QUALITE DE SEVICE WI-FI : ALGORITHME POUR LE SUPPORT DE DIFFSERV

Ilyes Gouta,
Ingénieur - Chercheur, ENSI
ilyes.gouta@ensi.rnu.tn

Pr. Abdelfettah Belghith,
Professeur, ENSI
abdelfattah.belghith@ensi.rnu.tn

Dr. Jean-Marie Bonnin,
Maître de conférences, ENST - Bretagne
jm.bonnin@enst-bretagne.fr

Adresse professionnelle,
Ecole Nationale des Sciences de l'Informatique, campus de Manouba,
Tunis, Tunisie

Résumé - Ce papier présente un algorithme destiné à offrir le support des classes de services DiffServ au-dessus de 802.11. L'objectif est de différentier les classes de service dans leur capacité à obtenir le support et ainsi d'offrir une certaine gestion de la QoS sur ce genre de médium. Ce papier présente le fonctionnement de l'architecture à différentiation de services (DiffServ) ainsi que la norme 802.11 de l'IEEE. Une étude de performances à l'aide d'une simulation sous NS met ensuite en relief les différences, en termes de débit et de gigue, entre le 802.11b original et le 802.11b modifié. Quelques perspectives et applications sont présentées en guise de conclusion.

Mots-clés: QoS, IEEE 802.11, DiffServ, CSMA/CA

QUALITE DE SEVICE WI-FI : ALGORITHME POUR LE SUPPORT DE DIFFSERV

1. INTRODUCTION

Le besoin en “informatique mobile” s’est fait sentir depuis le milieu des années 90. En effet, il est devenu indispensable de pouvoir accéder depuis n’importe quel emplacement aux données de son établissement/entreprise afin de répondre efficacement et rapidement aux exigences de l’entreprise mais aussi du particulier. Avant l’avènement des réseaux sans-fil, il fallait câbler d’une façon permanente l’endroit en question ce qui risquait de limiter/conditionner l’accès à ces données. Avec les technologies sans-fil, il est maintenant possible d’avoir un lien plus ou moins permanent avec son environnement qu’il soit de travail ou de loisir. L’introduction des applications telles que la visioconférence, le contrôle à distance donne à la technologie sans-fil une nouvelle dimension mais aussi de nouvelles contraintes en termes de délais de livraison et de bande passante allouée. En 1999, l’IEEE a normalisé la technique d’accès sans-fil 802.11b, largement utilisée de nos jours, offrant ainsi un débit théorique de 11 Mbps et assurant une couverture de l’ordre d’une centaine de mètres aux équipements qui intègrent ce standard. Techniquement 802.11b emploie une variante de CSMA, appelée CSMA/CA, comme protocole de niveau lien pour réguler les échanges des différentes entités communicantes. Le CSMA/CA est aussi connu pour ses limitations en termes de garantie de service.

Ce papier présente un algorithme permettant d’améliorer quelques aspects de son fonctionnement tout en offrant un support de la technique de différentiation de service (DiffServ). Une comparaison avec les performances du 802.11b original est présentée pour mettre en évidence les apports en termes de différentiation des différentes classes de trafic que ce soit pour le débit ou pour la variation d’interarrivée.

2. LA TECHNIQUE DIFFSERV

Définie par l’IETF, DiffServ [1] (pour Differenciated Services) est une technique destinée à intégrer les éléments de base de la gestion de la QoS pour les réseaux IP. DiffServ agit au niveau des agrégats de trafic en segmentant le trafic total en plusieurs classes (classes de services) dont le traitement est ensuite différentié dans les équipements d’interconnexion. L’allocation des ressources se fait par classe et non plus par application. Suivant la classe à laquelle un paquet appartient, un traitement plus ou moins privilégié lui est appliqué (ceci inclut la priorité d’acheminement, l’élimination sélective des paquets en cas de congestion). Cette solution est beaucoup moins lourde que l’approche à intégration de service (IntServ) qui suppose l’établissement explicite d’une réservation pour chaque flux de données. DiffServ définit 3 types de classes de services, l’administrateur d’un réseau ayant la liberté de n’implémenter que les classes qu’il juge nécessaires. Les différents types de classes sont les suivants :

EF (Expedited Forwarding) : c’est la classe d’excellence. Les paquets marqués EF doivent être acheminés avec un délai, une gigue et un taux de perte minimum. Des moyens techniques (contrôle d’accès, sur réservation,…) doivent être mis en oeuvre pour assurer le bon fonctionnement de celle-ci.

AF (Assured Forwarding) : quatre classes AF ont été défini, chacune d’elles comporte 3 sous-classes. Les paquets sont marqués AFxy tel que x dans l’intervalle [1,4] est le numéro de la classe AF et y dans [1,3] la précédence à l’écartement. Les paquets d’une même classe empruntent toujours la même file d’attente pour éviter le déséquencement. La précédence à l’écartement définit la priorité relative de rejet (ou un traitement spécifique) des paquets en cas de congestion.

BE (Best Effort) : c’est l’équivalent de l’Internet actuel où aucun traitement particulier ne vient améliorer le relayage des paquets appartement à cette classe.

Au total nous dénombrons 14 comportements possibles. Ils sont notés PHBs (Per Hop Behaviour). Un PHB est la manière avec laquelle un routeur traitera les paquets entrants (c’est à dire la mise en file d’attente plus le traitement en cas de congestion). Le PHB est déterminé à partir de DSPCs codés directement dans le champ TOS du paquet IP.

La norme recommande l’utilisation d’un ensemble précis de DSCPs pour marquer les

paquets IP. Le PHB par défaut (BE) est défini pour les paquets non marqués.

	Précédence 1	Précédence 2	Précédence 3
EF	46		
AF4	34	36	38
AF3	26	28	30
AF2	18	20	22
AF1	10	12	14
BE	0		

La norme recommande l'utilisation d'un ensemble précis de DSCPs pour marquer les paquets IP. Le PHB par défaut (BE) est défini pour les paquets non marqués.

3. LA NORME IEEE 802.11

L' IEEE (Institute of Electrical and Electronics Engineers) a normalisé plusieurs catégories de réseaux locaux : une variante de Ethernet (802.3), le Token Bus (802.4) et le Token Ring (802.5). En 1990 le projet d'un réseau local sans fil, nommé 802.11 [2], est lancé. Il a pour but de fournir une liaison sans fil à un ensemble d'utilisateurs fixes ou mobiles.

La norme 802.11 s'adresse essentiellement aux niveaux lien et physique du modèle OSI. En fait, elle introduit des modifications sur la couche basse du niveau lien (donc niveau MAC) et sur le niveau physique avec le support de plusieurs méthodes d'accès radio (donc la définition de plusieurs couches physiques). Il est à noter que la nouvelle couche MAC est commune à toutes les couches physiques.

3.1 La couche physique

L'IEEE a initialement défini trois couches physiques initiales :

FHSS : pour Frequency Hopping Spread Spectrum, c'est une technique d'accès radio qui consiste en l'émission des symboles sur une bande de fréquence de largeur fixe et de porteuse variable dans le temps. L'émetteur doit se mettre en accord avec le récepteur sur la séquence de porteuses à utiliser. En mode infrastructure, c'est le point d'accès qui annonce les fréquences à utiliser pendant les émissions.

DSSS : pour Direct Sequence Spread Spectrum, le fonctionnement est similaire à FHSS sauf qu'il n'y a pas de changement de porteuse dans le temps, d'où un plus large spectre d'émission et donc un meilleur débit.

IR : pour Infrared, les infra rouges sont utilisés pour le transfert des données. Cette méthode impose que les distances entre émetteurs/récepteurs soient limitées. Elle offre un débit de 1 Mbps.

L'amendement 802.11b [3], qui définit une quatrième couche physique, utilise la méthode DSSS, couplé avec un encodage utilisant la modulation de phase améliorée, pour la transmission des données. 802.11b permet d'atteindre un débit théorique de 11 Mbps. 802.11a, la cinquième couche, fait encore mieux en autorisant des débits jusqu'à 54 Mbps (codage OFDM). Toutefois, 802.11a utilise la gamme des 5 GHz et, est de facto, considérée comme étant un cas particulier de la norme. Enfin, l'IEEE vient de standardiser une nouvelle couche physique, nommée 802.11g, qui utilise les techniques de 802.11a pour obtenir le même débit dans la bande ISM (2,4 GHz).

3.2 La couche Liaison

Dans 802.11, la couche liaison est divisée en deux sous-couches : LLC (Logical Link Control) et MAC (Medium Access Control). La couche LLC a les mêmes propriétés que celles définies pour la couche LLC 802.2. Il est donc concevable de relier un réseau sans fil (WLAN) à n'importe quel réseau qui adopte cette couche pour gérer ses accès. La couche MAC est responsable de la procédure d'allocation du support (accès), de l'adressage des paquets, du formatage des trames, du contrôle d'erreur (via un CRC, Cyclic Redundancy Check), ainsi que de la fragmentation et du réassemblage.

3.3 Accès au support

L'accès au support est déterminé par une fonction dite fonction de coordination : c'est une fonction logique qui détermine l'instant d'émission/réception d'une station associée à un Basic Service Set (BSS). Le standard définit deux méthodes d'accès au support : DCF (Distributed Coordination Function) et PCF (Point Coordination Function). La première utilise la technique CSMA/CA (Carrier Sense Multiple Access/Collision Avoidance) pour arbitrer l'accès au support. Elle est conçue de façon à ce que tous les utilisateurs aient une chance égale d'accéder au médium. La deuxième technique est basée sur l'interrogation régulière, par le point d'accès, de l'ensemble des stations pour leur demander s'ils ont des données à transmettre. Chaque station, pour qu'elle soit interrogée, doit s'enregistrer et réserver préalablement un temps d'émission auprès du point d'accès. Actuellement, la plupart du matériel Wi-Fi disponible sur le marché n'implémente que la méthode d'accès DCF, que ce soit pour les points d'accès ou les stations clientes.

3.4 La fonction de coordination DCF

Dans un environnement radio, une station qui émet n'a pas la possibilité de vérifier, en parallèle,

l'intégrité des données qu'elle émet (s'il y a eu collision ou pas car en radio il est très difficile/coûteux d'émettre et de recevoir au même instant), d'où la nécessité de tenir compte de cette limitation dans la conception du contrôle d'accès au medium. DCF met en oeuvre la technique CSMA/CA pour résoudre ce problème, en instaurant une démarche particulière à exécuter avant et pendant l'émission d'une trame. De plus comme il n'y a pas de détection de collision, CSMA/CA utilise des trames ACK pour valider l'envoi des données. La norme définit 3 temps inter-trames destinés à harmoniser le dialogue entre les machines et à diminuer les collisions. Ce sont des temps qui espacent les transmissions inter et intra dialogue(s). Ces temps sont :

SIFS : pour Short Inter-Frame Spacing, qui vaut 28 ms. Ce temps est utilisé pour espacer les émissions d'un même dialogue. (machine A à machine B).

PIFS (PCF IFS) : pour Point Inter-Frame Spacing, égal à un SIFS + 78 ms. C'est un temps utilisé par le point d'accès pour assurer l'acquisition du support. (accès prioritaire)

DIFS (DCF IFS) : pour Distributed Inter-Frame Spacing, temps utilisé par une station pour commencer une nouvelle transmission. Il vaut un PIFS + 128 ms.

Chaque station possède un temporisateur, appelé NAV (Network Allocation Vector), qui sert à retarder le temps d'émission d'une éventuelle trame. Ce temporisateur est mis à jour dès le début de l'émission d'une trame par une autre station. NAV est alors incrémenté par la valeur du deuxième champ « Durée » situé dans l'entête MAC. La station procède alors à sa décrémentation. Une fois le champ NAV à zéro, cette dernière continue à attendre un DIFS. Si après cette période le support est libre (et si elle a des données à émettre), elle envoie ses données sinon elle continue à écouter le support jusqu'à qu'il soit libre. Elle retransmettra ses trames après un temps de reprise (back off) déterminé par l'algorithme BEB (Binary Exponential Back off).

Ce temps de reprise est déterminé par la quantité [22+i*rand()]*timeslot, avec i le nombre de tentatives consécutives d'émission, rand() un nombre aléatoire uniforme compris entre 0 et 1. Ce temps est toujours compris entre les deux valeurs CWmin et CWmax spécifiés par le point d'accès. Si une trame, envoyée par une station A, est reçue correctement par B, B doit répondre par une trame d'acquittement ACK après un SIFS pour indiquer à A le succès de la transmission. Si un ACK n'est pas reçu, la station A reprend la transmission. Les performances de cet algorithme peuvent être améliorées en utilisant un mécanisme optionnel basé sur la réservation de temps d'accès. Ce mécanisme s'apparente de la méthode RTS/CTS utilisée depuis longtemps dans des équipements comme les modems et les cartes réseaux. L'idée de base est d'envoyer une trame RTS (Request To Send) à la station destination avant l'émission d'une (ou plusieurs) trame. La durée totale d'émission est indiquée dans la trame RTS. Ceci inclut, dans notre cas (802.11b), le temps d'émission des données, des SIFS et des ACKs. La machine destination répond par un CTS (Clear To Send). En écoutant cette signalisation, les autres stations mettent à jour leur temporisateur NAV. Cette technique permet d'éviter une éventuelle collision lors d'un transfert de données de longues durées.

L'algorithme CSMA/CA est déroulé par l'ensemble des stations d'un BSS, ce qui permet de garantir une chance égale aux différentes stations pour accéder au support. En tant que tel, il n'offre pas de possibilités pour la gestion de la qualité de service.

4. SUPPORT DE DIFFSERV

Plusieurs mécanismes, tels que [4], [5] et [6] ont été proposés pour introduire la différentiation de service dans 802.11b. En effet, [5] et [6] proposent de jouer sur les temps d'inter-frame (les IFS) pour différentier les classes de service : une station avec un trafic prioritaire utilise des IFSs plus courts qu'une station ayant un trafic de basse priorité. Utiliser des IFSs plus courts revient à réduire la durée d'attente et par là même augmenter la probabilité d'accès au support. Il est à noter que toutes ces techniques présentent des compromis du genre ratio de bande passante utilisée, degrés de différentiation entre les classes de priorités hautes et faibles qu'il faut prendre en compte.

Pratiquement toutes ces approches proposent des solutions de niveau lien où il faut modifier la couche MAC pour les intégrer, ce qui rend assez difficile leur faisabilité dans le cadre réel et opérationnel actuel.

Dans des conditions idéales (toutes les stations sont en vue directe, distance raisonnable entre le point d'accès et les stations, pas d'obstacles, etc.), un réseau 802.11b offre un débit efficace de l'ordre de 5 Mbps. La bande passante restante est utilisée par les signaux de contrôle de CSMA/CA. Pire encore, sachant que les performances de CSMA/CA dépendent aussi de la charge imposée [8] par les

clients du BSS, il arrive dans la pratique que le canal soit saturé (CSMA/CA passe son temps à "backoffer") et qu'aucune transmission ne soit possible : il faut bien éliminer de l'information pour éviter ce genre de scénario.

L'idée mise en œuvre ici est d'attribuer des probabilités d'émission à chaque classe de trafic présente dans le BSS pendant une période de longueur *T* secondes. Ces probabilités, appelées ***Pe[k]*** k allant de 1 à 14, seront utilisées par les différentes stations pour décider d'émettre ou de détruire (voir de mémoriser temporairement) un paquet de données lorsqu'il est encore au niveau de la couche réseau, c'est à dire, avant que le système ne le passe à l'interface réseau en question pour l'émettre physiquement). Ces probabilités seront construites par une entité centrale (le point d'accès par exemple) sur la base d'informations envoyées par les stations du BSS pour chaque période *T*. Ces informations décrivent la charge par classe de service (un vecteur comportant 14 entrées) en termes de bande passante. Une fois déterminé, le vecteur ***Pe*** sera transmis aux stations clientes pour être utilisé pendant la prochaine période. On jouera sur ces probabilités pour différencier la probabilité d'accès à la ressource pour les différentes classes de service. A titre d'exemple, nous affecterons à la classe EF la probabilité d'émission 1 et 0.5 à BE. Ainsi, un paquet BE sur deux sera transmit soit un gain de la moitié de la bande passante attribuée à la classe BE qui sera réaffectée à la classe EF. Voici un exemple d'algorithme pour le calcul du vecteur ***Pe*** :

Algorithme *: Allocation des Pe[k]*

bp_eff=802.11(bp efficace)

pour *i* ***de*** *EF* ***à*** *BE* ***faire***
s[i] = bande passnate[i] demandé
si *(s[i] != 0)* ***alors***
si *(bp_eff > s[i])* ***alors*** *bande_allouée = s[i]*
sinon *bande_allouée = bande_efficace*
bp_eff = bp_eff - bande_allouée
Pe[k] = bande_allouée / s[i]
fin si
si *(Pe[k] < 1)* ***alors break***
fin pour

remise-en-échelle(Pe[k])

L'utilisation de la bande passante se fera en fonction du vecteur ainsi calculé et transmis aux clients du BSS. Cette allocation se fait par ordre de priorité, c'est à dire, en traitant les besoins de la classe EF, puis AFxy et enfin BE. Dès qu'il n'y a plus de bande passante disponible, on arrête le processus. Sous 802.11b, il s'agit donc de distribuer une bande passante efficace assez limitée (de l'ordre de 6 Mbps [10]) aux différentes classes de service par ordre de priorité. Il faut garder en esprit que la composante essentielle de notre algorithme est la fonction de calcul des ***Pe***. Elle utilise les structures de données décrivant le trafic présent dans le BSS et calcule efficacement le vecteur des probabilités d'émission. Dans notre cas, nous avons basé les calculs uniquement sur la bande passante requise pour chaque classe de service (sans plus de détails tels que la moyenne de la taille des paquets, la moyenne de l'interarrivée). C'est la fonction donnée en exemple qui sera utilisée pour la simulation de l'algorithme. La dernière étape consiste à appliquer une remise en échelle des différents ***Pe***. Il est ainsi possible d'affiner d'avantage ces probabilités afin de mieux différencier les classes de service et même d'offrir à un opérateur externe la possibilité de les modifier directement.

5. SIMULATION SOUS NS

Le but de la simulation est de comparer les performances en termes de répartition de la ressource radio d'un BSS entre les différents flux. L'algorithme proposé est ainsi comparé avec la solution classique (c'est à dire CSMA/CA pur). Le BSS en question est constitué de 9 stations dont une (la station 0) fait office d'un point d'accès. Quatre trafics CBR (la durée qui sépare deux émissions est constante) sont simulés :

2 vers 7	1er trafic marqué EF à 128 Kbps.
4 vers 8	2eme trafic marqué AF42 à 1.5 Mbps
1 vers 6	3emetrafic marqué AF22 à 2 Mbps
3 vers 5	4eme trafic marqué BE à 5 Mbps

Les différentes stations sont disposées sur un cercle de rayon 35 mètres ayant pour centre le point d'accès. La couche MAC mise en oeuvre est basée sur un modèle simulant le standard 802.11 (ns/mac/802_11.cc [9]). Une couche de niveau liaison est introduite (ns/mac/ll.cc) pour simuler la latence d'accès au support entre 25 µs et 50 µs.

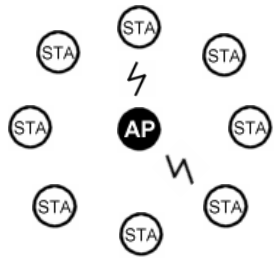

Fig 1 : disposition des nœuds pour la simulation

Finalement, le mécanisme de protection RTS/CTS a été désactivé vu que tous les paquets échangés ont une taille inférieure au seuil au delà duquel le mécanisme est déclenché (RTS_Threshold). Des simulations ont été faites pour des paquets de 192, 256, 384, 512, 768, 1024, 1512 et 2048 octets (les performances de CSMA/CA dépendent de la taille des paquets échangés, de la densité de la BSS en terme du nombre des stations actives, etc.). Chaque simulation dure 120 secondes. Chaque trafic possède une date de début et une date de terminaison. Ces dates sont spécifiées dans le tableau suivant :

Début	Fin	Priorité	Type
0	90	EF	Audio
0	90	AF42	Video
15	120	BE	CBR
40	105	AF22	CBR

Les courbes qui suivent ont été synthétisées à partir des traces recueillies après chaque simulation. Elles représentent la bande passante utilisée par chaque type de trafic (en CSMA/CA pur et modifié) en fonction du temps ainsi que l'interarrivée pour la classe EF, pour des paquets de 512 octets. Finalement, une dernière figure présente le taux d'utilisation (efficacité) du support en fonction de la taille des paquets.

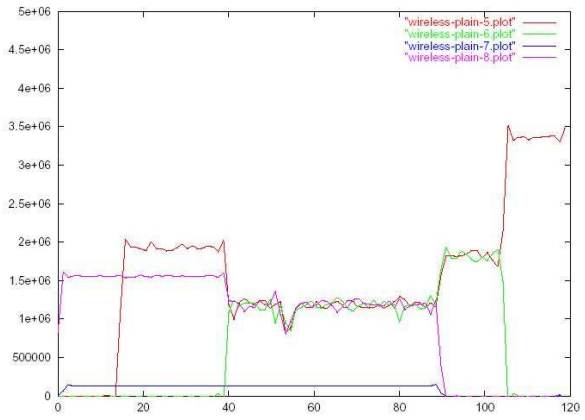

Fig 2 : bande passante utilisée par classe de trafic, 802.11b original, bits/s = f(t)

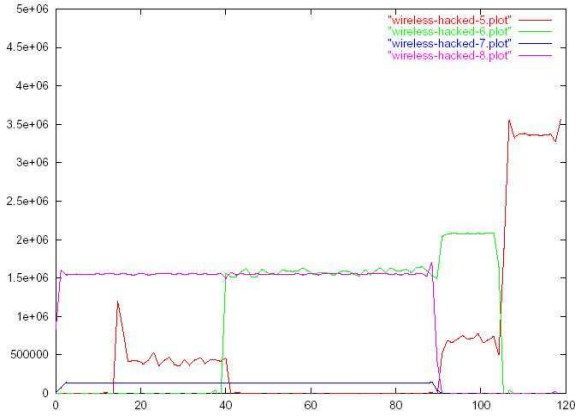

Fig 3 : bande passante utilisée par trafic, 802.11b modifié, bits/s = f(t)

Nous constatons un gain en bande passante significatif pour les classes de service AF22 et AF42. Nous arrivons même à satisfaire, pendant toute sa durée d'existence, le besoin en bande passante de ce dernier au détriment de la classe Best Effort. Un gain relativement important est décelable aussi au niveau de l'interarrivée. En effet, la simulation montre qu'elle est plus "bornée" pour les stations qui implémentent l'algorithme que pour celles qui utilisent le 802.11b original. Ce qui se traduit par une gigue plus faible.

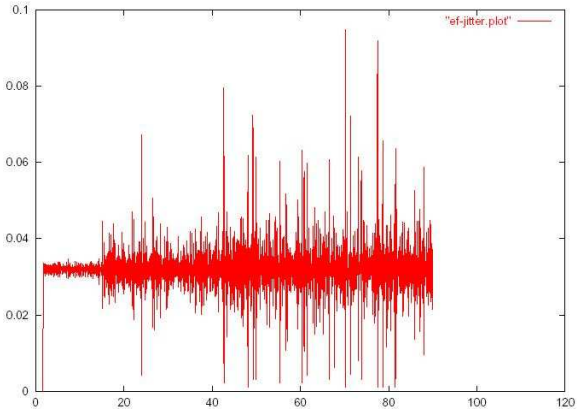

Fig 4 : interarrivée des paquets de la classe EF, 802.11 original, sec = f(t)

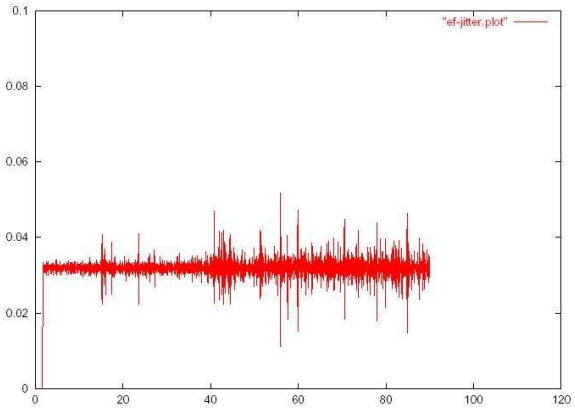

Fig 5 : interarrivée des paquets de la classe EF, 802.11 modifié, sec = f(t)

Cette amélioration est due à la libération du support au profit de la classe EF : en attribuant des probabilités inférieures à 1 pour les classes AF22 et BE, nous empêchons préventivement l'émission de paquets non prioritaires ce qui permet de libérer le support au profit de la classe EF dont la probabilité d'émission vaut 1. Cette amélioration est fonction de la taille des paquets échangés. En effet, en augmentant cette taille, il est possible d'améliorer significativement la qualité du trafic en limitant sensiblement la variation de l'interarrivée.

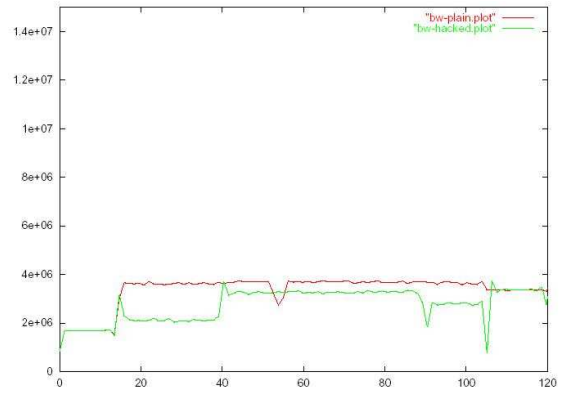

Fig 6 : bande passante totale utilisée par le 802.11 original/modifié, bits/s = f(t)

Cette courbe représente l'utilisation totale de la bande passante par les deux versions du protocole. Nous constatons que ce qui est utilisé par notre algorithme est inférieur à ce qui est normalement offert par 802.11b. Nous avons donc des gains en différentiation par classe de trafic et une perte, de l'ordre de 700 Kbps, sur l'ensemble de la bande passante ; c'est le prix de la différentiation introduite par notre algorithme. Ce prix dépend de la taille des paquets échangés. En faisant varier ce paramètre lors de nos simulations, nous avons obtenu les courbes suivantes. La première exprime le ratio de la bande passante inutilisée en fonction de la taille des paquets (l'activité sur le support varie inversement à la taille des paquets). La seconde courbe traduit la différence entre la moyenne des temps d'arrivées et la moyenne des temps d'inter-émission (trafics CBR -> temps d'inter-émission constant). L'intersection des deux courbes représente le meilleur compromis, en termes de taille de paquet, entre la perte de bande passante introduite par notre algorithme et la déviation par rapport à la valeur d'interarrivée idéale.

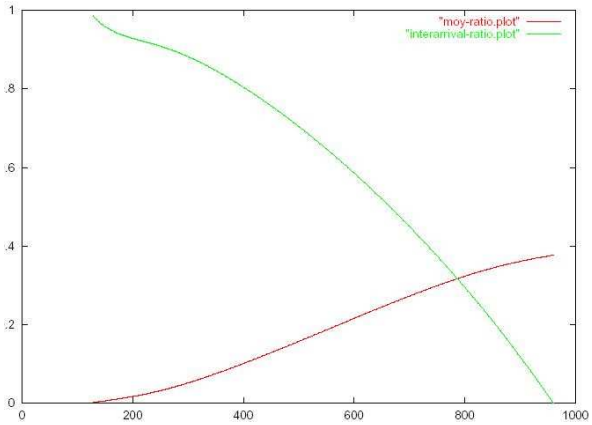

Figure 7 : ratio de bande passante gaspillée en fonction de la taille des paquets échangés / déviation entre interarrivée simulée et interarrivée théorique

Dans notre configuration, la taille des paquets ne doit pas dépasser 800 octets sous peine de perdre une partie importante de la bande passante efficace. De même des paquets de taille inférieure engendrent une variation d'interarrivée, certes moins importante qu'avec du 802.11b pur, mais qui reste toutefois assez importante. La perte en bande passante efficace dépend de l'algorithme de calcul des ***Pe*** ainsi que du nombre des paquets échangés sur une période de temps (i.e l'activité engendrée sur le support). Nous constatons une perte significative pour des paquets de 1024 octets. Ceci est dû à des rejets excessifs de paquets et donc de blocks de 1024 octets (mauvais ***Pe***).

Il faut noter que l'activité sur le support est plus importante avec des paquets de petites tailles qu'avec des paquets de tailles plus importantes (fréquence d'accès au médium plus élevée) étant donné un débit total constant. Les performances de notre algorithme sont affectées par cette activité. En effet, avec des paquets assez petits (inférieur à 192 octets), l'algorithme n'améliore que très légèrement la différentiation. Il sera donc nécessaire de prendre en compte la taille des paquets échangés dans le calcul des ***Pe***.

6. CONCLUSION ET PERSPECTIVES

Dans ce papier, nous avons présenté un mécanisme basé sur le principe d'un rejet sélectif permettant d'assurer une certaine différentiation entre les classes de service définies par DiffServ. Les performances du mécanisme observées lors de simulations sous NS sont discutées. Quelques points tels que l'efficacité de la génération des ***Pe*** et la minimisation de la perte de la bande passante sont à améliorer. Une deuxième alternative, pour le rejet définitif des paquets, basée sur la bufférisation temporaire et la réémission avec un débit adapté est en cours d'étude. Cette bufférisation servira surtout à améliorer et à protéger les performances de TCP.

Cet algorithme, même s'il est assez simple, permet entre autre de prévenir la pénurie en bande passante lorsqu'un nombre important d'utilisateurs se connectent à un BSS donné. En effet, une fois la charge limite (0.8) dépassée, CSMA/CA passera le plus clair de son temps à exécuter des backoff ce qui se traduit, pour les utilisateurs, par des taux de service s'approchant de 0. Notre algorithme permet d'affecter à chaque utilisateur une portion de la bande passante efficace tout en respectant les classes de services utilisées par le BSS. De fait il a une double fonction : éviter l'effondrement des performances de CSMA/CA et protéger les classes de service même dans des conditions de charge extrêmes.

Enfin, une modélisation formelle, basée sur l'augmentation du modèle proposé en [7] ainsi qu'une implémentation sous la forme d'un patch pour le driver HostAP [11], tournant sous Linux, sont en cours d'étude.

7. RÉFÉRENCES

[1] IETF, An expedited Forwarding PHB (Per Hop Behaviour) RFC 3246, Assured Forwarding Group PHB RFC 3260, Per Hop Behaviour Identification Codes RFC 3140. http ://www.ietf.org

[2] *ANSI/IEEE Std 802.11 Part 11 : Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, 1999.

[3] *ANSI/IEEE Std 802.11b Part 11 : Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications : Higher-Speed Physical Layer Extension in the 2.4 GHz Band*, 1999-2001.

[4] A. Veres, M. Barry, L. H. Sun, A. Campbell, "*Supporting Service Differentiation in Wireless Packet Networks using Distributed Control*", 2001.

[5] W. T. Chen, Y. M. Lin, S. C. Lo, "*Priority-Based Contention Control in IEEE 802.11 Wireless LANs*", Departement of Computer Science, National Tsing Hua University, Hsin-Chu, Taiwan 30043, R.O.C.

[6] Dr. J. Deng, R. S. Chang, "*A Priority scheme for IEEE 802.11 DCF Access Method*", IEICE Trans. Commun., Vol. E82-B, No.1, January 1999.

[7] G. Bianchi, "*Performance analysis of the IEEE 802.11 Distributed Coordination Function*", IEEE Journal on Selected Areas in Communications", Vol. 18 Issue : 3, March 2000.

[8] M. Natkaniec and A.R. Pach, "*An Analysis of the Backoff Mechanism used in IEEE 802.11 networks*", Proc. the fifth IEEE Symposium on Computers and Communications, 2000.

[9] NS, Network simulator v2.26, www.isi.edu/nsnam/ns/

[10] J. Jun, P. Peddabachagari, M. Sichitiu, *"Theoretical Maximum Throughput of 802.11 and its applications"*, Second IEEE International Symposium on Network Computing and Applications, April 16 - 18, 2003 Cambridge, Massachusette.

[11] HostAP, http://www.hostap.org, open source driver for Prism2 based WiFi cards for Linux.