Cette technologie permet de créer un réseau de 8 appareils en communication simultanée. La petite taille des composants Bluetooth lui permet d'être inséré dans des équipements tels que les claviers et les souris sans fil, les kits main libre ou écouteur et le transfert de données entre un pc et les PDA (Personal digital assistant) ou téléphones mobiles...etc. [11]

- **Zig Bee :**

Le standard IEEE 802.15.4 propose une norme pour les couches physique et liaison de données, orientée très faible consommation énergétique, qui rend cette technologie bien adaptée à de petits appareils électroniques (appareils électroménagers, hifi, jouets,.. .), et plus particulièrement aux réseaux de capteurs. La pile proposée par l'IEEE et la ZigBee qui a pour objectif de promouvoir une puce offrant un débit relativement faible (100Kbps environ) mais à un coût très bas, et une consommation électrique extrêmement réduite. [12]

- **HomeRF :**

Comme son nom l'indique, HomeRF est une norme de réseau destinée à un usage domestique pour partager un accès à Internet ou transporter des communications téléphoniques DECT3. Il a été lancé par le consortium industriel formé entre autres par HP, IBM, Intel et Microsoft. HomeRF proposait une couche physique travaillant dans la bande des 2,4 GHz, en FHSS (à 50 sauts par secondes), sur une modulation de type 2- FSK ou 4 -FSK. Le débit bande de base est de 1 Mbits/s ou 2 Mbits/s, suivant la modulation utilisée, avec une portée typique de l'ordre d'une cinquantaine de mètres. HomeRF n'a pas connue de succès et a été abandonnée.

- **Liaisons infrarouges :**

La technologie infrarouge ou IrDA est également utilisée dans ce type de réseaux. Cette technologie est cependant beaucoup plus sensible que Bluetooth aux perturbations lumineuses et nécessite une vision directe entre les éléments souhaitant communiquer, ce qui la limite bien souvent à un usage de type télécommande. [12]

d. Les réseaux sans fil étendus (WWAN) :

Les réseaux sans fil (WWAN pour Wireless Wide Area Network) Cette catégorie possède assez peu de technologies à l'heure actuelle. Les seules technologies de WWAN disponibles sont des technologies utilisant les satellites géostationnaires ou en orbite basse pour relayer l'information entre plusieurs points du globe. Les principales technologies sont les suivantes [13] :

- **GSM :**

GSM est l'abréviation de (Global System for Mobile Communication), c'est une norme établie en commun par les opérateurs européens depuis 1982, ayant pour objectif le développement d'un système de téléphonie mobile permettant des communications outre-mer. La communication a lieu par un paquet d'onde ayant deux fréquences : 900 MHz et 1800 MHz. Le GSM se distingue par plusieurs spécificités, le premier est l'aspect numérique du réseau, qui offre une qualité supérieure grâce à sa résistance aux interférences. La deuxième spécificité du réseau de GSM réside dans sa configuration cellulaire. Le territoire est subdivisé en petites cellules attachées les unes aux autres. Chaque cellule se voit assigner un certain nombre de canaux permettant les communications.

- **GPRS :**

Le GPRS (General Packet Radio Services) est une technologie de radiocommunication par commutation de paquets pour les réseaux de GSM. Les connexions des services de GPRS sont toujours ouvertes afin d'offrir aux utilisateurs des terminaux mobiles une disponibilité de réseau identique à celle qu'ils pourraient atteindre par des réseaux d'entreprise. Le GPRS offre une connectivité d'IP de bout en bout, du terminal GPRS jusqu'à n'importe quel réseau IP. Les terminaux peuvent être intégrés efficacement aux réseaux Internet. La vitesse "utile" sera d'environ 40 Kb/s (vitesse maximum : 171 Kb/s), l'un ou l'autre est quatre fois supérieure à celle du GSM.[13]

- **UMTS :**

UMTS L'abréviation de « Universal Mobile Telecommunications System», l'UMTS désigne une nouvelle norme de téléphonie mobile. Le principe de l'UMTS consiste à exploiter une bande de fréquences plus grande pour faire transmettre plus des données et donc obtenir un débit plus important. En théorie, il peut atteindre 2 Mb/s. La norme d'UMTS exploite de nouvelles bandes de fréquences situées entre 1900 et 2200 MHz. Cette technologie permet de faire passer des données simultanément et offre des débits nettement supérieurs à ceux atteints par le GSM et le GPRS. [13]

5.2.2 Selon l'infrastructure :

Les environnements mobiles sont des systèmes composés de sites mobiles et qui permettent à leurs utilisateurs d'accéder à l'information indépendamment de leurs positions géographiques. Les réseaux mobiles ou sans fil, peuvent être classés en deux classes : les réseaux avec infrastructure et les réseaux sans infrastructure. [14]

a. Réseaux cellulaires (avec infrastructure) :

Ce type de réseaux se compose des éléments suivants :

- ✓ Les "sites fixes" du réseau filaire.
- ✓ Les "sites mobiles", réseaux sans fils.

Certains sites fixes, appelés stations de base (SB) sont munis d'une interface de communication sans fil pour la communication directe avec les sites mobiles localisés dans une zone géographique limitée, appelée cellule. A chaque station de base correspond une cellule à partir de laquelle des unités mobiles peuvent émettre et recevoir des messages. Alors que les sites fixes sont interconnectés entre eux à travers un réseau de communication filaire. Une unité mobile ne peut être, à un instant donné, directement connectée qu'à une seule station de base. Elle peut communiquer avec les autres sites à travers la station à laquelle elle est directement rattachée.[14]

b. Réseaux ad hoc (sans infrastructure) :

L'évolution récente de la technologie dans le domaine de la communication sans fil et l'apparition des unités de calculs portables (les laptops par exemple), poussent aujourd'hui les chercheurs à faire des efforts afin de réaliser le but des réseaux :

Le concept des réseaux mobiles ad hoc essaie d'étendre les notions de la mobilité à tous les composants de l'environnement.

Ici, contrairement aux réseaux sans fil basés sur les points d'accès (avec infrastructure), aucune administration centralisée n'est disponible, ce sont les hôtes mobiles elles-mêmes qui forment, d'une manière ad hoc, une infrastructure du réseau. Aucune supposition ou limitation n'est faite sur la taille du réseau ad hoc, le réseau peut contenir des centaines ou des milliers d'unités mobiles. [15]

Les réseaux ad hoc sont idéals pour les applications caractérisées par une absence (ou la non fiabilité) d'une infrastructure préexistante, tel que les applications militaires et les autres applications de tactique comme les opérations de secours (incendies, tremblement de terre..) et les missions d'exploration

b.1. Définition :

Un réseau ad hoc, connu aussi sous le nom de MANET (Mobile Ad hoc Network), est un réseau dont la topologie ne bénéficie d'aucune infrastructure fixe préexistante ou administration topologie ne bénéficie d'aucune infrastructure fixe préexistante ou administration centralisée

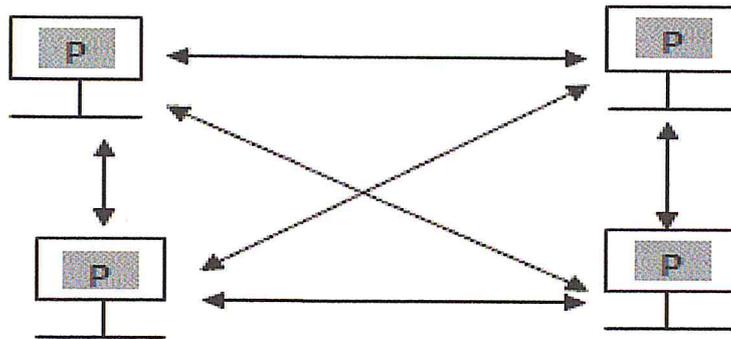


Figure 9: Réseau en mode ad hoc [15]

Ici, contrairement aux réseaux basés sur la communication cellulaire, ce sont les unités mobiles elles même qui forment, d'une manière ad hoc, une infrastructure du réseau et en maintiennent la connectivité d'une façon décentralisée. L'information est transmise par l'intermédiaire des mobiles présents.

Un réseau ad hoc est un réseau sans fil auto-configurable. Lorsque deux machines mobiles ou plus se retrouvent dans le même secteur géographique, elles doivent se reconnaître pour pouvoir s'échanger des données. Le réseau doit se configurer automatiquement ; périodiquement ou à la demande pour assurer la liaison entre ces machines

5.2.3 Comparaison entre les réseaux ad hoc et les réseaux cellulaires :

Le tableau suivant donne les principales différences entre les réseaux ad hoc et les réseaux cellulaires [15][16]

Réseau ad hoc	Réseau cellulaire
Des réseaux sans infrastructure	Avec infrastructure
Moins couteuse	Les couts d'installation sont élevés
Rapidité d'installation	Beaucoup de temps d'installation
Aucune station de base	Les stations de bases sont pré-localisées

Tableau 2: Comparaison réseau ad hoc/ cellulaire

6. Le vidéo streaming :

6.1 Introduction:

Les années précédentes, l'adoption générale de la connectivité de haut débit, pour les réseaux filaires et sans fil (mobile), permet le fonctionnement effectif des applications intensives, tel que les applications multimédias, les vidéos streaming, IPTV, et les vidéos conférences. Ces applications peuvent utiliser une très haute résolution, qui peut atteindre 4096*2160(4k) avec 60fps de la fréquence d'mafé. Cependant, un tel riche contenu du média a besoin des ressources des traitements suffisants pour décorer un comprimé flot binaire à l'ordre de 10Mbps.

Dans les services de vidéo streaming, la vidéo était délivrée comme tous les données qui utilisent les connexions TCP avides. Le flux vidéo étai tamponnée chez le destinataire pendant un moment avant le commencement de la lecture de vidéo donc cette courte durée décale entre la vidéo à un débit binaire et la bande passante de réseau disponible peut être absorbé, et les interruptions de la vidéo peuvent être atténuées. Néanmoins, si le décalage persiste le tampon peut finalement être vide et les interruptions de la lecture (playback) peuvent se réduire en affectant l'expérience d'utilisateur [17].

Pour résoudre ce problème, un nouveau type de vidéo streaming service apparait, c'est l'adaptable vidéo streaming services dans laquelle la source du vidéo est disponible dans les différents qualités et l'utilisateur peut de manière dynamique choisir la meilleure qualité en correspondant sa bande passante disponible.

Les services de vidéo streaming adaptatifs devenu un produit dans les appareils mobiles. Avec la popularité récente des smart phones, tablettes, notebook (mini portable) et les pc portables (lap top). Le portable (mobile) utilisé pour service streaming est considérablement en expansion.

Les services de vidéo streaming représentant une part de plus en plus importante de trafic web mondial, les consommateurs de vidéo représentent 69% de nombre totale de trafic internet en 2017.

6.2 TCP et le vidéo streaming sur internet :

Il existe plusieurs protocoles de la transition destinée à la vidéo streaming, tel que le TCP et le RTMP.

Bien que TCP a été considéré inapproprié pour le transport des protocoles des services de vidéo streaming. Elle a récemment vécu une acceptation plus large et elle est utilisée avec le http. Cet intérêt est motivé par les facteurs suivants :

- 1) Les applications ont commencé à être de plus utilisées a travers le navigateur web.
- 2) TCP est facile de se développer dans les CDNs. Ainsi, TCP qui délivre déjà la plupart de trafic internet est capable de garantir la stabilité du réseau pendant l'accession de tous les appareils connectés.

Les auteurs ont développé des modèles d'une performance analytique pour évaluer la performance de TCP quand il est utilisé pour transmettre une source d'une vidéo live Streaming sans l'utilisation de l'adaptation de la qualité. Les résultats obtenus, qui considèrent une vitesse d'un segment constant une source de la vidéo, suggéré que pour acquérir une bonne performance en délai de démarrage et le pourcentage des arrivants des paquets de l'envoi tardif, TCP a besoin d'une bande passante du réseau qui est environ deux fois la vitesse d'un segment de vidé [17]

6.3 HTTP Live streaming

HTTP live streaming est un protocole créé par Apple pour permettre le streaming à destination des terminaux, HLS permet de distribuer des contenus en direct et des fichiers à la demande et est l'unique technologie capable de streamer à destination des terminaux Apple. HLS a très vite été pris en main par les développeurs d'applications.

Enfin, le choix de compatibilité de beaucoup de serveurs de streaming vis-à-vis de ce protocole tel qu'Adobe, Microsoft, Akamai ainsi que des clients terminaux tel que Google depuis Android 3.0 ont permis de positionner ce protocole comme le leader du streaming adaptatif via HTTP. Le fonctionnement de HLS est approximativement le même que HDS.

Les différences principales résident dans le fait que HLS ne nécessite pas d'extension au serveur Web. Ainsi, toute la logique d'adaptation est gérée au niveau du client. Pour distribuer un multimédia à des clients utilisant HLS, il suffit d'encoder la source en plusieurs fichiers de débit binaire différent et de les diviser en partie, généralement de 10 secondes. Ces fichiers sont hébergés sur un serveur HTTP et un fichier supplémentaire de description leur est ajouté. Ce fichier de description est utilisé par le client pour choisir la partie de la vidéo à télécharger en fonction de la bande passante disponible à l'instant T. [19]

7. La congestion :

La congestion est la condition dans laquelle le paquet bloqué quelque part dans le réseau ce qui retarde le paquet à arriver à la destination.

C'est le moment où l'augmentation de trafic provoque un ralentissement. Les trames entrantes dans les buffers des commutateurs sont rejetées dans ce cas puisque le buffer est saturé, qu'il ne peut plus stocker de nouveaux paquets et qu'il y a par conséquent une perte des paquets.[20]

8. Le protocole CROSS-LAYER(Multicouches)

8.1 Définition :

En général, un protocole est dit protocole cross-layer si les dépendances entre ce protocole et des protocoles d'autres couches sont exploitées afin de surmonter les limites de chaque protocole et atteindre un gain en performance. [21]

Le design Cross-layer est défini comme suit : « la conception d'un protocole en violation avec l'architecture en couches de référence est une conception Cross-layer à l'égard de cette architecture ». Le terme violation englobe [22] :

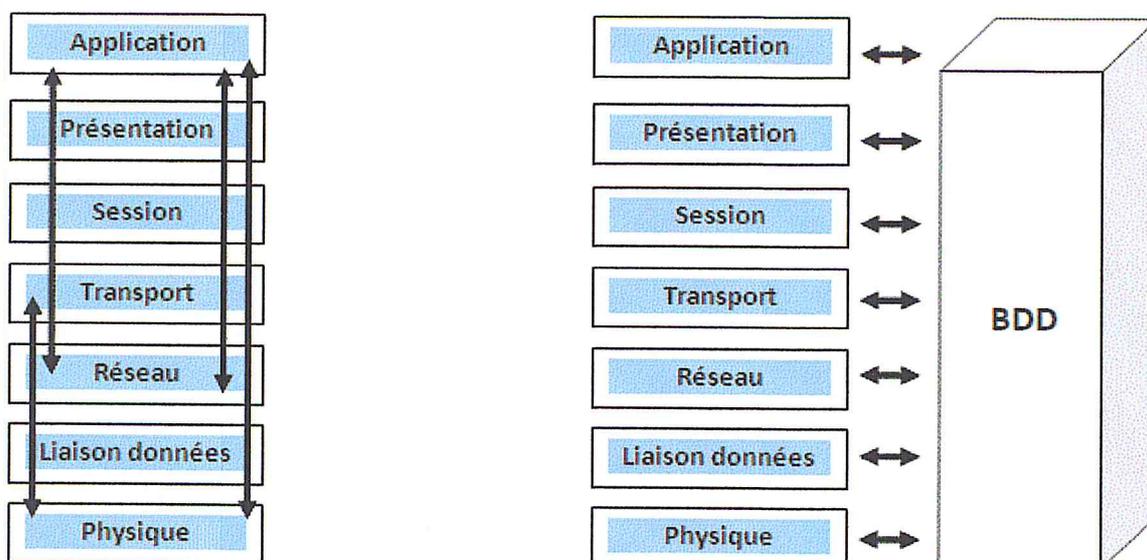
- La définition de nouvelles interfaces entre les couches.

- La redéfinition des limites des couches. - La conception d'un protocole sur une couche en se basant sur la conception d'un autre protocole sur une autre couche.
- La configuration commune des paramètres à travers les couches.

Le concept Cross-layer permet la définition de protocoles ou de mécanismes qui ne respectent pas l'isolation des couches du modèle OSI (respectivement TCP/IP). Ainsi, il autorise la communication entre deux, ou plusieurs, couches adjacentes, ou non adjacentes, dans le but d'améliorer les performances globales du système. Ceci peut être réalisé par la définition de nouvelles interfaces au niveau des couches qui permettent de récupérer leurs paramètres de performance mais aussi de configurer certains paramètres d'une manière dynamique. Ces paramètres peuvent être utilisés par les protocoles et/ou les mécanismes d'adaptation pour améliorer la performance globale de la communication.[22]

8.2 La communication dans les architectures Cross-layer :

Le principe de base du concept Cross-layer est de permettre l'échange d'informations entre les couches adjacentes et non adjacentes afin d'améliorer les performances de transmission. Cet échange d'informations peut être mis en œuvre suivant différents schémas. Parmi toutes les architectures Cross-layer proposées dans la littérature, deux modèles de communication peuvent être distingués [23][24]: La communication directe entre les couches et une base de données partagée entre les couches. Nous présentons ci-dessous une description de ces deux modèles représentés dans la Figure suivante [25]



(a) Communication direct

(b) Communication via une BDD partagée

Figure 10: Les modèles de communication Cross-Layer

8.2.1 Communication direct entre les couches :

La communication directe entre les couches est le modèle le plus utilisé par les architectures Cross-layer. Il permet à une couche d'accéder directement aux paramètres et aux variables d'une autre couche sans passer par un intermédiaire. Cette communication peut être « in-band » en utilisant les en-têtes des protocoles qui sont déployés actuellement, par exemple, la couche IP accède aux champs de l'en-tête TCP « ECN » pour indiquer une congestion dans le réseau. Dans certain cas, des extensions d'en-tête sont nécessaires pour faire passer des informations supplémentaires. La communication directe peut être « out-of-band » en utilisant un nouveau protocole dédié. Le protocole de signalisation CLASS (Cross-Layer Signaling Shortcuts), présenté dans [23], est un parfait exemple de ce type de protocole. La communication « out-of-band » peut s'effectuer aussi en définissant de nouvelles interfaces (API : Application Programming Interface) au niveau d'une couche qui seront utilisées directement par d'autres couches pour récupérer et configurer des paramètres de fonctionnement. Un exemple de cette API est proposé dans [25].

8.2.2 Communication via une base de données partagée :

Plusieurs architectures Cross-layer [25][26] proposent l'utilisation d'une base de données partagée afin de stocker et de récupérer des paramètres. Cette base est accessible par toutes les couches qui peuvent, ainsi, s'informer de l'état des autres couches ou récupérer des paramètres de configuration nécessaire à leur fonctionnement interne. La base de données est aussi accessible par un système d'optimisation responsable d'initialiser les paramètres avec les valeurs adéquates. La Figure 10 (b) illustre ce type de communication, la base de données est considérée comme une nouvelle couche en parallèle à toutes les autres. Cependant, pour mettre en œuvre cette base de données, il faut répondre à plusieurs contraintes conceptuelles, comme la localisation de la base de données (sur les nœuds communicants ou sur un nœud indépendant), le type de communication entre les couches et la base de données partagée, ainsi que le protocole de communication utilisé.[26]

8.3 Les approches du Cross-Layer dans le réseau sans fil :

Dans la littérature, plusieurs techniques Cross-layer ont été proposées pour améliorer les performances des transmissions sans fil. Au début, ces mécanismes étaient limités à l'interaction entre la couche physique et la couche liaison de données. De plus, les mécanismes proposés étaient indépendants et visaient l'amélioration d'une imperfection précise.

La figure suivante présente quelques approches de Cross-Layer [23]

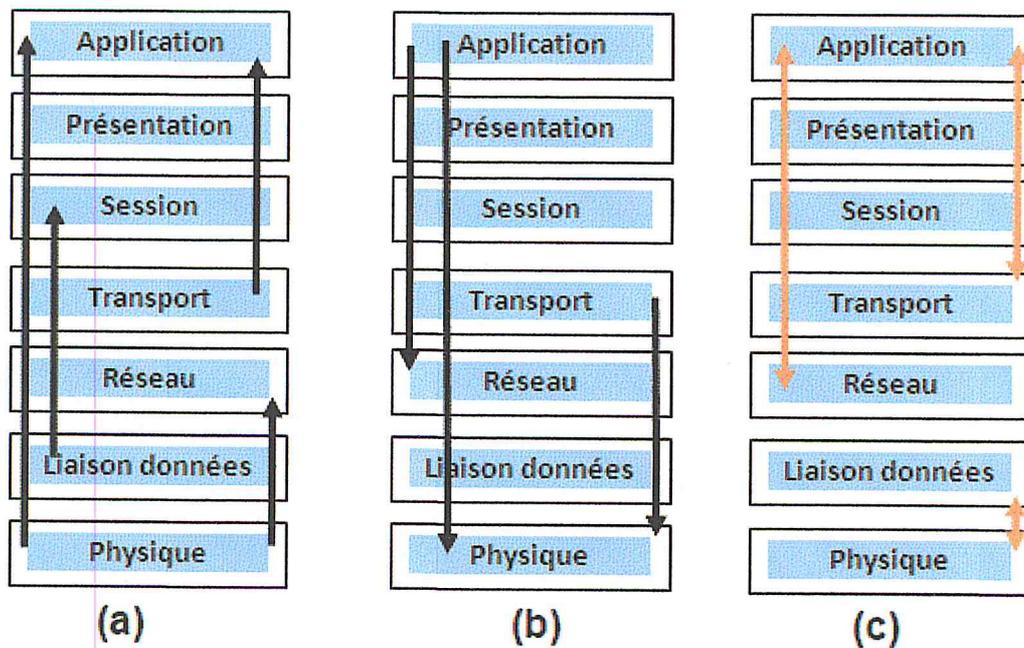


Figure 11: Les approches de Cross-Layer

- **(a) L'approche ascendante (Bottom-up):** Les couches supérieures optimisent leurs mécanismes en fonctions des paramètres (conditions) des couches inférieures.
- **(b) L'approche descendante (Top-down):** Les couches supérieures décident des paramètres de configuration des couches inférieures. Ou bien, les couches inférieures considèrent certaines spécificités du niveau applicatif pour exécuter leurs traitements.
- **(c) L'approche mixte (Integrated):** Cette approche exploite les deux approches précédentes dans une même architecture afin de trouver la meilleure configuration inter-couches pour un fonctionnement optimal du système.

9. L'architecture MEC (mobile edge computer)

9.1. Introduction :

L'Institut européen des normes de télécommunication (ETSI) produit actuellement des normes pour le calcul de bord multi-accès (MEC), anciennement connu sous le nom de Mobile Edge Computing, en collaboration avec OpenFog Consortium[26] . Ils ont formé un groupe de spécification de l'industrie (ISG) qui est chargé d'identifier les normes MEC qui optimisent les avantages pour tous les acteurs impliqués. L'objectif principal met l'accent sur la coopération entre les opérateurs de réseau, les

applications et les fournisseurs de contenu pour améliorer l'expérience utilisateur globale.

MEC contourne l'encombrement du réseau en connectant les périphériques au réseau. La périphérie du réseau est positionnée localement dans les stations de base, les centres de données et d'autres points chauds locaux dans le but d'optimiser le cloud computing. Une foule de périphériques s'appuient sur une connexion réseau stable pour fonctionner, comme l'Internet des objets (IoT), les voitures autonomes et les applications basées sur le cloud. Non seulement le fait d'exploiter un réseau de périphérie local rapproche l'informatique de l'utilisateur final, mais MEC réduit la latence et a une bande passante élevée. Le résultat est une connectivité réseau cohérente et la possibilité pour plusieurs périphériques de se connecter simultanément. [28]

9.2 Normes MEC par le MEC ISG:

- **Aspects de la mobilité de bout en bout** - Cette norme aborde le problème de la continuité de service lorsque l'appareil d'un utilisateur utilise un réseau de périphérie, puis quitte le réseau. Il décrit, en détail, les procédures de relocalisation, les solutions et les considérations pour divers scénarios afin de maintenir une communication constante entre l'appareil et le réseau.
- **Mobile Edge Management, Partie 1 : Gestion du système de l'hôte et de la plate-forme** : Cette norme définit le protocole de gestion du système de périphérie mobile des hôtes et des plates-formes
- **API de gestion de bande passante** - Cette norme traite principalement des problèmes de bande passante lorsque plusieurs périphériques utilisent le même réseau périphérique. Il se concentre sur les informations de stratégie d'application et sur la façon de traiter certains scénarios d'interface de programme d'application (API) qui affectent l'utilisation de la bande passante et le périmètre du réseau.
- **Interface d'application UE** - Cette norme explique comment gérer le cycle de vie de l'application sur l'interface de l'application connectée.
- **Principes généraux pour les API de service Edge Mobile** - Contient un glossaire des principes de conception du service mobile Edge RESTful. Ce document met également en évidence les lignes directrices et les modèles de l'API.

- **Mobile Edge Management, Partie 2** - Le MEC ISG décrit le protocole de gestion du cycle de vie de l'application dans cette norme. Le document énumère les règles et les exigences de gestion. Il illustre également les points de référence pour prendre en charge la gestion du cycle de vie.
- **Activation des applications de plate-forme mobile** - Ce document se concentre sur la manière dont le point de référence de la fonctionnalité de plate-forme mobile (Mp1) permet aux applications de communiquer avec le système de périphérie mobile.
- **API d'informations de réseau radio** - Les informations de réseau radio peuvent être utilisées par des plates-formes mobiles pour améliorer les processus de mobilité. Cette norme décrit quand il est utile d'adopter l'API Radio Network Information, et comment l'utiliser.
- **API de localisation**: cette norme établit des directives pour la détection des informations de localisation des appareils de l'utilisateur sur le réseau périphérique.
- **Accélération du marché Meilleures pratiques et consignes de MEC Metrics** - Ce document répertorie les métriques MEC afin d'évaluer les améliorations technologiques de MEC.
- **Exigences techniques** - Cette spécification couvre les principes génériques liés à MEC, tels que l'alignement NFV et le support de mobilité. En outre, il explique le cadre, les exigences de service et répertorie les fonctionnalités importantes.
- **Architecture du cadre et de référence** - Cette spécification illustre le cadre et l'architecture de référence de MEC. Ceci est particulièrement utile car il indique les points de référence mentionnés dans les documents standards ultérieurs.
- **Terminologie** - Glossaire des termes concernant les différents éléments du MEC.
- **Scénarios de service** - Un aperçu des services qui bénéficient de MEC.
- **Cadre de validation de concept (PoC)** - Il décrit le cadre qui sous-tend le PoC et ses objectifs. [29]

CHAPITRE 02 :

ETAT DE L'ART

Dans le cadre de notre thème, le travail demandé vise à améliorer la qualité d'expériences dans les services de vidéo streaming, en se basant sur une architecture MEC.

Ce chapitre, commence par décrire quelques approches de vidéo streaming adaptatif les plus utilisées, en suite il situe les avantages et les inconvénients de chaque approche. Dans une deuxième partie, il décrit et définit l'architecture MEC tout en citant quelques exemples de son utilisation notamment le contrôle de congestion dans les réseaux sans fil. Le chapitre se termine par une problématique.

1. différentes approches de vidéo streaming :

En raison de la forte demande des services de vidéo streaming sur internet, plusieurs approches ont été proposées. Dans cette partie on va citer quelques-unes de ces approches les plus utilisées de nos jours tout en présentant les avantages et les inconvénients de chacune des approches, puis on termine par une comparaison.

1.1 Dash (Dynamic Adaptive streaming over http) [30]

DASH est le dernier protocole de streaming HTTP adaptatif à avoir vu le jour. L'idée principale est la création d'un standard pour les systèmes de streaming adaptatif ayant le potentiel de remplacer toutes les technologies propriétaires existantes comme Microsoft Smooth Streaming, Adobe Dynamic Streaming et Apple HTTP Live Streaming. Un standard unifié serait très intéressant pour les créateurs de contenu, qui pourrait produire un seul service capable de jouer surtout les terminaux compatibles. Bien que Microsoft ait indiqué qu'il intégrerait le standard dès que celui-ci serait terminé, Adobe et Apple ne voient pas les choses de la même manière. La non-implémentation de ce standard dans les terminaux iOS est un frein considérable à son essor futur. Au sein du protocole DASH, nous retrouvons les deux composants principaux du système de streaming adaptatif basé sur HTTP, les fichiers audio/vidéo encodés de différentes manières et le fichier de description qui décrit au lecteur les URL de chaque source. Les fichiers A/V sont appelés « Media Presentation » tandis que le fichier de description est appelé « Media Presentation Description (MPD) ». La figure 12 [31] présente l'architecture de ce protocole.

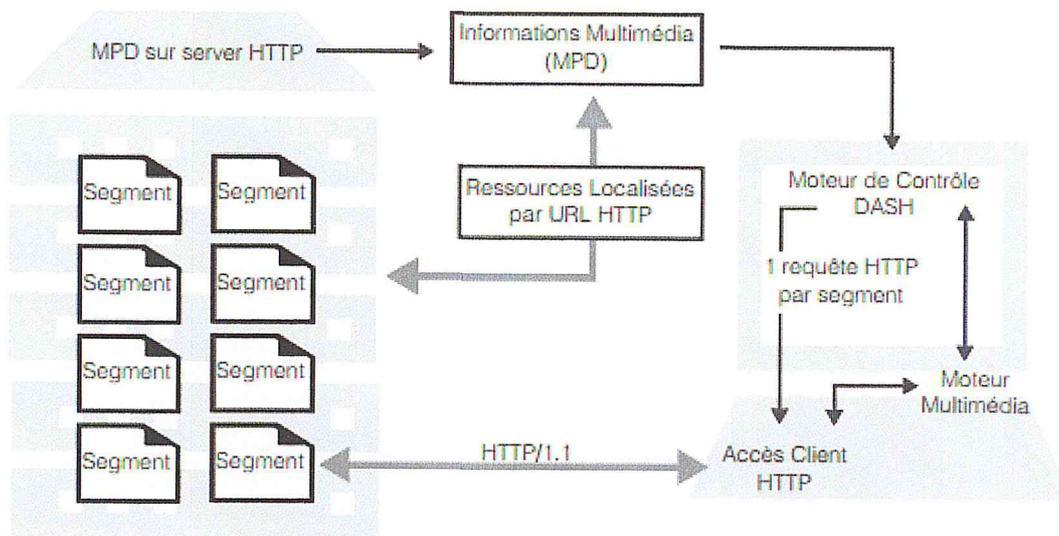


Figure 12: Schéma du fonctionnement de DASH

Dans DASH, le fichier multimédia est divisé en petits segments de durée fixe, par exemple, 2 secondes, qui peut être codé à différents débits binaires ou qualités (appelées représentations).

Par exemple, lorsque vous utilisez un seul codec (multicouche) tel que AVC (SVC), chaque segment a des versions différentes (couches). Les segments sont fournis sur un serveur Web et peuvent être téléchargés via des requêtes GET conformes à la norme HTTP. L'adaptation au débit est effectuée du côté client pour chaque segment.

La logique de streaming permet une évolutivité et une flexibilité élevées. Par conséquent, les algorithmes d'adaptation de débit ne sont pas standardisés dans DASH, l'objectif étant de choisir un débit garantissant une bonne qualité vidéo et d'éviter les interruptions de lecture vidéo inutiles (Congestion). Nous passons en revue les algorithmes d'adaptation proposés dans la littérature dans la section suivante.

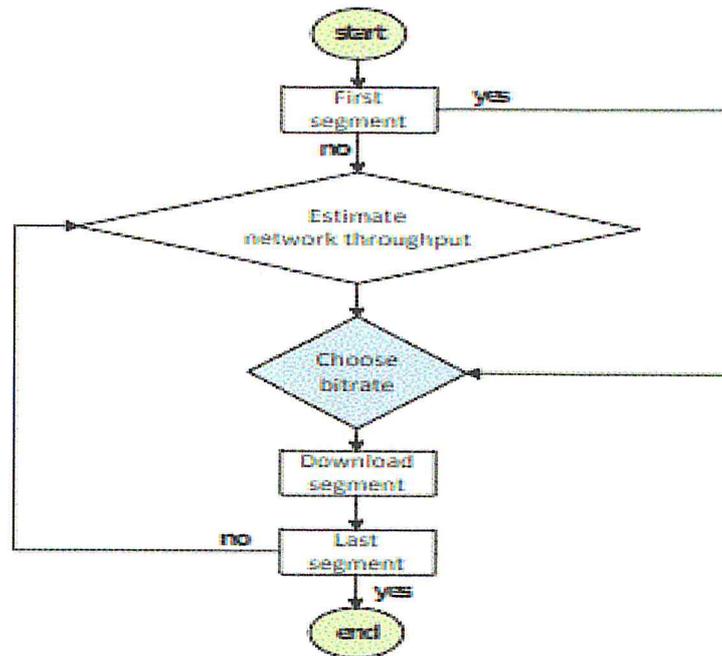


Figure 13: Diagramme de flux pour les algorithmes d'adaptation pour DASH

1.2. HLS (http Live streaming) : [32]

HTTP live streaming est un protocole créé par Apple pour permettre le streaming à destination des terminaux sous IOS, HLS permet de distribuer des contenus en direct et des fichiers à la demande et est l'unique technologie capable de streamer à destination des terminaux Apple. Sachant qu'en janvier 2013, 49% des terminaux mobiles étaient sous IOS 2, HLS est très vite a pris en main par les développeurs d'applications.

Enfin, le choix de compatibilité de beaucoup de serveurs de streaming vis-à-vis de ce protocole tel qu'Adobe, Microsoft, RealNetworks, Akamai ainsi que des clients terminaux tel que Google depuis Android 3.0 ont permis de positionner ce protocole comme le leader du streaming adaptatif via HTTP. Le fonctionnement de HLS est approximativement le même que HDS. Les différences principales résident dans le fait que HLS ne nécessite pas d'extension au serveur Web. Ainsi, toute la logique d'adaptation est gérée au niveau du client. Pour distribuer un multimédia à des clients utilisant HLS, il suffit d'encoder la source en plusieurs fichiers de débit binaire différent et de les diviser en partie, généralement de 10 secondes. Ces fichiers sont hébergés sur un serveur HTTP et un fichier supplémentaire de leur description est

ajouté. Ce fichier de description est utilisé par le client pour choisir la partie de la vidéo à télécharger en fonction de la bande passante disponible à l'instant T. La figure 14 schématise le fonctionnement du protocole HLS.

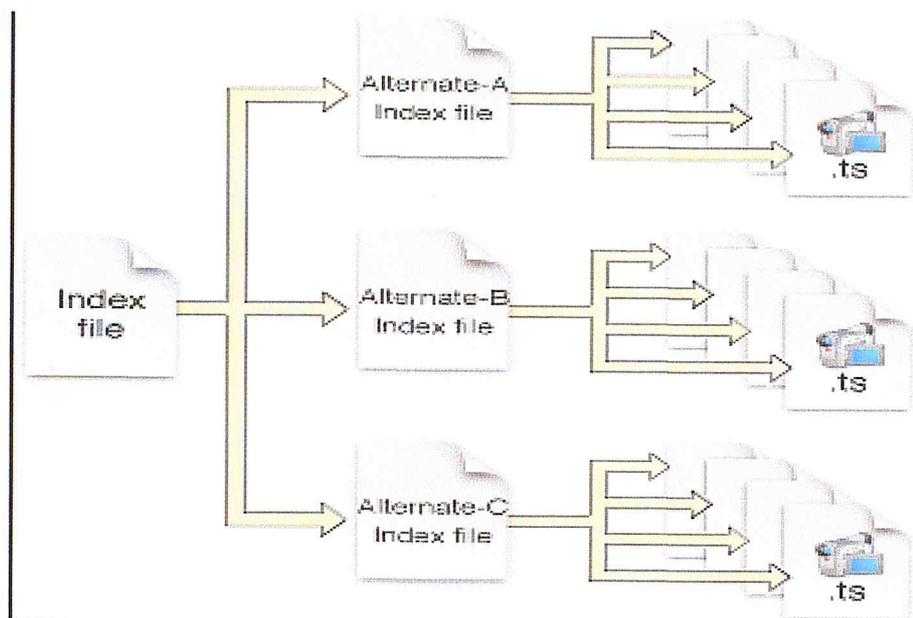


Figure 14 : schéma du fonctionnement de HLS

L'expérience HLS a deux composantes : un ensemble de fichiers fragmentés au format (.ts) et les fichiers manifestes requis. Dans un environnement à la demande, vous pouvez encoder les fichiers alternatifs à l'aide de n'importe quel outil d'encodage H.264 autonome, la dernière version de Sorensen Squeeze offrant un modèle d'encodage HLS à plusieurs fichiers. Les services de codage en nuage comme ceux fournis par Encoding.com peuvent également produire des fichiers compatibles HLS.

Une fois que vous avez les flux codés, vous pouvez utiliser les outils Apple pour créer les fichiers fragmentés et les listes de lecture. Voici les outils HLS disponibles en téléchargement à partir d'Apple :

- **Media Stream Segmenter** – saisit un flux de transport MPEG-2 et produit des fichiers.ts et des fichiers d'index. Il peut également chiffrer le média et produire des clés de chiffrement.

- **Media File Segmenter** – Entre les fichiers H.264 et produit des fichiers .ts et des fichiers d'index. Il peut également chiffrer le média et produire des clés de chiffrement.
- **Variant Playlist Creator** – Compile les fichiers d'index individuels créés par Media Stream ou Media File Segmenter dans un fichier maître .M3U8 qui identifie les autres flux.
- **Générateur de balises de métadonnées** – Crée des balises de métadonnées ID3 qui peuvent être écrites dans un fichier ou insérées dans des segments de flux sortants.
- **Media Stream Validator** – Examine les fichiers d'index, les flux alternés et les fichiers chunked.ts pour valider la compatibilité HLS.

Pour la distribution HLS en direct, vous avez besoin d'un outil de codage capable de coder les fichiers au format H.264, de créer les blocs de flux de transport MPEG-2 et de créer et mettre à jour les fichiers manifestes. Lorsque Apple a annoncé HLS pour la première fois en 2009, seuls deux encodeurs en direct étaient disponibles, chacun d'Intel (maintenant Cisco) et Envivio. Aujourd'hui, la plupart des fournisseurs de matériel d'encodage proposent également des produits HLS compatibles, notamment Digital.

1.3 HDS ([http dynamic streaming](#)) : [33]

HDS est un service de streaming obsolète, basé sur le protocole http. Ce service présente beaucoup d'inconvénients, d'où il est très rarement utilisé de nos jours, d'où nous n'avons pas présenté beaucoup de détails dans ce chapitre.

1.4 Etude comparative des approches :

Le tableau suivant décrit quelques différences entre le HDS, HLS et le Dash[34].

Fonctionnalité	HDS	HLS	Dash
Déploiement sur des serveurs HTTP ordinaires		✓	✓
Youtube, Netfixe...			✓
Norme internationale officielle		✓	✓
Changement de canal rapide	✓		✓
Agnostique aux codecs vidéo			✓
Agnostique aux codecs audio			✓

Tableau 3: Comparaison entre Dash,HLS et HDS

1.5 Etude critique :

A partir de tableau ci-dessus, on trouve que le service Dash est meilleur que le HLS. Le Dash est utilisé par un grand nombre de fournisseur de service de vidéo streaming les plus réponsu dans le monde tel que Youtube et Netflix, ainsi que par pleins d'autres sociétés tel que Microsoft et Adobe.

Seul le Dash offre un mécanisme de signalisation via base URLs au niveau MPD pour le client où plusieurs CDN sont utilisé contre le HLS qui est couteuse car il nécessite des fonctionnalités CDN spécialisées.

Apple HLS utilise généralement des de 10 secondes et est optimisé pour cette taille de bloc, MPEG-DASH fonctionne avec des ploques de 2 et 4 secondes qui permet de changement de chaine plus rapide que HLS car la communication rapide des canaux est une fonction directement liée à la taille de segment

2. L'architecture MEC (Mobile Edge Computing):

Cette partie définit l'architecture MEC, présente les avantages et les inconvénients de cette architecture, propose quelques exemples de son utilisation et enfin elle décrit avec détails l'approche qui vise à améliorer le mécanisme de congestion dans les réseaux sans fil et la compare avec quelques autres approches classiques.

2.1 Introduction

MEC est une architecture réseau qui permet des capacités informatiques et de cloud-computing à la périphérie du réseau cellulaire. L'idée principale derrière l'architecture est de réduire la congestion du réseau et d'améliorer les applications en effectuant des tâches de traitement connexes plus proches de l'utilisateur final. La technologie est conçue pour être mise en œuvre dans les stations de base cellulaires, permettant un déploiement rapide des applications et d'autres services à la clientèle. [35]

L'architecture comprend trois éléments essentiels:

- le système de gestion de l'infrastructure d'hébergement, le système de gestion de la plateforme applicative et le système de gestion des applications.
- Le système de gestion de l'infrastructure d'hébergement comprend un gestionnaire de virtualisation et une couche de virtualisation.
- Le système de plate-forme d'application fournit un contrôle du trafic, des services d'informations de réseau d'accès radio (RAN), des services de communication et un registre de service. Le système de gestion des applications est une machine virtualités pour les applications.

MEC offre une gamme d'avantages, en particulier pour les opérateurs, les fournisseurs d'équipements, les fournisseurs de plates-formes informatiques et les intégrateurs de systèmes. En plus de réduire la congestion des réseaux centraux mobiles, MEC offre une faible latence. De plus, l'informatique de périphérie fournit un moyen de collecter et de traiter des informations sur des périphériques informatiques locaux plutôt que dans le cloud. De plus, MEC peut fournir une analyse des données en temps quasi réel et réduire les dépenses liées aux opérations et à la gestion des données.

Dans la figure suivante on donne un exemple de placement de l'interface MEC :

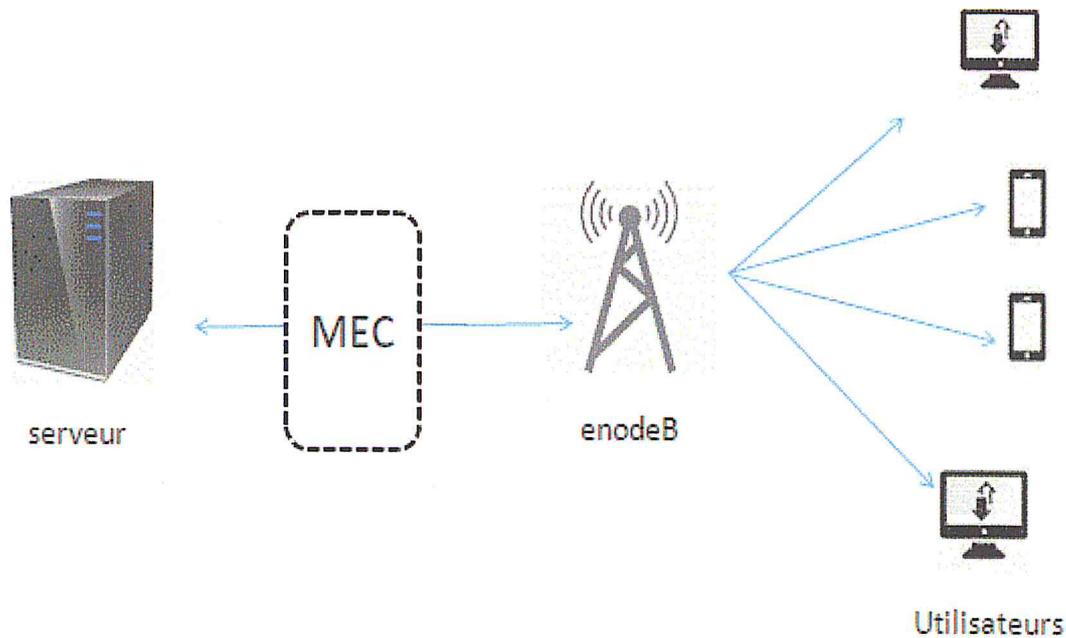


Figure 15: Placement de MEC

L'interface MEC est placée entre l'eNodeB et le serveur généralement pour le filtrage de trafic et la détection et le réglage de la congestion.

Le principe de l'architecture MEC est l'amélioration de la QoE de l'utilisateur de vidéo streaming adaptatif.

Les utilisateurs obtiendront la meilleure qualité vidéo qui va avec leurs commentaires et leurs paramètres de périphériques. Cela peut conduire le réseau à la congestion (si les utilisateurs consomment beaucoup de bande passante en même temps).

Si la probabilité d'encombrement du réseau est trop élevée, ce qui signifie que le réseau va entrer en phase de congestion, la station MEC supprime les plus hautes qualités vidéo disponibles (ce qui consomme beaucoup de ressources réseau) que les utilisateurs peuvent recevoir jusqu'à la congestion du réseau. La probabilité commence à décliner. Si la probabilité d'encombrement du réseau est faible, la station MEC ajoute les qualités vidéo les plus élevées pour permettre aux utilisateurs les plus gourmands de télécharger les qualités requises [35].

2.2 Solution basé sur l'architecture MEC : [36]

2.2.1 Introduction :

Avec la prolifération des appareils mobiles avancés tels que les tablettes et les Smartphones, le vidéo streaming, en particulier le streaming adaptatif basé sur HTTP, gagne en popularité et devient l'un des principaux services Internet pour les consommateurs mobiles. Puisque les dispositifs mobiles contemporains sont habituellement équipés de multiples interfaces radio (par exemple 3G / LTE, Wi-Fi, etc.), l'exploitation de réseaux d'accès sans fil aussi divers pour améliorer les services de vidéo streaming est prometteuse. En cas d'encombrement sur un réseau sans fil, un utilisateur devrait pouvoir passer à un autre réseau en douceur, sans dégradation de QoE. De même, du point de vue des opérateurs, il est essentiel de garantir la QoE des utilisateurs mobiles et de maximiser les performances du réseau en aidant les utilisateurs à prendre la décision de sélection du réseau pour le service de vidéo streaming fourni. Cependant, l'optimisation de la stratégie de sélection de la qualité vidéo pour le vidéo streaming sur plusieurs réseaux sans fil, compte tenu des exigences du service vidéo, des profils de canaux sans fil et des coûts des différents liens reste un problème ouvert. Par conséquent, il existe clairement un besoin pour une application qui peut faire à la fois une sélection de qualité vidéo et une sélection de réseau d'une manière optimale. [37]

La solution proposée est de vidéo streaming HTTP adaptative adaptée à un environnement MEC. Une caractéristique distinctive d'approche est que la fin MEC est un composant actif dans la chaîne de livraison vidéo. En particulier, un service spécial situé sur le bord mobile est chargé de contrôler dynamiquement les représentations vidéo disponibles pour la livraison, en exploitant les caractéristiques spécifiques de la norme MPEG-DASH et la disponibilité des informations sur les conditions dans l'accès sans fil réseau, qui peut être rendu disponible via une API MEC. Le mécanisme est conforme aux normes et compatible avec les algorithmes de livraison vidéo adaptatifs pilotés par le récepteur, avec lesquels il coopère de manière transparente. L'architecture est présentée à la figure 16

Le scénario d'utilisation implique un niveau de coopération entre le CP et le MNO, implémenté via le bord mobile: Un CP héberge des fichiers de descripteur vidéo, potentiellement, dans ses locaux, tout en utilisant des CDN pour héberger et distribuer des vidéos à partir d'Emplacements. Dans le même temps, le CP déploie

des services intelligents sur l'infrastructure périphérique mobile. Ces services MEC sont contrôlés par le CP mais utilisent des informations de niveau réseau fournies par le fournisseur d'accès Internet (FAI) à l'aide d'une API MEC pour aider les mécanismes sophistiqués qui optimisent la diffusion de la vidéo. Dans ce cas, un tel service de niveau MEC assume le rôle de nœud proxy / redirecteur entre l'utilisateur final et les serveurs vidéo. À la base, Le mécanisme intercepte et modifie de manière appropriée la description de présentation multimédia (MPD) MPEG-DASH à la volée visant à faire correspondre la demande de bande passante actuelle avec les ressources disponibles du réseau d'accès mobile et, par conséquent, pour améliorer l'expérience de l'utilisateur.

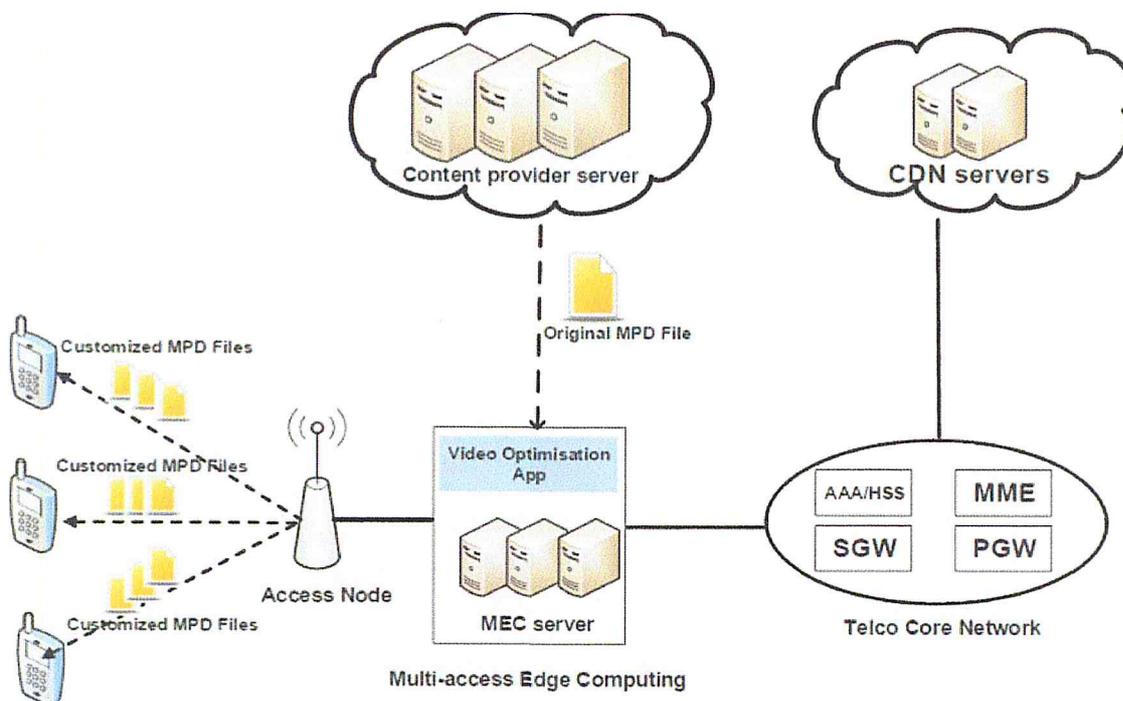


Figure 16: Une architecture pour le streaming adaptatif mobile avec MEC Assistance

La supposition est que l'équipement utilisateur (UE) est déjà connecté au réseau mobile et que le service MEC est capable de récupérer périodiquement des statistiques radios et d'autres informations de niveau réseau à partir du nœud d'accès au réseau. Avant de télécharger la vidéo, chaque utilisateur final accède au catalogue vidéo (potentiellement après un processus d'enregistrement / connexion) et sélectionne un fichier à visualiser. Le lecteur vidéo envoie ensuite une requête

HTTP pour extraire le fichier MPD du serveur du CP. Cette requête est interceptée par le service MEC. Il existe deux modes / stratégies alternatifs pour y parvenir: Mode de redirection Dans ce mode (Figure 17), lorsque le serveur CP reçoit la requête HTTP GET de l'UE pour télécharger un fichier MPD, il pousse le fichier MPD original sur le serveur MEC et avertit les UE de télécharger le fichier MPD à partir du serveur MEC donné en envoyant un HTTP REDIRECT au client. Ensuite, suite à l'indication de redirection fournie par le CP, l'UE envoie une autre requête HTTP GET pour le fichier MPD au serveur MEC approprié, qui répond directement au client en fournissant le fichier, après l'avoir modifié si nécessaire. Mode proxy Dans ce mode (Figure 18), l'application MEC fonctionne comme un proxy HTTP pour tout le trafic utilisateur; s'il détecte une requête HTTP pour un fichier MPD dirigé vers un fournisseur de contenu, il intercepte la réponse et modifie son contenu de manière appropriée, avant de le renvoyer à l'UE.

Dans les deux modes, la même logique d'adaptation est appliquée par le service MEC, les représentations sont supprimées / rajoutées au contenu MPD d'origine en réponse à l'évolution des conditions et de la demande du réseau, mais également avec la possibilité d'utiliser une foule d'autres informations, telles que les profils et les préférences des utilisateurs.

Après avoir reçu le fichier MPD (transformé), les UE commencent à télécharger du contenu vidéo morceau par morceau. Notez que, dans notre schéma, le fichier MPD doit être actualisé périodiquement. Il existe deux options conformes aux normes pour y parvenir. La première consiste à configurer son attribut minimum Update Period, qui demande au client vidéo de demander périodiquement une version à jour du fichier. L'autre option utilise un événement intra-bande (mécanisme de boîte de message d'événement, dans la terminologie BMFF (Base File Format) ISO . Dans ce cas, un champ de zone d'événement spécial est inséré dans le segment de support au moment de la préparation de la vidéo par le fournisseur de contenu, qui peut informer le client d'actualiser le fichier MPD à tout moment.

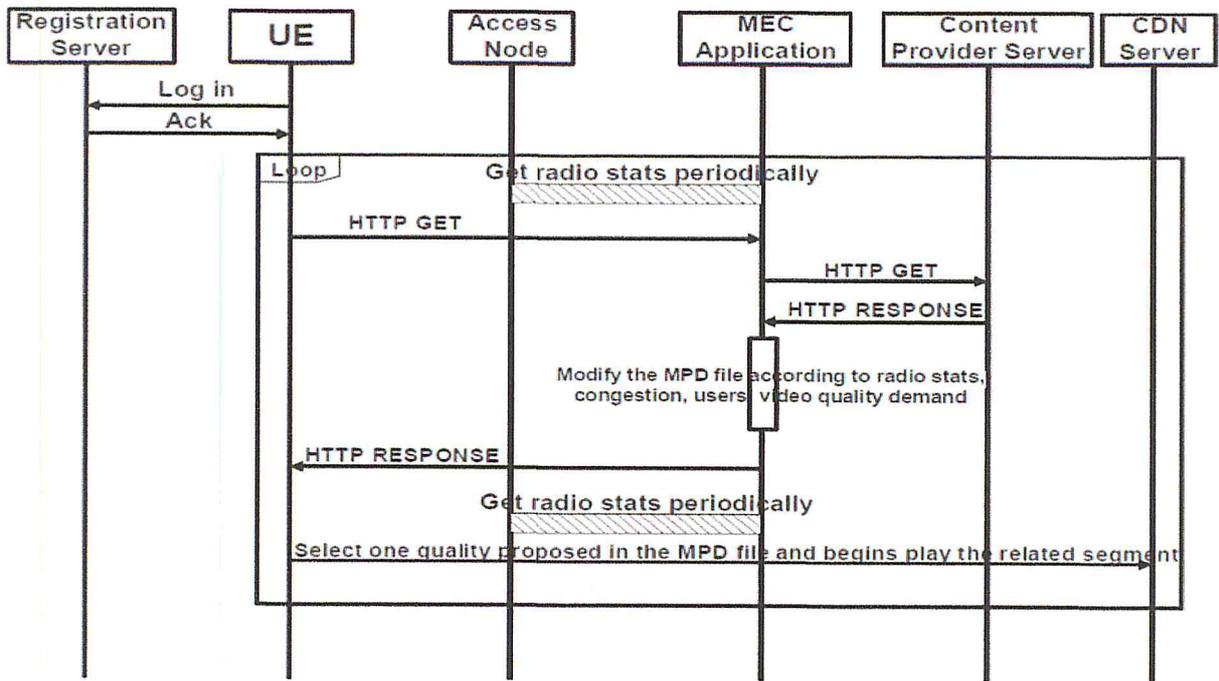


Figure 17: Séquence de message (mode de redirection)

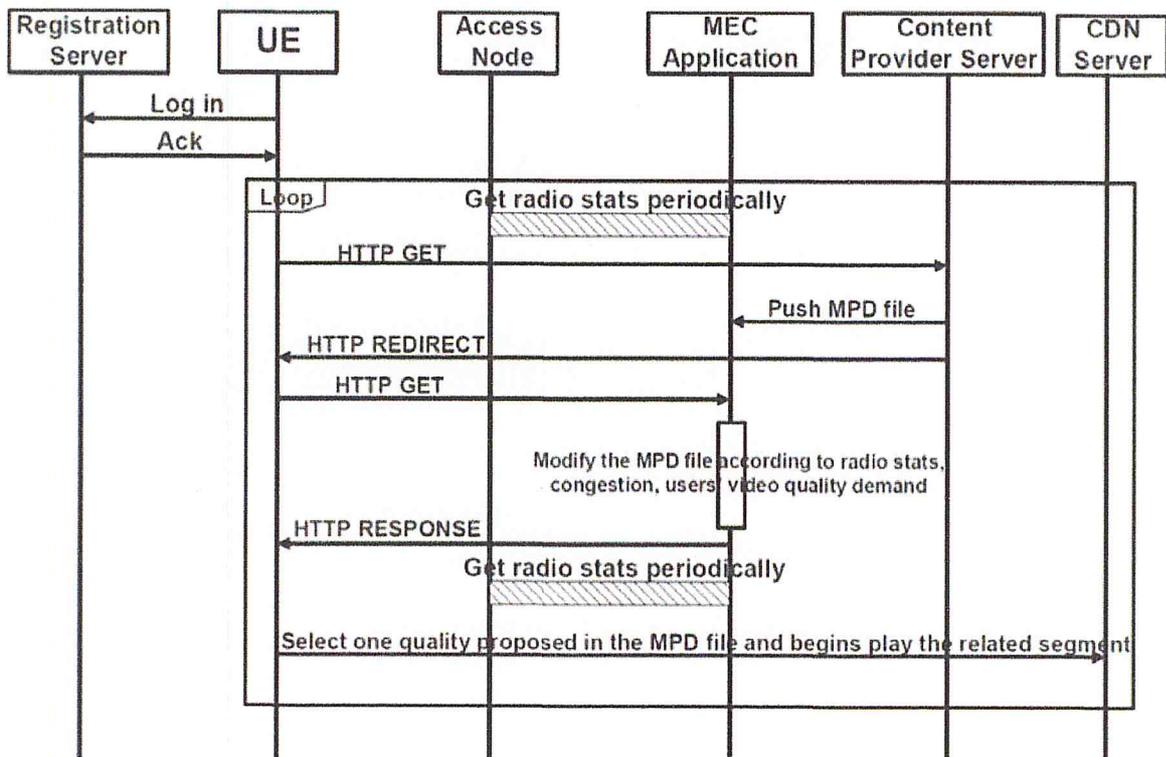


Figure 18: Séquence de message (mode proxy)

3. différences approches de la gestion de congestion dans les réseaux sans files :

De nos jours les services de vidéo streaming adaptatif sont parmi les services les plus utilisés sur internet. Plusieurs approches ont vu le jour dans ce domaine, quelques unes de ces approches ont prouvé leurs efficacités par rapport à d'autres notamment en matière de QoE. Dans ce qui va suivre on va décrire trois approches différentes : une qui est la plus utilisée de nos jours, une autre qui est concurrente à la première et enfin une qui a vu le jour récemment. On termine par une comparaison des trois approches.

Dans le cadre de l'amélioration de la solution de la gestion de la congestion dans les réseaux sans files plusieurs approches ont été proposées.

3.1 TCP Reno : [38]

Jacobson propose également dans un autre algorithme qui rend la transmission plus efficace. Cette transmission est qualifiée de plus proactive lors de la détection d'une perte lors de la transmission. En effet, dans la version Tahoe, ce n'est qu'après l'apparition du time-out que l'algorithme se rend compte d'une perte de paquet. La solution proposée dans l'algorithme de 'Fast Retransmit' est de partir sur cette constatation et de s'apercevoir d'une perte de paquet avant le time-out. L'algorithme de Fast Retransmit préconise que dès réception d'un nombre n de DUPACKs, un paquet a pu se perdre. Une perte de paquet, signe d'une congestion, déclenche un algorithme pour lequel on retransmet les paquets perdus. Cette retransmission s'effectue avec une fenêtre de congestion unitaire, même si la perte s'est produite avec une taille de fenêtre élevée. Ceci n'est pas efficace puisque la congestion s'est produite pour une $cwnd$ supérieure à cette valeur unitaire, et il n'est pas utile de la réduire autant. Le compromis proposé par Jacobson est de re-initier une transmission avec une valeur de fenêtre de congestion réduite de moitié et pour laquelle on continue la transmission en congestion avoidance. Il s'agit de l'algorithme de Faster-Recovery. L'ensemble de ces derniers apports à la version de TCP Tahoe forme ce que l'on appelle la version de TCP Reno. Cette version garde l'inconvénient d'être peu rapide pour la réémission de données suite à un Time Out à cause d'une granularité peu fine utilisée dans le système d'horloge pour le calcul des RTT

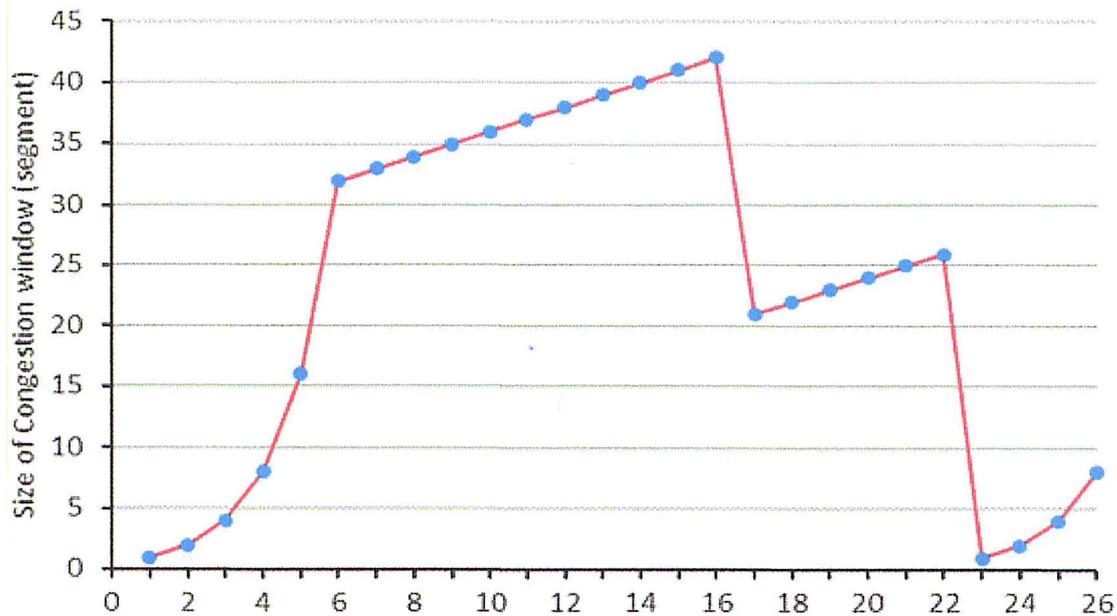


Figure 19: TCP Reno

3.2 Solution inter-couche pour le contrôle de Congestion de TCP dans les réseaux sans files : [39]

Le principe général de la solution 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.

L'approche cross-layer ou inter-couches s'appuie sur les interactions entre deux niveaux de l'architecture OSI (Open System Interconnexion). 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 le travaille, se justifie par l'utilisation du champ « Réserve » qui se trouve dans l'en-tête de segment TCP (figure 20). A l'origine ce champ a été réservé pour un usage futur.

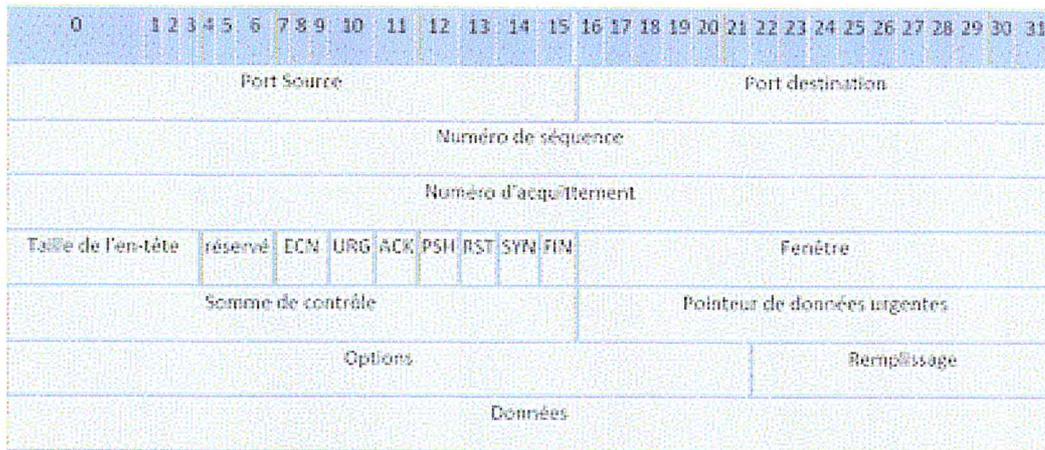


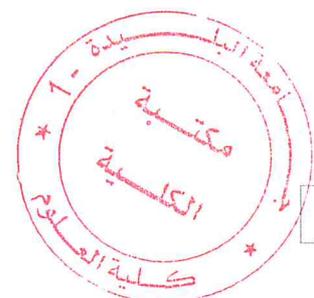
Figure 20: structure d'un segment TCP

3.2.1 Formulation de la solution:

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 considéré comme perdus ou pas.

$$RTO = \min [UBOUND, \max [LBOUND (BETA * RTT)]]$$



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. Après que TCP prédit les valeurs de PSSM et de PNMX de la prochaines séquence de paquets a envoyé, il commence à procéder par élimination.

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.

L'organigramme suivant explique les étapes de la solution :

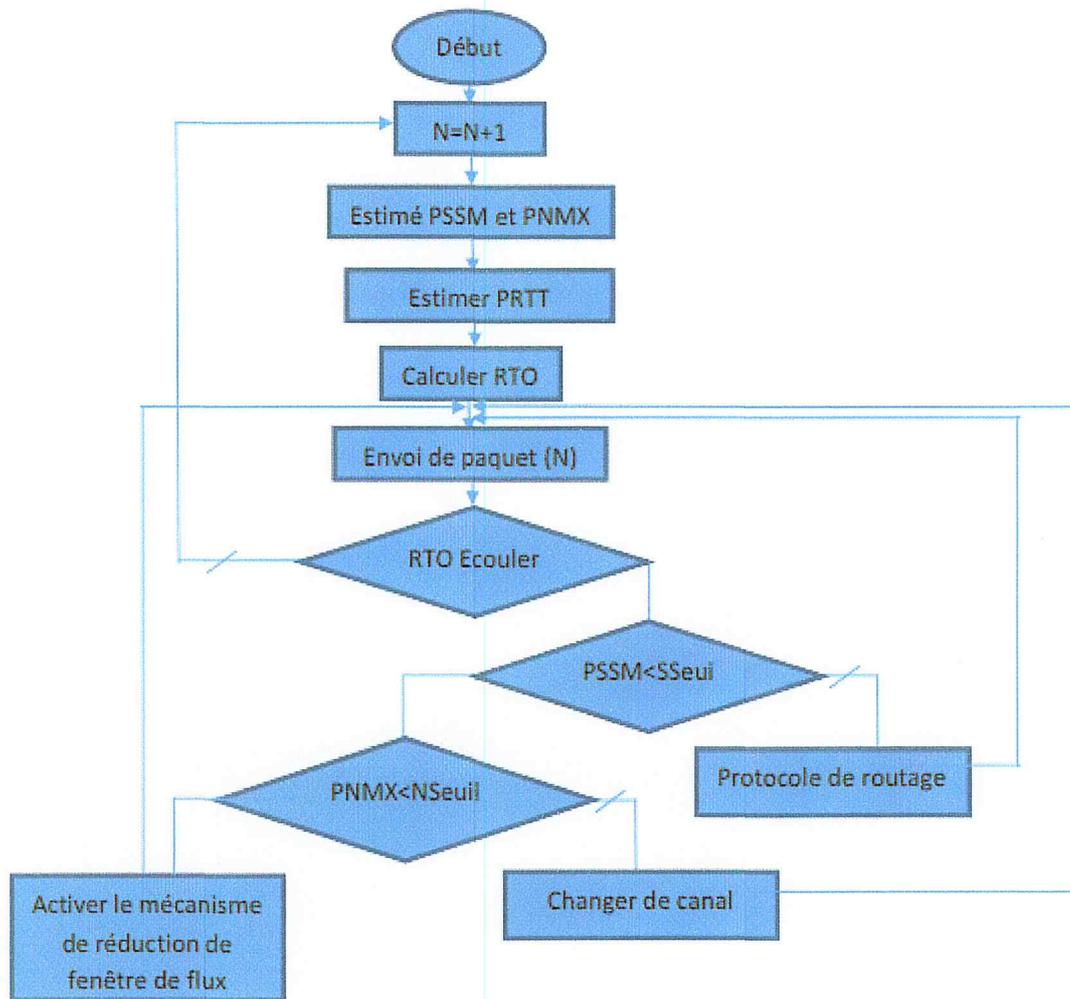


Figure 21: Organigramme de la solution

Algorithme de description du mécanisme proposée

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_écouler(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

3.3 Comparaison générale :

D'après les différentes approches qui essaient d'améliorer le comportement et les performances du protocole TCP et la gestion de congestion dans les réseaux sans files, on va citer quelques différences entre les trois approches:

Le TCP Reno est basé sur le type de couche réseau par contre la solution de contrôle de congestion est basée sur le type de multicouche.

Le TCP Reno ne donne pas un bon résultat et performance en ce qui concerne la gestion de congestion dans les réseaux sans files [40] puisque le TCP Reno divise la fenêtre de congestion par deux pour chaque perte, jusqu'à un minimum de 2 paquets.

Le TCP Reno est basé sur le fragment des paquets par contre la deuxième solution est basée sur la puissance de signal, et la solution qui de MEC est basée sur l'annulation des qualités par rapport au bande passante disponible.

4. La congestion dans les enodeBs : [41]

Vu l'amélioration et le développement de l'internet, ainsi le développement des outils de service de ses utilisateurs.

Et puis le problème de la surcharge et la congestion dans les enodeBs est apparu, ce problème est devenu un obstacle face au bon déroulement des services de l'internet.

La majeure cause de ce problème est le grand nombre des demandes des utilisateurs, ce qui amène à l'accroissement des débits qui va donc dépasser la contenance de ce dernier. En effet, la bande passante devient insuffisante et là le problème de congestion commence pour ne pas s'arrêter et il va s'élever avec l'augmentation des demandes des clients car le débit dépasse l'enodeB et puisque l'enodeB ne peut pas diminuer ou faire arrêter les demandes, ils restent toujours élevés ainsi la congestion ne cesse pas d'augmenter. Même quand on agrandit le matériels on ne peut pas réaliser un service qui nous qualifie à faire disparaître ce problème, alors on doit essayer de trouver une solution pour décroître la congestion et c'est ce qu'on a mis en lumière dans notre mémoire on a pour objectif de diminuer la congestion et d'améliorer la qualité d'expérience des services de vidéo streaming.

5. Etude critique :

Le problème 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.

C'est une des problèmes qui empêchent le cross-layer c'est que Les modifications aux différentes couches peuvent avoir comme objectif l'optimisation d'une métrique qui leur est commune (ex. Le délai) dans des directions opposées, ce qui peut avoir des conséquences négatives car la modification d'une métrique au niveau d'une couche peut avoir un impact implicite sur les autres

6. Problématique :

Avec la prolifération des appareils mobiles avancés tels que tablettes et Smartphones, le vidéo streaming devient l'un des principaux services Internet pour mobile consommateurs. Le trafic vidéo mobile, qui représentait 55% du trafic mobile total en 2015, représentera plus de 70% en 2020[42]. Cette croissance significative s'accompagne d'une adoption généralisée de la norme DASH [30], l'un des implémentations de streaming adaptatif basé sur http(HAS).Dash offre aux clients la possibilité de basculer entre différents qualités vidéos (et donc, débits binaires) en cas de variation des conditions du réseau. Cela permet d'éviter les interruptions de lecture et améliorer ainsi le QoE. Cependant, lorsque les clients partagent le même réseau, le protocole DASH montre quelques instabilités, injustice, et sous-utilisations de la bande passent. Il y a donc un besoin d'être conscient des conditions réseau au niveau de l'enodeB pour optimisé la diffusion des vidéos et traiter efficacement les cas de congestion, se contexte besoin d'une architecture bien définit. Dans ce contexte Mobile Edge Computing(MEC) fournit un environnement hautement distribué qui peut utiliser pour déployer des applications et des services, ainsi que stocker et traiter du contenu à proximité des utilisateurs mobile [28]. Les applications peuvent également être exposées à des informations réseau en temps réel et offrir une expérience personnalisée à l'abonné mobile.

CHAPITRE 03 :

CONCEPTION

1. Introduction

Dans ce chapitre, nous exposons notre solution d'amélioration de la qualité d'expérience dans les services de vidéo streaming basés sur l'architecture MEC. L'idée est de proposer un algorithme au niveau de la Station MEC qui permet d'éviter la congestion en se basant sur des études probabilistes de la surcharge de l'enzyme en agissant sur les qualités vidéo disponibles.

2. Architecture :

L'idée de Notre architecture est de centraliser tous les calculs et les traitements au niveau de la Station MEC. La Station MEC doit pouvoir détecter en temps réel la surcharge de l'Enodeb et ajuster en conséquence les qualités vidéo proposées à l'utilisateur. Si on est en face à une présence d'une surcharge de l'Enodeb (ce qui signifie que le réseau va entrer en phase de congestion), le processus est déclenché.

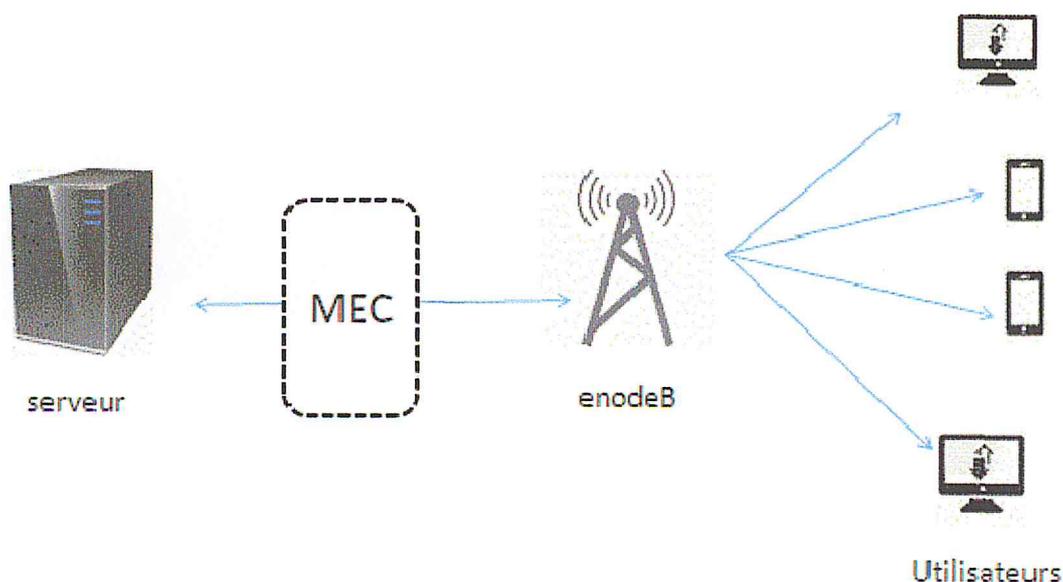


Figure 22: Placement de MEC

3. Solution et algorithme :

3.1 Description de la solution :

Dans cette section soit :

PC : probabilité de Congestion.

TTR : le temps d'attente après chaque calcul de probabilité de congestion.

T : compteur de temps.

Q1, ..., Q5 : Les qualités disponibles dans le fichier MPD, où Q1 la qualité faible et Q5 la qualité forte.

S1, ..., S5 : les seuils de la probabilité de congestion pour chaque qualité.

Intervalle de PC	La qualité
$<S_5$	Q5
$[S_5, S_4 [$	Q4
$[S_4, S_3 [$	Q3
$[S_3, S_2 [$	Q2
$\geq S_2$	Q1

Tableau 4: Les qualités selon la variété de PC

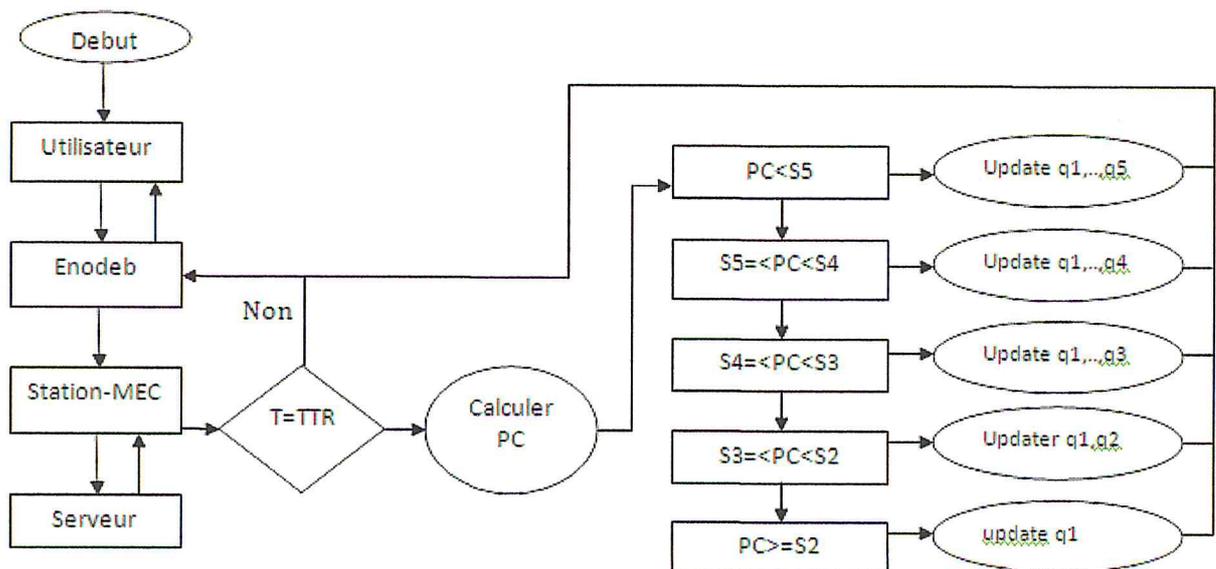


Figure 23: Schema de la solution proposée

Dans la figure ci-dessus, nous avons présenté les opérations effectuées dans les différents nœuds de notre solution.

Les notations utilisées dans la figure ci-dessus sont les suivantes :

1. Utilisateurs : qui effectuent des requêtes pour accéder au service de vidéo streaming afin de recevoir la vidéo stockée dans le serveur.
2. L'énodeB : qui est le point d'accès par lequel l'utilisateur se connecte.
3. La Station MEC : où nous allons implémenter notre algorithme proposé.
4. Le serveur : qui stocke le fichier MPD et les vidéos (pas forcément).

Formule de probabilité de congestion :

On considère la probabilité de congestion(PC), comme le résultat de rapport de la Bande Passante Utilisée dans le réseau BPU sur la Bande Passante Total BPT que la carte réseau peut servir :

$$PC = BPU/BPT$$

Lorsque l'utilisateur lance une requête de vidéo, l'énodeB envoie cette requête à la Station-MEC qui va contacter le serveur afin de récupérer le fichier MPD (Media Présentation Description) [43] de vidéo concernée, à la réception de fichier MPD par la Station-MEC, elle calcule la probabilité de congestion (PC) sur le réseau, selon cette valeur, la station-MEC mets à jour le fichier MPD selon l'Algorithme suivant :

3.2 Algorithme de la solution :

Algorithme de la solution proposée

Variable :

Q1, Q2, Q3, Q4, Q5; // les qualités disponibles dans le fichier MPD.

S1, S2, S3, S4, S5 ; // les seuils de la probabilité de congestion pour chaque qualité.

PC ; //probabilité de congestion

MPD : Fichier ;

T, TTL : réel ;

i ; [1,..., n] ; //ensemble des utilisateurs

Début

Start(T); // lancer un chronomètre ;

Listen(Req,i) ; // mettre la station-MEC en écoute d'une requête

Requête (req,i) ;

Send(Req,server) ; // envoyer la requête d'utilisateur vers le serveur

Recieve(Req) ;

Response(Mec) ;

Get(MPD,i) ; // récupérer le fichier MPD

Si(t \geq TTR) alor

PC \leftarrow Calcule () ; // calculer la probabilité de congestion

Fin Si ;

Selon(PC) alors

Cas 1: <S5

MPD \leftarrow Update (MPD, Q5); // modifier le fichier pour que les qualités de Q1 à Q5 soient disponibles

Cas 2: \geq S5 et <S4

MPD ← Update (MPD, Q4) // modifier le fichier pour que les qualités de Q1 à Q4 soient disponibles

Cas 3: $\geq S4$ et $< S3$

MPD ← Update (MPD, Q3) // modifier le fichier pour que les qualités de Q1 à Q3 soient disponibles

Cas 4: $\geq S3$ et $< S2$

MPD ← Update (MPD, Q2) // modifier le fichier pour que les qualités Q1 et Q2 soient disponibles

Cas 5: $> S2$

MPD ← Update (MPD, Q1), // modifier le fichier pour que Q1 soit la seule qualité disponible

Else : “le dernier message pour n’afficher que les conditions “ ;

Fin Selon ;

Send(MPD, user) ;

Reset(T);

Fin

Au début, on a lancé un chronomètre pour exécuter notre solution quand le temps dépasse une durée TTR, donc, après chaque TTR on calcule la probabilité de congestion, on récupère le fichier MPD de chaque utilisateur et on lance la modification dans le fichier MPD selon PC par la fonction Maj qui va laisser dans le fichier MPD les qualités jusqu’à la qualité mentionnée dans son paramètre.

Exemple : Maj (MPD, Q3) signifie qu’après l’exécution de cette fonction, le fichier MPD contient les qualités Q1, Q2, Q3.

Les étapes de notre solution qui sont mentionnées dans l’algorithme sont les suivantes :

On met la station MEC en écoute d’une requête.

L’utilisateur lance une requête pour une demande d’un vidéo streaming

Cette requête est récupérée par la station MEC ;

La station MEC sauvegarde l'adresse source et l'adresse de destination (adresse de client et l'adresse de serveur), ensuite, il transmet la requête vers le serveur

Le serveur reçoit la requête;

Il envoie tous les fichiers MPD à la station MEC

La station MEC reçoit les fichiers MPD et sauvegarde l'adresse de la destination (adresse de client)

La station MEC calcule la probabilité de congestion par rapport à la connexion disponible i.e.: la bande passante disponible sur le réseau (cette calcul est faite après chaque TTR)

D'où:

Si la probabilité de congestion est inférieure à la seuil de la qualité 5, il envoie le fichier MPD sans modification

Si la probabilité est supérieure à la seuil de la qualité 5 et inférieure à la seuil de la qualité 4, la station MEC supprime la qualité 5 (la qualité supérieure) dans le fichier MPD, et envoie le fichier au client.

Si la probabilité est supérieure à la seuil de la qualité 4 et inférieure à la seuil de la qualité 3, la station MEC supprime la qualité 4 et envoie le fichier MPD au client.

Si la probabilité de congestion est supérieure à la seuil de la qualité 3 et inférieure à la seuil de la qualité 2, la station MEC supprime la qualité 3 dans le fichier MPD et envoie le fichier MPD au client.

Si la probabilité de congestion est supérieure de la qualité 2, la station MEC supprime la qualité 2 dans le fichier MPD, et envoie le fichier MPD au client (une seule qualité).

CHAPITRE 04 :
IMPLIMENTATION ET
RESULTATS

1. Introduction :

Dans ce chapitre, nous présentons l'environnement d'émulation, de notre solution de l'amélioration de la qualité d'expérience des services de vidéo streaming adaptative, et les différents schémas de l'implémentation. Enfin nous discutons les résultats de cette émulation.

2. Environnement de test :

2.1 System d'exploitation :

L'implémentation et la validation de notre approche ont été réalisées dans un environnement Linux(Ubuntu)[44]. Celui-ci est un system d'exploitation basé sur la distribution Linux (noyau Linux), Ubuntu est donc une distribution GNU/Linux qui réunit stabilité et convivialité. Elle s'adresse aussi bien aux particuliers qu'à la professionnelle, débutante ou confirmé qui souhaitent disposer d'un système d'exploitation libre et sécurisé. « Ubuntu » est un ancien mot africain qui signifie « Humanité » et également « Je suis ce que je suis grâce à ce que nous sommes tous ».

Ubuntu est un projet communautaire de création du système d'exploitation et d'un ensemble complet de programmes informatiques utilisant des logiciels libres et gratuits (c'est pour ça on a choisi Ubuntu).

2.2 Topologie d'émulation :

La figure suivante représente la topologie de l'émulation :

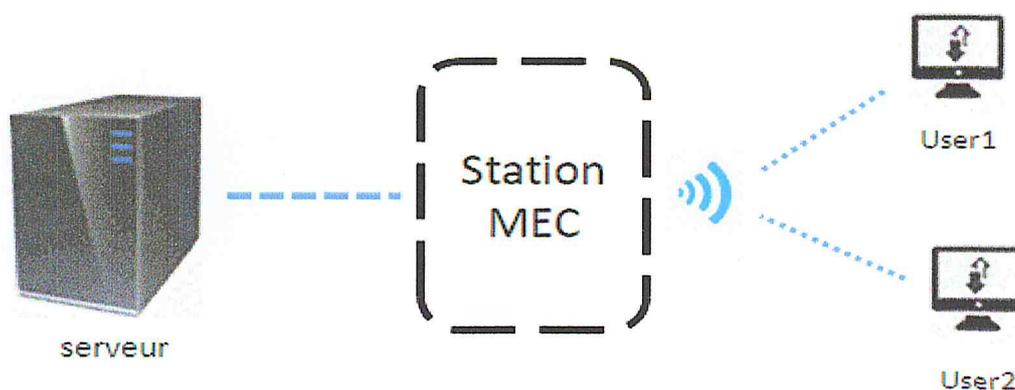


Figure 24: Topologie d'émulation

La topologie de réseau composée de :

- **Pc portable (PC1)** : De marque HP (pavillon g6), avec les caractéristiques suivantes :

- Processeur : Intel® I 3

- Mémoire installée (RAM) 2.00 Go

-Type de système : Système d'exploitation 64 bits

Et contenant :

Un serveur Apache [45] qui stocke les fichiers. Le serveur est le responsable pour l'exécution des deux logiciels (**FFMPEG**, **MP4Box**) et la fourniture de vidéo streaming pour les clients mobiles.

-**FFMPEG [46]** : est utilisé pour encoder un fichier source vidéo en différentes qualités vidéo et un seul fichier audio.

MP4Box [47] : consiste à diviser les fichiers encodés en petits segments et à créer un fichier manifeste (MPD) afin de relié les fichiers segmentés.

Lecteur DASH.js [48] :

Dash.js est un lecteur de vidéo MPEG-DASH open source écrit en JavaScript. Son objectif est de fournir un lecteur robuste, inter-plateformes qui peut être réutilisé librement dans les applications qui requièrent une lecture vidéo. Il assure la lecture MPEG-DASH dans n'importe quel navigateur prenant en charge W3C Media Source Extensions (MSE)

Pour créer une page web simple qui affiche un lecteur vidéo avec les contrôles courants comme Lecture, Pause, Retour rapide, etc., vous devez effectuer les tâches suivantes :

- Créer une page HTML
- Ajouter la balise vidéo

- Ajouter le lecteur dash.js
- Initialiser le lecteur
- Ajouter un style CSS
- Afficher les résultats dans un navigateur qui implémente MSE

- **PC portable (PC2)** : de marque **acer aspire-E1-760**, avec les caractéristiques suivantes :

- Processeur : Intel® I 03
- Mémoire installée (RAM) 4.00 Go
- Type de système : Système d'exploitation 64 bits

Ce pc représente notre station MEC

Et contenant :

Deux interfaces :

Enp2s0f0 :c'est une interface Ethernet relié la station MEC avec le serveur.

Wlp3s0 : c'est une interface sans file qui est représente le point d'accès pour les clients.

Ce PC est responsable de l'exécution des logicielles suivants :

-**WonderShaper**[49] : est un outil disponible pour les ordinateurs basés sur Linux. Il permet de façonner facilement la bande passante disponible sur une interface à une valeur définie.

On a utilisée cet outil pour atteinte les caractéristiques d'un réseau étendu.

On a utilisée la commande suivante :

```
Sudo wondershaper -a <network interface> -d <downspeed> -u <upspeed>
```

Les valeurs downspeed et upspeed représente la bande passante disponible pour télécharger et envoyer en kbits.

- **MITMProxy[50]** : Est un proxy interactive (homme au milieu) qui peut intercepte les requêtes et les réponses http et https et les modifier à la volée.

```
sudo apt install python3-pip
sudo pip3 install -U pip
sudo pip3 install mitmproxy
```

- **Le script :**

Afin de réalisé notre solution on a choisir le langage python [51] comme un langage de programmation.

On a divisé notre programme en deux parties qui fonctionnent en parallèle :

Partie 1 : un script (**bandwidth .py**) pour calculer la band passante dans l'interface (**wlp3s0**) qui partage les données aux clients.

Partie 2 : les opérations effectuer dans ce scripte sont les suivantes :

- Récupérer la valeur de la bande passante calculée dans **bandwidth.py** (figure25)
- Faire calculer la probabilité de congestion(PC),(figure26).
- Interceptor chaque fichier MPD échangé entre le serveur et les clients
- Stocker le fichier MPD (figure27)
- Modifie chaque fichier MPD selon la probabilité de congestion actuel
- Donner on réponse un fichier MPD modifié (figure28)

Ces opérations sont réalisées au niveau de la station MEC avec MITMProxy. Le gros avantage de MITMProxy par rapport à des solutions similaires, c'est la possibilité de créer des scripts en python qui appliquera divers transformations sur les requêtes et les réponses, en fonction de condition ou « event ».

```
def get_bytes(t, iface='wlp3s0'):
    with open('/sys/class/net/' + iface + '/statistics/' + t + '_bytes', 'r') as f:
        data = f.read()
    return int(data)
```

Figure 25: Récupération de la bande passante

```
used_rate = tx+rx #used rate in wlp3s0
seuil=3000000 #la seuil de bande passante imposer par wondershaper
proba= 1-(used_rate/seuil)
```

Figure 26: Calcule de PC

```
with open('modified.mpd', 'w') as mpd:
    data = mpd.write(flow.response.get_text())
```

Figure 27: Stockage de fichier MPD

```
doc.write("modified.mpd")
img = open("modified.mpd", "rb").read()
flow.response.content = img
```

Figure 28: Application de modification dans la réponse

3. Scenario de teste :

Durant toute la phase de test, on considère des scénarios avec deux utilisateur (les clients PC1et PC2) qui veulent voir un vidéo en directe, la durée de la vidéo utilisé est de 12 minutes, les clients reçoivent de serveur une mise à jour de fichier MPD chaque 5 secondes. Dans le fichier MPD, nous avons encodé une vidéo source avec différents débits binaires comme suite:

(Qualité, bit rate (kbps)):

(1080, 5300), (720, 2200), (480, 1060), (360, 600), (240, 260).

Les qualités sont représentées par des variables Ou : Q5=1080; Q4=720; Q3=480; Q2=360; Q1=240;

(Audio bit rate (kbps)): (128).

3.1 Scenario 01 :

Après l'installation de notre matériel on va commencer le teste en montrant l'effet de choix de seuil sur notre interface (wlp3s0) à la probabilité de congestion, pour obtenir la valeur de la bande passante qui va nous guider au cas de la congestion. Ce test nous facilite le travail qui suit après, on va lancer la vidéo sans appliquer notre solution et on laisse les demandes des deux clients telles qu'elles sont.

On a choisir de commencer le test avec un seuil de 100 Mbps et on diminue après chaque calcul de PC la valeur du seuil. Les résultats sont représentés dans la figure suivante :

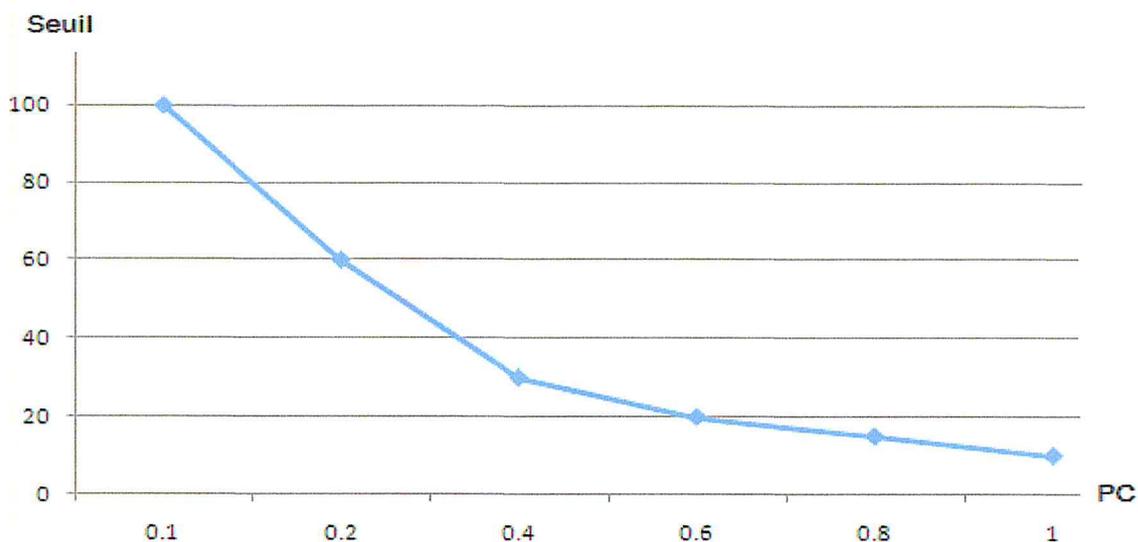


Figure 29: résultats de scénario 1

3.1.2 Discussion des résultats:

L'analyse du graphe (figure 29) montre que : Plus le seuil est petit, plus la probabilité de tomber dans un cas de congestion est grande. L'explication est que les deux clients demandent la meilleure qualité qui se trouve dans le fichier MPD, sans considérer l'abaissement du seuil de la bande passante.

3.2 Scenario02 :

Le but de ce scénario est de montrer les choix appliqués par notre solution aux qualités de vidéo selon la probabilité de congestion (PC).

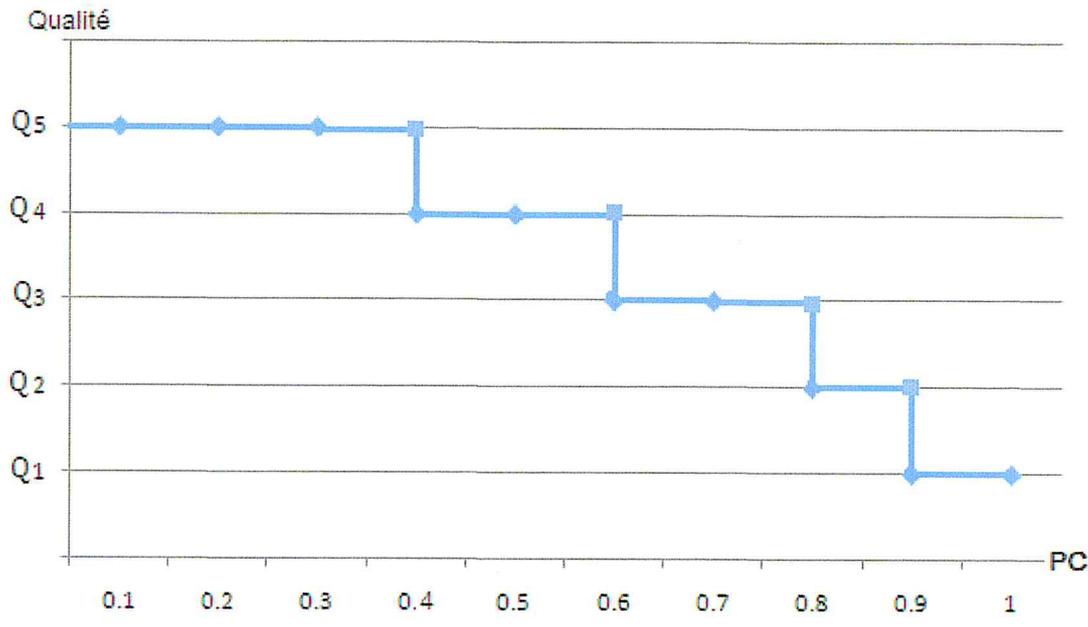


Figure 30: les changements de qualité selon PC

3.2.1 Discussion des résultats:

Dans le scénario illustré par la figure [30], la valeur de probabilité de congestion (PC) est le responsable de choix de qualité. Il existe pour chaque qualité une valeur de PC ou il faut changer la qualité afin d'éviter la congestion.

3.3 Scenario 03 :

Dans ce scénario on calcule le nombre des coupures par rapport à la probabilité de congestion.

On fait le test en deux étapes, la première sans script et la deuxième avec script, puis les clients regardent la vidéo avec différentes qualités.

On lance la vidéo et on démarre les calculs pendant 1 min (de 30s jusqu'à 1.5 min), et on prend le nombre des coupures chaque 5 s.

Avec la réalisation de tests on a obtenu le résultat suivant :

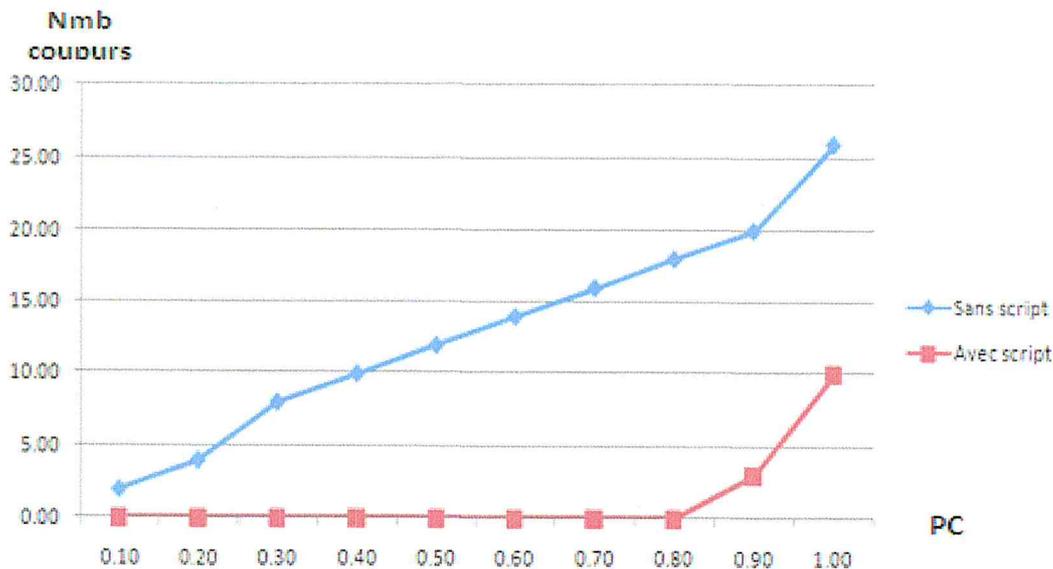


Figure 31: Nombre des coupures par rapport à PC

3.3.1 Discussion des résultats:

Dans ce graphe, on remarque que dans le teste sans script, le nombre de coupures est augment au même temps que la valeur de la probabilité de congestion augment.

Dans le teste avec script on remarque que le nombre des coupures est égale toujours zéro jusqu'à la valeur de la probabilité de congestion est égale 0.8 (dernière qualité de vidéo), donc le script ne trouve aucun qualité dans le fichier MPD pour supprimé

A mesure que la qualité du vidéo augment, la surcharge sur le réseau est augment, donc l'utilisation de la bande passent augment et le débit devient insuffisant pour répondre aux demandes des clients, alors il dépasse la seuil et ça créer une congestion sur le réseau, et elle va s'élever à chaque demande des clients.

Mais dans le cas du teste avec script à chaque fois que la probabilité de congestion arrivé à une grande valeur, on diminue la qualité du vidéo donc la probabilité se diminue et la vidéo est marche sans occupation, et le client regarde la vidéo avec qualité moins qu'elle a été avant.

3.4 Scenario 04 :

Dans ce scenario, nous proposons d'étudier les résultats de perception utilisateurs sur une vidéo encodée en différentes qualités, nous avons amené 20 client pour avoir un teste efficace, nous demandent a ces clients de regarder le vidéo, puis on mettre leurs votes en fonction de la note MOS attribuée et nous faisons un teste sans script et avec script, on a obtenu le résultat suivant :

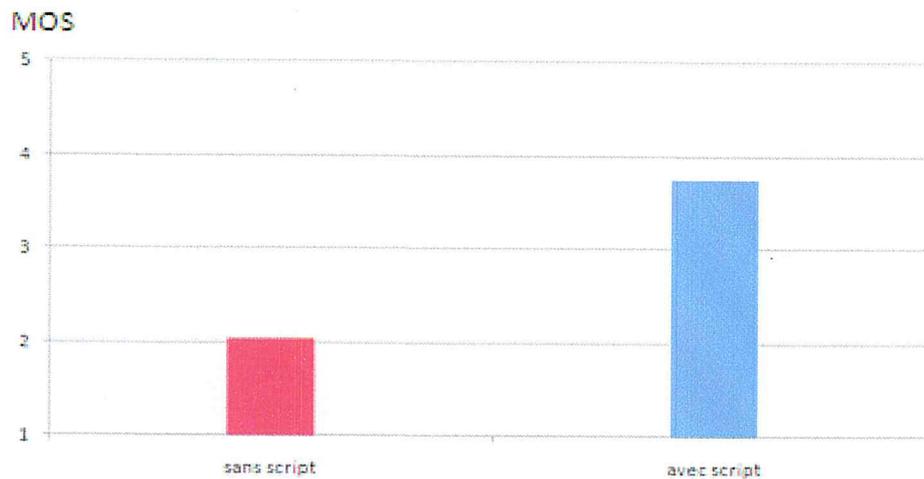


Figure 32: calcul de MOS

3.4.1 Discussion des résultats:

A partir de ce graphe, on remarque que la moyenne donné par les clients dans le premier test (sans script) est 2.1 qui représente la deuxième valeur de MOS (pauvre), et la moyenne donné dans le deuxième teste (avec script) est 3.7 qui est égale presque à la quatrième note de MOS (bien). Donc les résultats du test avec script sont mieux que les résultats de teste sans script.

Nous prenons comme conclusion que la plupart des utilisateurs ne s'intéressent pas à la qualité du vidéo, comme il est important pour eux que la vidéo ne se coupe pas et ils la regardent sans occupation, surtout ces derniers temps la plupart des vidéos sont regardées en direct « live ».

Conclusion générale & perspectives :

L'objectif de notre travail fut de proposer une solution pour l'amélioration de la qualité d'expérience des services de vidéo streaming adaptatif dans les réseaux sans fil basé sur l'architecture MEC. Pour faire ce la, il a été nécessaire dans un premier temps de mentionner et définir les notions de base afin d'enrichir les connaissances des lecteurs dans le domaine et mieux les aider à comprendre la suite de la mémoire en élevant toute ambiguïté. Dans l'état de l'art on a cité, critiqué et comparé les approches de vidéo streaming existants auparavant, on a fait la même chose avec des travaux existants sur notre problématique, cette phase nous a permis de choisir le protocole DASH et l'architecture MEC pour appliquer notre solution. L'idée de notre solution est d'améliorer la qualité d'expérience en contrôlant la congestion au niveau de l'endeb. Pour cela on a récupéré la surcharge de l'endeb et calculer la probabilité de congestion afin de modifier le fichier MPD selon la bande passante disponible. La solution a été testée dans différents scénarios avec différentes conditions. Les résultats obtenus ont été comparés avec d'autres résultats (sans appliqué la solution) sous les mêmes conditions. Après comparaison, notre approche a répondu à notre attente. On a diminué les nombre de fois ou l'endeb tomber dans un cas de congestion.

Perspectives :

Le travail réalisé a donné des résultats qui ont certes la satisfaction des utilisateurs, mais l'implémentation actuelle comprend tout de même certaines limites :

- L'implémentation de cette solution c'est juste une émulation, donc les résultats qu'elle nous a donnés ne sont pas à 100% complets. Notre perspective est d'appliquer cette solution sur des cas réels, et faire ainsi des vrais testes, ceci implique l'investissement d'un fournisseur de streaming acceptant d'intégrer le modèle au sein de son service.
- Dans les testes qu'on a fait on a basé sur un seul paramètre dans le processus d'adaptation qui est la congestion, donc comme suggestion au future, on peut prendre aussi d'autres critères du processus d'adaptation comme la résolution maximale d'affichage du terminal utilisé par l'utilisateur, et la taille de l'écran et le niveau de la batterie d'utilisateur.

Bibliographie :

- [1] www.journaldunet.com
- [2] [www. ISO.org](http://www.iso.org)
- [3] Romain LEGRANDE André VUCAMPS, Connaissance approfondies sur le réseau ,2006.
- [4] M.Gast. 802.11 wireless network ,the definitive guide. O'Reilly Media,2005.
- [5] Robert Arcomano, Guide pratique du réseau sans fil, Version français du Wireless Howto,2000-2002.
- [6] J.Postel, "Transmission Control Protocol", RFC 793, IETE , September 2009.
- [7] V.Paxson and M.Allman, Computing TCP retransmission timer, novembre 2000.
- [8] H. wang, G. Shin, Robust TCP Congestion Recovery. Distributed Computing Systems, Version français ,2007.
- [9] Vincent Roca. "Un Etat de l'art sur les Techniques de Transmission Multipoint Fiable. Distributed Systems Engineering (DSE) Community. Decembre 2001.
- [10] Khaldoun Al Agha, Guy Pujolle, Resaus de mobiles & réseaux sans fil, Edition Eyrolles, 2001.
- [11] Réseau locaux industriel, ARNATRONIC PLUS, Novembre 2002.
- [12] Boudjaadar Amina ; « Plateforme basée Agents pour l'aide à la conception et la simulation des réseaux sans fil ». Thèse de Magistère ; Université de Skikda ; 2009/2010.
- [13] Matther Gast, 802.11 Réseaux sans fil, 2 emme édition, O'Reille, 2005
- [14] www.urec.cnrs.fr
- [15] S.Pomports, Mobilité et gestion efficace de congestion dans un réseau ad hoc à fort efficacité, Université Paris-sud, décembre 2011.
- [16] Santi, Paolo, Topology Control in Wireless Ad Hoc sensor Network, 2005.
- [17] Mohamed Alreshoodi and John Woods . Seryery on qoe/qos correlation models for multimedia services, 2013
- [18] R. Atkinson, " Security architecture for the internet protocol" ,2004.

- [19] H.Zimmerman, "http Live Streaming",Apple Developer .web.2012.
- [20] S.Bauer,D.D Clark,and W .Leher,"The evolution of internet congestion" TPRC,2009.
- [21] 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.
- [22] Nasser Sedaghati-Mokhtari, Mahdi Nazm Bojnordi, Nasser Yazdani, "Cross-Layer Design:A New Paradigm", Proc In International Symposium on Communications and Information Technologies (ISCIT '06), On page(s): 183- 188, Bangkok, Sept 2006.
- [23] Q. Wang, M. A. Abu-Rgheff, "Cross-Layer Signalling for Next-Generation Wireless Systems," Proc In IEEE Wireless Communication and Networking , New Orleans, Mar. 2003.
- [24] V.T.Raisinghani, S.Iyer, "Cross-layer feedback architecture for mobile device protocol stacks", IEEE Communications Magazine, Volume 44, Issue 1, Page(s): 85 - 92, Jan 2006.
- [25] R.Winter, J.H.Schiller, N.Nikaein, C.Bonnet, "CrossTalk : cross-layer decision support based on global knowledge", in IEEE Communications Magazine, ISSN: 0163-6804, pages 93- 99, Volume: 44, Issue: 1, Jan 2006.
- [26] S.Khan, Y.Peng, E.Steinbach, M.Sgroi, W.Kellerer, "Application-driven cross-layer optimization for video streaming over wireless networks", In IEEE Communications Magazine, ISSN: 0163-6804, pages 122- 130, Volume: 44, Issue: 1, Jan 2006.
- [27] International Data Corporation (IDC), Worldwide Public Cloud Services Spending Forecast to Double by 2019, According to IDC, <https://www.idc.com/getdoc.jsp?containerId=prUS40960516>, [Online; accessed 15-September-2016] (2016).
- [28] M. T. Beck, M. Maier, Mobile Edge Computing: Challenges for Future Virtual Network Embedding Algorithms, in: Proceedings of the 8th International Conference on Advanced Engineering Computing and Applications in Sciences ,2014.
- [29] H. Li and G. Shou and Y. Hu and Z. Guo. Mobile Edge Computing: Progress and Challenges. In IEEE International Conference on Mobile Cloud Computing, Services, and Engineering, March 2016.
- [30] DASH Industry Forum. For Promotion of MPEG-DASH 2013. Available :<http://dashif.org> .
- [31] Ricky K. P.Mok and al. " QDASH :A QoE-aware DASH System".In :proceedings of the 3rd Multimedia Systems Conference :ACM,2015

[32]Apple-Inc. http live Streaming Overview 2013.

[33]R.Pantos, HTTP Dynamic Streaming on the Adobe Flash Platform,2015

[34] <https://bitmovin.com/mpeg-dash-vs-apple-hls-vs-microsoft-smooth-streaming-vs-adobe-hds>.

[35]Several authors, « "Mobile Edge Computing",Introduction Technical White Paper.2014

[36]M. P and AL, "Mobile edge computing introductory technical whitepaper",September 2015.

[37]d.Yue Li "Edge Computing-based access eNetwork Selection for Heterogeneous Wireless Networks", Université de Rennes1,septembre 2017.

[38]V. Jacobson,"Congestion avoidance and control", SIGCOMM symposium on Communications Architectures and protocols,2009.

[39]D .Yassine "Amélioration de la performance de TCP dans les réseaux mobiles ad hoc", thèse de doctorat, Novembre 2016

[40] S.Floyd,"TCP and successive fast retransmits", lawrence Berkeley Laboratory, september 2013.

[41] www.efort.com/r_tutoriels/CONGESTION_M2M_EFORT

[42] www.webrankinfo.com/chiffres-statistiques

[43] www.abbreviations.com

[44]www.docs.ubuntu.com

[45]www.httpd.apache.org/docs

[46]www.ffmpeg.org

[47] www.gpac.wp.imt.fr/mp4box

[48]www.dash.docs.org

[49]<https://documentation.online.net/fr/dedicatedserver/tutorials/network/configure-wondershaper>

[50] <https://docs.mitmproxy.org/stable/tools-mitmproxy/>

[51] www.python.org/doc

