I. Introduction :

Les réseaux locaux basés sur la technologie IEEE 802.11 ont pris une ampleur telle qu'ils sont déployés un peu partout dans notre entourage quotidien (aéroports, hôtels, gares, campus, etc.).Ce déploiement est favorisé par la maturité atteinte par le standard grâce aux travaux des groupes 802.11 chargés de rendre le standard plus compétitif (QoS, sécurité, haut débit).

Le groupe de travail 802.11 e répond aux challenges de garantie de la qualité se service (QoS) aux applications temps réel en définissant de nouveaux mécanismes d'accès au médium. Le draft résultant des travaux du groupe 802.11 e propose deux nouveaux mécanismes : EDCA et HCCA.

Dans ce chapitre nous nous intéresserons plus particulièrement à la gestion de la QoS et ses contraintes dans les réseaux IEEE 802.11. Dans un premier temps, nous passerons en revue les mécanismes de QoS les plus significatifs proposés dans la littérature ; nous nous attarderons sur les deux mécanismes du draft 802.11 e EDCA et HCCA. Dans un deuxième temps, nous introduirons un autre mécanisme de la QoS c'est le protocole du lien direct permettant la communication directe entre les stations dans un mode de fonctionnement avec infrastructure.

II. Généralité sur la qualité de service :

II.1 Définition de la QoS:

Plusieurs définitions ont été proposées pour le terme de la qualité de service dont les plus importantes sont :

- La **Qualité de Service** (QoS) est la capacité à véhiculer dans de bonnes conditions un type de trafic donné, en termes de disponibilité, débit, délais de transmission, taux de perte de paquets…
- La **Qualité de Service** est une notion subjective. Selon le type d'un service envisagé, elle pourra résider dans le débit (Un débit permet de mesurer le flux d'une quantité relative à une unité de temps au travers d'une surface quelconque.), le délai (pour les applications interactives ou la téléphonie), la disponibilité (accès à un service partagé) ou encore le taux de pertes de paquets (pertes sans influence de la voix ou de la vidéo (La vidéo regroupe l'ensemble des techniques, technologie, permettant l'enregistrement ainsi que la restitution d'images animées…). [1]
- La **Qualité de Service** regroupe un ensemble de technologies mises en œuvre pour assurer des débits suffisants et constants sur les réseaux, y compris Internet. [2]

II.2 But de la QoS :

Le but de la QoS est donc d'optimiser les ressources du réseau (Un réseau informatique est un ensemble d'équipements reliés entre eux pour échanger des informations. Par analogie avec un filet (un réseau est un « petit rets », c'est-à-dire un petit filet), on appelle nœud (node) l'extrémité d'une connexion, qui peut être une intersection de plusieurs connexions (un ordinateur, un routeur, un concentrateur, un commutateur) et de garantir de bonnes performances aux applications critiques. La Qualité de Service sur les réseaux permet d'offrir aux utilisateurs des débits et des temps (Le temps est un concept développé pour représenter la variation du monde : l'Univers n'est jamais figé, les éléments qui le composent bougent, se transforment et évoluent pour l'observateur qu'est l'homme. Si on considère l'Univers...) de réponse différenciés par application suivant les protocoles mis en œuvre au niveau de la couche réseau.

Elle permet ainsi aux fournisseurs de services (départements réseaux des entreprises, opérateurs…) de s'engager formellement auprès de leurs clients sur les caractéristiques de transport (Le transport, du latin trans, au-delà, et portare, porter, est le fait de porter quelque chose, ou quelqu'un, d'un lieu à un autre.) des données (Dans les technologies de l'information (TI), une donnée est une description élémentaire, souvent codée, d'une chose, d'une transaction d'affaire, d'un événement, etc.) applicatives sur leurs infrastructures IP. [4]

Selon le type d'un service envisagé, la qualité pourra résider :

- Le débit (téléchargement ou diffusion vidéo).
- Le délai (pour les applications ou la téléphonie).
- La disponibilité (accès à un service partagé).
- Le taux de pertes de paquets. [5]

II.3 Services de la QoS :

La mise en place de la qualité de service nécessite en premier lieu la reconnaissance des différents services:

- La source et la destination du paquet.
- Le protocole utilisé (UDP/TCP/etc.).
- Les ports de source et de destination dans le cas TCP et UDP.
- La congestion des réseaux.

- La validité du routage (gestion des pannes dans un routage en cas de routes multiples par ex.)
- La bande passante consommée.
- Les temps de latence.

II.4 Critères de la QoS :

Les principaux critères permettant d'apprécier la qualité de service sont les suivants :

- **Débit** (en anglais bandwidth): parfois appelé bande passante, il définit le volume maximal d'information (bits) par unité de temps (b/s).
- **Perte de paquet** (en anglais *packet loss*): elle correspond à la non-délivrance d'un paquet de données, la plupart du temps due à un encombrement du réseau.
- **Gigue** (en anglais jitter) : C'est un paramètre important pour les applications communicantes de type voix ou vidéo où la gigue doit être la plus faible possible. La gigue est due principalement aux délais de transferts variables dans les nœuds du réseau (swicthes et routeurs).
- **Latence** (en anglais *delay*) : elle caractérise le retard entre l'émission et la réception d'un paquet. [5]

II.5 Degrés de la QoS :

Les trois principaux degrés de Qualité de Service (trois niveaux de services), du plus fiable au plus lâche, sont les suivants :

II.5.1 Le service garanti ou premium :

Il vise à émuler une liaison spécialisée : malgré un multiplexage des paquets sur le médium, le lien propose les mêmes garanties que s'il était basé sur une ligne indépendante. Des pertes de paquets ou une certaine gigue peuvent néanmoins être acceptées en fonction du contrat négocié. Au niveau technologies, le service garanti se retrouve avec le GS d'IntServ, l'EF de DiffServ et le CBR de l'ATM que nous détaillerons plus loin.

II.5.2 Le service « mieux que Best-Effort ».

II.5.3 Le service Best-Effort :

Le protocole IP de base en est un exemple, ou encore UBR de l'ATM. [6]

III. Qualité de service suivant le standard IEEE 802.11 :

Pour assurer une qualité de service adéquate dans les réseaux sans fil le standards IEEE 802.11 à définit deux méthodes d'accès au canal:

- Distributed Coordination Function (DCF)

- Point Coordination Function (PCF)

Les deux méthodes sont bien illustrer dans le chapitre précédant.

III.1 Problématique de la QoS dans les réseaux IEEE 802.11:

Le développement du réseau Internet et le grand nombre d'utilisateurs connectes a ce réseau imposent le recours a des supports de qualité de service. Dans cette perspective, plusieurs groupes de travail ont vu le jour pour les réseaux filaires. Les nouveaux besoins en termes de mobilité des utilisateurs et la croissance des réseaux permettant le nomadisme des utilisateurs ont fait migrer le problème vers la boucle locale sans fils, entre autres les réseaux IEEE 802.11. Actuellement, le marché des télécommunications des réseaux Hots-pot est relativement faible mais on s'attend à ce qu'il subisse une croissance accrue les prochaines années. Les fournisseurs d'accès a Internet commencent à mettre en place un large nombre de hots-pots 802.11 ou Wifi dans les divers lieux publics. Des applications multimédia telles que la voix sur IP ou la vidéo sur demande en plus des applications classiques seront de plus en plus utilisées dans ce type de réseaux. Ces applications multimédia nécessitent un niveau minimal de qualité de service en termes de bande passante, de délai, de gigue ou de taux de perte. D'autres types d'applications avec des contraintes plus aigues en termes de QoS commencent à émerger. Des applications du standard 802.11 en milieu industriel pour la commande et la supervision des systèmes ou en milieu médical pour la télémédecine imposent des exigences strictes en termes de QoS (délais + taux d'erreurs). La réponse à ces besoins accrus en QoS dans les hots-pots 802.11 est d'autant plus difficile a cause des caractéristiques spécifiques du medium sans fils. En effet, pour la couche physique DSSS permettant un débit au-delà de 11 Mbps, parmi 11 canaux possibles, seulement 3 ne se chevauchent pas. Ce medium présente alors un taux de perte assez élève à cause des interférences. En plus, les caractéristiques du support physique ne sont pas constantes et varient dans le temps et dans l'espace. Quand les utilisateurs bougent, les chemins de bout en bout changent et les utilisateurs se réassocient chaque fois a des nouveaux APs.

Ces utilisateurs doivent avoir la même QoS indépendamment de leurs associations et du chemin de bout en bout du trafic. Plusieurs travaux de recherche ont essaye d'évaluer les performances du standard IEEE802.11 quant a sa capacité de répondre aux besoins en termes de QoS des utilisateurs. Ces travaux ont investigue essentiellement les possibilités offertes par la sous couche MAC du standard pour garantir un niveau

minimal de QoS pour les utilisateurs. Dans le même objectif, d'autres travaux ont adopte des modèles analytiques ou des approches par simulation. Plusieurs solutions ou approches pour l'amélioration du support de QoS par la couche MAC 802.11 ont été proposées. [3]

Toutes ces insuffisances dans les modes de fonctionnement DCF et PCF du standard ont conduit à plusieurs activités de recherche pour améliorer les performances de la sous couche MAC 802.11.

III.2 Limites en termes de QoS du standard IEEE 802.11 : [3]

Le contrôle d'accès au medium, le maintien de la QoS et la sécurité sont les fonctions les plus importantes de la sous couche MAC 802.11. Cependant plusieurs limitations se présentent quant au support de la qualité de service.

III.2.1 Limitations de la méthode d'accès de base DCF :

Le protocole CSMA/CA utilisé avec cette méthode permet un accès Best Effort au canal. Les utilisateurs ne peuvent avoir aucune garantie de qualité de service minimale. Toutes les stations d'un même BSS concourent pour l'accès au canal et aux ressources du réseau avec les mêmes priorités. Aucun mécanisme de différenciation entre plusieurs types de flux n'est mis en place pour garantir la bande passante, le délai de bout en bout ou la gigue pour des trafics a hautes priorités tels que la voix sur IP ou la vidéo/visioconférence. Le taux des erreurs dues a la couche physique 802.11 est a peu prés trois fois plus grand que celui observe dans les réseaux locaux filaires. Le nombre important de collisions et de retransmissions implique des délais de transmission imprévisibles et une dégradation de la qualité de transmission des flux temps réel tels que pour la voix ou la vidéo.

III.2.2 Limitations de la méthode d'accès PCF :

Spécialement conçue pour apporter un support de qualité de service en priorisant les applications temps réel par rapport aux autres, cette procédure d'accès avec scrutation souffre de plusieurs défaillances. Tout d'abord ce mode ne peut être utilisé qu'en alternance avec le mode d'accès DCF et ne peut jamais fonctionner a part entière. PCF présente tous les inconvénients d'une approche centralisée tel que l'effet d'une défaillance du point central. En plus, à faible charge, les stations voulant émettre en mode PCF subiront des délais très élevés.

Elles seront obligées d'attendre d'être scrutées avant d'émettre. De plus, le coordinateur (généralement confondu avec le point d'accès) doit systématiquement

accéder au canal sans fil lors de la période DCF afin de débuter la période PCF suivante. Dans le mode PCF, il sera très difficile de répondre aux besoins d'un nombre important de trafics temps réel sans pénaliser les applications qui se dérouleront par la suite dans la période avec contention. Un autre problème de ce mode est l'impossibilité de prévoir la durée de transmission des stations sollicitées. Une station sollicitée par le point coordinateur peut transmettre un MSDU de taille maximale 2304 octets. Cependant, le standard n'empêche pas sa fragmentation en plusieurs MPDU. Ceci, en plus des débits de transmission dépendant de l'état du canal physique, conduit a une durée de transmission d'un MSDU non contrôlée par le point coordinateur ce qui induira des délais supplémentaires pour le reste des stations en mode PCF. Enfin le mode PCF est géré par un algorithme de scrutation Round-Robin à une seule classe. Il ne lui est donc pas possible de répondre aux besoins de QoS de plusieurs types de flux (voix, vidéo,…).

III.3 Les différentes solutions de QoS dans les réseaux IEEE 802.11 :

Depuis l'écriture du standard IEEE 802.11 a la fin des années 90, plusieurs propositions, issues de travaux de recherches et/ou d'initiatives de la part de constructeurs, ont vu le jour pour l'amélioration du support de qualité de service dans ces réseaux. Un groupe de travail spécifique à été forme au sein de l'IEEE dans l'objectif de normaliser des amendements de la qualité de service pour le protocole 802.11. La norme 802.11e a ainsi été élaborée. Elle reprend entre autres des techniques introduites dans divers travaux de recherche. Dans la suite de ce chapitre nous présentons tout d'abord la norme IEEE 802.11e puis nous présenterons plusieurs approches visant à améliorer la QoS dans les réseaux 802.11.

III.3.1 Le nouveau standard IEEE 802.11 e :

Pour supporter la qualité de service, le groupe de travail "e" du standard 802.11 définit des améliorations de la couche MAC de 802.11 en introduisant une fonction de coordination hybride HCF. Ce dernier définit deux mécanismes d'accès au canal (synonyme d'accès au medium dans 802.11e) : accès avec contention et accès contrôlé. La méthode d'accès avec contention est nommée EDCA. La deuxième méthode, offrant un accès contrôlé, est nommée HCCA. Les stations sans fils opérant sous 802.11e sont appelées stations améliorées. La station améliorée qui joue le rôle de contrôleur central au sein de la même cellule QBSS est appelée le point de coordination hybride (HC). Le point de coordination hybride est typiquement combine au point d'accès. Un QBSS est un BSS qui inclut un HC et des stations améliorées. Les paramètres QoS sont ajustes au

cours du temps par le coordinateur hybride et sont annonces périodiquement a travers les trames balises. Plusieurs entités de Backoff (Backoff Entity) fonctionnent en parallèle dans une station améliorée. Une entité de Backoff est une file de transmission pour une classe de trafic bien déterminée avec des paramètres d'accès au canal spécifiques. Une station 802.11e ou plus précisément une entité de Backoff ne peut utiliser le canal que pour une durée limitée. L'intervalle de temps durant lequel la station a le droit d'émettre est appelé l'opportunité de transmission TXOP. TXOP est défini par un instant de début et une durée. Un intervalle TXOP obtenu suite à une contention au canal est appelé EDCA-TXOP. Quand cet intervalle est obtenu dans la période contrôlée par le HC, il est appelé HCCA-TXOP. La durée d'une EDCA-TXOP est limitée par la valeur du paramètre QBSS-limit-TXOP régulièrement distribuée par le point d'accès à travers les trames balises (beacon). Ce paramètre permet donc de contrôler la durée maximale d'une transmission en cours ce qui est important pour les délais d'accès et de transmission de l'ensemble des stations. L'utilisation de ce paramètre permet aussi d'assurer a un instant précis et sans retard, le démarrage de chaque période d'accès contrôlée par le HC.

Une autre amélioration est apportée par le nouveau standard : les stations améliorées sont maintenant autorisées à transmettre directement des trames a une autre entité du QBSS sans être obligées de passer par le point d'accès. Ce fait permet d'optimiser l'utilisation de la bande passante partagée entre les utilisateurs. Dans le standard 802.11, toutes les communications passaient obligatoirement par le point d'accès. [3]

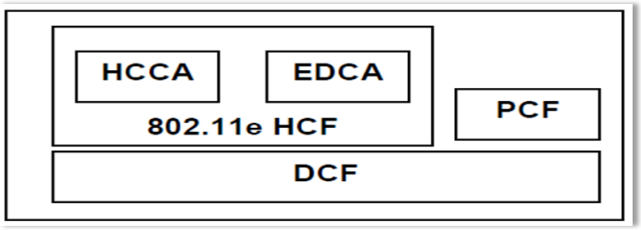

Figure 3.1 : Architecture de la norme 802.11e

i. HCF : une fonction d'accès au médium avec QoS

HCF est utilisée uniquement dans ce que le standard appelle un réseau QoS (c'est le réseau où le point d'accès met en place un HC). La modification IEEE 802.11e a introduit cette nouvelle méthode ainsi que d'autres mécanismes afin d'apporter certaines propriétés QoS au niveau de l'accès. HCF introduit des modifications à DCF et PCF ainsi qu'un certain nombre de mécanismes et de types de trames permettant la mise en place de transferts avec qualité de service pendant la CP et la CFP. HCF

introduit la notion d'opportunité de transmission (TXOP) qu'une QoS-STA peut obtenir en utilisant l'une des méthodes d'accès d'HCF : la méthode d'accès avec contention EDCA ou la méthode d'accès par scrutation (HCF). L'obtention d'un TXOP peut permettre l'envoi d'une ou plusieurs trames. Si TXOP vaut 0, une seule trame donnée peut être envoyée par opportunité de transmission. [9]

- **La méthode d'accès EDCA : [5], [7]**

Il s'agit d'une amélioration du DCF qui ajoute un système de priorité pour la gestion de l'accès au support. Ce dernier se fait alors selon le niveau de priorité de la trame. Selon la définition du dernier draft de la norme 802.11e, la couche MAC au niveau d'une station est formée de quatre files de transmission dont chacune fonctionne comme une entité de Backoff en mode DCF. La structure de cette couche est illustrée par la figure 3.1.

La norme IEEE 802.11e a donc défini, au niveau MAC, quatre catégories d'accès : AC relatives aux applications traitées dans les couches supérieurs. Chaque catégorie de trafic constitue une file d'attente FIFO. Elles sont notées respectivement :

- ***AC_VO*** : pour les applications temps réels tel que la voix
- ***AC_VI*** : pour les applications vidéo
- ***AC_BE*** : pour le trafic « Best Effort »
- ***AC_BK*** : pour le trafic Background

Pour introduire la notion de différentiation entre les différentes AC, Chaque catégorie de trafic possède son propre DIFS, on parle donc de AIFS. Ces catégories de trafic, gèrent huit niveaux de priorités allant de 0 à 7 relatives à la norme 802.1D. Les correspondances entre ces priorités et les catégories d'accès sont récapitulées aussi au niveau de la figure 3.1. En outre, il est important de signaler que les tailles limites de la fenêtre de contention diffèrent selon la classe de trafic. On parle alors de CW_{Min} [AC] et CW_{Max} [AC].

Chaque AC détient son propre compteur de Backoff qui est désormais compris entre 1 et 1 + CW [AC].

Quand deux ACs finissent en même temps leur durée de Backoff, alors c'est le paquet de plus haute priorité qui sera transmis, les autres entités doivent augmenter leurs fenêtres de Backoff.

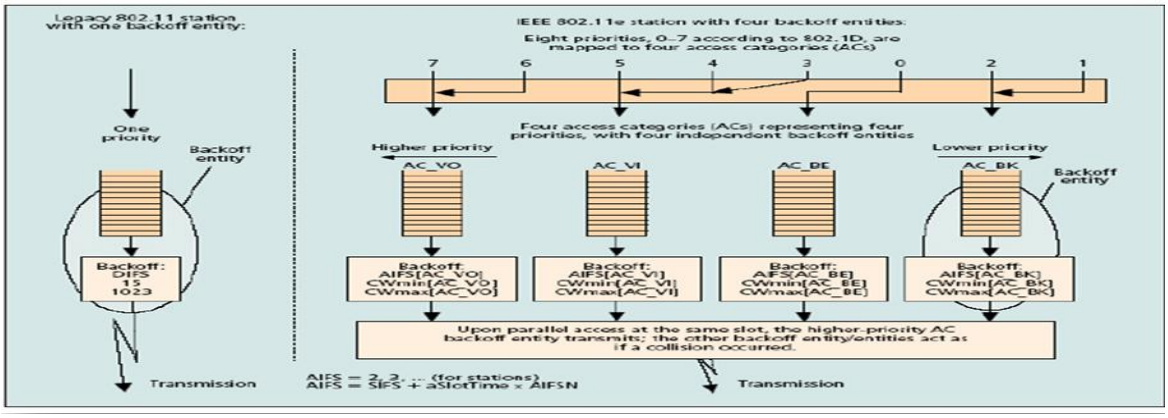

Figure 3.2 : Une station implémentant IEEE802.11 e

Les paramètres décrits ci-dessus sont annoncés par le point d'accès AP à travers des trames balises. Ce dernier peut alors les adapter aux conditions du réseau. La figure 3.2 illustre le mécanisme d'accès au support en mode EDCA.

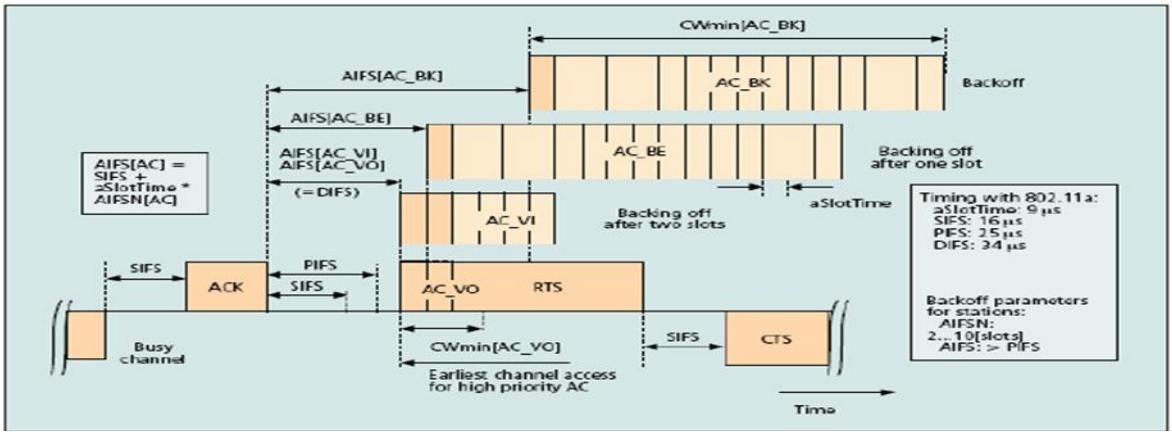

Figure 3.3 : L'accès en mode EDCA

Pour sa version actuelle, la norme 802.11e a aussi introduit le paramètre TXOP. Il s'agit d'un intervalle de temps pendant lequel une station a le droit d'émettre. Au niveau de la trame balise, l'AP annonce aussi à chaque AC la limite de l'intervalle TXOP (TXOPLimit [AC]) tout en définissant aussi la date de début de transmission. Durant un TXOP, la station peut transmettre plusieurs MPDUs pour un seul AC. Ces MPDUs sont espacés d'un SIFS de leurs acquittements. Cette transmission de plusieurs MPDUs est notée CFB. La figure 3.3 présente la structure du CFB :

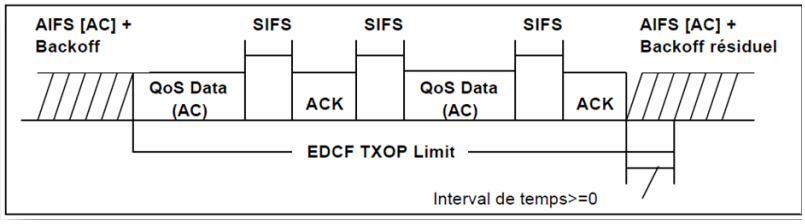

Figure 3. 4 Structure temporelle du CFB

- **La méthode d'accès HCCA:**

Le mécanisme d'accès HCCA combine les avantages des modes DCF et PCF. Il utilise un coordinateur central appelé HC qui utilise des règles différentes de celles du mode PCF. Avec HCCA, le TXOP est alloué par l'AP et peut-être actif à la fois dans la période sans contention (CFP) mais aussi dans la période avec contention (CP). En effet, il est possible de découper l'intervalle de temps CP en une nouvelle période sans contention appelée CAP qui utilise le mécanisme HCCA, et une période avec contention qui utilise EDCA. Les périodes CAP sont utiles pour rendre indépendante la fréquence d'émission des balises (beacons) des contraintes de latence que peuvent avoir les applications multimédias. D'autre part, pour remédier au phénomène de désynchronisation des beacons qui se produit avec le mode PCF avec HCCA, une station n'est autorisée à émettre un paquet que dans la mesure où sa transmission ne gène pas l'émission de la prochaine balise. Afin de garantir un service différencié, le mécanisme HCCA se base sur une négociation de trafic TSPEC entre le point d'accès et les stations. Avant de transmettre un flot qui nécessite une garantie de service, un circuit virtuel appelé TS doit s'établir entre l'AP et les différentes stations pour échanger certains paramètres (comme le débit du flot, la taille des paquets, la latence maximale acceptable, etc.) En fonction des paramètres TSPEC, un ordonnanceur localisé dans l'AP calcule une durée de TXOP pour chacune des stations. [8]

ii. Autres améliorations :

Le standard présente différents mécanismes, complémentaires à HCF, permettant d'offrir une QoS pour l'accès 802.11. L'essentiel de ces mécanismes fut introduit par la modification IEEE 802.11e. Nous en exposons certains dans ce paragraphe, celui qui nous intéresse étant essentiellement le protocole de lien direct.

- **Direct Link Protocol :**

Les spécifications de trafic dans le standard 802.11 original en mode AP ne permettent l'écoulement du trafic entre stations qu'en passant par l'AP uniquement.

Le protocole de liaison directe (DLP) dans la norme 802.11 e donne la possibilité aux stations d'envoyer le trafic directement entre elles sans traverser l'AP. Cette possibilité peut potentiellement augmenter la largeur de bande disponible pour la communication de station à station. Le DLP fonctionnera seulement quand les stations qui veulent communiquer sont dans la portée l'une de l'autre. Le DLP pourrait également augmenter potentiellement le temps d'exécution dans le cas où le lien entre les stations

qui communiquent est meilleur que le lien entre les stations et l'AP. Ceci a pu être le cas quand les stations sont plus près l'une de l'autre que de l'AP. Si après la durée « *DLPIdleTimeout* » il n'y a aucune transmission de trames entre les deux stations, le lien direct est coupé. [10]

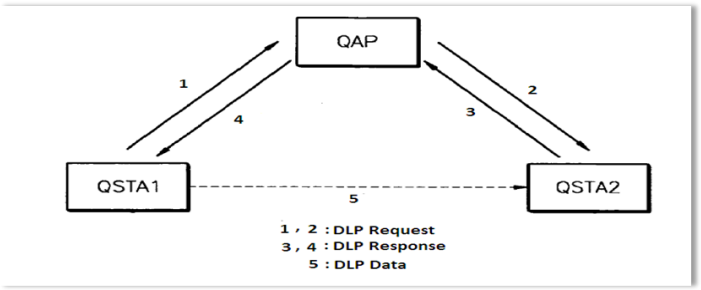

Figure 3.5 : Dialogue DLP

Avec le DLP, l'expéditeur envoie d'abord un message de demande de lien direct (DLP Request) au récepteur par le QAP. Une fois que le récepteur reconnait la demande, le lien direct entre les deux stations est établi.

Plusieurs formats de trames sont définis dans le but de la gestion du DLP citons : [11]

- **DLP Request :**

La trame DLP Request est utilisée pour l'établissement du lien direct entre deux stations dans un même BSS. Le corps de la trame DLP Request contient les informations mentionnées dans le tableau suivant :

Order	Information
1	Category
2	Action
3	Destination MAC Address
4	Source MAC Address
5	Capability Information
6	DLP Timeout Value
7	Supported rates

Tableau 3.1: Corps de la trame DLP Request

- **Category :**

Le tableau suivant représente les codes des catégories ainsi avec la signification :

Code	Signification
1	QOS
2	DLP
3	Block ACK

Tableau 3.2 : Les codes du champ « Category »

Dans la trame DLP Request la valeur du « Category » vaut **2** (représentant le DLP).

- **Action :**

Les différentes valeurs du champ « Action » avec leurs significations sont présentées dans le tableau suivant :

Code	Signification
0	DLP Request
1	DLP Response
2	DLP Teardown

Tableau 3.3 : Les codes du champ « Action »

Dans la trame DLP Request la valeur du champ « Action » vaut **0** (représentant le DLP Request).

- **Destination MAC Adress :**

Représente l'adresse MAC de la station destination.

- **Source MAC Adress :**

Représente l'adresse MAC de la station émettrice.

- **Capability information :**

Informations sur la capacité de la station émettrice de la demande.

- **DLP Timeout Value :**

Champ utilisé pour indiquer la valeur du temps de mort du lien direct, la longueur de cette valeur est 2 octets. Ce champ contient la durée en seconde après laquelle le lien direct est terminé s'il n'y a aucune trame échangée entre les deux QSTAs.

- **Supported rates :**

Contient les informations de taux de charge de la station émettrice.

- **DLP Response :**

La trame DLP Response est envoyée comme réponse à une trame DLP Request. Le corps de la trame DLP Response contient les informations mentionnées dans le tableau suivant :

Order	Information
1	Category
2	Action
3	Status Code
4	Destination MAC Address
5	Source MAC Address
6	Capability Information
7	Supported rates

Tableau 3.4 : Corps de la trame DLP Request

- **Category :**

Dans la trame DLP Response la valeur du « Category » vaut **2** (représentant le DLP).

- **Action :**

Dans la trame DLP Response la valeur du champ « Action » vaut **1** (représentant le DLP Response).

- **Status code :**

Les différentes valeurs du champ « Status » avec leurs significations sont présentées dans le tableau suivant :

Code	Signification
0	Établissement du lien direct avec Succès.
32	Échec non précisée.
33	Association (avec le QBSS) refusée car le QAP n'a pas une bande passante suffisante pour traiter une autre QSTA.
34	Association (avec le QBSS) refusée du a un taux de perte de trame excessif
35	Association (avec le QBSS) refusée car la station demandée ne supporte pas la QOS.
37	La demande a été refusée
38	La demande n'a pas été couronnée de succès, un ou plusieurs paramètres ont des valeurs invalides.
39	Le TS n'a pas été créé car la demande ne peut pas être honorée. Cependant une TSPEC suggéré est prévu pour que la QSTA source puisse tenter d'établir un autre TS avec les modifications proposées à la TSPEC.
40	Le TS n'a pas été créé car la demande ne peut pas être honorée.

	Cependant le HC peut être en mesure de créer un TS en réponse à une demande après le temps indiqué dans l'élément de retard TS.
41	Le lien direct n'est pas autorisé dans ce BSS
42	La station destination n'est pas présente dans le même QBSS
43	La station destination n'est pas une QSTA

Tableau 3.5 : Les codes du champ « Status »

- **Destination MAC Adress and the source Mac Adress:**

Elles sont copiées du champ correspondant dans la trame DLP Request.

- **Capability information :**

Informations sur la capacité de la station destination. Cette information est incluse seulement dans le cas où la valeur du « DLP Status Code » vaut **0** (Succès).

- **Supported rates :**

Contient les informations de taux de charge de la station destination. Cette information est incluse seulement dans le cas où la valeur du « DLP Status Code » vaut **0** (Succès).

- **DLP Teardown :**

La trame DLP Teardown est envoyée pour terminer le lien direct avec. Le corps de la trame DLP Teardown contient les informations mentionnées dans le tableau suivant :

Order	Information
1	Category
2	Action
3	Destination MAC Address
4	Source MAC Address

Tableau 3.6 : Corps de la trame DLP Teardown

- **Category :**

Dans la trame DLP Teardown la valeur du « Category » vaut **2** (représentant le DLP).

- **Action :**

Dans la trame DLP Teardown la valeur du champ « Action » vaut **2** (représentant le DLP Teardown).

- **Destination MAC Adress :**

Représente l'adresse MAC de la station destination.

- **Source MAC Adress :**

Représente l'adresse MAC de la station émettrice.

- **Block ACK (acquittement groupé):**

Cette procédure optionnelle permet d'améliorer l'utilisation du médium. En effet, elle permet à une station d'envoyer plusieurs paquets sans que ceux-là soient acquittés individuellement. Le bloc de paquets pourra être acquitté à la fin de l'envoi du bloc ou dans un TXOP ultérieur. L'utilisation du réseau s'en trouve ainsi améliorée. [9]

- **Le contrôle d'admission**

Un cadre pour le contrôle d'admission a été mis en place par la modification 802.11e. Ce cadre concerne l'accès par HCF avec ou sans contention (par EDCA ou par HCCA). Le contrôle d'admission servira à la gestion et la régulation de la bande passante disponible. Une QSTA souhaitant avoir des garanties de QoS (sur les délais d'accès, sur les débits ou sur le taux de pertes par exemple) devra passer par le contrôle d'admission. Les algorithmes de contrôle d'admission ne sont pas définis par le standard, le choix de l'algorithme utilisé est laissé à l'équipementier. Le standard définit cependant un cadre et un certain nombre de règles que les algorithmes devront respecter. [9]

- **NoAck**

Permet la mise en place de classes de services avec lesquels les messages transmis ne sont pas acquittés. Cette amélioration permet d'éviter la retransmission inutile de données à haute criticité temporelle. [6]

- **Respect des échéances**

D'autres modifications ont été introduites par IEEE 802.11e permettant d'améliorer le respect des échéances en contraignant les durées d'accès des stations :

- Les stations utilisant le médium sont contraintes de respecter le TBTT annoncé par le paquet Beacon. Une station voulant accéder au médium doit vérifier que la transmission entamée (jusqu'à la réception éventuelle du ACK) ne doit pas dépasser le TBTT annoncé. Le CFP ne sera, par conséquent, pas retardé par les stations accédant au médium.
- Chaque accès au médium se fait dans la limite de l'opportunité de transmission (TXOP) accordée. La valeur du TXOP est fixée par le HC. [9]

III.3.2 Les mécanismes de qualité de service niveau IP : IntServ / DiffServ

i. Le protocole à intégration de service IntServ :

Les applications traditionnelles non temps réels comme FTP se sont longtemps satisfaites du service best effort. Mais avec l'arrivée des communications multimédias, de nombreuses applications sont devenues sensibles au délai si bien que le service best effort traditionnel ne suffit plus. Bien que certaines applications soient adaptatives, il est souvent nécessaire de fournir de nouvelles classes de service offrant une meilleure qualité de service (en termes de bande passante, délai ou pertes). Ces nouvelles classes de service s'ajoutent au best effort traditionnel pour créer un Internet à intégration de services.

Un mécanisme explicite est utilisé pour signaler les exigences de qualité de service par flot aux éléments du réseau (hôtes, routeurs ou sous-réseaux). Les éléments du réseau, selon les ressources disponibles, implémentent l'un des services IntServ en fonction du type de qualité de service souhaité pendant la transmission des données. Le modèle distingue plusieurs types de services, en fonction du délai de transit par paquet **(**Service à contrôle de charge (CL**)**, Service garanti (GS)**).**

L'architecture IntServ repose sur deux principes fondamentaux :

- le réseau doit être contrôlé et soumis à des mécanismes de contrôle d'admission,
- des mécanismes de réservation de ressources sont nécessaires pour fournir des services différenciés.

Le modèle IntServ définit une architecture capable de prendre en charge la qualité de service en définissant des mécanismes de contrôle complémentaires sans toucher au fonctionnement IP. C'est un modèle basé sur un protocole de signalisation RSVP. Dans le modèle présenté par, les routeurs réservent les ressources pour un flot de données spécifiques en mémorisant des informations d'état. Il est important de rafraîchir périodiquement les informations au cas où il y a eu changement de la route emprunté par le flot. En effet, il est inutile de continuer à réserver les ressources sur un routeur qui ne fait plus partie du chemin emprunté. Au niveau technique, la faiblesse principale de l'architecture IntServ est sa non-résistance au facteur d'échelle. Le nombre de flux qui peuvent bénéficier d'une réservation est assez limité, en particulier dans les routeurs du cœur du réseau. Ces équipements doivent traiter des milliers des flux simultanément, et le coût introduit par la gestion d'états et l'ordonnancement par flux peut entraîner une réduction considérable de leur performance. **[**7]

ii. Le protocole à différentiation de service DiffServ :

L'approche DiffServ est souvent comparée à celle de IntServ pour sa capacité à être déployée sur de « grands réseaux » ; alors que IntServ, de par le traitement par flots, ne peut s'appliquer que sur des réseaux de « petite taille », DiffServ réduit au maximum la taille des tables en considérant des agrégations de flots. Ainsi, les flots ne sont pas trait´es individuellement mais par agrégats, ce qui allège considérablement la charge des routeurs du réseau. De plus, le contrôle d'admission n'est plus assuré individuellement par chaque routeur travers´e, mais par les routeurs de bordures, rendant DiffServ beaucoup plus adapté aux grands réseaux, et notamment aux réseaux d'opérateurs.

Le niveau de QoS d'un flot est indiqué dans un champ de l'en-tête de ses paquets. Lorsqu'un routeur de bordure décide d'admettre un nouveau trafic dans le réseau, il fixe la valeur du champ DSCP dans l'en-tête IP :

• le code 0 signifie que le paquet doit être trait´e en Best-Effort, après tous les autres (niveau le moins prioritaire),

• un code autre que 0 aura une autre signification ; cette correspondance est fixée par l'opérateur lui-même, selon les contrats qu'il propose à ses clients.

Lorsqu'un routeur du cœur du réseau devra traiter ce paquet, il devra inspecter le champ DSCP et traiter le paquet en conséquence. Ainsi, DiffServ présente l'avantage de ne pas nécessiter de signalisation puisque le décodage du champ DSCP se fait via des tables inscrites dans la mémoire du routeur. Le contrôle de congestion se fait directement par les routeurs de bordure, selon le principe de la capacité finie et connue du réseau. Dans le cœur du réseau, il n'y a pas de réservation de ressources, seule la différentiation de traitement des paquets suffit à appliquer une qualité de service pour la traversée du paquet dans le réseau. Cependant, afin de palier les pertes sur le médium, le réseau cœur, par exemple un réseau ATM, nécessite d'être légèrement surdimensionné. [6]

IV. Conclusion :

Dans ce chapitre, nous avons défini la problématique de la qualité de service dans les réseaux sans fil. Nous avons ensuite présenté un ensemble de solutions de qualité de service apportées à ces technologies. Les améliorations apportées à la technologie 802.11 ont été détaillées.

I Introduction :

Les réseaux informatiques connaissent une expansion importante grâce à plusieurs moyens qui ont pu se développer au cours du temps, donc il est très coûteux de déployer un banc d'essai complet contenant plusieurs ordinateurs, des routeurs et des liaisons de données pour valider et vérifier un protocole de réseau ou un certain algorithme spécifique. C'est pour cela les simulateurs de réseaux viennent pour pallier à ce problème.

Les simulateurs du réseau offrent beaucoup d'économie de temps et d'argent pour l'accomplissement des tâches et sont également utilisés pour que les concepteurs des réseaux puissent tester les nouveaux protocoles ou modifier les protocoles déjà existants d'une manière contrôlée et productrice.

La problématique étudiée dans ce chapitre étant la simulation des réseaux sans fil, et en particulier le wifi, la méthode de simulation à événements discrets s'avère la plus adéquate pour notre cas. En effet, il est facile de modéliser un réseau sans fil sous la forme d'entités (les nœuds sans fil) et de modéliser les interactions entre elles aux moyens d'événements, comme l'événement de "*transmission d'un paquet*" ou "*réception d'un paquet*". Nous présentons dans ce qui suit le déroulement des étapes de simulation à événements discrets, que nous avons menée dans ce travail de fin d'études et consistant à simuler l'impact du Direct Link (mécanisme de QoS introduit par le standard 802.11^{e} et qui n'a jamais été simulé auparavant dans les différents modules de simulation du WIFI avec QoS présents dans l'état de l'art). D'où notre intérêt pour évaluer ce mécanisme sur les performances globales du réseau et donc ceci permettra de savoir s'il est intéressant d'implémenter ce mécanisme dans les AP par les constructeurs ou pas, dans la mesure ou leur implémentation apporterait un réel bénéfice dans l'amélioration de la QoS des applications.

II Simulation

Simuler c'est modéliser un système complexe, afin de prévoir son comportement dans le monde réel. Il s'agit d'une approche permettant de représenter le fonctionnement d'un système réel constitué de plusieurs entités, de modéliser les différentes interactions entre elles, et enfin d'évaluer le comportement global du système et son évolution dans le temps.

Le recours à la simulation permet de contourner les limites de la complexité des modèles analytiques. Toutefois, il est nécessaire de bien identifier les caractéristiques du système afin de la représenter, le plus finement possible, par des modèles abstraits.

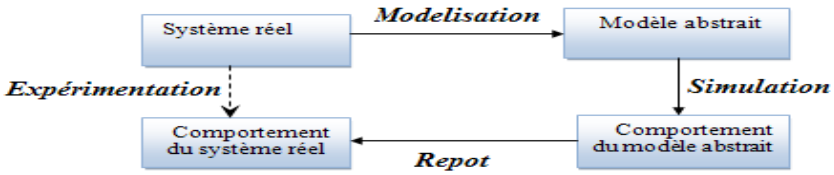

Figure 4.1 : Cycle modélisation-simulation

Si la représentation du système réel par des modèles abstraits est suffisamment réaliste et précise, il est alors possible de reporter les résultats obtenus avec ces modèles sur le système réel. Le cycle correspondant aux étapes de modélisation, simulation et report des résultats est illustré dans la figure 4.1. [25]

III Choix du simulateur

Les simulateurs réseaux sont utilisés par des personnes de différents domaines tels que les chercheurs universitaires, industriels et, d'assurance de qualité (AQ) pour concevoir, simuler, vérifier et analyser les performances des protocoles de différents réseaux. Ils peuvent également être utilisés pour évaluer l'effet des différents paramètres des protocoles étudié. En général, un simulateur de réseau est composé d'un large éventail de technologies et de protocoles réseaux et aide les utilisateurs à construire des réseaux complexes à partir de blocs de construction de base comme des grappes de nœuds et de liens. Avec leur aide, nous pouvons concevoir différentes topologies de réseau en utilisant différents types de nœuds tels que les nœuds terminaux, concentrateurs, ponts de réseau, routeurs, des dispositifs optiques de couche de liaison, et des unités mobiles.

Il existe plusieurs simulateurs réseaux tel que les simulateurs NS-2 et NS-3, OPNET.

Le simulateur NS-2 a été un simulateur populaire pour la recherche et l'éducation sur les systèmes Internet, dont notamment mobiles, les systèmes sans fil. NS-2 bénéficie d'utilisation répandue dans le milieu de la recherche, le code de simulation a été contribué par plus de cent personnes et organisations, et l'utilisation du simulateur est toujours référencé dans de nombreux travaux de recherche en réseau.

Cependant, une lacune majeure de NS-2 est son évolutivité en termes d'utilisation de la mémoire et du temps d'exécution de la simulation. Ceci est particulièrement un problème en ce qui concerne les nouveaux domaines de recherche dans les réseaux

informatiques, tels que les réseaux de capteurs sans fil, les réseaux peer-to-peer ou des architectures maillés qui exigent une simulation de réseaux très larges.

Outre NS-2, plus d'une douzaine de simulateurs des réseaux sont actuellement utilisés dans les universités et l'industrie. Parmi les simulateurs les plus connus nous choisissons le simulateur NS-3 pour réaliser notre travail.

Des travaux de comparaison entre NS-2 et NS-3 ont été réalisés et ont montré que NS-3 est meilleur que NS-2 de plusieurs façons, notamment:

- Un noyau logiciel remanié pour améliorer l'évolutivité et l'extensibilité, y compris le soutien pour les simulations distribuées.
- Une architecture pour soutenir la création de logiciels réseaux open source tels que les machines virtuelles, les piles de protocoles, les démons de routages et de paquets analyseurs de traces.
- La mise à disposition de nouveaux modèles sans fil pour IEEE 802.11, et éventuellement d'autres modèles tels qu'IEEE802.16.
- Une capacité de réseau réorganise l'émulation,
- Une version révisée de recherche et de collecte de statistiques.

En outre, NS-3 regroupe un grand nombre de mécanismes fondés sur le succès et évite les échecs de NS-2. [26], [27]

IV Présentation du Simulateur NS3

NS-3 est conçu pour remplacer le NS-2 courant populaire. Toutefois, NS-3 n'est pas une version mise à jour de NS2. NS-3 est un nouveau simulateur et il n'est pas rétro-compatible avec NS-2.

NS-3 est un simulateur réseau à évènements discrets. Il vise à remplacer son prédécesseur NS-2, écrit en C++ et OTcl (version orientée objet de Tcl), pour tenter de remédier à ses limites (mauvaise gestion des traces ou encore, plus gênant l'utilisation de multiples interfaces sur un noeud…). NS-3 est écrit en C++ et Python, et peut être utilisé sur les plateformes Linux/Unix, OS X (Mac) et Windows.

Son développement a d'abord commencé en Juillet 2006, et devait durer quatre ans, Il est financé par les instituts comme l'Université de Washington, Georgia Institute of Technology et le Centre de l'ICSI pour la recherche sur Internet, la première version majeure publique et stable a été publiée en juin 2008.

Les développeurs de NS-3 ont décidé que l'architecture de simulation devait être remaniée complètement en partant du zéro. Dans cette optique, l'expérience tirée de

NS-2 doit être associé avec les progrès des langages de programmation et du génie logiciel. L'idée de la rétrocompatibilité avec NS-2 a été abandonné dés le départ. Cela libère NS-3 de contraintes héritées de NS-2 et permet la construction d'un simulateur qui est bien conçu depuis le début. [27], [28], [29].

V Terminologie et abstractions

Il est important de bien comprendre le sens des termes employés au sein du simulateur, ainsi que les abstractions qui ont été faites.

NS-3 utilise des termes largement employés dans le domaine des réseaux, mais qui peuvent avoir une signification particulière au sein du simulateur. Voici les principaux :

- **Un nœud *Node :***

Représente tout élément de réseau. La composition d'un Node peut être gérée (ajout de composants, de protocoles, d'applications).

- **Une application *Application* :**

Représente un code exécuté par un utilisateur. Ce code peut être nécessaire au déroulement d'une simulation. L'échange de paquets durant une simulation nécessite par exemple la description d'une Application au sein des nœuds participants (par exemple, *UdpEchoClientApplication* d'un côté et *UdpEchoServerApplication* de l'autre pour réaliser une application en mode client/serveur). NS-3 ne fait pas de distinction entre les "applications système" (souvent exécutées par le noyau) et les applications des utilisateurs (exécutées dans le user-space). Les applications peuvent ensuite être attachées à un Node.

- **Un canal de communication *Channel***

Représente le lien qui relie des nœuds, ou plus exactement les NetDevices installés dans les nœuds. Des spécialisations de cette classe sont définies, comme par exemple *CsmaChannel* pour modéliser un lien Ethernet utilisant CSMA, ou encore *WifiChannel* pour modéliser un lien WiFi.

- **Une interface de communication, ou interface réseau :**

Appelée **NetDevice**, qui modélise à la fois l'équipement (carte réseau) et le pilote dont un ordinateur a besoin pour pouvoir communiquer avec d'autres. Des spécialisations de NetDevice sont fournies, comme par exemple *CsmaNetDevice* qui simule une carte réseau Ethernet et peut être reliée à un *CsmaChannel*, ou encore **WifiNetDevice** pour un lien à un canal de type *WifiChannel*.

Pour établir une connexion entre deux nœuds, il faut alors :

- Équiper chaque nœud de *NetDevice.*
- Configurer la couche protocolaire sur chaque nœud (élément non représenté qui se situe entre les applications et les NetDevice).
- Dans le cas d'une couche protocolaire TCP/IP, configurer les adresses MAC et IP sur chaque NetDevice.
- Créer le canal de communication correspondant.
- Connecter chaque NetDevice au canal de communication.

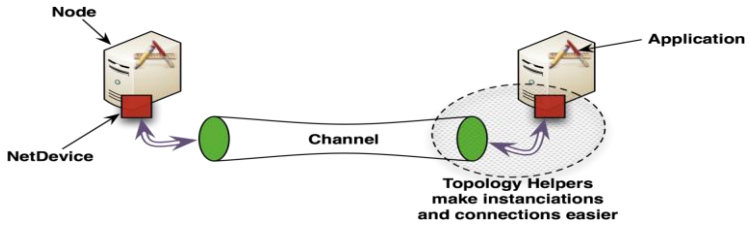

Figure 4.2 : Établissement d'une connexion entre deux nœuds

S'il s'agit de connecter un grand nombre de nœuds les uns avec les autres, ce processus peut être très lourd. NS-3 fournit des **TopologyHelpers** pour faciliter ce genre de tâches. Les classes représentant ces objets sont dans le dossier *NS-3.10/src/helper*. On y trouve par exemple :

- un *CsmaHelper* pour instancier des *CsmaNetDevice* sur un nœud, instancier un *CsmaChannel* et les connecter.
- un *WifiHelper*, qui a la même fonction que le CsmaHelper, mais pour une interface et un canal qui utilisent le *WiFi.*
- un I*nternetStackHelper* pour instancier au sein d'un nœud la couche protocolaire *IP/TCP/UDP* et des fonctions de routage (*IPv4/IPv6*). [30]

VI Modules du simulateur NS3 :

NS-3 fournit différents modules qui peuvent être modifiés et effectivement utilisés. L'organisation du logiciel de NS-3 est illustrée dans la figure 4.4. Voici les modules de base de NS-3 :

- ***Core Module:***

Module de base, il est indépendant, il permet la création d'une structure de classes hiérarchiques dérivant de la classe Object de base. Cette hiérarchie possède plusieurs fonctionnalités conçus spécialement pour répondre aux besoins de la simulation.

Parmi lesquels : l'agrégation d'objets, l'enregistrement actif(TypeId) avec les attributs publiques…etc.

- ***Common Module :***

Ce module gère les actions liées à la génération et la réception des paquets, ce module est centré sur la *classe Packet* utilisé pour la simulation de réseau au niveau packet. L'utilisation de cette dernière classe est conçu de façon à optimiser la gestion de la mémoire en utilisant la méthode (*copy on write*) et en donnant la possibilité de manipuler des paquets vides et dont la taille est le paramètre suffisant et nécessaire pour la simulation. La plupart des autres modules ont utilisé les fonctionnalités de ce module.

- ***Simulator Module*** :

Les Simulations des événements sont gérées par le module de simulation. Il prévoit explicitement la possibilité de planifier des événements à des moments différents et ensuite d'exécuter ces événements. La classe *Time* représente le temps simulé à haute résolution en utilisant un entier de 128 bits. Cette classe est la classe la plus importante du simulateur.

- ***Mobility Module :***

Ce module permet de définir la position des nœuds et associe un modèle de mouvement aux agents de la simulation. NS-3 fournit sept modèles de mobilité.

- ***Node Module***

La classe Node peut contenir plusieurs *NetDevices*. Chaque *NetDevice* est attaché à un *channel* par lequel il envoie et reçoit des paquets.

Un nœud peut contenir plusieurs gestionnaires de protocole, qui acceptent les paquets reçus par le NetDevice. Pour lancer la transmission des paquets, chaque nœud peut également contenir une liste d'applications.

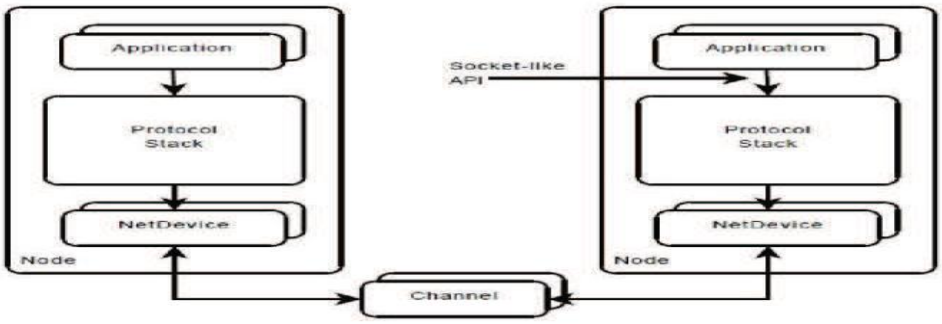

Figure 4.3 : Architecture du nœud NS-3

- ***Helper Module :***

Ce modèle peut être considéré comme un emballage de haut niveau. Il facilite la construction des scénarios complexes de simulation et l'installation des différents modules dans des agents différents.

- ***Application Module :***

Certaines applications sont intégrées et fournies par NS-3. Elles sont installées dans les nœuds et peuvent être démarrées/arrêtées à des moments précis dans la simulation.

- ***InternetStack Module :***

Les classes de ce module définissent les protocoles TCP/IP des couches réseaux trois et quatre (TCP/IP).

- ***Devices Module :***

Les composants de ce type représentent des périphériques réseaux et transmettent des paquets via un canal virtuel à d'autres instances de la même classe *NetDevice*.

- ***Routing Module :***

Deux algorithmes de routage sont disponibles dans NS-3. Le premier appelé *GlobalRouter*, utilise des routes statiques et le deuxième met en œuvre le protocole *OLSR* pour les réseaux dynamiques ad-hoc.

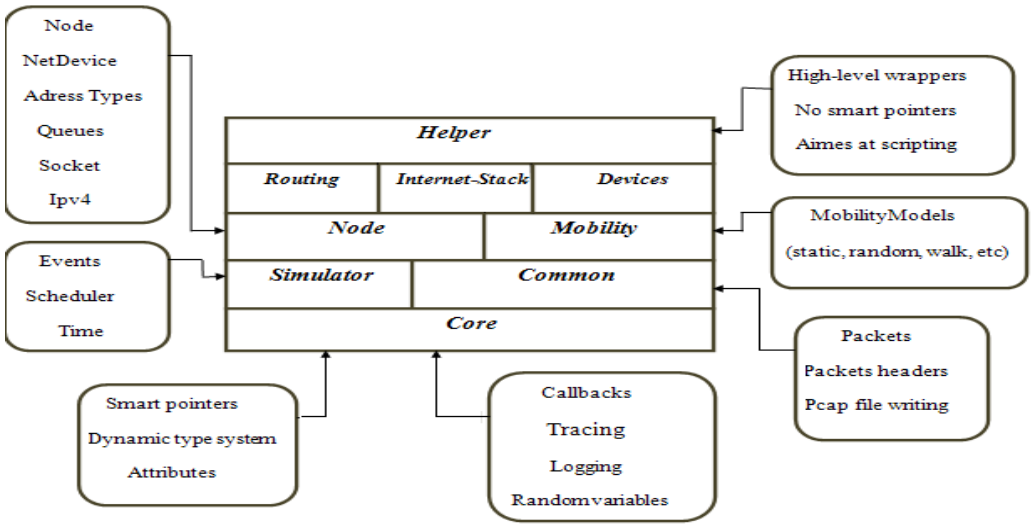

Figure 4.4:Les principaux modules de NS3

VII Modèles et mise en œuvre de réseaux Wifi dans ns-3

Pour mettre en œuvre des réseaux Wifi (802.11) dans NS-3, tout doit être décrit, des couches les plus basses (canal de communication) aux éléments à mettre en œuvre dans les nœuds (interfaces, couches physique et MAC qu'elles utilisent).

Nous avons détaillé quelques classes de base et les modèles fournis dans NS-3.

VII.1 Classes de base :

- **WifiNetDevice :**

Tout d'abord, un nœud souhaitant communiquer en Wifi doit disposer d'une **WifiNetDevice.** Cette classe, qui hérite de la classe générique **NetDevice**, est définie dans le fichier */src/devices/wifi/wifi-net-device.h* et implémentée dans le fichier /src/devices/wifi/wifi-net-device.cc. En plus des éléments caractérisant une **NetDevice** (notamment le nœud auquel elle est attachée), elle est caractérisée par :

- Un modèle de la couche physique **WifiPhy**
- Un modèle de la couche MAC **WifiMac**
- Un **WifiRemoteStationManager**, qui maintient une liste des stations connectées au réseau sans fil, des informations sur leur état, et qui incarne un algorithme d'adaptation du débit.

- **WifiChannel :**

Cette **WifiNetDevice** doit être connectée à un **WifiChannel**. La classe **YansWifiChannel** est la seule implémentation d'un modèle de canal wifi. Elle est caractérisée par :

- Une liste contenant des pointeurs vers les modèles de couche physique de chaque nœud connecté, qui doivent être du type **YansWifiPhy.**
- Un modèle de **perte/atténuation** lors de la propagation.
- Un modèle de **délai de propagation**.

La classe **WifiChannel** peut être utilisé pour relier un ensemble d'interfaces réseau **WifiNetDevice**. La classe **WifiPhy** est la partie dans le **WifiNetDevice** qui reçoit les bits du canal. Le **WifiPhy** modélise un canal 802.11 en termes de fréquences, de modulation, de débit binaire et interagit avec le **PropagationLossModel** et **PropagationDelayModel** trouvés dans le canal. La couche physique peut être dans l'un des trois états : **TX (**Transmission**), RX (**Reception**)** ou **IDLE (**neutre**).**

VII.2 Les modèles :

- **Attachés à une WifiNetDevice :**

i. Modèle de couche physique :

Le modèle de couche physique qu'utilise une **WifiNetDevice** conditionne sa capacité à envoyer ou recevoir des signaux et la façon dont elle le fait. Ce modèle a pour vocation de traiter des considérations telles que la puissance d'émission/réception

des interfaces, la bande de fréquence utilisée selon le standard 802.11 choisi, les interférences, etc.

ii. Modèle de couche MAC :

Le modèle de couche MAC gère l'accès au canal, les éléments d'identification du réseau sans fil (**SSID**), l'association/désassociassions des nœuds et leur adressage physique (adresse MAC). Le modèle de couche MAC est décrit dans NS-3 par la classe **WifiMac.** Le fichier */src/devices/wifi/wifi-mac.cc* contient l'implémentation d'un certain nombre de méthodes renvoyant des valeurs par défaut et de **callbacks** (**TraceSource** notamment), mais ne propose pas d'implémentation du modèle en soit. La classe **RegularWifiMac** est une implémentation générique de ce modèle de couche MAC qui traite de la majorité des fonctionnalités de cette couche.

Il existe actuellement six modèles Mac de haut niveau, trois pour le Mac sans gestion de la QoS et trois avec la gestion de QoS, dans chaque groupe (avec et sans QoS) on trouve les modèles suivants :

- **AdhocWifiMac :**

Surcharge quelques méthodes pour représenter le fonctionnement des nœuds en mode **AdHoc**.

- **ApWifiMac :**

Implémente les méthodes permettant l'envoi de beacons, l'association/désassociassions et l'authentification des stations auprès d'un point d'accès.

- **StaWifiMac :**

Implémente les méthodes qui permettent la réception de beacons émis par des points d'accès et la gestion de l'état d'une station en termes d'association.

Avec les modèles Mac QoS, il est possible de travailler avec un trafic appartenant à quatre classes d'accès différentes :

AC_VO	Pour le trafic voix
AC_VI	Pour le trafic vidéo
AC_BE	Pour le trafic best-effort
AC_BK	Pour le trafic d'arriere-plan

Figure 4.5 : Classe de trafic

Afin de déterminer la classe d'accès, chaque paquet provenant d'un Mac autre que le wifi et transmis vers un Mac Wifi doit être marqué au moyen d'un objet **QoSTag**

afin d'associer un **TID** (trafic id) pour ce paquet. Dans le cas où un **QoSTag** est *absent* le paquet sera considéré comme appartenant à la classe d'accès *AC_BE*.

- **Attachés au WifiChannel :**

L'implémentation fournie par la classe **YansWifiChannel** admet l'utilisation de modèles permettant de simuler la perte de puissance (atténuation) d'un signal et le délai de propagation sur le canal.

Nous avons détaillés les modèles proposés dans NS-3.

i. Modèle de perte/atténuation lors de la propagation :

Ce modèle permet de simuler la perte de puissance (atténuation) d'un signal évoluant sur un canal de transmission. Il permet en fait de calculer la puissance de réception d'un nœud destination, ce qui permet de déterminer s'il est susceptible de recevoir le signal. Ce calcul repose sur la puissance d'émission du nœud source et la position des nœuds source et destination (qui dépend d'un modèle de mobilité).

ii. Modèle de délai de propagation :

Ce modèle permet de simuler le délai de propagation sur le canal de transmission. Le calcul de ce délai repose sur la position des nœuds source et destination (qui dépend d'un modèle de **mobilité**).

VII.3 Les helpers :

Les helpers comme pour tout ce que l'on peut définir dans NS-3, les instanciations puis les configurations des éléments instanciés se font principalement en passant par les helpers fournis. Concernant la mise en œuvre de réseaux Wifi, tout commence par l'utilisation d'un premier helper : **Le WifiHelper.**

- **WifiHelper :**

Ce helper sert à la création et à l'installation de **WifiNetDevice pré-configurées** sur différents nœuds.

- **WifiPhyHelper :**

Ce helper a pour vocation de permettre la création et la configuration de la couche physique à mettre en œuvre dans les **WifiNetDevice**.

- **WifiMacHelper :**

Ce helper a pour vocation de permettre la création et la configuration de la couche MAC à mettre en œuvre dans **lesWifiNetDevice.** Seulement, aucune implémentation n'est directement fournie par cette classe. Par contre, deux classes en héritent et fournissent une implémentation :

i. La classe NqosWifiMacHelper :

Permet d'instancier un modèle de couche MAC simple, qui ne gère pas la qualité de service(QoS).

ii. La classe QosWifiMacHelper :

Permet d'instancier un modèle de couche MAC qui gère la qualité de service : définition de classes de service, gestion des files d'attentes associées etc. Il faut donc utiliser, au choix, un de ces deux helpers pour instancier et configurer un modèle de couche MAC.

Les constructeurs de ces deux classes n'instancient rien du tout. Une méthode **Default ()** définie dans chacun de ces helpers initialise à la fois le modèle de couche MAC à utiliser et le fait qu'il supporte ou non la QoS :

- ✓ **pour un NqosWifiMacHelper :**

Le modèle est initialisé à *AdhocWifiMac* et l'attribut **QosSupported** à *false*.

- ✓ **pour un QosWifiMacHelper :**

Le modèle est initialisé à *StaWifiMac* et l'attribut **QosSupported** à *true*.

Un des intérêts de NS-3, c'est de pouvoir faire aux nœuds un certain nombre de traitements, que ce soit pour tester des protocoles ou pour évaluer des approches à plus haut niveau. Ces traitements doivent être développés en utilisant le concept d'application. Nous avons optés pour les applications de type OnOffApplication dans nos expérimentations afin d'étudier et de tester le protocole de lien directe. [30], [32]

VIII Notre proposition

Dans notre travail nous avons utilisé une machine virtuelle sur laquelle nous avons installé le système linux fédora.

Nos simulations ont été faites sous NS3 qui possède un module qui permet de simuler le réseau Wifi avec une variété de mécanismes de qualités de service.

Nous avons ainsi utilisés le logiciel Wireshark pour l analyse de nos fichiers trace et le logiciel Matlab pour la représentation graphique de nos résultats.

Notre application est réalisée selon les deux scripts suivants :

- ***Le premier script***

Dans notre premier script et dans un premier cas nous avons créé un réseau wifi composé de deux stations et un point d'accès qui supportent la QoS avec un adressage aléatoire où la station de base du réseau n'est pas spécifiée. Donc dans ce cas là les stations vont communiquer via le point d'accès.

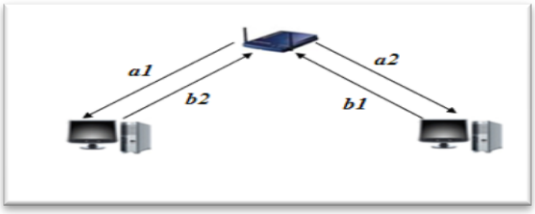

Figure 4.6 : Réseau wifi test

Dans le deuxième cas nous essayons d'augmenter le nombre des stations communiquant via le point d'accès jusqu'à se qu'on arrive à une saturation du point d'accès du réseau, ce qui nous permet de découvrir le nombre de stations supporté par ce dernier.

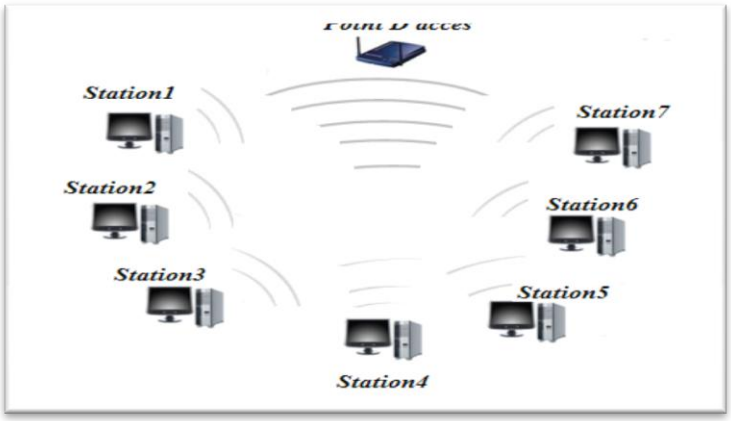

Figure 4.7 : Augmentation du nombre des stations

- ***Le deuxième script :***

Dans ce deuxième script nous avons créé une topologie réseau mixte contenant une infrastructure wifi, utilisant un AP relié à un réseau LAN Ethernet filaire.

Dans cette topologie la communication se fait entre le premier nœud du réseau infrastructure wifi et le dernier nœud du réseau LAN tout en passant par le point d'accès du réseau infrastructure wifi. Nous avons considéré que les stations sont dans le même BSS donc les stations vont communiquer directement, et le point d'accès n'interviendra que dans le cas d'une communication entre une station wifi et une autre station dans un autre réseau. (Dans notre cas c'est le réseau LAN).

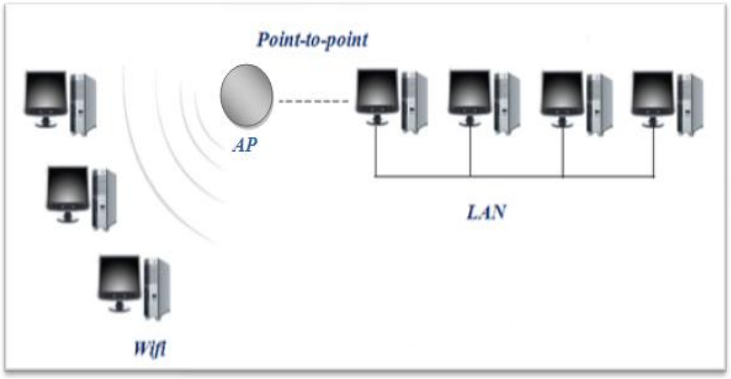

Figure 4.8 : Topologie réseau mixte test

Puis nous commençons à augmenter le nombre des stations du réseau wifi et nous ajoutons des applications entre elles.

Ensuite nous augmentons le nombre d'applications entre les stations du réseau wifi et les stations dans les autres réseaux et nous testons le nombre de clients que le point d'accès peut supporter. Si le nombre des stations trouvés dans le deuxième script est plus grand que celui du premier script alors le DLP apporte un avantage très intéressant dans le domaine du réseau, et cela prouve qu'il est un mécanisme de qualité de service très important.

IX Résultats de simulation

On a testé plusieurs paramètres de qos pour voir l'influence du mécanisme DLP sur ces dits paramètres comme suit :

- **Débit utile :**

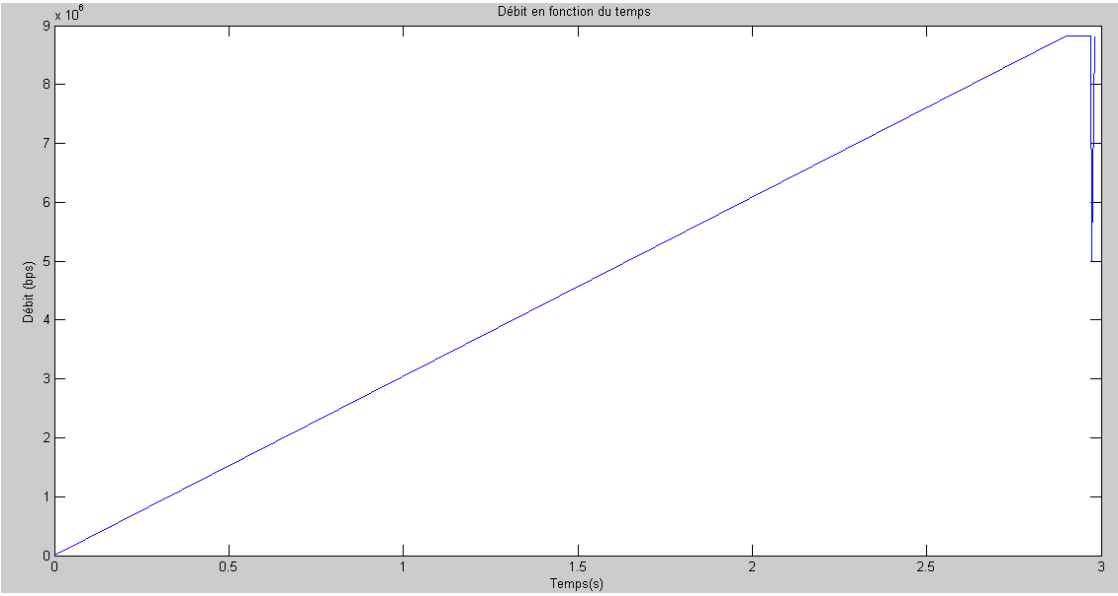

Figure 4.9 : Débit sans DLP

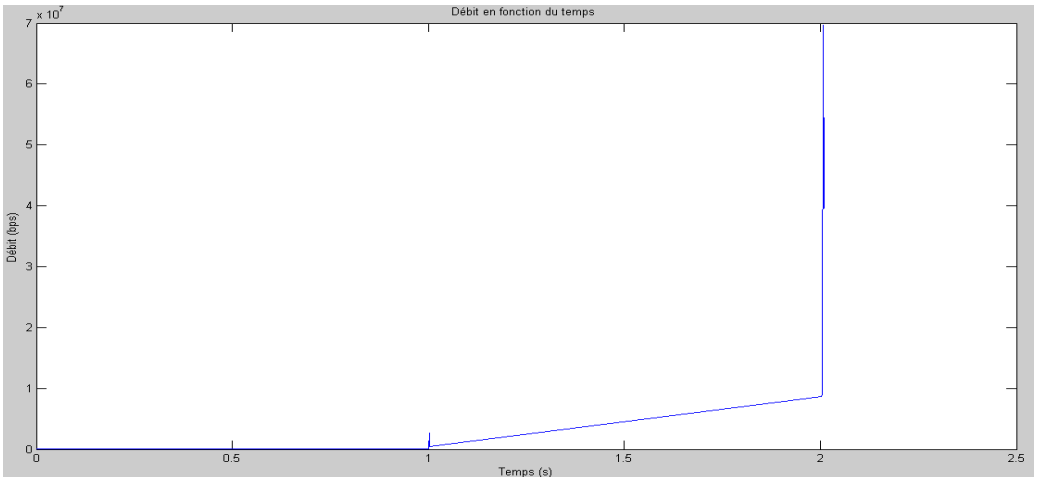

Figure 4.10 : débit avec DLP

L'analyse des deux courbes précédentes montre bien que le débit avec le mécanisme DLP est plus grand que dans le deuxième cas.

- **Délai :**

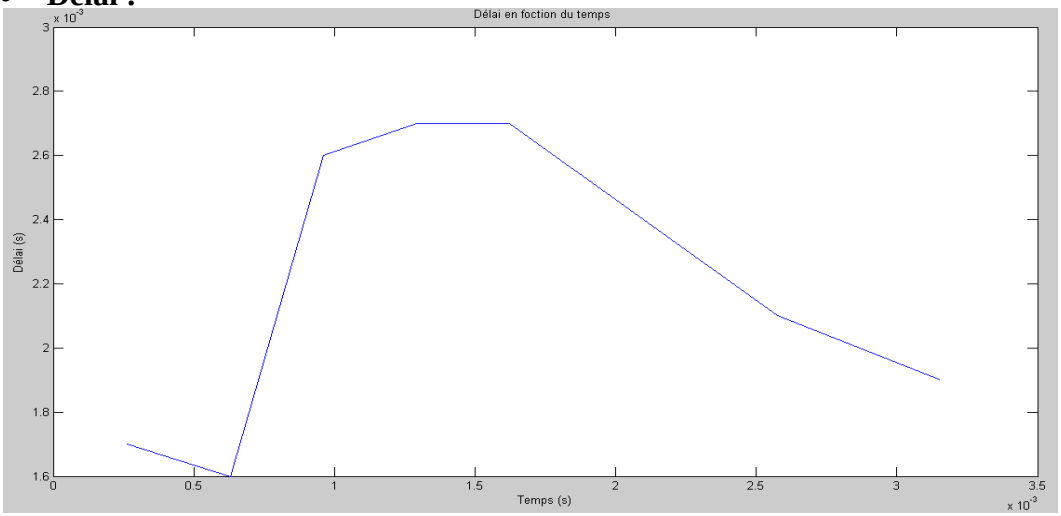

Figure 4.11: Délai sans DLP

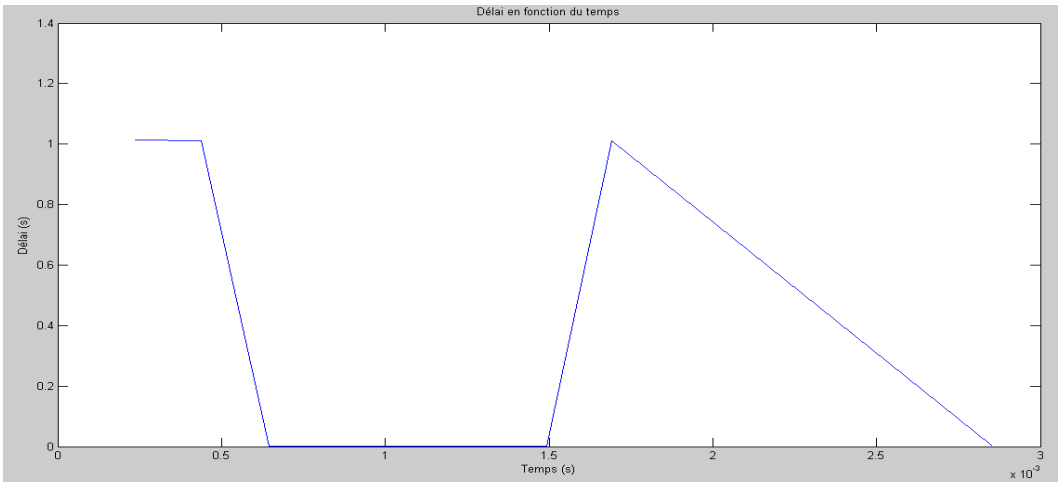

Figure 4.12: Délai avec DLP

L'analyse des deux courbes précédentes montre bien que le délai sans le mécanisme DLP est plus grand que dans le cas avec DLP.

- **Latence :**

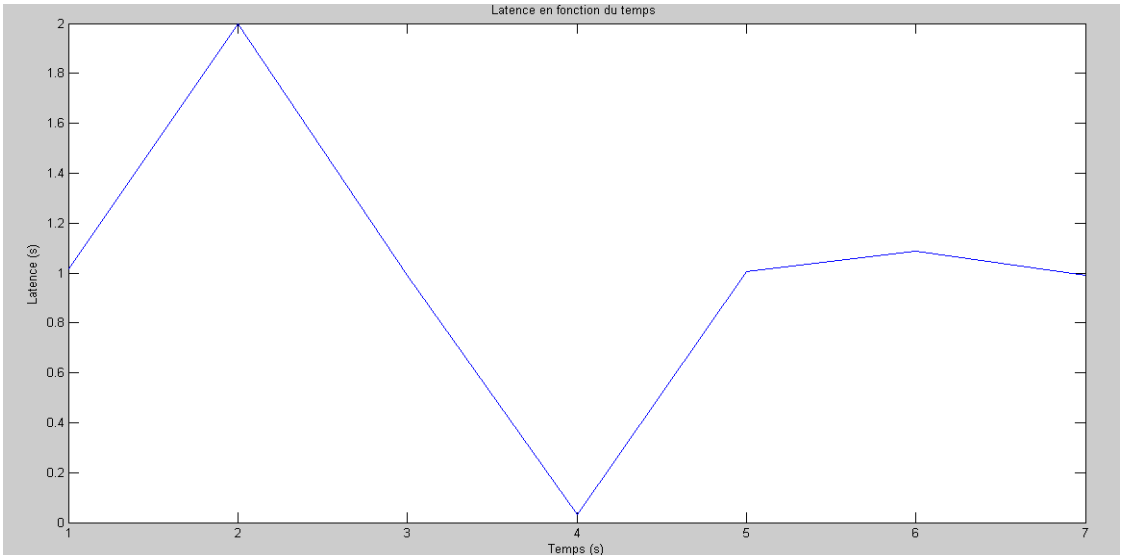

Figure 4.13 : Latence sans DLP

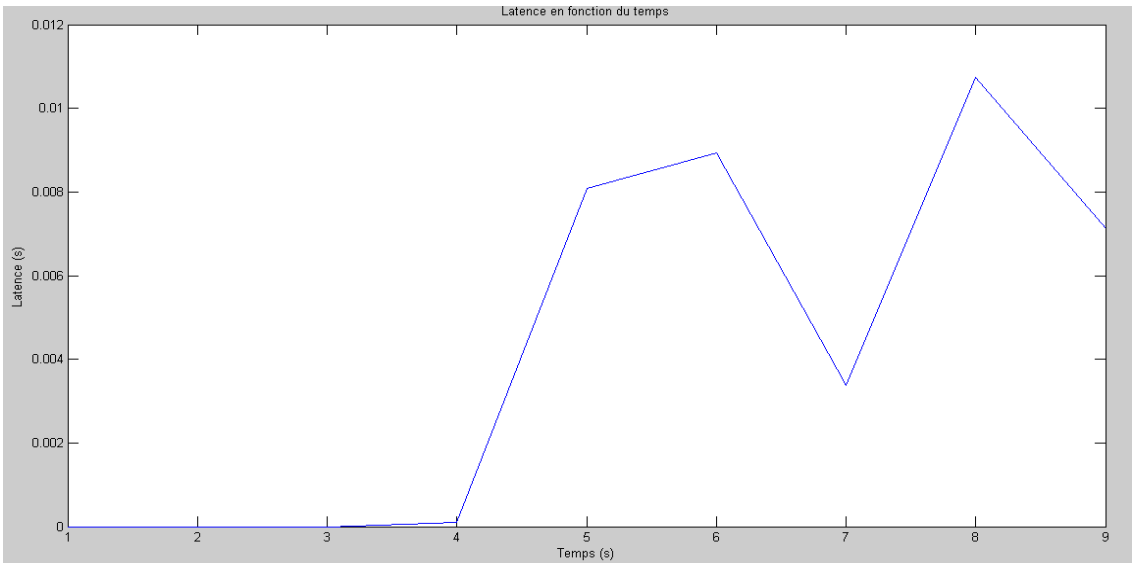

Figure 4.14: Latence avec DLP

La valeur maximal de la latence pour le premier script est 2mn par contre pour le deuxième script cette valeur ne dépasse pas 0.012mn se qui signifie que les paquets d'une même application arrive a la destination avec une différence de temps négligeable.

- **Nombre d'applications :**

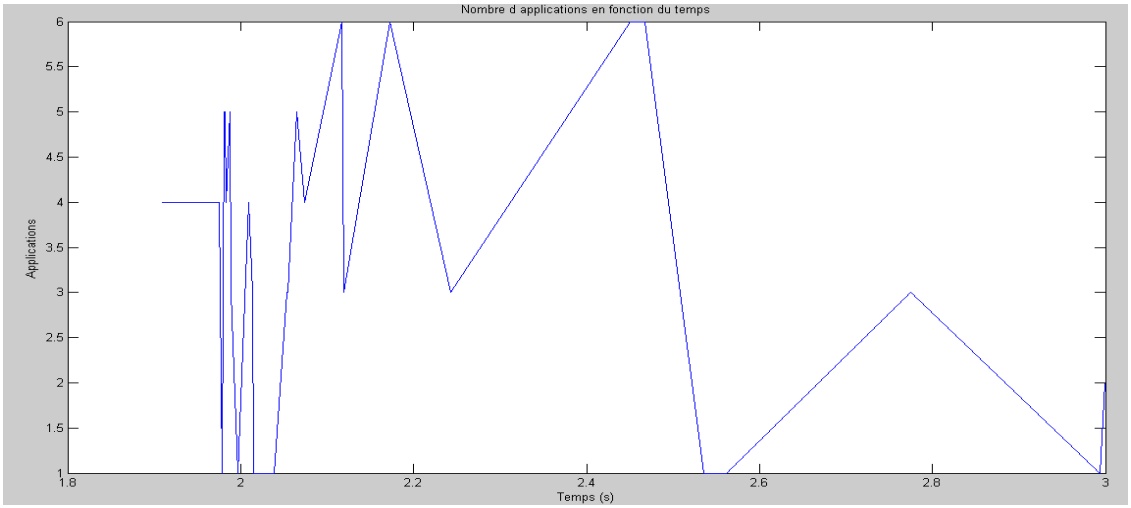

Figure 4.15: Nombres d'applications sans DLP

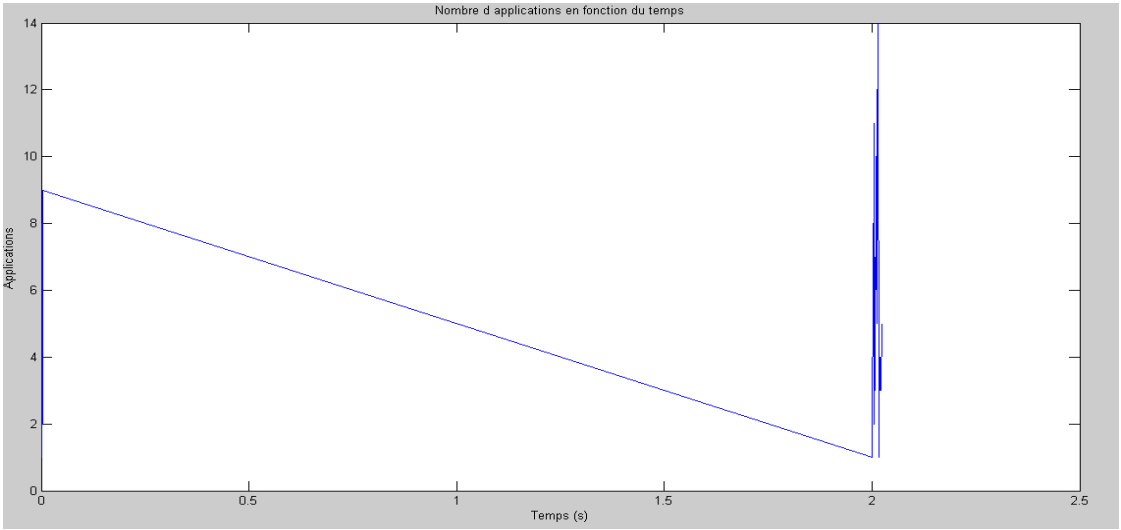

Figure 4.16 : Nombres d'applications avec DLP

Nous remarquons que le point d'accès se saturera avec six applications seulement dans le premier script par contre dans le deuxième se nombre peut arriver jusqu'à **quatorze** applications.

- **Nombre de paquets envoyés :**

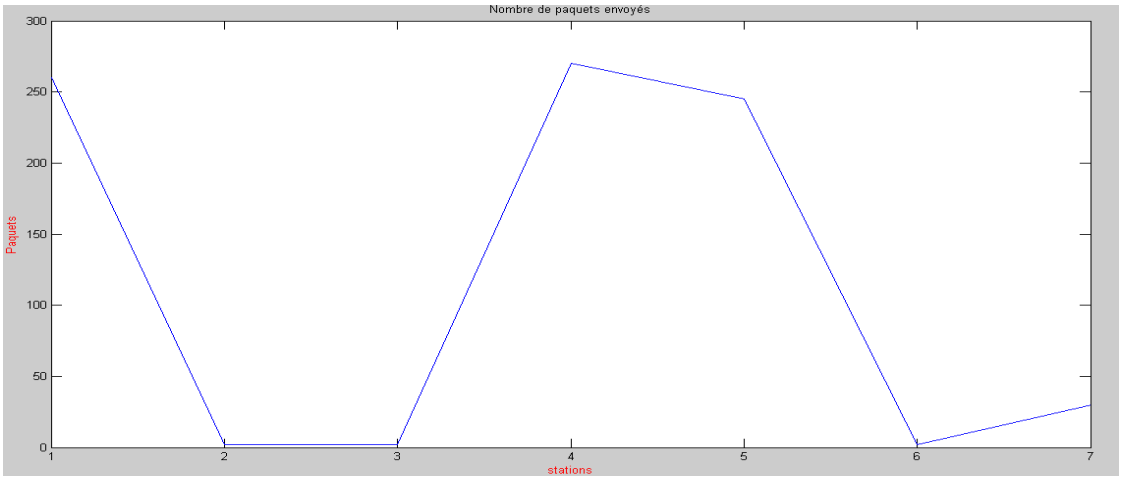

Figure 4.17: Nombre de paquets envoyés sans DLP

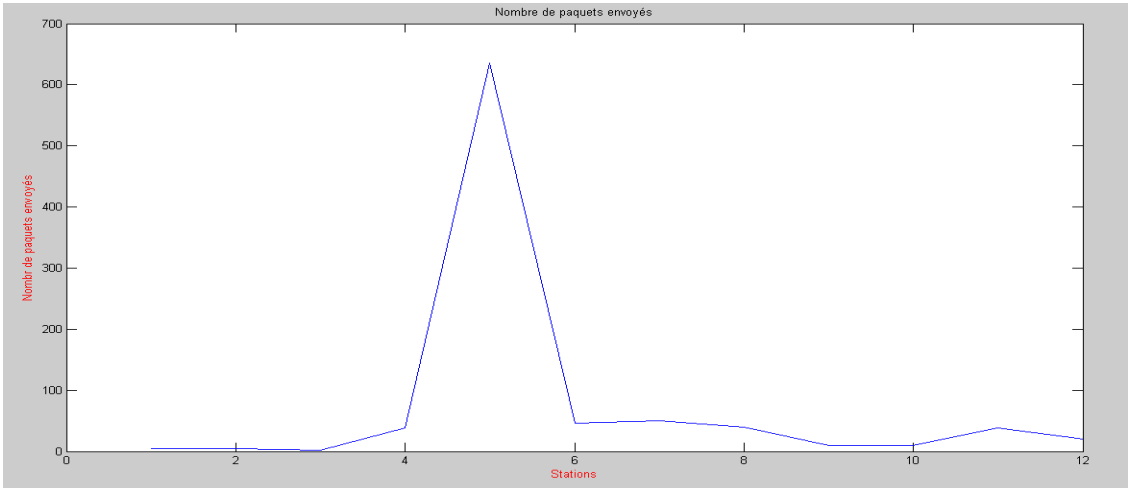

Figure 4.18: Nombre de paquets envoyés avec DLP

Le nombre de paquets maximum qui peut être envoyés dans le premier script est **250** paquets se nombre peut atteindre **600** paquets dans le deuxième script mais dans les deux cas aucun paquet n'est perdu.

X Conclusion :

Direct Link Protocol comme son nom l'indique est un protocole permettant la communication directe entre deux stations d une même cellule et se trouvant l'un à la portée de l'autre ; ce protocole permet en fait l'envoi de message dans un pseudo mode ad hoc, c a d sans passer par le point d'accès ce qui permet d augmenter le nombre de clients de ce dernier. On a réussi à tester le bon fonctionnement du DLP et démontrer l'avantage certain qu'apporte ce mécanisme dans le gain de ressources du réseau.