

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
Ministère de L'Enseignement Supérieur et de la Recherche Scientifique

UNIVERSITE SAAD DAHLAB – BLIDA1

Faculté des Sciences

Département d'Informatique



**Mémoire de Fin d'Étude pour l'Obtention du Diplôme de Master
en Informatique**

OPTION : Systèmes Informatiques et Réseaux

Thème :

**Évaluation des performances du protocole IEEE
802.11e**

Réalisé par : LAOUAR Samir

Soutenu le : 28/09/2020 devant le jury composé de :

- Mr BENYAHIA Mohamed	President
- Mr KAMECH Abdallah	Examineur
- Mr OULD-KHAOUA Mohamed	Promoteur

Année Universitaire : 2019-2020

Remerciements

Je remercie **ALLAH** de m'avoir donné la santé, la volonté et la persévérance pour terminer ce travail.

Au terme de ce travail, Je tiens à exprimer mes vifs remerciements à toutes les personnes qui ont contribué de près ou de loin à sa réalisation.

Ma profonde gratitude à Monsieur Mohamed Ould-Khaoua mon promoteur qui a proposé et dirigé ce sujet de mémoire de master, pour la documentation mise à ma disposition ainsi que son encadrement précieux et ses conseils ses critiques et sa continuelle présence tout au long de ce travail.

Je tiens à adresser mes sincères remerciements à Hicham qui m'a aidé tout au long de ce travail.

Je remercie tout le personnel du département informatique pour l'excellent environnement et l'ambiance agréable tout au long de cette formation.

Je présente aussi mes remerciements à Monsieur Guesmia Abdelkader, le chef du département, pour son aide et son encouragement tout au long de ma formation.

Pour finir, je remercie les membres du jury d'avoir accepté d'évaluer ce travail.

Merci à tous

Dédicaces

Je dédie ce modeste travail :

A mes chers parents, pour leurs sacrifices, leur tendresse et leurs prières tout au long de mes études,

A ma chère femme, pour son encouragement permanent, sa patience et son soutien moral,

A mes enfants, pour leur sourire et leur innocence,

A toute ma famille,

A tous mes amis,

A toute personne qui m'a aidé, m'a encadré, m'a orienté, m'a conseillé tout au long de ce travail.

Résumé

Avec le déploiement rapide des réseaux locaux sans fil (WLAN), l'incapacité du WLAN 802.11 à prendre en charge des services en temps réel avec des exigences strictes en matière de qualité de service (QoS) est apparue, la méthode d'accès de base DCF (Distributed Coordination Function) dans les réseaux locaux sans fil du standard 802.11 est incapable d'assurer la performance demandée aux applications voix/vidéo, cela a conduit à l'exigence selon laquelle les WLAN 802.11 doivent fournir une qualité de service (QoS).

Cette thèse présente une évaluation du mécanisme EDCA (Enhanced Distributed Channel Access) d'IEEE 802.11e, conçu par le groupe IEEE 802.11 pour améliorer le MAC 802.11 original, ce protocole est évalué pour vérifier qu'il offre des performances de qualité de service supérieures pour les applications en temps réel par rapport à l'ancienne méthode d'accès IEEE 802.11 DCF.

Pour cela, nous menons une étude approfondie des mécanismes utilisés pour la prise en charge de la qualité de service dans le protocole IEEE 802.11e et ses variantes améliorées, on procède ensuite à l'analyse des capacités de différenciation des paramètres EDCA QoS importants au moyen d'une analyse de débit et de délai. Le but principal d'une telle analyse est d'obtenir un aperçu de la meilleure façon d'appliquer les paramètres de QoS pour prendre en charge les exigences de QoS. Nous allons entre autres contribuer à la comparaison du mécanisme 802.11 DCF et 802.11e EDCA à l'aide du simulateur de réseau (NS-2) avec deux paramètres qui sont le débit et le délai. Une discussion est présentée en détail à l'aide d'évaluations basées sur la simulation.

Les résultats de la simulation prouvent essentiellement l'efficacité de la différenciation des services MAC fournie par EDCA par rapport à DCF, et nous confirmons la QoS de IEEE 802.11e par rapport à IEEE 802.11.

Mots clés : Réseaux locaux sans-fil, IEEE 802.11, IEEE 802.11e, DCF, EDCA, QoS, WLAN, MAC.

Abstract

With the rapid deployment of wireless local area networks (WLANs), the inability of WLAN 802.11 to support real-time services with stringent quality of service (QoS) requirements emerged, the access method of DCF (Distributed Coordination Function) base in wireless LANs of the 802.11 standard is unable to ensure the performance demanded of voice / video applications, this has led to the requirement that 802.11 WLANs must provide quality of service (QoS).

This thesis presents an evaluation of the IEEE 802.11e Enhanced Distributed Channel Access (EDCA) mechanism, designed by the IEEE 802.11 group to improve the original 802.11 MAC, this protocol is evaluated to verify that it offers higher quality of service performance for real-time applications compared to the older IEEE 802.11 DCF access method.

To do this, we conduct an in-depth study of the mechanisms used for the support of the quality of service in the IEEE 802.11e protocol and its enhanced variants, the next step is to analyze the differentiation capabilities of important EDCA QoS parameters using throughput and delay analysis. The main purpose of such an analysis is to gain insight into how best to apply QoS settings to support QoS requirements. Among other things, we will contribute to the comparison of the 802.11 DCF and 802.11e EDCA mechanism using the network simulator (NS-2) with two parameters which are the throughput and the delay. A discussion is presented in detail using simulation-based evaluations.

The simulation results essentially prove the effectiveness of MAC service differentiation provided by EDCA compared to DCF, and we confirm the QoS of IEEE 802.11e compared to IEEE 802.11.

Keywords: Wireless Local Area Network (WLAN), IEEE 802.11, IEEE 802.11e, DCF, EDCA, Quality of Service (QoS), WLAN, MAC.

خلاصة المشروع

مع الانتشار السريع للشبكات المحلية اللاسلكية (WLAN)، ظهر عدم قدرة 802.11 WLAN على دعم خدمات الوقت الفعلي بمتطلبات جودة الخدمة (QoS) الصارمة، آلية DCF في الشبكات المحلية اللاسلكية لمعيار 802.11 غير قادرة على توفير الأداء المطلوب لتطبيقات الصوت / الفيديو، وقد أدى ذلك إلى اشتراط أن توفر شبكات WLAN 802.11 جودة الخدمة (QoS).

تقدم هذه الأطروحة تقييماً لآلية EDCA (Enhanced Distributed Channel Access) لبروتوكول IEEE 802.11e، الذي صممه مجموعة IEEE 802.11 لتحسين MAC 802.11 الأصلي، ويتم تقييم هذا البروتوكول للتحقق من أنه يوفر جودة أعلى من أداء الخدمة لتطبيقات الوقت الفعلي مقارنة بطريقة الوصول الأقدم IEEE 802.11 DCF.

للقيام بذلك، تجري دراسة متعمقة للآليات المستخدمة لدعم جودة الخدمة في بروتوكول IEEE 802.11e ومتغيراته المحسنة، ثم نحلل قدرات التمايز لمعلمات EDCA QoS المهمة باستخدام تحليل التدفق Débit والتأخير Délai. الهدف الرئيس من هذه الدراسة هو الكشف عن الطريقة الأفضل لتطبيق عوامل جودة الخدمة المتغيرة (Paramètres de QoS) لدعم احتياجات جودة الخدمة. سنساهم في مقارنة آلية DCF 802.11 و EDCA 802.11e باستخدام محاكي الشبكات اللاسلكية (NS2) مع معيارين هما التدفق والتأخير، يتم عرض مناقشة بالتفصيل باستخدام التقييمات القائمة على المحاكاة.

تثبت نتائج المحاكاة بشكل أساسي كفاءة تمايز خدمة MAC التي توفرها EDCA مقابل DCF، ونؤكد جودة الخدمة لـ IEEE 802.11e مقارنة بـ IEEE 802.11.

الكلمات المفتاحية: الشبكات اللاسلكية WLAN، IEEE 802.11، IEEE 802.11e، DCF، EDCA، QoS، شبكات WLAN، MAC.

TABLE DES MATIERES

Résumé	IV
Abstract.....	V
خلاصة المشروع	VI
TABLE DES MATIERES	VII
LISTE DES FIGURES	X
LISTE DES TABLEAUX.....	XI
Abréviations.....	XII
Chapitre 1 Introduction.....	1
1.1 Introduction générale.....	1
1.2 Problématique.....	3
1.3 Objectif du projet.....	4
1.4 Contributions.....	5
1.5 Plan du manuscrit	6
Chapitre 2 Réseau LAN sans fil.....	7
2.1 Introduction	7
2.2 Présentation de la norme IEEE 802.11.....	8
2.3 Fonction de coordination distribuée (DCF)	9
2.3.1 Procédure d'évitement des collisions et d'interruption (Collision Avoidance and Backoff Procedure)	11
2.4 Limitations de IEEE 802.11.....	13
2.4.1 Qualité de Service.....	13
2.4.2 Les paramètres de qualité de service.....	14
2.4.3 Limitations de QoS de DCF	16

2.5	La norme IEEE 802.11e.....	17
2.6	EDCA.....	18
2.6.1	Catégories d'accès (AC).....	18
2.6.2	Les paramètres EDCA.....	20
2.7	Le mécanisme EDCA dans IEEE 802.11e.....	24
2.8	Travaux connexes et motivations.....	25
2.9	Résumé.....	27
Chapitre 3 Impact des paramètres de différenciation QoS sur les performances EDCA.....		
		28
3.1	Introduction.....	28
3.2	Présentation du simulateur NS2.....	29
3.2.1	Principe de base.....	30
3.2.2	Visualisation des résultats.....	30
3.3	Impact des paramètres de différenciation QoS sur les performances EDCA	32
		32
3.3.1	Scénario : Différenciation par l'un des quatre paramètres EDCA QoS.....	32
3.3.2	Topologie de réseau pour le scénario.....	32
3.3.3	Configuration de la simulation Figure 3.3: Fichier NAM de la topologie du scénario.....	34
3.3.4	Les paramètres de simulation Tableau 3.1: Paramètres EDCA par défaut.....	35
3.4	Résultats des simulations et analyse pour le scénario.....	36
3.4.1	Différenciation par CWmin.....	36
3.4.2	Différenciation par CWmax.....	39
3.5	Conclusions.....	42

Chapitre 4 Etude de comparaison entre les protocoles IEEE 802.11e et IEEE 802.11	43
.....	
4.1 Introduction	43
4.2 Configuration de la simulation	43
4.3 Les paramètres de simulation	44
4.4 Comparaison DCF et EDCA Figure 4.1: Débit DCF scénario2	45
4.5 Analyse de simulation de l'EDCA	48
4.6 Conclusion	55
Conclusions et Futurs travaux	56
Références	58
Annexe A: Script OTCL de différenciation	60
Annexe B: Script OTCL de comparaison	63

LISTE DES FIGURES

Figure 2.1: Mécanisme d'accès DCF de base [7]	9
Figure 2.2: Relations entre les espaces inter trames [8][9]	10
Figure 2.3: Quatre CA, chacun avec sa propre file d'attente et ses paramètres EDCA [9]	20
Figure 2.4: Priorité du trafic basée sur AIFS [17]	22
Figure 3.1: L'interface NAM [26]	31
Figure 3.2: La topologie du scénario	33
Figure 3.3: Fichier NAM de la topologie du scénario	34
Figure 3.4: Débit EDCA, Paramètres $CW_{min}=7$ et $CW_{min}=15$	36
Figure 3.5: Délai EDCA, Paramètres $CW_{min}=7$ et $CW_{min}=15$	37
Figure 3.6: Débit EDCA, Paramètres $CW_{max}=31$ et $CW_{max}=1023$	39
Figure 3.7: Délai EDCA, Paramètres $CW_{max}=31$ et $CW_{max}=1023$	40
Figure 4.1: Débit DCF scénario2	45
Figure 4.2: Débit EDCA scénario2	46
Figure 4.3: Délai DCF scénario2	47
Figure 4.4: Délai EDCA scénario2	47
Figure 4.5: Délai EDCA scénario1	49
Figure 4.6: Délai EDCA scénario2	50
Figure 4.7: Délai EDCA scénario3	51
Figure 4.8: Débit EDCA scénario1	52
Figure 4.9: Débit EDCA scénario2	53
Figure 4.10: Débit EDCA scénario3	54

LISTE DES TABLEAUX

Tableau 2.1: Dimensions des performances QoS de l'application [15]	15
Tableau 2.2: Relation entre les priorités et les catégories d'accès dans EDCA [9]...	19
Tableau 2.3: Valeurs standard des paramètres EDCA [8].....	21
Tableau 2.4: Valeurs standard des paramètres CW pour différentes couches physiques [3]	23
Tableau 3.1: Paramètres EDCA par défaut [8].....	35
Tableau 3.2: Paramètres EDCA pour la simulation [27]	35
Tableau 4.1: Paramètres MAC pour la couche PHY 802.11b [28].....	44

Abréviations

AC	Access Category
AP	Access Point
AIFS	Arbitrary Inter-frame Spacing
BSS	Basic Service Set
CSMA/CA	Carrier Sense Multiple Access with Collision Avoidance
CSMA/CD	Carrier Sense Multiple Access with Collision Detection
CW	Contention Window
CR-EDCF	Collision Rate-based Adaptive EDCF
DCF	Distributed Coordination Function
DSSS	Direct Sequence Spread Spectrum
DIFS	DCF Inter-Frame Space
EDCA	Enhanced Distributed Channel Access
EDCAF	Enhanced Distributed Channel Access Function
FHSS	Frequency Hopping Spread Spectrum
HCF	Hybrid Coordination Function
IP	Internet Protocol
IR	Infrared
ISM	Industrial, Scientific and Medical
IBSS	Independent Basic Service Set
MAC	Media Access Control
NS2	Network Simulator2
OFDM	Orthogonal Frequency Division Multiplexing
PCF	Point Coordinator Function
PC	Point Coordinator
PIFS	PCF Inter-Frame Space
PHY	Physical
QoS	Quality-of-Service
QSTA	Quality of Service Station

SIFS	Short Inter-Frame Space
TXOP	Transmission Opportunity
VoIP	Voice over IP
WLANs	Wireless Local Area Networks

Chapitre 1 Introduction

1.1 Introduction générale

Nous vivons dans un monde sans précédent, avec des dimensions et des horizons totalement nouveaux, dans lequel les limites de la technologie et de l'imagination ont été repoussées à l'infini. La révolution informationnelle est due principalement au développement fulgurant des technologies de l'information qui permettent actuellement le traitement, le stockage et la transmission d'énormes quantités de données quasiment en temps réel. L'une de ces technologies est celle des réseaux sans fil qui a fait son apparition vers la fin des années 80. Dans le marché des réseaux sans fil, il existe plusieurs normes sans fil : HyperLan, HomeRF, IEEE 802.11, Bluetooth ...etc., la plus utilisée pour le moment, est sans doute, l'IEEE 802.11, à cause de son coût, sa rapidité et facilité d'installation par rapport aux autres normes sans fil. Cette technologie offre aux gens un environnement de communication et informatique omniprésent dans les bureaux, les hôpitaux, les campus, les usines, les aéroports, les marchés boursiers, etc.

Avec la hausse de la demande et des besoins en QoS [1] (Quality of Service, en français: Qualité de service) par les utilisateurs et spécialement dans les applications multimédia, beaucoup de recherches se sont orientées vers la garantie de la QoS dans les réseaux. Ainsi, on note ces dernières années des avancées rapides dans la standardisation de nombreuses technologies sans fils allant des réseaux personnels à faible couverture jusqu'aux réseaux à couverture mondiale.

La norme IEEE 802.11 décrit les deux couches basses de la pile OSI (la couche physique PHY et la sous-couche MAC). Au départ 3 couches physiques [2] ont été définies à savoir : le spectre étalé à saut de fréquence (FHSS), le spectre étalé à séquence directe (DSSS) et l'infrarouge (IR), avec un taux de transmission de données maximal pouvant

atteindre 2 Mbps. Les couches physiques DSSS et FHSS fonctionnent dans la bande ISM (2,4 GHz industrielle, scientifique et médicale) sans licence.

En 1999, le standard 802.11 [3] a défini la méthode d'accès distribuée DCF. Cette dernière a été conçue pour le support unique des applications élastiques de transfert de données et du web, et il n'a pas été prévu initialement que le standard DCF puisse être utilisé par les applications exigeantes en termes de QoS comme la voix et la vidéo. Dès lors, 802.11 DCF a été incapable d'assurer la performance demandée par ce type d'applications. La présence de cette limitation accompagnée du besoin croissant du multimédia a poussé les activités de recherche à investiguer et proposer différentes solutions possibles capables d'améliorer la performance des WLANs. Parmi ces différentes solutions, une solution basée sur la différenciation de service a été retenue par l'IEEE 802.11 qui a fondé un groupe de travail qui a abouti en 2005 à la standardisation d'un amendement de QoS appelée 802.11e. Dans cet amendement, DCF a été remplacée par EDCA (Enhanced Distributed Channel Access). Qui supporte quatre catégories d'accès (Access Category ou AC), chacune possède ses propres paramètres d'accès: Arbitration Inter Frame Spacing (AIFS), Contention Window (CW) et Transmission Opportunity Limit (TXOPLimit). La priorité de chaque catégorie d'accès est définie par les valeurs de ces paramètres qui ne sont pas fixes comme dans DCF mais ajustables selon les besoins.

802.11e EDCA a réussi à améliorer la performance du 802.11. La différenciation de service qui y est introduite a abouti à de bons comportements du protocole avec les applications ayant des contraintes de QoS. Cependant, lorsque le réseau est saturé, aucune garantie de QoS ne peut être assurée par EDCA. Plusieurs études d'évaluation d'EDCA ont démontré cette limitation, et plusieurs propositions d'amélioration ont essayé de résoudre ce problème.

1.2 Problématique

Les réseaux locaux sans fil (Wireless Local Area Network ou WLAN) ont reçu ces dernières années un grand succès grâce à leur simplicité, rapidité et faible coût de déploiement. Dès lors, ils constituent une alternative sérieuse aux réseaux locaux filaires. On commence d'ailleurs à les trouver partout : dans les aéroports, les hôtels, les bureaux et ainsi que dans les environnements domestiques.

Simultanément, les applications multimédias ont connu une croissance explosive. Les gens doivent maintenant recevoir des services vidéo, audio, voix et Web à haute vitesse même lorsqu'ils se déplacent dans des bureaux ou se déplacent sur les campus.

Bien que IEEE 802.11 est devenu populaire en raison de son faible coût et le déploiement facile, il ne fournit pas de support de qualité de service (QoS). La QoS se réfère à la capacité d'un réseau à fournir certains services cohérents pour la transmission de données, et elle est mesurée en termes de caractéristiques qualitatives, telles que le débit, le retard et la perte de paquets, qui décrivent la qualité du trafic de données sur le réseau. IEEE 802.11 a été incapable d'assurer la performance demandée par ce type d'applications, cette incapacité du mécanisme MAC 802.11 à fournir le support QoS a été un obstacle dans l'adaptation des applications multimédias modernes dans les WLAN 802.11. Pour résoudre ce problème, le comité d'IEEE 802.11 a formé le comité du groupe de travail E (802.11e) pour définir des perfectionnements au MAC 802.11 original.

1.3 Objectif du projet

L'objectif de ce projet est de :

1. Mener une étude approfondie des mécanismes utilisés pour la prise en charge de la qualité de service dans le protocole IEEE 802.11e et ses variantes améliorées.
2. L'impact des paramètres importants de l'EDCA, y compris CWmin [AC] et CWmax [AC] respectivement, sera évalué sur la performance des mécanismes de QoS suggérés dans IEEE 802.11e. Notre objectif est d'identifier un réglage optimal de ces paramètres qui améliore la performance de la QoS du protocole IEEE 802.11e.
3. Une étude comparative de la norme IEEE802.11e et de la norme simple IEEE 802.11 sera menée pour accéder à la performance WLAN.

Pour atteindre l'objectif visé, nous suivons la démarche suivante :

- Comme première étape de notre recherche, nous réalisons une étude détaillée du standard IEEE 802.11 et ses mécanismes d'accès, nous discuterons en suite leurs limitations, qui sont à la base de l'apparition du mécanisme EDCA à l'avènement de l'amendement 802.11e.
- Pour identifier les métriques de QoS à adopter dans le contrôle d'admission, nous choisissons le délai de transmission et le débit atteignable. Ce qui nous intéresse dans cette thèse, c'est la performance du mécanisme d'accès EDCA uniquement. Autrement dit, nous cherchons à identifier le délai et le débit atteignables qui résultent de l'application de la méthode d'accès EDCA.

- Nous allons étudier les performances de l'IEEE 802.11e en faisant varier les paramètres de priorité en utilisant le simulateur NS2, ensuite nous allons réaliser une étude comparative des résultats de simulation avec celles de 802.11.

1.4 Contributions

Notre contribution dans ce mémoire consiste à étudier et évaluer les performances du protocole IEEE 802.11e. Pour faire cela nous réalisons une étude détaillée du protocole IEEE 802.11 et de l'état de l'art en relation avec celui-ci. Le chapitre 2 donne un aperçu de la norme IEEE 802.11.

Afin de garantir que la prise en charge de la QoS est disponible et que les performances du système sont améliorées dans les WLAN, nous affirmons qu'il est important de rechercher le domaine des techniques d'accès aux canaux distribués qui prennent en compte les exigences de QoS. Cette recherche se concentre sur les améliorations de la couche MAC de la fonction EDCA de la norme 802.11e.

Notre approche consiste à commencer par étudier l'impact des paramètres EDCA importants, tels que CW_{min} , CW_{max} respectivement, sur les performances des mécanismes de QoS de la fonction EDCA (EDCF). L'objectif est d'évaluer le rôle de chaque paramètre dans les capacités de différenciation de l'EDCF.

1.5 Plan du manuscrit

Pour détailler l'ensemble de nos contributions et notre démarche de travail, ce manuscrit de thèse est organisé comme suit :

Après ce chapitre introductif, le Chapitre 2 a comme objectif la présentation d'une étude de l'état de l'art couvrant le standard 802.11. Ainsi, nous décrivons les spécifications des mécanismes d'accès de base dans 802.11, puis nous discutons des limitations de QoS de ces mécanismes et des propositions d'amélioration de leurs performances. Ensuite, nous décrivons les spécifications des mécanismes d'accès de l'amendement de QoS du standard 802.11e. Nous nous intéressons plus particulièrement à la méthode d'accès EDCA.

Le chapitre 3 fournit une vue d'ensemble de l'outil de simulation NS2, et décrit le scénario de simulation utilisé pour la différenciation par deux paramètres EDCA. Le chapitre fournit également une analyse des résultats de la simulation.

Le chapitre 4 concerne la comparaison des performances entre les standards IEEE 802.11e et IEEE 802.11 simples, en vue de démontrer l'efficacité de l'EDCA par rapport à DCF.

Le mémoire s'achève avec une conclusion générale récapitulant le contexte de recherche de notre étude, la démarche suivie, nos contributions et énonce un ensemble de perspectives.

Chapitre 2 Réseau LAN sans fil

2.1 Introduction

Le réseau local sans fil IEEE 802.11 (WLAN) [3] est l'une des technologies de réseau sans fil les plus largement déployées. Ces dernières années, de nombreuses normes et technologies de connexion sans fil ont vu le jour. Ces technologies permettent aux utilisateurs de se connecter. Il n'est pas nécessaire d'acheter, et il est facile de transporter ou de connecter des câbles et de simplement mettre en œuvre un grand nombre d'équipements de télécommunications.

Le mécanisme MAC (Media Access Control) de base du 802.11 est appelé fonction de coordination distribuée (DCF). DCF est basé sur l'accès aux canaux distribués et utilise le protocole CSMA (Carrier Sense Multiple Access) pour l'accès aux médias. Par conséquent, il est nécessaire de fournir des informations plus détaillées sur la norme 802.11, car la recherche et l'amélioration de la norme sont au cœur de nos travaux de recherche. Dans ce chapitre, nous décrirons la norme 802.11 et ses mécanismes d'accès, tels que DCF. Ensuite, nous discuterons des limitations de IEEE 802.11 dans la fourniture de QoS.

Une autre variante de 802.11 (802.11e) a été développée par IEEE pour garantir Service QoS au niveau de la couche MAC. Dans le reste de ce chapitre, nous présentons la nouvelle variante 802.11e et son mécanisme pour fournir la QoS. Enfin, il y a un résumé de ce chapitre.

2.2 Présentation de la norme IEEE 802.11

L'IEEE a publié la norme 802.11 Wireless Local Area Network (WLAN) [3] en 1997. Cette norme appartient au groupe des normes IEEE 802.x populaires, par exemple IEEE 802.3 CSMA / CD [4]. IEEE 802.11 définit les spécifications des couches Media Access Control (MAC) et Physical (PHY) pour les LAN sans fil.

La norme comprend trois couches PHY, qui sont une bande de base infrarouge (IR) PHY ; une radio à spectre étalé à sauts de fréquence (FHSS) et une radio à spectre étalé en séquence directe (DSSS). Ces choix entiers prennent en charge le taux PHY 1 et 2 Mbps. En 1999, l'IEEE définit deux taux élevés : 802.11b dans la bande 2,4 GHz à 11 Mbps, basé sur la technologie DSSS ; et 802.11a dans la bande 5 GHz à 54 Mbps, basée sur la technologie de multiplexage par répartition en fréquence orthogonale (OFDM). Récemment, le 802.11g a été finalisé pour être une extension du 802.11b avec 54 Mbps dans la bande 2,4 GHz [5].

IEEE 802.11 a gagné en popularité en raison de sa rentabilité et est considérée comme l'une des technologies de réseau sans fil les plus largement déployées au monde. IEEE 802.11 définit deux fonctions de coordination d'accès moyen, la fonction de coordination distribuée de base (DCF) et la fonction de coordination de point facultative (PCF) (IEEE 802.11 WG, 1999). IEEE 802.11 définit deux modes : le mode ad-hoc et le mode infrastructure. Le premier mode forme un BSS indépendant (IBSS) où les STA peuvent communiquer directement les uns avec les autres en utilisant uniquement le DCF, sans aucune connectivité à aucune dorsale filaire. Dans le deuxième mode, les STA communiquent avec la dorsale filaire via le pont de point d'accès (AP), qui peut utiliser à la fois DCF et PCF.

La couche MAC de 802.11 peut utiliser deux modes de fonctionnement : Distributed Coordination Function (DCF) et Point Coordination Function (PCF). DCF est un mode qui peut être utilisé par tous les mobiles, et qui permet un accès équitable au canal radio

sans aucune centralisation de la gestion de l'accès (mode totalement distribué). Ce mode peut aussi bien être utilisé lorsqu'il n'y a pas de stations de base (mode Ad hoc) que lorsqu'il y en a (mode infrastructure), PCF est une méthode optionnelle implémentée uniquement dans les architectures basées sur une infrastructure, qui consiste en une gestion centralisée des ressources, où les stations de base ont la charge de la gestion de l'accès au canal dans leur zone de couverture pour les mobiles qui leur sont rattachés [6].

Dans ce projet, on ne considère que le mode DCF, qu'on va discuter plus en détail dans la section suivante, car il est la base de l'accès amélioré aux canaux distribués (EDCA) introduit dans IEEE 802.11e.

2.3 Fonction de coordination distribuée (DCF)

La méthode d'accès fondamentale DCF utilise le modèle d'accès CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance). Ce dernier fonctionne comme une écoute avant la conversation, c'est-à-dire la station émettrice transmet le signal et écoute en même temps le canal. Si le support est trouvé inactif au moins pendant la période de temps de l'espace inter-trame DCF (DIFS), la station commence la transmission et les autres stations attendent que le support redevient inactif pendant au moins la période de temps DIFS.

La station de destination reçoit avec succès une trame, elle le reconnaît en renvoyant une trame ACK après une courte période d'espace inter-trame (SIFS). La figure 2.1 illustre le mécanisme.

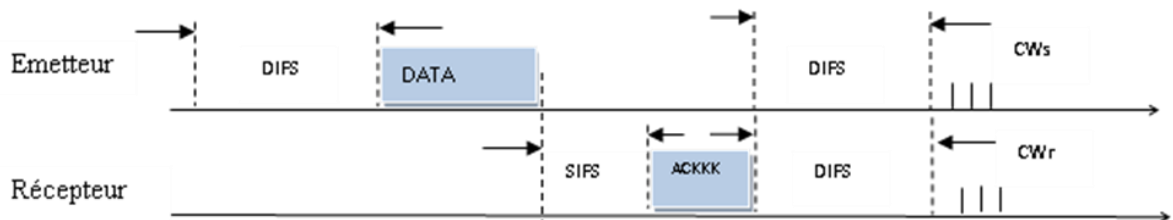


Figure 2.1: Mécanisme d'accès DCF de base [7]

Pour assurer le fonctionnement d'accès au canal, le DCF se base sur un jeu d'intervalles de temps appelé IFS (Inter Frame Spacing). Comme le montre la figure 2.2.

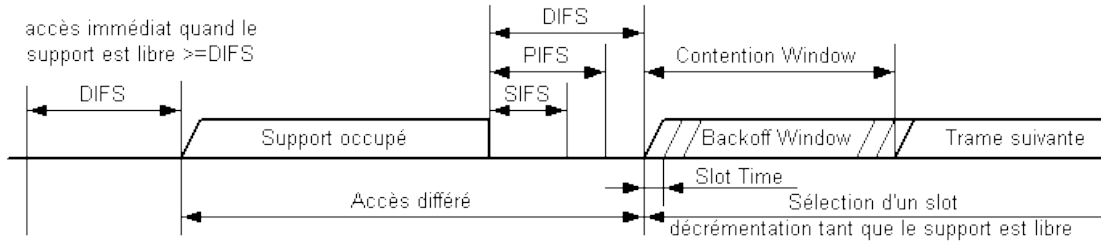


Figure 2.2: Relations entre les espaces inter trames [8][9]

PIFS est utilisé par les points d'accès dans le PCF (mécanisme d'accès IEEE 802.11 optionnel), où le point d'accès (AP) contrôle de manière centralisée l'accès aux supports en interrogeant les stations individuelles. Dans PCF, la priorité de l'AP est supérieure à celle des stations ordinaires, il doit donc attendre PIFS au lieu de DIFS plus long avant d'envoyer des trames.

SIFS est le plus court d'IFS. SIFS est le temps entre la trame de données et la trame ACK, comme le montre la figure 2.2, de sorte qu'il peut empêcher d'autres stations d'envoyer en même temps. Le récepteur envoie une trame ACK car les autres stations doivent attendre le temps DIFS, qui est plus long que SIFS, avant de commencer la transmission. Ainsi, de cette manière, la station émettant la trame ACK a priorité sur la station qui tente de transmettre la trame de données. Les valeurs IFS sont définies en temps de créneau et dépendent de la couche physique sous-jacente (PHY).

Il existe deux types de détection de porteuse utilisés pour déterminer si le support est inactif ou occupé. Grâce à la détection de porteuse physique, le canal sans fil peut être détecté au niveau de la couche physique elle-même. Lors de l'utilisation de la détection de porteuse virtuelle dans la couche MAC, la station recevant la trame qui ne lui est pas dirigée ; elle vérifiera le champ de durée dans l'en-tête de trame, qui spécifie le temps

requis pour envoyer la trame et recevoir l'ACK en réponse, puis diffère l'accès au support pour cette période de temps particulière.

2.3.1 Procédure d'évitement des collisions et d'interruption (Collision Avoidance and Backoff Procedure)

Une collision se produit si deux stations ou plus détectent le ralenti moyen et tentent de transmettre en même temps. Pour éviter de telles collisions, la station doit attendre un autre temps avant de transmettre si le support est détecté occupé pendant la période DIFS ou, si le support était occupé juste avant que la station ne commence à attendre la période DIFS. Dans ces cas, la station retardera l'accès jusqu'à ce que le support devient inactif, puis sélectionne une valeur d'interruption aléatoire, qui spécifie la période de temps en unités de tranches de temps ; Après que le support est devenu inactif, la station doit attendre en plus de DIFS.

Ce retard aléatoire supplémentaire sous forme de décroissance permet d'éviter les collisions. Sinon, une fois que le support devient inactif pendant DIFS, toutes les stations tentent de transmettre. Ce mécanisme est appelé évitement des collisions (CA) et l'ensemble du mécanisme d'accès est appelé CSMA/CA.

La détection de porteuse avec prévention des collisions est utilisée dans les réseaux sans fil au lieu d'utiliser la détection de collision (CSMA/CD) dans les réseaux câblés, comme IEEE 802.3 Ethernet [4], en raison du manque de détection de collision dans les réseaux sans fil. Dans un réseau câblé, l'émetteur-récepteur a la capacité de recevoir et d'envoyer simultanément, de sorte qu'il peut détecter les collisions, mais dans un réseau sans fil, les stations n'ont pas la capacité de fonctionner simultanément.

Après avoir sélectionné la valeur d'interruption (Backoff), étant donné que le support a été détecté comme étant inactif pendant la période de temps DIFS, la station a commencé à décrémenter son temporisateur d'interruption pour chaque intervalle de

temps. Si le support devient occupé pendant cette interruption, la station suspendra son temporisateur d'interruption. Une fois que le support est de nouveau inactif pendant DIFS, le temporisateur de retour reprend. Lorsque le temporisateur d'interruption atteint zéro, la station est autorisée à émettre.

Bien que la station doit choisir une nouvelle valeur d'interruption aléatoire, la première station qui a tentée continuera de décompter son temporisateur d'interruption au lieu de sélectionner un nouveau temporisateur d'interruption. Par conséquent, de cette manière, la station qui essaie en premier et attend donc plus longtemps est meilleur que la station qui essaie plus tard, car elle n'attendra que son temps de pause restant.

Sélectionnez aléatoirement une valeur d'interruption aléatoire dans l'intervalle de 0 à la fenêtre de collision $[0, CW]$ et réglez CW sur la taille minimale de la fenêtre de collision CW_{min} lors de la première tentative de transmission. Après chaque transmission infructueuse (collision), utilisez l'équation 2.1 pour doubler la CW et l'augmenter de façon exponentielle jusqu'à ce que la taille maximale de la fenêtre de collision CW_{max} soit atteinte. Les valeurs de CW_{min} et CW_{max} dépendent de la couche physique de base (PHY).

Pour la couche physique OFDM, les valeurs sont respectivement de 15 et 1023. Dans ce cas, la taille de la fenêtre de collision augmente sous la forme de 15, 31, 63, 127, 255, 511, 1023. Par conséquent, ce mécanisme est également appelé interruption exponentielle.

$$CW_{nouveau} = 2 \times CW_{courant} + 1 \dots\dots\dots(2.1)$$

Si l'expéditeur ne reçoit pas la trame ACK dans le délai ACK spécifié, il est déterminé que la transmission est infructueuse (collision). Après la période de temporisation ACK, la station suppose qu'une collision s'est produite, attend que le support soit inactif pour DIFS, puis entre à nouveau dans un temps d'interruption (backoff) avec une taille

doublée de CW, ceci réduit la probabilité de collision au cas où il y aurait plusieurs stations qui tenteraient d'accéder au canal.

DCF définit un nombre limité de tentative de retransmission qui est le nombre de fois qu'une trame peut être retransmise. S'il est déterminé que la transmission est infructueuse après avoir atteint la limite de retransmission, la trame sera abandonnée.

Après chaque transmission réussie, la station émettrice effectue un autre backoff (la taille CW est réinitialisée à CW_{min}) appelé le post-backoff, même si elle n'a pas de trame à émettre. Ce dernier garantit au minimum une période de backoff entre deux transmissions successives, et il permet aux autres stations de décrémenter leurs backoff et d'accéder au support.

Si les temporisateurs d'interruption (backoff) pour deux stations ou plus atteignent zéro en même temps, ou si deux stations ou plus obtiennent accidentellement les mêmes valeurs d'interruption. Cela indique que la probabilité de collision est inversement proportionnelle à la taille de la CW, c'est-à-dire que si la taille de la CW est plus petite, le taux de collisions plus élevé et vice versa.

2.4 Limitations de IEEE 802.11

2.4.1 Qualité de Service

La qualité de service (QDS) ou quality of service (QoS) est la capacité à véhiculer, dans de bonnes conditions, un type de trafic donné, en termes de disponibilité, débit, délai de transmission, gigue, taux de perte de paquets [10]. Appliquée aux réseaux à commutation de paquets (réseaux basés sur l'utilisation de routeurs) la QoS désigne l'aptitude à pouvoir garantir un niveau acceptable de perte des paquets, défini contractuellement, pour un usage donné (voix sur IP, vidéoconférence, etc.) [11].

2.4.2 Les paramètres de qualité de service

Les exigences de QoS peuvent être classées en trois dimensions principales : la bande passante (Bandwidth), le retard et la perte de données [12]. Le tableau 2.1 présente ces dimensions.

La bande passante :

C'est la quantité de données pouvant être transmise durant un laps de temps. Le plus souvent, elle est mesurée en nombre de bits/seconde (bits par seconde, bps) où un de ses ordres de grandeur : kilobits par seconde ou Kbps (10³ bps), mégabits par seconde ou Mbps (10³Kbps), gigabits par seconde ou Gbps (10³Mbps), etc. La bande passante est un paramètre clé de la vitesse de transmission des données : plus la bande passante est élevée, plus la transmission d'un fichier sera rapide. Ce paramètre est également crucial pour les applications temps réel. Par exemple, la vidéo en direct nécessite une bande passante plancher sous laquelle on ne doit pas descendre pour garantir une transmission en temps réel des images. La contrainte de bande passante est donc une valeur seuil de bande passante disponible sous laquelle il ne faut pas descendre [13].

Le Retard :

C'est le temps qui sépare l'émission d'un paquet de données de l'arrivée à sa destination. Il est mesuré en secondes où un de ses ordres de grandeur (de quelques millisecondes pour les transmissions nécessitant une très grande réactivité à plusieurs minutes pour les applications tolérantes aux délais). Une contrainte de délai définit un temps maximum à ne pas dépasser pour tous les paquets. Les applications temps réel et interactives sont très sensibles au délai. Par exemple un jeu vidéo en ligne nécessite un délai très court (non perceptible par les joueurs) pour garantir la fluidité de l'interaction entre les joueurs et avec le jeu [13]. Des retards importants détruisent gravement les performances de ces applications.

La Perte de données :

Elle caractérise le retard entre l'émission et la réception d'un paquet. La plupart des applications multimédias sont classées comme sensibles à la bande passante et au retard car elles nécessitent des garanties strictes de bande passante et de retard, mais sont généralement tolérantes aux pertes car elles peuvent tolérer une certaine quantité de pertes de données. La perte de données dans ces applications entraîne une qualité de sortie légèrement réduite, mais peut être négligée dans une certaine mesure. Les effets de ces pertes de données sur la qualité de l'application et le montant des pertes tolérables dépendent de l'application et du schéma de codage utilisé [14]. D'un autre côté, les applications orientées données telles que le transfert de fichiers et le courrier électronique sont considérées comme sensibles aux pertes. Ils peuvent tolérer des retards et des variations de bande passante mais nécessitent un transfert de données fiable.

Performance Dimensions			
Application	Sensibilité à :		
	Bande passante	Retard	Perte de données
VoIP	Faible	Haute	Moyenne
Vidéo conférence	Haute	Haute	Moyenne
Transactions client / serveur	Moyenne	Moyenne	Haute
E-mail	Faible	Faible	Haute
Transfert de fichier	Moyenne	Faible	Haute

Tableau 2.1: Dimensions des performances QoS de l'application [15]

2.4.3 Limitations de QoS de DCF

IEEE 802.11 est basé sur le meilleur service possible, DCF n'a aucune garantie de supporter la QoS ; en fait, il ne garantit pas la bande passante, le retard et la perte de données [16]. Tout le trafic de données est traité par une stratégie FIFO (First In First Out) de façon de meilleur effort.

Tous les STAs dans le BSS contestent pour le support sans fil avec la même priorité. Ceci entraîne le débit asymétrique entre la liaison montante et la liaison descendante, car l'AP a la même priorité que les autres STAs mais avec beaucoup d'exigence de haut débit. Aussi, Il n'y a aucune différenciation entre les flux de données pour supporter le trafic avec des exigences de QoS.

Quand le nombre de STAs dans un BSS augmente, la probabilité des collisions devienne plus haute et a comme conséquence les retransmissions fréquentes. Par conséquent la QoS diminue comme le débit global dans le BSS. Ce manque de garanties est une grande limitation de la fourniture de services QoS pour les applications multimédias dans les réseaux 802.11 [14].

En conséquence, le groupe de travail IEEE 802.11 a travaillé sur une nouvelle version de 802.11, appelée 802.11e [3], pour introduire la prise en charge de la QoS dans les réseaux 802.11.

2.5 La norme IEEE 802.11e

Toutes ces limitations pour DCF ont mené à un grand nombre d'activités de recherches pour améliorer le support de la QoS. En 2005, l'IEEE a publié une nouvelle norme appelée IEEE 802.11e [3], qui est un amendement à la norme IEEE 802.11 qui introduit des améliorations sur le plan de la QoS.

IEEE 802.11e prend en charge la QoS en introduisant un mécanisme de priorité. Il prend en charge la différenciation des services en affectant le trafic de données avec différentes priorités en fonction de leurs exigences de QoS.

Quatre catégories d'accès (AC) différentes ont été définies chacune pour le trafic de données d'une priorité différente. L'accès au support est accordé sur la base des priorités du trafic de données, de sorte que chaque trame avec une priorité particulière est mappée à un AC, et la différenciation de service est obtenue en utilisant différents ensembles de paramètres de conflit pour chaque AC à combattre pour le support. Dans IEEE 802.11e, la STA qui fournit des services QoS est appelé QSTA (QoS Station).

IEEE 802.11e introduit une nouvelle fonction, appelée fonction de coordination hybride (HCF), pour fournir une prise en charge QoS. HCF fournit des mécanismes d'accès aux canaux distribués et contrôlés centralement. Le mécanisme d'accès aux canaux distribué et basé sur le conflit de HCF est appelé Enhanced Distributed Channel Access (EDCA) qui sera décrit dans la section suivante.

2.6 EDCA

Toutes ces limitations pour DCF ont mené à un grand nombre d'activités de recherches pour améliorer le support de la QoS. En 2005, l'IEEE a publié une nouvelle norme appelée IEEE 802.11e [3], qui est un amendement à la norme IEEE 802.11 qui introduit des améliorations sur le plan de la QoS.

Dans EDCA, le support de la QoS est obtenu en introduisant plusieurs catégories d'accès (AC) dans chaque station prenant en charge la QoS (QSTA). EDCA définit quatre AC. Différents ACs ont des priorités différentes et desservent différents types de trafic.

2.6.1 Catégories d'accès (AC)

EDCA (Enhanced Distributed Channel Access) qui définit quatre catégories d'accès aux canaux ou priorités :

- ✓ une priorité à la voix.
- ✓ une priorité à la vidéo.
- ✓ une priorité dite "Best Effort" pour les applications standards.
- ✓ une priorité dite "Background" lorsque le trafic est faible.

Chaque trame de la couche supérieure arrive à la couche MAC avec une valeur de priorité. On distingue au total huit niveaux de priorités des utilisateurs (User Priority ou UP) allant de 0 à 7. La relation entre les UPs et ACs est montrée dans le Tableau 2.2.

Priorité	UP	AC	Désignation
Plus basse ↓ Plus haute	1	AC_BK	Background
	2	AC_BK	Background
	0	AC_BE	Best Effort
	3	AC_BE	Best Effort
	4	AC_VI	Vidéo
	5	AC_VI	Vidéo
	6	AC_VO	Voix
	7	AC_VO	Voix

Tableau 2.2: Relation entre les priorités et les catégories d'accès dans EDCA [9]

Chaque QSTA implémente quatre files d'attente de transmission, une pour chaque AC et quatre fonctions indépendantes (EDCAF), une pour chaque file d'attente, comme le montre la figure 2.3.

EDCAF représente une version améliorée de la méthode d'accès DCF, par l'introduction de la différenciation de service ; elle différencie l'accès au support via le principe des priorités où chaque priorité est associée à l'AC qui lui correspond.

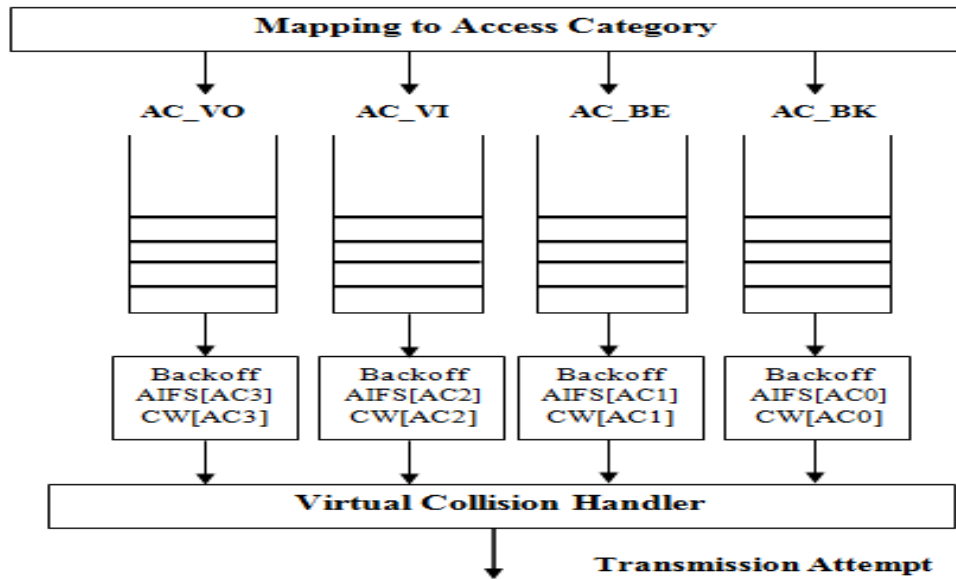


Figure 2.3: Quatre CA, chacun avec sa propre file d'attente et ses paramètres EDCA [9]

2.6.2 Les paramètres EDCA

Les paramètres suivants associés à un AC définissent la manière dont EDCAF recherche le support.

AIFS: L'intervalle minimum du temps entre le temps où le média sans fil devient libre et le début de la transmission d'une trame.

CW_{min}, CW_{max}: taille (en intervalles de temps) de la fenêtre de contention utilisée pour le mécanisme de backoff.

TXOP Limit: Durée de transmission maximale après que le support est pris.

Comme le montre le tableau 2.3, chaque AC possède ses propres paramètres et son propre comportement EDCA Function (EDCAF), les AC de priorité plus élevée (voix et vidéo) attendent une courte période d'AIFS, tandis que les AC de priorité inférieure (pour background et best effort) doivent attendre une plus longue période d'AIFS avant d'accéder au support. La taille du CW est différente, donc l'AC avec une priorité plus élevée sélectionnera la valeur backoff à partir du CW plus petit au lieu de l'AC avec une priorité plus faible.

Le dernier paramètre de différenciation TXOPLimit permet d'augmenter le débit des ACs, il est donc utile d'avoir une grande valeur de ce paramètre pour les applications gourmandes en bande passante. Ceci justifie le choix des TXOPLimit par défaut du standard dans le Tableau 2.3, ou l'AC_VI qui nécessite une bonne part de la bande passante se voit attribuer la plus grande valeur de TXOPLimit.

Fondamentalement, pour les AC à haute priorité, les AIFS, CWmin et CWmax ont des valeurs plus petites, tandis que la limite TXOP a des valeurs plus grandes.

Les valeurs des paramètres EDCA sont parfois appelées AIFS [AC], CWmin [AC], CWmax [AC] et TXOP Limit [AC].

AC	CWmin	CWmax	AIFSN	TXOP (ms)
AC_VO	$(aCWmin+1)/4 - 1$	$(aCWmin+1)/2 - 1$	2	1.504
AC_VI	$(aCWmin+1)/2 - 1$	aCWmin	2	3.008
AC_BE	aCWmin	aCWmax	3	0
AC_BK	aCWmin	aCWmax	7	0

Tableau 2.3: Valeurs standard des paramètres EDCA [8]

Fondamentalement, la principale différence entre DCF et EDCAF est que EDCAF utilise des valeurs de paramètres spécifiques pour chaque AIFS AC [AC], CWmin [AC] et CWmax [AC], tandis que DCF utilise des valeurs fixes DIFS, CWmin et CWmax. Dans ce qui suit, chaque paramètre EDCA et sa contribution à la différenciation des services sont brièvement décrits.

AIFS (Arbitration Interframe Space): La période minimale pendant laquelle le support doit être détecté au repos avant que QSTA ne commence la transmission ou l'interruption. Il s'agit d'une valeur variable et dépend de AC. Pour assurer la bonne différenciation, au lieu d'utiliser une valeur DIFS fixe, un AIFS est appliqué pour chaque AC selon l'équation suivante :

$$\text{AIFS} = \text{aSIFSTime} + \text{AIFSN} \times \text{aSlotTime}.$$

AIFSN est un nombre déterminé pour chaque AC, spécifique pour le paramètre AIFS, aSlotTime est la durée d'un slot de temps. La valeur minimale d'AIFSN est égale à 2 et sa valeur maximale est égale à 15. Les AC à haute priorité utilisent des valeurs AIFSN plus petites que les AC à basse priorité. Le tableau 2.3 montre les valeurs AIFSN par défaut pour les quatre AC.

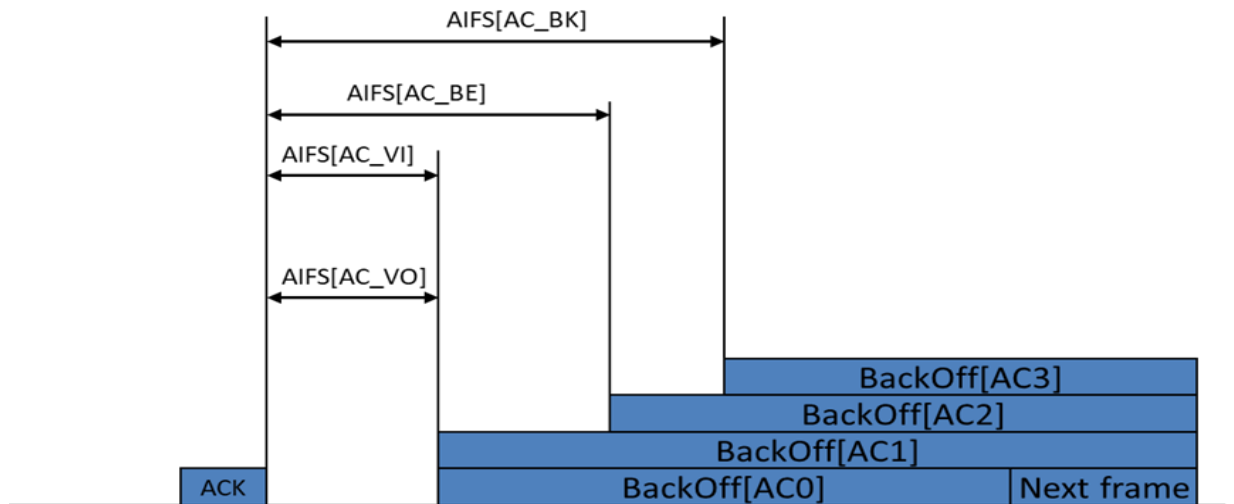


Figure 2.4: Priorité du trafic basée sur AIFS [17]

La figure 2.4 montre comment attribuer la priorité de transmission à différents AC en fonction de la période AIFS. Pour les AC de priorité plus élevée, une valeur AIFSN plus petite oblige leur EDCAF à attendre un temps plus court avant de pouvoir commencer

la transmission où le compte à rebours de son temporisateur d'interruption par rapport à l'EDCAF pour un AC de faible priorité. Ainsi, les AC plus prioritaires auront une plus grande part de la bande passante. De plus, la longueur AIFS plus courte garantit que les AC de priorité plus élevée ne subiront pas de retards plus longs, ce qui est une exigence critique pour les applications sensibles aux retards, comme décrit ci-dessus.

CWmin et CWmax: Les limites minimale et maximale de la fenêtre de collision dépendent de l'AC. Les AC de priorité supérieure ont des valeurs standard CWmin et CWmax plus petites tandis que les AC de priorité inférieure ont des valeurs plus grandes. Le tableau 2.3 montre les valeurs standard des paramètres CWmin et CWmax des quatre AC. Le tableau 2.4 répertorie les paramètres aCWmin et aCWmax spécifiques aux couches physiques OFDM et DSSS.

Paramètre	OFDM	DSSS
aCWmin	15	31
aCWmax	1023	1023

Tableau 2.4: Valeurs standard des paramètres CW pour différentes couches physiques [3]

Le tableau 2.3 montre les valeurs standard des paramètres CW pour différents AC. Une petite CW de AC amènera son EDCAF à choisir une petite valeur de backoff aléatoire, ce qui se traduira par une période de temps plus courte en plus de l'AIFS lorsque le support deviendra inactif. Il donne à CA une priorité plus élevée que CA avec une CW plus grande, ce qui se traduit par une valeur de backoff plus importante et un délai de retard plus long.

TXOP (Transmission Opportunity) : La limite TXOP est la période de temps que EDCAF peut transmettre après avoir accédé au support. Lorsque EDCAF obtient TXOP, tant que le temps de transmission ne dépasse pas la limite TXOP, il peut commencer à transmettre des trames. Cette limite de transmission couvre toute la séquence d'échange de trames, y compris les périodes SIFS et les accusés de réception (ACKs).

Le tableau 2.3 montre les valeurs limites TXOP par défaut pour différents AC. Aucune valeur nulle des limites TXOP n'indique que l'EDCAF peut transmettre plusieurs trames dans une opportunité de transmission, de sorte que la durée de transmission ne dépasse pas la limite TXOP. C'est ce qu'on appelle l'éclatement sans contention (CFB). Les transmissions de trames consécutives dans une opportunité de transmission sont séparées par des périodes de temps SIFS.

2.7 Le mécanisme EDCA dans IEEE 802.11e

EDCA s'appuie sur le concept de « catégories d'accès » (AC) et ses entités indépendantes de backoff. La station QoS (QSTA) implémente quatre AC. Chaque AC a un ensemble de paramètres EDCA associés. Ces paramètres incluent l'espace d'arbitrage entre les trames (AIFS), la fenêtre de contention (CW) et leurs valeurs minimale et maximale respectives CW_{min} et CW_{max} .

Chaque AC de chaque station est en concurrence avec d'autres AC pour obtenir une opportunité de transmission (TXOP). Pour ce faire, il démarrera indépendamment un temporisateur de coupure après avoir détecté que le canal est inactif pendant l'intervalle AIFS.

La période d'interruption (backoff) de chaque AC est sélectionnée en fonction de la distribution uniforme sur $[0, CW [AC]]$. Une valeur CW_{min} a été initialement affectée à la taille CW, et elle sera doublé lorsque la transmission a échoué jusqu'à CW_{max} . Plus

le CW_{min} [AC] est petit, plus le temps d'accès au canal de la priorité correspondante est court, donc pour une condition de trafic donnée, il y a de meilleures chances d'accéder au média. En fonction de chaque type de trafic généré (notamment audio, vidéo et texte), le trafic provenant d'une application est associé à certains paramètres de QoS.

2.8 Travaux connexes et motivations

La mobilité des utilisateurs et leurs besoins d'accès itinérant aux réseaux informatiques rendent les réseaux traditionnels (filaires) obsolètes. Pour surmonter certains défis qui n'existent pas dans les réseaux câblés traditionnels plusieurs recherches ont été faites par la communauté de la recherche sans fil [18] [12]. Parmi les problèmes les plus difficiles : l'attribution des fréquences, les interférences et la fiabilité, la sécurité, la consommation d'énergie, la sécurité humaine, la mobilité et le débit [19].

Ces dernières années, des recherches ont été activement menées pour fournir une prise en charge du trafic en temps réel afin de prendre en charge les applications multimédias dans les réseaux locaux sans fil [20] [21]. La plupart des recherches introduisent le concept de différenciation de services dans le standard pour la gestion de différentes priorités afin qu'elle soit en mesure de fournir la prise en charge de la qualité de service requise pour le trafic en temps réel que l'on trouve dans les applications multimédias (audio, vidéo et texte).

Aad et Castelluccia [22] proposent trois mécanismes de différenciation entre différents utilisateurs du standard. Le premier mécanisme porte sur la différenciation par CW, le deuxième sur la différenciation par DIFS et le dernier sur la différenciation par la taille maximale des trames permise à un utilisateur.

Devant la nécessité d'introduire la QoS au niveau de la couche MAC du standard 802.11, le groupe de travail 802.11 a publié en 2005, sous le nom de 802.11e [2], un amendement au standard 802.11. La nouvelle norme a introduit une technique EDCA basée sur la fenêtre de contention, cette dernière peut prendre en charge la QoS en introduisant une différenciation de service.

Plusieurs solutions [23] [24] ont été proposées par la communauté de recherche pour développer des solutions plus robustes pour la fourniture de la QoS dans le standard 802.11. La plupart d'entre eux sont basés sur une simulation logicielle. Dans [23] les auteurs ont présenté une étude de simulation sur le mécanisme de l'IEEE 802.11e Enhanced Distributed Coordination Function (EDCF).

Une étude des capacités de différenciation des mécanismes de QoS dans la norme 802.11e a été présentée dans [24]. Ces études ont examiné l'impact potentiel des schémas de différenciation des services dans l'EDCF en utilisant des outils de simulation pour mettre en œuvre les installations EDCF avec des files d'attente distinctes dédiées aux quatre catégories d'accès. Ensuite, ils ont comparé les performances des mécanismes MAC 802.11 et 802.11e. Cependant, les études existantes ont utilisé les valeurs exactes des paramètres de QoS explicitement spécifiées dans la norme 802.11e.

2.9 Résumé

Ce chapitre a décrit le DCF et comment il peut coordonner l'accès au support radio partagé. Après cela, il a présenté la procédure d'évitement de collision et d'interruption dans DCF et comment le DCF a défini le CW après chaque transmission réussie ainsi qu'après une collision. Ensuite, nous avons discuté de l'IEEE 802.11e, qui est une version améliorée de la norme IEEE 802.11 d'origine afin de prendre en charge la qualité de service (QoS). IEEE 802.11e prend en charge la QoS en introduisant un mécanisme de priorité qui prend en charge la différenciation des services en affectant le trafic de données avec différentes priorités en fonction de leurs exigences de QoS. Pour le trafic de données de priorités différentes, quatre catégories d'accès (AC) différentes ont été définies.

En outre, ce chapitre a fourni des détails sur l'EDCA et ses paramètres QoS importants, tels que: AIFS: la période de temps pendant laquelle le support doit être inactif avant le début de la transmission ou backoff est démarré; CWmin, CWmax: taille (dans les intervalles de temps) de la fenêtre de contention utilisée pour backoff; TXOP Limit: La durée maximale de la transmission après que le support est pris. Le chapitre suivant abordera les études de simulation, l'outil utilisé et les paramètres de simulation.

Chapitre 3 Impact des paramètres de différenciation QoS sur les performances EDCA

3.1 Introduction

La simulation permet à l'utilisateur de définir un réseau et de simuler des communications entre les nœuds de ce réseau, de tester à moindre coût les nouveaux protocoles, d'anticiper les problèmes qui pourront se poser dans le futur et d'implémenter la technologie la mieux adaptée aux besoins.

Pour étudier et évaluer les performances de la norme 802.11e, nous utilisons un simulateur, puisque les tests réels ne sont pas possibles. Beaucoup de simulateurs de réseaux sans fil ont été développés comme OPNET, OMNET, NS2, J-SIM ...etc. Nous avons choisi le logiciel NS2 car c'est un très bon outil dans le domaine de la recherche, gratuit, et contient beaucoup de protocoles dans des différentes couches. Il est extensible car il permet facilement d'ajouter de nouveaux protocoles.

Ce chapitre est composé de deux parties, dans la première partie nous allons présenter le simulateur utilisé pour cette simulation, qui est le NETWORK SIMULATOR2 ou NS2. Dans la deuxième partie nous évaluerons l'impact de deux paramètres de différenciation de la QoS (CWmin et AIFS) sur les performances EDCA. Cette partie présente les paramètres du scénario de simulation et les résultats pour les réseaux ad hoc. Pour implémenter cette simulation, on a développé notre propre script OTCL de différenciation [Annexe A].

Le reste de ce chapitre est organisé comme suit. La section 3.3.1 décrit le scénario qui sera utilisé pour évaluer l'impact des paramètres EDCA sur les performances du réseau au moyen d'une analyse de débit. Dans la section 3.3.2 on présente la topologie du réseau

pour le scénario. La section 3.4 présente les résultats des simulations du scénario avec les discussions et l'analyse des résultats. Enfin, la section 3.5 conclut ce chapitre.

3.2 Présentation du simulateur NS2

NS2, Network Simulator, est un logiciel de simulation open source gratuit (libre) à évènements discrets probablement le plus utilisé par la communauté scientifique et universitaire pour simuler tout ce qui se rapporte aux réseaux. Il est développé dans le cadre du projet VINT qui regroupe plusieurs laboratoires de recherche comme AT&T institut de recherche à Berkeley (ACIRI), Xerox PARC et Sun Microsystems. Ce projet est soutenu par DARPA3 (Defense Advanced Projects Agency) Ou (agence avancée de projets de la défense).

NS 2 est un outil de recherche très utile pour la compréhension des protocoles. Il sert aussi bien dans l'étude des protocoles de routage qu'à l'étude des réseaux mobiles ou les communications par satellites. Il permet à l'utilisateur de définir un réseau et de simuler les communications entre nœuds. Le simulateur utilise le langage orienté objet OTCL (Object Tools Command Language) pour la description des conditions de simulation sous forme de script. Un script de simulation doit être écrit en langage OTCL, dont un extrait a été donné ci-dessous et va lancer l'exécution des différents scénarios. Les résultats sont collectés dans un fichier trace au format texte. Il est tout à fait possible de créer un nouveau protocole ou de nouveaux objets par l'intermédiaire des langages C++ et OTCL [25] [26].

Par ailleurs le simulateur permet la création d'un fichier d'animation (d'extension.tr), permettant de visualiser la simulation sur l'interface graphique NAM. Ce visualisateur fournit une représentation du graphe du réseau sur laquelle on peut voir les paquets circuler, suivre le niveau des files d'attente et observer le débit courant des liaisons [25].

3.2.1 Principe de base

L'application NS se compose de deux éléments fonctionnels :

- **Un interpréteur** : au moyen de l'interpréteur l'utilisation est capable de créer le modèle de simulation ce qui revient à assembler les différents composants nécessaires à l'étude. Les composants du modèle de simulation sont appelés (objets) ou encore (instances de classe).

- **Le moteur de simulation** : effectue les calculs applicables au modèle préalablement construit par l'utilisateur via l'interpréteur.

NS bénéficie de toutes les possibilités qu'offrent les techniques objets comme l'héritage, le polymorphisme, la surcharge, etc. l'héritage permet d'élaborer des arborescences de classes.

3.2.2 Visualisation des résultats

La visualisation des résultats est à la base de deux fichiers, Trace et Nam.

➤ **Fichier trace :**

Les fichiers de trace contiennent les traces d'événement pouvant être traitées ultérieurement pour comprendre les performances du réseau. L'exemple ci-dessous décrit le processus général de la manière dont une simulation de réseau est effectuée sous NS-2. Les fichiers de sortie tels que les fichiers de trace doivent être analysés pour extraire des informations utiles. L'analyse peut être effectuée à l'aide de la commande awk (sous UNIX et LINUX ou le script perl) [26]. En traitant ce fichier de très grande taille on va pouvoir en extraire l'information souhaitée : calcul de débit aux nœuds, évaluation des pertes, ...etc.

➤ **Fichier Interface graphique Nam (Network Animator) :**

NS-2 ne permet pas de visualiser le résultat des expérimentations. Il permet uniquement de stocker une trace de la simulation, de sorte qu'elle puisse être exploitée par un autre logiciel, comme NAM. Le Nam est un outil d'animation basé sur Tcl/Tk, utilisé dans NS afin de visualiser le tracé de simulation des réseaux, ainsi que les tracés de données. Le modèle théorique du NAM a été non seulement créé pour lire un large ensemble de données d'animation, mais aussi suffisamment extensible pour être utilisé quel que soit le type de réseau simulé (fixe, mobile ou mixte). Ce qui permet de visualiser tout type de situation possible.

Par exemple, il est capable de représenter des paquets TCP ou UDP, la rupture d'un lien entre nœuds, ou encore de représenter les paquets rejetés d'une file d'attente pleine. Ce logiciel est souvent appelé directement depuis les scripts TCL de NS2, pour visualiser directement le résultat de la simulation. Le modèle théorique du Nam a été non seulement créé pour lire un large ensemble de données d'animation, mais aussi suffisamment extensible pour être utilisé quel que soit le type de réseau simulé (fixe, mobile ou mixte), ce qui permet de visualiser tout type de situation possible [26].

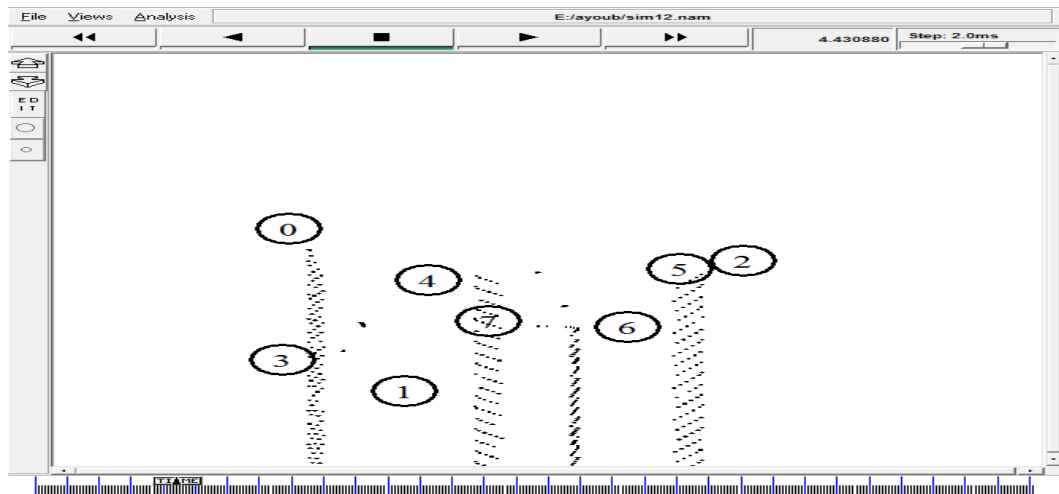


Figure 3.1: L'interface NAM [26]

➤ **L'interface XGRAPH :**

C'est un outil qui fait la visualisation des données contenue dans les fichiers TRACE sous forme de graphes.

3.3 Impact des paramètres de différenciation QoS sur les performances EDCA

Nous allons évaluer deux paramètres de différenciation de la QoS d'impact, c'est-à-dire CWmin et CWmax sur les performances EDCA.

3.3.1 Scénario : Différenciation par l'un des quatre paramètres EDCA QoS

Dans ce scénario, nous étudions la différenciation par un seul paramètre dans un réseau QIBSS. Le nombre de stations actives reste fixe pendant une seule simulation. Chaque station génère du trafic et il est supposé que le trafic est toujours disponible pour la transmission. La section 3.3.2 décrit la configuration de la topologie de réseau simulée.

3.3.2 Topologie de réseau pour le scénario

La figure 3.2 montre la topologie d'un QIBSS qui pourrait être représentative du réseau de notre scénario. Le réseau se compose de 4 stations, chaque station utilise seulement 1 AC, et tous les paramètres sont définis selon la norme 802.11b, dans ce scénario, le débit et le délai par valeur des paramètres EDCA sont obtenus. La comparaison des débits et délais est également étudié.

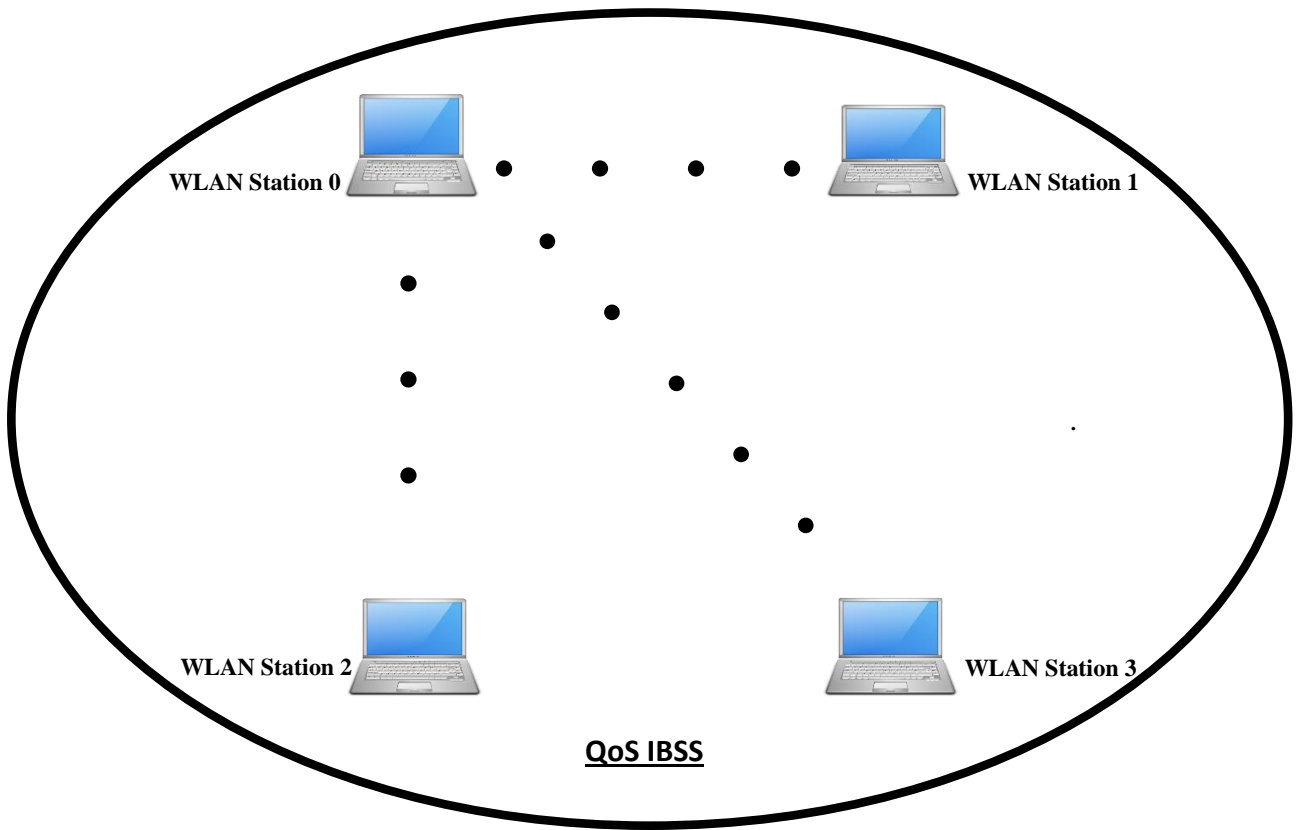


Figure 3.2: La topologie du scénario

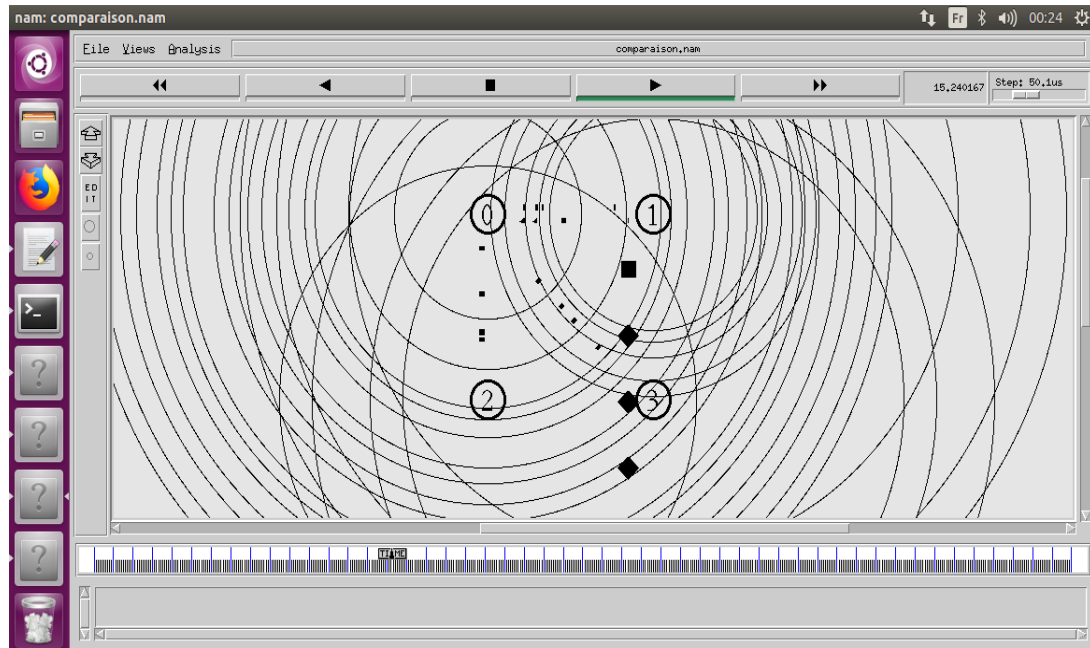


Figure 3.3: Fichier NAM de la topologie du scénario

3.3.3 Configuration de la simulation

Dans cette section, nous utilisons le simulateur de réseau 2 (NS 2) pour étudier l'impact des paramètres de différenciation QoS sur les performances EDCA, dans notre simulation nous étudions la différenciation par un seul trafic (VoIP) dans un réseau QIBSS, nous avons considéré un seul scénario, dans ce scénario, toutes les stations émettent vers la même destination, chaque station génère du trafic et il est supposé que le trafic est toujours disponible pour la transmission, le scénario se compose d'une seule connexion (flux) VoIP. Conformément aux spécifications 802.11e, le trafic VoIP est acheminé sous AC3. Dans notre scénario le trafic VoIP commence à 5 secondes.

3.3.4 Les paramètres de simulation

Dans notre scénario de simulation, nous allons utiliser les valeurs par défaut pour les paramètres EDCA QoS importants ainsi que différents paramètres. Ces paramètres sont présentés dans le tableaux 3.1 selon à la norme IEEE 802.11e [3], nous choisissons 802.11b comme couche PHY et le débit de données PHY est fixé à 11 Mb / s. le tableau 3.1 montre les valeurs par défaut des paramètres d'accès aux canaux définis dans EDCA pour les quatres AC (VO = voix, VI = vidéo, BE = meilleur effort, BK = arrière-plan), les paramètres de simulation sont indiqués dans le tableau 3.2.

AC	CWmin	CWmax	AIFSN	TXOP (ms)
AC_VO	$(aCW_{min}+1)/4 - 1$	$(aCW_{min}+1)/2 - 1$	2	1.504
AC_VI	$(aCW_{min}+1)/2 - 1$	aCWmin	2	3.008
AC_BE	aCWmin	aCWmax	3	0
AC_BK	aCWmin	aCWmax	7	0

Tableau 3.1: Paramètres EDCA par défaut [8]

	Voice	Video	Best Effort	Background
Transport protocol	UDP	UDP	UDP	UDP
AC	VO	VI	BE	BK
CWmin	7	15	31	31
CWmax	15	31	1023	1023
AIFSN	2	2	3	7
Taille du paquet (Packet Size)	500 bytes	500 bytes	200 bytes	200 bytes
Taux d'envoi (Sending rate)	512 kb/s	1024 kb/s	256 kb/s	128 kb/s

Tableau 3.2: Paramètres EDCA pour la simulation [27]

3.4 Résultats des simulations et analyse pour le scénario

Cette section contient deux parties et chaque partie présente les résultats de simulation de la différenciation par l'un des quatre paramètres EDCA QoS, ces résultats sont à la base de l'exécution du script OTCL [Annexe A].

3.4.1 Différenciation par CWmin

Le paramètre CWmin détermine la taille de l'intervalle à partir duquel le compteur d'interruption est tiré. Un CWmin plus petit entraîne un compteur de coupure plus petit et une station atteindra 0 plus rapidement. Parce que les stations ont toujours des paquets à transmettre, les stations avec un CWmin plus petit peuvent transmettre plus de paquets. La figure 3.4 présente les résultats de simulation de débit effectif en ko par rapport au temps pour les deux valeurs CWmin = 7 et CWmin = 15.

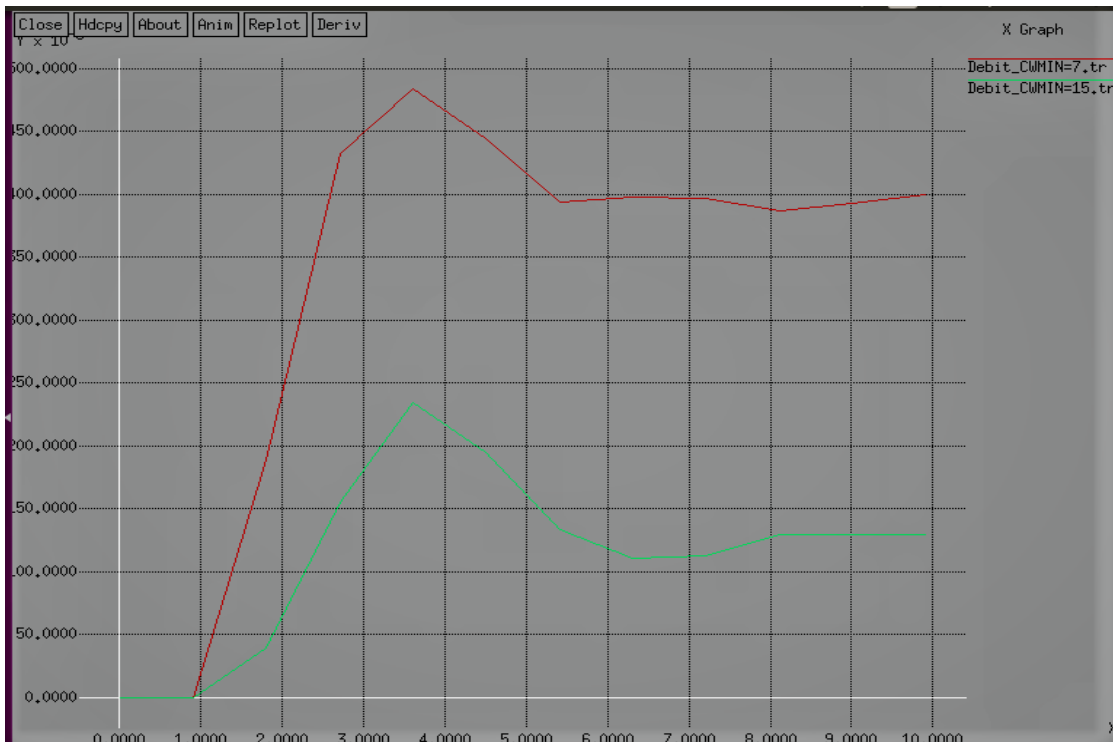


Figure 3.4: Débit EDCA, Paramètres CWmin= 7 et CWmin= 15

La figure 3.5 présente les résultats de simulation de délai en ko par rapport au temps pour les deux valeurs $CW_{min} = 7$ et $CW_{min} = 15$.

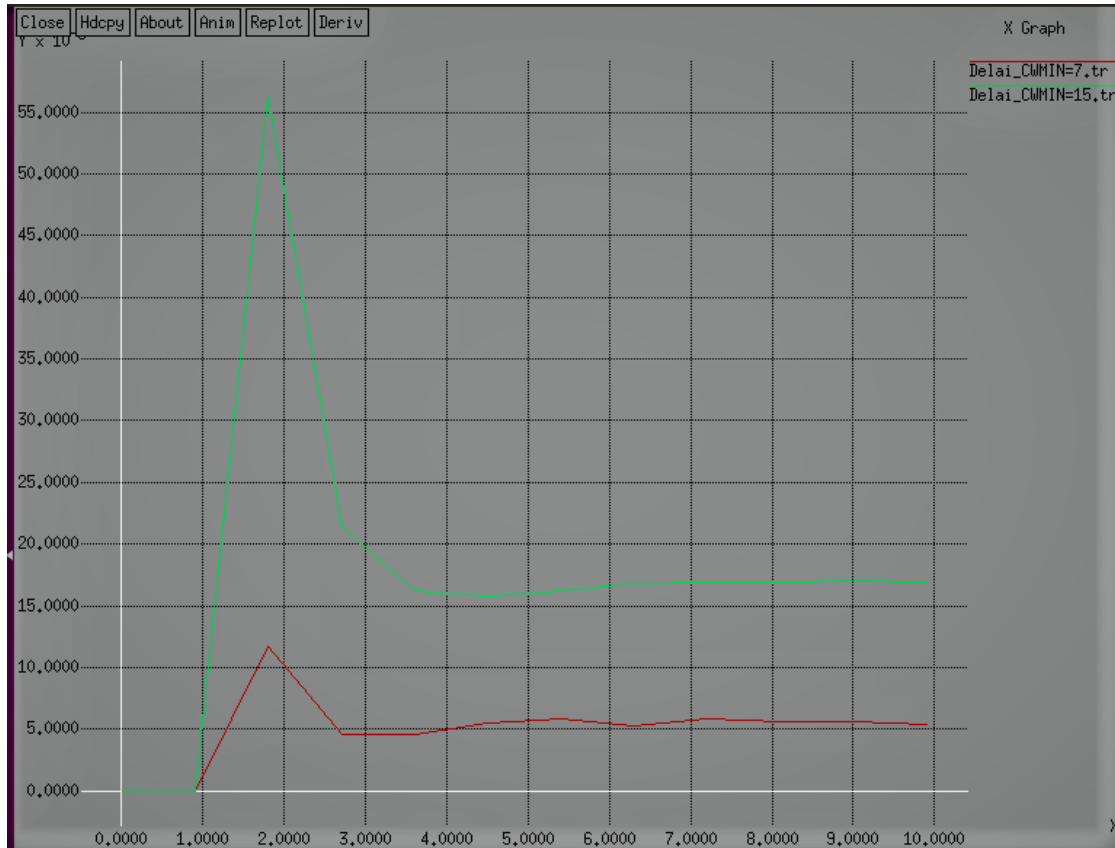


Figure 3.5: Délai EDCA, Paramètres $CW_{min}= 7$ et $CW_{min}= 15$

L'analyse de la figure 3.4 montre bien que le débit fourni par l'EDCA_802.11e avec CW_{min} plus petite est plus élevé que celui fourni lorsque CW_{min} a une valeur grande, autrement dit que pour de faibles valeurs du CW_{min} , la performance de débit d'EDCA s'améliore (La performance de débit lorsque $CW_{min} = 7$ est environ le double de la performance de débit correspondante à $CW_{min} = 15$).

On constate en effet que les stations avec une valeur CW_{min} plus petite obtiennent une plus grande part de la capacité du canal que les stations avec une valeur CW_{min} grande.

En général, les performances de débit de l'EDCA prouvent de meilleurs résultats de performance comme le montre la figure 3.4, au fur et à mesure que la valeur du CW_{min} diminue, le débit de l'EDCA augmente considérablement.

L'analyse de la figure 3.5 montre bien que l'augmentation de la valeur CW_{min} à 15 a un impact négatif sur le délai du trafic, on constate aussi que la courbe du délai de l'EDCA avec $CW_{min}= 15$ s'accroît d'une façon brusque jusqu'à atteindre sa valeur maximale, par rapport à la courbe avec $CW_{min}= 7$. D'après ce résultat, on constate que l'algorithme EDCA_802.11e avec valeur CW_{min} plus petite est le mieux adapté afin d'avoir une qualité de service en s'appuyant sur le paramètre de délai de bout en bout.

La différenciation CW_{min} utilisée dans EDCA consiste à modifier la quantité de TXOP fournie à chaque classe de trafic. Une station avec une valeur CW inférieure réduira le temps moyen nécessaire pour livrer avec succès un paquet et bénéficiera ainsi de performances améliorées par rapport aux stations avec des valeurs CW plus élevées (Une petite valeur CW_{min} a un effet positif sur la performance, cela est dû à des courtes périodes d'attente).

3.4.2 Différentiation par CWmax

Le paramètre CWmax n'est efficace que si une station est impliquée dans une séquence de collisions et que CW a grandi comme CWmax; CW ne doit pas être doublé pour les retransmissions suivantes.

La figure 3.6 présente les résultats de simulation de débit effectif en ko par rapport au temps pour les deux valeurs CWmax = 31 et CWmax = 1023.

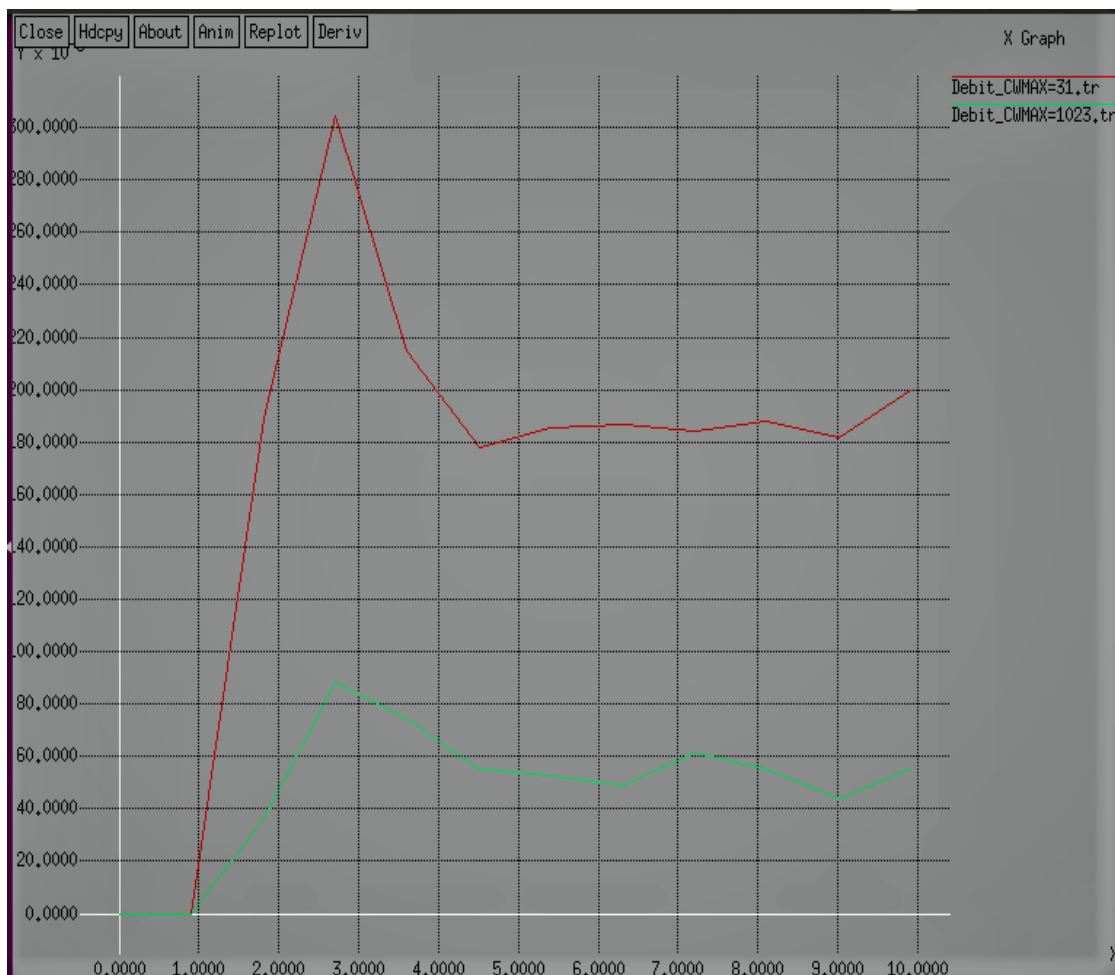


Figure 3.6: Débit EDCA, Paramètres CWmax = 31 et CWmax = 1023

La figure 3.7 présente les résultats de simulation de délai en ko par rapport au temps pour les deux valeurs $CW_{max} = 31$ et $CW_{max} = 1023$.



Figure 3.7: Délai EDCA, Paramètres $CW_{max} = 31$ et $CW_{max} = 1023$

L'analyse de la figure 3.6 montre bien que le débit fourni par l'EDCA_802.11e avec CW_{max} plus petite est plus élevé que celui fourni lorsque CW_{max} a une valeur grande, autrement dit que pour de faibles valeurs du CW_{max} , la performance de débit d'EDCA s'améliore (La performance de débit lorsque $CW_{max} = 31$ est environ le double de la performance de débit correspondante à $CW_{max} = 1023$).

En général, les performances de débit de l'EDCA prouvent de meilleurs résultats de performance comme le montre la figure 3.6, au fur et à mesure que la valeur du CW_{max} diminue, le débit de l'EDCA augmente considérablement.

L'analyse de la figure 3.7 montre bien que l'augmentation de la valeur CW_{max} à 1023 a un impact négatif sur le délai du trafic, on constate aussi que la courbe du délai de l'EDCA avec $CW_{max}= 1023$ s'accroît d'une façon brusque jusqu'à atteindre sa valeur maximale, par rapport à la courbe avec $CW_{max}= 31$.

D'après ce résultat, on constate que l'algorithme EDCA_802.11e avec valeur CW_{max} plus petite est le mieux adapté afin d'avoir une qualité de service en s'appuyant sur le paramètre de délai de bout en bout.

En résumé, nous pouvons dire que les capacités de différenciation de CW_{max} apparaissent lorsque de nombreuses collisions se produisent. CW_{max} ne fournit pas un rapport constant de partage de capacité, par exemple, comme le fait CW_{min} , mais il peut affamer les AC avec des valeurs plus élevées de CW_{max} . Les capacités de différenciation se font également au détriment d'une efficacité moyenne.

3.5 Conclusions

Dans ce chapitre, nous avons étudié la différenciation par un seul trafic (VoIP) dans un réseau QIBSS.

Les résultats ont révélé que les paramètres EDCA peuvent fournir certaines exigences de QoS. Le paramètre CWmin détermine la taille de l'intervalle à partir duquel le compteur d'interruptions est tiré. Un CWmin plus petit entraîne un compteur de backoff plus petit et une station atteint 0 plus rapidement, parce que les stations ont toujours des paquets à transmettre. Les stations avec un CWmin plus petit peuvent transmettre plus de paquets.

On voit en effet que les stations avec une valeur CWmin plus petite obtiennent une plus grande part de la capacité du canal que les autres stations. Une conclusion importante est que le paramètre CWmin n'affaiblit pas les autres AC, les autres stations peuvent toujours obtenir l'accès au canal.

Les capacités de différenciation dues au paramètre CWmax deviennent perceptibles lorsque de nombreuses collisions se produisent. CWmax ne fournit pas un rapport constant de partage de capacité, par exemple, comme le fait CWmin, mais il peut affamer les AC avec des valeurs plus élevées de CWmax.

Les capacités de différenciation se font également au détriment d'une efficacité moyenne. Les paramètres CWmin et CWmax se différencient bien, mais pour des charges élevées, les capacités de différenciation deviennent coûteuses au détriment d'une efficacité moyenne.

Chapitre 4 Etude de comparaison entre les protocoles IEEE 802.11e et IEEE 802.11

4.1 Introduction

Ce chapitre présentera l'évaluation des performances sous trafic audio et vidéo en temps réel grâce à des simulations utilisant Network Simulator-2 (NS 2), des paramètres tels que le délai moyen, le débit sont calculés et des graphiques ont été tracés, pour implémenter les scénarios de simulation de cette évaluation, on a développé notre propre script OTCL de comparaison [Annexe B].

Le chapitre est organisé comme suit. La section 4.2 décrit la configuration de notre simulation. La section 4.3 décrit les paramètres de simulation. La section 4.4 présente la comparaison entre DCF et EDCA, où nous comparons le DCF et le mécanisme EDCA en simulant le scénario 2. Dans la section 4.5 nous allons procéder à une analyse détaillée de la simulation des trois scénarios. Enfin, la section 4.6 conclut le chapitre.

4.2 Configuration de la simulation

Dans cette section, nous utilisons le simulateur de réseau NS2 pour évaluer les performances du mécanisme EDCA IEEE 802.11e, pour la topologie du réseau, notre simulation se base sur la même topologie utilisée dans le chapitre 3.

Dans notre simulation, nous avons considéré trois scénarios, à savoir le scénario 1, le scénario 2 et le scénario 3, dans chaque scénario, toutes les stations émettent vers la même destination. Le scénario 1 se compose de deux connexions (flux) VoIP, une connexion vidéo, deux connexions pour le trafic d'arrière-plan, et deux connexions pour les données de meilleur effort, dans le scénario 2, nous avons augmenté le nombre de connexions VoIP à sept, en conservant les autres connexions intactes, dans le scénario3,

nous avons augmenté le nombre de connexions BE / BK à quatre chacune, en conservant les autres connexions comme dans le scénario 1. Les trafics au mieux et en arrière-plan ont été créés avec un taux d'envoi moyen de 256 kb/s et 128 kb/s, respectivement.

Conformément aux spécifications 802.11e, le trafic VoIP est acheminé sous AC3, la vidéo sous AC2, les données au mieux sous AC1 et le trafic d'arrière-plan sous AC0. Dans tous les scénarios, le trafic vidéo commence à 5s, le trafic VoIP commence à 5 secondes et BK / BE le trafic commence à 15 secondes.

4.3 Les paramètres de simulation

Tous les paramètres MAC, y compris SIFS, DIFS, Slot Time, CWmin et CWmax dépendent de la couche physique sous-jacente (PHY), nous choisissons 802.11b comme couche PHY en utilisant ses caractéristiques physiques dans nos scénarios de simulation, le débit de données PHY est fixé à 11 Mb / s. Le tableau 4.1 montre ces valeurs pour IEEE 802.11b PHY [28]. DIFS est déterminé par $SIFS + 2 \times SlotTime$, quel que soit le PHY.

Paramètres	SIFS	DIFS	Slot Time	CWmin	CWmax
PHY 802.11b	10	50	20	31	1023

Tableau 4.1: Paramètres MAC pour la couche PHY 802.11b [28]

Dans notre scénario de simulation, nous allons utiliser les valeurs par défaut pour les paramètres EDCA QoS importants ainsi que différents paramètres. Ces paramètres sont présentés dans le tableaux 3.1 du chapitre 3, selon à la norme IEEE 802.11e [25], les paramètres EDCA de simulation sont indiqués dans le tableau 3.2 du chapitre 3.

4.4 Comparaison DCF et EDCA

Nous comparons le DCF et le mécanisme EDCA en simulant le scénario 2, avec sept connexions VoIP, une connexion vidéo et deux connexions BK / BE chacune.

En comparant les Figure 4.1 et Figure 4.2 qui tracent le débit de chaque type de trafic, nous observons que les débits des trafics VoIP, vidéo et BE sont significativement différents pour le DCF et l'EDCA alors que le débit du trafic BK est capable de maintenir son débit dans les deux cas. En figure 4.1, nous pouvons observer que le débit du trafic vidéo passe d'environ 900 kbps à 400 kbps, confirmant que le trafic vidéo est bien servi avec l'EDCA alors que de nombreuses images vidéo sont abandonnées avec le DCF. On peut également voir que le débit du trafic VoIP est faible en DCF par rapport à EDCA.

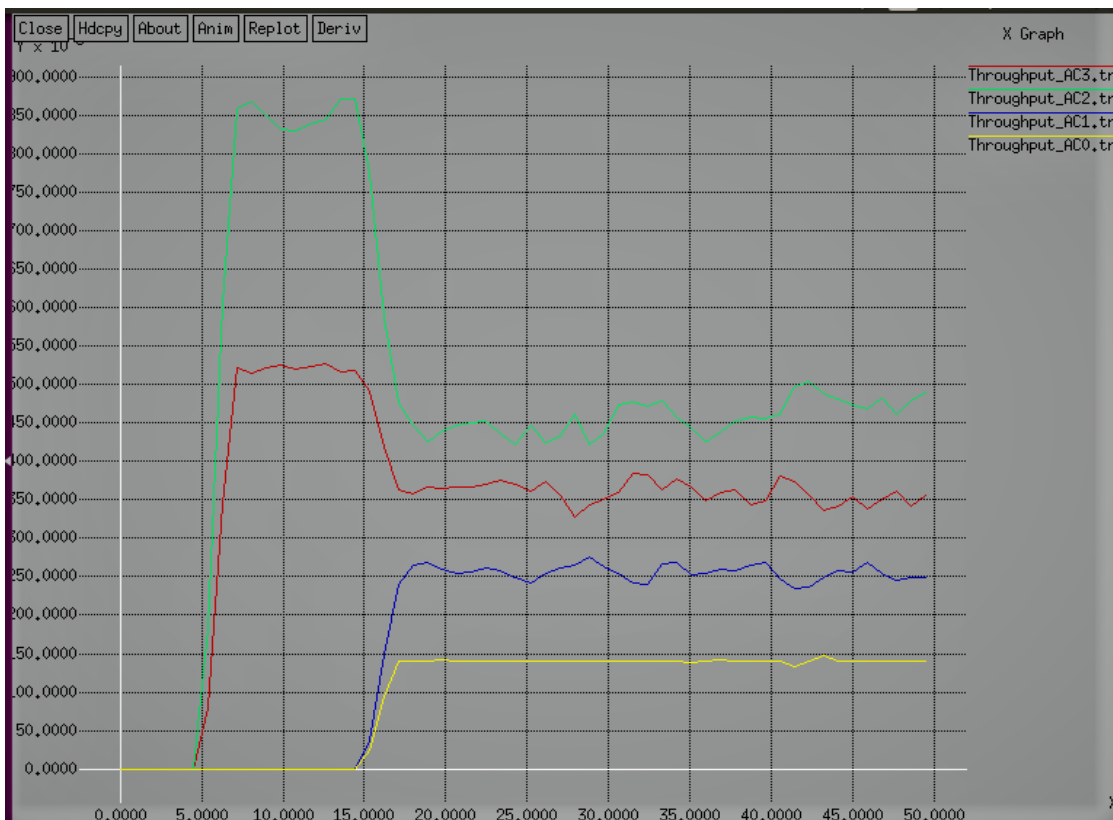


Figure 4.1: Débit DCF scénario2

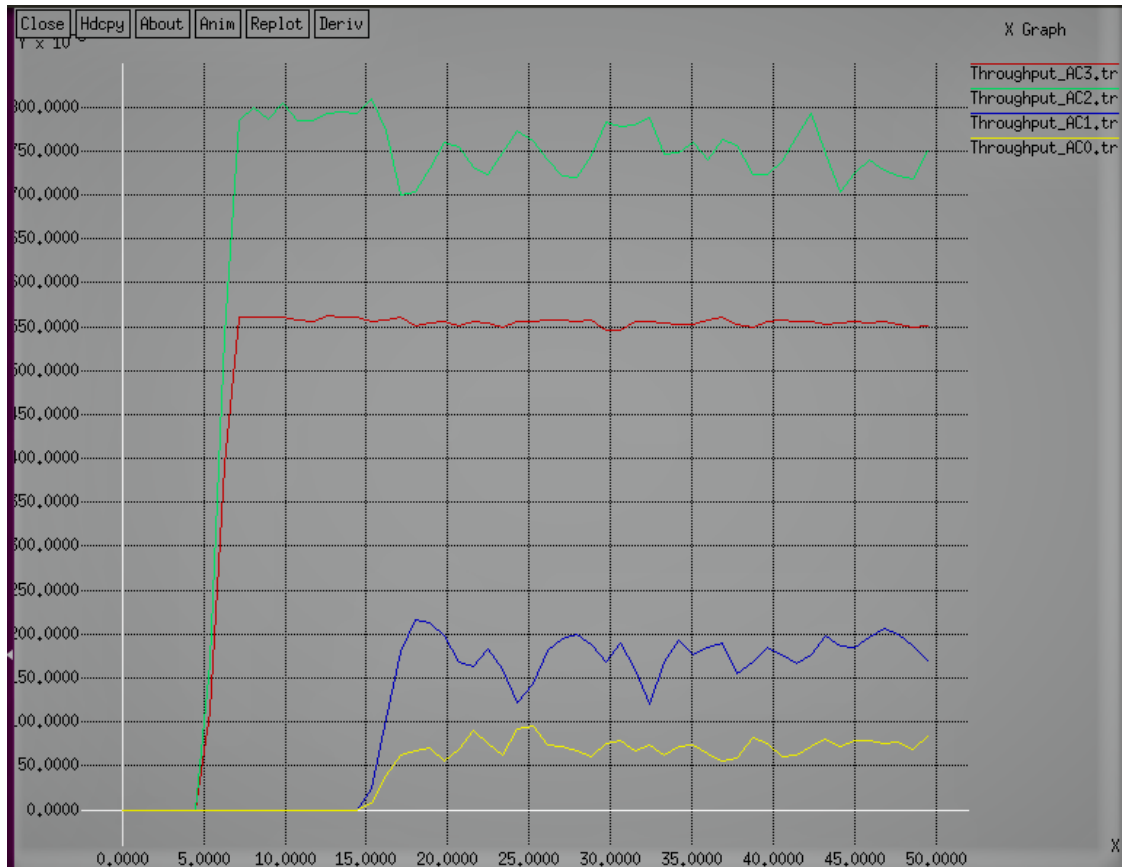


Figure 4.2: Débit EDCA scénario2

En figure 4.3 et figure 4.4, nous observons que les performances VoIP sont considérablement améliorées via EDCA. Notez qu'avec le DCF, le délai de trame vocale atteint parfois 60 ms, ce qui n'est pas acceptable dans la plupart des cas. On peut également voir le retard pour le trafic BE est également élevé dans le DCF par rapport à EDCA.

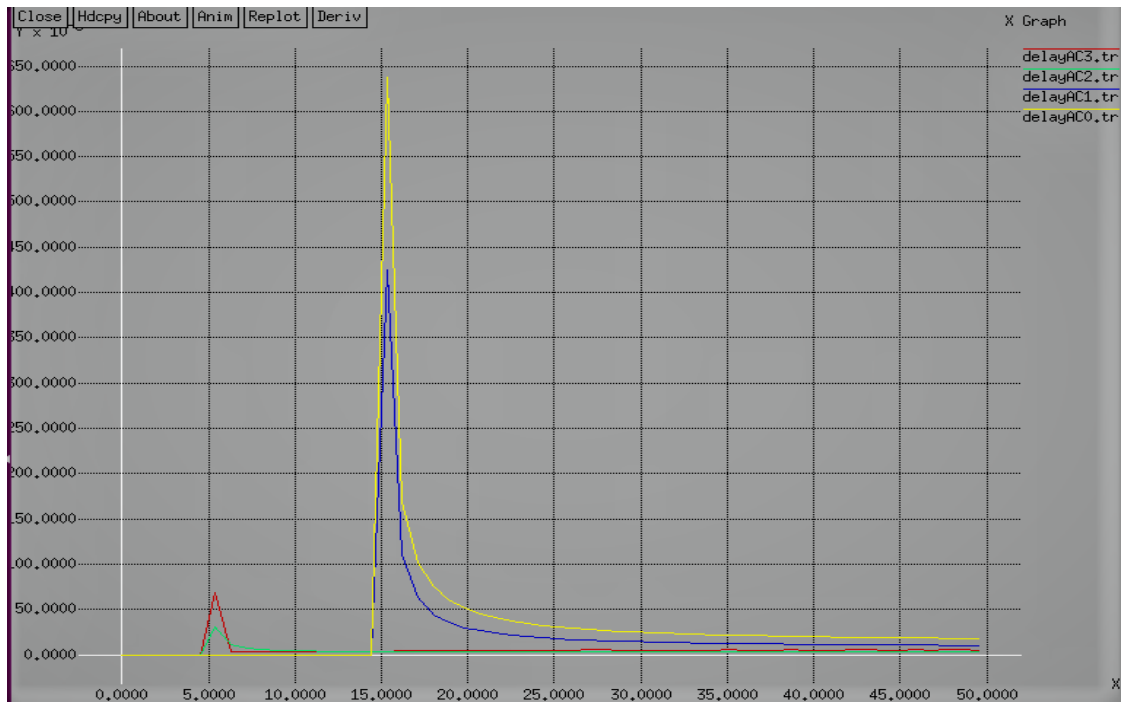


Figure 4.3: Délai DCF scénario2

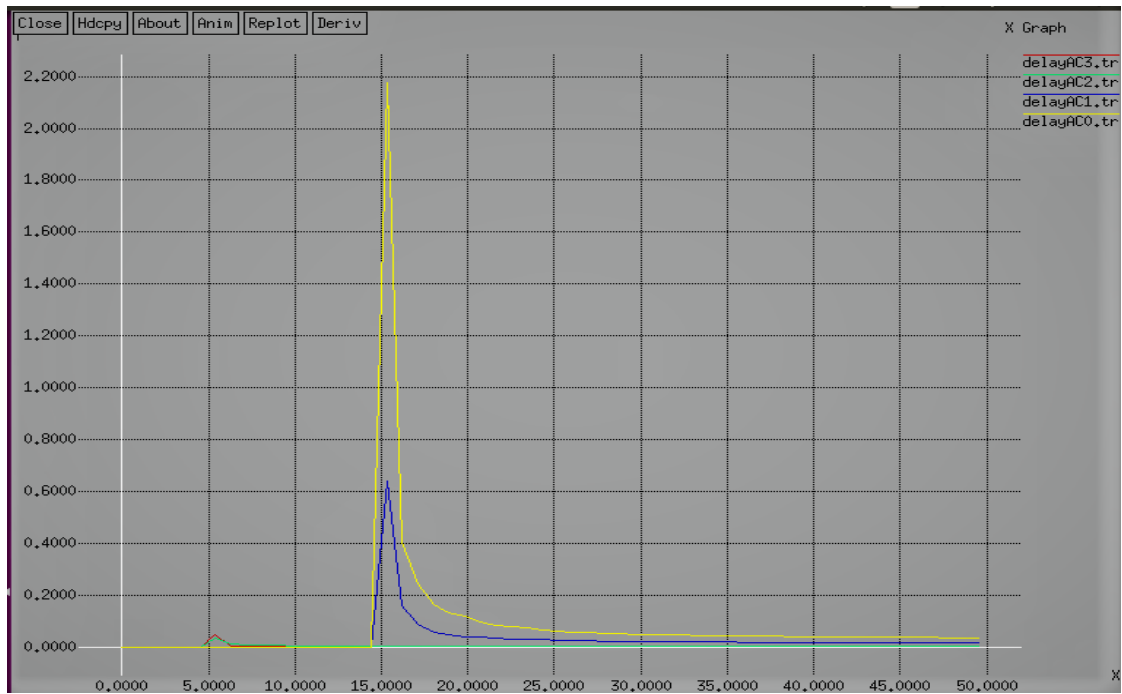


Figure 4.4: Délai EDCA scénario2

Ces résultats de simulation montrent qu'il n'y a pas de différenciation de service entre les différents types de flux de trafic en DCF, ce qui pose un problème de QoS pour les applications multimédias lorsque la charge de trafic est élevée. Le mécanisme EDCA fournit un accès différencié aux canaux pour différents types de trafic et nous nous attendons à ce que l'EDCA puisse prendre en charge des applications en temps réel avec un trafic vocal et vidéo avec une qualité de service raisonnable.

4.5 Analyse de simulation de l'EDCA

Tout d'abord, nous considérons le scénario 1, composé de deux connexions VoIP, une connexion vidéo et deux connexions chacune de trafic de fond et de données au mieux. Comme mentionné ci-dessus, les applications sont démarrées à des moments différents afin d'illustrer l'impact des flux de trafic supplémentaires sur la charge existante. La figure 4.5 montre les performances de retard de ces flux de trafic. Le délai pour les images vidéo est petit (environ 2 ms) de 0 à 5 s, car il s'agit du deux seuls trafic du réseau afin qu'il n'ait pas à affronter le canal avec d'autres sources. On peut observer à partir de la figure 4.5 que lorsque le trafic BK / BE est démarré à 15 secondes, le délai pour la vidéo et la VoIP n'a pas augmenté de manière significative.

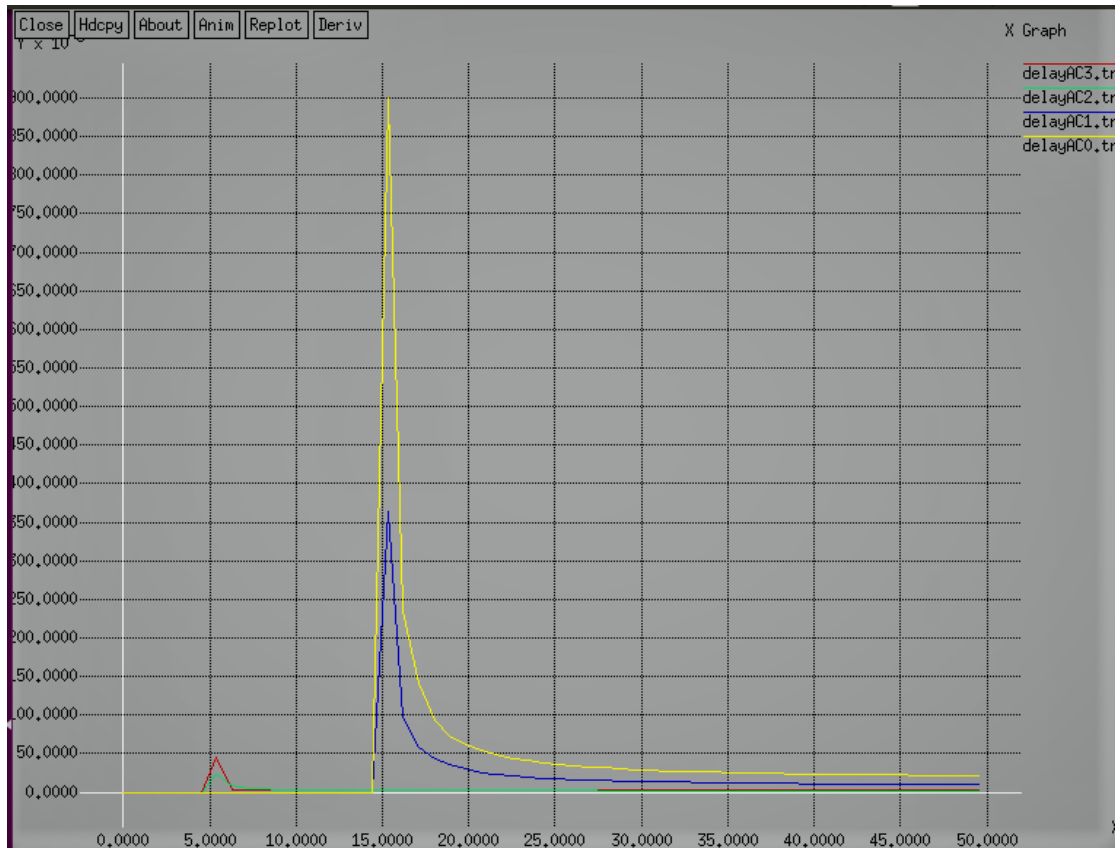


Figure 4.5: Délai EDCA scénario1

Ensuite, nous simulons le scénario 2, dans lequel nous augmentons le nombre de connexions VoIP à sept. Sur la figure 4.6, l'impact de l'augmentation des connexions VoIP de priorité la plus élevée peut être vu sur les performances de retard du trafic de faible priorité. Lorsque tous les flux de trafic sont présents, le délai pour le trafic BK / BE monte à 220 ms et 60 ms contre 90 ms et 35 ms dans le scénario 1. Le trafic prioritaire peut être vu sur les performances de retard du trafic de priorité inférieure.

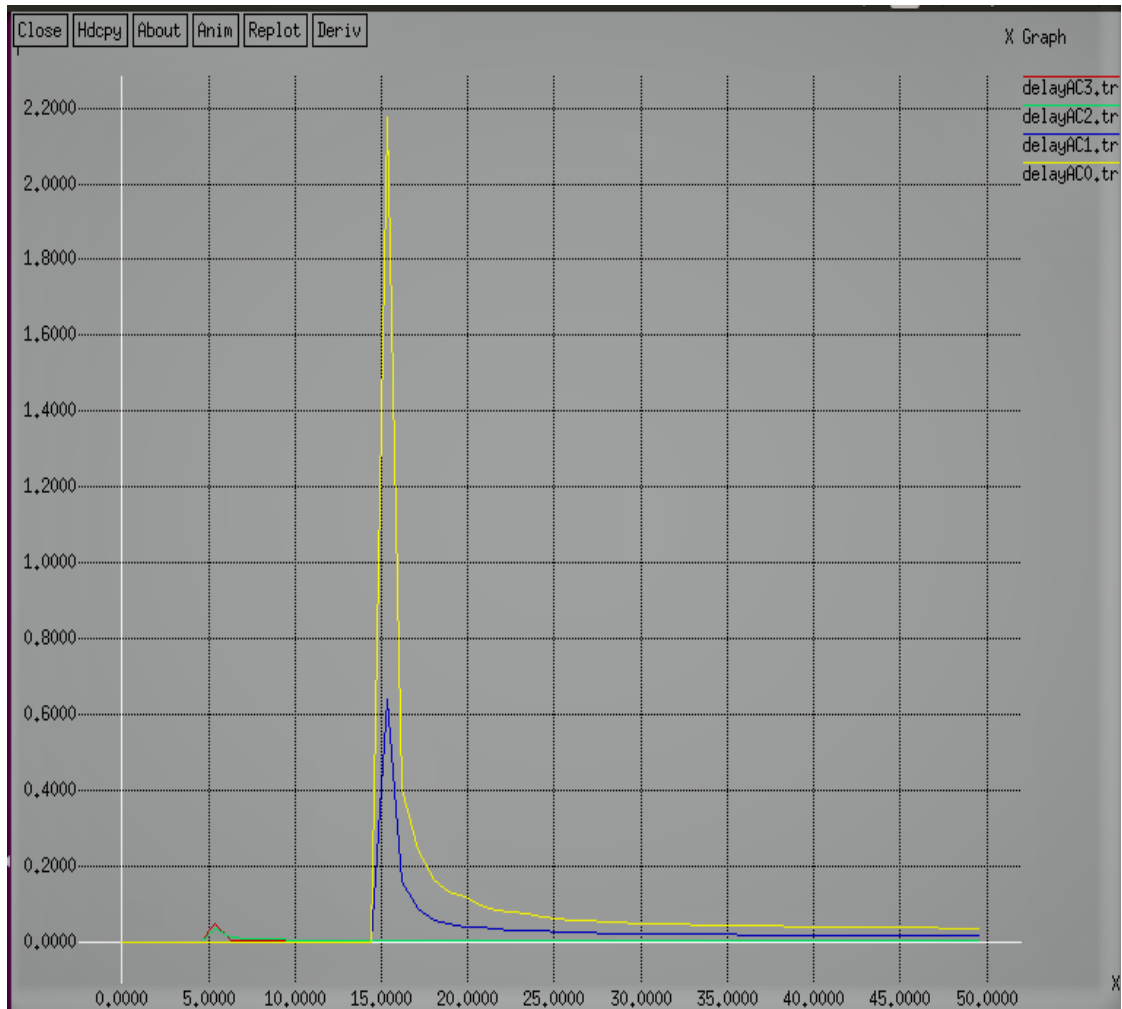


Figure 4.6: Délai EDCA scénario2

Dans le scénario 3, nous augmentons le nombre de trafic en arrière-plan et les connexions de données au mieux à quatre. En figure 4.7 nous observons que l'augmentation du trafic de faible priorité n'a pas d'impact négatif sur le retard du trafic de priorité plus élevée. On peut voir que le délai pour le trafic VoIP et vidéo est presque le même pour le trafic BK / BE faible et le trafic BK / BE élevé. Par rapport aux augmentations de charge VoIP, les augmentations de charge BK et BE n'affectent pas le retard vidéo de la figure 4.7 autant que celui de la figure 4.6, en grande partie en raison du courant alternatif plus élevé utilisé par le trafic vidéo que BE et BK.

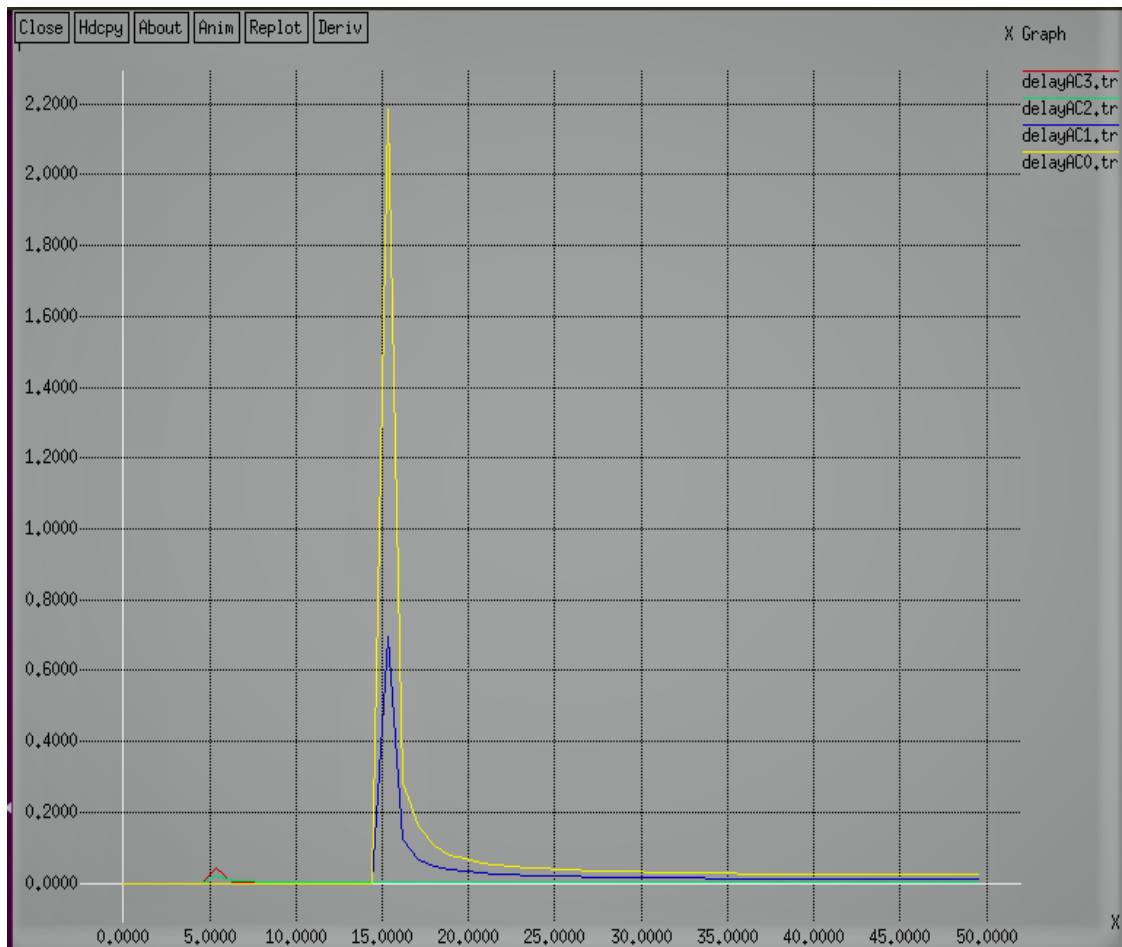


Figure 4.7: Délai EDCA scénario3

La figure 4.8, la figure 4.9 et la figure 4.10 montrent respectivement les performances de débit des flux de trafic dans les scénarios ci-dessus. Dans les figures ci-dessus, nous pouvons observer que l'augmentation de la charge de trafic de priorité inférieure n'affecte pas le débit des flux de trafic de priorité plus élevée. Il est intéressant de noter qu'en raison de la faible valeur CW_{max} de 15, le nombre total de connexions VoIP dans un IBSS doit être faible pour maintenir la stabilité du réseau. Sinon, si le numéro de connexion VoIP est supérieur à CW_{max} , il peut y avoir un nombre infini de collisions entre les connexions VoIP car au moins deux stations VoIP auront le même temporisateur d'interruption. Nous constatons que l'ajout de connexions BE et BK supplémentaires n'affecte pas le débit VoIP. De même, l'ajout de connexions VoIP et

BK / BE supplémentaires n'affecte pas le débit du trafic vidéo. On peut observer d'après la figure 4.9 que lorsque le nombre de connexions VoIP a augmenté, le débit des flux de priorité inférieure, c'est-à-dire le trafic BK / BE, a diminué.

Par conséquent, à partir des résultats ci-dessus, nous concluons que l'EDCA est capable de fournir une différenciation de service entre différents types de flux de trafic. Les flux de trafic de priorité plus élevée sont mieux servis que les flux de trafic de priorité inférieure. L'augmentation de la charge de trafic des flux de priorité plus élevée conduit à une diminution du débit et à une augmentation du délai des flux de priorité inférieure.



Figure 4.8: Débit EDCA scenario1

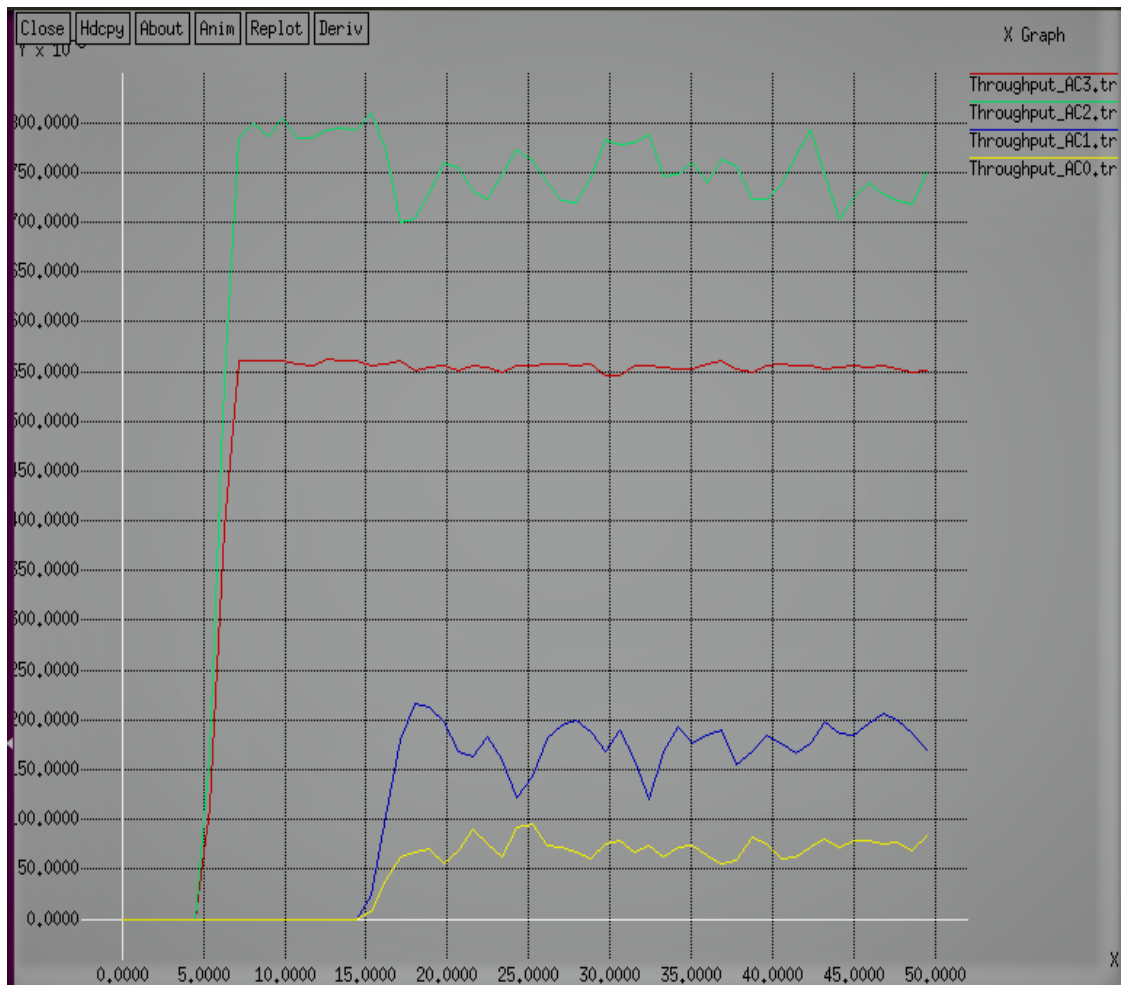


Figure 4.9: Débit EDCA scénario2

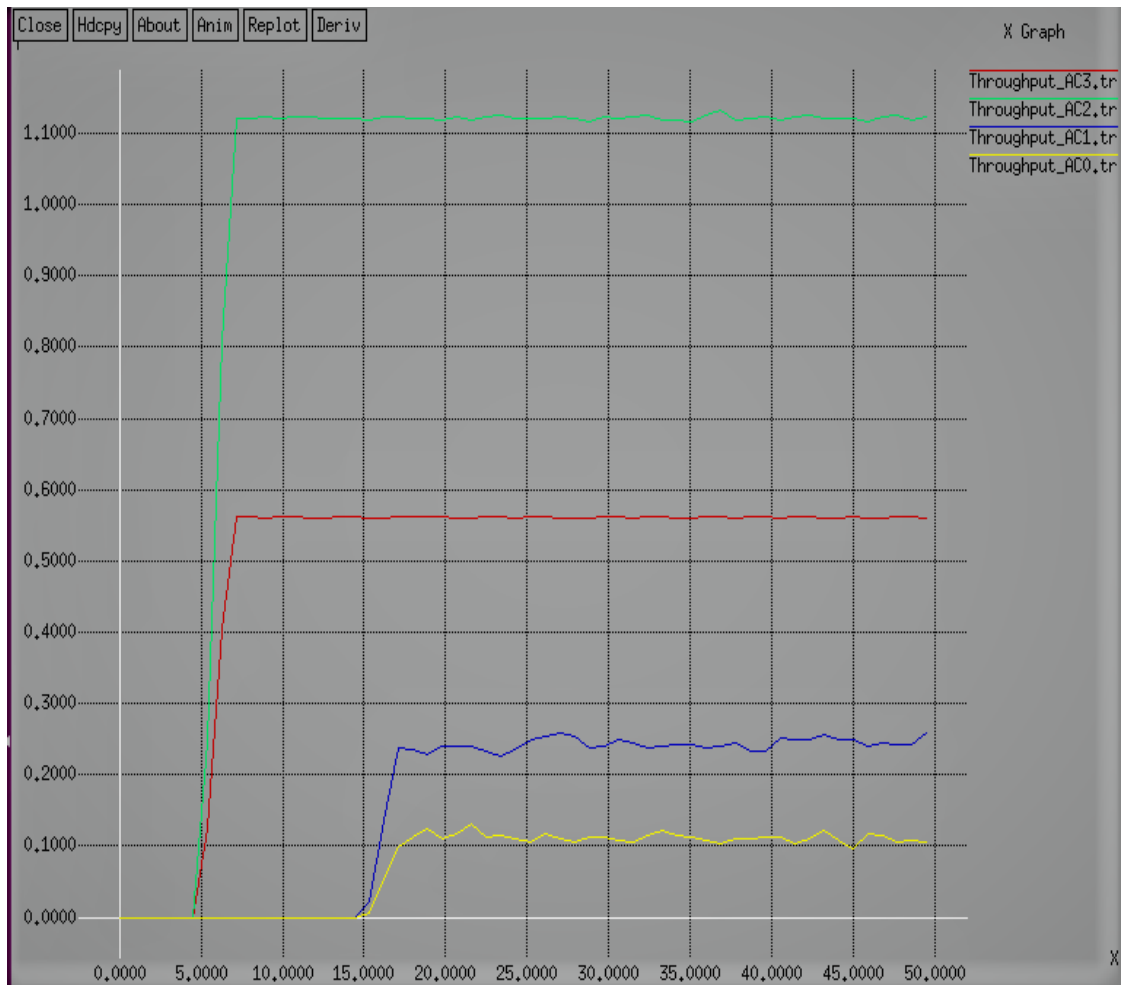


Figure 4.10: Débit EDCA scenario3

4.6 Conclusion

Dans ce chapitre, nous avons évalué les performances du mécanisme EDCA pour la prise en charge de la qualité de service dans le WLAN IEEE 802.11e. Grâce à nos simulations, nous avons comparé l'ancien 802.11 DCF et le 802.11e EDCA pour montrer que l'EDCA fournit un accès différencié aux canaux pour différents types de trafic et est mieux équipé que DCF pour gérer des applications en temps réel avec des exigences de qualité de service strictes. Nous constatons qu'avec des connexions de trafic très chargées sous un trafic de fond non négligeable, le mécanisme EDCA n'est pas en mesure de fournir une garantie de qualité de service. De meilleurs résultats peuvent être obtenus si nous pouvons adapter les paramètres EDCA pendant l'exécution en fonction de la charge du réseau et des applications prises en charge.

Conclusions et Futurs travaux

Avec la popularité croissante et rapide des WLAN, la demande d'amélioration des normes WLAN existantes est plus importante que jamais. Par conséquent, le groupe de travail IEEE 802.11 a pris la responsabilité d'améliorer le MAC 802.11 [3] pour inclure la qualité de service (QoS) afin de prendre en charge les applications multimédias, telles que la VoIP et la vidéoconférence. En conséquence, la norme IEEE 802.11e [3] a été publiée pour fournir des capacités de QoS dans les WLAN émergents en introduisant une technique Enhanced Distributed Channel Access (EDCA) basée sur une fenêtre de contention.

La fonction EDCA (désignée en abrégé EDCAF) introduite par l'IEEE repose sur le concept de catégories d'accès (AC). Il peut prendre en charge la qualité de service en introduisant une différenciation de service [18].

Dans cette recherche, nous avons d'abord étudié l'impact des paramètres EDCA QoS importants, tels que CW_{min} , CW_{max} respectivement, au moyen d'une analyse de débit et de délai sur les performances de la fonction EDCA. Ensuite, nous avons présenté une étude de simulation sur la prise en charge de la QoS dans les réseaux locaux sans fil à l'aide d'EDCF 802.11e et comparé ses performances avec 802.11. La simulation a été effectuée sur la plate-forme NS2 où nous avons utilisé les codes DCF 802.11 existants et implémenté les fonctionnalités EDCF de base dans NS2.

D'une manière générale à partir de notre simulation :

- Le mécanisme d'accès prioritaire aux canaux fourni par l'EDCF 802.11e s'est avéré efficace. Un trafic de priorité plus élevée peut obtenir un débit plus élevé et un délai d'accès MAC plus faible.

- Cependant, les AC à haute priorité souffrent quand il y a beaucoup de collisions sur le canal, résultant d'une charge élevée sur le réseau et du comportement statique d'EDCF dans la définition de la valeur CW, c'est-à-dire lorsque les files d'attente deviennent pleines, toutes les trames qui arrivent à ces stations seront rejetées, ce qui affecte la QoS, c'est-à-dire, diminue le débit global de bout-en-bout du réseau, et augmente le délai et la gigue.

Plusieurs techniques ont été suggérées pour améliorer les performances de l'EDCF en adaptant CW à l'état du réseau (SSD, CR-EDCF, SR-EDCF), Les schémas proposés visent à améliorer les performances de l'EDCF sous différents taux de charge et à accroître la différenciation des services dans les réseaux basés sur l'EDCF. Les schémas utilisent des procédures dynamiques pour modifier la valeur de la fenêtre de conflit après une collision ou une transmission réussie.

Pour les travaux futurs, nous espérons améliorer les schémas proposés et les tester avec différents scénarios en utilisant divers trafics (uniforme, avec rafales (burst),...).

Nous essayerons aussi d'étudier la méthode EDCA et l'effet de chaque paramètre pour trouver une relation entre les différents paramètres pour les ajuster dynamiquement en prenant compte aussi l'état du lien, mais il faut encore plus de recherche pour obtenir une meilleure qualité de service (QoS).

Les travaux futurs peuvent être effectués en se concentrant sur l'amélioration du débit, la diminution du délai moyen et la réduction des inconvénients des performances de l'EDCA. Cependant, améliorer la transmission des paquets et éliminer la perte de paquets.

Références

- [1] Anand & Neeli: "802.11 WLANs and IP Networking: Security, QoS, and Mobility", ARTECH HOUSE, 2005, ISBN: 1-58053-789-8.
- [2] IEEE 802.11, IEEE Standard for Information technology— Telecommunications and information exchange between systems Local and metropolitan area networks— Specific requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. Edition 2016.
- [3] IEEE, "Standard 802.11," in *PART 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, ed, 2007.
- [4] IEEE, "Standard 802.3 " in Part 3: Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications, ed, 2009.
- [5] IEEE 802.11 WG, Std 802.11, 2016 edition International Standard {for} Information Technology – Telecommunications and information exchange between systems-Local and metropolitan area network-Specific Requirements – Part 11:Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications, 2016.
- [6] Farès MALOUM, « La norme IEEE 802.11ac pour des réseaux WLANs à très haut débit : Présentation et Evaluation des performances » , Mémoire de Master, Université Abderrahmane Mira de Béjaïa, 2012.
- [7] Belkhir Thanina, Proposition d'une nouvelle méthode d'accès multi-canaux avec qos dans les réseaux wlans 802.11ac, Mémoire de Master, Université Abderrahmane Mira de Béjaïa, 2017.
- [8] Radwan et al, Performance evaluation of power management, Engineering Sciences Series Vol. (38) No. (2) 2016.
- [9] Aboulwafa & Enneya ,Introducing a Mobility Access Category for, International Journal of Innovative Technology and Exploring Engineering (IJITEE) Volume-9 Issue-5, March 2020.
- [10] Belabdelli & Oukaz « Dimensionnement D'un Réseau Sans Fil Wifi », Mémoire de fin d'études. Université abou bekr belkaid .2012.
- [11] Dufresne Loïc. «Quel est l'effet de la QoS sur des petits réseaux de labo ?», Thèse doctorat. Haute école d'ingénierie et d'architecture fribourg.2018.
- [12] J. Kurose & K.Ross, Eds., Computer networking: a top-down approach featuring the Internet. Addison-Wesley, 2003, p.^pp. Pages.
- [13] Mohamed Lamine Lamali . «Qualité de service et calcul de chemins dans les réseaux inter-domaine et multicouches», Thèse doctorat. Université de Versailles Saint-Quentin.2015.

- [14] Nabil OUAZENE, « Pour une QoS au niveau de la Couche MAC dans les Réseaux Sans Fil », Mémoire de Magister, Université Hadj Lakhdar BATNA, 2009.
- [15] Malisuwan et al, Quality of Service (QoS) and Quality of Experience (QoE), International Journal of Future Computer and Communication, Vol. 5, No. 3, June 2016.
- [16] Q. Ni, *et al.*, "A Survey of QoS Enhancements for IEEE 802.11 Wireless LAN," *Journal of Wireless Communications and Mobile Computing*, vol. 4, 2004.
- [17] Lee et al, "Dynamic_Traffic_Prioritization_and_TXOP_Allocation", 2012.
- [18] A. Helal, *et al.*, "Any Time Anywhere Computing: Mobile Computing Concepts and Technology," *Kluwer Academic Publishers*, October 1999.
- [19] B. Crow, *et al.* (September 1997) IEEE 802.11 Wireless Local Area Networks. *IEEE Communications Magazine*.
- [20] Hama Amin, Rebeen R., "Video QoS/QoE over IEEE802.11n/ac: A Contemporary Survey" (2016). Thesis. Rochester Institute of Technology.
- [21] M. Visser and M. ElZarki, "Voice and data transmission over an 802.11 wireless network," *Proceedings of IEEE Personal, Indoor & Mobile Radio* 1995.
- [22] I. Aad and C. Castelluccia, "Differentiation mechanisms for IEEE 802.11," *Proceedings of IEEE Infocom 2001, Anchorage, Alaska, USA*, 2001.
- [23] D. He and C. Shen, "Simulation Study of IEEE 802.11e EDCF," *IEEE Vehicular Technology Conference*, 2003.
- [24] F. Roijers, *et al.*, "A performance study on service integration in IEEE 802.11E wireless LANs," *Computer Communications* 29 2006.
- [25] The VINT Project: "The ns Manual", <http://www.isi.edu/nsnam/ns/nsdocumentation>, March 2007.
- [26] The network simulator, NS version 2, <https://ns2tutor.weebly.com/>.
- [27] Vaishali & Srikhant, Simulation Based Evaluation of Highway Road Scenario between DSRC802.11p MAC Protocol and STDMA for Vehicle-to-Vehicle Communicatio, *Journal of Transportation Technologies*, 2013, 3, 88-104.
- [28] Jamil et al "Radio Resource Management of Composite Wireless Networks: Predictive and Reactive Approaches", 2012.

Annexe A: Script OTCL de différenciation

```

# Dif_CWMIN.tcl
#
=====

set ns [new Simulator]

set num_mobile_nodes 5

set num_nodes $num_mobile_nodes

set opt(chan) Channel/WirelessChannel ;# channel type
set opt(prop) Propagation/TwoRayGround ;# radiopropagation model
set opt(netif) Phy/WirelessPhy ;# network interface type
set opt(mac) Mac/802_11e ;# MAC type
set opt(ifq) Queue/DTail/PriQ ;# interface queue type
set opt(ll) LL ;# link layer type
set opt(ant) Antenna/OmniAntenna ;# antenna model
set opt(ifqlen) 50 ;# max packet in ifq
set opt(adhocRouting) AODV ;# routing protocol
set opt(packet_size) 256

set opt(x) 600 ;# X dimension of the topography
set opt(y) 600 ;# Y dimension of the topography

set nf [open simulation.nam w]

$ns namtrace-all-wireless $nf $opt(x) $opt(y)

set ntr [open simulation.tr w]

$ns trace-all $ntr

$ns use-newtrace

set chan [new $opt(chan)]

set topo [new Topography]

$topo load_flatgrid $opt(x) $opt(y)

create-god $num_mobile_nodes

#
=====

$swl_node_(3) set Y_ 450
$swl_node_(3) set Z_ 0.0
$swl_node_(4) set X_ 220
$swl_node_(4) set Y_ 180
$swl_node_(4) set Z_ 0.0
$swl_node_(0) label "Nodo TX/RX"
$swl_node_(1) label "Nodo TX +"
$swl_node_(2) label "Nodo RX +"
$swl_node_(3) label "Nodo RX -"
$swl_node_(4) label "Nodo RX -"

set src_udp0 [new Agent/UDP]
Mac/802_11e set CWMin_ 7 ;

$src_udp0 set prio_ 0
$src_udp0 set fid_ 0

set dst_udp0 [new Agent/LossMonitor]
$ns attach-agent $swl_node_(0) $src_udp0
$ns attach-agent $swl_node_(3) $dst_udp0
set app [new Application/Traffic/CBR]
$app attach-agent $src_udp0

$ns connect $src_udp0 $dst_udp0

$ns at 1.1 "$app start"

set src_udp1 [new Agent/UDP]
$src_udp1 set prio_ 1
$src_udp1 set fid_ 1

Mac/802_11e set CWMin_ 15;

set dst_udp1 [new Agent/LossMonitor]
$ns attach-agent $swl_node_(0) $src_udp1
$ns attach-agent $swl_node_(4) $dst_udp1

```

```
# *** Throughput Trace ***
```

```
set f0 [open Debit_CWMIN_7.tr w]
```

```
set f1 [open Debit_CWMIN_15.tr w]
```

```
set f2 [open out22.tr w]
```

```
# *** Packet Loss Trace ***
```

```
set f4 [open lost02.tr w]
```

```
set f5 [open lost12.tr w]
```

```
set f6 [open lost22.tr w]
```

```
# *** Packet Delay Trace ***
```

```
set f8 [open Delai_CWMIN_7.tr w]
```

```
set f9 [open Delai_CWMIN_15.tr w]
```

```
set f10 [open delay22.tr w]
```

```
Phy/WirelessPhy set bandwidth_ 11Mb
```

```
Application/Traffic/CBR set packetSize_ $opt(packet_size)
```

```
$ns node-config -adhocRouting $opt(adhocRouting) \
```

```
-llType $opt(ll) \
```

```
-macType $opt(mac) \
```

```
-ifqType $opt(ifq) \
```

```
-ifqLen $opt(ifqlen) \
```

```
-antType $opt(ant) \
```

```
-propType $opt(prop) \
```

```
-phyType $opt(netif) \
```

```
-channel $chan \
```

```
-topoInstance $topo \
```

```
-wiredRouting OFF \
```

```
-agentTrace ON \
```

```
-routerTrace ON \
```

```
-macTrace ON \
```

```
-movementTrace ON
```

```
set wl_node_(0) [$ns node]
```

```
$wl_node_(0) random-motion 0
```

```
set wl_node_(1) [$ns node]
```

```
set app1 [new Application/Traffic/CBR]
```

```
$app1 attach-agent $src_udp1
```

```
$ns connect $src_udp1 $dst_udp1
```

```
$ns at 1.5 "$app1 start"
```

```
set src_udp2 [new Agent/UDP]
```

```
set dst_udp2 [new Agent/LossMonitor]
```

```
$ns attach-agent $wl_node_(1) $src_udp2
```

```
$ns attach-agent $wl_node_(2) $dst_udp2
```

```
set app2 [new Application/Traffic/CBR]
```

```
$app2 attach-agent $src_udp2
```

```
$ns connect $src_udp2 $dst_udp2
```

```
$ns at 2.0 "$app2 start"
```

```
for {set i 0} {$i < $num_mobile_nodes} {incr i} {
```

```
$ns initial_node_pos $wl_node_($i) 40
```

```
}
```

```
for {set i 0} {$i < $num_mobile_nodes} {incr i} {
```

```
$ns at 10.0 "$wl_node_($i) reset";
```

```
}
```

```
# Initialize Flags
```

```
set holdtime 0
```

```
set holdseq 0
```

```
set holdtime1 0
```

```
set holdseq1 0
```

```
set holdtime2 0
```

```
set holdseq2 0
```

```
set holdrate1 0
```

```
set holdrate2 0
```

```
set holdrate3 0
```

```
# Function To record Statistcis (Bit Rate, Delay, Drop)
```

```
proc record {} {
```

```

$swl_node_(1) random-motion 0
set wl_node_(2) [$ns node]
$swl_node_(2) random-motion 0
set wl_node_(3) [$ns node]
$swl_node_(3) random-motion 0
set wl_node_(4) [$ns node]
$swl_node_(4) random-motion 0
$swl_node_(0) set X_ 300
$swl_node_(0) set Y_ 300
$swl_node_(0) set Z_ 0.0
$swl_node_(1) set X_ 100
$swl_node_(1) set Y_ 300
$swl_node_(1) set Z_ 0.0
$swl_node_(2) set X_ 500
$swl_node_(2) set Y_ 300
$swl_node_(2) set Z_ 0.0
$swl_node_(3) set X_ 370

```

```

global dst_udp0 dst_udp1 dst_udp2 f0 f1 f2 f4 f5 f6
holdtime holdseq holdtime1 holdseq1 holdtime2
holdseq2 f8 f9 f10 holdrate1 holdrate2 holdrate3

set ns [Simulator instance]

set time 0.9 ;#Set Sampling Time to 0.9 Sec

set bw0 [$dst_udp0 set bytes_]
set bw1 [$dst_udp1 set bytes_]
set bw2 [$dst_udp2 set bytes_]
set bw4 [$dst_udp0 set nlost_]
set bw5 [$dst_udp1 set nlost_]
set bw6 [$dst_udp2 set nlost_]
set bw8 [$dst_udp0 set lastPktTime_]
set bw9 [$dst_udp0 set npkts_]
set bw10 [$dst_udp1 set lastPktTime_]
set bw11 [$dst_udp1 set npkts_]
set bw12 [$dst_udp2 set lastPktTime_]
set bw13 [$dst_udp2 set npkts_]

```

Résultats de la simulation

Paramètre CWmin (CWmin = 7 et CWmin = 15)

```
# Dif_CWMIN.tcl
```

```
# =====
```

```
Mac/802_11e set CWmin_ 7 ; Mac/802_11e set CWmin_ 15 .
```

```

ns2@ns2-virtual-machine: ~/differentiation
ns2@ns2-virtual-machine:~/differentiation$ ns235-edca Dif_CWMIN.tcl

```

Paramètre CWmax (CWmax = 31 et CWmax = 1023)

```
# Dif_CWMAX.tcl
```

```
# =====
```

```
Mac/802_11e set CWmax_ 31 ; Mac/802_11e set CWmax_ 1023 .
```

```

ns2@ns2-virtual-machine: ~/differentiation
ns2@ns2-virtual-machine:~/differentiation$ ns235-edca Dif_CWMAX.tcl

```

Annexe B: Script OTCL de comparaison

```

# scenario_final.tcl
# =====
# Passage des paramètres :
#1 type de La couche MAC (0: 802.11, 1: 802.11e);
#2 nombre de connexions VoIP (AC_3);
#3 nombre de connexion vidéo (AC_2);
#4 nombre de connexion meilleur effort (AC_1);
#5 nombre de connexion d'arrière-plan (AC_0);
# =====
proc getopt {argc argv} {
    global opt
    lappend optlist nn
    for {set i 0} {$i < $argc} {incr i} {
        set opt($i) [lindex $argv $i]
    }
}
getopt $argc $argv
set max_fragmented_size 1024
set val(chan) Channel/WirelessChannel ;# channel type
set val(prop) Propagation/TwoRayGround ;# propagation model
set val(netif) Phy/WirelessPhy ;# network interface type
if {$Sopt(0) > 0} {
    set val(mac) Mac/802_11e ;# MAC type
    set val(ifq) Queue/DropTail/PriQ ;# interface queue type
    Mac/802_11e set dataRate_ 1Mb
    Mac/802_11e set basicRate_ 1Mb
    Phy/WirelessPhy set bandwidth_ 11e6
    Phy/WirelessPhy set CPTthresh_ 10.0
    Phy/WirelessPhy set CSTthresh_ 1.2589e-11
    Phy/WirelessPhy set RXThresh_ 2.04365e-10
    Phy/WirelessPhy set Pt_ 0.1
    Phy/WirelessPhy set freq_ 2.412e9 ;# Canal 1
    Mac/802_11e set SlotTime_ 0.000009 ;# 9us
    Mac/802_11e set SIFS_ 0.000130 ;# 10us
    Mac/802_11e set PreambleLength_ 72 ;# 72 bits
    Mac/802_11e set PLCPDataRate_ 2.0e6 ;# 2Mbps
    Mac/802_11e set dataRate_ 54.0e6 ;# 54Mbps
} else {
    set val(mac) Mac/802_11 ;# MAC type
    set val(ifq) Queue/DropTail/PriQueue ;# interface queue type
    Mac/802_11 set dataRate_ 1Mb
    Mac/802_11 set basicRate_ 1Mb
    Phy/WirelessPhy set bandwidth_ 11e6
    Phy/WirelessPhy set CPTthresh_ 10.0
    Phy/WirelessPhy set CSTthresh_ 1.2589e-11
    Phy/WirelessPhy set RXThresh_ 2.04365e-10
    Phy/WirelessPhy set Pt_ 0.1
    Phy/WirelessPhy set freq_ 2.412e9 ;# Canal 1
    Mac/802_11 set SlotTime_ 0.000009 ;# 9us
    Mac/802_11 set DIFS_ 0.000028 ;# 28us
    Mac/802_11 set SIFS_ 0.000080 ;# 10us
    Mac/802_11 set PreambleLength_ 72 ;# 72 bits
}

# *** Packet Delay Trace ***
set f8 [open delay3.tr w]
set f9 [open delay2.tr w]
set f10 [open delay1.tr w]
set f11 [open delay0.tr w]
# =====
# Initialization
# =====
# Create a ns simulator
set ns [new Simulator]
# set up topography object
set topo [new Topography]
$topo load_flatgrid $val(x) $val(y)
create-god $val(nn)
# Open the NS trace file
set f [open test.tr w]
$ns trace-all $f
# Open the NAM trace file
set namfile [open out.nam w]
$ns namtrace-all $namfile
$ns namtrace-all-wireless $namfile $val(x) $val(y)
# create channel
set chan [new $val(chan)]
# =====
# Mobile node parameter setup
# =====
$ns node-config -adhocRouting $val(rp) \
    -llType $val(ll) \
    -macType $val(mac) \
    -ifqType $val(ifq) \
    -ifqLen $val(ifqlen) \
    -antType $val(ant) \
    -propType $val(prop) \
    -phyType $val(netif) \
    -channel $chan \
    -topoInstance $topo \
    -agentTrace ON \
    -routerTrace ON \
    -macTrace ON \
    -movementTrace ON
# =====
# Nodes Definition
# =====
# Create 4 nodes
for {set i 0} {$i < 4} {incr i} {
    set node_($i) [$ns node]
    $node_($i) random-motion 0
}
$node_(0) set X_ 300
$node_(0) set Y_ 400
$node_(0) set Z_ 0.0
$ns initial_node_pos $node_(0) 20

```

```

Mac/802_11 set PLCPDataRate_ 2.0e6 ;# 2Mbps
Mac/802_11 set dataRate_ 54.0e6 ;# 54Mbps
}
set val(ll) LL ;# link layer type
set val(ant) Antenna/OmniAntenna ;# antenna model
set val(ifqlen) 50 ;# max packet in ifq
set val(nn) 4 ;# number of mobilenodes
set val(rp) DSDV
set val(x) 500 ;# X dimension of topography
set val(y) 500 ;# Y dimension of topography
set val(stop) 50.0 ;# time of simulation end
set opt(choice) $opt(0)
set opt(voiceflow) $opt(1)
set opt(videoflow) $opt(2)
set opt(TCPflow) $opt(3)
set opt(CBRflow) $opt(4)
# *** Throughput Trace ***
set f0 [open Throughput_AC3.tr w]
set f1 [open Throughput_AC2.tr w]
set f2 [open Throughput_AC1.tr w]
set f3 [open Throughput_AC0.tr w]
# *** Packet Loss Trace ***
set f4 [open Loss_AC3.tr w]
set f5 [open Loss_AC2.tr w]
set f6 [open Loss_AC1.tr w]
set f7 [open Loss_AC0.tr w]
#1st priority traffic (VoIP)
for {set i 0} {$i < $opt(voiceflow)} {incr i} {
    set udpA_($i) [new Agent/UDP]
    $udpA_($i) set prio_ 0
    $ns attach-agent $node_(1) $udpA_($i)
    set sinkA_($i) [new Agent/LossMonitor] ; # Create Loss
    Monitor Sink in order to be able to trace the number obytes received
    $ns attach-agent $node_(0) $sinkA_($i) ; # Attach Agent to
    sink node
    $ns connect $udpA_($i) $sinkA_($i) ; # Connect the
    nodes
    set voip_($i) [new Application/Traffic/CBR]
    $voip_($i) attach-agent $udpA_($i)
    $voip_($i) set packetize_ 500
    $voip_($i) set rate_ 512k
    $ns at 5.0 "$voip_($i) start"
}
#2nd priority traffic (Video, Foreman)
for {set i 0} {$i < $opt(videoflow)} {incr i} {
    set udpB_($i) [new Agent/UDP]
    $udpB_($i) set prio_ 1
    $ns attach-agent $node_(2) $udpB_($i)
    set sinkB_($i) [new Agent/LossMonitor] ; # Create Loss
    Monitor Sink in order to be able to trace the number obytes received
    $ns attach-agent $node_(0) $sinkB_($i) ; # Attach Agent to
    sink node
    $ns connect $udpB_($i) $sinkB_($i) ; # Connect the nodes
    set vid_($i) [new Application/Traffic/CBR]
    $vid_($i) attach-agent $udpB_($i)
    $vid_($i) set packetize_ 450
    $vid_($i) set rate_ 1024k
    $ns at 5.0 "$vid_($i) start"
}

```

```

$node_(1) set X_ 400
$node_(1) set Y_ 400
$node_(1) set Z_ 0.0
$ns initial_node_pos $node_(1) 20
$node_(2) set X_ 300
$node_(2) set Y_ 300
$node_(2) set Z_ 0.0
$ns initial_node_pos $node_(2) 20
$node_(3) set X_ 400
$node_(3) set Y_ 300
$node_(3) set Z_ 0.0
$ns initial_node_pos $node_(3) 20
set holdtime 0
set holdseq 0
set holdtime1 0
set holdseq1 0
set holdtime2 0
set holdseq2 0
set holdtime3 0
set holdseq3 0
set holdrate1 0
set holdrate2 0
set holdrate3 0
set holdrate4 0

set bw7 [$sinkD_(0) set nlost_]
set bw8 [$sinkA_(0) set lastPktTime_]
set bw9 [$sinkA_(0) set npkts_]
set bw10 [$sinkB_(0) set lastPktTime_]
set bw11 [$sinkB_(0) set npkts_]
set bw12 [$sinkC_(0) set lastPktTime_]
set bw13 [$sinkC_(0) set npkts_]
set bw14 [$sinkD_(0) set lastPktTime_]
set bw15 [$sinkD_(0) set npkts_]
set time 0.9 ;#Set Sampling Time to 0.9 Sec
set now [$ns now]
# Record Bit Rate in Trace Files
puts $f0 "$now [expr
(($bw0+$holdrate1)*8)/(2*$time*1000000)]"
puts $f1 "$now [expr
(($bw1+$holdrate2)*8)/(2*$time*1000000)]"
puts $f2 "$now [expr
(($bw2+$holdrate3)*8)/(2*$time*1000000)]"
puts $f3 "$now [expr
(($bw3+$holdrate4)*8)/(2*$time*1000000)]"
# Record Packet Loss Rate in File
puts $f4 "$now [expr $bw4/$time]"
puts $f5 "$now [expr $bw5/$time]"
puts $f6 "$now [expr $bw6/$time]"
puts $f7 "$now [expr $bw7/$time]"
# Record Packet Delay in File
if { $bw9 > $holdseq } {
    puts $f8 "$now [expr ($bw8 -
$holdtime)/($bw9 - $holdseq)]"
} else {
    puts $f8 "$now [expr ($bw9 - $holdseq)]"
}
}

```

```

#3rd priority traffic (CBR)
for {set i 0} {$i < Sopt(CBRflow)} {incr i} {
    set udpC_($i) [new Agent/UDP]
    SudpC_($i) set prio_2
    Sns attach-agent $node_(3) SudpC_($i)
    set sinkC_($i) [new Agent/LossMonitor]; # Create Loss
Monitor Sink in order to be able to trace the number obytes received
    Sns attach-agent $node_(0) SsinkC_($i); # Attach Agent to
sink node
    Sns connect SudpC_($i) SsinkC_($i) ; # Connect the
nodes

    set cbr_($i) [new Application/Traffic/CBR]
    Scbr_($i) attach-agent SudpC_($i)
    Scbr_($i) set packetSize_200
    Scbr_($i) set rate_256k
    Scbr_($i) set random_false
    Sns at 15.0 "Scbr_($i) start"
}

#4th priority traffic (TCP)
for {set i 0} {$i < Sopt(TCPflow)} {incr i} {
    set tcp($i) [new Agent/UDP]
    Stcp($i) set prio_3
    Sns attach-agent $node_(0) Stcp($i)
    set sinkD_($i) [new Agent/LossMonitor]
    Sns attach-agent $node_(1) SsinkD_($i)
    Sns connect Stcp($i) SsinkD_($i)
    set ftp($i) [new Application/Traffic/CBR]
    Sftp($i) set packetSize_200
    Sftp($i) set rate_128k
    Scbr_($i) set random_false
    Sftp($i) attach-agent Stcp($i)
    Sns at 15.0 "Sftp($i) start"
}

set n0_ifq [$node_(0) set ifq_(0)]
$N0_ifq set choice $sopt(choice)
$N0_ifq set threshold1 10
$N0_ifq set threshold2 40
$N0_ifq set prob0 0
$N0_ifq set prob1 0.6
$N0_ifq set prob2 0.9
# Function To record Statistcis (Bit Rate, Delay, Drop)
proc record {} {
    global sinkA_ sinkB_ sinkC_ sinkD_ f0 f1 f2 f3 f4 f5 f6 f7
    holdtime holdseq holdtime1 holdseq1 holdtime2 holdseq2 holdtime3
    holdseq3 f8 f9 f10 f11 holdrate1 holdrate2 holdrate3 holdrate4 opt
    set ns_ [Simulator instance]
    set bw0 [$sinkA_(0) set bytes_]
    set bw1 [$sinkB_(0) set bytes_]
    set bw2 [$sinkC_(0) set bytes_]
    set bw3 [$sinkD_(0) set bytes_]
    set bw4 [$sinkA_(0) set nlost_]
    set bw5 [$sinkB_(0) set nlost_]
    set bw6 [$sinkC_(0) set nlost_]
}

```

```

    if { $bw11 > $holdseq1 } { puts $f9 "$now
[expr ($bw10 - $holdtime1)/($bw11 - $holdseq1)]"
    } else { puts $f9 "$now [expr ($bw11 -
$holdseq1)]" }
    if { $bw13 > $holdseq2 } { puts $f10 "$now
[expr ($bw12 - $holdtime2)/($bw13 - $holdseq2)]"
    } else { puts $f10 "$now [expr ($bw13 -
$holdseq2)]" }
    if { $bw15 > $holdseq3 } {
        puts $f11 "$now [expr ($bw14 -
$holdtime3)/($bw15 - $holdseq3)]"
    } else {
        puts $f11 "$now [expr ($bw15 -
$holdseq3)]"
    }
# Reset Variables
$SinkA_(0) set bytes_0
$SinkB_(0) set bytes_0
$SinkC_(0) set bytes_0
$SinkD_(0) set bytes_0
$SinkA_(0) set nlost_0
$SinkB_(0) set nlost_0
$SinkC_(0) set nlost_0
$SinkD_(0) set nlost_0
set holdtime $bw8
set holdseq $bw9
set holdrate1 $bw0
set holdrate2 $bw1
set holdrate3 $bw2
set holdrate4 $bw3
$ns_ at [expr $now+$time] "record" ;# Schedule
Record after $time interval sec
}

# Start Recording at Time 0
$ns at 0.0 "record"
$ns at 50.0 "stop"
proc stop {} {
    global ns f f0 f1 f2 f3 f4 f5 f6 f7 f8 f9 f10 f11
    Throughput_AC3 Throughput_AC1
    Throughput_AC2 Throughput_AC3
# Close Trace Files
close $f close $f0 close $f1 close $f2 close $f3
close $f4 close $f5 close $f6 close $f7 close $f8
close $f9 close $f10 close $f11
exec xgraph $Throughput_AC0 $Throughput_AC1
$Throughput_AC2 $Throughput_AC3 -geometry
800x400 &
exec xgraph delay0.tr delay1.tr delay2.tr delay3.tr -
geometry 800x400 &
exit 0
}
for {set i 0} {$i < 3} {incr i} {
    $ns at 50.0 "$node_($i) reset";
}
puts "Starting Simulation..."
$ns at $val(stop) "$ns nam-end-wireless $val(stop)"
$ns at $val(stop) "finish"
$ns at $val(stop) "puts \"done\""; $ns halt"
$ns run

```

Résultats de la simulation

Partie1 DCF

- **Scénario 1** : Le scénario 1 se compose de deux connexions VoIP, une connexion vidéo, deux connexions pour le trafic d'arrière-plan, et deux connexions pour les données de meilleur effort.

```
ns2@ns2-virtual-machine: ~/comparaison
ns2@ns2-virtual-machine:~/comparaison$ ns235-edca scenario.tcl 0 2 1 2 2
```

- **Scénario 2** : Le scénario 2 se compose de sept connexions VoIP, une connexion vidéo, deux connexions pour le trafic d'arrière-plan, et deux connexions pour les données de meilleur effort.

```
ns2@ns2-virtual-machine: ~/comparaison
ns2@ns2-virtual-machine:~/comparaison$ ns235-edca scenario.tcl 0 7 1 2 2
```

- **Scénario 3** : Le scénario 3 se compose de deux connexions VoIP, une connexion vidéo, quatre connexions pour le trafic d'arrière-plan, et quatre connexions pour les données de meilleur effort.

```
ns2@ns2-virtual-machine: ~/comparaison
ns2@ns2-virtual-machine:~/comparaison$ ns235-edca scenario.tcl 0 2 1 4 4
```

Partie2 EDCA

- **Scénario 1** : Le même scénario 1 de la partie1 DCF.

```
ns2@ns2-virtual-machine: ~/comparaison
ns2@ns2-virtual-machine:~/comparaison$ ns235-edca scenario.tcl 1 2 1 2 2
```

- **Scénario 2** : Le même scénario 2 de la partie1 DCF.

```
ns2@ns2-virtual-machine: ~/comparaison
ns2@ns2-virtual-machine:~/comparaison$ ns235-edca scenario.tcl 1 7 1 2 2
```

- **Scénario 3** : Le même scénario 3 de la partie1 DCF.

```
ns2@ns2-virtual-machine: ~/comparaison
ns2@ns2-virtual-machine:~/comparaison$ ns235-edca scenario.tcl 1 2 1 4 4
```