Évaluation des performances du système d'agrégation implémenté dans les réseaux 802.11n

Damien Breck

▶ **To cite this version:**

Damien Breck. Évaluation des performances du système d'agrégation implémenté dans les réseaux 802.11n. Automatique / Robotique. Université de Lorraine, 2015. Français. NNT : 2015LORR0278 . tel-01320691v2

HAL Id: tel-01320691
https://hal.science/tel-01320691v2

Submitted on 25 May 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

UNIVERSITÉ DE LORRAINE

École doctorale IAEM Lorraine

Évaluation des performances du système d'agrégation implémenté dans les réseaux 802.11n

THÈSE

soutenance prévue le 4 décembre 2015

pour l'obtention du

Doctorat de l'Université de Lorraine

(Automatique, Traitement du Signal et Images, Génie Informatique)

par

Damien Breck

Composition du jury

Rapporteurs :	Professeur Sébastien Lahaye	Université d'Angers, LARIS
	Professeur Abdelhamid Mellouk	Université Paris Est Créteil, LISSI
Examinateurs :	Professeur David Brie	Université de Lorraine, CRAN
	Docteur Maxime Wack	Maître de conférences HDR, Université de Technologie de Belfort Montbéliard, IRTES
Directeurs de thèse :	Professeur Thierry Divoux	Université de Lorraine, CRAN
	Docteur Jean-Philippe Georges	Maître de Conférences, Université de Lorraine, CRAN

Centre de Recherche en Automatique de Nancy — UMR 7039 / CNRS — UL

Table des matières

Introduction **ix**

1 L'agrégation **1**

1.1 Le besoin d'agréger . . . 1

1.1.1 Une opportunité à l'interface de réseaux hétérogènes . . . 2

1.1.2 Des exigences client et fournisseur difficilement conciliables . . . 4

1.1.3 Considérations supplémentaires à propos des fournisseurs de service . . . 5

1.2 L'agrégation dans le domaine des réseaux et télécommunications . . . 5

1.2.1 Agrégation de topologie . . . 6

1.2.2 Agrégation de porteuses . . . 7

1.2.3 Agrégation des sources de bande passante . . . 8

1.2.4 Agrégation de données . . . 8

1.2.5 Agrégation de trames . . . 9

1.2.6 Agrégation de paquets . . . 10

1.2.7 Synthèse dans le contexte client/fournisseur . . . 11

1.3 L'agrégation de paquets dans 802.11n/ac . . . 12

1.3.1 802.11 de a à ac . . . 12

1.3.2 Formats et particularités des agrégats dans 802.11n/ac . . . 17

1.3.3 Comportement du système d'agrégation 802.11n/ac . . . 18

1.4 Conclusion et problématique . . . 19

2 Évaluations des performances du système d'agrégation **21**

2.1 État de l'art . . . 21

2.2 Positionnement . . . 23

2.2.1 Outil de simulation . . . 24

2.2.2 Théorie de modélisation . . . 24

2.3 Système d'agrégation considéré . . . 25
2.3.1 Un système, plusieurs définitions . . . 26
2.4 Modèle analytique . . . 27
2.4.1 Présentation du calcul réseau . . . 27
2.4.2 Formalisation du système modélisé . . . 29
2.4.3 Courbe du service offert à tous les flux . . . 29
2.4.4 Courbe du service offert à un seul flux . . . 35
2.4.5 Application numérique . . . 37
2.4.6 Discussion de l'ajout d'un seuil temporel . . . 40
2.4.7 Une première conclusion . . . 40
2.5 Pessimisme du modèle établi . . . 41
2.5.1 Cas d'étude . . . 41
2.5.2 Résultats théoriques (IEEE 802.11n) . . . 42
2.5.3 Application numérique . . . 46
2.6 Comparaison des différentes méthodes d'évaluation de performances . . . 49
2.6.1 Résultats du simulateur confrontés à ceux de l'analyse de la norme . . . 49
2.6.2 ...confrontés au modèle analytique établi . . . 52
2.7 Conclusion . . . 53

3 Vers une régulation des paramètres d'agrégation 55

3.1 Périmètre de l'étude . . . 57
3.1.1 Spécification des flux entrants . . . 59
3.1.2 Configuration des équipements . . . 60
3.2 Étude des cas de concurrence avec un scénario simple . . . 62
3.2.1 Un seul flux dans une classe de service . . . 62
3.2.2 Deux classes différentes sur une seule station . . . 63
3.2.3 Une classes différente sur deux stations distinctes . . . 64
3.3 Vers une exhaustivité des scénarios . . . 66
3.3.1 Topologie considérée . . . 66
3.3.2 Définition des flux utilisés . . . 66
3.3.3 Influence du mécanisme de tirage du *backoff* . . . 68
3.3.4 Influence de la taille des paquets . . . 74
3.3.5 Influence de l'inter-arrivée des paquets . . . 83

3.4 Influence des paramètres d'agrégation . . . 86
3.4.1 Sur l'arriéré de traitement . . . 88
3.4.2 Sur le délai . . . 96
3.4.3 Sur les pertes de paquets . . . 106
3.4.4 Discussion . . . 115

4 Configuration du système vis à vis des exigences client et fournisseur 123
4.1 Définition des exigences . . . 124
4.1.1 Du point de vue d'un client . . . 124
4.1.2 Du point de vue du fournisseur d'accès . . . 125
4.1.3 Oppositions entre exigences . . . 125
4.1.4 Autour de la neutralité du Net . . . 126
4.2 Adaptation des paramètres d'agrégation . . . 126
4.2.1 Positionnement scientifique . . . 126
4.2.2 Proposition d'un algorithme de configuration de l'agrégation . . . 127
4.3 Évaluation de l'algorithme proposé . . . 131
4.3.1 Avec notre propre programme . . . 131
4.3.2 Avec Riverbed Modeler . . . 133
4.4 Discussion et perspectives . . . 136
4.4.1 Limites et compromis client/fournisseur . . . 136
4.4.2 Modalités d'implémentation de l'algorithme . . . 137
4.4.3 Un algorithme à adapter . . . 138
4.5 Conclusion . . . 139

Conclusion et perspectives générales 141

Liste des publications 143

Annexes 145

A Définitions et acronymes 802.11n 145
A.1 Paramètres nécessaires au calcul du temps d'émission . . . 145
A.2 Définitions, abréviations et acronymes . . . 145

B Distribution de l'algorithme proposé sur plusieurs stations physiques 147

Bibliographie 149

Table des figures

1.1 Illustration des types de réseaux considérés (accès, distribution et coeur de réseau) 3
1.2 Vue agrégée du réseau lors d'une agrégation de topologie 7
1.3 Multiplexage spécifié pour SDH par l'ETSI 10
1.4 Représentation du mode infrastructure dans 802.11 13
1.5 Un exemple de transmissions avec DCF 13
1.6 Décomposition de la couche MAC d'une station 802.11n 15
1.7 Illustration de la différence de rendement avec et sans agrégation 16
1.8 Format A-MSDU 17
1.9 Format A-MPDU 17
1.10 Comportement du système d'agrégation 802.11n/ac 19
1.11 Configuration du système en fonction d'exigences de QdS 19

2.1 Topologie et explosion combinatoire de méthodes utilisées dans la littérature 25
2.2 Absence d'agrégation lorsque le trafic est faible 27
2.3 Représentation fonctionnelle de la couche MAC 802.11n 29
2.4 Évolution théorique de l'arriéré de traitement dans un système d'agrégation 30
2.5 Enveloppe considérée sur l'arrivée des paquets pour notre modélisation théorique 30
2.6 Cas considérés permettant d'établir un modèle du service offert à un flux particulier 31
2.7 Courbes de services pour différentes tailles d'agrégat 34
2.8 Application directe du service résiduel 35
2.9 Illustration du calcul de δ_1, le temps maximal entre deux agrégats selon notre modèle 37
2.10 Comparaison des courbes de service avec des simulations 37
2.11 Calcul des majorants du délai et de l'arriéré 38
2.12 Calcul d'une valeur de comparaison pour le délai 39
2.13 Topologie pour l'évaluation du pessimisme introduit par le modèle 41
2.14 Étapes d'une transmission avec HCF EDCA 43
2.15 Format d'une trame de données 44

2.16 Format d'une A-MSDU 44
2.17 Format d'une PPDU HT Mixed 44
2.18 Format d'un acquittement 45
2.19 Format d'une PPDU non-HT 46
2.20 Chronogramme entre entrée et sortie de la couche MAC de 802.11n lorsque l'agrégation est active 47
2.21 Statistiques relevées avec Modeler pour une comparaison avec notre étude de la norme . . . 50
2.22 Comparaison d'un relevé du trafic observé sur le réseau sans fil avec une prévision issue de notre étude de la norme 51

3.1 Représentation du système sous forme de boîte noire et positionnement des différentes contraintes, indicateurs et paramètres du système 56
3.2 Position de l'agrégation dans la couche MAC 58
3.3 Schéma fonctionnel de l'environnement du système d'agrégation 58
3.4 Les trois scénarios de concurrence considérés 59
3.5 Un exemple de flux entrant dans la couche MAC 60
3.6 Topologie utilisée pour les simulations Riverbed Modeler 61
3.7 Nombre de paquets par agrégats pour un seul flux dans une seule classe 62
3.8 Arriéré et délai avec et sans agrégation sur une classe de service (500 octets / 50 μs, *backoff* inactif) 63
3.9 Nombre de paquets par agrégats pour deux classes de service sur une seule station 63
3.10 Arriéré et délai avec et sans agrégation sur deux classes de service dans une station (500 octets / 50 μs, *backoff* inactif) 64
3.11 Nombre de paquets par agrégats pour deux classe différentes sur deux stations différentes . 65
3.12 Arriéré et délai avec et sans agrégation sur deux classes de service dans deux stations distinctes (500 octets / 50 μs, *backoff* inactif) 65
3.13 Rappel de la topologie et de l'affectation des flux considérés 66
3.14 Flux utilisés pour l'étude du *backoff* 67
3.15 Cas 1 de l'étude du *backoff* (flux constant $500\upsilon_{50}$ et backoff inactif) 69
3.16 Cas 2 de l'étude du *backoff* (flux exponentiel $500\upsilon_{e50}$ et backoff inactif) 71
3.17 Cas 3 de l'étude du *backoff* (flux constant $500v_{50}$ et backoff actif) 72
3.18 Backoffs tirés en nombre de slots 72
3.19 Cas 4 de l'étude du *backoff* (flux exponentiel $500v_{e50}$ et backoff actif) 73
3.20 Évolution de l'arriéré suite à une diminution de la taille des paquets (*best-effort* agrégé) . . 74
3.21 Évolution du délai suite à une diminution de la taille des paquets (*best-effort* agrégé) 74
3.22 Évolution de l'arriéré suite à une diminution de la taille des paquets (*voice* agrégé) 75
3.23 Évolution du délai suite à une diminution de la taille des paquets (*voice* agrégé) 75

3.24 Comparaison arriéré avec inter-arrivée constante et exponentielle sur 30 tirages 76
3.25 Arriéré de traitement avec un flux de 500 octets / 50 μs 77
3.26 Trafic entrant 78
3.27 Évolution de l'arriéré lors de l'augmentation de taille des paquets (aucun puis deux flux agrégés) 80
3.28 Évolution de l'arriéré lors de l'augmentation de taille des paquets (un seul flux agrégé) . . . 81
3.29 Évolution du délai lors de l'augmentation de taille des paquets (aucun puis deux flux agrégés) 82
3.30 Évolution du délai lors de l'augmentation de taille des paquets (un seul flux agrégé) 83
3.31 Trafic entrant 84
3.32 Évolution de l'arriéré lors de l'augmentation de l'inter-arrivée des paquets (aucun puis deux flux agrégés) 85
3.33 Évolution de l'arriéré lors de l'augmentation de l'inter-arrivée des paquets (un seul flux agrégé) 86
3.34 Délai lorsque l'inter-arrivée des paquets augmente 87
3.35 Évolution de l'arriéré de traitement lorsque aucun flux n'est agrégé ($e125v_{e40}$) 88
3.36 Évolution de l'arriéré de traitement lorsque le flux *best-effort* est agrégé ($e125v_{e40}$) 88
3.37 Évolution de l'arriéré de traitement lorsque le flux *voice* est agrégé ($e125v_{e40}$) 89
3.38 Évolution de l'arriéré de traitement lorsque les deux flux sont agrégés ($e125v_{e40}$) 89
3.39 Évolution de l'arriéré de traitement lorsque aucun flux n'est agrégé ($e250v_{e50}$) 90
3.40 Évolution de l'arriéré de traitement lorsque le flux *best-effort* est agrégé ($e250v_{e50}$) 90
3.41 Évolution de l'arriéré de traitement lorsque le flux *voice* est agrégé ($e250v_{e50}$) 91
3.42 Évolution de l'arriéré de traitement lorsque les deux flux sont agrégés ($e250v_{e50}$) 91
3.43 Évolution de l'arriéré de traitement lorsque aucun flux n'est agrégé ($e500v_{e50}$) 92
3.44 Évolution de l'arriéré de traitement lorsque le flux *best-effort* est agrégé ($e500v_{e50}$) 92
3.45 Évolution de l'arriéré de traitement lorsque le flux *voice* est agrégé ($e500v_{e50}$) 93
3.46 Évolution de l'arriéré de traitement lorsque les deux flux sont agrégés ($e500v_{e50}$) 93
3.47 Évolution de l'arriéré de traitement lorsque aucun flux n'est agrégé ($e1500v_{e150}$) 94
3.48 Évolution de l'arriéré de traitement lorsque le flux *best-effort* est agrégé ($e1500v_{e150}$) . . . 94
3.49 Évolution de l'arriéré de traitement lorsque le flux *voice* est agrégé ($e1500v_{e150}$) 95
3.50 Évolution de l'arriéré de traitement lorsque les deux flux sont agrégés ($e1500v_{e150}$) 95
3.51 Évolution du délai lorsque aucun flux n'est agrégé ($e125v_{e40}$) 96
3.52 Évolution du délai lorsque le flux *best-effort* est agrégé ($e125v_{e40}$) 96
3.53 Évolution du délai lorsque le flux *voice* est agrégé ($e125v_{e40}$) 97
3.54 Évolution du délai lorsque les deux flux sont agrégés ($e125v_{e40}$) 97
3.55 Évolution du délai lorsque aucun flux n'est agrégé ($e250v_{e50}$) 98
3.56 Évolution du délai lorsque le flux *best-effort* est agrégé ($e250v_{e50}$) 99
3.57 Évolution du délai lorsque le flux *voice* est agrégé ($e250v_{e50}$) 99

3.58 Évolution du délai lorsque les deux flux sont agrégés ($e250v_{e50}$) . . . 100
3.59 Évolution du délai lorsque aucun flux n'est agrégé ($e500v_{e50}$) . . . 100
3.60 Évolution du délai lorsque le flux *best-effort* est agrégé ($e500v_{e50}$) . . . 101
3.61 Évolution du délai lorsque le flux *voice* est agrégé ($e500v_{e50}$) . . . 101
3.62 Évolution du délai lorsque les deux flux sont agrégés ($e500v_{e50}$) . . . 102
3.63 Évolution du délai lorsque aucun flux n'est agrégé ($e1500v_{e150}$) . . . 102
3.64 Évolution du délai lorsque le flux *best-effort* est agrégé ($e1500v_{e150}$) . . . 103
3.65 Évolution du délai lorsque le flux *voice* est agrégé ($e1500v_{e150}$) . . . 103
3.66 Évolution du délai lorsque les deux flux sont agrégés ($e1500v_{e150}$) . . . 104
3.67 Délai maximum pour le flux *best-effort* ($e125v_{e40}$) . . . 106
3.68 Délai maximal pour le flux *voice* ($e125v_{e40}$) . . . 107
3.69 Délai maximum, flux *best-effort*, $e250v_{e25}$ (avec pertes) . . . 107
3.70 Délai maximum, flux *best-effort*, $e250v_{e25}$ (sans pertes) . . . 108
3.71 Arriéré deux flux agrégés $e250v_{e25}$ . . . 108
3.72 Délai maximum, flux *voice*, $e250v_{e25}$ (avec pertes) . . . 109
3.73 Délai maximum, flux *voice*, $e250v_{e25}$ (sans pertes) . . . 109
3.74 Délai maximum, flux *best-effort*, $e500v_{e50}$ (avec pertes) . . . 110
3.75 Délai maximum, flux *best-effort*, $e500v_{e50}$ (sans pertes) . . . 111
3.76 Délai maximum, flux *voice*, $e500v_{e50}$ (sans pertes) . . . 111
3.77 Délai maximum, flux *best-effort*, $e1500v_{e150}$ (sans pertes) . . . 112
3.78 Délai maximum, flux *voice*, $e1500v_{e150}$ (sans pertes) . . . 113
3.79 Délai maximum, flux *best-effort*, flux différents (*best-effort* le plus important) . . . 114
3.80 Délai maximum, flux *voice*, flux différents (*best-effort* le plus important) . . . 114
3.81 Délai maximum, flux *best-effort*, flux différents (*voice* le plus important) . . . 115
3.82 Délai maximum, flux *voice*, flux différents (*voice* le plus important) . . . 115

4.1 Représentation du système sous forme de boîte noire et positionnement des différentes contraintes, indicateurs et paramètres du système . . . 123
4.2 Machines à état pour la configuration de l'agrégation sur une classe de service . . . 129
4.3 Implémentation de l'algorithme établi dans Riverbed Modeler . . . 134
4.4 Résultats de simulation de notre algorithme dans Riverbed Modeler sur un scénario issu de l'étude de performances précédente . . . 136

Introduction

La demande croissante de performances et de réduction des coûts pour les réseaux de télécommunications a conduit au développement de divers systèmes dits d'agrégation qu'il est nécessaire d'évaluer. La plupart des avancées récentes en la matière concernent les réseaux sans fil même si le principe d'agrégation existe depuis l'aube des télécommunications modernes comme par exemple avec l'avènement de la Synchronous Digital Hierarchy (SDH).

L'agrégation de trames ou de paquets participe à l'amélioration du rendement offert par le réseau. Cette considération est critique sur les réseaux sans fil actuels puisque la rétro-compatibilité leur impose des entêtes imposantes. Agréger permet donc d'améliorer le rendement de ces technologies en augmentant la part de données utiles encapsulée. Cette amélioration vient s'ajouter aux augmentations des performances brutes offertes notamment dans les dernières version de la norme Wi-Fi (IEEE-802.11n, 2012).

Ce type de système est particulièrement utile lors de la transition entre des réseaux aux capacités différentes. Dès sa conception SDH est une structure hiérarchique de réseaux aux capacités croissantes à mesure que l'on s'approche du cœur de réseau. L'agrégation est effectuée au sein des passerelles entre ces réseaux. Une approche similaire apparaît dans 802.11n mais cette fois l'agrégation s'effectue sur les paquets introduits en couche MAC.

Un système d'agrégation trouve donc naturellement sa place entre les réseaux dits d'accès et les réseaux de distribution. Ce premier réseau est vu comme le lien entre un équipement terminal client et le réseau de son fournisseur d'accès. Aujourd'hui, il est fréquemment constitué d'un réseau sans-fil pour sa flexibilité et sa compatibilité avec de multiples terminaux. Le réseau du fournisseur d'accès peut être découpé en deux parties une première partie qui permet de remonter les informations de tous ses clients et une deuxième qui est son cœur de réseau. La première partie du réseau fournisseur sert également à distribuer les données demandées par ses clients d'où sa dénomination de réseau de distribution. Ce réseau est la plupart du temps filaire et offre des capacités supérieures à un réseau d'accès. L'écart de performances entre ces deux réseaux motive l'utilisation de techniques comme l'agrégation qui permet d'optimiser l'utilisation de la bande passante.

Cette optimisation du débit utile offert à une application est réalisée en permettant à une station de transmettre une plus grande quantité de données à chaque transmission. Il est prévisible qu'elle conduise à une amélioration du service pour l'application concernée mais qu'en est-il des applications concurrentes ? Toutes les applications agrégées ensemble bénéficient elles de ce système ? Ne sont elles pas en concurrence pour l'espace disponible dans un agrégat ? Quid des stations en concurrence pour l'accès au medium avec une station qui agrège ?

Au travers de toutes ces questions nous chercherons à comprendre les implications d'un système d'agrégation dans une passerelle entre deux réseaux aux capacités différentes. Ce système a déjà été étudié en

termes de performances globales. Aussi nous nous positionnons sur une étude plus fine des performances offertes par un système d'agrégation à une application, un flux en particulier.

Le système d'agrégation fonctionne de façon dynamique en se basant sur plusieurs paramètres comme un seuil de taille sur les agrégats. Aujourd'hui, les paramètres sont fixés à l'avance pour un équipement et n'ont pas vocation à être modifiés. Il est cependant compliqué de choisir ces paramètres *a priori* sans connaissance des flux entrant dans le système. Nous proposons donc un algorithme permettant de choisir dynamiquement qui agrège et de fixer les paramètres d'agrégation à partir d'exigences de qualité de service fournies d'une part par le client du réseau et d'autre part par le fournisseur d'accès. Cet algorithme est basé sur une évaluation des performances d'un système d'agrégation en fonction de sa configuration.

Organisation du document

Ce document est divisé en deux parties, la première consiste en une étude du fonctionnement théorique du système d'agrégation et en sa modélisation. Elle est constituée des deux premiers chapitres. La seconde partie couvre les deux chapitres suivants et vise à la proposition d'un algorithme permettant de configurer un système d'agrégation à partir d'exigences du client et du fournisseur d'accès.

Le premier chapitre débute par une étude des multiples significations du terme agrégation dans les réseaux et télécoms. Elle permet de mieux situer le type d'agrégation considéré afin d'éviter la confusion avec d'autres techniques mieux connues (agrégation de données en particulier). Ensuite, nous posons la définition et le contexte du système d'agrégation de paquets implémenté dans 802.11n puisqu'il constitue notre cas d'étude.

Dans le deuxième chapitre nous nous intéressons à un système d'agrégation de paquets indépendamment de la technologie utilisée. Le système est formalisé et nous le modélisons mathématiquement et de façon déterministe avec la théorie du calcul réseau. Le pessimisme de ce modèle est évalué à l'aide d'une étude approfondie de la norme 802.11n et de simulations utilisant Riverbed Modeler.

Dans les chapitres suivants nous proposons un algorithme pour la configuration du système de manière dynamique. Pour cela, nous débutons par une étude de performances étendue à l'aide de Riverbed Modeler. Cette étude nous permet en particulier d'identifier l'influence des paramètres d'agrégation sur les performances du système. Ainsi nous pouvons poser une liste de recommandations pour une configuration efficace de l'agrégation. Ces recommandations nous servent de base pour l'algorithme proposé dans le dernier chapitre. Ce dernier sera évalué d'abord grâce à notre connaissance de la norme puis avec son implémentation dans Riverbed Modeler.

Chapitre 1

L'agrégation

1.1 Le besoin d'agréger

Depuis les débuts d'Internet, les réseaux de télécommunication ont évolué vers une interopérabilité complète qui est aujourd'hui atteinte et forme Internet. En France, la fin des années 80 a été marquée par l'avènement du RNIS (Réseau Numérique à Intégration de Service) soit les débuts de la convergence entre réseaux téléphoniques, réseaux informatiques et réseaux des câblo-opérateurs. Malgré leurs différences de conception, ces réseaux se sont mis à inter-opérer pour former des années plus tard (moyennant quelques évolutions) le réseau Internet actuel. Cette connectivité est rendue possible par les passerelles réalisant l'interface entre des technologies hétérogènes.

Ces interfaces doivent pouvoir évoluer pour assurer un service satisfaisant alors que les besoins en termes de bande passante ou encore de mémoire ne cessent d'augmenter. L'évolution la plus souvent constatée est une augmentation de capacité des équipements passerelles (quantité de mémoire, capacité du medium) par le remplacement d'un de leurs composants (carte réseau) ou même leur remplacement complet. Cette solution, bien que très efficace, coûte très cher à la fois financièrement mais aussi écologiquement à cause de la quantité de déchets électroniques produite.

D'autres solutions existent non pas pour augmenter la capacité d'un équipement mais pour optimiser l'utilisation de ses capacités nominales. En particulier, les mécanismes dits d'agrégation permettent d'améliorer le rendement d'un équipement et/ou d'un protocole dans divers contextes réseau (filaire et sans-fil, au niveau local et mondial) et pour divers objectifs (réduction de l'empreinte mémoire, maximisation de l'utilisation de la bande passante, etc.). Ces mécanismes ont parfois vu le jour lors de la conception même du réseau pour palier les limitations des technologies disponibles à cette époque.

À l'aurore de l'Internet des objets, le nombre d'équipements et de réseaux à interconnecter poursuit son expansion ce qui accroît les besoins en performances des passerelles. L'utilisation de mécanismes d'agrégation est une solution pour absorber l'excédent de bande passante consommée par ces équipements supplémentaires.

L'académie française pose la définition suivante du verbe agréger : "*Unir en un tout ce qui à l'origine était sans liaison*". Cette définition nous renseigne sur l'essence même de l'agrégation qui est de réunir de manière à créer un ensemble. Se posent alors les questions suivantes :

— Quel est cet ensemble ?
— Quels éléments le constituent ?

Et plus spécifiquement vis à vis des problématiques soulevées pour le réseau :

— Comment faire le lien entre l'unité en entrée et en sortie du système ?
— Ce mécanisme permet-il d'améliorer les performances offertes aux clients ?
— Comment le configurer afin d'assurer un progrès en termes de service rendu ?

En cela les exigences des différentes parties prenantes (fournisseur d'accès et client) sont un point clé de la configuration de telles passerelles. Or les mécanismes d'agrégation complexifient la transition d'un monde à l'autre. Ils doivent donc être évalués puis configurés pour respecter les exigences des parties prenantes. C'est tout l'enjeu que nous proposons de relever dans cette thèse.

La section suivante débute avec les considérations technologiques de ces réseaux et la distinction de deux types de réseaux (réseau d'accès et de distribution) en fonction des exigences qui les animent.

1.1.1 Une opportunité à l'interface de réseaux hétérogènes

Nous distinguons trois fonctions dans les réseaux de télécommunication, l'accès, la distribution et le cœur de réseau (figure 1.1). L'accès est le premier lien permettant au client d'accéder au réseau du fournisseur d'accès. Ensuite, les données de ce client remontent jusqu'au cœur de réseau via le réseau de distribution. Elles sont routées dans le cœur de réseau avant d'atteindre le destinataire via un autre réseau de distribution puis un autre réseau d'accès. Le réseau de distribution a souvent une topologie en arbre alors que le cœur de réseau est maillé. C'est une construction usuelle des réseaux qui forment Internet, l'accès suivi d'une distribution hiérarchique et un cœur de réseau maillé.

L'accès peut être constitué d'une cellule GSM (*global system for mobile communication*), d'une paire de câbles téléphoniques (ADSL par exemple), d'un câble coaxial, d'un câble Ethernet, d'une fibre optique ou encore d'un BSS (*basic service set*) dans les réseaux sans-fil 802.11. La distribution et le cœur de réseau sont constitués de routeurs qui peuvent être reliés par des fibres optiques et sur lesquels transitent les données d'un groupe d'utilisateurs. Notons que le réseau de distribution peut également être nommé réseau d'agrégation ((Erman et Zhao, 2011) et (Breuer *et al.*, 2013)), référence à l'agrégation de trames réalisée jusqu'à atteindre le cœur de réseau.

La transition entre accès et distribution est assurée par une passerelle qui peut interagir avec les technologies souvent différentes qui assurent chaque fonction. Les BS (*base station*) du GSM, les DSLAM (*digital subscriber line access multiplexer*) de l'ADSL, ou encore les points d'accès 802.11n sont autant d'exemples de passerelles entre réseaux d'accès et de distribution.

L'accès et la distribution sont assurés par des réseaux de conception souvent différente puisqu'ils sont destinés à respecter des exigences différentes. L'accès doit être offert à de nombreux clients, donc de multiples flux concurrents, avec chacun ses exigences de qualité de service (QdS). La distribution et le cœur de réseau assurent le transport de flux applicatifs plus importants, résultats d'une multitude de flux clients.

Quelque soit le réseau considéré, les fournisseurs d'accès se sont lancés dans une véritable course à la performance afin de satisfaire au mieux leurs clients. Fibres aux capacités toujours plus élevées, antennes nouvelle génération, ... la capacité des réseaux ne cesse d'augmenter. Ces capacités croissantes ont permis à certains clients (les fournisseurs de service) d'augmenter leurs exigences en termes de bande passante disponible afin de proposer de nouveaux services.

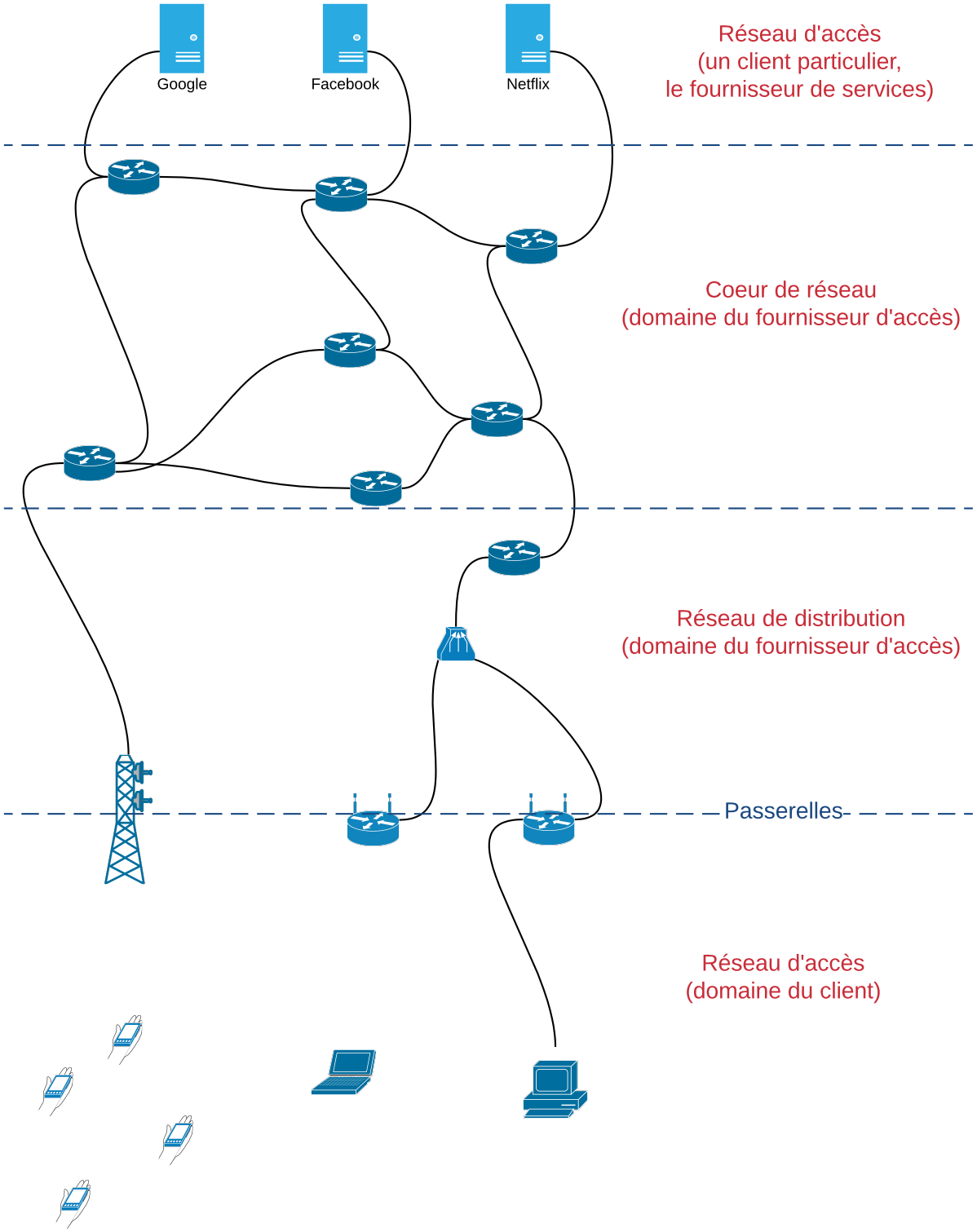


Figure 1.1 – Illustration des types de réseaux considérés (accès, distribution et coeur de réseau)

Dans la suite de cette thèse, nous utiliserons le terme de distribution pour faire référence à la fois au réseau de distribution et à son homologue cœur de réseau puisqu'ils diffèrent souvent uniquement par la topologie utilisée. Or comme nous le verrons par la suite, nous nous intéresserons principalement aux passerelles à l'interface de ces deux réseaux qui ne sont pas influencées qu'elles soient connectées au cœur de réseau directement ou à un réseau de distribution. D'ailleurs nous considérons le fournisseur de service comme un client, un client potentiellement plus important que les autres mais un client tout de même. Ces précisions posées, intéressons nous maintenant aux exigences des deux parties prenantes considérées (client et fournisseur d'accès).

1.1.2 Des exigences client et fournisseur difficilement conciliables

Au delà des flux aux caractéristiques différentes qui les empruntent, les réseaux d'accès et de distribution se démarquent par les exigences parfois divergentes entre client et fournisseur d'accès. Du point de vue du client (côté réseau d'accès) peu importe le nombre de stations sur le réseau, du moment qu'il obtient le meilleur service. Que ce soit en termes de bande passante ou de délai, le client souhaite toujours avoir le meilleur au détriment de ses concurrents si nécessaire. Cependant la capacité offerte par le réseau est limitée et comme le fournisseur d'accès cherche à rentabiliser son réseau, il souhaite utiliser le maximum de bande passante au meilleur prix. Pour rentabiliser son réseau, il peut fournir un grand nombre de clients qui consomment peu de bande passante (comme des particuliers). Une autre stratégie est de fournir seulement quelques fournisseurs de service qui consomment plus de bande passante et qui payeront souvent plus cher pour obtenir des garanties de qualité de service (QdS).

Afin d'imaginer le problème, considérons qu'un gâteau représente la capacité du réseau. Un vendeur (fournisseur) réussit à intéresser une petite foule de personnes qui souhaitent une part de ce gâteau. Il souhaite vendre son gâteau le plus cher possible. Il peut faire de petites parts vendues à petit prix ou quelques grosses parts vendues à des gourmands pour un bon prix. Il devient évident que lorsqu'un gourmand achète une grosse part il en reste moins pour les clients qui suivent qui auront le choix entre se contenter de parts réduites (clients insatisfaits) et aller voir un autre marchand de gâteau (fournisseur insatisfait).

Traduit dans le domaine des réseaux, si une grande partie de la bande passante est occupée par un client aux applications gourmandes (type fournisseur de service de vidéo à la demande) il en reste peu à distribuer aux autres clients.

Le rôle du fournisseur d'accès est ici de trouver un compromis entre toutes ces contraintes pour satisfaire le maximum de clients. Cette problématique d'équité ne trouve pas pour l'heure de solution satisfaisant l'ensemble des parties prenantes. La plupart du temps, les fournisseurs d'accès portent leurs efforts sur une augmentation des capacités offertes en remplaçant leurs équipements.

D'autres approches ont vu le jour et permettent des gains sensibles sans même parfois nécessiter une évolution matérielle qui reste coûteuse. Parmi ces mécanismes se trouvent les techniques dites d'agrégation permettant d'augmenter la bande passante et/ou d'en réduire la consommation. Après un aparté autour de la neutralité d'Internet, les différentes significations du terme agrégation dans le domaine des réseaux et télécommunications seront étudiées et nous les placerons dans le contexte de la relation entre le client et le fournisseur.

1.1.3 Considérations supplémentaires à propos des fournisseurs de service

Aujourd'hui certains fournisseurs de services sont devenus si gourmands qu'ils provoquent des débats autour de la "neutralité du net". La notion de neutralité du réseau implique que chaque client puisse profiter de celui-ci dans les mêmes conditions. La contractualisation des garanties de QdS offertes à un client via un SLA (Service Level Agrement) passé avec le fournisseur d'accès est pourtant la solution actuelle pour partager la bande passante entre clients. Ce contrat permet de lier la QdS offerte à un client avec le montant qu'il est prêt à payer au fournisseur d'accès, ce qui représente une forme d'équité. Même si tous les utilisateurs d'Internet sont soumis à un SLA, celui offert aux particuliers ou aux petites entreprises présente peu ou pas de garanties de QdS. La masse de ces utilisateurs se retrouve alors en forte concurrence et doit partager la bande passante restante après la réservation effectuée par les grands clients.

Puisque le nombre d'utilisateurs ne fait que croître au même titre que les besoins en bande passante des fournisseurs de service, la concurrence s'accroît et les particuliers ont peur de voir leur service s'amenuiser. Bien que cette conclusion semble évidente, le point de vue du fournisseur d'accès n'a pas encore été évoqué. En effet, ce dernier peut facturer des garanties beaucoup plus cher que ce qu'elles représentent en bande passante consommée. Le bénéfice peut ainsi être utilisé pour développer les infrastructures ou appliquer de nouvelles techniques (comme l'agrégation) dans des systèmes vieillissant augmentant par la même la quantité de bande passante offerte à tous les clients. Un client qui n'a donc pas les moyens de payer pour des garanties de QdS pourra se voir offrir un surplus de bande passante grâce aux investissements des fournisseurs de service.

Pour reprendre l'analogie de la section précédente, le fournisseur d'accès cuisinera d'autres gâteaux le lendemain s'il a de nombreux clients ou si un client gourmand achète le gâteau entier.

Même si la discussion de cette problématique de neutralité du réseau se poursuivra à la fin de cette thèse, nous ne nous y intéresserons pas outre mesure. Elle constitue simplement une illustration des problématiques rencontrées par les fournisseurs d'accès et de la potentielle opposition entre les exigences clients et fournisseur d'accès (mais également entre les clients eux-mêmes).

1.2 L'agrégation dans le domaine des réseaux et télécommunications

Dans cette section nous nous sommes intéressés aux sens que pouvait avoir le terme d'agrégation dans le domaine des réseaux de télécommunication. À l'issue de notre étude bibliographique, nous proposons la classification suivante pour l'agrégation.

Il est possible d'agréger :

— une topologie (ex : Asynchronous Transfert Mode).
— des porteuses (ex : réseaux sans fil, HSPA+, LTE Advanced, 802.11n et 802.11ac).
— des sources de bande passante (ex : équipement terminal ayant plusieurs cartes réseau).
— des données (ex : protocoles applicatifs, réseaux de capteurs sans fil).
— des trames (ex : Synchronous Digital Hierarchy).
— des paquets (ex : 802.11 (à partir de sa version n), 802.15.3 (à partir de sa version c)).

Nous détaillerons les différentes formes d'agrégation dans la suite de cette section mais nous pouvons résumer leurs principales caractéristiques comme suit :

Agréger la topologie est une méthode permettant de réduire la topologie logique vue par les équipements du réseau à l'aide d'algorithmes de routage dynamique et d'un calcul de la QdS offerte par les routes établies par ce protocole. Ainsi un ensemble de routeurs et de liens est créé.

Agréger les porteuses consiste à permettre aux stations d'utiliser une largeur de bande d'au moins deux canaux au lieu d'un seul. La nouvelle largeur de bande créée (un ensemble de canaux) peut être utilisée de deux manières : soit l'information est répartie sur les différents canaux mais reste modulée sur une porteuse par canal, soit le signal est modulé sur toute la bande constituée ce qui nécessite que les canaux soient contigus.

Agréger des sources de bande passante consiste à créer un ensemble de canaux (comme précédemment) à ceci près qu'ils peuvent appartenir à différentes technologies. Ce système permet de profiter de plusieurs réseaux différents, le plus souvent sans fil, pour augmenter à moindre coût la bande passante disponible. La complexité réside dans l'algorithme redistribuant les données à travers les différents réseaux disponibles.

Agréger des données consiste à interpréter un ensemble de données et à le réduire à une valeur (en calculant la moyenne de ces données par exemple). Cette agrégation peut être réalisée sur la station source du message ou encore sur un équipement du réseau avec pour objectif de réduire la quantité de données à transmettre au nœud suivant.

Les deux derniers types d'agrégation forment simplement des agrégats à partir d'un ensemble de paquets ou de trames pour réduire la proportion d'en-têtes utilisée et ainsi augmenter le débit utile. La différence entre eux tient simplement au fait que l'agrégation est réalisée sur des SDU [1] (*service data unit*) différentes donc dans des couches différentes, liaison pour l'agrégation de paquets et physique pour celle de trames.

Voyons dans les sections suivantes ce que révèle une étude en profondeur de toutes ces agrégations et des technologies qui les utilisent.

1.2.1 Agrégation de topologie

L'agrégation de topologie (*topology aggregation*) peut être définie comme l'agrégation des informations sur la topologie et/ou des informations de routage pour former un ensemble. Elle repose sur une organisation hiérarchique du réseau. Une topologie logique est créée dans laquelle les routeurs sont groupés (en systèmes autonomes par exemple). Chaque routeur a une vue complète de son groupe et une vue logique agrégée du réseau entier (figure 1.2). Sur cette figure, le réseau complet est illustré à gauche alors que la vue du routeur R1 est illustrée à droite.

La première proposition concernant l'agrégation de topologie est connue sous le nom d'Architecture de Nimrod (Castineyra *et al.*, 1996). Inspiré par ces travaux, l'ATM Forum propose la spécification de la Private Network to Network Interface (PNNI 1.0, 1996) utilisant des méthodes d'agrégation similaires (*Asynchronous Transfer Mode* (ATM)) (IUT-T-I.361, 1999).

ATM n'a pas réussi à s'imposer pour constituer la base d'Internet mais les principes développés lors de sa création sont repris dans d'autres technologies. L'idée d'abstraction hiérarchique de la topologie est reprise notamment avec l'organisation en Systèmes Autonomes dont le fonctionnement est très proche de la PNNI. C'est un principe de base du réseau Internet actuel dont le routage inter-domaine est assuré par le protocole Border Gateway Protocol (BGP) (Rekhter *et al.*, 2006) qui utilise cette abstraction.

1. Une SDU est l'unité élémentaire de donnée échangées entre deux couches d'une pile protocolaire au sein d'un équipement. À opposer à une PDU (*protocol data unit*) échangée entre deux couches identiques sur deux équipements différents.

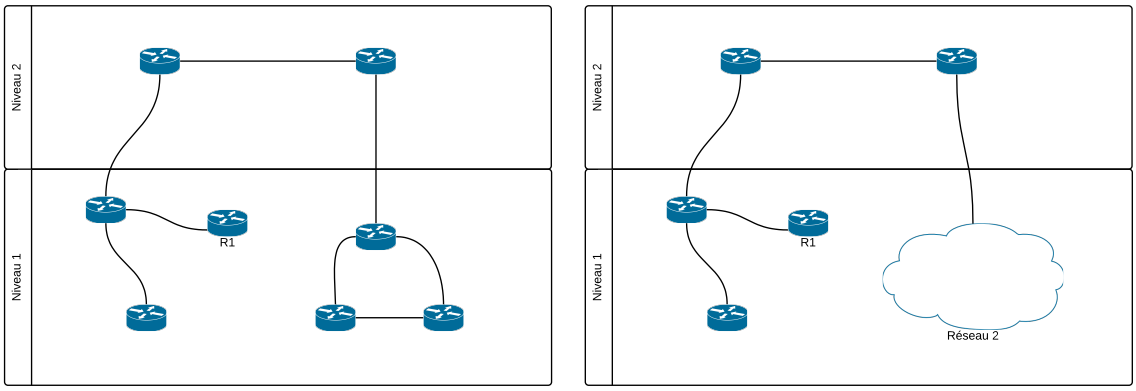


Figure 1.2 – Vue agrégée du réseau lors d'une agrégation de topologie

En plus de la réduction des données traitées, échangées ou stockées, agréger la topologie facilite le passage à l'échelle tout en permettant de conserver des performances de routage similaires (Fang et Zegura, 2000). Elle facilite également la gestion du réseau puisque l'agent à l'œuvre peut surveiller une vue agrégée du réseau.

La complexité d'un tel système repose sur une multitude d'exigences de qualité de service (QdS) à considérer lors de la conception d'algorithmes de routage et d'agrégation (Iwata *et al.*, 1998). C'est pourquoi de nombreux travaux ont porté sur l'évaluation de ces algorithmes avec des paramètres de QdS comme le délai et la bande passante (Lee, 1995), (Lui *et al.*, 2004), (Korkmaz et Krunz, 2000). D'autres études sont recensées dans (Uludag *et al.*, 2007) qui introduit également les motivations ayant conduit à l'utilisation de l'agrégation de topologie.

Hiérarchiser la topologie du réseau et en agréger la vue pour ses routeurs répond à un besoin du fournisseur d'accès. Ce besoin est issu de la multitude de services proposés aux clients par les fournisseurs de services. En ce sens, une telle méthode ne provient pas d'une demande de la part du client mais d'une volonté du fournisseur d'étendre ses services. Le client bénéficiera à la marge d'améliorations de ses délais et de sa bande passante grâce à une diminution des échanges entre routeurs. Ce qui ne constitue pas l'objectif initial qui consiste à réduire la quantité de mémoire consommée dans les routeurs afin d'optimiser leur usage et ainsi retarder leur fin de vie.

1.2.2 Agrégation de porteuses

Sur un réseau sans-fil, l'agrégation de porteuses (*carrier aggregation*) consiste à utiliser au moins deux canaux pour une transmission. C'est un moyen simple d'augmenter la bande passante disponible pour un équipement. Il existe principalement deux méthodes permettant de profiter de ces canaux multiples (Yuan *et al.*, 2010) :

— Les données à transmettre sont réparties sur plusieurs porteuses (une par canal), modulées comme s'il n'y avait pas d'agrégation et réassemblées à la réception (HSPA+, LTE Advanced)
— Les bandes de fréquences contiguës sont agrégées et le signal est modulé sur l'agrégat (IEEE 802.11n et 802.11ac)

La seconde technique ayant la densité spectrale la plus élevée, elle permet des débits crête plus élevés. C'est pourquoi elle a été retenue pour 802.11n et 802.11ac. Il est plus compliqué de la mettre en œuvre sur les réseaux mobiles (HSPA+ ou LTE-Advanced) puisque les bandes de fréquences des opérateurs peuvent être fragmentées (Yuan *et al.*, 2010), (Iwamura *et al.*, 2010). De plus cette technique ne permet pas une rétrocompatibilité avec les précédentes normes (IEEE-802.11n, 2012) ce qui était requis pour les réseaux LTE (Pedersen *et al.*, 2011).

Ici le bénéfice est direct pour le client puisque c'est principalement sur le réseau d'accès que cette technologie est utilisée. L'augmentation de bande passante est très significative et permet aux réseaux mobiles d'être vecteurs de nouveaux services. Des études portant sur les performances d'un tel système sont proposées par (Johansson *et al.*, 2009) et (Pedersen *et al.*, 2011).

1.2.3 Agrégation des sources de bande passante

Agréger les sources de bande passante (*bandwidth aggregation*) consiste à utiliser plusieurs canaux pour transmettre le message tout comme pour la technique d'agrégation de porteuses évoquée en section précédente. La différence réside dans l'agrégation ici de technologies/interfaces réseau différentes. Utiliser simultanément les différentes cartes réseau d'un appareil est une solution plutôt novatrice développée en particulier par l'IMT-Advanced qui donne les recommandations pour la 3G et ses successeurs (3G+, 4G). Le point qui nous intéresse ici est situé section 4.2.6 de (IUT-R-M.1645, 2003) qui s'intéresse aux réseaux d'accès et évoque la possibilité de leur utilisation simultanée. Puisque les terminaux mobiles actuels disposent de multiples interfaces de communication vers des réseaux différents (4G et 802.11ac par exemple), cette agrégation permettrait d'augmenter à faible coût la bande passante disponible pour l'utilisateur. C'est pourquoi le terme d'agrégation de sources de bande passante est utilisé.

Agréger les sources de bande passante permet :

— d'augmenter la bande passante
— d'améliorer la fiabilité de transmission des paquets
— de distribuer/répartir la charge (*load-balancing*)
— d'augmenter la capacité du réseau d'accès à faible coût

Les problèmes majeurs sont le rétablissement de l'ordre des paquets à la réception et l'augmentation de la consommation énergétique de l'appareil sollicité (Ramaboli *et al.*, 2012). D'autres études portent sur la tolérance aux pertes de ce type de système (Wu *et al.*, 2013) ou leur compatibilité avec des applications temps-réel (Chebrolu et Rao, 2006). La façon dont les incertitudes du réseau peuvent être gérées (Oliveira *et al.*, 2011) sont toujours des problèmes majeurs.

Les bénéfices en terme de performances vont donc directement au client car encore une fois le gain est situé sur le réseau d'accès. Le problème le plus important pour lui sera l'augmentation de la consommation de son appareil due à l'utilisation simultanée de plusieurs cartes réseau.

1.2.4 Agrégation de données

L'agrégation de données (*data aggregation*) vise à réduire la quantité de données applicatives. Elle est utilisée en particulier dans les réseaux de capteurs sans-fil pour lesquels elle permet d'augmenter la durée de vie des équipements ce qui la rend très intéressante. Dans de tels réseaux, la plupart des informations

mesurées par les capteurs sont redondantes et corrélées et si le réseau est important, la quantité de données à traiter par la destination est énorme. Il est donc nécessaire d'utiliser des méthodes d'agrégation de ces données pour réduire le nombre de paquets transmis à la destination (souvent nommée puits) (Rajagopalan et Varshney, 2006).

Agréger les données suppose souvent une perte d'information due aux algorithmes employés. Cet agrégation permet d'adapter la quantité de données à transmettre à la bande passante disponible sur le réseau ou encore d'optimiser certains critères comme la consommation d'énergie dans les réseaux de capteurs sans fil (Intanagonwiwat *et al.*, 2002), (Krishnamachari *et al.*, 2002). Dans ces derniers l'économie d'énergie est un enjeu vital qui assure la durée de vie du réseau lui-même, et agréger les données peut prolonger cette durée de vie.

En effet, la topologie du réseau est souvent hiérarchique avec au sommet le nœud puits, destination de tous les messages. Comme les nœuds ont tendance à se raréfier à mesure que l'on s'approche du puits, ces derniers consommeront plus de bande passante mais également plus d'énergie que leurs voisins plus éloignés du puits. Or si ces nœuds viennent à épuiser leur énergie c'est toute la route vers le puits qui est remise en question. Il est donc plus équitable que la quantité de données passant par ces nœuds-relais soit moindre que celle traversant ceux plus éloignés. Dans certains cas, une augmentation de la consommation du processeur est négligeable au regard de celle de la partie radio de ces capteurs (Perrig *et al.*, 2002). Il est dès lors acceptable de consommer un peu plus de processeur pour effectuer une opération réduisant la taille du message. Cette remarque est vérifiée en particulier pour des opérations simples comme une moyenne ou une compression simple. Pour des opérations plus complexes la consommation du processeur peut être largement supérieure à celle de la transmission (Duran-Faundez, 2009).

L'agrégation de donnée est devenue un paradigme indispensable dans les réseaux de capteurs sans-fil (Krishnamachari *et al.*, 2002). Cependant l'application de telles techniques n'est possible qu'avec une maîtrise totale des équipements du réseau et en particulier de leur logiciel ainsi que de la nature de l'information échangée.

Ce type d'agrégation des données suppose également une connaissance parfaite des exigences du client ainsi que des caractéristiques des flux. Dès que la teneur des données échangées au sein des différents flux est inconnue, il est difficile voire impossible de juger de la pertinence d'une agrégation de celles-ci. La solution alors privilégiée est de transmettre la totalité du message originel. Pour les réseaux de capteurs sans fil, la quantité de données à transmettre est plus importante que les en-têtes car les protocoles utilisés ont été spécifiquement pensés pour cela. Cependant, il y a beaucoup à gagner sur les en-têtes nécessaires aux protocoles utilisés pour des applications grand public, d'où le développement des agrégations étudiées en sections suivantes.

1.2.5 Agrégation de trames

Au niveau physique, certaines technologies (IUT-T-G.707/Y.1322, 2007; UIT-T-G.708, 1999) réalisent l'agrégation de trames (*aggregation network*) au sein des nœuds qui mènent à leur cœur de réseau.

Synchronous Digital Hierarchy (SDH), la version européenne de Synchronous Optical NETwork (SONET), était prévue pour l'interconnexion des réseaux d'opérateurs téléphoniques. Un certain isochronisme était donc nécessaire. Cette exigence a imposé une topologie hiérarchique basée sur des capacités de lien

croissantes à mesure que les flux sont agrégés (vers le cœur de réseau). Sur ce réseau une trame est transmise toutes les 125 μs quelles que soient les données à transmettre.

Si elle reste simpliste dans son fonctionnement, cette agrégation est un bon exemple du travail fourni par l'opérateur pour satisfaire les exigences client. La technologie SDH n'était pas prévue pour accueillir des données autres que la voix mais elle a su s'adapter avec le temps. La problématique de départ est différente des autres types d'agrégation puisque nous avons ici agrégation des trames par la spécification de la topologie et des capacités des liens sur le réseau imposée par l'ETSI (European Telecommunications Standards Institute).

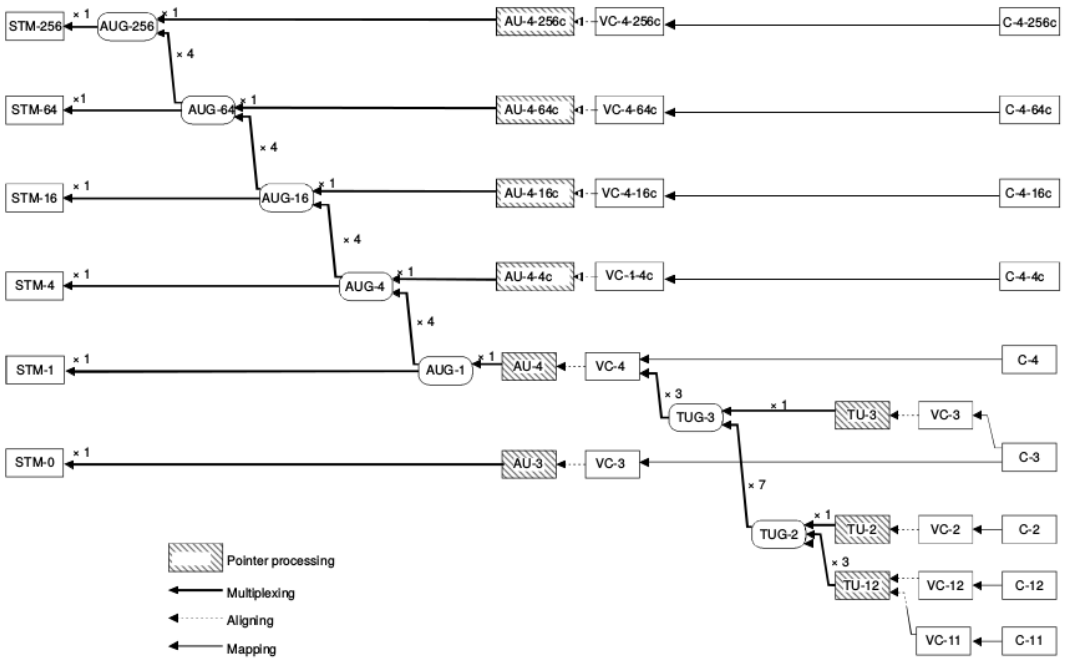


Figure 1.3 – Multiplexage spécifié pour SDH par l'ETSI

L'adoption du schéma proposé par (ETS 300 147, 1992) (repris figure 1.3) ajoutée à un dimensionnement correct du réseau permet par construction d'échapper à toute problématique d'évaluation de performances. En effet, ce réseau ne souffre d'aucune congestion puisque la capacité du lien de sortie est à minima égale à la somme des capacités des liens en entrée du réseau. La congestion est donc déplacée à l'entrée de ce réseau et peut être gérée par des mécanismes de contrôle d'admission.

1.2.6 Agrégation de paquets

L'idée derrière l'agrégation de paquets (*packet aggregation*) est la même que celle de l'agrégation de trames de la précédente section. Elles diffèrent par les couches sur lesquelles elles sont appliquées mais également par leurs modalités de mise en œuvre. Si l'agrégation de trames est réalisée par construction de l'architecture du réseau, l'agrégation de paquets est un mécanisme présent au niveau de la couche liaison qui possède son propre algorithme de décision.

Comme dit précédemment, une économie substantielle peut être réalisée sur les en-têtes, en particulier au niveau de la couche MAC (*medium access control*) et d'autant plus sur les réseaux sans fil.

D'autres techniques ont été utilisées pour réduire l'impact de ces en-têtes comme la variation de leur contenu en fonction des équipements sur le réseau dans 802.11n. L'interopérabilité entre équipements est assurée par une partie de l'en-tête compréhensible de tous et transmise au débit de base minimal supporté par toutes les stations en présence. Cependant, cette réduction d'en-têtes augmente seulement de façon marginale la bande passante disponible sur le réseau. On retrouve le même objectif dans la définition de *6lowPan* (Hui et Thubert, 2011) qui propose une compression des en-têtes IPv6 afin de permettre leur utilisation dans des réseaux de capteurs sans fil.

L'agrégation de paquets est un mécanisme qui s'attache à augmenter la proportion de données utiles dans une trame. Son fonctionnement repose simplement sur l'encapsulation de plusieurs paquets dans une seule trame.

Ce mécanisme est utilisé dans 802.11n ce qui a permis d'améliorer le débit utile au niveau MAC (Skordoulis *et al.*, 2008) jusqu'à atteindre les 100 Mbits/s, objectif du groupe de travail. L'agrégation de paquets est étudiée dans des contextes aussi variés que la virtualisation dans le *cloud computing* (Bourguiba *et al.*, 2012), la commutation de réseaux optiques (Khattab *et al.*, 2002), les réseaux de capteurs sans-fil (Petrovic *et al.*, 2003), le protocole de réservation PRMA (*Packet Reservation Multiple Access*) (Zhang *et al.*, 2008), dans les systèmes UWB (*Ultrawideband*) (Lu *et al.*, 2007) ou encore dans 802.15.3c (IEEE-802.15.3c, 2009)(An *et al.*, 2009). Certaines études portent même uniquement sur le principe d'agrégation de paquets indépendamment de la technologie sur laquelle elle sera utilisée (Hong et Sohraby, 2010).

1.2.7 Synthèse dans le contexte client/fournisseur

Un point ressort pour chacune des agrégations définies en section précédente : la multitude d'exigences à considérer est un frein à leur développement. Ces exigences proviennent des deux parties en présence : le client et le fournisseur d'accès. C'est pourquoi nous nous intéressons à leurs points de vue et en particulier aux moyens de trouver un compromis entre leurs exigences souvent divergentes.

Agrégation	Impact sur le client	Impact sur le fournisseur
topologie	faible augmentation de la bande passante disponible	forte réduction de la mémoire consommée dans les routeurs, augmentation de leur durée de vie
porteuses	forte augmentation de bande passante, consommation d'énergie de l'équipement plus élevée, abonnement plus cher	montée en gamme au niveau des services et tarifs mais rafales plus importantes à gérer sur le réseau de distribution
sources de bande passante	forte augmentation de bande passante, consommation d'énergie de l'équipement plus élevée	rafales plus importantes à gérer sur le réseau de distribution
données	perte de précision, gain en bande passante disponible	réduction de la bande passante consommée, durée de vie du réseau augmentée, calculs potentiels sur les nœuds réseau
trames	service fiable	réseau facile à maintenir mais difficile à faire évoluer
paquets	augmentation de la bande passante	augmentation de la bande passante

Tableau 1.1 – Impact des différentes agrégations sur le client et le fournisseur d'accès

Le tableau 1.1 synthétise les impacts identifiés pour les différentes agrégations sur le client et sur le fournisseur d'accès. Parmi toutes les agrégations considérées, l'influence d'une agrégation de paquets sur le réseau dans lequel elle opère n'est pas encore bien connue. C'est en partie ce qui motive notre choix de l'étudier.

Nous avons vu que le terme agrégation peut revêtir de multiples significations même dans le domaine restreint des réseaux de télécommunication. De l'agrégation de topologies à l'agrégation de paquets, chacune apporte son lot d'améliorations et de problématiques. Si l'agrégation de topologie permet des gains mémoire substantiels elle ne participe en rien à l'interopérabilité entre réseaux hétérogènes. L'agrégation de données, quand à elle, est restreinte à des applications maîtrisées afin d'assurer une réduction sémantiquement correcte de la quantité de données transitant sur le réseau. L'agrégation sans perte d'information de niveau physique proposée par SDH est bien trop rigide, demande une maîtrise totale de la partie physique et de la topologie et ne présente pas une complexité nécessitant de l'étudier plus en détail (excepté dans l'optique de définir un contrôleur d'admission).

L'agrégation de paquets est un système plus complexe, plus flexible et pouvant être utilisé afin d'améliorer la transition entre deux mondes différents. Compte tenu du caractère polyvalent de cette agrégation et des améliorations de performances démontrées dans les études citées précédemment, c'est sur celle-ci que portera cette thèse. Afin de cibler l'étude, nous avons choisi d'étudier en particulier l'agrégation de paquets utilisée dans 802.11n. La définition de ce type d'agrégation et l'histoire de son développement pour IEEE 802.11n sont détaillées en section 1.3.

1.3 L'agrégation de paquets dans 802.11n/ac

Avant d'introduire l'agrégation de paquets dans 802.11n/ac, il est utile de comprendre dans les grandes lignes le développement de la norme 802.11 puisque chaque version n'est qu'un amendement de l'original.

1.3.1 802.11 de a à ac

À ses débuts, ce protocole était développé avec pour seul objectif de proposer un protocole de communication sans-fil pour les réseaux locaux. Ce protocole devait assurer la possibilité à plusieurs équipements de se connecter (à un point d'accès ou entre eux[2]) afin de créer dynamiquement un réseau sur un canal prédéfini aussi appelé BSS (*Basic Service Set*). Le BSS, associé au DS (Distribution System) forment un ESS (Extended Service Set) comme illustré en figure 1.4.

Le problème consiste alors à trouver un moyen de gérer la concurrence pour ce canal. À une époque ou l'algorithme CSMA/CD (*carrier sense multiple access / collision detection*), utilisé dans le protocole Ethernet est très répandu dans les réseaux filaires et fonctionne à merveille, il est logique de chercher à s'en inspirer.

Le protocole CSMA/CA (*carrier sense multiple access / collision avoidance*) est donc créé (voir algorithme 1). Il s'inspire de CSMA/CD mais la détection de collisions se transforme en évitement de collisions (illustré par un exemple figure 1.5). En effet, il est physiquement impossible de transmettre et d'écouter le medium simultanément sur une même antenne. Cette contrainte physique influe sur le développement

2. mode ad hoc.

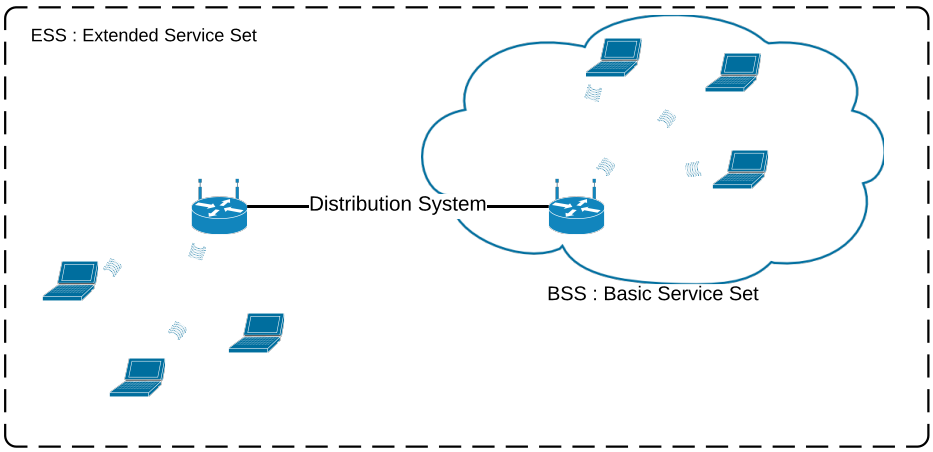


Figure 1.4 – Représentation du mode infrastructure dans 802.11

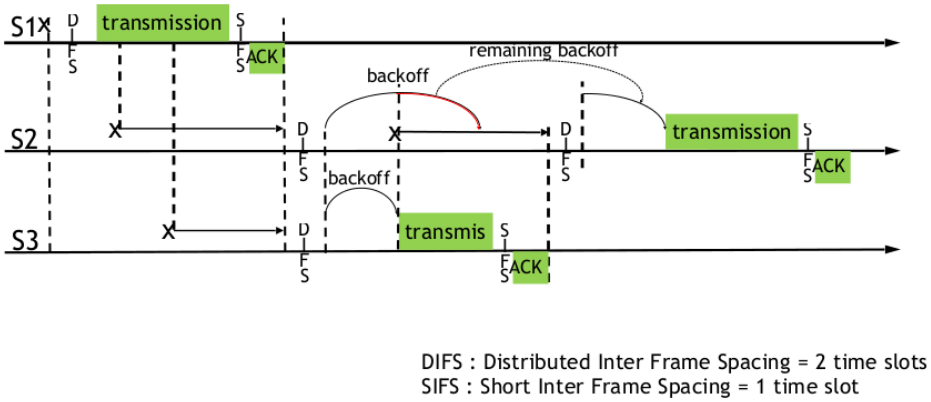


Figure 1.5 – Un exemple de transmissions avec DCF

de l'algorithme d'accès au medium au niveau MAC. De plus des stations sans fil sur le même BSS font face au problème de la station cachée qui peut provoquer des collisions indétectables pour les auteurs des transmissions en collision.

Impossible alors de s'inspirer des réseaux filaires et de l'algorithme CSMA/CD en particulier pour gérer les collisions puisqu'elles ne peuvent être détectées. CSMA/CD comme son nom l'indique repose sur la détection des collisions par une écoute simultanée à la transmission. La collision détectée sera résolue par le tirage d'un temps d'attente aléatoire par chacune des stations sur le bus avant de reprendre la transmission. Dans CSMA/CA, ce temps d'attente aléatoire (aussi appelé *backoff*) est tiré avant toute transmission qui en suit une autre (le médium n'était pas libre lorsque la station à voulu transmettre). Cet algorithme repose sur l'espoir qu'une éventuelle autre station tirera un *backoff* différent.

Données : état du médium
Résultat : décision de transmission
initialisation ($\tau \leftarrow$ durée de la tranche canal ; $cw \leftarrow cw_{min}$);
début
  écouter la voie;
  si *la voie est libre pendant un temps donné (DIFS)* **alors**
    transmission;
    si *collision* **alors** $cw = \max(cw \times 2, cw_{max})$; reprendre l'algorithme du début;
  sinon
    /* choisir un temps d'attente aléatoire (backoff) dans $[0, cw[$ */
    $backoff \leftarrow random(0, cw) \times \tau$;
    tant que *backoff > 0* **faire**
      tant que *médium libre et backoff > 0* **faire**
        /* décompter des intervalles de temps (tranche canal) */
        $backoff \leftarrow backoff - k \times \tau$;
      /* suspension du décompte en attendant */
    transmission;
    si *collision* **alors** $cw = \max(cw \times 2, cw_{max})$; reprendre l'algorithme du début;

Algorithme 1 : Algorithme CSMA/CA

Ensuite, comme le BER (*bit error rate*) est beaucoup plus élevé sur un réseau sans fil, il devient absolument nécessaire de mettre en œuvre un moyen d'être sûr que le destinataire à bien reçu le message (d'autant plus avec le problème de la station cachée évoqué auparavant). Les acquittements sont nés. Tous ces mécanismes réunis forment la méthode d'accès au medium la plus utilisée dans 802.11, DCF (*distributed coordination function*) qui est d'ailleurs la seule méthode obligatoire pour tout équipement qui veut être normalisé 802.11. Comme son nom l'indique, elle est distribuée et permet aux équipements de partager l'accès au medium sans recourir à un contrôleur centralisé. Des intervalles de temps spécifiques sont définis en fonction du débit de base proposé dans 802.11 soit 1 Mb/s. Ces intervalles (DIFS (*distributed inter frame space*) et SIFS (*short inter frame space*)) permettent de s'assurer qu'aucune autre station n'émet avant de débuter la communication (voir l'exemple de transmission figure 1.5.

Plus tard, les membres du groupe de travail 802.11 ont intégré un mécanisme de classification de service. Ce mécanisme défini dans l'amendement 802.11e de la norme, propose de classer les paquets entrant dans le système dans quatre classes de service (AC) distinctes qui sont liées aux priorité utilisateurs (UP) définies par le protocole de couche supérieure comme défini en tableau 1.2. Chacune de ces files est en concurrence avec les autres et les priorités entre elles sont réglées statistiquement par une modification de l'intervalle de tirage de cw pour chaque classe. cw étant la borne supérieure de l'intervalle de tirage du *backoff*. Cette borne est modifiée en fonction des collisions qui se produisent sur le réseau, elle évolue dans l'intervalle $[cw_{min}, cw_{max}]$. Le DIFS est lui aussi modifié et devient un AIFS (*arbitration inter frame space*) qui est différent pour chaque classe (voir tableau 1.2 pour les valeurs de AIFSN (AIFS *number*)) [3].

Une classe de service 802.11e est considérée comme une station virtuelle (montrée en figure 1.6) ce qui signifie que chaque station physique se transforme en quatre stations virtuelles avec cet amendement. Dans

3. sachant que $AIFS = SIFS + AIFSN \times SlotTime$

UP	AC	AIFSN
1,2	BK	7
0,3	BE	3
4,5	VI	2
6,7	VO	1

Tableau 1.2 – Mapping entre classe de service (AC) et priorité utilisateur (UP)

802.11e, la notion de TXOP (*transmission opportunity*) est également introduite. Une TXOP est définie par un instant de début et une durée maximale. Cette limite de temps varie en fonction de la classe de service qui obtient l'accès au medium. Une station peut transmettre plusieurs trames supplémentaires dans une TXOP à condition que la durée de celle-ci ne soit pas révolue après la réception de l'acquittement positif de la trame précédente. Une durée nulle de TXOP signifie que la classe est autorisée à transmettre une seule trame à la fois avant de remettre en jeu l'accès au medium. Tous ces mécanismes sont inclus dans une extension du mode DCF, HCF EDCA (*hybrid coordination function enhanced distributed channel access*) [4].

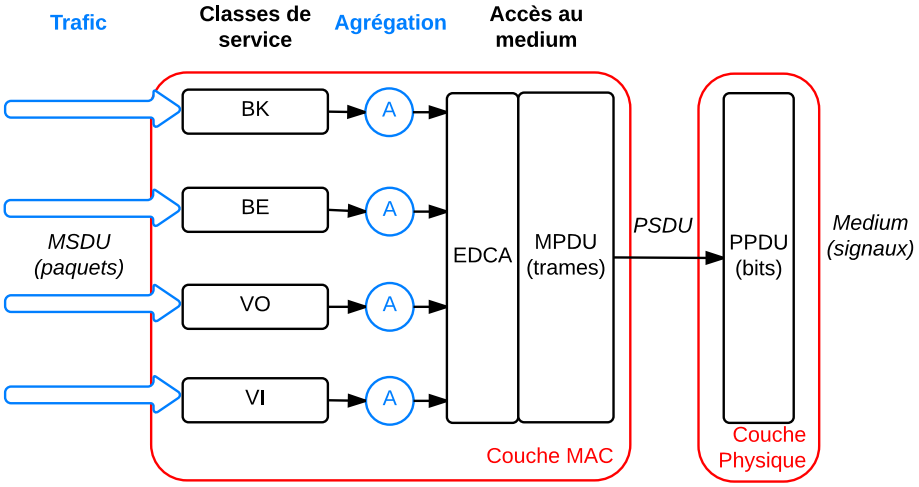


Figure 1.6 – Décomposition de la couche MAC d'une station 802.11n

Le protocole permet donc de classer les paquets en fonction de priorités. Il reste alors à améliorer le débit utile offert. C'était l'objectif principal du groupe de travail 802.11n, un débit utile supérieur à 100 Mb/s en sortie de la couche MAC. Pour atteindre cet objectif, de nombreuses améliorations ont été réalisées au niveau de la couche Physique (multiples antennes, *beamforming*, largeur de bande doublée à 40MHz, utilisation de fréquences plus élevées à 5GHz ...). Même si les progrès sont très significatifs, il subsiste un problème de rendement au niveau de la couche MAC. En effet, le rendement de cette couche qui est de 42 % à 54 Mbits/s, descend jusqu'à 10 % à 432 Mbits/s (Li *et al.*, 2009). Ce rendement entre données utiles et en-têtes est grevé par les en-têtes physiques qui assurent la rétrocompatibilité du système avec ses anciennes versions. Lorsque le débit physique augmente, la transmission des données utiles de la trame MAC est effectivement plus rapide. Cependant, une part de l'encapsulation (comme les en-têtes physiques et le temps d'accès au medium) n'est pas impactée par cette augmentation de débit. Leur proportion sur le temps de transmission d'une trame augmente donc et ce même en considérant un canal idéal.

4. HCF propose deux autres méthodes d'accès au medium, PCF (*point coordination function*) et HCCA (HCF *controlled channel access*) qui reposent toutes deux sur un contrôleur central distribuant l'accès au medium.

Ajouté à ces considérations, l'accès au medium est pénalisant puisque, comme dans tout réseau sans-fil, des collisions se produisent avec les stations concurrentes et la distance avec le point d'accès va influencer les débits. Du point de vue d'une station, il est donc intéressant lorsqu'elle obtient l'accès au medium de pouvoir transmettre autant de données que possible. Agréger va permettre de réduire le nombre de transmissions et donc le nombre de fois où l'accès au canal est remis en jeu. Ce principe est illustré figure 1.7 où seuls six paquets sont transmis contre onze lorsque l'agrégation est active.

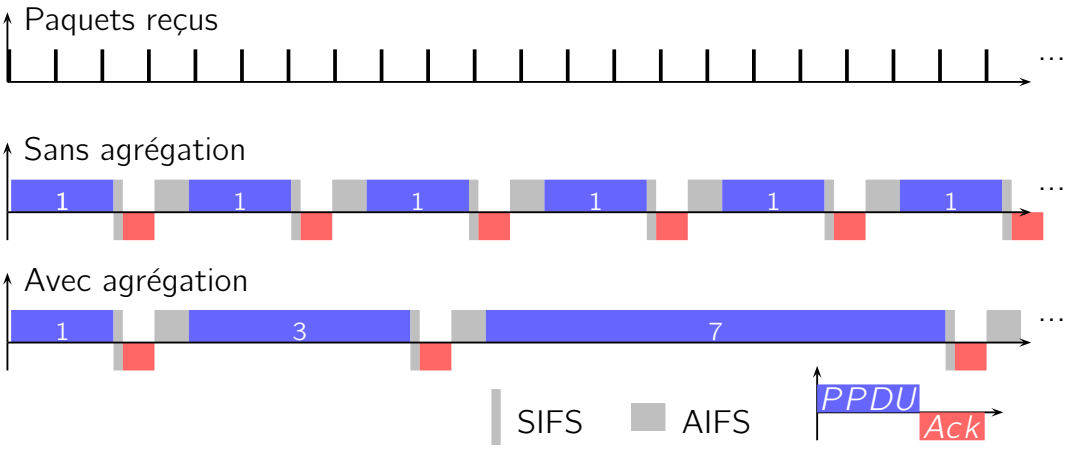


Figure 1.7 – Illustration de la différence de rendement avec et sans agrégation

De plus le rendement offert à l'application est amélioré par l'utilisation de l'agrégation. Sans détailler les calculs maintenant, nous pouvons évaluer le rendement avec et sans agrégation sur le cas simple illustré en figure 1.7. La capacité du medium dans ce cas est de 65 Mb/s. Les paquets font une taille de 500 octets à l'entrée de la couche MAC et appartiennent à la classe *best-effort* (soit un AIFS de 37 μs). Les en-têtes MAC représentent un total de 34 octets. Nous pouvons ainsi calculer le temps d'occupation du medium pour transmettre un paquet lorsque la station n'agrège pas (équation (1.1)). Le *backoff* considéré vaut 135 μs ce qui est une valeur moyenne pour cette classe de service. Le temps de transmission d'un acquittement est de 34 μs et un SIFS vaut 10 μs.

$$\begin{aligned} T_{noagg} &= AIFS + backoff + PHY + \frac{MAC + donnes}{capacit} + SIFS + ACK \\ &= 37 + 135 + 110 + 10 + 34 = 326\ \mu s \end{aligned} \tag{1.1}$$

Le débit utile offert pour un paquet est donc de $\frac{8*500}{326} = 12,26$ Mb/s, soit environ 19% de rendement (comparé aux 65 Mb/s de capacité). Avec agrégation il faut 575 μs pour transmettre 7 paquets de 500 octets, soit un débit utile de 48,7 Mb/s et donc un rendement de presque 75% [5]. Agréger permet donc de multiplier presque par 4 le rendement de la couche MAC dans le scénario présenté ici.

On notera que les mécanismes d'agrégation évoqués ici ne sont en rien modifiés dans la plus récente version de la norme (802.11ac). Ce dernier protocole représente une évolution majeure des capacités physiques des stations mais la couche MAC n'est modifiée que de façon marginale.

5. le détail des calculs effectués ici est donné au chapitre 2

1.3.2 Formats et particularités des agrégats dans 802.11n/ac

La solution permettant d'augmenter le rendement au niveau MAC depuis 802.11n repose sur deux systèmes d'agrégation. Le premier est situé en entrée de la couche MAC et effectue une agrégation des MSDU (Mac Service Data Unit) d'où son nom de A-MSDU (*aggregated* MSDU). Ce système a été créé pour permettre une meilleure transition entre les normes IEEE 802.3 (Ethernet) et IEEE 802.11n (Wi-Fi n). Des pseudo-en-têtes Ethernet sont visibles dans le format des A-MSDU (figure 1.8). Elles n'ont pas été supprimées afin de réduire les latences au sein des ponts entre les deux protocoles.

Le second système d'agrégation est situé en sortie de la couche MAC et agrège les MPDU (Mac Protocol Data Unit). Les agrégats sont appelés A-MPDU (*aggregated* MPDU) et leur format est illustré figure 1.9.

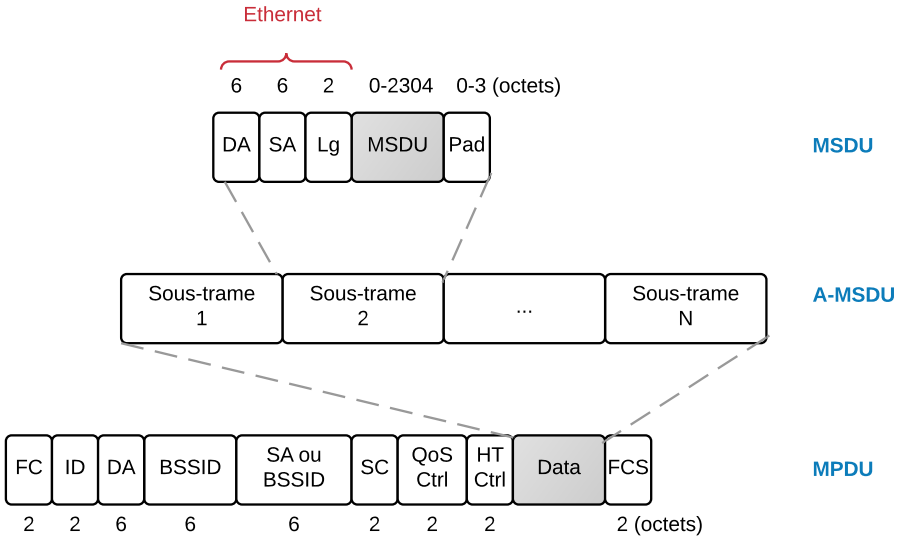


Figure 1.8 – Format A-MSDU

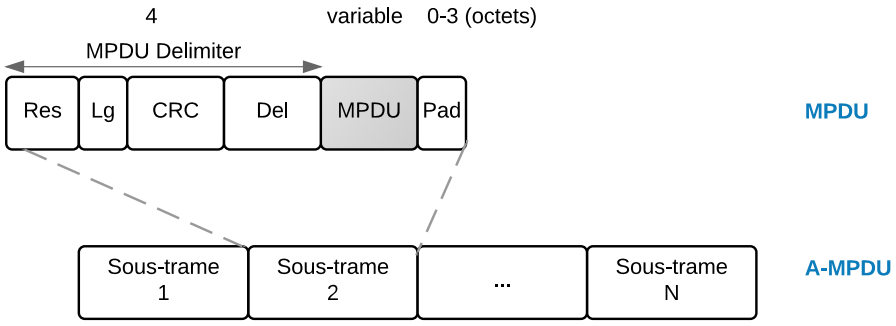


Figure 1.9 – Format A-MPDU

Le calcul du FCS (frame check sequence)[6] est effectué pour chaque paquets dans une A-MPDU. Ce dernier permet donc de retransmettre uniquement le paquet corrompu dans un agrégat contrairement à

6. le FCS permet de détecter une trame erronée. Il est le reste de la division modulo 2 de la trame par un polynôme de degré égal au nombre de bits alloués à ce champs (ici 16). Le FCS est transmis en fin de trame et est recalculé à réception afin de le comparer avec la valeur reçue

l'A-MSDU. Cette dernière est donc moins robuste aux erreurs de propagation puisqu'elle nécessite la retransmission de l'agrégat entier en cas de corruption d'un seul paquet. Cependant, l'agrégation de MSDU reste plus efficace pour la transmission de petits paquets (comme des paquets voix) puisque le volume des en-têtes MAC est fortement réduit comparé à l'A-MPDU. De plus, certains travaux portant sur la mise en œuvre d'une retransmission sélective des paquets contenus dans les A-MSDU (Maqhat *et al.*, 2012) permettraient de résoudre ce défaut de l'agrégation de MSDU.

Au vu de ces récents travaux et comme nous considérons uniquement des transmissions en espace libre et sans pertes par propagation, nous étudierons uniquement l'agrégation de MSDU. L'étude menée est utilisable également pour ce deuxième niveau d'agrégation qui ne diffère du premier que sur la taille maximale des agrégats et les en-têtes qui le composent.

Comme évoqué au début de cette section, la norme IEEE 802.11n propose de multiples mécanismes pour gérer l'accès au medium. Parmi ceux-ci nous avons considéré un système basé sur HCF (Hybrid Coordination Function) EDCA (Enhanced Distributed Channel Access). Ces deux fonctions constituent le contexte dans lequel le système d'agrégation est intégré. Elles forment une part importante de la couche MAC de 802.11n telle que nous la considérons (figure 1.6).

Comme illustré en figure 1.6, l'agrégation se situe immédiatement après chaque file associée à une classe de service. En pratique, cela signifie que seul des paquets d'une même classe peuvent être agrégés. Cette contrainte évite la remise en question des avantages offerts par la classification de service et assure que 802.11 à partir de sa version n assure la rétrocompatibilité avec 802.11e.

1.3.3 Comportement du système d'agrégation 802.11n/ac

Un système d'agrégation de paquets va transformer plusieurs flux applicatifs en un "super flux" agrégé. Comme défini dans (IEEE-802.11n, 2012), le système d'agrégation considéré accumule des paquets (MSDU) et les libère selon deux critères :

— le seuil de taille s est atteint

— la classe associée obtient l'accès au medium

Le seuil de taille est une limite inviolable sur la taille des agrégats constitués. Elle est configurée *a priori* dans le système et la norme préconise deux valeurs de seuil : 3839 et 7935 octets. Si la classe de service qui agrège gagne l'accès au medium avant que cette taille soit atteinte, l'agrégat en cours est libéré immédiatement.

Ce fonctionnement permet au système de ne pas retarder la transmission d'un paquet contrairement à d'autres mécanismes d'agrégation de paquets dans lesquels un temps d'attente seuil est défini (IEEE-802.15.3c, 2009). Ici nous avons donc un système passif qui se contente de concaténer des paquets en attendant une opportunité de transmission.

Le comportement de ce système d'agrégation est représenté en figure 1.10. Nous avons indiqué le seuil temporel h qui peut exister mais pas dans 802.11n/ac. Ces deux seuils sont des paramètres du système mais on remarque que la boucle depuis EDCA vers l'agrégation représente le contrôle opéré sur ce système par le mécanisme d'accès au medium. Les agrégats peuvent être poussés vers l'accès au medium si l'un des seuils est atteint, ou ils peuvent être tirés si l'accès au medium est gagné plus tôt.

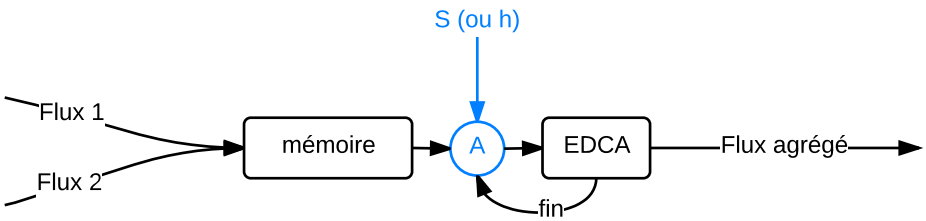


Figure 1.10 – Comportement du système d'agrégation 802.11n/ac

1.4 Conclusion et problématique

Ce premier chapitre a été l'occasion de définir le terme d'agrégation dans le domaine des réseaux et télécommunications et de voir quelques technologies qui utilisent ce concept. Nous avons défini le type d'agrégation que nous étudions ainsi que le contexte technologique dans lequel nous l'étudierons.

Nous avons répondu dans ce chapitre aux questions suivantes : Que signifie le terme agrégation dans le domaine des réseaux et télécommunications ? Sur quel type d'agrégation porte notre étude ? Quel est le contexte technologique choisi en tant que cas d'étude ?

Nous pouvons donc maintenant poser la problématique de cette thèse vis à vis du système défini ci-avant :

Comment optimiser la configuration du système d'agrégation de paquets présent depuis 802.11n à partir d'exigences de qualité de service exprimées d'une part par le client du réseau et d'autre part par le fournisseur d'accès ?

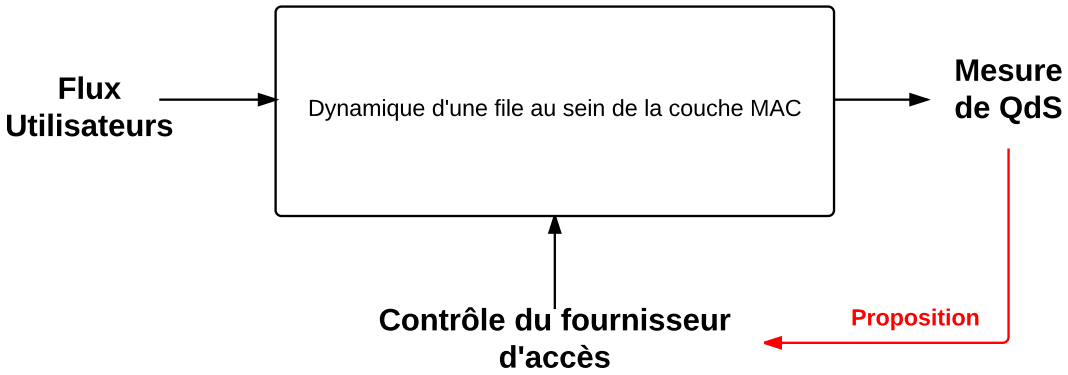


Figure 1.11 – Configuration du système en fonction d'exigences de QdS

La figure 1.11 représente notre objectif à terme. Il s'agit de fournir une solution de configuration du système d'agrégation au fournisseur d'accès. Cette solution utilisera une mesure sur plusieurs indicateurs de QdS et en fonction des exigences du client et du fournisseur d'accès, adaptera la configuration du système afin de les respecter.

Or ces exigences peuvent se révéler opposées. Le client souhaite obtenir le maximum de la bande passante alors que le fournisseur veut la partager entre ses clients. Le client veut toujours les meilleurs délais, le fournisseur souhaite aussi rentabiliser les ressources dans lesquelles il a investi. Le fournisseur peut être concerné par des problèmes d'équité entre ses clients, équité qui peut être en partie assurée par la classification de service. Afin de trouver un compromis nécessaire entre exigences client et exigences fournisseur, nous

cherchons à évaluer les performances d'un système d'agrégation de paquets, passerelle entre deux technologies (802.11 et 802.3). Ces technologies représentent un exemple classique de la dichotomie entre réseau d'accès (802.11) et réseau de distribution (802.3). Nous chercherons néanmoins à proposer des solutions généralisables à tout type de réseau présentant cette dichotomie (LTE, WiMAX, UWB, ...).

Afin de répondre à cette problématique nous avons d'abord besoin d'une connaissance intime du système. C'est l'objet du Chapitre 2 dans lequel le système sera étudié comme une "boite blanche". À partir de son comportement nous établirons un modèle qui nous permettra d'étudier les performances offertes à un client particulier (ce qui se distingue de la littérature qui n'étudie que les performances globales).

Après avoir détaillé le comportement théorique du système à l'issue d'une lecture très fine de la norme 802.11n, nous nous intéresserons à son fonctionnement au sein d'un simulateur (Riverbed Modeler (Riverbed, 2014)). Nous l'utiliserons afin de déterminer l'influence des paramètres d'agrégation sur quelques indicateurs de performance. Ce problème est pour nous une étape préliminaire à la réalisation de notre objectif final dans le Chapitre 4 dans lequel nous proposerons un algorithme de configuration automatique de l'agrégation basé sur les exigences du client et du fournisseur d'accès.

Chapitre 2

Évaluations des performances du système d'agrégation

2.1 État de l'art

De nombreux auteurs se sont intéressés aux performances du système d'agrégation de paquets implémenté dans 802.11n. Les objectifs sont aussi divers que les méthodes utilisées. Les travaux évoluent de la simple évaluation des performances de la norme jusqu'à la proposition de nouveaux algorithmes d'agrégation en passant par des algorithmes d'ordonnancement de paquets optimisant les bénéfices apportés par l'agrégation. Les auteurs utilisent des outils de simulation, des théories comme les chaînes de Markov ou encore un simple modèle analytique déduit du fonctionnement attendu de ce système.

Même si la plupart de ces études portent sur 802.11n, certains auteurs ont utilisé des versions antérieures pour étudier les performances d'un système d'agrégation pour 802.11 (a/b/g) (Kim *et al.*, 2004), (Choudhury *et al.*, 2006). D'autres études portent aujourd'hui sur l'agrégation dans la récente 802.11ac (Bellalta *et al.*, 2012) cependant cette dernière version n'apporte aucune modification au système d'agrégation. Nous étudions 802.11n parce que cette version reste plus répandue et mieux implémentée dans les simulateurs existants.

Parmi ceux qui étudient la norme 802.11n, différentes méthodes sont utilisées afin d'évaluer les performances du système d'agrégation. L'étude menée par (Lin et Wong, 2006) est basée sur la théorie des files d'attentes. Ils étendent d'abord le modèle de DCF composé de deux chaines de Markov et proposé par (Bianchi, 2000) afin d'étudier les deux mécanismes d'agrégation dans un canal soumis aux erreurs. Les en-têtes étant transmises à un débit inférieur, les auteurs considèrent qu'elles ne seront jamais erronées. Seul un accès au médium avec DCF et utilisant le mécanisme du RTS/CTS est étudié. Les probabilités de transmission sont calculées à la fois pour des transmissions uni-directionnelles et bi-directionnelles. Chacune des N stations considérées transmet de manière à saturer le medium.

Ce modèle analytique est ensuite confronté à des simulations menées à l'aide du simulateur NS-2. Dans ces simulations les paquets sont d'une taille fixe de 100 octets. Les résultats fournis confirment la bonne précision du modèle proposé. Ils montrent qu'un BER qui augmente cause un délai plus important et un débit utile plus faible. L'A-MSDU est plus touchée par cette réduction que l'A-MPDU puisque la corruption

d'un seul bit nécessite la retransmission de l'agrégat entier pour l'A-MSDU contrairement à l'A-MPDU. Il est alors facile de conclure que l'A-MSDU est plus performante que l'A-MPDU sur un canal sans erreurs et vice et versa. Enfin cet article se termine par la proposition d'un algorithme optimal d'adaptation de la taille des A-MSDU. À partir du BER et du modèle analytique développé, une station est capable de trouver le nombre optimal de paquets de 100 octets à agréger. Notons que cette taille annoncée comme optimale n'est pas sensible aux nombre de stations en concurrence pour l'accès au medium.

Cet article compte parmi les premiers publiés alors que la norme 802.11n n'était qu'à l'état de proposition du groupe de travail. Le débit utile ainsi que le délai sont les indicateurs de performances utilisés. Ils sont représentés en fonction du nombre de paquets dans un agrégat. À la même époque, (Kuppa et Dattatreya, 2006) proposent une étude en profondeur des performances de l'agrégation de MPDU avec un medium non saturé et en considérant une file d'attente de capacité finie. La file des transmission de la couche MAC est modélisée comme une file $M/G^{[K]}/1/N$. Les trames entrent dans cette file en suivant une loi de Poisson. Le serveur est oisif et attend que K trames soient dans le système avant de les transmettre. Les indicateurs utilisés sont le temps d'attente moyen des agrégats et l'utilisation moyenne de bande passante tracés pour plusieurs tailles de trames et en fonction du trafic entrant dans le système. À partir des résultats fournis par leur modèle, les auteurs concluent qu'il n'y a pas une taille unique pour les agrégats qui maximise l'utilisation de la bande passante sur tous les scénarios étudiés. Un algorithme d'adaptation dynamique de la taille des agrégats leur semble donc nécessaire.

Au delà des études basées sur la théorie des files d'attente, (Ginzburg et Kesselman, 2007) propose une approche intéressante. Plus question de probabilités, ici les auteurs créent un modèle déterministe dans lequel sont additionnés tous les temps élémentaires nécessaires pour transmettre un agrégat. Contrairement aux travaux de (Kuppa et Dattatreya, 2006) les termes de A-MPDU (Aggregated MAC Protocol Data Unit) et A-MSDU (Aggregated Mac Service Data Unit) [7] sont utilisés et ces deux mécanismes sont étudiés. Leur indicateur est le pourcentage d'utilisation du canal. Ils utilisent un canal idéal puis bruité et les flux sont basés sur TCP puis UDP. Il ressort de cette étude que l'agrégation de MPDU est la plus efficace quelque soit la configuration utilisée. Ces résultats sont logiques puisque nous avons déjà vu que cette agrégation est plus efficace sur un canal bruité grâce à son mécanisme de retransmission sélective. De plus, la taille des A-MPDU est largement supérieure à celle des A-MSDU dans cette étude (64 trames de 1500 octets contre 5 pour l'A-MSDU) ce qui réduit d'autant les en-têtes physiques si coûteuses puisque transmises à un débit de base très inférieur au débit de transmission des données.

D'autres auteurs se sont attachés à implémenter l'agrégation 802.11n dans des outils de simulation. C'est le cas de (Wang et Wei, 2009), qui implémente les fonctions d'agrégation dans un modèle NS-2 de 802.11e. Afin de limiter la complexité d'une telle implémentation, le système est limité à l'agrégation de MPDU car ils considèrent comme négligeable le gain apporté par l'agrégation de MSDU dans une agrégation à deux niveaux. Après avoir observé les améliorations apportées par l'agrégation en termes de délai et de débit, l'étude se concentre sur des trafics voix agrégés avec du trafic de priorité la plus basse (*background*). Cette agrégation est rendue possible en plaçant tous les flux dans la même catégorie au niveau MAC. Il apparaît alors que sans agrégation la qualité du signal audio des flux voix chute à partir d'un flux *background* de 1 Mb/s.

Les travaux de (Bhanage *et al.*, 2009) s'approchent un peu plus de la réalité puisque des expérimentations sont menées en utilisant un driver open-source populaire nommé MadWifi. Celui-ci est implémenté sur la

7. voir annexe A pour une définition de ces termes

plateforme de tests ORBIT ce qui limite l'étude à 802.11 a/b/g uniquement. Cependant, des mécanismes d'agrégation similaires à ceux de 802.11n (en particulier l'A-MSDU) sont implémentés dans le driver utilisé. Les auteurs expliquent que les applications telles que Skype ou VLC media player n'utilisent pas le champ *Type Of Service* (TOS). Ce choix conduit à l'éventualité que des paquets d'applications très différentes soient agrégés ensemble lors de leur passage dans la couche MAC d'un point d'accès 802.11n (cette observation rejoint les suppositions de (Wang et Wei, 2009)). Le résultat majeur de ces expérimentations concerne une augmentation de la gigue lorsque l'agrégation est effective.

Pour sa part, (Skordoulis *et al.*, 2008) utilise un modèle du système d'agrégation développé par Intel et implémenté dans le simulateur Riverbed Modeler (ex Opnet Modeler). À l'aide de ce modèle, ils étudient les performances en termes de débit utile des deux systèmes d'agrégation d'abord utilisés séparément puis combinés dans une agrégation à deux niveaux. Les résultats obtenus montrent qu'il est dans tous les cas préférable d'utiliser une agrégation à deux niveaux combinant les avantages de l'A-MSDU et de l'A-MPDU. Si une telle agrégation n'est pas possible, le plus performant reste l'A-MPDU (cette conclusion corrobore celle de (Kuppa et Dattatreya, 2006) et ce pour les mêmes raisons que précédemment).

D'après les auteurs, plusieurs problèmes doivent encore être étudiés :

— dans quelle mesure le temps de calcul nécessaire à la formation des agrégats augmente le délai global ?
— quel seuil de taille des agrégats doit être choisi ?
— quel algorithme d'ordonnancement des paquets permettrait d'améliorer les performances de ce système ?

Nous pouvons citer également les travaux de (Bhanage *et al.*, 2011) qui évaluent d'abord le temps de transmission des paquets à l'aide de leur connaissance de la couche MAC de 802.11. Ensuite, ils utilisent à nouveau la plateforme ORBIT afin de mener des expérimentations destinées à évaluer les performances de leur proposition d'algorithme. Les expérimentations révèlent des améliorations significatives offertes par l'agrégation en termes de délai moyen, de débit et de gigue.

D'autres auteurs ont proposé des algorithmes permettant d'optimiser les performances de l'agrégation et ils les ont évaluées à l'aide d'indicateurs comme le débit utile, la proportion d'en-têtes (Saif *et al.*, 2011), le délai des agrégats (Selvam et Srikanth, 2010), la gigue, ou encore le délai de bout en bout d'applications (Dely *et al.*, n.d.).

2.2 Positionnement

Ce chapitre est décomposé en trois parties organisées comme suit. D'abord nous faisons une abstraction du cas d'étude représenté par la norme 802.11n. Le système est formalisé pour en établir un modèle déterministe à l'aide de la théorie du calcul réseau. Ensuite, nous évaluons le pessimisme introduit par ce modèle en l'utilisant sur un exemple reprenant notre cas d'étude. Pour cela, nous utilisons notre connaissance approfondie de la norme 802.11n comme effectué par (Ginzburg et Kesselman, 2007) et (Bhanage *et al.*, 2011). L'idée est de calculer chaque temps de transmission des agrégats transmis, et en connaissant leur taille, de déduire les délais subis par les paquets. Enfin des simulations utilisant Riverbed Modeler sont menées pour comparaison, à la fois avec notre compréhension de la norme, mais aussi avec le modèle établi en Calcul Réseau.

2.2.1 Outil de simulation

Dans l'état de l'art précédent, nous avons vu que plusieurs outils permettent d'évaluer les performances de l'agrégation dans 802.11n. La liste de ces outils n'est pas exhaustive mais nous pouvons citer NS-2, NS-3, Riverbed Modeler (ex Opnet Modeler) ainsi que des plateformes diverses comme ORBIT ou KAUMesh.

Riverbed modeler embarque un moteur de simulation à événement discret. Ses modèles sont écrits à partir de machines à états et chaque état est programmé en langage C. Les modèles sont fournis soit par les constructeurs des équipements réseau soit par Riverbed comme modèles génériques qui s'engagent à respecter la standardisation. Des paramètres technologiques comme les tailles mémoires, les fréquences processeurs et les capacités des cartes réseaux peuvent également être inclus dans les modèles. Nous utiliserons cet outil de simulation et en particulier ses modèles génériques dans la mesure ou ce sont eux qui ont servi à la vérification lors de la standardisation de 802.11n (Skordoulis *et al.*, 2008).

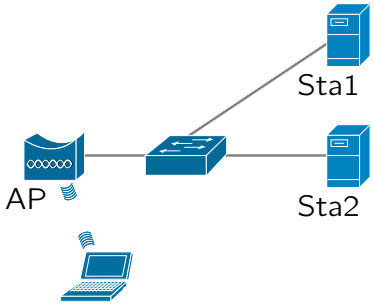
2.2.2 Théorie de modélisation

Nous pouvons faire plusieurs critiques vis-à-vis des études de performances citées précédemment. D'abord, les indicateurs de performance choisis restent macroscopiques. En effet, les auteurs cherchent uniquement à comparer les performances globales avec et sans système d'agrégation. Le délai d'une sous-trame, par exemple, n'est pas évalué. Une sous-trame correspond à un paquet pour l'utilisateur. Il est donc impossible de quantifier l'impact de l'agrégation sur la qualité de service pour une application cliente. Agréger peut devenir contraignant dès lors que l'on considère des applications critiques. Les paquets peuvent en effet être retardés s'ils entrent en concurrence avec un agrégat en lieu et place d'un paquet. C'est pourquoi nous devons être capables de quantifier l'influence de l'agrégation sur les délais. Il est notamment intéressant pour des applications critiques (un flux voix, une opération chirurgicale à distance, la commande d'un véhicule, avion, voiture, tramway, ...) de pouvoir obtenir des majorants *a priori*. En outre, de nombreuses études fournissant des résultats en moyenne ont été réalisées. C'est le cas des études utilisant la théorie des files d'attentes et des simulations.

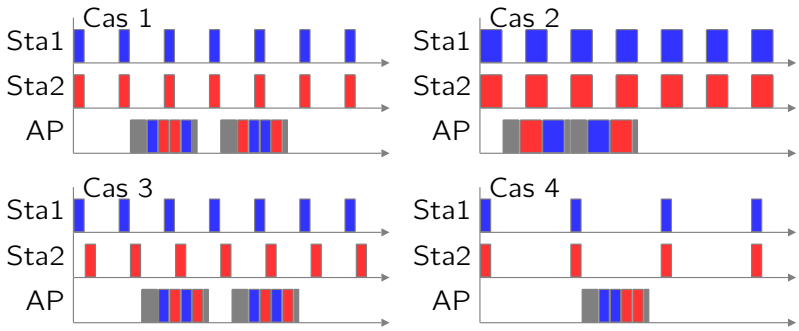
Quelques auteurs obtiennent des majorants comme dans (Ginzburg et Kesselman, 2007). Cependant, leurs résultats sont contraints par une connaissance approfondie des trafics entrants, ce qui reste complexe en pratique. Prenons, par exemple, un cas simple avec deux stations 802.11n Sta1 et Sta2 émettant vers un point d'accès AP qui va effectuer l'agrégation (illustré en figure 2.1a). Le nombre de scénarios possibles devient rapidement important dès que certaines hypothèses sont relâchées. Les flux entrants peuvent être modifiés en termes d'inter-arrivée, taille des paquets, gigue etc. De plus, les seuils du système d'agrégation peuvent changer. Un extrait de cas possibles est représenté sous la forme de chronogrammes en figure 2.1b.

Le premier cas présenté en figure 2.1b montre le résultat de l'agrégation des paquets des stations 1 et 2 lorsque les flux sont synchronisés et de taille égale. En cas numéro deux, une deuxième configuration est possible en modifiant uniquement la taille des paquets entrants et celle-ci conduit à un résultat agrégé complètement différent. Les cas 3 et 4 représentent respectivement une modification de la synchronisation entre les paquets et de leur inter-arrivée. À chacune de ces modification, qui peut prendre un nombre infini de variations, le calcul doit être mené à nouveau avec la méthode utilisée par (Ginzburg et Kesselman, 2007).

Pour pallier cette multitude de cas à considérer, nous nous intéressons dans un premier temps à l'identification du pire cas et proposons ainsi d'utiliser la théorie du calcul réseau dans l'établissement d'un modèle analytique (section 2.4).



(a) Topologie considérée



(b) Explosion combinatoire

Figure 2.1 – Topologie et explosion combinatoire de méthodes utilisées dans la littérature

2.3 Système d'agrégation considéré

Il existe deux systèmes d'agrégation au sein de la norme 802.11n. Le premier agrège les paquets à l'entrée de la couche MAC et est nommé agrégation de MSDU. Le second agrège les trames à leur sortie de la couche MAC, c'est l'agrégation de MPDU. La suite de cette section détaille notre choix et définit ce système.

Plusieurs études (McCreary et claffy, 2000),(Sinha *et al.*, 2007) ont montré que le trafic Internet était constitué pour moitié de paquets de moins de 200 octets. Ces paquets de petites taille ont forcé les réseaux sans-fil à s'adapter pour proposer un rendement supérieur d'où l'apparition de techniques d'agrégation qui permettent de réduire la part d'en-têtes. L'agrégation de MSDU dans 802.11n, technique la plus performante de ce point de vue, possède deux limites spécifiques :

— pas de retransmission sélective des paquets erronés

— des agrégats de taille limitée à 7935 octets

Ces spécificités, comparées à l'agrégation de MPDU, conduisent la plupart des auteurs à évaluer les performances soit des deux systèmes, soit de l'A-MPDU. Cependant nous pouvons voir celles-ci comme des avantages puisque la première permet l'économie d'en-têtes supplémentaires nécessaires à une retransmission sélective (un CRC pour le moins) ce qui la rend particulièrement bien adaptée à de petits paquets. La seconde permet d'éviter des retransmissions trop importantes et diminue le risque de pertes en environnement bruité

et donc le besoin de retransmissions.

Nous considérerons donc uniquement l'agrégation de MSDU puisqu'elle propose une plus grande réduction des en-têtes utilisées. Nous nous intéresserons aux variations de la taille des agrégats et la plage proposée par l'A-MSDU nous suffit. Quant à la retransmission sélective, elle prend son sens lorsque l'environnement est bruité ce qui ne sera pas notre cas. Notons toutefois que des travaux sont menés pour proposer une retransmission sélective des MSDU dans une A-MSDU (Saif et Othman, 2013).

La littérature révèle que moins d'études ont été menées sur cette agrégation et n'oublions pas que les travaux menés ici sont facilement transposable à l'agrégation de MPDU. Enfin ce choix n'est qu'un exemple puisque nous cherchons avant tout un cadre nous permettant d'étudier l'agrégation de paquets.

2.3.1 Un système, plusieurs définitions

Une analyse des études citées précédemment nous apprend que les auteurs ne considèrent pas tous le même système. Pour la plupart, le système est paramétré par deux seuils, un seuil de taille et un seuil temporel (définition 1).

Définition 1. *Le système d'agrégation libère un agrégat dès que :*

- *la quantité de données dans la mémoire tampon à l'entrée de la couche MAC est supérieure ou égale à un seuil de taille S.*
- *le temps passé dans le système par le premier paquet de cet agrégat est supérieur à un seuil temporel T.*

L'agrégat formé ne sera transmis que lorsque l'un de ces seuils est atteint. Cette hypothèse mène à un système potentiellement oisif et conduit à plusieurs remarques. Nous avons vu que si le champ TOS n'est pas utilisé, des paquets de priorités différentes seront agrégés ensemble. L'existence d'un seuil temporel implique de meilleures performances en termes de délai lorsque le seuil de taille est atteint avant l'expiration du seuil temporel. Les flux de plus basse priorité peuvent donc agir comme des amis en aidant les flux de plus haute priorité à atteindre ce seuil de taille au plus vite. Cette conséquence remet en cause l'intérêt de la classification de service et incite donc à ne pas l'utiliser. La version n de 802.11 n'a pas vocation à remettre en cause les développements effectués par les précédentes versions (802.11e en particulier), il est plus probable qu'elle constitue une extension de ses mécanismes comme pour toutes les versions précédentes.

Ces considérations nous ont conduits à une étude en détail de la norme ainsi qu'à une analyse des résultats fournit par Riverbed Modeler sur quelques simulations. Les premiers résultats de simulation collectés nous ont permis d'observer que l'agrégation ne s'applique que lorsque le medium est occupé (Figure **??**) ce qui invite à revenir sur la propriété d'oisiveté définie auparavant. En effet, nous avons remarqué qu'un agrégat en préparation était immédiatement libéré dès que le medium était libre sans pour autant attendre une durée temporelle T (paramètre d'ailleurs non présent dans le modèle générique utilisé par ce simulateur ni défini dans la norme). Nous proposons donc la définition ci-dessous qui reprend les éléments introduits au Chapitre 1.

Définition 2. *Le système d'agrégation libère un agrégat dès que :*

- *la quantité de données dans la mémoire tampon à l'entrée de la couche MAC est supérieure ou égale à un seuil de taille S.*
- *la classe de service de la station gagne l'accès au médium.*

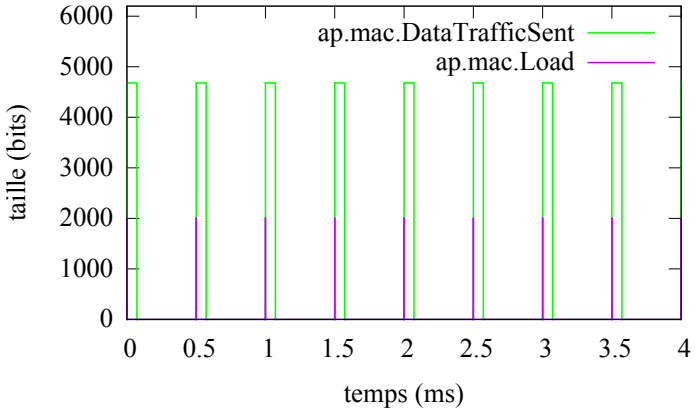


Figure 2.2 – Absence d'agrégation lorsque le trafic est faible

De plus, si le medium est libre, une trame ne sera pas retenue pour former un agrégat mais immédiatement émise. Un agrégat ne pourra donc être formé que si le medium est resté suffisamment occupé pour qu'au moins deux trames aient été reçues. On notera d'ailleurs que seuls des paquets partageant la même classe de service peuvent être agrégés. Ce comportement semble logique si l'on considère ce système comme un moyen d'améliorer le débit utile au niveau de la couche MAC. À première vue, conserver les paquets en mémoire alors que le medium est libre ne présente aucun avantage. Les paquets sont alors émis et la station agrège uniquement quand le medium est occupé.

2.4 Modèle analytique

Notre objectif dans cette étude est de modéliser un système d'agrégation de paquets quelconque. À l'instar de la définition du système d'agrégation de MSDU 802.11n posé en section précédente, il est nécessaire ici de formaliser le système que nous allons modéliser et de poser l'espace de validité de notre modèle. Auparavant, certaines notions sur la théorie du calcul réseau sont nécessaires à la bonne compréhension de cette étude.

2.4.1 Présentation du calcul réseau

Les principaux résultats concernant cette théorie peuvent être trouvés dans (Cruz, 1991), (Chang, 2000) et (Le Boudec et Thiran, 2001). Le calcul réseau permet l'obtention de majorants du délai et de l'arriéré de traitement d'un système en considérant uniquement le pire cas possible. Cette étude permettra d'aider à la décision de mettre en œuvre un système d'agrégation en tenant compte des contraintes applicatives.

En calcul réseau, la modélisation du réseau est effectuée par des fonctions non décroissantes qui représentent une quantité cumulée de données à un instant t.

L'ensemble de ces fonctions est donné par :

$$\mathcal{F} = \left\{ f : \mathbb{R}^+ \to \mathbb{R}^+, \forall t \geq s : f(t) \geq f(s), f(0) = 0 \right\}$$

Ces fonctions sont d'abord utilisées pour modéliser l'arrivée réelle $R(t)$ et la sortie réelle $R^*(t)$. $R(t)$ et

$R^*(t)$ représentent le nombre cumulé de bits qui sont respectivement entrés et sortis d'un système $\mathcal{S}$. Dans cet article, nous considérons que les fonctions sont continues dans le temps et l'espace. Ce n'est pas une restriction majeure étant donnée l'existence de transformations de modèles discrets vers des modèles continus (Chang, 2000). Les deux principaux résultats du calcul réseau concernent alors l'arriéré de traitement et le délai. Considérons un flux avec une fonction d'arrivée réelle R qui traverse un système $\mathcal{S}$ dont résulte une sortie réelle R^*. L'arriéré de traitement du flux $b(t)$ est défini comme :

$$b(t) = R(t) - R^*(t)$$

Le délai virtuel (correspondant au délai dans le cas d'un ordonnancement FIFO) est défini comme :

$$d(t) = \inf\{\tau \geq 0 \ : \ R(t) \leq R^*(t+\tau)\}$$

$R(t)$ étant inconnu, le calcul réseau introduit la notion suivante de courbe d'arrivée qui s'apparente à une contrainte sur le flux de données. Considérant un flux avec une fonction d'arrivée réelle R, une fonction $\alpha \in \mathcal{F}$ est une courbe d'arrivée pour R si :

$$\forall t, s \geq 0, t \geq s : R(t) - R(t-s) \geq \alpha(s)$$

De manière similaire, le service offert par un système à un flux R, R^*, est modélisé par une courbe de service minimale $\beta \in \mathcal{F}$ telle que :

$$R^* \geq \inf\{R(s) + \beta(t-s)\}$$

Une courbe de service β sera qualifiée de stricte pour le système $\mathcal{S}$ si au cours de n'importe quel intervalle $]s, t]$ durant lequel l'arriéré de traitement est non nul, au moins $\beta(t-s)$ données sont servies. Ces courbes d'arrivée et de service permettent alors l'obtention des majorants suivants :

Théorème. *(Chang, 2000; Le Boudec et Thiran, 2001). Considérons un système $\mathcal{S}$ qui offre une courbe de service β et qui stocke les données entrantes dans une file d'attente FIFO. Considérons un flux R traversant le système qui possède une courbe d'arrivée α. Nous obtenons alors les majorants suivants pour l'arriéré de traitement b et le délai d.*

$$b(t) \geq \sup\{t \geq 0 | \alpha(t) - \beta(t)\} \tag{2.1}$$

$$d(t) \leq \inf\{d \geq 0 | \forall t \geq 0, \alpha(t) \leq \beta(t+d)\} \tag{2.2}$$

Ces définitions sont données pour un seul système et un seul flux entrant. Dans le cas où plusieurs flux sont en compétition en entrée d'un système, la littérature introduit la notion de courbe de service résiduel.

Lemme. *(Service résiduel (Le Boudec et Thiran, 2001; Schmitt* et al.*, 2008)). Soit un système offrant une courbe de service stricte β avec deux flux entrant de courbes d'arrivée respectives α_1 et α_2 . Le service dédié au flux 1 est alors minoré par la courbe de service :*

$$\beta_1 = (\beta - \alpha_2)^+ \tag{2.3}$$

Dans cette section, notre objectif est de calculer la courbe de service offerte plus particulièrement à un

flux en entrée du système d'agrégation défini en section précédente. Nous restreindrons notre modélisation à des courbes de types affines uniquement, ce qui simplifie les calculs sans remettre en cause la précision des majorants calculés. Une étude du service offert par un tel système montrera les limites de la notion de service résiduel pour ce type de systèmes.

2.4.2 Formalisation du système modélisé

Nous avons vu que deux conditions permettent la libération des agrégats. Du point de vue Calcul réseau, nous devons considérer le pire cas qui correspond à atteindre la taille maximale d'agrégat admise par le système. Si nous considérons la représentation fonctionnelle introduite en figure 2.3, le système est séparé en quatre fonctions différentes, une sérialisation suivie d'une répartition des paquets dans la classe de service associée puis l'agrégation ou non des paquets au sein de chaque mémoire pour terminer sur la gestion de l'accès au medium.

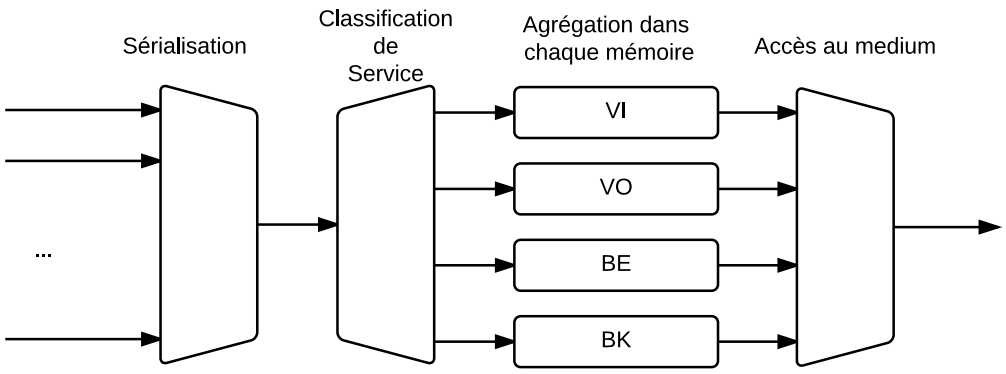


Figure 2.3 – Représentation fonctionnelle de la couche MAC 802.11n

Afin de nous abstraire du cas d'étude utilisé, nous mettons de côté toutes les fonctions excepté l'agrégation. Nous nous intéressons uniquement à la concurrence pour être inclus dans un agrégat qui est gérée par un ordonnancement *First In First Out* (FIFO). Cela signifie que l'ordonnancement entre les classes et la gestion du *backoff* et des TXOP ne font pas partie de l'étude. Différents modèles ont à ce propos été développé. On pourra alors se référer à la littérature concernant ce type de serveur. Le but est donc bien de modéliser le service rendu à un paquet et non à un agrégat. Le système considéré accumule donc des paquets en mémoire jusqu'à atteindre un seuil de taille défini au préalable.

2.4.3 Courbe du service offert à tous les flux

Commençons par exprimer le service offert à tous les flux. Le système d'agrégation est défini pour la suite pour un "super flux" entrant constitué de deux flux tel que $R(t) = R_1(t) + R_2(t)$.

Définition 3. *Considérons un système d'agrégation $\mathcal{S}$ avec un flux agrégé entrant $R(t)$. $\mathcal{S}$ émet un agrégat avec une capacité C lorsque son arriéré de traitement est supérieur ou égal à un seuil de taille s. La taille de l'agrégat est comprise dans l'intervalle $]s - l_{max}, s]$, avec l_{max} la taille maximale d'un paquet en entrée.*

La définition 3 n'est valide encore une fois que lorsque le système est occupé. La figure 2.4 montre l'évolution de l'arriéré de traitement en fonction de l'arrivée des trames. Nous pouvons observer que l'arriéré ne décroît que lorsque celui-ci a atteint une taille critique définie par le seuil s, soit $R(t) - R^*(t) \geq s$. En conséquence, le système présente un comportement oisif et le service est fonction de l'arrivée des données (à contrario de systèmes à service de type taux constant $\beta(t) = Ct$).

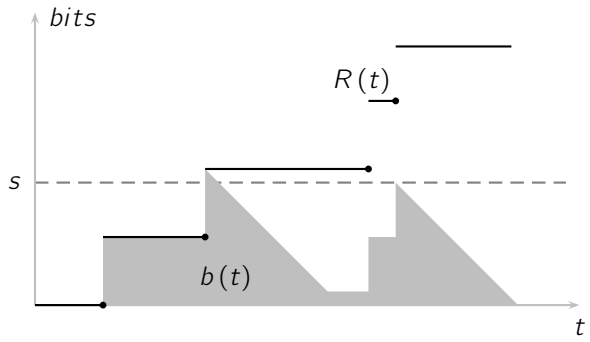


Figure 2.4 – Évolution théorique de l'arriéré de traitement dans un système d'agrégation

En d'autres termes, l'émission d'un agrégat est conditionnée par la quantité de trafic entrant. Comme le service est fonction du trafic entrant dans le système, nous proposons de compléter la contrainte sur l'arrivée par une borne minimale comme dans le cas du *Real-Time Calculus* (Thiele *et al.*, 2000). Nous admettrons que chaque flux i est enveloppé par deux courbes d'arrivée, une majorante et une minorante. Cette hypothèse est représentée figure 2.5. Nous avons ainsi $\forall 0 \leq t_0 \leq t$:

$$\begin{aligned} \underline{\alpha_i}(t - t_0) \leq R_i(t) - R_i(t_0) \leq \alpha_i(t - t_0) \\ \underline{\rho_i}(t - t_0 - \tau_i)^+ \leq R_i(t) - R_i(t_0) \leq \sigma_i + \rho_i(t - t_0) \end{aligned} \tag{2.4}$$

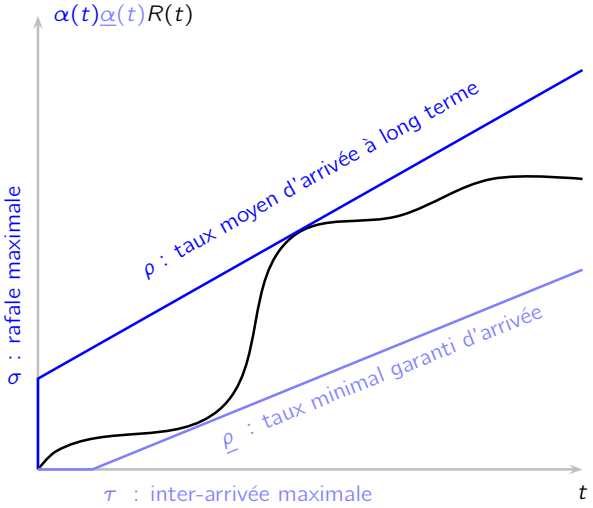


Figure 2.5 – Enveloppe considérée sur l'arrivée des paquets pour notre modélisation théorique

Nous proposons alors de modéliser le service offert par le système d'agrégation comme suit.

Proposition 1. *Soit un système d'agrégation $\mathcal{S}$ avec deux flux en entrée contraints par les courbes d'arrivée $\underline{\alpha_i}(t)$ et $\alpha_i(t)$ avec $i = 1, 2$. Le service global offert par le système est contraint par la courbe $\beta(t)$ tel que :*

$$\beta(t) = R(t - \Delta)^+$$

avec, $R = \underline{\rho} = \underline{\rho_1} + \underline{\rho_2}$

$$\Delta = \begin{cases} \tau_i + s/\underline{\rho_i} & \text{si } s/\underline{\rho_i} \leq \tau_j - \tau_i \\ \tau_i + \left(s + \underline{\rho_j}(\tau_j - \tau_i)\right) / \left(\underline{\rho_i} + \underline{\rho_j}\right) & \text{sinon} \end{cases}$$

pour $i = \arg\min_{i=1,2} \tau_i$ *et* $j = \arg\max_{j=1,2} \tau_j$

L'obtention de cette courbe s'appuie sur l'identification d'une part de l'inter-arrivée maximale entre deux agrégats en sortie du système et d'autre part du taux de service minimal.

Notons Δ la latence maximale avant qu'un agrégat soit émis. En supposant que durant l'intervalle $[t, t + \Delta]$ aucun service ne soit rendu, nous avons :

$$\Delta \leq \sup_{t \geq 0} \{\inf \{\delta \geq 0 \mid (R_1(t + \delta) + R_2(t + \delta)) - (R_1^*(t + \delta) + R_2(t + \delta)) \geq s\}\}$$

Durant cette durée, puisque la quantité en attente n'est pas suffisante pour qu'un agrégat soit généré, nous avons $R_i^*(t + \delta) = R_i(t)$. La définition des courbes d'arrivée 2.4 nous donne alors $\forall \delta \geq 0$:

$$\Delta \leq \sup_{t \geq 0} \left\{\inf \left\{\delta \geq 0 \mid \left(\underline{\alpha_1}(t + \delta) + \underline{\alpha_2}(t + \delta)\right) - \left(\underline{\alpha_1}(t) + \underline{\alpha_2}(t)\right) \geq s\right\}\right\}$$

Pour que le seuil soit atteint, il est donc nécessaire qu'au moins l'un des flux apporte du trafic en entrée. Comme l'arrivée des flux est minorée par une fonction de type *rate-latency* (comme $\underline{\alpha}(t) = \underline{\rho}(t - \tau)^+$), différents cas de figure doivent être pris en compte suivant t et suivant les latences maximales avant l'arrivée d'un paquet τ_1 et τ_2. On note alors $i = \arg\min_{i=1,2} \tau_i$ et $j = \arg\max_{j=1,2} \tau_j$. Nous pouvons maintenant exprimer la latence maximale avant le départ d'un agrégat notée Δ. Commençons par exprimer Δ pour chacun des cas représentés en figure 2.6.

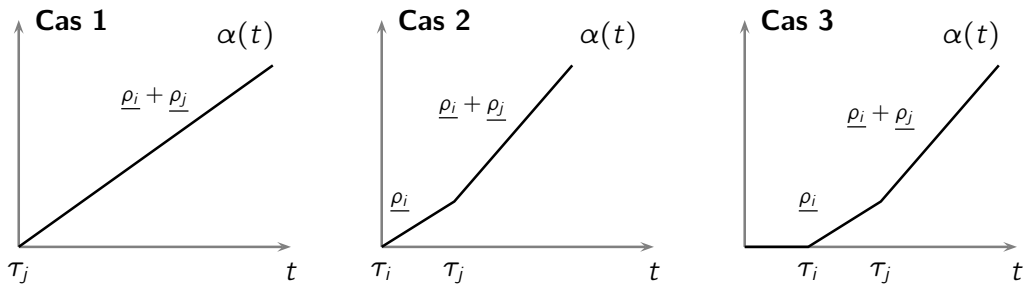


Figure 2.6 – Cas considérés permettant d'établir un modèle du service offert à un flux particulier

Cas 1 : les deux flux apportent du trafic à l'instant t ($t \geq \tau_j$)

$$\begin{aligned}\Delta &\leq \sup_{t \geq \tau_j} \left\{ \inf \left\{ \delta \geq 0 \mid \left(\underline{\alpha_i}(t+\delta) + \underline{\alpha_j}(t+\delta) \right) - \left(\underline{\alpha_i}(t) + \underline{\alpha_j}(t) \right) \geq s \right\} \right\} \\ &\leq \inf \left\{ \delta \geq 0 \mid \delta \left(\underline{\rho_i} + \underline{\rho_j} \right) \geq s \right\} \\ &\leq \frac{s}{\underline{\rho_i} + \underline{\rho_j}}\end{aligned}$$

Cas 2 : un seul flux apporte du trafic à l'instant t ($\tau_i \leq t < \tau_j$)

$$\begin{aligned}\Delta &\leq \sup_{\tau_i \leq t < \tau_j} \left\{ \inf \left\{ \delta \geq 0 \mid \left(\underline{\alpha_i}(t+\delta) + \underline{\alpha_j}(t+\delta) \right) - \left(\underline{\alpha_i}(t) + \underline{\alpha_j}(t) \right) \geq s \right\} \right\} \\ &\leq \sup_{\tau_i \leq t < \tau_j} \left\{ \inf \left\{ \delta \geq 0 \mid \underline{\rho_i}(t+\delta-\tau_i) + \underline{\rho_j}(t+\delta-\tau_j)^+ - \underline{\rho_i}(t-\tau_i) \geq s \right\} \right\} \\ &\leq \sup_{\tau_i \leq t < \tau_j} \left\{ \inf \left\{ \delta \geq 0 \mid \underline{\rho_i}\delta + \underline{\rho_j}(t+\delta-\tau_j)^+ \geq s \right\} \right\}\end{aligned}$$

Nous devons ici prendre en compte la durée d'accumulation des données $s/\underline{\rho_i}$ pour savoir si le deuxième flux participe à l'atteinte du seuil. Deux sous-cas apparaissent selon que les données du premier flux suffisent à faire dépasser le seuil ou non.

Nous pouvons dans ce cas décomposer l'inégalité en :

$$\begin{aligned}\Delta \leq &\sup_{\tau_i \leq t < \tau_j - s/\underline{\rho_i}} \left\{ \inf \left\{ \delta \geq 0 \mid \underline{\rho_i}\delta + \underline{\rho_j}(t+\delta-\tau_j)^+ \geq s \right\} \right\} \\ &\vee \sup_{\tau_j - s/\underline{\rho_i} \leq t < \tau_j} \left\{ \inf \left\{ \delta \geq 0 \mid \underline{\rho_i}\delta + \underline{\rho_j}(t+\delta-\tau_j)^+ \geq s \right\} \right\}\end{aligned} \tag{2.5}$$

Cas 2.1 : le premier flux fait dépasser le seuil ($\tau_i \leq t < \tau_j$ et $s/\underline{\rho_i} \leq \tau_j - \tau_i$)

Pour le premier supremum de l'équation 2.5, nous avons $\tau_j - t > s/\underline{\rho_i}$, alors $\underline{\alpha_i}(\tau_j) - \underline{\alpha_i}(t) > s$.

Comme $\underline{\alpha_j}(\tau_j) = 0$, δ existe $\forall t < \tau_j - s/\underline{\rho_j}$ avec $\underline{\rho_j}(t+\delta-\tau_j)^+ = 0$. À l'inverse si $\tau_j - t \leq s/\underline{\rho_i}$, alors $\underline{\alpha_i}(\tau_j) - \underline{\alpha_i}(t) \leq s$ et l'existence de δ implique que $\underline{\rho_j}(t+\delta-\tau_j)^+ = \underline{\rho_j}(t+\delta-\tau_j)$.

$$\begin{aligned}\Delta \leq &\sup_{\tau_i \leq t < \tau_j - s/\underline{\rho_i}} \left\{ \inf \left\{ \delta \geq 0 \mid \underline{\rho_i}\delta \geq s \right\} \right\} \\ &\vee \sup_{\tau_j - s/\underline{\rho_i} \leq t < \tau_j} \left\{ \inf \left\{ \delta \geq 0 \mid \underline{\rho_i}\delta + \underline{\rho_j}(t+\delta-\tau_j) \geq s \right\} \right\} \\ &\leq \frac{s}{\underline{\rho_i}} \vee \sup_{\tau_j - s/\underline{\rho_i} \leq t < \tau_j} \left\{ \inf \left\{ \delta \geq 0 \mid \left(\underline{\rho_i} + \underline{\rho_j} \right) \delta \geq s - \underline{\rho_j}(t-\tau_j) \right\} \right\} \\ &\leq \frac{s}{\underline{\rho_i}} \vee \sup_{\tau_j - s/\underline{\rho_i} \leq t < \tau_j} \left\{ \frac{s - \underline{\rho_j}(t-\tau_j)}{\underline{\rho_i} + \underline{\rho_j}} \right\} \\ &\leq \frac{s}{\underline{\rho_i}} \vee \frac{s + \underline{\rho_j} s/\underline{\rho_i}}{\underline{\rho_i} + \underline{\rho_j}} \leq \frac{s}{\underline{\rho_i}}\end{aligned}$$

Cas 2.2 : les deux flux font dépasser le seuil ($\tau_i \leq t < \tau_j$ et $s/\underline{\rho_i} > \tau_j - \tau_i$)

Ici l'existence de δ implique que $\underline{\rho_j}\,(t+\delta-\tau_j)^+ = \underline{\rho_j}\,(t+\delta-\tau_j)$. D'où :

$$\begin{aligned}\Delta &\leq \sup_{\tau_i \leq t < \tau_j} \left\{\inf\left\{\delta \geq 0 \mid \underline{\rho_i}\delta + \underline{\rho_j}\,(t+\delta-\tau_j) \geq s\right\}\right\} \\ &\leq \sup_{\tau_i \leq t < \tau_j} \left\{\inf\left\{\delta \geq 0 \mid \left(\underline{\rho_i}+\underline{\rho_j}\right)\delta \geq s - \underline{\rho_j}\,(t-\tau_j)\right\}\right\} \\ &\leq \sup_{\tau_i \leq t < \tau_j} \left\{\frac{s-\underline{\rho_j}\,(t-\tau_j)}{\underline{\rho_i}+\underline{\rho_j}}\right\} \leq \frac{s+\underline{\rho_j}\,(\tau_j-\tau_i)}{\underline{\rho_i}+\underline{\rho_j}} \leq \frac{s}{\underline{\rho_i}}\end{aligned}$$

Cas 3 : aucun flux n'apporte du trafic à l'instant t ($0 \leq t < \tau_i$)

$$\begin{aligned}\Delta &\leq \sup_{0 \leq t < \tau_i} \left\{\inf\left\{\delta \geq 0 \mid \underline{\rho_i}\,(t+\delta-\tau_i)^+ + \underline{\rho_j}\,(t+\delta-\tau_j)^+ - \underline{\rho_i}\,(t-\tau_i) - \underline{\rho_j}\,(t-\tau_j) \geq s\right\}\right\} \\ &\leq \sup_{0 \leq t < \tau_i} \left\{\inf\left\{\delta \geq 0 \mid \underline{\rho_i}\,(t+\delta-\tau_i)^+ + \underline{\rho_j}\,(t+\delta-\tau_j)^+ \geq s\right\}\right\}\end{aligned}$$

De la même manière que dans le cas précédent, nous devons considérer deux sous-cas.

Cas 3.1 : le premier flux fait dépasser le seuil ($0 \leq t < \tau_i$ et $s/\underline{\rho_i} \leq \tau_j - \tau_i$)

Dans ce cas, $\underline{\alpha_i}\left(\tau_i + s/\underline{\rho_i}\right) = s$ tandis que $\underline{\alpha_j}\left(\tau_i + s/\underline{\rho_i}\right) = 0$. Aussi le seuil s sera atteint pour $\underline{\rho_j}\,(t+\delta-\tau_j)^+ = 0$ et :

$$\begin{aligned}\Delta &\leq \sup_{0 \leq t < \tau_i} \left\{\inf\left\{\delta \geq 0 \mid \underline{\rho_i}\,(t+\delta-\tau_i) \geq s\right\}\right\} \\ &\leq \sup_{0 \leq t < \tau_i} \left\{\inf\left\{\delta \geq 0 \mid \delta \geq \frac{s}{\underline{\rho_i}} + (\tau_i - t)\right\}\right\} \leq \frac{s}{\underline{\rho_i}} + \tau_i\end{aligned}$$

Cas 3.2 : les deux flux font dépasser le seuil ($0 \leq t < \tau_i$ et $s/\underline{\rho_i} > \tau_j - \tau_i$)

Cela signifie que $\underline{\alpha_i}\,(\tau_j) < s$ et donc que l'existence de δ implique que :

$$\underline{\rho_j}\,(t+\delta-\tau_j)^+ = \underline{\rho_j}\,(t+\delta-\tau_j)$$

$$\begin{aligned}\Delta &\leq \sup_{0 \leq t < \tau_i} \left\{\inf\left\{\delta \geq 0 \mid \underline{\rho_i}\,(t+\delta-\tau_i) + \underline{\rho_j}\,(t+\delta-\tau_j) \geq s\right\}\right\} \\ &\leq \sup_{0 \leq t < \tau_i} \left\{\inf\left\{\delta \geq 0 \mid \left(\underline{\rho_i}+\underline{\rho_j}\right)\delta \geq s - t\left(\underline{\rho_i}+\underline{\rho_j}\right) + \underline{\rho_i}\tau_i + \underline{\rho_j}\tau_j\right\}\right\} \\ &\leq \sup_{0 \leq t < \tau_i} \left\{\inf\left\{\delta \geq 0 \mid \delta \geq \frac{s}{\underline{\rho_i}+\underline{\rho_j}} - t + \frac{\underline{\rho_i}\tau_i + \underline{\rho_j}\tau_j}{\underline{\rho_i}+\underline{\rho_j}}\right\}\right\} \\ &\leq \frac{s}{\underline{\rho_i}+\underline{\rho_j}} + \frac{\underline{\rho_i}\tau_i + \underline{\rho_j}\tau_j}{\underline{\rho_i}+\underline{\rho_j}} = \tau_i + \frac{s+\underline{\rho_j}\,(\tau_j-\tau_i)}{\underline{\rho_i}+\underline{\rho_j}}\end{aligned}$$

Finalement, on obtient donc,

si le premier flux fait dépasser le seuil ($s/\underline{\rho_i} \leq \tau_j - \tau_i$) :

$$\begin{aligned}
&\Delta \leq \tau_i + s/\underline{\rho_i} && \text{si } 0 \leq t < \tau_i \\
&\Delta \leq s/\underline{\rho_i} && \text{si } \tau_i \leq t < \tau_j - s/\underline{\rho_i} \\
&\Delta \leq \left(s + \underline{\rho_j} s/\underline{\rho_i}\right) / \left(\underline{\rho_i} + \underline{\rho_j}\right) && \text{si } \tau_j - s/\underline{\rho_i} \leq t < \tau_j \\
&\Delta \leq s/\left(\underline{\rho_i} + \underline{\rho_j}\right) && \text{sinon } (t \geq \tau_j)
\end{aligned}$$

sinon si les deux flux font dépasser le seuil ($s/\underline{\rho_i} > \tau_j - \tau_i$) :

$$\begin{aligned}
&\Delta \leq \tau_i + \left(s + \underline{\rho_j}\,(\tau_j - \tau_i)\right) / \left(\underline{\rho_i} + \underline{\rho_j}\right) && \text{si } 0 \leq t < \tau_i \\
&\Delta \leq \left(s + \underline{\rho_j}\,(\tau_j - \tau_i)\right) / \left(\underline{\rho_i} + \underline{\rho_j}\right) && \text{si } \tau_i \leq t < \tau_j \\
&\Delta \leq s/\left(\underline{\rho_i} + \underline{\rho_j}\right) && \text{sinon } (t \geq \tau_j)
\end{aligned}$$

La valeur maximale de Δ ($\sup \Delta$) est ainsi obtenue pour $0 \leq t < \tau_i$. On ajoute dans ce cas la latence τ_i modélisée par les courbes d'arrivées minimales.

Le second point pour l'obtention de la proposition 1 concerne le taux de service minimum. Les agrégats sont transmis par le système à un taux constant noté C. Dans le pire cas, on aura donc un agrégat tous les Δ unités de temps comme illustré à la figure 2.7.

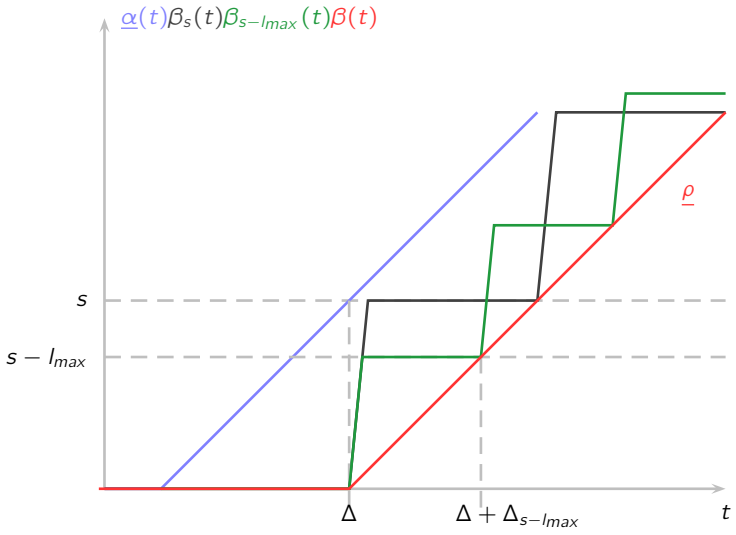


Figure 2.7 – Courbes de services pour différentes tailles d'agrégat

En cas de non fragmentation des paquets à l'entrée du système (comme dans le cas de 802.11n), la taille d'un agrégat peut varier dans l'intervalle $]s - l_{max}, s]$. En effet, dans le meilleur cas la somme des paquets contenus dans le système est égale au seuil s. Dans le pire cas, le dernier paquet est de taille maximale l_{max}, et dépasse d'un bit le seuil s. Il est donc exclu de l'agrégat. La figure 2.7 montre l'évolution du service lorsque la taille de l'agrégat vaut $s - l_{max}$ et lorsqu'elle vaut s. Il s'agit là de deux évolutions possibles, aussi nous

proposons de nous référer à la courbe de type *rate-latency* $\beta(t)$ qui minore toutes les évolutions possibles. On obtient ainsi $\beta(t) = R(t-\Delta)^+$ avec :

$$R = \frac{s - l_{max}}{\frac{s-l_{max}}{\underline{\rho}}} = \underline{\rho} = \underline{\rho_1} + \underline{\rho_2}$$

Cette proposition suppose que le débit de la ligne C est très grand devant la somme des taux d'arrivée des paquets $\sum_i \underline{\rho_i}$. De plus la stabilité du système est assurée selon son domaine de validité, c'est à dire lorsqu'il est congestionné mais non saturé.

2.4.4 Courbe du service offert à un seul flux

La proposition 1 correspond au service offert au super flux $R_1(t) + R_2(t)$. Le but de cette section est d'identifier la courbe de service réservé au flux R_1, que l'on notera $\beta_1(t)$.

Comme notre courbe de service est stricte, nous pouvons appliquer le service résiduel défini par l'équation (2.3). Avec les courbes d'arrivée et de service considérées, le service résiduel s'exprime comme,

$$\begin{aligned}\beta_1(t) &= (\beta(t) - \alpha_2(t))^+ \\ &= \left((\underline{\rho_1} + \underline{\rho_2})(t-\Delta)^+ - (\sigma_2 + \rho_2)\right)^+ \end{aligned} \tag{2.6}$$

La figure 2.8 montre que le service résiduel revient à considérer le flux 2 comme prioritaire (pire cas).

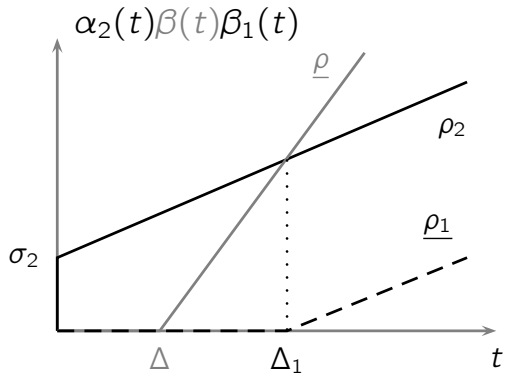


Figure 2.8 – Application directe du service résiduel

Lorsque $t > \Delta_1$, le taux de service dédié au flux 1 sera donc égal à $\underline{\rho_1} + \underline{\rho_2} - \rho_2$. De plus nous avons :

$$\begin{aligned}\sigma_2 + \rho_2\Delta_1 &= (\underline{\rho_1} + \underline{\rho_2})(\Delta_1 - \Delta)^+ \\ &= (\underline{\rho_1} + \underline{\rho_2})(\Delta_1 - \Delta) \\ \Delta_1 &= \frac{\sigma_2 + \Delta(\underline{\rho_1} + \underline{\rho_2})}{\underline{\rho_1} + \underline{\rho_2} - \rho_2}\end{aligned} \tag{2.7}$$

Nous obtenons donc la proposition suivante :

Proposition 2. *Le système d'agrégation défini à la proposition 1 offre au flux $R_1(t)$ une courbe de service minimale :*

$$\beta_1(t) = (\underline{\rho_1} + \underline{\rho_2} - \rho_2)(t - \Delta_1)^+$$

Pour cette proposition 2, on note que le taux de service offert au flux 1 peut être inférieur au taux d'arrivée des paquet de ce même flux. Cette situation conduit à l'impossibilité d'obtenir des majorants autres que l'infini pour le délai et l'arriéré de traitement. En effet, cela peut entrainer une divergence entre la courbe de service de pente $\underline{\rho_1} + \underline{\rho_2} - \rho_2 \leq \rho_1$ et la courbe d'arrivée (maximale) de pente ρ_1. Nous avons donc besoin d'avoir le même taux d'arrivée que ce soit pour les courbes d'arrivées majorante ou minorante. Dans la suite nous supposerons donc que $\underline{\rho_i} = \rho_i$. Dans ce cas, le taux de service pour le flux 1 devient $\underline{\rho_1} + \underline{\rho_2} - \underline{\rho_2} = \underline{\rho_1}$. Cela traduit dans ce type de système que le pire taux de service revient à la situation où un flux (utilisateur) est seul.

On note également que si Δ représente le pire délai avant qu'un agrégat soit prêt à être transmis, cela n'implique pas explicitement que cet agrégat contienne des données du flux 1. Ensuite, l'expression de Δ_1 prend en considération la rafale du flux 2, or si une telle rafale avait été présente, un agrégat serait parti plus tôt, et donc le service rendu aurait été meilleur. Nous proposons donc d'améliorer cette expression. L'étude est alors la même que pour la proposition 1 excepté pour la définition de Δ. Nous chercherons un nouveau Δ_1, l'inter-arrivée maximale entre deux agrégats contenant des données du flux R_1, de sorte que $\Delta_1 \geq \Delta$.

On peut déjà noter que suivant le principe FIFO, si les données du flux 1 arrivent avant celles du flux 2 ($\tau_1 < \tau_2$), alors $\Delta_1 = \Delta$ car il n'y a pas de concurrence. Dans le cas inverse, cela signifie que l'on peut avoir $\Delta \leq \tau_1$, et qu'il est donc nécessaire d'attendre le prochain agrégat qui sera émis après τ_1. Dans ce cas, l'inter-arrivée sera majorée par :

$$\delta_1 = \tau_1 + \frac{s - \underline{\rho_2}(\tau_1 - \tau_2) + \left\lfloor \frac{\tau_1 - \tau_2}{s/\underline{\rho_2}} \right\rfloor s}{\underline{\rho_2} + \underline{\rho_1}}$$

Nous savons que $\delta_1 > \tau_1$. Il nous faut soustraire au seuil de taille s la quantité de données apportée par le flux 2. Elle s'exprime comme $\underline{\alpha_2(\tau_1)} = \underline{\rho_2}(\tau_1 - \tau_2)$, la quantité de données du flux 2 à l'instant τ_1 dont on a soustrait $\left\lfloor \frac{\tau_1 - \tau_2}{s/\underline{\rho_2}} \right\rfloor s$, les éventuels agrégats émis à l'aide du flux 2 seul. Le calcul de δ_1 est illustré par la figure 2.9. Il correspond au moment ou la courbe de service $\beta(t)$ croise la courbe d'arrivé du flux concurrent $\alpha_2(t)$

Nous obtenons alors une nouvelle courbe de service.

Proposition 3. *Le système d'agrégation défini à la proposition 1 offre au flux $R_1(t)$ une courbe de service minimale :*

$$\beta_1(t) = \underline{\rho_1}(t - \Delta_1)^+$$

avec $\Delta_1 = \max(\Delta, \delta_1)$.

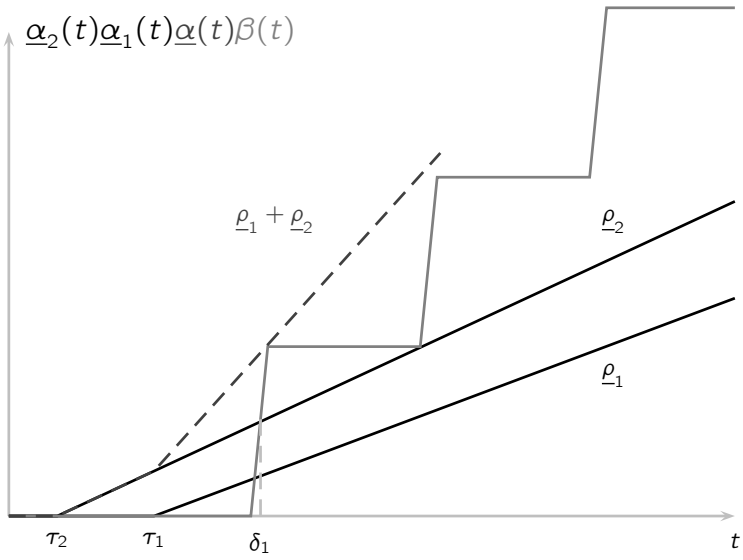
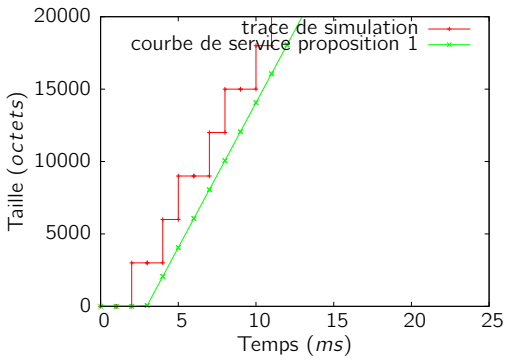


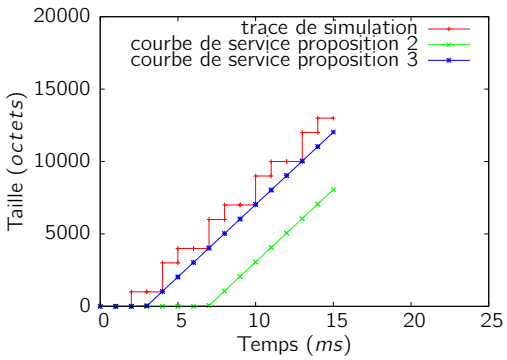
Figure 2.9 – Illustration du calcul de δ_1, le temps maximal entre deux agrégats selon notre modèle

2.4.5 Application numérique

Nous considérons toujours la topologie représentée en figure 2.1a. Le scénario comprend deux flux périodiques de période 1 ms, la taille d'une MSDU sera de 1 000 octets et le seuil de taille sera de 3 839 octets (qui est l'un des deux seuils définis pour 802.11n). Nous avons développé un programme qui génère les traces de la sortie des paquets en se basant sur la définition 2. À l'aide de ce générateur de traces, nous pouvons suivre le service réel pour le comparer à notre modèle. Les courbes d'arrivée seront calculées selon la taille des paquets, l'inter-arrivée minimale entre deux paquets et la gigue comme mentionnée dans (Chakraborty *et al.*, 2000).



(a) Service global, $\beta(t)$



(b) Service dédié, $\beta_1(t)$ et $\beta'_1(t)$

Figure 2.10 – Comparaison des courbes de service avec des simulations

La figure 2.10a représente la courbe de service obtenue avec la proposition 1 comparée à la sortie réelle. La figure 2.10b représente pour la même trace le service dédié à un flux selon les propositions 2 et 3. Le tracé du service dédié à un flux montre qu'avec ce scénario, la proposition 3 est la plus proche de la simulation, et donc que l'application du service résiduel conduit ici à un pessimisme de modélisation. Ce pessimisme est dû à l'originalité du système considéré qui à l'instar des systèmes classiques va offrir une meilleure qualité de service lorsqu'il y aura plus de trafic. Les performances du système du point de vue d'un flux (utilisateur) seront meilleures même si le trafic supplémentaire est offert par d'autres flux. Or le service résiduel est obtenu en calcul réseau en considérant que les autres flux sont exclusivement des concurrents. Cette hypothèse de départ conduit à une estimation pessimiste du service rendu à un flux. C'est pourquoi nous proposons une extension du service résiduel permettant d'éliminer des cas impossibles pour de tels systèmes.

2.4.5.a Calcul des majorants

Nous allons maintenant calculer des majorants de l'arriéré de traitement et du délai à partir des équations (2.1) et (2.2). La figure 2.11 montre que les majorants de l'arriéré de traitement et du délai sont respectivement les distances verticale et horizontale maximales entre les courbes d'arrivée et de service.

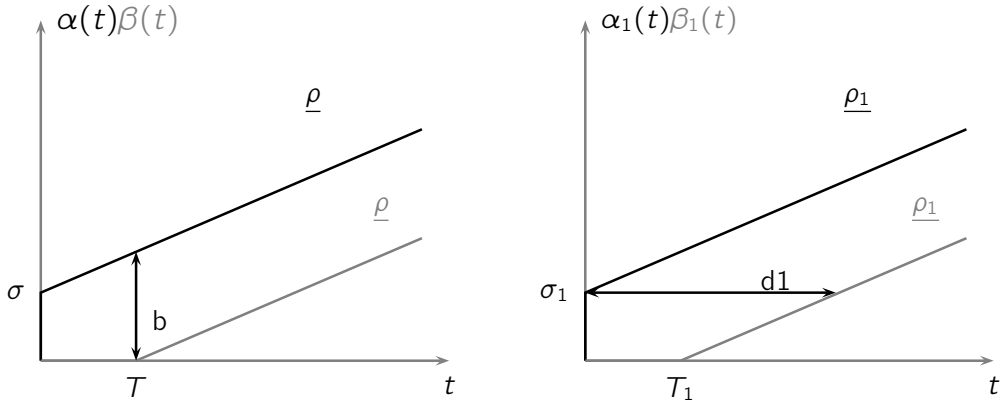


Figure 2.11 – Calcul des majorants du délai et de l'arriéré

La distance verticale est maximale à $t = T$ comme représenté en figure 2.11. Ici comme $T = \Delta$, le majorant de l'arriéré de traitement est donc,

$$b = \alpha(\Delta) - \beta(\Delta) \tag{2.8}$$

$$b = \sigma_1 + \sigma_2 + \Delta(\underline{\rho_1} + \underline{\rho_2}) \tag{2.9}$$

Dans le cas de 802.11n, ce majorant de l'arriéré de traitement permet de dimensionner la taille mémoire allouée aux données à agréger.

Nous cherchons maintenant à calculer un majorant du délai subi par un paquet dans le système d'agrégation. Ce majorant peut être comparé à l'exigence de délai maximal tolérée par le client. Il permet dans un premier temps de vérifier que les paramètres du système permettent aux applications de respecter leurs échéances. A terme, il permettra en inversant le problème de paramétrer le système d'agrégation en fonction des échéances des applications.

Le délai sera calculé pour chaque MSDU composant l'agrégat. Nous utiliserons donc les courbes dédiées à un flux pour l'exprimer. Comme le montre la figure 2.11, la valeur maximale du délai est obtenue en considérant la rafale maximale σ_1, d'où :

$$d_1(t) \leq \inf\{d \geq 0 | \forall t \geq 0, \beta_1(t+d) \leq \sigma_1\}$$
$$d_1 = \frac{\sigma_1}{\underline{\rho_1}} + \Delta_1$$

Lors du calcul du majorant du délai, il est nécessaire que $\forall i,\ \underline{\rho_i} = \rho_i$. En effet, si cette contrainte n'est pas respectée, le calcul du délai virtuel donnera un majorant infini. Ce résultat est obtenu car le taux de service vaut $\underline{\rho_i}$ alors que la courbe d'arrivée maximale évolue à long terme en suivant ρ_i.

Nous allons considérer plusieurs scénarios de communication pour illustrer le pessimisme de nos majorants. Prenons le même scénario qu'en section 2.4.5. En sommant les différentes latences élémentaires composant la traversée du système nous sommes capables d'estimer une valeur analytique du délai. Dans notre cas, nous additionnons le temps mis pour recevoir la totalité des paquets nécessaires à la formation d'un agrégat ($P + T_R$), le temps de transmission de l'agrégat (T_E) et la latence de traversée de la couche MAC (T_{MAC}) fixée à 2 μs comme spécifié dans (IEEE-802.11n, 2012). Le calcul de ce point de comparaison est illustré en figure 2.12.

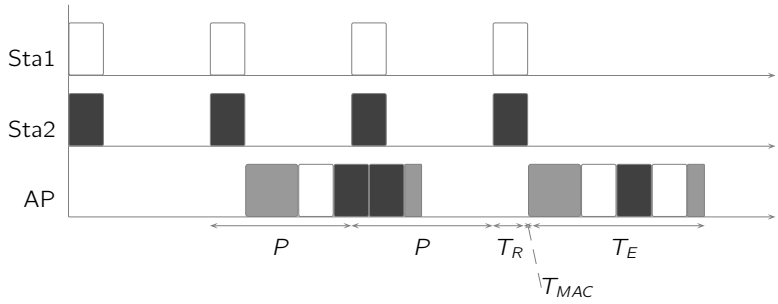


Figure 2.12 – Calcul d'une valeur de comparaison pour le délai

Avec 14 octets d'en-têtes par MSDU et 18 octets d'en-tête MAC pour l'agrégat, la valeur est calculée comme suit,

$$D = 2P + T_R + T_E + T_{MAC} = 2,52 \text{ ms}$$

Calculons maintenant d_1 dans ce scénario. Dans notre cas $s/\underline{\rho_i} > \tau_j - \tau_i$, donc

$$\Delta = \tau_2 + \left(s + \underline{\rho_1}\left(\tau_1 - \tau_2\right)\right) / \left(\underline{\rho_1} + \underline{\rho_2}\right).$$

Nous obtenons $d_1 = 3,91$ ms, avec $\tau_1 = \tau_2 = 1$ ms, $\underline{\rho_1} = \underline{\rho_2} = 1\,000$ octets/ms et $\sigma_2 = 1\,000$ octets.

Le délai virtuel d_1 majore bien le délai calculé analytiquement. L'écart avec D est d'environ 35 % ce qui peut laisser penser que nos hypothèses sont trop pessimistes ou que notre modèle de l'arrivée n'est pas assez précis. Cependant si on calcule la valeur obtenue à l'aide du service résiduel originel, la valeur obtenue est de $7,82$ ms ce qui est plus de 3 fois supérieur à la valeur de comparaison. On notera que la majeure partie

temps (ms)	0	1	1,5	2,1	2,5	2,85
flux	1	2	2	1	1	2
taille (octets)	1000	500	1300	250	650	139

Tableau 2.1 – Caractéristiques des flux entrants

de ce sur-pessimisme est liée à la prise en compte d'un retard maximal d'arrivée τ_1. Or dans cette trace, le flux 1 est arrivé immédiatement ($\tau_1 = 0$).

Ce résultat est de plus uniquement valable pour ce scénario. Considérons maintenant un second scénario, toujours compatible avec les courbes d'arrivée minorantes précédentes. Le tableau ci-dessous donne les caractéristiques des trames en entrée du système pour ce scénario.

La valeur de comparaison devient ici $D = 3,16$ ms. Il est donc possible de se rapprocher du majorant avec un écart de 20 %. Cet écart est en fait principalement lié à la qualité de la modélisation du trafic d'une part et d'autre part, au mode de calcul du délai. En effet, le délai correspond à la distance horizontale entre la courbe de service minimale établie via les courbes d'arrivée minorantes et la courbe d'arrivée majorante. Il est à noter que l'écart entre les courbes d'arrivées majorantes et minorantes, et donc l'incertitude sur la date d'arrivée d'une trame est ici de 2 ms tandis que la différence entre le majorant calculé et la trace est seulement de $0,75$ ms. Par ailleurs, l'erreur sur la majoration de la rafale du flux considéré peut être de surcroît source de pessimisme. Plus l'écart entre ses deux courbes sera faible, plus le majorant calculé sera atteignable.

2.4.6 Discussion de l'ajout d'un seuil temporel

Nous avons vu que dans le cas de 802.11n, le seuil de temps, sorte de chien de garde qui assure que les paquets sortent du système dans un temps défini par configuration, n'existe pas. D'autres systèmes comme 802.15.3c définissent cependant un mécanisme d'agrégation pourvu d'un seuil temporel.

Pour tenir compte de ce seuil il faut établir une seconde courbe de service basée uniquement sur celui-ci. Par exemple, une courbe affine retardée avec un retard égal à ce seuil de temps et dont le taux correspondrait à la quantité de données minimale reçue lors de cette durée. Nous voyons qu'encore une fois, l'utilisation d'une courbe d'arrivée minorante est inévitable.

Afin de construire le modèle d'un système à deux seuils (taille et temps), l'utilisation de la composition de serveurs définie dans le calcul réseau semble toute indiquée. Avec des courbes affines retardées cela se traduit par un cumul des retards et un choix du minimum des taux de service. Cette somme des retards introduits par chacun des seuils n'a pas de sens puisque le comportement du système voudrait que l'on choisisse le minimum d'entre eux. Encore une fois, les outils prévus par le calcul réseau ne semblent pas bien adaptés à la modélisation de ce système puisqu'ils introduisent un sur-pessimisme.

2.4.7 Une première conclusion

Nous avons obtenu une première modélisation du service offert dans le cas d'une agrégation 802.11n. Comparé à la littérature en calcul réseau, la problématique mise en avant concerne ici la relation directe entre arrivée de données et service (voire service résiduel) et la nécessité de disposer de courbe d'arrivée minimale. Nous avons proposé deux courbes de service pour ce système. La courbe du service offert à tous les flux

entrants dans le système permet de calculer un majorant pour l'arriéré de traitement. La courbe du service dédié à un flux particulier permet de majorer le délai subi par ce flux. Le service résiduel tel que défini par le calcul réseau ne correspondait pas à nos attentes. Nous l'avons donc raffiné pour réduire significativement le pessimisme introduit lors de la modélisation de systèmes d'agrégation. Nous avons comparé notre majorant du délai avec une valeur calculée analytiquement sur un scénario d'étude.

Dans la section suivante, nous évaluerons le pessimisme des majorants obtenus avec ce modèle à l'aide d'une étude en profondeur de la norme 802.11n.

2.5 Pessimisme du modèle établi

Cette partie est similaire aux travaux menés par (Ginzburg et Kesselman, 2007). Avec une configuration fixée, la norme 802.11n est étudiée en détail ce qui nous permet de calculer avec précision les délais subis par les paquets. Ces délais sont ensuite comparés avec ceux obtenus en utilisant le modèle établi en section précédente et avec des résultats de simulation.

2.5.1 Cas d'étude

Dans ce scénario, une station Ethernet émet des trames à une station sans fil au travers d'un point d'accès en utilisant un lien Gigabit Ethernet (Figure 3.6). Le point d'accès agrège les paquets en provenance de cette station pour former des A-MSDU[8]. Le flux entrant est composé de paquets de 500 octets émis toutes les 50 μs soit un taux de 80 Mb/s de données utiles. Ce flux appartient à la classe de service *Best-effort* (AC_BE). La configuration des équipements sans fil est résumée dans le Tableau 2.2.

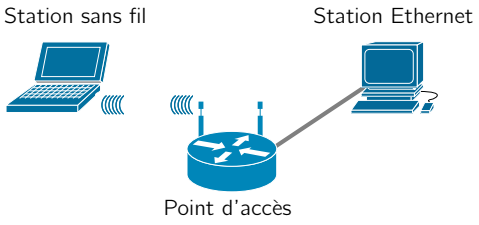


Figure 2.13 – Topologie pour l'évaluation du pessimisme introduit par le modèle

De tels paramètres permettent de déduire une capacité du medium de 65 Mb/s à partir du Tableau 20-30 dans (IEEE-802.11n, 2012). Le taux d'arrivée des paquets est donc supérieur à la capacité du medium, le système est congestionné. Cette condition est nécessaire pour rester dans le domaine de validité de notre modèle. Le système ne doit cependant pas être saturé c'est pourquoi nous considérerons que cette rafale ne dure que 5 ms et qu'elle est répétée toutes les 20 ms minimum (ce qui permet de passer à un taux moyen à long terme de 20 Mb/s).

8. définies au paragraphe 8.3.2.2, (IEEE-802.11n, 2012)

Access medium policy	HCF EDCA
A-MSDU	3839 bytes
A-MPDU	desactivé
Block Ack	desactivé
Backoff	forcé à 0
TXOP limit	une MSDU
AIFSN	3
SigExt	6 μs
ISM Band	2.4 GHz
Base Data Rate	65 Mb/s
Maximal Data Rate	600 Mbps
Spatial Stream	1
Guard Interval	800 ns (regular)
Channel spacing	20 MHz
Slot Time	9 μs
SIFS	10 μs
PPDU format	HT-mixed

Tableau 2.2 – Configuration des stations sans fil

2.5.2 Résultats théoriques (IEEE 802.11n)

Avec un seul flux dans notre scénario, aucune station n'occupera le medium excepté le point d'accès. Il agrégera donc des paquets lorsqu'il émet lui-même sur le medium. Ainsi, le premier paquet reçu sera immédiatement émis sans être agrégé.

Pendant la durée de cette transmission, le point d'accès recevra plusieurs paquets et sera à même de former un premier agrégat à transmettre ensuite. Le scénario décrit dans la section précédente a été choisi afin de respecter la condition de congestion nécessaire à l'activation de l'agrégation. Dans le cas contraire, aucun agrégat n'aurait été émis. La quantité de paquets reçue est alors supérieure à la capacité de traitement du système. A terme, cela conduira à des pertes de paquets par débordement des mémoires tampons. La station Ethernet émet ici des paquets sur une durée limitée (5 ms) pour éviter ce phénomène. La valeur maximale de cette durée est directement liée à la taille de la mémoire au niveau de la couche MAC du point d'accès. Avant toute perte de paquet, des agrégats de taille croissante sont émis jusqu'à atteindre le seuil de taille. Puis des agrégats de taille maximale sont émis car seule la première condition du corollaire s'applique.

L'objectif de cette section est de définir, en analysant la norme, les étapes nécessaires à l'émission d'une trame et de calculer le temps d'émission de chaque agrégat jusqu'à atteindre la taille maximale des agrégats (limitée par les seuils définis par la norme). Ces valeurs seront comparées dans la section suivante avec les résultats issus d'une simulation.

2.5.2.a Règles d'émission avec HCF EDCA

Comme introduit dans le chapitre 1, nous utilisons la méthode d'accès au medium *HCF EDCA* qui permet à chaque station virtuelle [9] de gagner une *TXOP* [10] en fonction du temps d'attente aléatoire tiré avant transmission. La station qui tire le temps le plus court peut transmettre. Ce temps d'attente est composé de l'AIFS défini pour la station et du *backoff* tiré aléatoirement entre 0 et *CW*. Cependant afin de pouvoir calculer le temps d'attente et de le comparer avec les résultats issus de notre modèle analytique nous éliminons le tirage aléatoire du *backoff*. C'est pourquoi il n'apparaît pas en Figure 2.14. L'absence de *backoff* peut conduire à des collisions internes ce que nous souhaitons éviter. Dans le scénario présenté ici, seule une classe de service sur une unique station émet des trames ce qui évite toute collision.

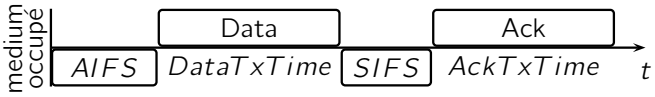


Figure 2.14 – Étapes d'une transmission avec HCF EDCA

La Figure 2.14 représente les étapes nécessaires à l'émission et à l'acquittement d'une *PPDU* [11] de données à partir des choix effectués pour ce cas d'étude (section 2.5.1). La totalité du temps nécessaire à la transmission réussie d'une trame est donc constituée d'un *AIFS*, du temps de transmettre une trame de donnée (*DataTxTime*), d'un *SIFS* et du temps de recevoir un acquittement (*AckTxTime*)

La durée limite de la TXOP est configurée ici à une MSDU. Cette valeur signifie qu'une station peut émettre seulement une MSDU ou une A-MSDU [12] lorsqu'elle gagne le medium. L'AIFS et calculé suivant l'équation (2.10) dans laquelle le *SlotTime* se traduit par tranche canal [13].

$$AIFS = SIFS + AIFSN * SlotTime \tag{2.10}$$

Nous avons vu au chapitre précédent que la valeur de l'*AIFSN* varie pour chaque classe de service (pour mémoire les valeurs de l'*AIFSN* sont rappelées en Tableau 2.3 [14]).

AC	Nom commun	AIFSN
AC_BK	Background	7
AC_BE	Best effort	3
AC_VI	Video	2
AC_VO	Voice	2

Tableau 2.3 – Valeurs par défaut de l'AIFSN en fonction de sa classe de service

La méthode de calcul du temps de transmission d'une trame de donnée est présentée en partie 2.5.2.b de cette section. L'acquittement par bloc (*Block Ack*) étant désactivé, chaque MPDU est acquittée immédia-

9. une station est composée de quatre stations virtuelles qui sont ses classes de service
10. *TXOP*, l'opportunité de transmettre définie par un instant de début et une durée limite
11. *Physical PDU*, l'unité échangée entre les deux couches physiques lors d'une transmission
12. comme décrit paragraphe 9.19.2.2, (IEEE-802.11n, 2012)
13. la tranche canal est définie par les caractéristiques physiques de la chaîne de transmission comme la bande de fréquence utilisée pour émettre
14. établi à partir du Tableau 8-105, (IEEE-802.11n, 2012)

tement par l'émission d'une trame d'acquittement (Ack) après un SIFS. Le calcul du temps de transmission d'un acquittement est décrit dans la partie 2.5.2.c de cette section.

2.5.2.b Calcul du temps d'émission d'une trame de données

Les paramètres du cas d'étude permettent d'identifier le format de la trame MAC décrit en Figure 2.15. Le champ *Data* peut alors contenir une MSDU ou une A-MSDU. Comme dit précédemment, le premier paquet sera émis immédiatement après réception par le point d'accès mais les paquets suivants seront agrégés. Agréger implique un nouveau format de trame avec de nouvelles en-têtes et de nouvelles règles de remplissage.

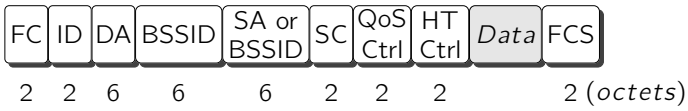


Figure 2.15 – Format d'une trame de données

Le format d'une A-MSDU est décrit en Figure 2.16. Chaque sous-trame de l'A-MSDU (sauf la dernière) est remplie pour que sa taille soit un multiple de 4 octets [15].

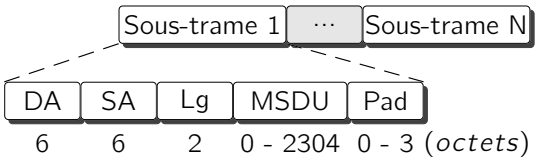


Figure 2.16 – Format d'une A-MSDU

Tous les paramètres concernant le format de la trame sont maintenant définis. Le reste de cette section explique le format de PPDU choisi et le temps nécessaire à son émission.

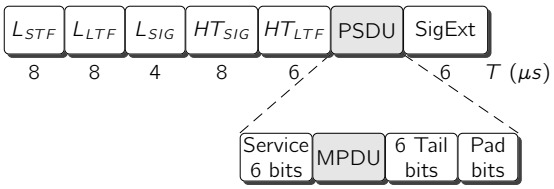


Figure 2.17 – Format d'une PPDU HT Mixed

Le format de PPDU défini est HT-mixed (comme défini Tableau 2.2). Ce format est décrit en Figure 2.17 [16]. Avec un seul flux spatial et sans STBC (Space Time Block Coding) les stations sans fil n'utilisent qu'un seul flux spatio-temporel [17]. STBC peut être utilisé lorsque l'émetteur possède au moins deux chaînes de transmission [18]. Un seul flux spatio-temporel conduit à un seul champ Data HT-LTF [19] et

15. ce pré-requis sert en particulier à la décomposition des agrégats à la réception : paragraphe 8.3.2.2, (IEEE-802.11n, 2012)
16. cette figure est établie à partir de la Figure 20-1, (IEEE-802.11n, 2012)
17. comme indiqué en Tableau 20-12, (IEEE-802.11n, 2012)
18. décrit en Tableau 20-21, (IEEE-802.11n, 2012)
19. voir le Tableau 20-13, (IEEE-802.11n, 2012)

sans flux spatial d'extension, il n'y a aucun champ Extension HT-LTF [20]. Comme décrit dans la section 9.3.8 de (IEEE-802.11n, 2012), une durée SigExt sans aucune transmission doit être respectée après l'émission d'une PPDU au format HT-mixed.

Avec toutes ces informations nous pouvons maintenant calculer le temps de transmission de cette trame de données (équation (2.11) [21]).

$$DataTxTime = T_{L_PRE} + T_{L_SIG} + T_{HT_PRE} + T_{HT_SIG} + T_{SYM} * N_{SYM} + SigExt \tag{2.11}$$

$$\text{avec, } N_{SYM} = m * \lceil \frac{16 + 8 * l + 6 * N_{ES}}{m * N_{DBPS}} \rceil \text{ et } T_{L_PRE} = T_{L_STF} + T_{L_LTF} \tag{2.12}$$

La signification des paramètres de ces équations est donnée en annexe A. Les valeurs des quatre premiers termes de l'équation (2.11) sont identifiées à partir de la configuration de la couche physique des stations de ce cas d'étude (section 2.5.1). La dernière partie de cette équation donne le temps nécessaire à la transmission du champ *Data*. Cette transmission doit respecter un nombre entier de symboles *OFDM* (Orthogonal Frequency-Division Multiplexing) [22]. Des bits de remplissage sont donc ajoutés à ce champ *Data*. Dans le calcul de N_{SYM} (équation (2.12), le numérateur de la fraction représente la longueur des données de la PPDU (Figure 2.17) et l représente la taille de la MPDU en octets (Figure 2.15).

2.5.2.c Calcul du temps de transmission d'un acquittement

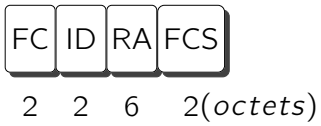


Figure 2.18 – Format d'un acquittement

Les acquittements (Ack) sont définis comme des trames de contrôle. Elles doivent être émises dans une PPDU différente de celle utilisée pour les données. En effet, il y a sept conditions [23] à remplir pour pouvoir émettre un acquittement dans une PPDU HT. Aucune de ces conditions n'est respectée dans notre cas. Les acquittements sont donc émis dans une PPDU non-HT. Le format de trame d'un acquittement et celui d'une PPDU non-HT sont respectivement illustrés en Figure 2.18 et 2.19.

Afin de calculer le temps nécessaire à la transmission d'un acquittement, le débit doit être sélectionné en deux étapes :

— identifier le débit de référence (non-HT) relatif au débit de transmission de la trame précédente [24]

— choisir le plus grand débit défini pour ce BSS qui est supérieur ou égal à ce débit de référence.

Le débit de transmission choisi pour les trames est de 65 Mb/s. Il correspond au MCS (Modulation and Coding Scheme) [25] numéro sept (modulation 64-QAM, taux de codage 5/6) [26]. À partir du Tableau 9-5

20. d'après le Tableau 20-14, (IEEE-802.11n, 2012)
21. établie à partir des équations (20-92) et (20-32), (IEEE-802.11n, 2012)
22. OFDM est le codage utilisé ici
23. listées en paragraphe 9.7.6.1 de (IEEE-802.11n, 2012)
24. voir les paragraphes 9.7.6.5.2 et 9.7.9, (IEEE-802.11n, 2012)
25. le MCS est une spécification du codage et de la modulation utilisée dans 802.11n
26. détaillé dans le Tableau 20-30 de (IEEE-802.11n, 2012) (bande passante 20 MHz, un flux spatial)

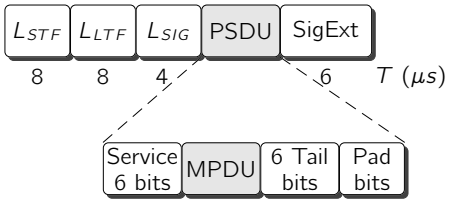


Figure 2.19 – Format d'une PPDU non-HT

(IEEE-802.11n, 2012), le débit de référence déduit est égal à 54 Mbps.

"If the received frame is of the modulation class HT and the control response frame is carried in a non-HT PPDU, the control response frame shall be transmitted using one of the ERP-OFDM or OFDM modulation."[27]

Une modulation OFDM implique un ensemble de débits obligatoire composé de 6, 12 et 24 Mb/s pour les trames de contrôle. [28]

"Only the data transfer rates of the mandatory rate set of the attached PHY are guaranteed to be supported [...]. Higher layer protocols may negotiate a rate outside the mandatory rate set."[29]

Sans information sur les débits négociés par les protocoles des couches supérieures, le débit de transmission d'un acquittement est égal à 24 Mb/s. Au même titre que pour la transmission d'une trame de données, une durée SigExt doit être respectée lorsque le format de PPDU est non-HT. Le temps de transmission de cet acquittement est calculé selon l'équation (2.13) [30] :

$$AckTxTime = T_{PRE} + T_{SIG} + T_{SYM} * N_{SYM} + SigExt \quad (2.13)$$

$$\text{avec, } N_{SYM} = \left\lceil \frac{16 + 8 * l + 6}{N_{DBPS}} \right\rceil \quad (2.14)$$

La signification des paramètres de ces équations est donnée en annexe A. Les équations (2.11) et (2.12) sont respectivement proches des équations (2.13) et (2.14). Les explications sur la signification de chaque partie des équations posées ici sont donc les mêmes que pour les équations précédentes.

2.5.3 Application numérique

Le chronogramme présenté en Figure 2.20 représente les flux entrants et sortants du système. Cette section a pour but d'expliquer le raisonnement conduisant à ce chronogramme tout en calculant les valeurs temporelles de chaque étape.

Les valeurs contenues dans le Tableau 2.4 ont été collectées à partir des Tableaux (20-6) et (20-30) (IEEE-802.11n, 2012). Celles contenues dans le Tableau 2.5 proviennent des Tableaux (18-4) et (18-5) (IEEE-802.11n, 2012). L'application respective des valeurs des Tableaux 2.4 et 2.5 aux équations (2.11) et (2.13) conduit aux résultats suivants (équation (2.15) et (2.16)).

27. citation du paragraphe 9.7.6.5.2, (IEEE-802.11n, 2012)
28. voir Tableau 18-16, (IEEE-802.11n, 2012)
29. citation du paragraphe 9.7.1, (IEEE-802.11n, 2012)
30. établie à partir des équations (18-29) et (18-11), (IEEE-802.11n, 2012)

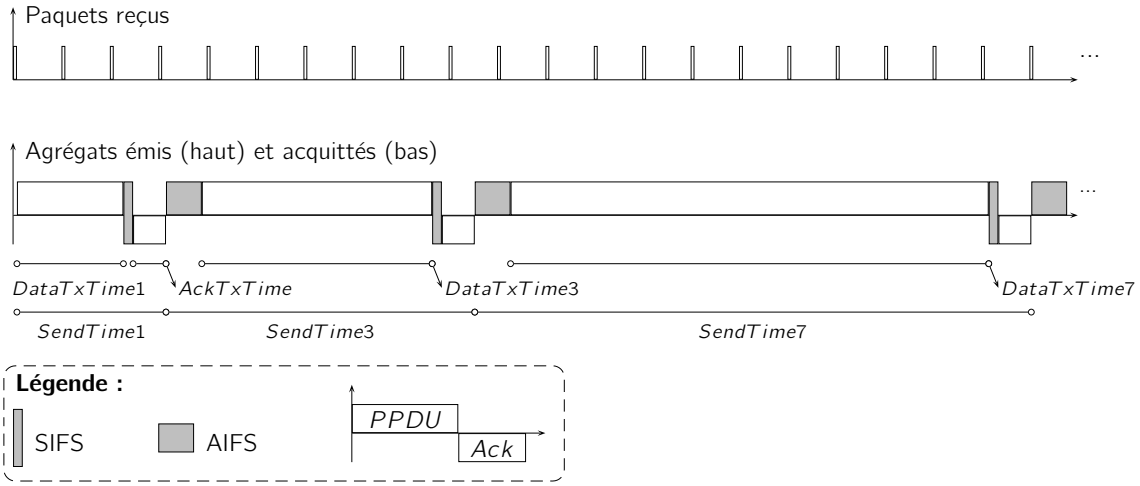


Figure 2.20 – Chronogramme entre entrée et sortie de la couche MAC de 802.11n lorsque l'agrégation est active

Paramètre	Valeur
T_{L_PRE}	16 μs
T_{L_SIG}	4 μs
T_{HT_PRE}	8 μs
T_{HT_SIG}	8 μs
T_{SYM}	4 μs
m	1
N_{DBPS}	260
N_{ES}	1

Tableau 2.4 – Valeurs appliquées au temps de transmission des données

Le temps nécessaire à l'émission d'une trame de données contenant une seule MSDU est calculé dans l'équation 2.15. La numérotation de chaque valeur de $DataTxTime$ correspond au nombre de MSDUs contenues dans l'agrégat. Le format des sous-trames de cet agrégat est décrit en Figure 2.16.

$$\begin{aligned} l_{data1} &= 30 + 500 + 4 = 534 \text{ octets} \\ DataTxTime1 &= 36 + 4 * \left\lceil \frac{16 + 8 * 534 + 6}{260} \right\rceil + 6 \\ &= 36 + 4 * 17 + 6 = 110\ \mu s \end{aligned} \tag{2.15}$$

Le temps d'émission des acquittements suivant chaque transmission d'une trame de données est calculé

Paramètre	Valeur
T_{PRE}	16 μs
T_{SIG}	4 μs
T_{SYM}	4 μs
N_{DBPS}	192

Tableau 2.5 – Valeurs appliquées au temps de transmission des acquittements

en équation 2.16.

$$\begin{aligned} l_{ack} &= 14 \text{ octets} \\ AckTxTime &= 20 + 4 * \left\lceil \frac{16 + 8 * 14 + 6}{192} \right\rceil + 6 \\ &= 20 + 4 * 2 + 6 = 34\ \mu s \end{aligned} \tag{2.16}$$

Le medium ayant été libre pour une durée supérieure à un AIFS, le premier paquet est immédiatement émis. Le temps total nécessaire à l'émission de la première trame est calculé dans l'équation (2.17).

$$\begin{aligned} SendTime1 &= DataTxTime + SIFS \\ &+ AckTxTime = 154\ \mu s \end{aligned} \tag{2.17}$$

Après écoulement de cette durée, une nouvelle transmission va débuter mais elle est précédée d'un AIFS. Lors de cet AIFS, le système poursuit l'accumulation de paquets afin de former un agrégat. Après $SendTime1 + AIFS = 191\ \mu s$, trois paquets attendent en mémoire (arrivés à 50, 100 et 150 μs).

Procédons au calcul du temps requis pour émettre cet agrégat de trois paquets. Sans procédure de tirage du $backoff$, seule la taille de la trame de données l change ce qui modifie la valeur de son temps de transmission, $DataTxTime$ (equation (2.19)).

$$\begin{aligned} l_{data3} &= 30 + 3 * 14 \text{ (en-têtes)} + 3 * 500 + 2 * 2 \text{ (remplissage)} + 4 \\ &= 1580 \text{ octets} \end{aligned} \tag{2.18}$$

$$DataTxTime3 = 36 + 4 * 49 + 6 = 238\ \mu s \tag{2.19}$$

Cette nouvelle taille de MPDU (équation (2.18)) est calculée en utilisant les informations sur les A-MSDU données en partie 2.5.2.b. Calculée de la même manière que dans l'équation (2.17), la valeur du temps total d'émission est donnée dans l'équation (2.20).

$$\begin{aligned} SendTime3 =& AIFS + DataTxTime + SIFS + AckTxTime = 319\ \mu s \\ \text{avec, } AIFS =& 3 * 9 + 10 = 37\ \mu s \end{aligned} \tag{2.20}$$

Après $SendTime3 + AIFS = 356\ \mu s$, sept paquets attendent en mémoire. Les paquets arrivent toujours toutes les 50 μs et nous sommes à $t = 547\ \mu s$.

$$\begin{aligned} l_{data7} &= 30 + 7 * 14 + 7 * 500 + 6 * 2 + 4 \\ &= 3644 \text{ octets} \\ DataTxTime7 &= 36 + 4 * 113 + 6 = 494\ \mu s \quad (2.21) \\ SendTime7 &= 575\ \mu s \quad (2.22) \end{aligned}$$

Les valeurs données par les équations (2.21) et (2.22) concernent un agrégat de sept paquets. Après l'écoulement de $SendTime7 + AIFS = 612\ \mu s$, onze paquets attendent en mémoire mais le système n'est pas capable d'envoyer un agrégat d'une taille supérieure à sept paquets à cause du seuil de 3839 octets. Le nombre de paquets dans la mémoire va donc croître à l'infini tout comme le délai d'accès au medium.

Pour éviter les débordements mémoire, les stations arrêtent d'émettre après 5 *ms*. Cette limite temporelle permet de calculer une approximation B de la taille minimale de la mémoire dans le but d'éviter toute perte (équation (2.23)).

$$B = \left\lceil \frac{20}{0.575} \right\rceil * (12 - 7) * 500 * 8 = 700000 \text{ bits} \quad (2.23)$$

Nous avons obtenu les valeurs nécessaires à une comparaison, mais à quel prix. Le nombre de calcul à effectuer est fonction du nombre de paquets reçus et des caractéristiques du flux. Déjà nombreux alors que les paquets sont de taille fixe, le nombre de calculs à effectuer peut exploser avec des variables supplémentaires comme la taille des paquets, le nombre de flux, etc. Dans la suite de cette section, nous comparerons ces résultats d'abord avec Riverbed Modeler et ensuite avec le modèle calcul réseau établi précédemment.

2.6 Comparaison des différentes méthodes d'évaluation de performances

2.6.1 Résultats du simulateur confrontés à ceux de l'analyse de la norme

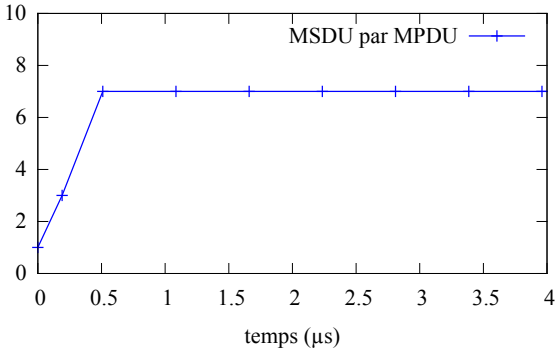
Le choix de Riverbed Modeler en tant que simulateur a été motivé en section 2.2.1. Cet outil permet de choisir des modèles de propagation particuliers aux réseaux sans fil. Dans notre cas, nous nous sommes attachés à considérer le modèle de propagation en espace libre ce qui nous assure, au vu des faibles distances choisies, un très faible BER et ainsi de ne subir aucune perte de paquet.

Les paramètres du simulateur (en particulier ceux du Tableau 2.2) sont choisis en considérant le cas d'étude utilisé à la section précédente. Une taille de 1024000 bits pour la mémoire permet de respecter la contrainte évoquée à la fin de la partie 2.5.3. La station sans fil et le point d'accès sont placés proches l'un de l'autre afin de minimiser le délai de propagation. Les sondes de collecte choisies sont listées dans le Tableau 2.6.

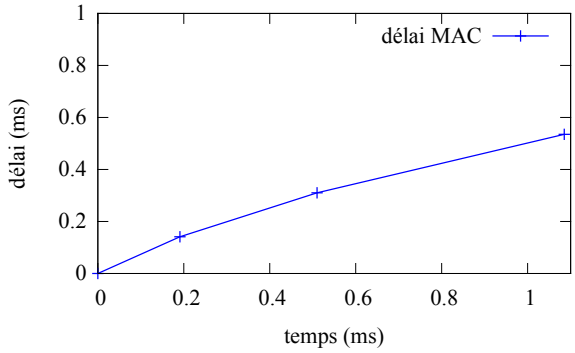
En complément des sondes listées dans le Tableau 2.6, plusieurs sondes collectant les pertes de paquets ont été placées depuis la couche MAC de la station Ethernet jusqu'à la couche MAC de la station sans fil. Les valeurs nulles collectées par celles-ci nous assurent qu'aucune perte n'est apparue. D'autre part, les statistiques sur le nombre de paquets dans l'agrégat (Figure 2.21a) correspondent aux résultats donnés par

Nom	Point de collecte
load (b/s)	point d'accès
data traffic sent (b/s)	point d'accès
data traffic received (b/s)	station sans fil
control traffic sent (b/s)	station sans fil
control traffic received (b/s)	point d'accès
MSDU per MPDU	point d'accès
MAC queue size (packets)	point d'accès
MAC delay (s)	point d'accès

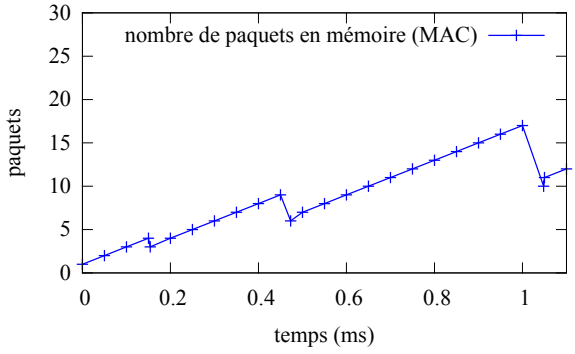
Tableau 2.6 – Sondes choisies dans Riverbed Modeler



(a) Nombre de MSDU par MPDU



(b) Délai d'accès au medium



(c) Nombre de paquets dans la mémoire de la couche MAC

Figure 2.21 – Statistiques relevées avec Modeler pour une comparaison avec notre étude de la norme

la norme avec un paquet, puis trois paquets, puis sept paquets jusqu'à la fin de la simulation. De plus, les graphiques représentant le délai d'accès au medium (Figure 2.21b) et le nombre de paquet en mémoire (Figure 2.21c) montrent que leurs valeurs augmentent continuellement ce qui constitue le comportement attendu. Comme le montre la Figure 2.21c tant qu'un agrégat est en formation le nombre de paquets en mémoire augmente mais dès lors qu'un agrégat est définitivement libéré tout les paquets qu'il contient sont supprimé de la mémoire ce qui provoque une chute de la courbe à cet instant.

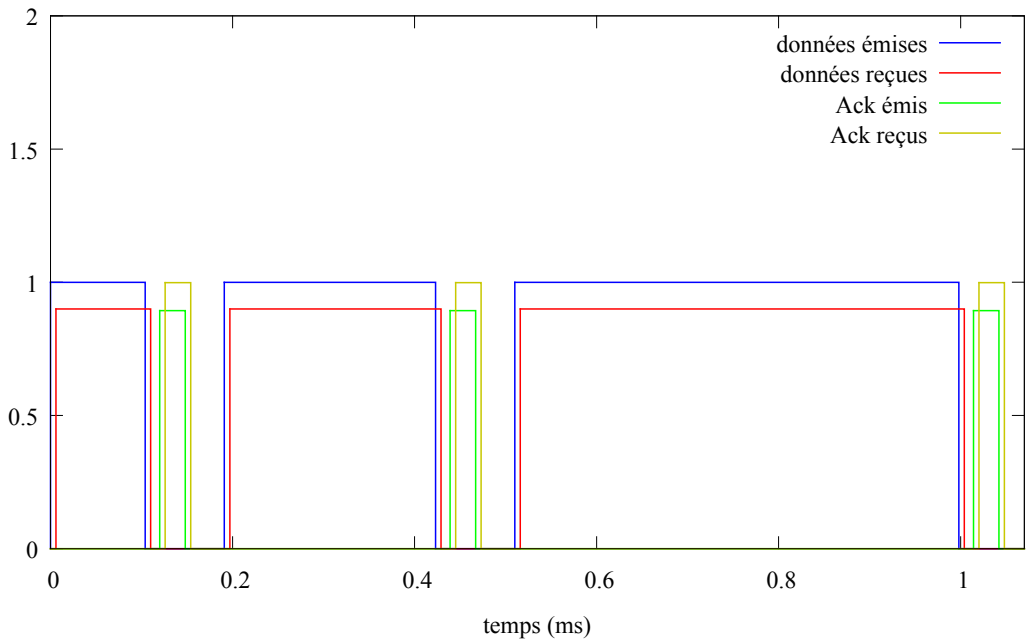


Figure 2.22 – Comparaison d'un relevé du trafic observé sur le réseau sans fil avec une prévision issue de notre étude de la norme

La Figure 2.22 représente le trafic sur le réseau sans fil vu du point d'accès (courbe haute) et vu de la station sans fil (courbe basse). Ce graphique permet de calculer les valeurs des temps d'émission d'un agrégat (données) et d'un acquittement. Le Tableau 2.7 montre que la simulation approche les résultats calculés à partir de la norme de 19 *ns*. Cet écart de 19 *ns* entre les résultats correspond au délai de propagation, étant donné qu'il est constant pour chaque trame émise. En faisant abstraction de cet écart, on remarque une parfaite égalité entre les délais calculés suivant la norme et ceux observés en simulation. Ces résultats nous conduisent donc à être confiants avec l'implémentation du système d'agrégation de 802.11n au sein de Riverbed Modeler, au moins sur ce cas d'étude.

Nom	Norme	Simulation
$AckTxTime$	34 μs	34.019 μs
$DataTxTime1$	110 μs	110.019 μs
$SendTime1$	154 μs	154.038 μs
$DataTxTime3$	238 μs	238.019 μs
$SendTime3$	319 μs	319.038 μs
$DataTxTime7$	494 μs	494.019 μs
$SendTime7$	575 μs	575.038 μs

Tableau 2.7 – Comparaison des délais fournis d'une part par notre analyse de la norme et d'autre part par le simulateur

2.6.2 ...confrontés au modèle analytique établi

Afin de rester dans le domaine de validité du modèle analytique et ainsi pouvoir le comparer avec le second modèle et les simulations, nous avons vu qu'un profil de trafic particulier est considéré. Les caractéristiques choisies précédemment sont réutilisées pour effectuer la comparaison, ce qui nous conduit à un profil de trafic constitué de rafales de 5 ms toutes les 20 ms minimum. Ces rafales sont elles même constituées de paquets de 500 octets émis toutes les 50 μs.

Chaque flux entrant i est enveloppé par une courbe minorante et une courbe majorante exprimées en équations 2.24 et 2.25 respectivement.

$$\underline{\beta_i} = \rho_i (t - \tau_i)^+ \tag{2.24}$$

$$\beta_i = \alpha_i + \rho_i t \tag{2.25}$$

$$\text{avec, } \rho_i = 20 \text{ Mb/s}$$

$$\tau_i = 50 \ \mu s$$

$$\alpha_i = 320 \text{ kbits}$$

Cette configuration nous conduit à exprimer Δ_1 comme défini en équation 2.26 (car $\tau_1 = \tau_2$) et à calculer un majorant du délai de traversée de la couche MAC (équation (2.27).

$$\Delta_1 = \tau_1 + \frac{S}{\rho_1 + \rho_2} = 0.00005 + \frac{3839 * 8}{40000000} = 817,8 \ \mu s \tag{2.26}$$

$$d_1 = \frac{\sigma_1}{\rho_1} + \Delta_1 = 16,8178 \text{ ms} \tag{2.27}$$

Calculons maintenant la valeur obtenue en utilisant notre connaissance de la norme. Nous avons besoin de calculer l'écart entre le moment d'émission du dernier agrégat et l'instant de réception du premier paquet qui compose cet agrégat. D'abord l'instant d'émission du dernier agrégat : nous avons vu qu'il faut $SendTime1 + SendTime3$ pour transmettre 4 paquets. Vu les caractéristiques évoquées, il y a 100 paquets par flux dans une rafale soit 200 avec deux flux. Il reste donc 196 paquets à transmettre ce qui fait 28 agrégats de taille maximale de 7 paquets. Pour transmettre un agrégat de 7 paquets il faut $SendTime7$ donc le dernier agrégat sera transmis à $t = t_d$:

$$t_d = SendTime1 + SendTime3 + 28 \times SendTime7 = 0,154 + 0,319 + 0,575 * 28 = 16,573 \text{ ms}$$

Le premier paquet qui compose cet agrégat est reçu à $t = t_a$, six périodes avant la fin de la rafale de 5 ms :

$$t_a = 5 - 6 * 0,05 = 4,7 \ ms$$

Ainsi d_1', le pire délai subi par un paquet suivant le scénario défini vaut :

$$d_1' = t_d - t_a = 16,573 - 4,7 = 11,873\ ms$$

L'écart avec la valeur obtenue en calcul réseau est de presque 5 ms ce qui représente une majoration de l'ordre de 40 %. Le pessimisme est important mais il est intéressant de noter à quel point les valeurs de d_1 et t_d sont proches. En théorie, il est possible que les paquets arrivent à une plus grande vitesse, que la rafale soit plus prononcée mais qu'elle respecte la contrainte imposée par la courbe d'arrivée majorante. Si les paquets se suivaient sur un réseau Gigabit il faudrait environ 400 μs pour recevoir la rafale entière et moins de 50 μs sur un réseau à 10 Gb/s. Dans ce cas le pessimisme de notre modèle serait réduit à moins de 2 %. Une solution pour améliorer la précision du modèle est d'utiliser des courbes d'arrivée différentes des affines considérées ici (voir (Mao et Panwar, 2006) pour un recensement des modèles de courbes d'arrivées). Encore une fois, l'écart entre l'enveloppe sur l'arrivée et l'arrivée réelle ($\alpha(t)$ et $R(t)$) produit la majeure partie du pessimisme observé.

2.7 Conclusion

Nous avons établi un modèle analytique d'un système d'agrégation de paquets en utilisant la théorie du calcul réseau. Le pessimisme de ce modèle a été évalué à l'aide d'une analyse approfondie de la norme 802.11n qui a été vérifiée par des simulations utilisant Riverbed Modeler.

Les résultats obtenus montrent que le modèle calcul réseau souffre d'un certain pessimisme, qui peut toutefois se limiter à quelques % dans certaines configurations.

Les méthodes d'évaluation de performances proposées ne sont pas exemptes de défauts. L'évaluation via une analyse de la norme implique une connaissance intime du trafic et requiert une série de calculs fonction du nombre de paquets entrant dans le système pour obtenir une valeur de délai. Le modèle analytique résout un certain nombre de ces problèmes mais pour conserver cette simplicité nous avons du restreindre son domaine d'application. De plus, le mécanisme de tirage du *backoff* a été négligé puisque son caractère aléatoire aurait empêché la comparaison entre modèles.

Enfin, la dynamique du système vis à vis des paramètres de contrôle n'est jamais considérée. L'accès au medium non déterministe peut être résolu par l'utilisation des méthodes définies dans le calcul réseau stochastique (voir (Bredel et Fidler, 2009; Xie et Jiang, 2010) pour des modèles de l'accès au medium DCF établis avec cette approche).

Comme ces méthodes d'évaluation ont montré leurs limites, nous nous tournons dans le chapitre suivant vers une étude du système d'agrégation de 802.11n basée sur l'utilisation intensive de Riverbed Modeler. Cette étude permet de considérer une modification des paramètres de configuration de la couche MAC 802.11e. Ce simulateur nous permettra à terme d'établir et de qualifier un algorithme de configuration dynamique du système d'agrégation basé sur des exigences de qualité de service.

Chapitre 3

Vers une régulation des paramètres d'agrégation

Ce chapitre étudie l'influence des paramètres d'agrégation sur quelques indicateurs de QdS afin d'établir une régulation du système au chapitre suivant. Cette régulation servira à adapter les paramètres d'agrégation du système afin de lui permettre de respecter des contraintes/exigences de qualité de service (QdS) fournies d'une part par le client et d'autre part par le fournisseur d'accès. Cette boucle de retour est illustrée en figure 3.1.

Nous considérons donc un système "boite noire" paramétré par un opérateur et recevant des flux utilisateurs. Ce système, représenté en figure 3.1, reproduit la dynamique d'une file associée à une classe de service au niveau MAC. Les flux utilisateurs en entrée du système sont spécifiés selon la distribution suivie par l'inter-arrivée et la taille des paquets. Une distinction est représentée entre le flux considéré et ses concurrents puisque nous cherchons à isoler un flux spécifique en vue d'évaluer ses performances (contrairement à la plupart des articles publiés sur cette agrégation (Kuppa et Dattatreya, 2006; Lin et Wong, 2006; Ginzburg et Kesselman, 2007; Skordoulis *et al.*, 2008; Wang et Wei, 2009).

Ce positionnement nous conduit à relever le délai entre l'arrivée d'un paquet dans la couche MAC et le départ de l'agrégat qui le contient. D'autres indicateurs seront relevés, en particulier l'arriéré de traitement qui, comparé à la taille mémoire imposée par le constructeur de l'équipement, nous assure que la mémoire n'est pas saturée. De plus, il permet de représenter visuellement les quantités transmises à chaque accès au medium. Enfin, le troisième indicateur sera le nombre de paquets perdus lors d'une transmission qui n'est qu'une extension de l'arriéré de traitement mais qui nous permet de juger des améliorations apportées par l'agrégation dans certains cas extrêmes.

Afin de modifier le comportement du système, un certain nombre de paramètres peuvent être configurés comme l'activation de l'agrégation sur certaines files uniquement, un changement du seuil de taille des agrégats ou encore une modification de la durée de TXOP. De notre point de vue, ces paramètres sont contrôlés par le fournisseur d'accès même si dans le cas de 802.11n il est possible que le client soit apte à les modifier (dans le cas d'un point d'accès sans fil domestique). Parmi ces paramètres de contrôle nous nous intéresserons uniquement à l'agrégation de MSDU et à ses paramètres. Une étude en surface de l'influence du mécanisme du *backoff* sera néanmoins menée afin de distinguer ses effets de ceux de l'agrégation.

Nous avons vu au chapitre précédent les limites inhérentes aux modèles analytiques proposés. C'est

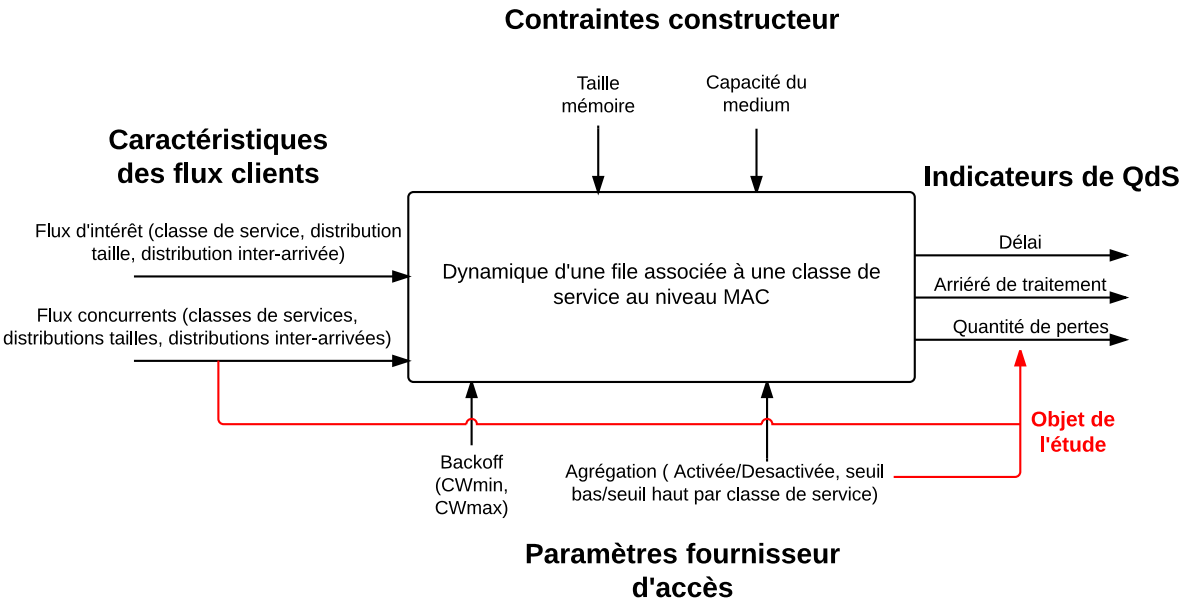


Figure 3.1 – Représentation du système sous forme de boîte noire et positionnement des différentes contraintes, indicateurs et paramètres du système

pourquoi dans ce chapitre nous utiliserons un outil de simulation aux capacités plus versatiles. Les principaux simulateurs utilisés dans les études précédentes sont ns2 et Riverbed Modeler (ex Opnet Modeler). D'autres auteurs utilisent des simulateurs qu'ils ont eux-mêmes créés ce dont nous n'avons pas besoin puisque nous ne modifions en rien le standard. Nous choisissons donc l'un des deux simulateurs listés, Riverbed Modeler.

Cet outil fait office de référence en simulation de réseaux et est notamment utilisé pour la spécification de normes comme 802.11n. Dès les prémices de cette norme, Intel a développé un modèle de ce que devait être l'agrégation de paquets mais malgré nos efforts, il nous a été impossible d'obtenir ce modèle. Ce n'est plus un problème aujourd'hui puisque Modeler implémente nativement et entièrement les mécanismes d'agrégation de 802.11n. Il nous permettra alors d'étudier en profondeur l'impact des paramètres d'agrégation.

Concernant son fonctionnement, il embarque un moteur de simulation à événement discret. Ses modèles sont composés de machines à états dont chaque état est programmé en pseudo-langage C. Chaque modèle représente le fonctionnement prévu par le constructeur de l'équipement (lorsqu'il écrit le modèle) ou attendu selon la norme en vigueur (dans le cas de modèles génériques). Les tailles mémoires, les fréquences processeurs et les capacités des cartes réseaux sont autant de paramètres permettant des calculs proches de la réalité. Nous utilisons des modèles génériques puisque nous cherchons à évaluer le système tel que prévu dans la norme IEEE 802.11n. Dans Modeler, le trafic est modélisé par des processus dont les paramètres permettent d'obtenir tous les types de trafic (périodique, apériodique, stochastique, ...). Nous restreindrons cependant l'étude aux processus périodiques constants d'une part, et aux processus suivant une loi de poisson d'autre part (exponentielle sur l'inter-arrivée et la taille des paquets). Un modèle de propagation peut être choisi afin d'influencer les temps de propagation des trames. Nous choisissons le modèle de propagation par défaut (espace libre) ce qui nous assure, au vu des faibles distances choisies, de ne subir aucune perte de paquets lors de leur propagation (hors collision).

Ce cadre qui vient d'être ébauché est détaillé et chiffré en section 3.1 accompagné d'une description

des scénarios de simulation choisis. La section 3.2 introduit quelques résultats de simulation sur un scénario simple tout en comparant différents niveaux de concurrence (au niveau de l'entrée et de la sortie de la couche MAC). La section 3.3 présente des extraits de simulations réalisées avec des scénarios de plus forte complexité. Elle concerne cependant un seul scénario de concurrence afin de réduire légèrement cette complexité. Elle est divisée en plusieurs sous-sections qui étudient chacune l'influence d'un paramètre qui ne peut être maîtrisé par le fournisseur d'accès (*backoff*, taille et inter-arrivée des paquets). La section 3.4 se concentre sur l'évaluation de l'influence des paramètres d'agrégation sur les indicateurs choisis (cf figure 3.1). Chacun de ces paramètres sera étudié avec des caractéristiques de flux variées et à l'aide de trois indicateurs : l'arriéré de traitement, le délai et les pertes. Enfin, la section 3.4.4 propose un tableau de synthèse des résultats observés ainsi que plusieurs tableaux des résultats mesurés. L'impact des règles comportementales déduites de cette étude y est discuté et les recommandations établies y sont rappelées.

3.1 Périmètre de l'étude

Rappelons que le système d'agrégation opère sur les MSDU reçues par la couche MAC de 802.11n. Ces MSDU sont stockées dans une file d'attente associée à une classe de service comme illustré en figure 3.2. Ce système clôt un agrégat si :

— la quantité de données dans cette mémoire est supérieure ou égale à un seuil de taille S.

— la classe de service associée à cette file gagne l'accès au médium.

Le seuil de taille des agrégats, premier critère permettant de clore un agrégat, est mentionné en figure 3.1 dans la partie contrôle opérateur. Le second critère de clôture d'un agrégat est directement lié à la fonction d'accès au medium (ici HCF EDCA qui est détaillé au chapitre 1). L'accès au medium pourra influencer la taille et la date d'émission des agrégats au travers de certains paramètres comme l'$AIFS$, les bornes de l'intervalle de tirage du *backoff* (CW_{min} et CW_{max}) ou encore la durée de la TXOP. Ils peuvent varier d'une classe de service à l'autre en fonction de la configuration choisie. Cependant comme la norme leur impose des valeurs par défaut, nous avons choisi de ne pas les considérer comme des paramètres de contrôle à la portée du fournisseur d'accès. Ils sont donc fixés aux valeurs définies en tableau 3.2. De plus, comme le tirage du *backoff* est un processus aléatoire, il nous empêche de comparer ces simulations avec les modèles précédemment établis. C'est pourquoi une étude de son influence sera réalisée avec deux états possibles, actif ou inactif (tirage fixé à 0).

À partir de la figure 3.2 trois niveaux de concurrence peuvent être identifiés :

1. entre les paquets stockés dans une classe de service de la couche MAC.
2. entre les classes de service.
3. entre les stations.

La concurrence entre paquets en entrée de la couche MAC est résolue de deux façons, par un ordonnancement effectué par les couches supérieures s'il y a agrégation à la source (couche applicative, système d'exploitation le plus souvent) ou par une sérialisation effectuée par les équipements réseaux précédents sur le chemin. Exemple, si un point d'accès 802.11n effectue l'agrégation de trames Ethernet, le commutateur ou concentrateur aura sérialisé les données (si le point d'accès ne dispose que d'une interface Ethernet). Le schéma fonctionnel en figure 3.3 résume ce paragraphe.

À partir de ces lieux de concurrence, trois scénarios de simulation peuvent être listés :

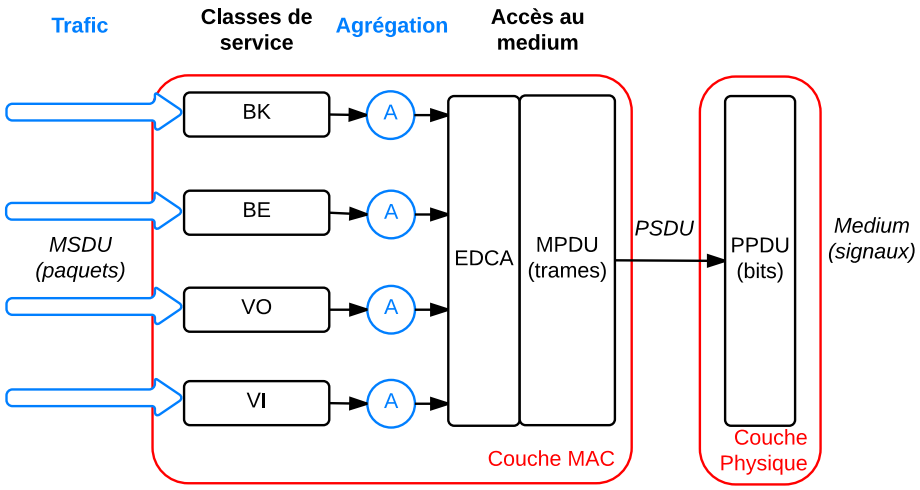


Figure 3.2 – Position de l'agrégation dans la couche MAC

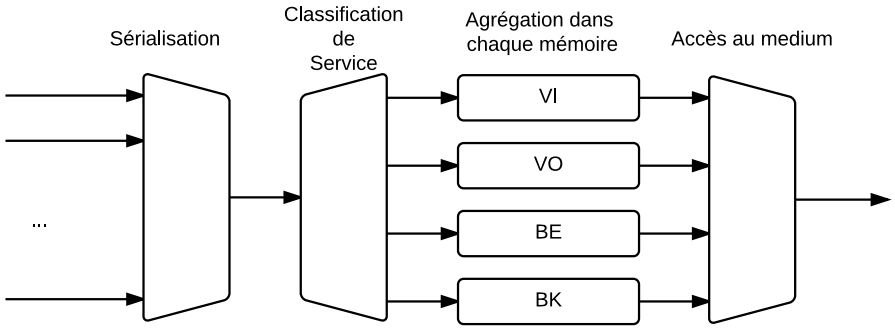
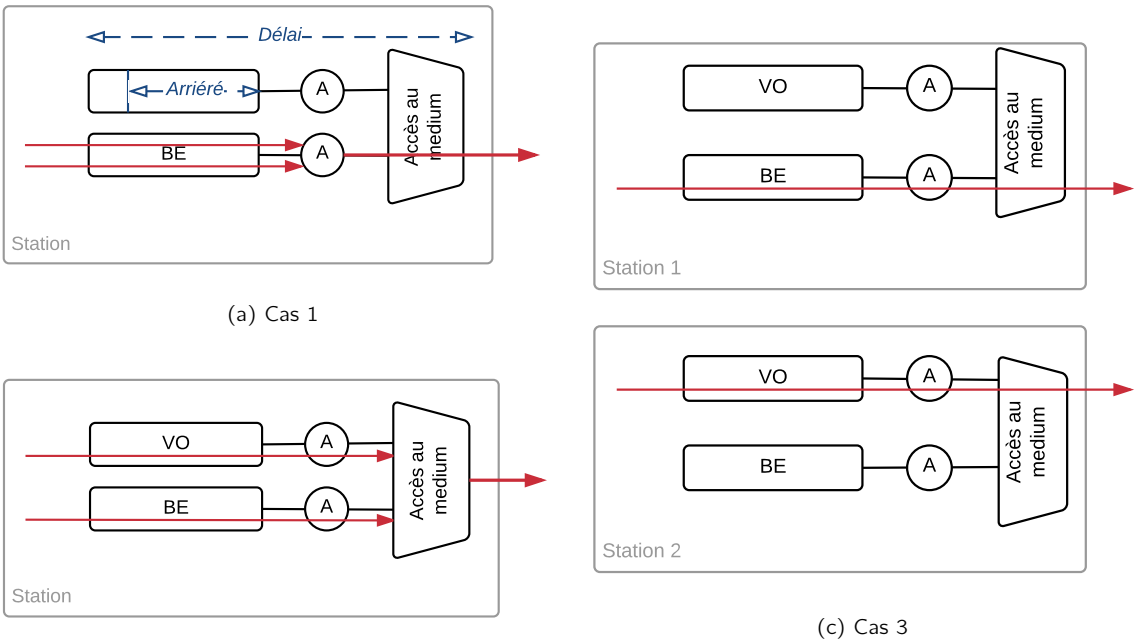


Figure 3.3 – Schéma fonctionnel de l'environnement du système d'agrégation

1. avec deux flux agrégés au sein d'une classe de service (figure 3.4a).
2. avec deux flux agrégés au sein de deux classes de service différentes partageant la même station (figure 3.4b).
3. avec deux flux agrégés au sein de deux classes de service différentes sur deux stations distinctes (figure 3.4c).

Nous considérons seulement deux flux concurrents et uniquement deux classes de service parmi les quatre proposées par la norme (*best-effort* et *voice*, figure 3.4). Les indicateurs de performance comme le délai et l'arriéré de traitement sont représentés en figure 3.4a. Notons que dans la littérature (Le Boudec et Thiran, 2001) montre (Théorème 1.6.4) que sous certaines conditions arriéré et délai sont liés et qu'un majorant de ce dernier peut être obtenu à partir de l'arriéré.

Après avoir décrit le trafic ainsi que la configuration utilisée, nous étudierons chacun des scénarios cités ci-avant. L'objectif est d'utiliser une première configuration simple afin de poser les bases des simulations de ce chapitre.



(a) Cas 1

(b) Cas 2

(c) Cas 3

Figure 3.4 – Les trois scénarios de concurrence considérés

3.1.1 Spécification des flux entrants

Afin d'obtenir des délais significatifs, il est nécessaire de stresser notre système. Ce phénomène est très bien expliqué par (Cruz, 1991) lors de sa proposition de modèles pour des éléments réseau simples. Il explique que les rafales augmentent le délai puisqu'elles augmentent l'arriéré de traitement qui participe au délai. C'est pourquoi nous avons utilisé un profil de trafic basé sur des rafales tout en laissant au système le temps de traiter au moins une partie de cette rafale. Les flux entrant dans le système sont donc spécifiés par un trafic "on/off" constitué d'une phase de rafale de 5 ms suivie d'une phase de pause de 20 ms au cours de laquelle aucun paquet n'entre dans le système.

La figure 3.5 montre un exemple de flux avec la taille d'un paquet en ordonnée (en bits) et son instant d'arrivée en abscisse (en millisecondes). Ce flux est constitué de 100 paquets de 500 octets émis toutes les 50 μs. Comme notre système est la couche MAC de 802.11n cette taille de 500 octets inclut les en-têtes des couches supérieures (UDP et IP).

La phase de rafale (qui est représentée par les points à 4000 bits) est reproduite toutes les 25 *ms* pour un débit moyen de 16 Mb/s ce qui respecte la condition de stabilité d'un tel système puisque le taux d'arrivée moyen est inférieur au taux de service de 65 Mb/s (tableau 3.2). Les tailles de paquets choisies sont identiques à celles considérées par (Skordoulis *et al.*, 2008) qui propose une étude en utilisant lui aussi le simulateur Riverbed Modeler mais en utilisant le modèle développé par Intel à cette époque. Quand à l'inter-arrivée, elle a été obtenue par tâtonnement afin de trouver un point de rupture (juste avant d'observer des pertes de paquets) afin d'atteindre les limites du système. De plus, ce point de rupture nous permet une comparaison avec nos modèles analytiques en particulier celui développé à l'aide du calcul réseau.

À partir de cette valeur de flux, nous avons cherché à rester à débit constant afin de mesurer l'impact

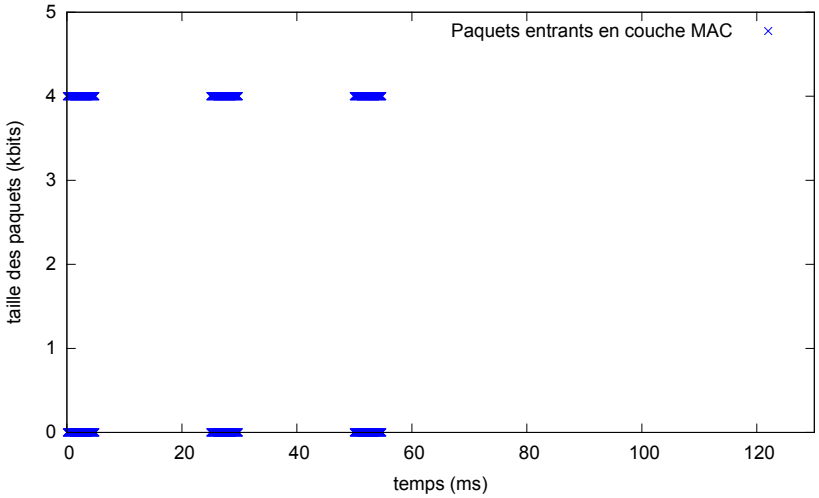


Figure 3.5 – Un exemple de flux entrant dans la couche MAC

distributions	{constante, exponentielle}
taille de paquets (octets)	{125, 250, 500, 1500}
inter-arrivées (μs)	{50, 100}

Tableau 3.1 – Paramètres des flux entrants

des changements de taille et d'inter-arrivée des paquets. Ensuite, les flux restants sont choisis arbitrairement afin de compléter l'étude. Les différentes valeurs utilisées pour l'inter-arrivée et la taille des paquets ainsi que la distribution de ces flux sont indiquées en tableau 3.1.

3.1.2 Configuration des équipements

Au vu des multiples paramètres offerts par les couches MAC et physique de 802.11n nous avons restreint cette étude en fixant d'une part les paramètres physiques (fréquence, largeur de bande, nombre de flux spatiaux, etc.) et d'autre part certains paramètres dits de contrôle opérateur (cf figure 3.1) comme la durée de TXOP ou la taille mémoire.

La topologie utilisée est représentée en figure 3.6 avec une ou deux stations actives et un point d'accès relayant les trames vers un réseau filaire quelconque. Nous nous intéressons uniquement à la partie sans fil de ce réseau, entre les stations et le point d'accès. Cette configuration nous permet d'étudier l'influence de l'agrégation entre plusieurs stations pour l'accès au medium. Même si cette topologie représente un cas peu commun, elle simplifie l'étude et reste comparable à d'autres scénarios plus représentatifs de l'utilisation généralement admise d'un tel réseau sans fil (par exemple : une station en concurrence avec le point d'accès, un point d'accès avec plusieurs flux issus de classes de service différentes). Une station sans fil ne diffère d'un point d'accès que par une valeur potentiellement plus faible de l'AIFS pour les classes *voice* et *video*. De plus ce scénario est particulièrement adapté à une étude dans le domaine des réseaux de capteurs sans-fil (les capteurs transmettant vers leur puits). Même si des protocoles différents sont le plus souvent utilisés, il n'en reste pas moins qu'un mécanisme d'agrégation aurait tout son sens dans ce type de réseau.

La configuration des équipements sans fil est résumée en Tableau 3.2. Cette configuration compte parmi

les plus simple, il n'y a qu'une seule antenne, pas de *beamforming*, des canaux de 20 MHz uniquement et opérant en mode de rétrocompatibilité (HT-mixed). Cette configuration est représentative du minimum qu'un équipement doit fournir pour être certifié 802.11n (excepté pour la partie agrégation qui n'est pas obligatoire à l'émission).

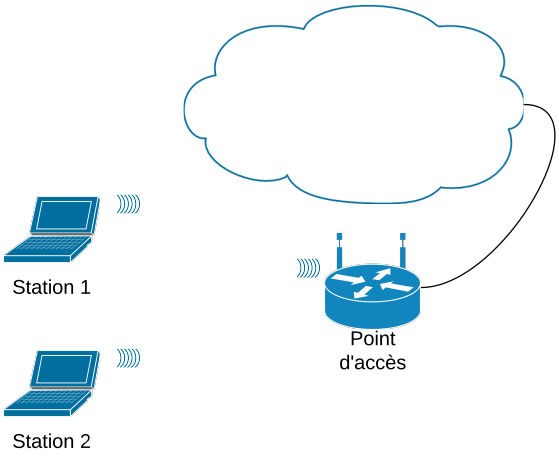


Figure 3.6 – Topologie utilisée pour les simulations Riverbed Modeler

Politique d'accès au medium	HCF EDCA
A-MPDU	desactivé
Acquittement par Bloc	desactivé
Limite de TXOP	une MSDU
Taille mémoire	1024000 bits
SigExt	6 μs
ISM Band	2.4 GHz
Débit de base	65 Mb/s
Débit Maximal	600 Mbps
Flux spatiaux	1
Intervalle de garde	800 ns (regular)
Largeur de bande	20 MHz
Temps Slot	9 μs
SIFS	10 μs
Format de PPDU	HT-mixed
Sensibilité	95 dBm
Propagation	espace libre
Mémoire	partagée

Tableau 3.2 – Configuration des stations sans fil pour les simulations Riverbed Modeler

De tels paramètres permettent de déduire un débit opérationnel de 65 Mb/s à l'aide du tableau 20-30 dans (IEEE-802.11n, 2012).

Les équipements ne sont pas encore complètement configurés et les derniers paramètres constituent les variables de contrôle dont l'influence sera étudiée en détail dans la suite de ce chapitre. Leurs valeurs sont indiquées en tableau 3.3.

Classes de service (AIFSN)	{BE(3), VO(2)}
A-MSDU	{activé, desactivé}
seuil A-MSDU	{3839, 7935}
CW_{min}	{0, $PHYCWmin$ (BE) ou $(PHYCWmin+1)/4-1$ (VO)}
CW_{max}	{0, $PHYCWmax$ (BE) ou $(PHYCWmin+1)/2-1$ (VO)}

Tableau 3.3 – Valeurs des paramètres MAC

La classe de service elle-même et son AIFSN sont des valeurs fixes. Les valeurs de CW_{min} et CW_{max} varient entre 0 et la valeur par défaut pour la classe spécifiée. Les paramètres d'agrégation varient d'une

classe de service à l'autre sans valeurs spécifiques à l'une d'entre elles.

3.2 Étude des cas de concurrence avec un scénario simple

Dans cette section, l'étude commence avec une configuration simple du système et des flux entrant dans celui-ci. D'abord, les trois scénarios de concurrence (figure 3.4) sont étudiés pour un scénario constitué de flux périodiques (de période 50 μs) et à taille de paquets constante (500 octets), suivant le profil "on/off" défini auparavant. Ensuite, une distribution exponentielle de l'inter-arrivée introduira les modifications engendrées par l'utilisation de trafic stochastique. Pour l'étude suivante, le débit du réseau sans fil est configuré à 65 Mb/s (voir le tableau 3.2 pour plus de détails). Le seuil d'agrégation des A-MSDU est fixé au minimum lorsque celle-ci est activée.

3.2.1 Un seul flux dans une classe de service

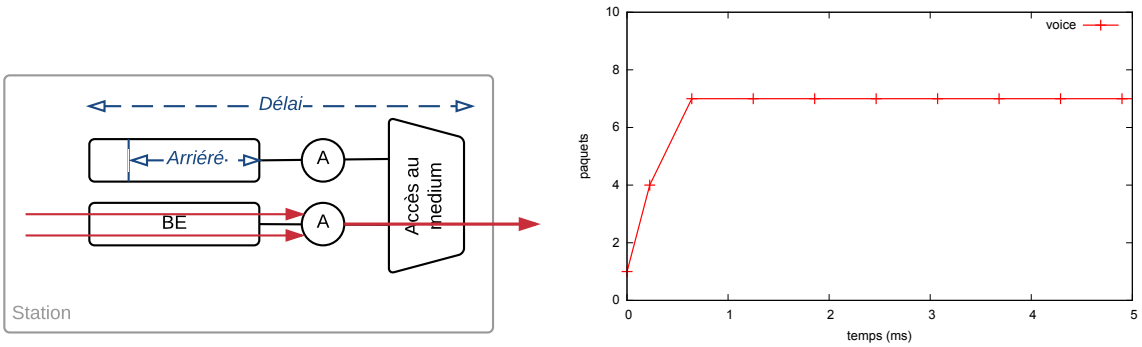
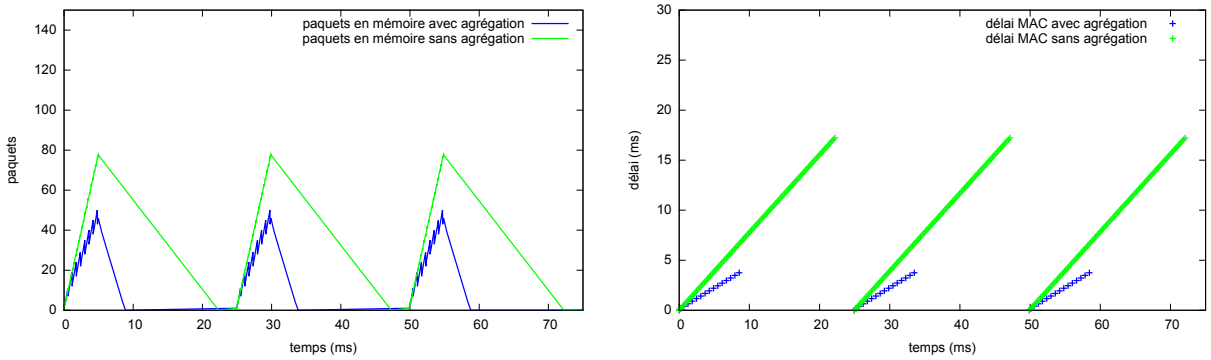


Figure 3.7 – Nombre de paquets par agrégats pour un seul flux dans une seule classe

Ce premier cas reprend les paramètres d'un précédent article (Breck *et al.*, 2014) qui a été décrit au chapitre 2. L'unique différence réside dans le sens des flux qui sont ici de la station vers le point d'accès. On observe l'influence de ce changement en figure 3.7 puisque la taille de l'agrégat passe de 1 à 4 pour atteindre 7 paquets. Dans notre précédent article cette taille évoluait de 1 à 3 avant d'atteindre 7 paquets. Cette différence s'explique par une valeur plus grande pour l'AIFS lorsque l'émetteur est une station plutôt qu'un point d'accès. L'AIFS est le temps d'attente obligatoire avant une transmission après une occupation du medium.

Observons maintenant l'évolution du délai et de l'arriéré de traitement avec et sans agrégation afin d'en mesurer l'impact sur ce scénario simple. Les deux phases du trafic "on/off" sont bien visibles en figure 3.8a. La partie croissante représente la phase de rafale ou le taux d'arrivée est supérieur à la capacité du réseau. On remarque une forme en dents de scie dans laquelle alternent arrivée de paquets et acquittement d'un paquet ou d'un agrégat selon la courbe (ce qui implique le retrait d'un paquet ou d'un agrégat de la file). Lorsque le trafic en entrée du système s'arrête, l'arriéré de traitement va logiquement décroître de façon linéaire (en suivant le débit utile du réseau). Ce tracé permet de mettre en avant les bénéfices de l'agrégation en terme d'occupation maximale de la mémoire (la pointe de la courbe est plus basse avec agrégation) et en terme de débit utile (la pente de la courbe descendante est plus forte avec agrégation). Le débit utile est défini comme le débit offert aux données des couches supérieures du point de vue de la couche MAC (ie,



(a) Arriéré de traitement (b) Délai

Figure 3.8 – Arriéré et délai avec et sans agrégation sur une classe de service (500 octets / 50 μs, *backoff* inactif)

les en-têtes IP et UDP sont incluses dans les données). C'est ici que se situe le gain offert par l'agrégation car lorsque plusieurs paquets partagent les mêmes en-têtes MAC et en particulier physiques, le débit utile qui leur est offert est plus grand. Cela ne modifie en rien la capacité du medium qui dépend des paramètres physiques posés précédemment.

L'agrégation permet ici de réduire l'arriéré de traitement maximal de l'ordre de 36 %. Le temps passé dans la mémoire de la couche MAC est réduit de 78 % en moyenne (mais aussi dans le pire cas) par l'utilisation de l'agrégation comme le montre la figure 3.8b. Ces résultats sont à modérer au regard du profil de trafic très particulier choisi dans cette simulation. C'est le rôle d'une étude plus exhaustive, utilisant d'autres profils de trafics (notamment stochastiques) qui sera menée plus loin dans ce chapitre.

3.2.2 Deux classes différentes sur une seule station

L'étude porte ici sur une seule station mais cette fois deux classes de service différentes sont utilisées, la classe *best-effort* et la classe *voice*. La quantité de mémoire allouée est partagée entre ces deux classes.

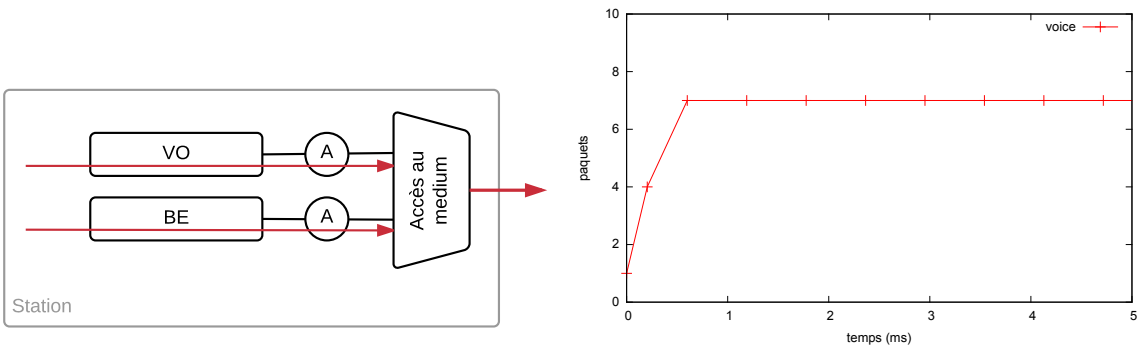


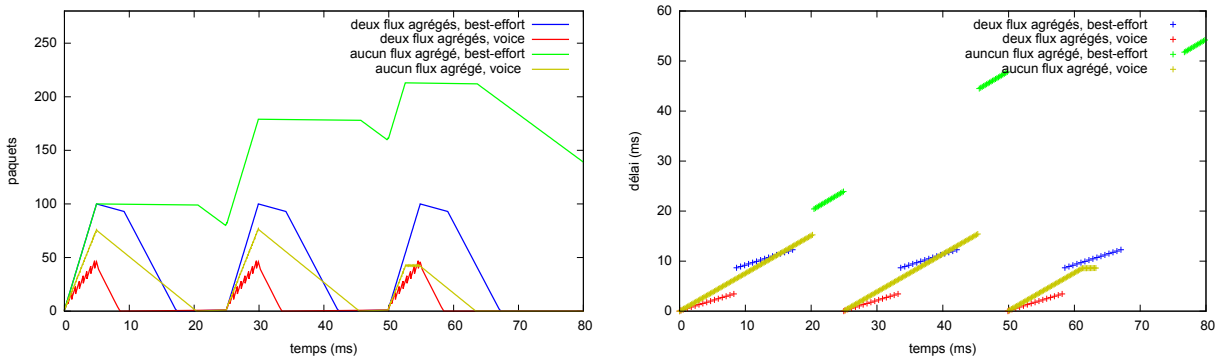
Figure 3.9 – Nombre de paquets par agrégats pour deux classes de service sur une seule station

La figure 3.9 est complètement identique à la figure 3.7. Ce comportement est parfaitement logique puisque le flux de la classe *voice* monopolise le réseau au début de la simulation. Ce résultat est produit par

l'ordonnancement effectué entre les flux par l'AIFS qui est d'une durée inférieure pour le flux *voice* et qui n'est pas contrarié par le tirage du *backoff* que nous avons fixé à 0.

Le délai du flux *best effort* (figure 3.10b) ne cesse d'augmenter alors que la fin de la rafale est atteinte. Ce flux ne sera donc jamais servi si son concurrent n'est pas agrégé puisque dans ce cas cela lui laisse du temps pour transmettre ses paquets. Ce comportement est confirmé par la figure 3.10a sur laquelle l'arriéré de traitement augmente après chaque rafale.

Sur la figure 3.10a on observe bien l'ordonnancement priorité stricte auquel conduit l'ordonnancement évoqué ci-avant qui place le flux *best-effort* dans une situation de famine. Les paquets *best-effort* doivent attendre que tous les paquets *voice* aient été émis avant de pouvoir partir. On remarque également que le nombre de paquets dans la file ne diminue pas suffisamment pour la vider, le système est saturé. Le flux *best-effort* sature la mémoire ce qui influence donc la quantité de mémoire disponible pour le flux *voice*. On remarque lors de la troisième rafale (figure 3.10a) que la courbe de l'arriéré du flux *voice* lorsque aucun flux n'est agrégé ne monte pas à la même hauteur que lors des rafales précédentes. À ce moment, l'arriéré du flux *voice* est limité par la taille mémoire disponible ce qui implique des pertes de paquets.



(a) Arriéré de traitement

(b) Délai

Figure 3.10 – Arriéré et délai avec et sans agrégation sur deux classes de service dans une station (500 octets / 50 μs, *backoff* inactif)

Les figures 3.10a et 3.10b montrent les deux améliorations apportées par l'agrégation et observées dans le cas précédent :

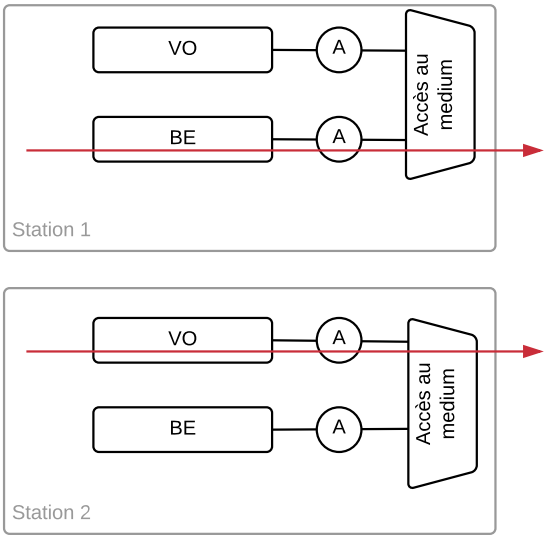
— une diminution de l'arriéré de traitement de 53 % pour le flux *best-effort* et de 39 % pour le flux *voice*.

— une réduction des délais de 82 % (en moyenne et dans le pire cas) pour le flux *best-effort* et de 77 % pour le flux *voice*.

Ces deux améliorations permettent d'éviter des pertes de paquets pour les deux flux.

3.2.3 Une classes différente sur deux stations distinctes

Poursuivons les simulations avec un cas très similaire au précédent puisque les deux même classes de service sont utilisées mais sont ici placées sur des stations distinctes. D'après la norme 802.11e, une classe de service représente virtuellement une station et se comporte comme tel. C'est ce que nous allons vérifier ici.



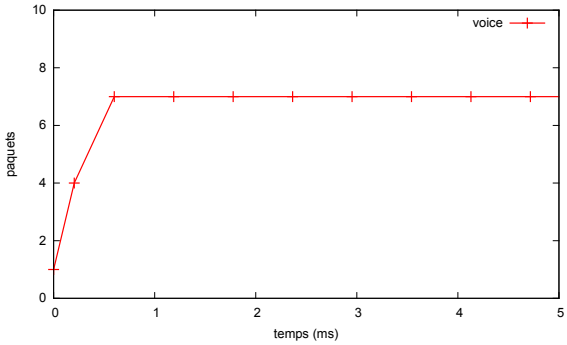
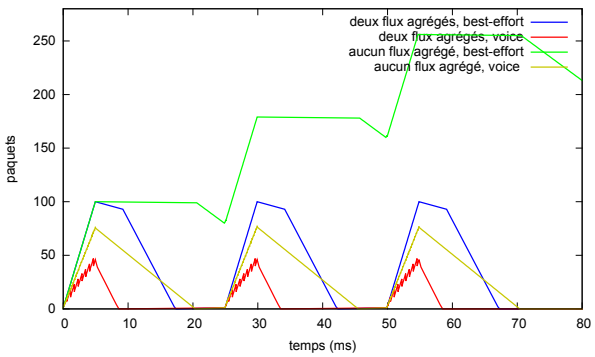
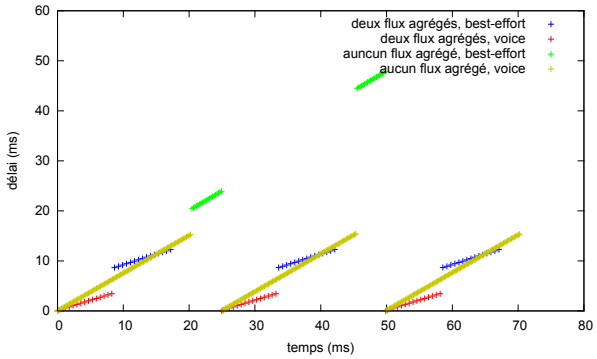


Figure 3.11 – Nombre de paquets par agrégats pour deux classe différentes sur deux stations différentes



(a) Arriéré de traitement



(b) Délai

Figure 3.12 – Arriéré et délai avec et sans agrégation sur deux classes de service dans deux stations distinctes (500 octets / 50 μs, *backoff* inactif)

Les résultats obtenus sont similaires au cas précédent excepté pour la limitation de mémoire. En effet, deux stations signifie deux mémoires et donc pas de mémoire partagée entre les deux flux en concurrence. La troisième rafale ne provoque donc pas de pertes de paquets pour le flux *voice* cette fois et beaucoup moins pour le flux *best-effort*.

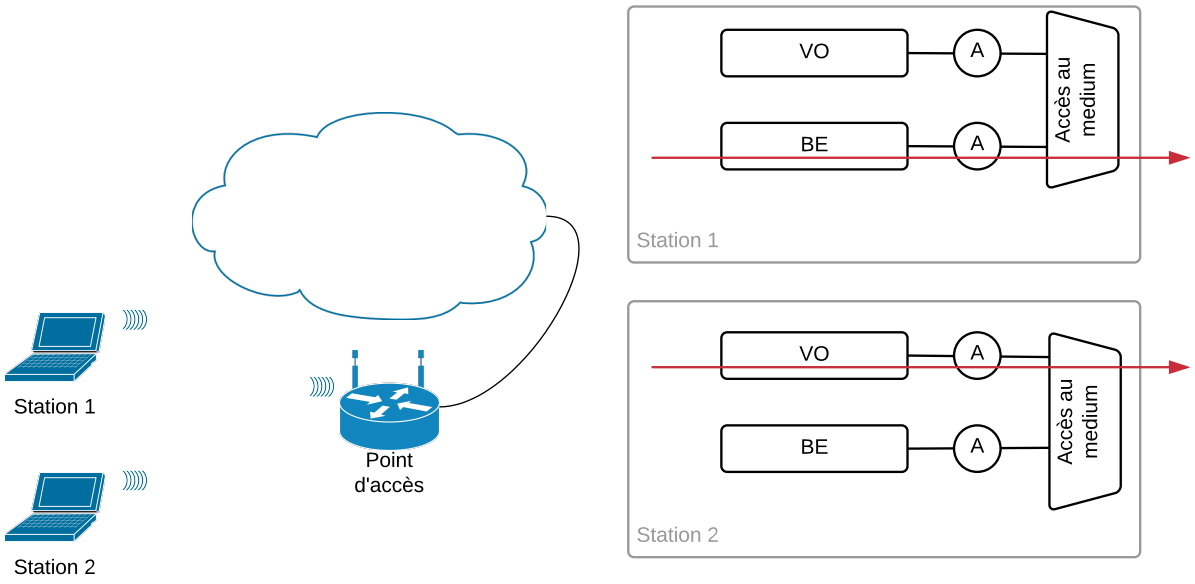
Ce détail mis à part, les résultats confirment qu'une classe de service est une station virtuelle (au sens de 802.11e). Ce résultat nous permettra de considérer des simulations avec des flux sur des stations distinctes. Cela permet d'évaluer la mémoire disponible pour chaque flux indépendamment et rend la comparaison plus facile avec un cas dans lequel les flux sont sur la même station. Sur leur première phase, les simulations utilisant deux flux peuvent être ramenées au comportement d'une simulation avec un seul flux. La classe de service prioritaire monopolisant le medium, l'influence de la classe non prioritaire reste inexistante. Ce résultat sera modéré par l'utilisation du mécanisme du *backoff* dont le but est justement de permettre à une file moins prioritaire d'accéder quelquefois au medium.

3.3 Vers une exhaustivité des scénarios

S'il est impossible d'être exhaustif vu le cadre d'application choisi, nous pouvons présenter un extrait suffisamment important pour être significatif. C'est bien de cela qu'il s'agit, un extrait des milliers de simulations que nous avons réalisées afin de tirer des schémas récurrents, des principes, concernant la dynamique d'une file au niveau MAC dans 802.11n et son contrôle. Une synthèse des résultats obtenus ainsi qu'une discussion autour des observations et recommandations qui en découlent sont présentées en section 3.4.4. La suite de cette section marque le début d'un extrait des milliers de simulations menées dont la lecture sera alourdie par de nombreux détails.

3.3.1 Topologie considérée

Comme évoqué précédemment, nous nous contenterons de simulations utilisant deux classes de service (*best-effort* et *voice*) dans lesquelles un flux transitera (voir la section 3.3.2 pour une définition de ces flux). Chacune de ces classes est placée sur une station distincte qui transmet ses paquets vers un point d'accès (voir figure 3.13).



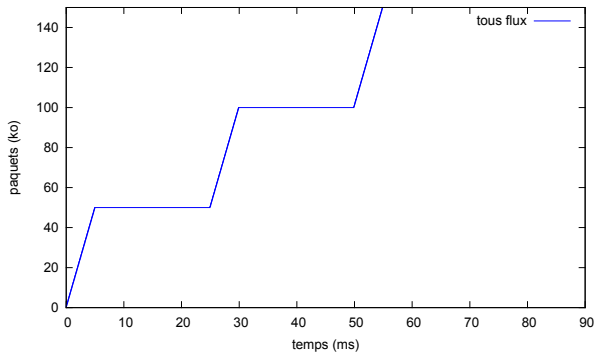
(a) Topologie (b) Représentation fonctionnelle

Figure 3.13 – Rappel de la topologie et de l'affectation des flux considérés

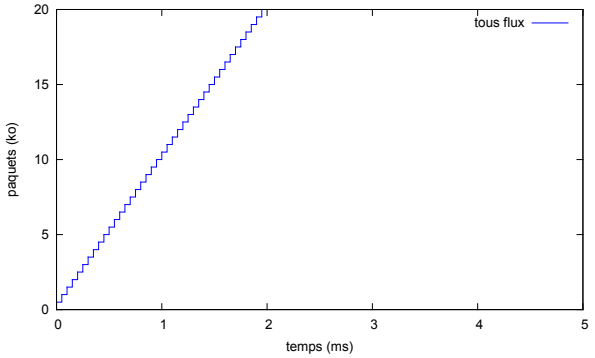
3.3.2 Définition des flux utilisés

Un exemple d'évolution des flux choisis est illustré en figure 3.14. Ils sont constitués de paquets de taille fixe (500 octets) transmis toutes les 50 μs. L'inter-arrivée suit une distribution constante dans un premier temps puis exponentielle de moyenne et d'écart-type 50 μs. Il existe bien sur bien d'autres distributions mais nous nous sommes concentrés sur l'essentiel. Une loi constante permet de prédire le comportement du

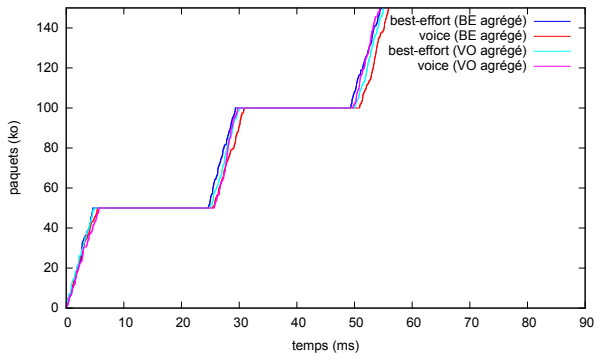
système en éliminant l'inconnu sur le trafic entrant ce qui nous autorise à une comparaison avec le modèle analytique que nous avons développé. Distribuer de façon exponentielle les inter-arrivées des paquets et leurs tailles revient à suivre une loi de Poisson sur l'occurrence de l'arrivée d'un paquet sur un intervalle de temps défini. Cette loi de Poisson est généralement admise comme la plus représentative du trafic sur un réseau.



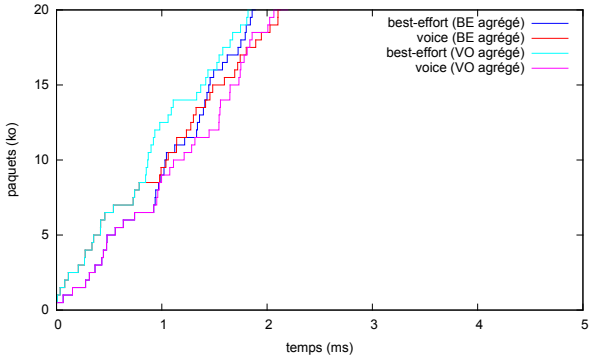
(a) Flux de distribution constante (500 octets / 50 μs)



(b) Agrandissement du début de la rafale de ce flux constant



(c) Flux de distribution exponentielle (500 octets / 50 μs)



(d) Agrandissement du début de la rafale de ce flux exponentiel

Figure 3.14 – Flux utilisés pour l'étude du *backoff*

En comparant les figures 3.14b et 3.14d, la différence entre un flux suivant une distribution constante et l'autre suivant une distribution exponentielle est clairement visible. L'agrandissement effectué en figures 3.14b et 3.14d montre bien la taille et l'inter-arrivée constante puis exponentielle des paquets.

Afin de représenter ces flux nous proposons d'utiliser une notation similaire à celle utilisée par (Le Boudec et Thiran, 2001). Nous considérons donc que chaque flux est représenté par les fonctions définies à l'équation 3.1 pour les flux à distribution constante et à l'équation 3.2 pour ceux à distribution exponentielle.

$$R(t) = k\upsilon_T(t) \tag{3.1}$$
$$\upsilon_T(t) = \left\lceil \frac{t}{T} \right\rceil \text{ pour } t > 0 \text{ et } 0 \text{ sinon.}$$

avec k, la taille moyenne des paquets en octets et T, l'inter-arrivée moyenne des paquets en μs

$$R(t) = ek\upsilon_{eT}(t) \tag{3.2}$$
$$\upsilon_{eT}(t) = \left\lceil \frac{t}{eT} \right\rceil \text{ pour } t > 0 \text{ et } 0 \text{ sinon.}$$
$$\text{avec } ek = \left(\frac{1}{k}e^{\frac{-t}{k}}\right)^+ \text{ et } eT = \left(\frac{1}{T}e^{\frac{-t}{T}}\right)^+$$

Ainsi en suivant l'équation 3.1 on exprimera notre flux comme $500\upsilon_{50}(t)$ si la distribution est constante. Lorsque la distribution est exponentielle, nous sommes censés utiliser l'équation 3.2 ce qui donnerait $e_{500}\upsilon_{e50}(t)$. Or nous avons ici uniquement une distribution exponentielle sur l'inter-arrivée des paquets, il est alors possible de combiner les deux équations pour formaliser le flux ainsi $500\upsilon_{e50}(t)$. Enfin, dans un souci de simplification nous référerons aux flux en omettant la relation avec le temps soit $500\upsilon_{e50}$ dans cet exemple.

3.3.3 Influence du mécanisme de tirage du backoff

La variable principale de ces simulations est l'activation ou non du *backoff*. Si on ajoute la distribution du flux entrant (seconde variable), nous avons alors quatre cas à étudier (qui correspondent aux quatre parties de cette section) :

— *backoff* inhibé et distribution constante
— *backoff* inhibé et distribution exponentielle
— *backoff* actif et distribution constante
— *backoff* actif et distribution exponentielle

Dans chaque cas nous comparerons les résultats entre deux scénarios : la classe *best-effort* est agrégée ou la classe *voice* est agrégée.

Le mécanisme du *backoff* est désactivé en forçant un tirage nul. Ce choix implique que tous les agrégats de la classe prioritaire (*voice*) seront émis avant ceux de la classe non prioritaire (*best effort*). On pourrait vouloir fixer ce *backoff* à d'autres valeurs afin de changer l'ordonnancement priorité stricte produit par la combinaison de l'AIFS et du *backoff* nul. En utilisant le processus de conservation du reste du *backoff*, il reste toutefois théoriquement possible d'arriver à un ordonnancement déterministe différent bien que respectueuse de la procédure de tirage du *backoff* définie dans la norme 802.11n. Néanmoins, cela reste impossible sans modifier le protocole. En effet, si l'on considère la procédure de tirage du *backoff* suivante :

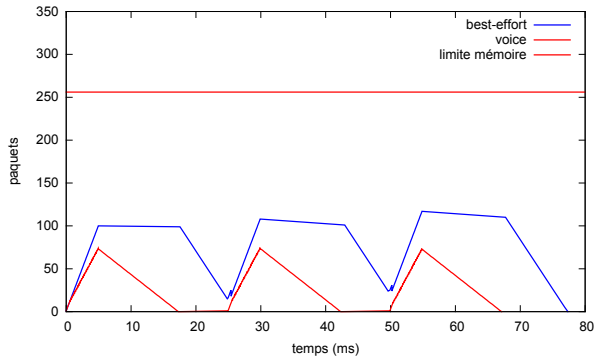
1. medium occupé
2. medium libre
3. AIFS
4. Tirage du *backoff* dans l'intervalle $[0, CW]$
5. Attente pendant le temps tiré

6. — si medium libre → transmettre puis attendre acquittement si positif supprimer la trame de la mémoire et revenir au début, sinon revenir au début pour retransmettre.
   — sinon (medium occupé) → retenir la valeur de *backoff* restante pour la réutiliser et revenir au début.

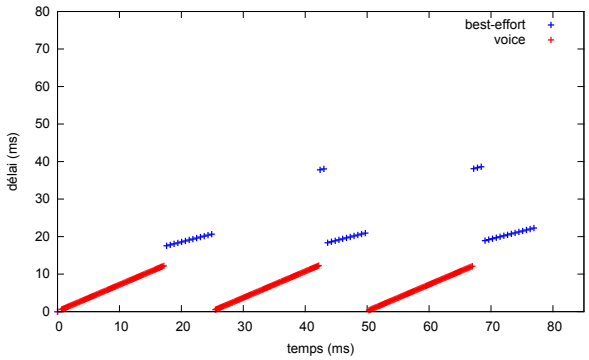
La borne inférieure de l'intervalle de tirage du *backoff* ne peut être modifiée. Sachant par ailleurs que CW est initialisé à CW_{min}, qu'il est doublé à chaque retransmission jusqu'à atteindre CW_{max} et qu'il est réinitialisé à CW_{min} après chaque transmission réussie. L'unique solution permettant d'obtenir un comportement déterministe est donc de fixer CW à 0 et par extension le *backoff*.

3.3.3.a Backoff inhibé et distribution constante de l'inter-arrivée

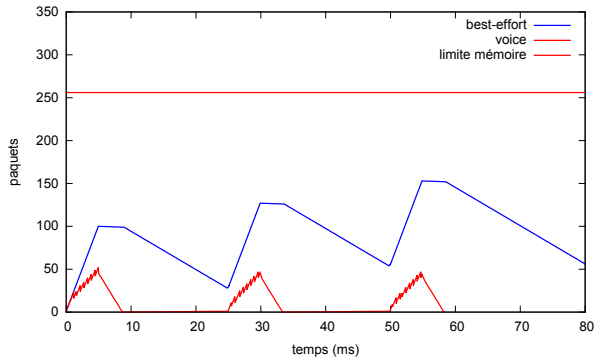
Dans ce premier cas, chaque flux sur chaque classe de service sera agrégé à tour de rôle. Cela nous permet d'ores et déjà d'observer certaines tendances quand à l'influence de l'agrégation sur ces flux.



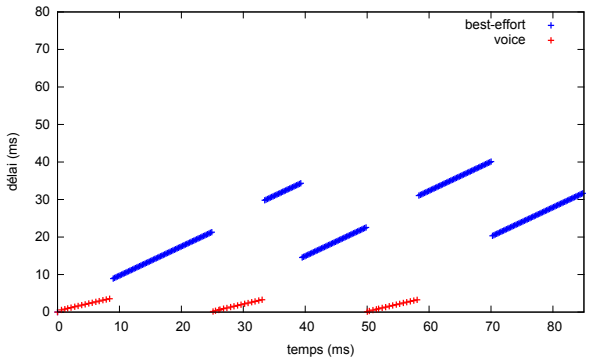
(a) Arriéré de traitement (*best-effort* agrégé)



(b) Délai (*best-effort* agrégé)



(c) Arriéré de traitement (*voice* agrégé)



(d) Délai (*voice* agrégé)

Figure 3.15 – Cas 1 de l'étude du *backoff* (flux constant $500\upsilon_{50}$ et backoff inactif)

L'évolution de l'arriéré de traitement (figures 3.15a et 3.15c) indique que le système sera saturé si les flux continuent de suivre le profil défini. En effet, l'arriéré du flux *best-effort* croît à chaque rafale et la pause n'est pas suffisante pour lui permettre d'être complètement traité. Ce résidu de données à traiter s'accumule et finira par causer le débordement des mémoires.

Quelque soit la file agrégée, elle transmettra ses paquets plus rapidement (le traitement de l'arriéré

est visiblement plus rapide). Une comparaison des figures 3.15a et 3.15c nous révèle les différences entre l'agrégation de la classe *best-effort* et celle de la classe *voice*. Comme le *backoff* est désactivé, tous les paquets du flux *voice* seront transmis avant ceux du flux *best-effort*. Ce comportement associé au profil de trafic utilisé permet au flux *best-effort* de bénéficier de plus de temps pour transmettre ses paquets lorsque son concurrent est agrégé (figure 3.15c). D'autre part, si le flux *best-effort* est agrégé, il transmet un plus grand nombre de paquets dans le temps laissé par le flux *voice*.

Toutes ses observations sont corroborées par les tracés du délai (figures 3.15b et 3.15d). Agréger uniquement le flux prioritaire offre une amélioration significative de ses délais alors que le résultat est plus mitigé pour le flux *best-effort* puisque le délai maximal subi par ses paquets sera sensiblement le même. Seul le nombre de paquets qui subiront des délais élevés sera plus important. En moyenne, les délais sont plus faibles pour le flux *best-effort* lorsqu'il est lui-même agrégé (21 *ms* contre 25 *ms* lorsque le flux *voice* est agrégé). Cependant le délai minimum est beaucoup plus faible pour le flux *best-effort* lorsque son concurrent est agrégé (9 *ms* contre 17 *ms*). Agréger le flux prioritaire exacerbe donc les inégalités entre les deux flux en terme d'arriéré et de délai (et donc d'autant entre les clients). Du point de vue d'un fournisseur d'accès, la question est plus complexe et la réponse dépend de sa politique. S'il souhaite fournir un meilleur service aux abonnés qui sont prêt à payer pour cela qu'il active l'agrégation uniquement sur les classes prioritaires (*voice* et *video* par exemple). Au contraire, s'il souhaite fournir un service certes différentié mais néanmoins plus égalitaire l'agrégation doit être activée uniquement sur les classes les moins prioritaires (*best-effort* et *background* par exemple).

3.3.3.b Backoff inhibé et distribution exponentielle de l'inter-arrivée

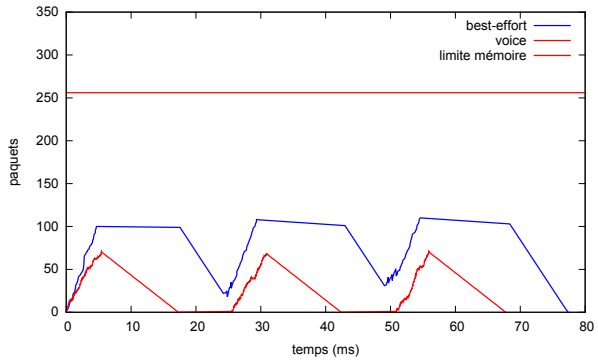
Le tirage du *backoff* est toujours inactif ici mais la distribution du flux change pour devenir exponentielle (flux $500\upsilon_{e50}$).

La modification de la distribution de l'inter-arrivée pour une loi exponentielle de moyenne identique à la constante utilisée précédemment ne modifie pas significativement les valeurs de délai et d'arriéré relevées. Un léger bruit apparaît sur les tracés de l'arriéré en figures 3.16a et 3.16c et cette variation (due au caractère stochastique de l'inter-arrivée) pourrait influer sur le délai subi par certains paquets. Par exemple, au départ de la troisième rafale en figure 3.16a (t=50 *ms*), le flux *best-effort* peut transmettre quelques agrégats supplémentaires car les paquets du flux *voice* arrivent légèrement plus tard. En moyenne (du point de vue fournisseur d'accès) les délais sont identiques (à quelques dizaines de μs près) alors que les gains pour ces paquets spécifiques (point de vue client) sont plus significatifs (de 200 à 700 μs).

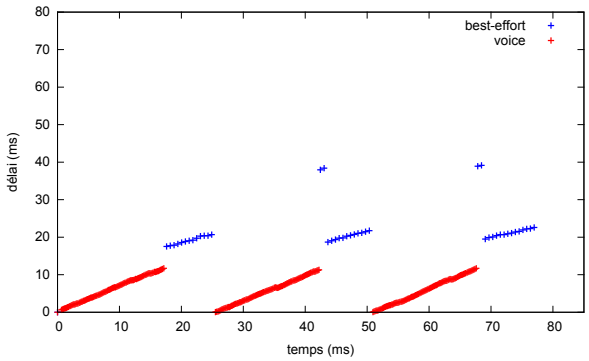
3.3.3.c Backoff actif et distribution constante de l'inter-arrivée

Le *backoff* est maintenant actif et prend ses valeurs dans l'intervalle $[CW_{min}, CW_{max}]$ alors que le flux utilisé est de nouveau constant ($500\upsilon_{50}$).

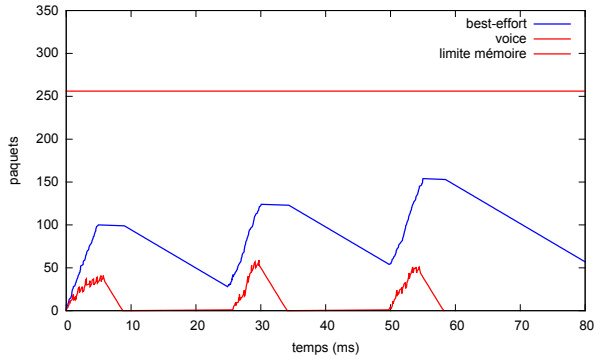
Le tirage du *backoff* influe de deux manières sur les résultats. D'abord, le débit utile est moindre puisque les stations attendent un temps supplémentaire avant de transmettre. Ensuite, le flux *best-effort* bien que moins prioritaire aura ici accès quelquefois au medium alors que le flux *voice* a encore des trames à transmettre. Ce dernier point est mis en valeur par des relevés de délais au milieu d'une phase de transmissions du flux prioritaire (figure 3.17b). En effet, le délai n'est mesuré pour un paquet que lorsque l'acquittement pour ce paquet a été reçu. Il est aisé de croire que le *backoff* tiré par le flux moins prioritaire est inférieur



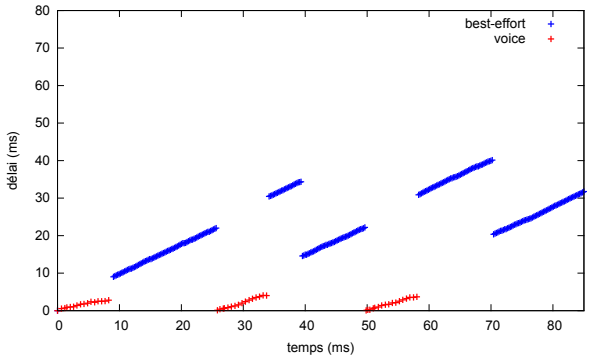
(a) Arriéré (*best-effort* agrégé)



(b) Délais (*best-effort* agrégé)



(c) Arriéré (*voice* agrégé)



(d) Délais MAC (*voice* agrégé)

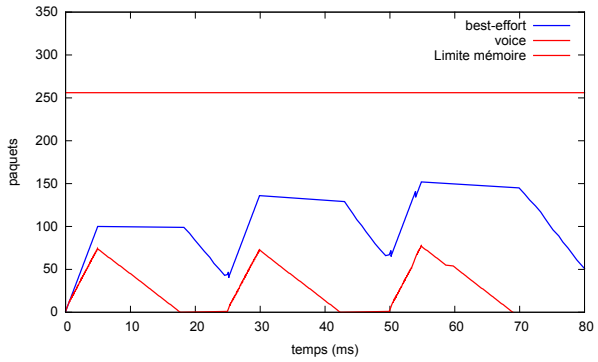
Figure 3.16 – Cas 2 de l'étude du *backoff* (flux exponentiel $500\upsilon_{e50}$ et backoff inactif)

à ce moment mais la figure 3.18 montre que jamais le flux *best-effort* ne bénéficie de cet avantage. Le mécanisme mis en œuvre ici est celui de la conservation du *backoff* (*remaining backoff*).

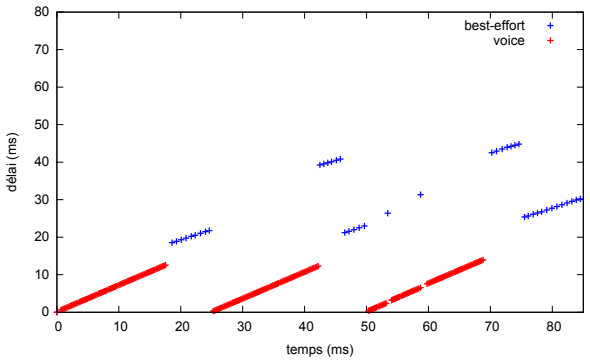
Lorsque le flux *voice* est agrégé, aucun paquet du flux *best-effort* n'a été transmis alors que dans le scénario inverse (*best-effort* agrégé et *voice* non agrégé) on observe des transmissions d'agrégats (figures 3.17a et 3.17b au cours de la troisième rafale entre t=50 et t=60 ms). Cet écart comportemental est expliqué par une combinaison de plusieurs facteurs :

— le flux *best-effort* tire des valeurs de *backoff* beaucoup plus importantes que son concurrent *voice* (comme le montre la figure 3.18)
— le flux *voice* agrégé transmet ses paquets sur une durée plus faible (grâce à l'augmentation de débit utile due à la réduction des en-têtes apportée par l'agrégation)
— le principe même de l'agrégation réduit le nombre de transmission et donc d'autant le nombre de *backoff* tirés ce qui diminue la vitesse de décrémentation *backoff* conservé par le flux moins prioritaire.

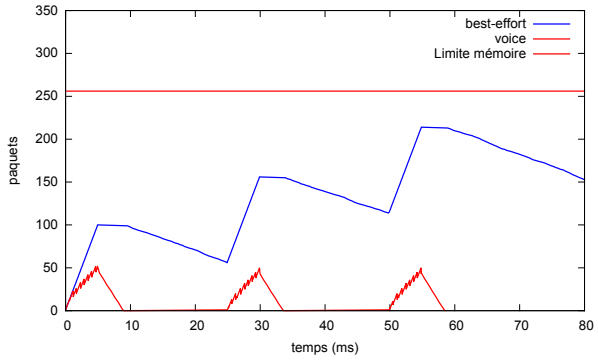
Lorsque le mécanisme du *backoff* est activé, être en concurrence avec un flux prioritaire et agrégé réduit les chances pour le flux *best-effort* de remporter l'accès au medium. À l'inverse, être en concurrence avec un flux moins prioritaire agrégé peut augmenter significativement le délai subi par certains paquets.



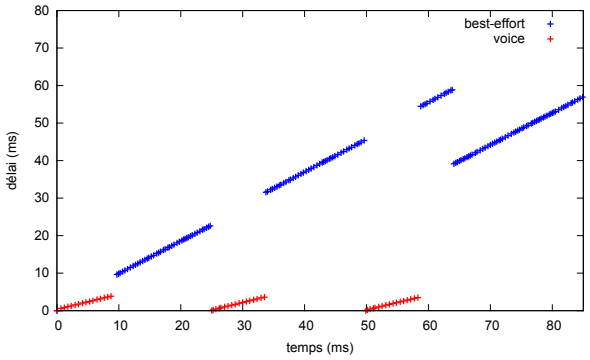
(a) Arriéré de traitement (*best-effort* agrégé)



(b) Délai (*best-effort* agrégé)



(c) Arriéré de traitement (*voice* agrégé)



(d) Délai (*voice* agrégé)

Figure 3.17 – Cas 3 de l'étude du *backoff* (flux constant $500v_{50}$ et backoff actif)

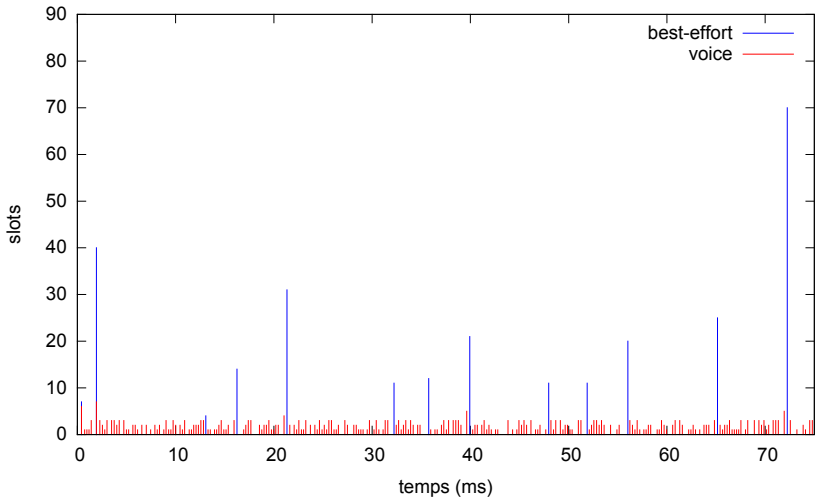
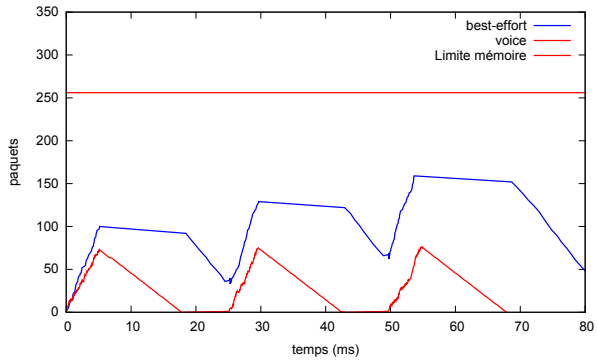


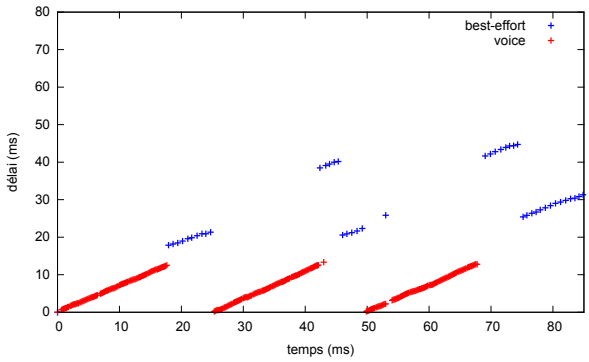
Figure 3.18 – Backoffs tirés en nombre de slots

3.3.3.d Backoff actif et distribution exponentielle de l'inter-arrivée

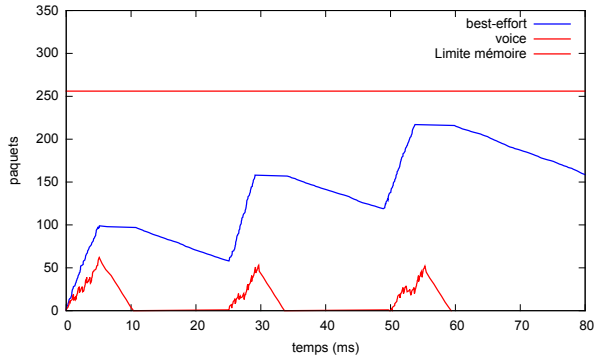
Voici le dernier cas considéré avec distribution exponentielle des flux entrants ($500v_{e50}$) et le tirage du *backoff* actif. En figure 3.19b on observe à nouveau que le flux *best-effort* profite de la conservation de son *backoff* pour transmettre quelques agrégats lors des phases de transmission du flux *voice*.



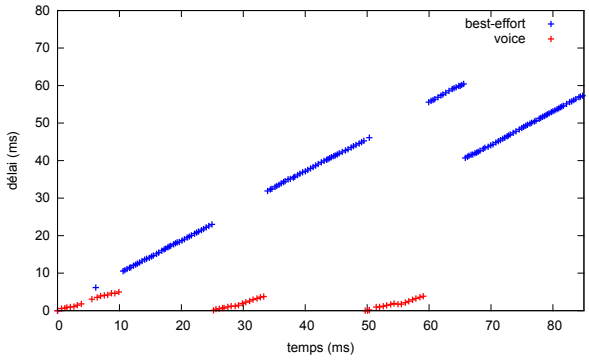
(a) Arriéré de traitement (*best-effort* agrégé)



(b) Délai (*best-effort* agrégé)



(c) Arriéré de traitement (*voice* agrégé)



(d) Délai (*voice* agrégé)

Figure 3.19 – Cas 4 de l'étude du *backoff* (flux exponentiel $500v_{e50}$ et backoff actif)

L'utilisation d'une distribution exponentielle au lieu de la distribution constante précédemment utilisée ne modifie en rien le comportement du système. Les valeurs de l'arriéré ou des délais restent très similaires puisque les variations enregistrées évoluent de -3,7% à 3,3% en moyenne. De telles variation sont logiques sachant que les mesures sont basées sur un seul tirage de la distribution exponentielle ce qui nous empêche de les généraliser pour l'instant. Nous nous attacherons à multiplier le nombre de tirages lors d'une étude de l'influence du système d'agrégation en section 3.4.

Voici les observations que nous pouvons faire pour l'instant concernant le mécanisme du *backoff* :

— il permet aux flux de moindre priorité d'avoir quelquefois l'accès au medium
— cela entraîne une augmentation des délais subis par certains paquets de plus haute priorité

L'influence du mécanisme de tirage du *backoff* se limite aux deux affirmations ci-dessus et à une réduction du débit utile (par l'ajout d'un temps d'attente avant l'émission d'une trame). Comme ce mécanisme participe à rendre la concurrence plus équitable nous avons choisi de l'activer tout au long de l'étude d'influence du système d'agrégation (section 3.4) et de ne pas le considérer comme paramètre de contrôle modifiable par un fournisseur d'accès ou même par un client.

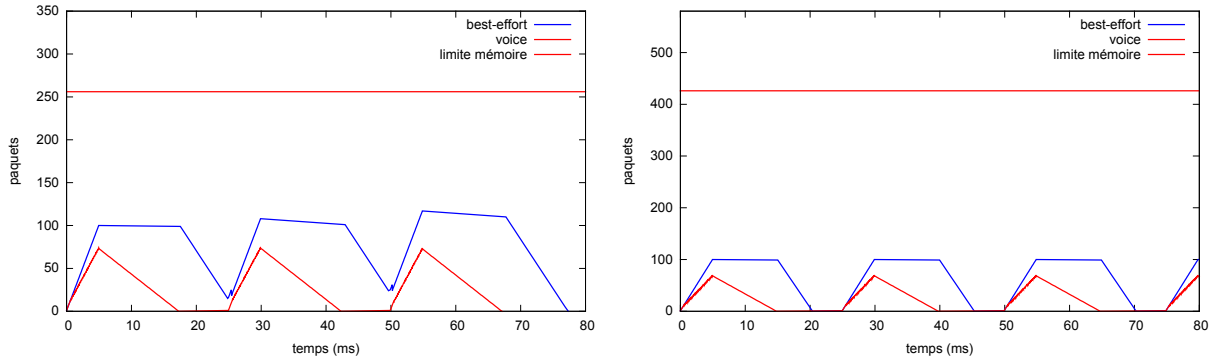
3.3.4 Influence de la taille des paquets

Dans la section précédente, nous nous sommes intéressés aux différences entre des lois constante et exponentielle pour l'inter-arrivée des paquets, à la présence ou non du mécanisme de tirage du *backoff* ainsi qu'à l'agrégation d'une classe de service où de l'autre. Intéressons nous maintenant à l'influence d'un changement de taille des paquets. D'abord à l'aide de flux périodiques à taille de paquets constante, l'étude se poursuivra avec des flux totalement stochastiques.

La topologie, les classes utilisées ainsi que leur répartition sur deux stations ne changent pas pour cette étude. Nous nous intéressons uniquement aux variations des flux entrants.

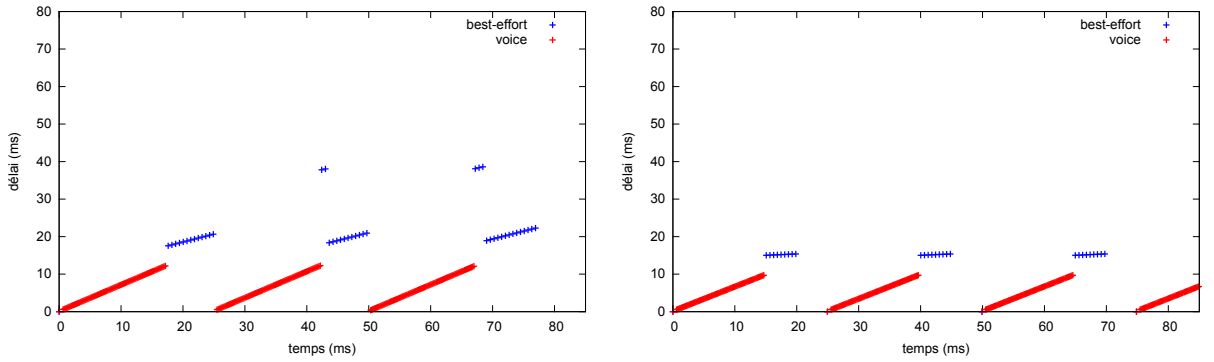
3.3.4.a Après une légère réduction de la taille des paquets (flux constants)

À flux constant, une comparaison est réalisée entre des flux constitués de paquets de 500 et de 300 octets émis toutes les 50 μs ($500\upsilon_{50}$ et $300\upsilon_{50}$). Le mécanisme de tirage du *backoff* reste pour l'instant désactivé.



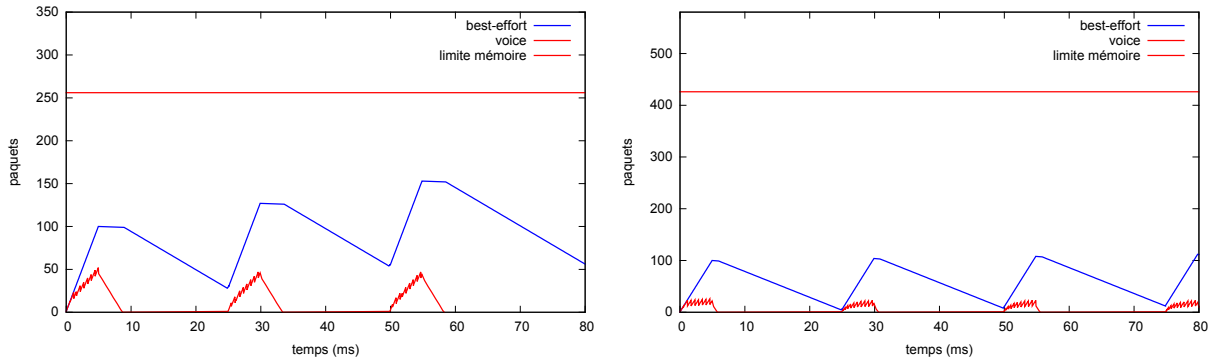
(a) Flux $500\upsilon_{50}$ (b) Flux $300\upsilon_{50}$

Figure 3.20 – Évolution de l'arriéré suite à une diminution de la taille des paquets (*best-effort* agrégé)



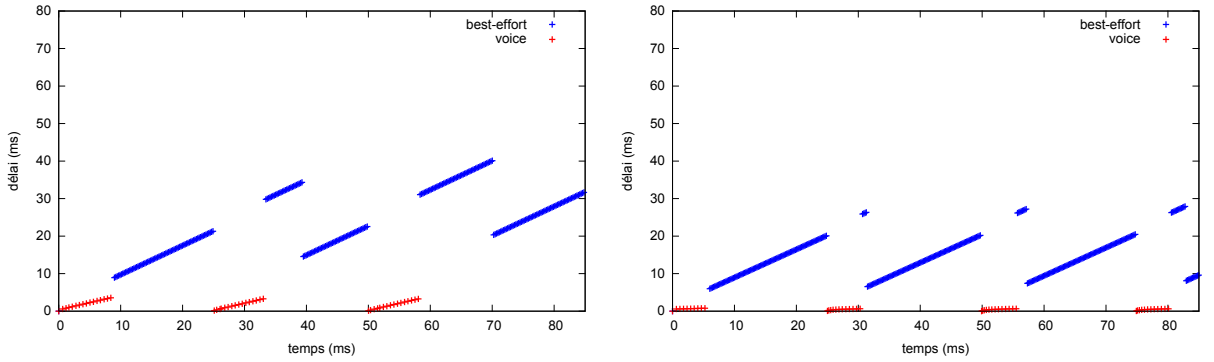
(a) Flux $500\upsilon_{50}$ (b) Flux $300\upsilon_{50}$

Figure 3.21 – Évolution du délai suite à une diminution de la taille des paquets (*best-effort* agrégé)



(a) Flux 500υ_{50} (b) Flux 300υ_{50}

Figure 3.22 – Évolution de l'arriéré suite à une diminution de la taille des paquets (*voice* agrégé)



(a) Flux 500υ_{50} (b) Flux 300υ_{50}

Figure 3.23 – Évolution du délai suite à une diminution de la taille des paquets (*voice* agrégé)

Une réduction de la taille des paquets réduit le temps nécessaire à leur transmission. Ainsi, sur les figures 3.20b et 3.22b l'évolution de l'arriéré montre que tous les paquets sont transmis entre chaque rafale contrairement au cas utilisant des paquets de 500 octets illustré en figures 3.20a et 3.22a. Le système n'est plus saturé. Les délais sont eux aussi fortement réduits comme en attestent la comparaison entre les figures 3.21a et 3.21b d'une part et 3.23a et 3.23b d'autre part.

Ces simulations montrent également que l'agrégation continue d'opérer malgré la diminution de la taille des paquets et donc de la bande passante consommée (cependant, si la taille devient trop faible, il n'y a plus d'agrégation). Le système d'agrégation n'a donc besoin que d'arriéré de traitement, d'être congestionné pour s'activer.

3.3.4.b Après une forte augmentation de la taille des paquets (flux stochastiques)

Lors de l'étude d'influence du mécanisme du *backoff*, l'utilisation d'une distribution exponentielle sur l'inter-arrivée des paquets présentait quelques changements comportementaux intéressants. Ce type de flux permet d'étendre les observations effectuées lorsque les flux ne sont pas maîtrisés (contrairement à un flux périodique avec des paquets de taille constante). Cependant, afin d'améliorer encore la crédibilité des résultats il est nécessaire d'effectuer plusieurs tirages d'une distribution stochastique.

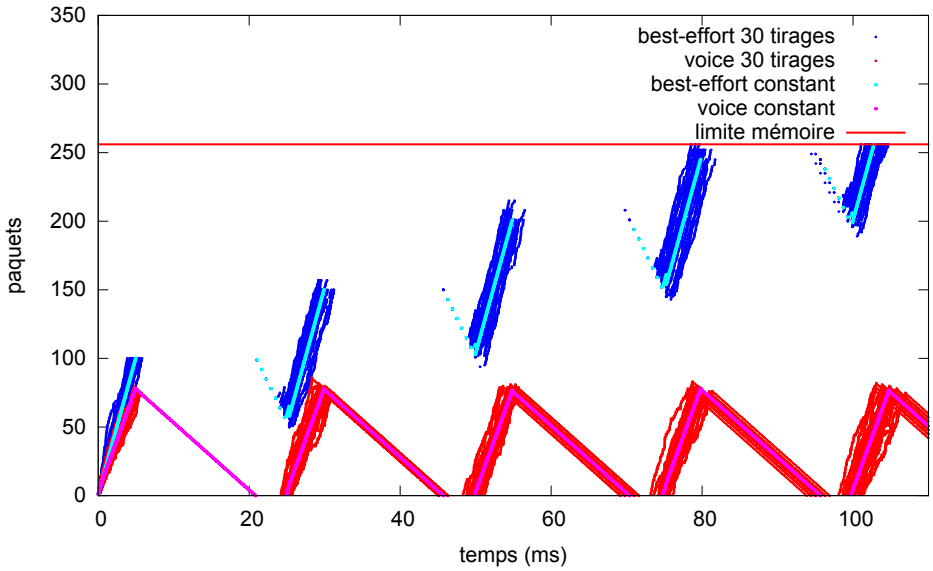


Figure 3.24 – Comparaison arriéré avec inter-arrivée constante et exponentielle sur 30 tirages

Nous avons choisi d'utiliser 30 tirages de cette loi. Les résultats de ces différents tirages sont comparés avec un tirage constant en figure 3.24 sur laquelle sont tracés les arriérés des deux flux considérés lorsqu'ils suivent une distribution constante et exponentielle. Même si l'évolution moyenne est similaire, les variations autour de cette moyenne expliquent les résultats différents obtenus auparavant (qui peuvent être favorables ou défavorables ici). Le service est, pour sa part, effectué au même taux quelque soit le profil du trafic entrant (pentes identiques). À la cinquième phase la limite mémoire est atteinte par le flux agrégé (qui est non prioritaire).

En vue d'une introduction de flux totalement aléatoires dans le système, il est nécessaire d'appliquer une loi exponentielle sur la taille des paquets entrants. C'est pourquoi chaque file reçoit un flux aux caractéristiques suivantes :

— la taille des paquets suit une loi exponentielle de moyenne et d'écart-type 500 octets
— l'inter-arrivée suit une loi exponentielle de moyenne et d'écart-type 50 μs

Les résultats obtenus avec ces paramètres de flux servent de référence. Ils sont ensuite comparés avec des flux de taille et d'inter-arrivée variable.

Les constantes de simulation (tableau 3.2) ne changent pas et les variables prennent les valeurs précisées en tableau 3.4. Contrairement à la section précédente, le mécanisme du *backoff* est actif et prend ses valeurs dans l'intervalle spécifié. Dans un premier temps, nous avons fixé le seuil de taille des agrégats à 3839 octets mais pour observer un effet de l'agrégation nous considérons les quatre cas suivants :

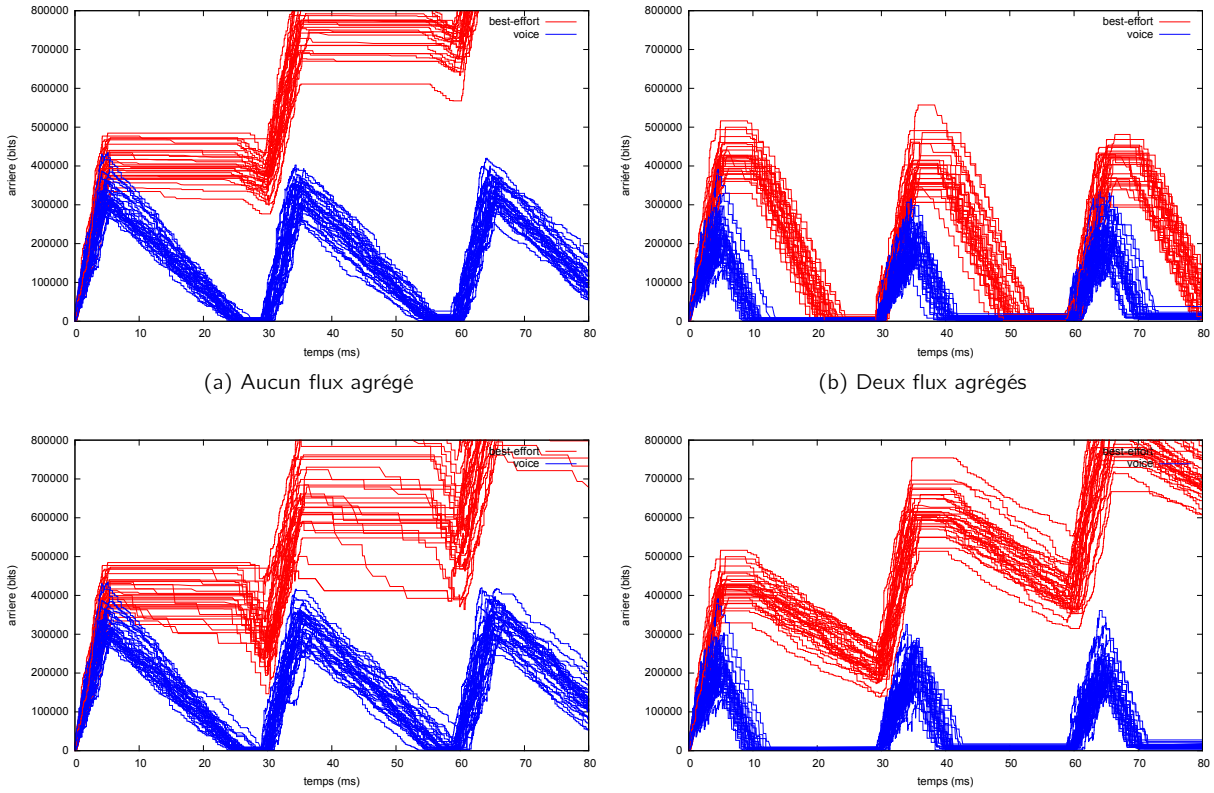
— deux flux agrégés
— seul le flux *best-effort* est agrégé
— seul le flux *voice* est agrégé
— aucun flux n'est agrégé

L'arriéré de traitement évolue comme illustré en figure 3.25. De multiples courbes sont tracées puisqu'il y a trente tirages de la loi exponentielle qui régit l'inter-arrivée et la taille des paquets.

Le système est capable d'absorber la totalité des paquets du flux de classe *voice* quels que soient les flux

seuil A-MSDU	{3839}
best-effort : CW_{min}, CW_{max}	$PHYCWmin$, $PHYCWmax$
voice : CW_{min}, CW_{max}	$(PHYCWmin+1)/4-1$, $(PHYCWmin+1)/2-1$

Tableau 3.4 – Paramètres MAC



(a) Aucun flux agrégé

(b) Deux flux agrégés

(c) Flux *best-effort* agrégé

(d) Flux *voice* agrégé

Figure 3.25 – Arriéré de traitement avec un flux de 500 octets / 50 μs

agrégés. L'arriéré du flux de classe *best-effort* augmente à chaque rafale dans tous les cas, excepté lorsque les deux flux sont agrégés. Cette évolution conduit à une saturation de la mémoire du système puis à des débordements et donc des pertes de paquets. Ces pertes seront un indicateur des performances du système utilisé dans la suite de cette étude.

Une comparaison des figures 3.25c et 3.25d révèle que l'arriéré maximal du flux *best-effort* est inférieur lorsque le flux *voice* est agrégé (figure 3.25d). Ce résultat est contre-intuitif : face à une telle situation on souhaiterait agréger le flux subissant des pertes pour améliorer ses performances alors qu'il est préférable ici d'agréger son concurrent. Sachant que la meilleure combinaison reste d'agréger les deux.

D'autre part, en comparant les figures 3.25a et 3.25c on notera que le fait d'agréger le flux *best-effort* semble augmenter légèrement la dispersion temporelle du flux *voice*. Ces deux observations seront discutées dans la section 3.4 qui étudie l'influence de l'agrégation en détail. L'augmentation de la variation d'arriéré entre les différents tirages est d'autant plus marquée pour le flux *best-effort*. En effet, les bénéfices apportés par l'agrégation sont contraint par l'accès au medium, agréger le flux *best-effort* permet de réduire son

arriéré de traitement à condition que les agrégats soient émis. Ce qui produit cette dispersion est donc le mécanisme de tirage du *backoff* qui peut être plus ou moins favorable selon les tirages mais également la "chance" vis à vis du tirage des instants d'arrivée des paquets du flux *voice* concurrent. Ici, dans les cas les plus favorables, les paquets *voice* peuvent être arrivés légèrement plus tard au début de la seconde rafale et le système transmet des paquets *best-effort* pendant cette pause.

À partir de notre flux de référence, nous souhaitons maintenant augmenter la taille des paquets de manière radicale (afin de pousser le système dans ses derniers retranchements) tout en différenciant cette taille pour les deux flux. En multipliant la taille des paquets pour un seul flux par trois nous atteignons une charge en entrée du système de l'ordre de 64 Mb/s ce qui est très proche de la capacité du médium (65 Mb/s). Les deux flux définis dans cette section suivront donc les paramètres suivants :

— un flux suivant une loi exponentielle de 500 octets toutes les 50 μs (en moyenne)
— un flux suivant une loi exponentielle de 1500 octets toutes les 50 μs(en moyenne)

Avec quatre combinaisons pour l'agrégation et deux pour les flux entrants, huit cas différents sont considérés ici :

— Aucun flux n'est agrégé
  — *best-effort* 1500 octets / 50 μs
  — *voice* 1500 octets / 50 μs
— Le flux *best-effort* est agrégé
  — avec des paquets de 500 octets / 50 μs
  — avec des paquets de 1500 octets / 50 μs
— Le flux *voice* est agrégé
  — avec des paquets de 500 octets / 50 μs
  — avec des paquets de 1500 octets / 50 μs
— Les deux flux sont agrégés
  — *best-effort* 1500 octets / 50 μs
  — *voice* 1500 octets / 50 μs

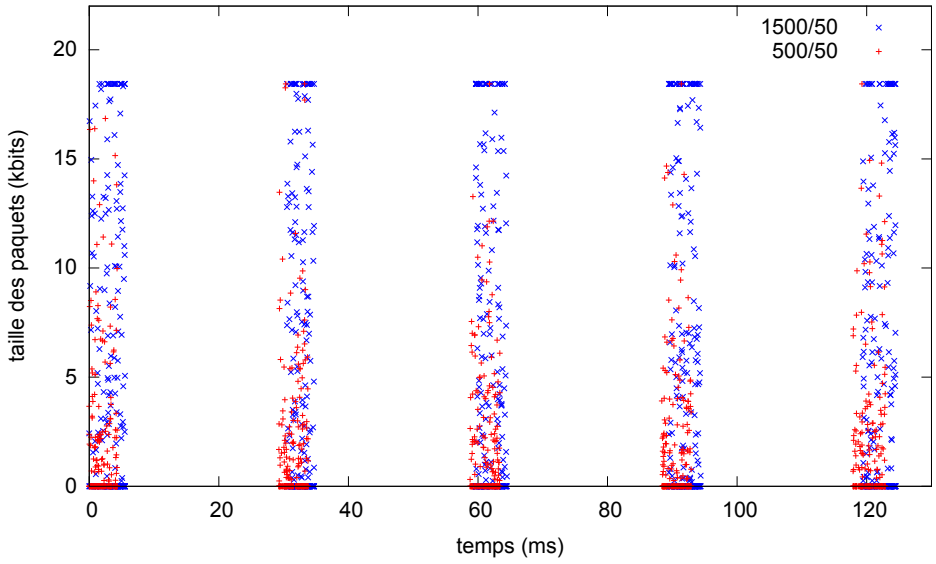


Figure 3.26 – Trafic entrant

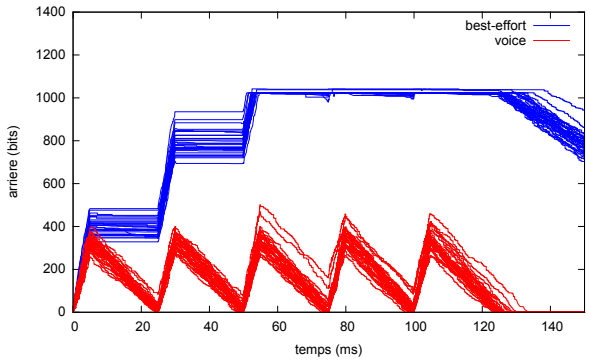
Le tracé du trafic entrant dans le système (figure 3.26) montre la distribution aléatoire des paquets. Dans ces variations, on remarque un effet de seuil sur la taille des paquets suivant la distribution de moyenne 1500 octets. Ce seuil correspond à la taille maximale d'une MSDU, 2304 octets soit 18342 bits. Ce seuil sera souvent atteint puisque la moyenne de la distribution est suffisamment proche de cette valeur seuil. Les segments transmis au niveau des couches supérieures sont pourtant potentiellement de taille supérieure ce qui conduit ici à des rafales de fragments créés par la couche 3 (ici IP). Invisibles sur cette figure, ils sont mis en évidence par la présence de plusieurs MSDU de tailles maximales suivies d'une MSDU de taille inférieure qui arrivent toutes au même instant (sur les fichiers sources de cette figure).

Figure 3.27, l'arriéré est tracé en colonne de gauche lorsque aucun flux n'est agrégé et sur la droite lorsque les deux le sont. Figure 3.28, l'arriéré est tracé en colonne de gauche lorsque la classe *best-effort* est agrégée et sur la droite lorsque la classe *voice* l'est. Chaque ligne sur ces deux figures reprend une des trois combinaisons de flux définies ($e500\upsilon_{e50}$ et $e1500\upsilon_{e50}$ pour le flux *best-effort* puis *voice*). Les différentes évolutions de l'arriéré de traitement représenté en figures 3.27 à 3.28 montrent bien que l'augmentation de la taille des paquets même pour un seul flux surpasse les capacités du système (en particulier sur les figures 3.27e et 3.27c où aucun flux n'est agrégé).

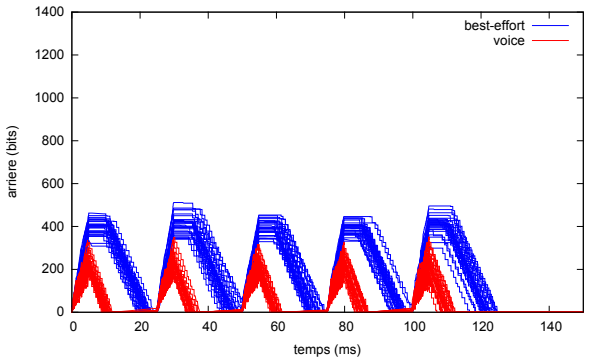
Sur cette première figure on remarque la réduction importante de l'arriéré offerte par l'agrégation lorsque les deux flux sont identiques (figure 3.27b et 3.27a). Agréger les deux flux lorsque le flux *voice* est constitué de paquets de 1500 octets (en moyenne) permet d'améliorer ses propres performances seulement puisque le temps laissé au flux *best-effort* est négligeable, l'arriéré du flux *best-effort* s'envole et provoque rapidement une saturation de la mémoire (figure 3.27c et 3.27d). Notons que si les deux flux partageaient la même mémoire, le flux *voice* aurait été également impacté. Ce n'est pas le cas ici et on observe que l'arriéré est presque entièrement traité pour chaque tirage et à chaque rafale. Ensuite, lorsque le flux *best-effort* est composé de paquets de 1500 octets à son tour, les deux flux réussissent à réduire significativement leur arriéré (figure 3.27e et 3.27f) ce qui est impossible pour le flux *best-effort* lorsque aucun flux n'est agrégé car le flux *voice* occupait la totalité du temps entre deux rafales. Le flux *best-effort* subira néanmoins des pertes contrairement au flux *voice*.

Maintenant que les deux cas les plus extrêmes ont été comparés (aucun flux agrégé et deux flux agrégés), intéressons nous aux deux cas intermédiaires soit lorsqu'un seul flux est agrégé. Lorsque la taille des paquets du flux *voice* est de 1500 octets et qu'il n'est pas agrégé (figures 3.28c et 3.27c), le changement est minime pour le flux *best-effort* puisqu'il était déjà dans une situation proche de la famine (figure 3.28a). Le changement est évidemment radical pour le flux *voice* qui va subir des pertes importantes. La figure 3.28e met en évidence une saturation extrêmement rapide de la mémoire par le flux *best-effort* alors même que celui-ci est agrégé. Cependant aucune influence n'est relevée sur le flux *voice*.

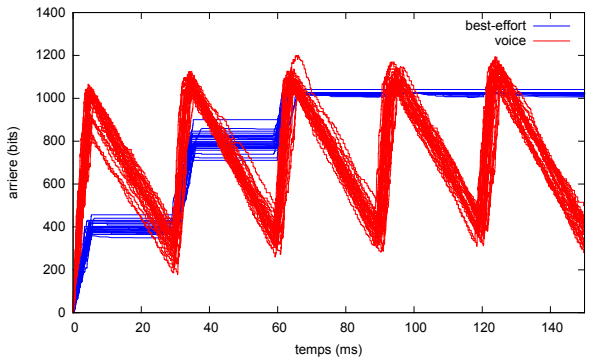
C'est lorsque son concurrent de classe *voice* transmet des agrégats de paquets de plus faible taille (500 octets) que le flux *best-effort* transmet un nombre significatif de paquets (figure 3.28f). Cette figure comparée à la figure 3.28b montre que la quantité de données transmise est plus grande lorsque le flux *best-effort* reçoit des paquets de taille plus importante. Dans ce dernier cas et une fois la limite mémoire atteinte, l'arriéré de traitement est largement inférieur (après transmission des trames). Ce mécanisme peut s'apparenter à une agrégation des données à la source qui, réduisant la proportion d'en-têtes en augmentant la taille des paquets, augmente le débit utile du système. Agréger le flux *voice* lui permet de réduire le nombre de paquets perdus (figures 3.28c et 3.28d) ce qui est visible par un arriéré inférieur voire nul sur la seconde figure.



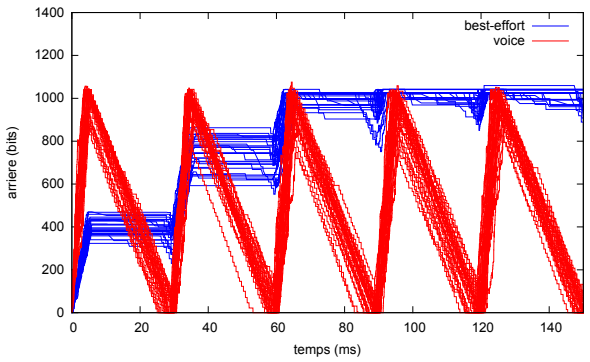
(a) Flux $e500v_{e50}$ (aucun flux agrégé)



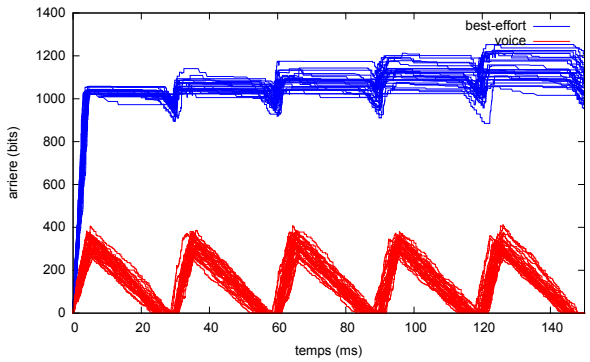
(b) Flux $e500v_{e50}$ (deux flux agrégés)



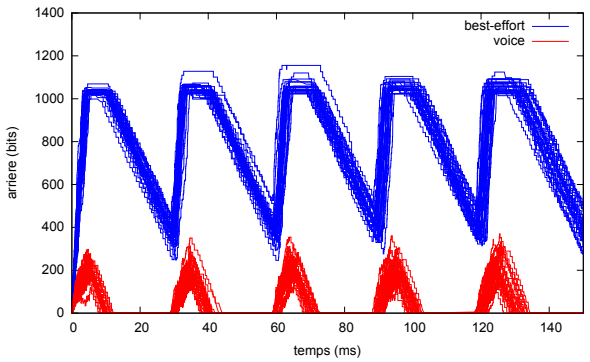
(c) Flux *voice* $e1500v_{e50}$ (aucun flux agrégé)



(d) Flux *voice* $e1500v_{e50}$ (deux flux agrégés)



(e) Flux *best-effort* $e1500v_{e50}$ (aucun flux agrégé)



(f) Flux *best-effort* $e1500v_{e50}$ (deux flux agrégés)

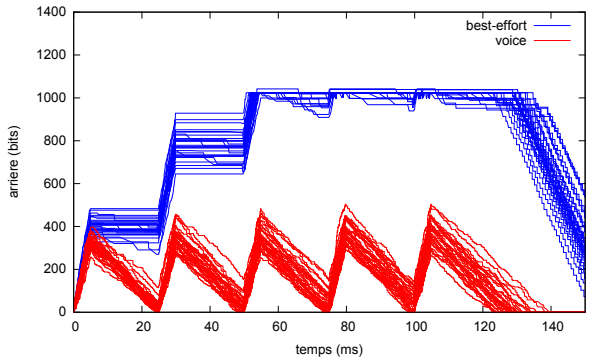
Figure 3.27 – Évolution de l'arriéré lors de l'augmentation de taille des paquets (aucun puis deux flux agrégés)

Les résultats montrés en figure 3.28 sont différents de ceux illustrés figure 3.27 pour les couples de figures suivants :

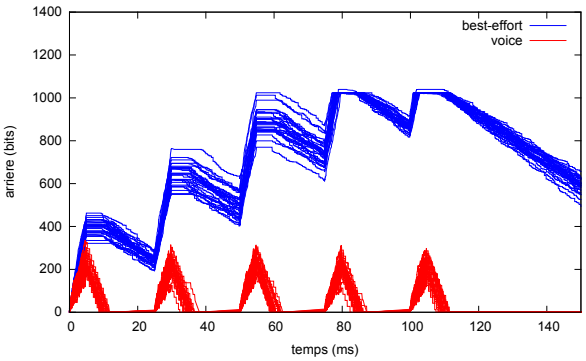
— figures 3.27b et 3.28b, deux flux $e500v_{e50}$, les deux sont agrégés puis seul le flux *voice* l'est.

— figures 3.27f et 3.28f, seul le flux *best-effort* suit $e1500v_{e50}$, les deux sont agrégés puis seul le flux *voice* l'est.

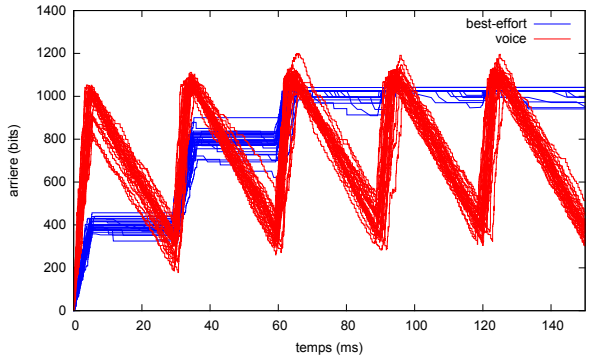
Dans tous ces cas, le flux *voice* est composé de paquets de 500 octets (en moyenne). En figures 3.27b et 3.27f, le flux *best-effort* est agrégé ce qui lui permet de transmettre un plus grand nombre de paquets. Pour les figures 3.28c, 3.28d,3.27c et 3.27d quel que soit le nombre de flux agrégés, le flux *best-effort* ne



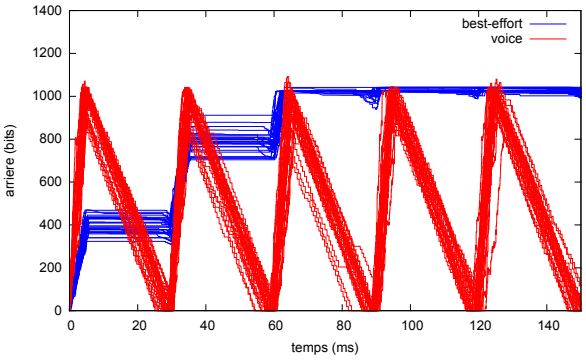
(a) Flux $e500v_{e50}$ (*best-effort* agrégé)



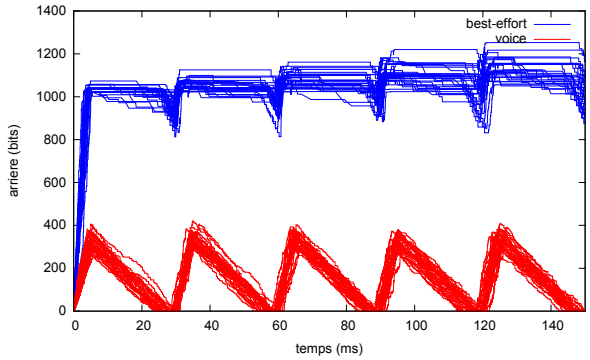
(b) Flux $e500v_{e50}$ (*voice* agrégé)



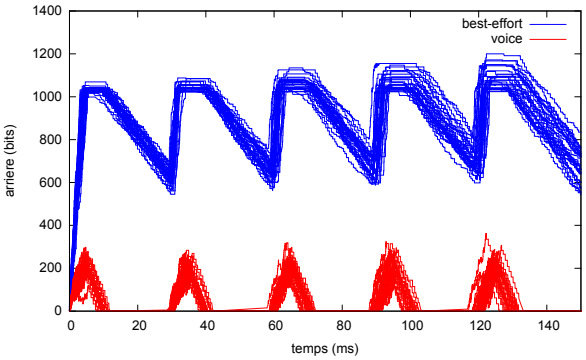
(c) Flux *voice* $e1500v_{e50}$ (*best-effort* agrégé)



(d) Flux *voice* $e1500v_{e50}$ (*voice* agrégé)



(e) Flux *best-effort* $e1500v_{e50}$ (*best-effort* agrégé)

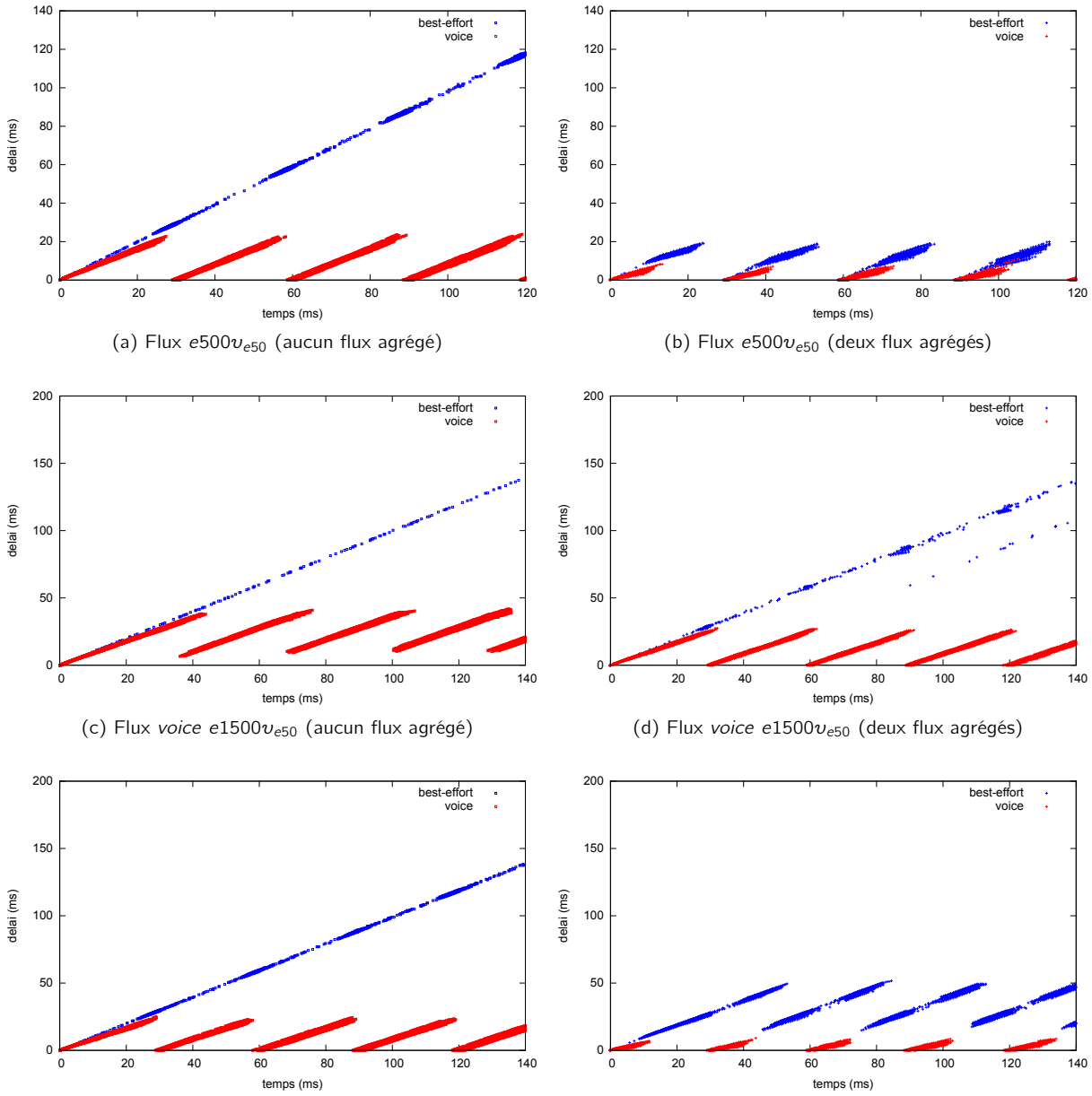


(f) Flux *best-effort* $e1500v_{e50}$ (*voice* agrégé)

Figure 3.28 – Évolution de l'arriéré lors de l'augmentation de taille des paquets (un seul flux agrégé)

bénéficiera que très peu des améliorations apportées par l'agrégation puisqu'il reste dans une situation proche de la famine même lorsque le flux *voice* est agrégé. Agréger le flux *best-effort* ne permet pas d'améliorer ses performances puisqu'il n'a même pas accès au medium.

Les mesures de l'arriéré et du délai montrent que le système est complètement saturé. À tel point qu'il en devient inutile d'agréger le flux *best-effort* puisqu'il est dans l'incapacité de transmettre un nombre significatif de trames. Ces conclusions nous permettent de proposer plusieurs recommandations sur la façon de configurer le système d'agrégation en fonction des indications fournies en observant l'arriéré de traitement



(a) Flux $e500v_{e50}$ (aucun flux agrégé)

(b) Flux $e500v_{e50}$ (deux flux agrégés)

(c) Flux *voice* $e1500v_{e50}$ (aucun flux agrégé)

(d) Flux *voice* $e1500v_{e50}$ (deux flux agrégés)

(e) Flux *best-effort* $e1500v_{e50}$ (aucun flux agrégé)

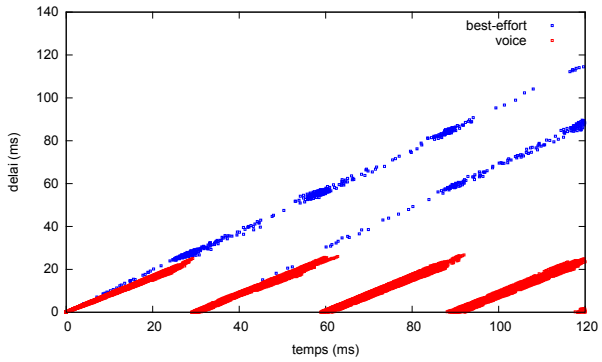
(f) Flux *best-effort* $e1500v_{e50}$ (deux flux agrégés)

Figure 3.29 – Évolution du délai lors de l'augmentation de taille des paquets (aucun puis deux flux agrégés)

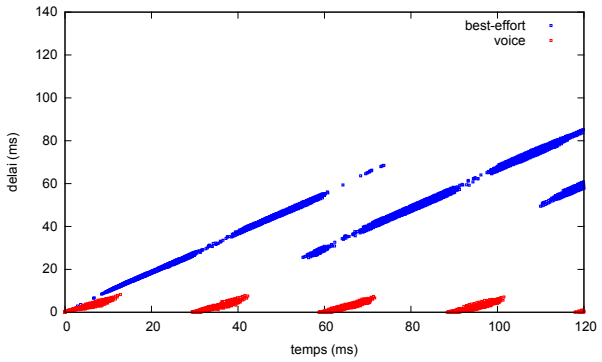
et le délai.

Recommandation 1 : *Lorsqu'un flux de classe inférieure est dans une situation proche de la famine, les flux concurrents de classes supérieures doivent être agrégés en priorité.*

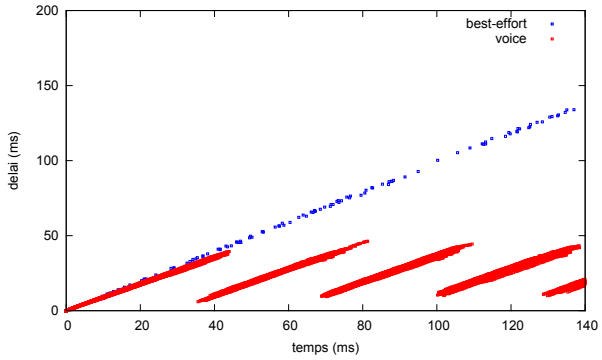
Recommandation 2 : *S'il n'est pas souhaitable d'agréger les flux concurrents de classes supérieures ou si les délais et arriérés mesurés ne sont toujours pas satisfaisants, le flux considéré doit être agrégé. Cette mesure peut être inefficace si les flux de classes supérieures saturent le système.*



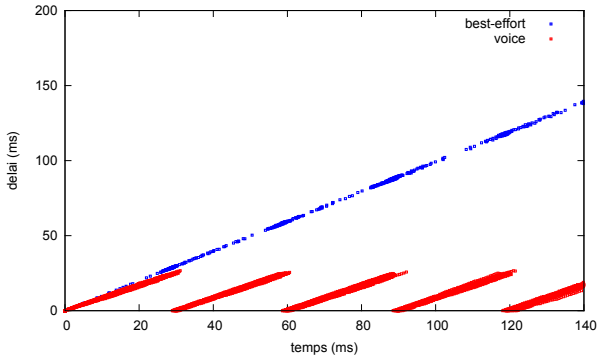
(a) Flux $e500v_{e50}$ (*best-effort* agrégé)



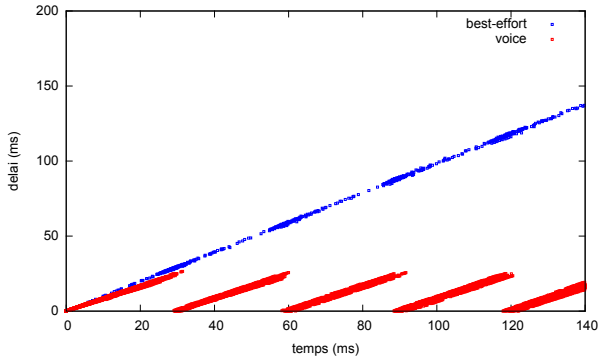
(b) Flux $e500v_{e50}$ (*voice* agrégé)



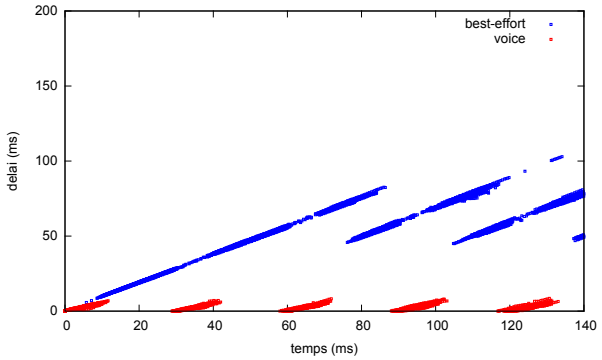
(c) Flux *voice* $e1500v_{e50}$ (*best-effort* agrégé)



(d) Flux *voice* $e1500v_{e50}$ (*voice* agrégé)



(e) Flux *best-effort* $e1500v_{e50}$ (*best-effort* agrégé)



(f) Flux *best-effort* $e1500v_{e50}$ (*voice* agrégé)

Figure 3.30 – Évolution du délai lors de l'augmentation de taille des paquets (un seul flux agrégé)

3.3.5 Influence de l'inter-arrivée des paquets

Dans cette partie c'est non seulement l'inter-arrivée des paquets qui est modifiée mais également leur synchronisation ou plutôt la non-synchronisation des rafales (comme le montre la figure 3.31). Ce comportement est obtenu en configurant un temps de pause constant après chaque rafale et en laissant le même nombre de paquets (100) à chaque flux indépendamment de leurs inter-arrivées respectives. Cela a deux conséquences, la rafale dure plus longtemps lorsque l'inter-arrivée vaut 150 μs et la pause retarde le début de la rafale suivante puisque sa valeur est fixe.

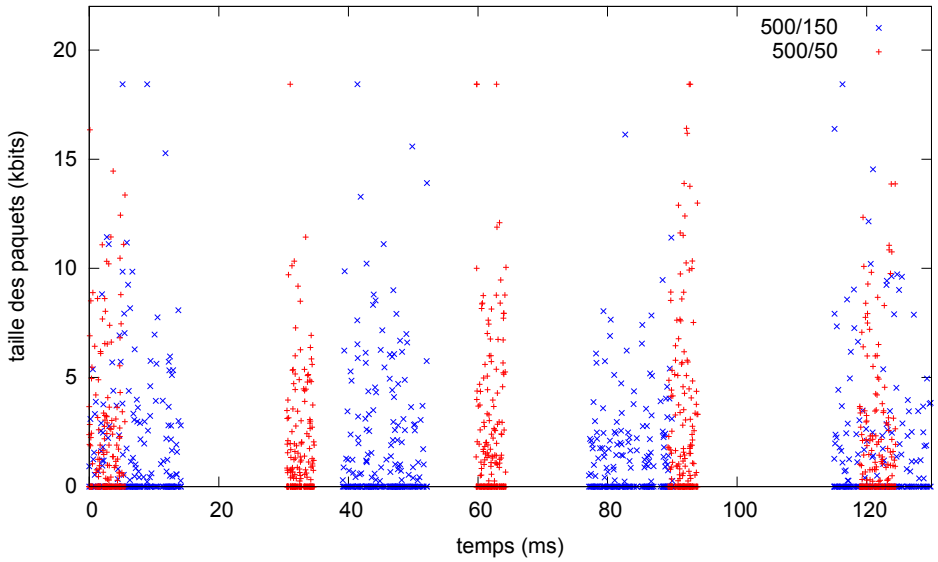


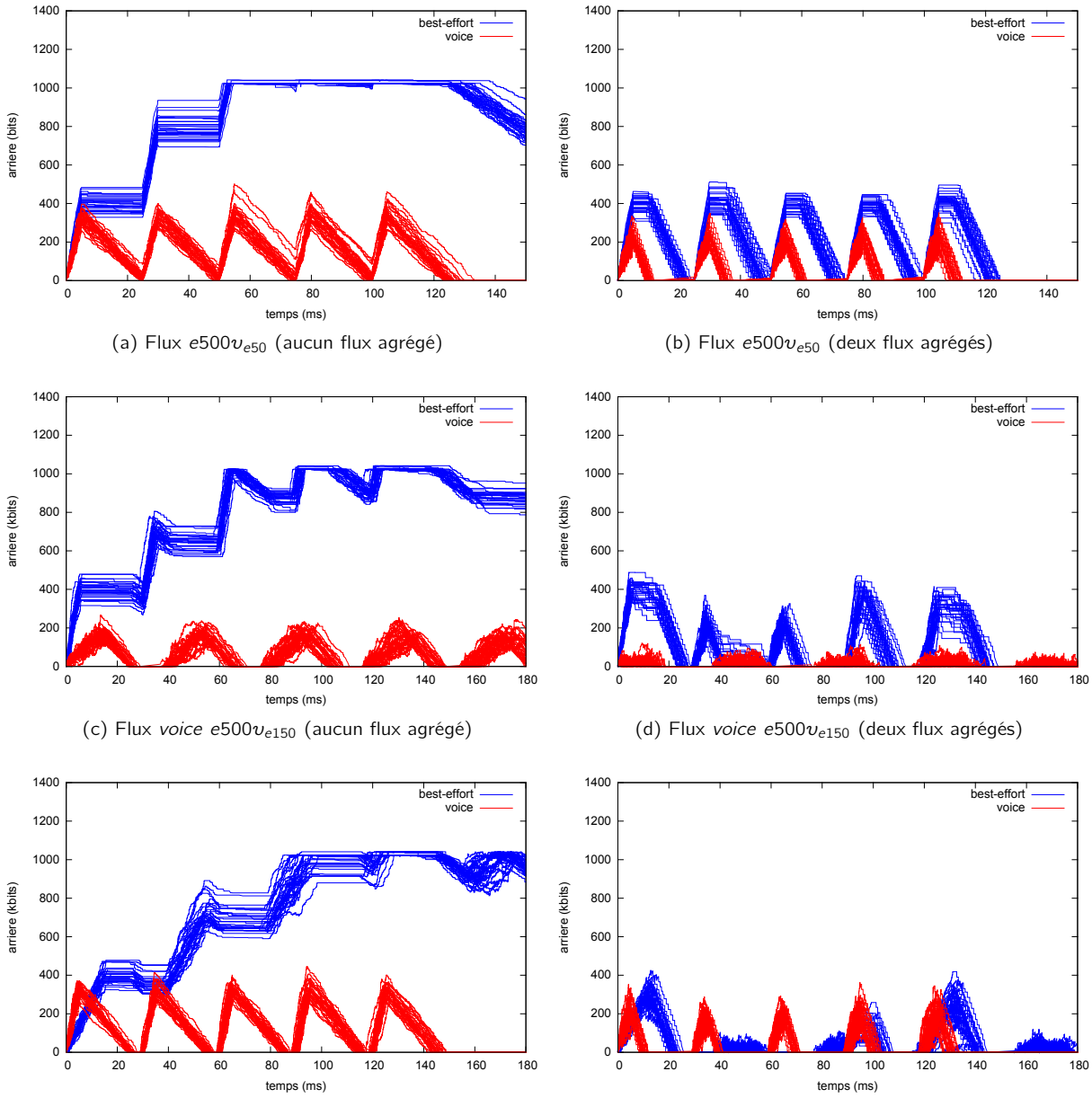
Figure 3.31 – Trafic entrant

Dans le même esprit que la section précédente, la figure 3.32 montre l'arriéré en colonne de gauche lorsque aucun flux n'est agrégé et sur la droite lorsque les deux le sont. La figure 3.33 montre l'arriéré en colonne de gauche lorsque la classe *best-effort* est agrégée et sur la droite lorsque la classe *voice* l'est. Chaque ligne sur ces deux figures reprend une des trois combinaisons de flux définies ($e500\upsilon_{e50}$ et $e1500\upsilon_{e50}$ pour le flux *best-effort* puis *voice*).

Même si aucun flux n'est agrégé, l'arriéré de traitement est réduit lorsque les paquets sont plus espacés et avec des rafales non synchronisées (figures 3.32c et 3.32e). Le flux *best-effort* sature toujours la mémoire dans ce cas mais légèrement plus tardivement. Les bénéfices apportés par l'agrégation des deux classes restent importants malgré cette amélioration due aux caractéristiques des flux (figures 3.32d et 3.32f). On observe d'ailleurs l'influence de la synchronisation des rafales sur ces figures. Lorsque les deux flux débutent leurs rafales au même moment, l'arriéré du flux *best-effort* augmente significativement car il doit attendre que le flux *voice* ait transmis ses agrégats.

Le flux *voice* changeant d'inter-arrivée moyenne (de 50 à 150 μs), l'arriéré du flux *best-effort* n'est pas fortement impacté et atteint la même valeur que précédemment sur la première phase ou les flux sont encore synchronisés (figure 3.33a jusque t=25 ms). Cependant, la non synchronisation des rafales va fortement influer sur ce même arriéré comme illustré en figure 3.33c. C'est la combinaison de cette absence de synchronisation avec le temps supplémentaire laissé par un flux concurrent moins conséquent qui permettra au flux *best-effort* de transmettre plus d'agrégats. La remarque précédente est valable pour la figure 3.33e également mais il n'y a aucune influence notable de l'augmentation de l'inter-arrivée du flux *best-effort* sur le flux *voice*. Comme dans le cas précédent l'arriéré le plus faible pour le flux *best-effort* est obtenu lorsque son concurrent est agrégé (figures 3.33d et 3.33f).

Une inter-arrivée plus élevée pour un flux peut diminuer son arriéré de traitement en fonction de ses concurrents. Couplée à une non synchronisation des rafales entre flux, elle permet à coup sûr de réduire l'arriéré et les délais (figure 3.34). Agréger est donc plus intéressant lorsque les rafales sont synchronisées. À ce moment de forte concurrence, il est nécessaire de transmettre le plus rapidement possible les paquets accumulés. Agréger aide à éviter de dépasser une exigence de délai potentiellement entamée par la trans-



(a) Flux $e500v_{e50}$ (aucun flux agrégé)

(b) Flux $e500v_{e50}$ (deux flux agrégés)

(c) Flux *voice* $e500v_{e150}$ (aucun flux agrégé)

(d) Flux *voice* $e500v_{e150}$ (deux flux agrégés)

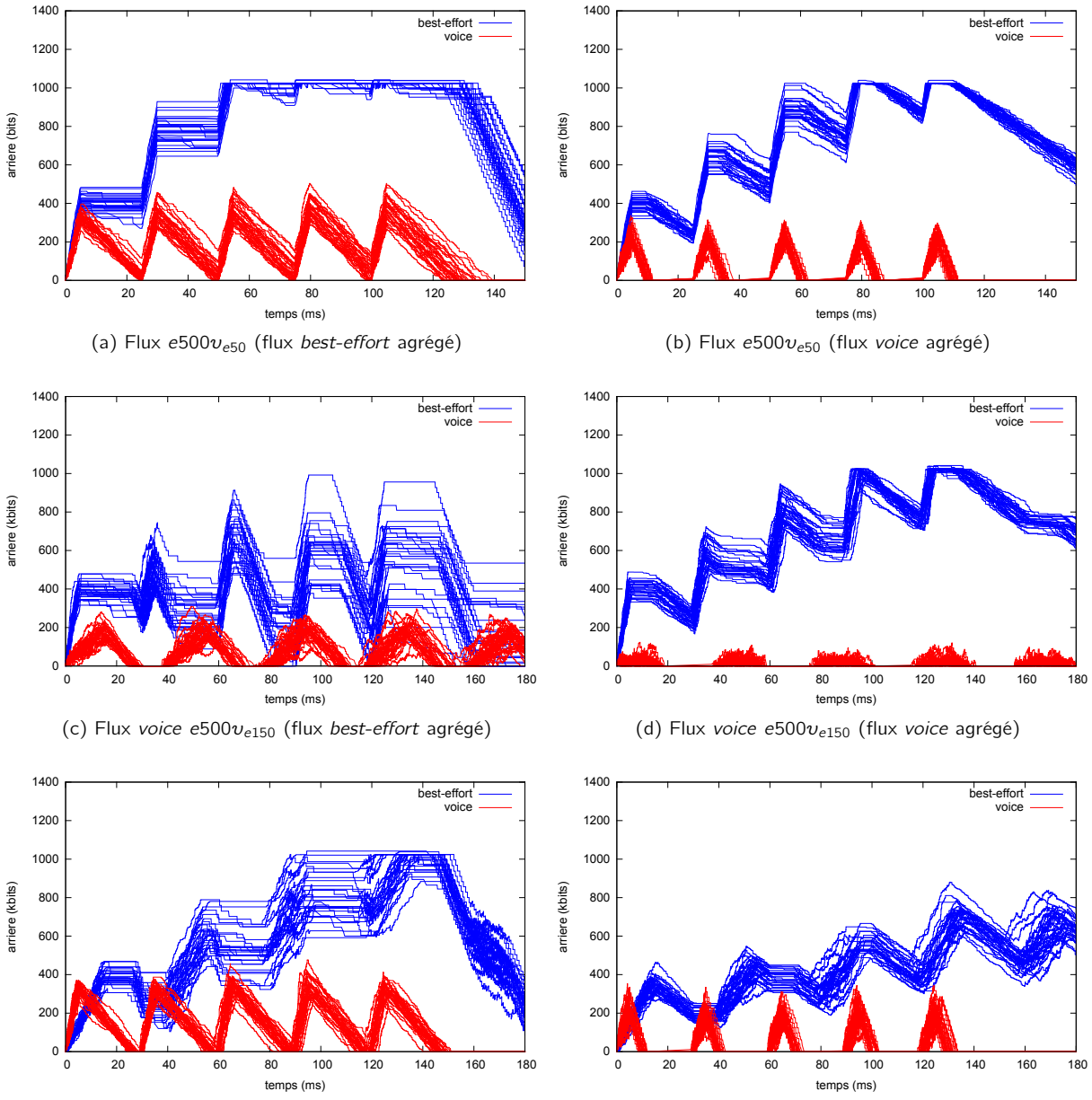
(e) Flux *best-effort* $e500v_{e150}$ (aucun flux agrégé)

(f) Flux *best-effort* $e500v_{e150}$ (deux flux agrégés)

Figure 3.32 – Évolution de l'arriéré lors de l'augmentation de l'inter-arrivée des paquets (aucun puis deux flux agrégés)

mission d'une rafale en provenance d'un flux concurrent prioritaire. Nous pouvons ainsi ajouter une nouvelle condition à la première recommandation proposée précédemment et la reformuler comme suit :

Recommandation 1 : *Lorsqu'un flux de classe inférieure est dans une situation proche de la famine OU lorsque les rafales des flux en concurrence pour le medium sont synchronisées (par définition de l'application par exemple), les flux de classes supérieures doivent être agrégés en priorité.*



(a) Flux $e500v_{e50}$ (flux *best-effort* agrégé)

(b) Flux $e500v_{e50}$ (flux *voice* agrégé)

(c) Flux *voice* $e500v_{e150}$ (flux *best-effort* agrégé)

(d) Flux *voice* $e500v_{e150}$ (flux *voice* agrégé)

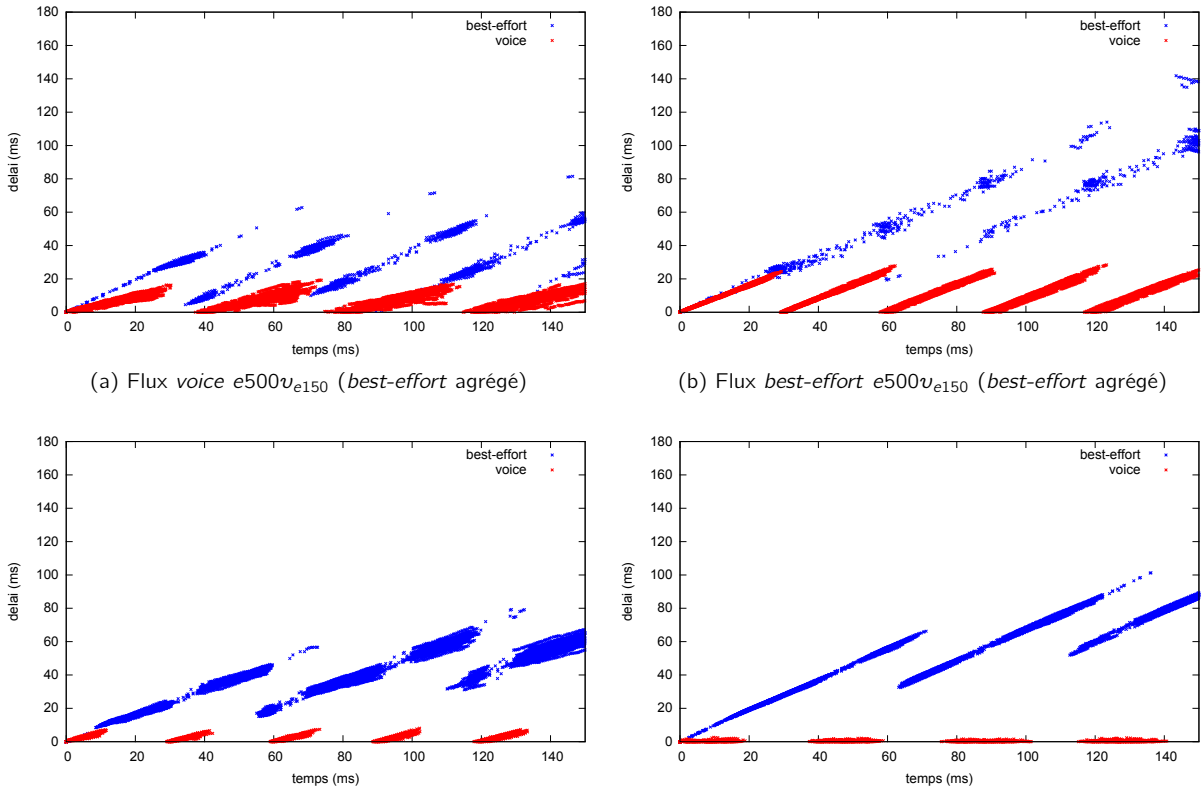
(e) Flux *best-effort* $e500v_{e150}$ (flux *best-effort* agrégé)

(f) Flux *best-effort* $e500v_{e150}$ (flux *voice* agrégé)

Figure 3.33 – Évolution de l'arriéré lors de l'augmentation de l'inter-arrivée des paquets (un seul flux agrégé)

3.4 Influence des paramètres d'agrégation

Dans cette section nous cherchons à déterminer l'influence de l'agrégation, dans le choix de la ou des file(s) qui agrègent ainsi que dans le choix du seuil de taille des agrégats. Malgré des caractéristiques plus variées, les flux considérés dans la suite de cette section suivent toujours une distribution exponentielle sur trente tirages. Dans la suite de cette étude les tracés de l'arriéré de traitement reprendront uniquement ses valeurs minimales et maximales ce qui formera une enveloppe autour des toutes les évolutions mesurées. Le second indicateur que nous utilisons est le délai et il sera également utilisé ici. Cela nous permettra d'étudier



(a) Flux *voice* $e500v_{e150}$ (*best-effort* agrégé)

(b) Flux *best-effort* $e500v_{e150}$ (*best-effort* agrégé)

(c) Flux *best-effort* $e500v_{e150}$ (*voice* agrégé)

(d) Flux *voice* $e500v_{e150}$ (*voice* agrégé)

Figure 3.34 – Délai lorsque l'inter-arrivée des paquets augmente

une éventuelle corrélation avec l'arriéré de traitement.

Les variables utilisées ainsi que leurs valeurs sont définies en tableau 3.3 pour les paramètres de contrôle et en tableau 3.1 pour ceux des flux utilisateur. Nous considérons toujours un scénario dans lequel deux stations émettent chacune un flux de classe de service différente (*best-effort* pour l'une, *voice* pour l'autre).

Les quatre cas définis auparavant sont rappelés ci-dessous :

— aucun flux n'est agrégé.
— seul le flux *best-effort* est agrégé.
— seul le flux *voice* est agrégé.
— les deux flux sont agrégés.

Pour chacun de ces cas, les flux agrégés peuvent prendre deux valeurs de seuil (qui sont définies dans le standard) ce qui donne les neuf combinaisons suivantes :

— les deux flux sont agrégés.
  — les deux seuils de taille sont au minimum (3839 octets).
  — seul le seuil de taille du flux *best-effort* est au maximum (7935 octets).
  — seul le seuil de taille du flux *voice* est au maximum.
  — les deux seuils de taille sont au maximum.
— seul le flux *best-effort* est agrégé
  — son seuil est au minimum

  - son seuil est au maximum
- seul le flux *voice* est agrégé
  - son seuil est au minimum
  - son seuil est au maximum
- aucun flux n'est agrégé

Cette étude est basée sur les évolutions du délai et de l'arriéré de traitement pour chaque flux et pour chaque combinaison exprimée ci-dessus (soit neuf figures pour l'arriéré de traitement et autant pour le délai).

3.4.1 Sur l'arriéré de traitement

3.4.1.a Avec un flux de 125 octets / 40 μs ($e125\upsilon_{e40}$)

Les figures 3.35 à 3.38a représentent l'évolution de l'arriéré de traitement dans chacun des neuf cas détaillés précédemment. Pour les 30 tirages effectués, l'arriéré évolue au sein de l'enveloppe tracée dans ces figures. Les caractéristiques choisies ici pour les flux ne saturent pas le système puisque même dans le pire cas (figure 3.35) celui-ci est capable d'absorber complètement le trafic à chaque phase de rafale.

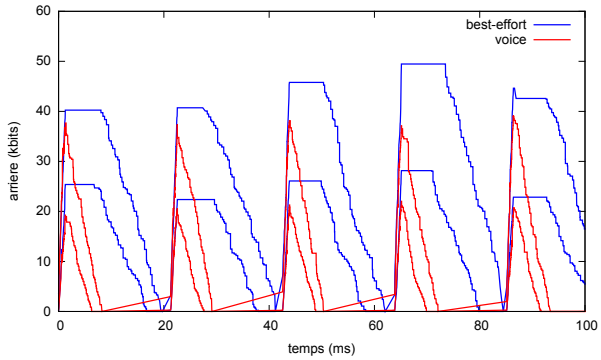
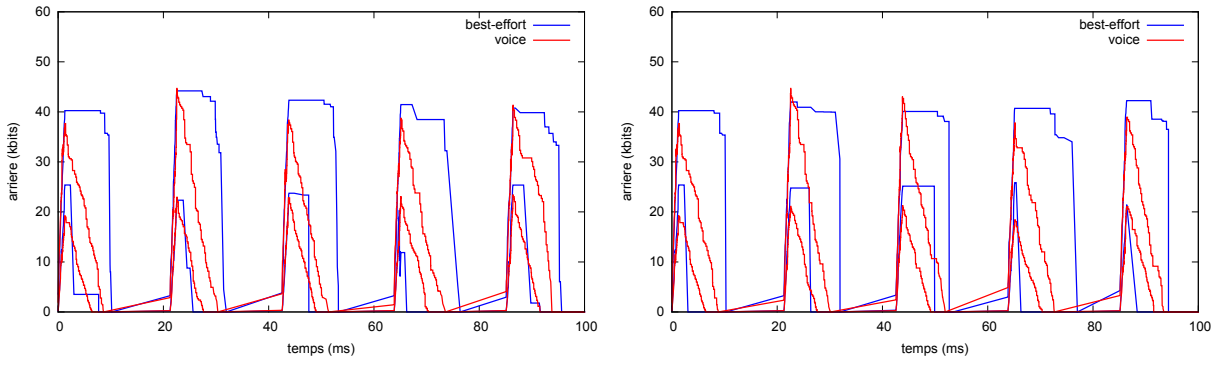


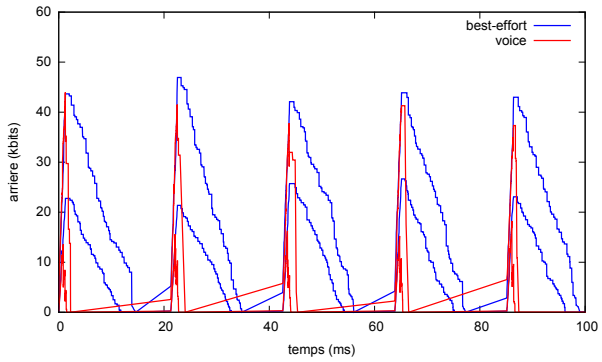
Figure 3.35 – Évolution de l'arriéré de traitement lorsque aucun flux n'est agrégé ($e125\upsilon_{e40}$)



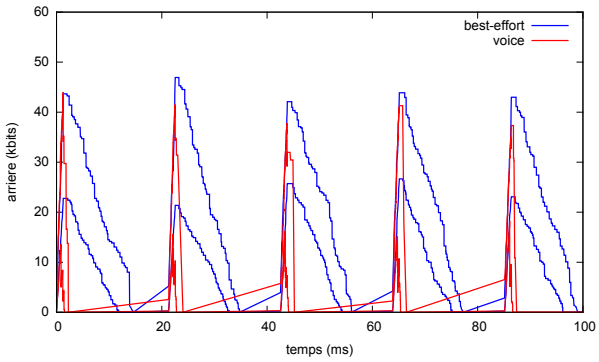
(a) Au seuil bas (b) Au seuil haut

Figure 3.36 – Évolution de l'arriéré de traitement lorsque le flux *best-effort* est agrégé ($e125\upsilon_{e40}$)

Contrairement au scénario utilisé en figure 3.8a agréger ne réduit pas systématiquement l'arriéré de traitement maximal dans le système. En effet, une comparaison entre les figures ci-dessus montre que

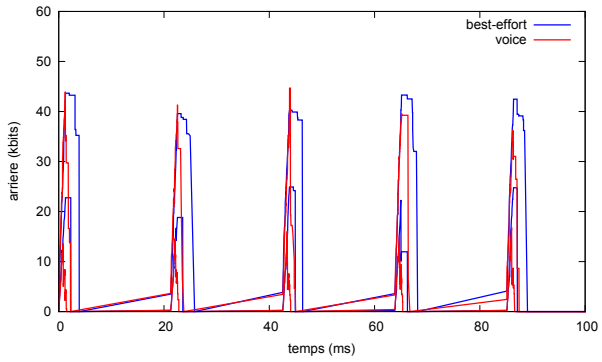


(a) Au seuil bas

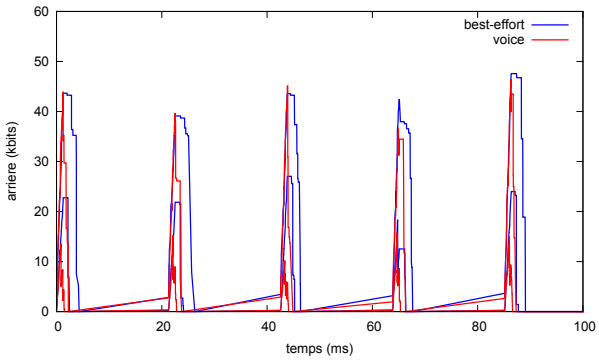


(b) Au seuil haut

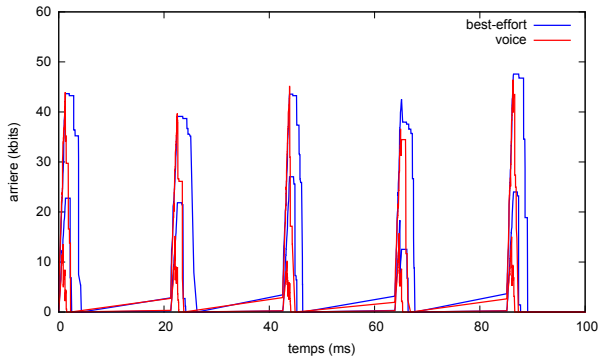
Figure 3.37 – Évolution de l'arriéré de traitement lorsque le flux *voice* est agrégé ($e125v_{e40}$)



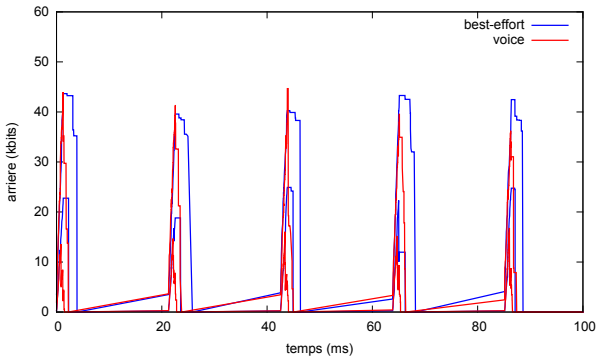
(a) Deux flux au seuil haut



(b) Flux *voice* au seuil haut



(c) Deux flux au seuil bas



(d) Flux *best-effort* au seuil haut

Figure 3.38 – Évolution de l'arriéré de traitement lorsque les deux flux sont agrégés ($e125v_{e40}$)

l'arriéré maximal atteint est sensiblement identique dans tous les scénarios. Cependant lorsque les flux sont agrégés la quantité de données en attente dans le système diminue soit plus tôt (si le concurrent prioritaire est agrégé) soit plus rapidement (si le flux considéré est agrégé) ou encore une combinaison des deux. Les délais d'attente sont donc réduits plus où moins efficacement en fonction du scénario.

De nouveau, la configuration la plus efficace est l'agrégation des deux flux qui combine les avantages décrits ci-avant. Parmi les quatre configurations de seuil possibles aucune n'apporte de meilleures performances

qu'une autre.

Deux autres choix sont possibles, agréger uniquement le flux *best-effort* ou uniquement le flux *voice*. Au vu des profils d'arriéré tracés, en particulier sur la partie descendante des courbes, il apparaît qu'agréger uniquement le flux *best-effort* réduise l'écart entre les délais subis par les paquets de ce flux et n'apporte aucune amélioration au flux prioritaire sur lui. L'agrégation du flux *voice* seul améliore ses performances mais permet également aux trames du flux *best-effort* de subir des délai moins importants. Le fait d'utiliser un seuil plus grand dans un cas comme dans l'autre ne change pas significativement l'évolution de l'arriéré de traitement ce qui nous permet de poser une nouvelle recommandation.

Recommandation 3 : *Lorsque les paquets sont de petite taille (125 octets et moins) il est inutile de modifier le seuil de taille des agrégats pour réduire l'arriéré de traitement maximal.*

3.4.1.b Avec un flux de 250 octets / 25 μs ($e250v_{e50}$)

Le flux considéré ici ainsi que les suivants sont assez importants en terme de bande passante pour saturer le système. La taille des paquets et leur inter-arrivée changent proportionnellement afin de rester à débit constant. Les trois trafics utilisés suivent une loi exponentielle de paramètres 250 octets / 25 μs, 500 octets / 50 μs et 1500 octets / 150 μs.

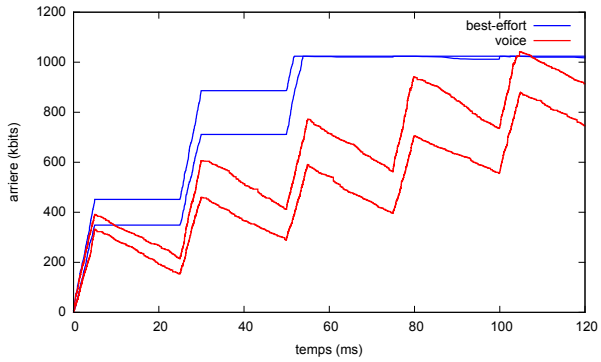
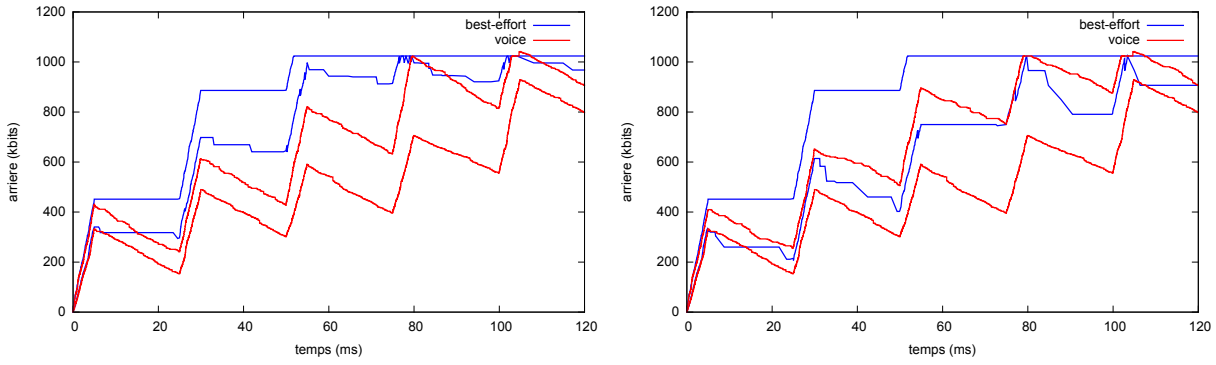
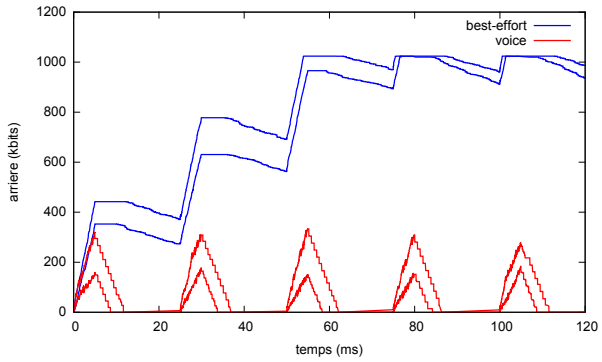


Figure 3.39 – Évolution de l'arriéré de traitement lorsque aucun flux n'est agrégé ($e250v_{e50}$)

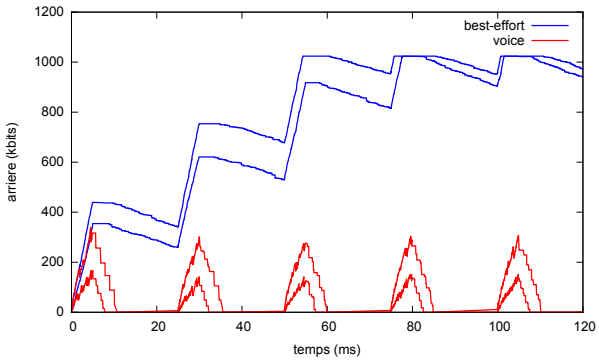


(a) Au seuil bas (b) Au seuil haut

Figure 3.40 – Évolution de l'arriéré de traitement lorsque le flux *best-effort* est agrégé ($e250v_{e50}$)

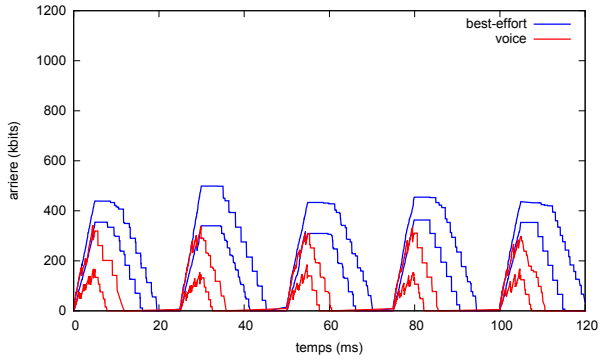


(a) Au seuil bas

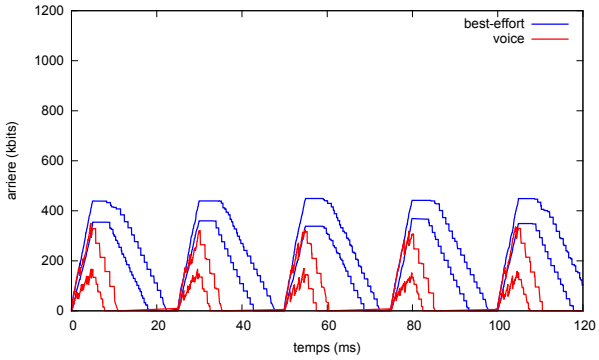


(b) Au seuil haut

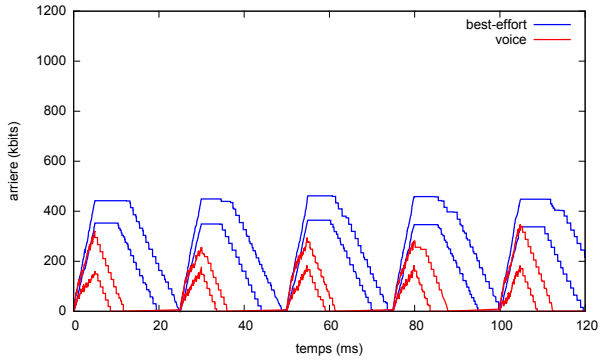
Figure 3.41 – Évolution de l'arriéré de traitement lorsque le flux *voice* est agrégé ($e250v_{e50}$)



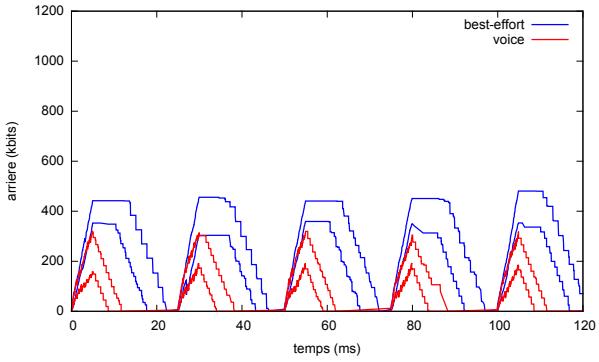
(a) Deux flux au seuil haut



(b) Flux *voice* au seuil haut



(c) Deux flux au seuil bas



(d) Flux *best-effort* au seuil haut

Figure 3.42 – Évolution de l'arriéré de traitement lorsque les deux flux sont agrégés ($e250v_{e50}$)

L'amélioration apportée par l'agrégation des deux flux est évidente. Sur les figures 3.42d à 3.42a, l'arriéré maximal est divisé par plus de deux ce qui permet au système d'assurer un service sans pertes. Au contraire sur les figures 3.40a à 3.39, l'arriéré dépasse la limite mémoire à chaque fois pour le flux *best-effort* et dans trois cas sur cinq pour le flux *voice* (les cas ou il n'est pas agrégé). Comme observé auparavant, si l'on fait abstraction du cas ou les deux flux sont agrégés, le cas offrant les meilleures performances pour les deux flux est l'agrégation du flux *voice* (figures 3.40a et 3.40b comparées aux figures 3.41a et 3.41b respectivement).

Cette étude nous permet de confirmer les trois recommandations proposées et d'éditer la définition d'un paquet de petite taille pour la troisième recommandation :

Recommandation 3 : *Lorsque les paquets sont de petite taille (250 octets et moins) il est inutile de modifier le seuil de taille des agrégats pour réduire l'arriéré de traitement maximal.*

3.4.1.c Avec un flux de 500 octets / 50 μs ($e500v_{e50}$)

Cette étude se poursuit maintenant avec un profil de trafic entrant dont le taux d'arrivée moyen à long terme est équivalent au précédent (500 octets / 50 μs).

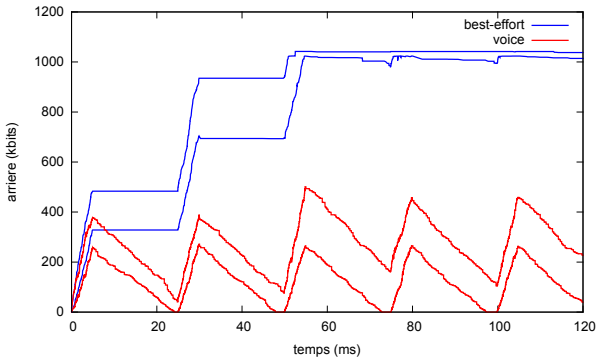
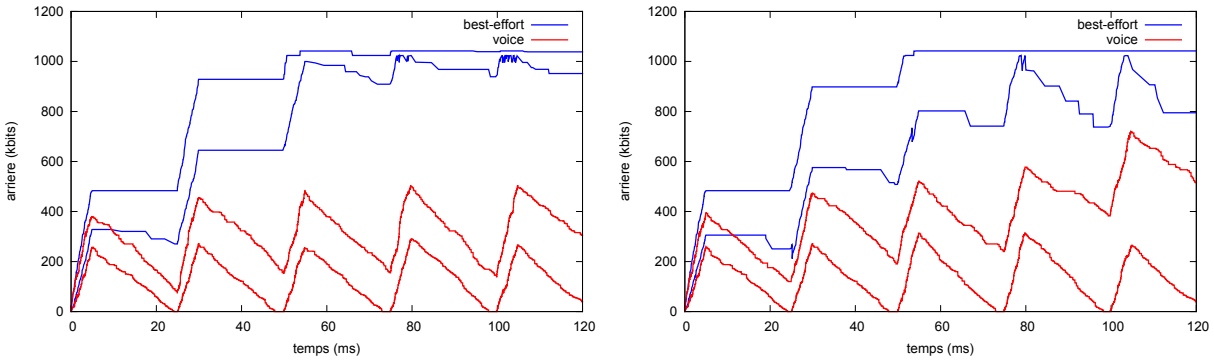


Figure 3.43 – Évolution de l'arriéré de traitement lorsque aucun flux n'est agrégé ($e500v_{e50}$)

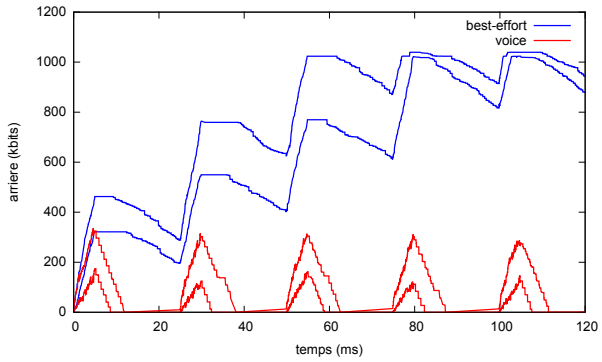


(a) Au seuil bas (b) Au seuil haut

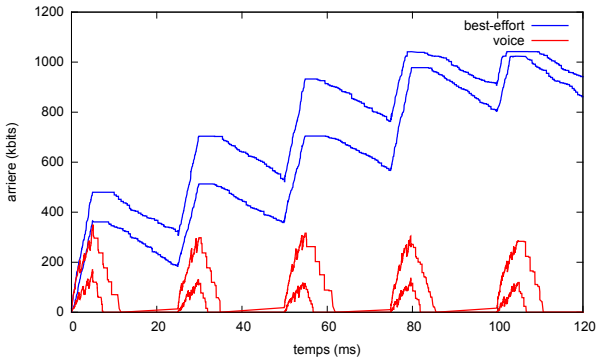
Figure 3.44 – Évolution de l'arriéré de traitement lorsque le flux *best-effort* est agrégé ($e500v_{e50}$)

Dans ce cas également, agréger les deux flux offre une réduction conséquente de l'arriéré de traitement. De plus l'augmentation du seuil de taille ne semble pas avoir d'influence sur l'arriéré maximal tracé sur les figures 3.46a à 3.46d. De même, l'arriéré du flux *best-effort* sera toujours inférieur si son concurrent est agrégé plutôt que lui même. En effet, sur les figures 3.45a et 3.45b la limite mémoire sera atteinte beaucoup plus tard que sur les figures 3.44a et 3.44b. Cette limite peut également ne pas être atteinte du tout selon les tirages. Le fait que le flux *voice* transmette plus rapidement ses paquets en les agrégeant laisse du temps au flux *best-effort* pour transmettre les siens. L'arriéré de celui-ci est donc contenu.

Néanmoins, le fait de former d'agréger le flux *best-effort* peut jouer sur les délais du flux *voice* et ce quelle que soit la taille limite des agrégats. En effet, nous avons vu que le flux *best-effort* pouvait

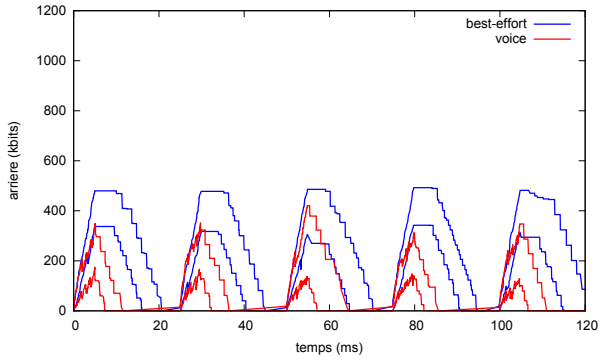


(a) Au seuil bas

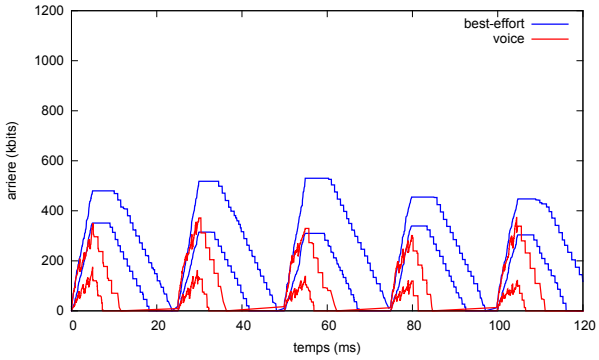


(b) Au seuil haut

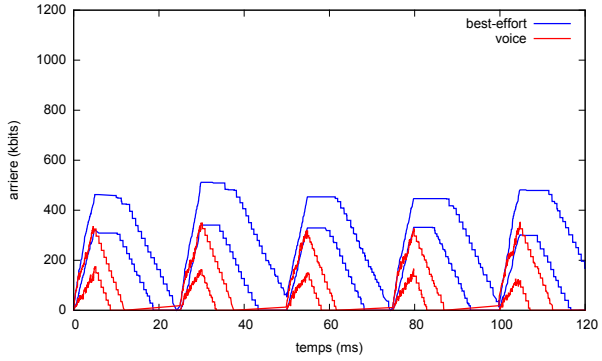
Figure 3.45 – Évolution de l'arriéré de traitement lorsque le flux *voice* est agrégé ($e500v_{e50}$)



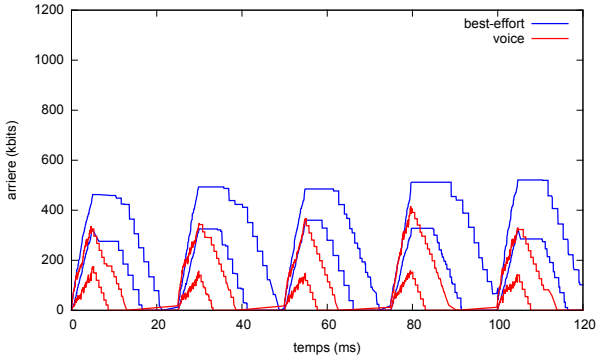
(a) Deux flux au seuil haut



(b) Flux *voice* au seuil haut



(c) Deux flux au seuil bas



(d) Flux *best-effort* au seuil haut

Figure 3.46 – Évolution de l'arriéré de traitement lorsque les deux flux sont agrégés ($e500v_{e50}$)

transmettre des paquets avant ceux de son concurrent prioritaire grâce au mécanisme de conservation du *backoff*. Cela conduit à une nette augmentation de l'arriéré du flux *voice* entre les figures 3.43, 3.44a et 3.44b. Cette augmentation pour le flux *voice* se traduit par une potentielle augmentation de l'arriéré et des délais pour le flux *best-effort*. En effet, augmenter le seuil de taille augmente la taille de l'enveloppe sur les résultats obtenus ce qui signifie que selon les tirages, augmenter le seuil aura des effets incertains. Les recommandations proposées sont encore une fois confirmées par l'étude malgré l'utilisation d'un flux entrant

aux caractéristiques différentes. Nous pouvons en proposer une nouvelle et éditer encore une fois la troisième recommandation sur la définition d'un paquet de petite taille.

Recommandation 3 : *Lorsque les paquets sont de petite taille (moins de 500 octets) il est inutile de modifier le seuil de taille des agrégats pour réduire l'arriéré de traitement maximal.*

Recommandation 4 : *Lorsque l'arriéré de traitement d'un flux de classe supérieure augmente jusqu'à atteindre la limite mémoire, l'agrégation doit être désactivée sur les flux de classe inférieure.*

3.4.1.d Avec un flux de 1500 octets / 150 μs ($e1500v_{e150}$)

Les flux considérés ici sont composés de paquets de 1500 octets (émis toutes les 150 μs en moyenne) ce qui correspond à la MTU (Maximum Transmission Unit) d'un réseau Ethernet. En effet, ce système d'agrégation était prévu à l'origine pour améliorer les performances des passerelles entre ces deux technologies (Ethernet et Wi-Fi). Voyons les résultats offerts avec la taille maximale des paquets sur Ethernet. Nous n'avons pas relevé de cas produisant des pertes ici.

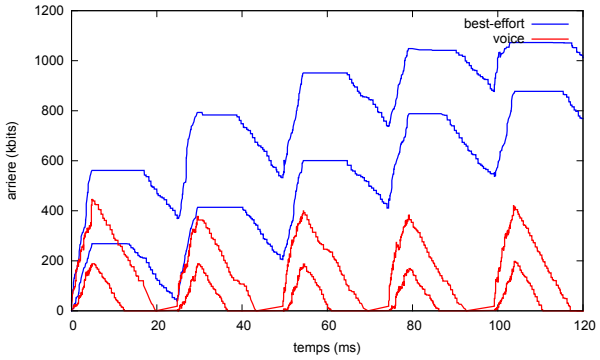
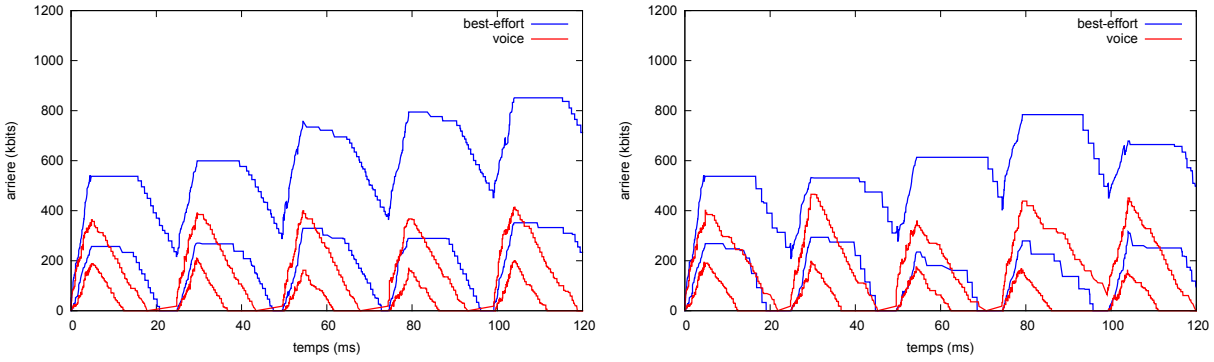


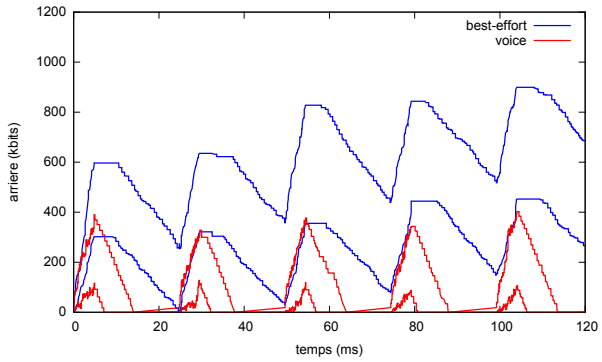
Figure 3.47 – Évolution de l'arriéré de traitement lorsque aucun flux n'est agrégé ($e1500v_{e150}$)



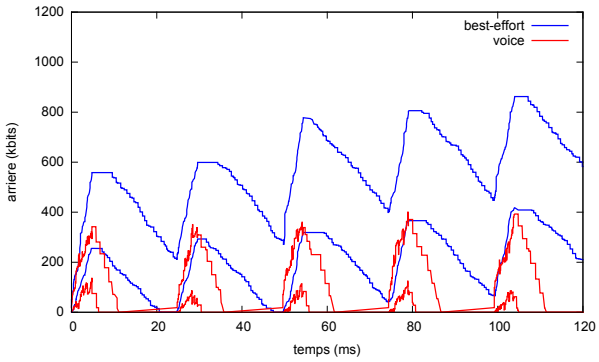
(a) Au seuil bas (b) Au seuil haut

Figure 3.48 – Évolution de l'arriéré de traitement lorsque le flux *best-effort* est agrégé ($e1500v_{e150}$)

D'abord, les améliorations apportées par l'agrégation sont moins conséquentes que précédemment. Cela est du à une taille plus importante des paquets qui pourrait être vue comme une agrégation à la source. Celle-ci diminue donc l'impact des en-têtes sur les trames transmises et l'arriéré tracé en figure 3.47 (sans

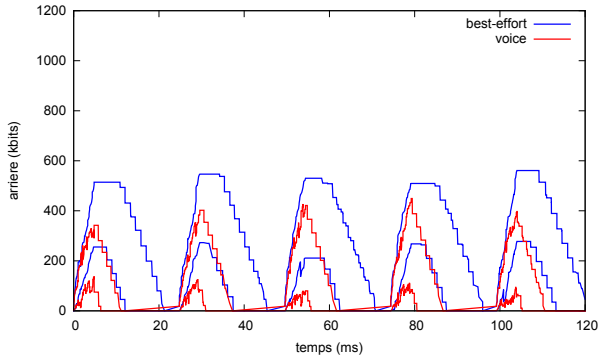


(a) Au seuil bas

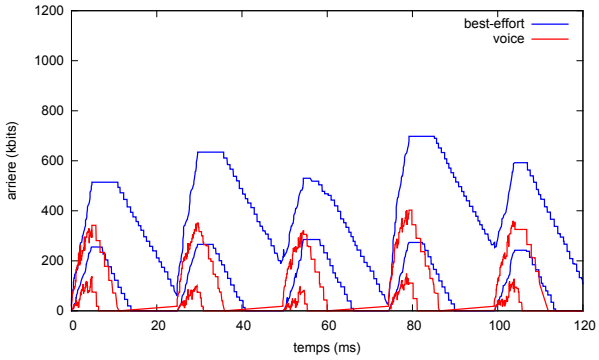


(b) Au seuil haut

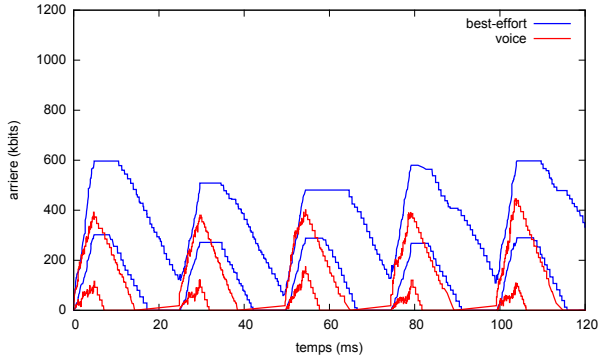
Figure 3.49 – Évolution de l'arriéré de traitement lorsque le flux *voice* est agrégé ($e1500v_{e150}$)



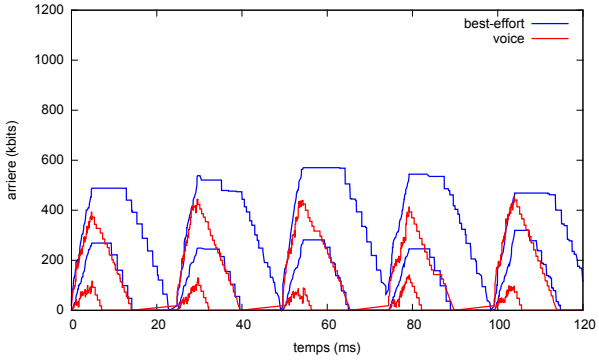
(a) Deux flux au seuil haut



(b) Flux *voice* au seuil haut



(c) Deux flux au seuil bas



(d) Flux *best-effort* au seuil haut

Figure 3.50 – Évolution de l'arriéré de traitement lorsque les deux flux sont agrégés ($e1500v_{e150}$)

agrégation) est déjà bien inférieur aux tracés précédents. C'est pourquoi aucun cas ne produira de pertes. Contrairement aux cas précédents, agréger le flux *best-effort* réduira plus efficacement l'arriéré qu'agréger son concurrent (figures 3.48a et 3.48b comparées aux figures 3.49a et 3.49b respectivement). Dans l'espace temporel laissé libre par le concurrent même s'il n'est pas agrégé, il est possible d'améliorer significativement le nombre de paquets transmis en agrégeant. Avec des paquets plus importants, l'influence du changement de seuil apparaît clairement et un seuil plus grand sera bénéfique pour l'arriéré ce qui confirme la troisième

recommandation et nous permet d'en ajouter une nouvelle.

Recommandation 5 : *Lorsque les paquets sont de grande taille (1500 octets et plus) les flux doivent être agrégés au seuil maximal pour réduire l'arriéré de traitement maximal.*

3.4.2 Sur le délai

Cette section sera menée de façon identique à la précédente ce qui indique également que nous utiliserons des profils de trafic identiques à ceux utilisés pour l'étude de l'arriéré de traitement.

3.4.2.a Avec un flux de 125 octets / 40 μs ($e125\upsilon_{e40}$)

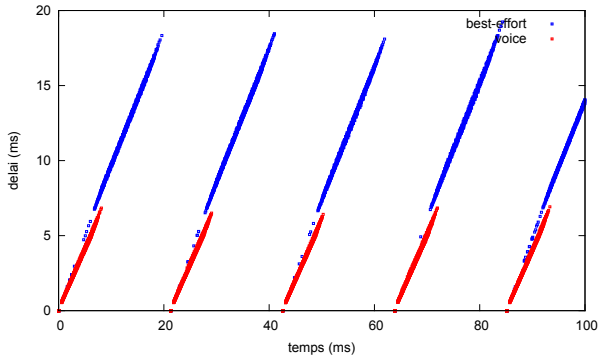
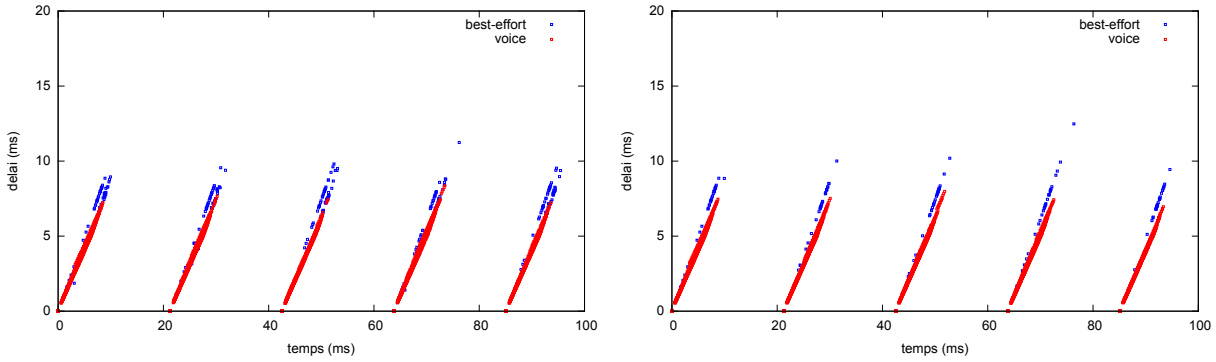


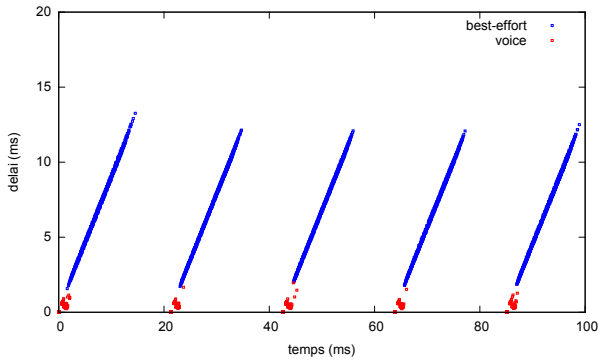
Figure 3.51 – Évolution du délai lorsque aucun flux n'est agrégé ($e125\upsilon_{e40}$)



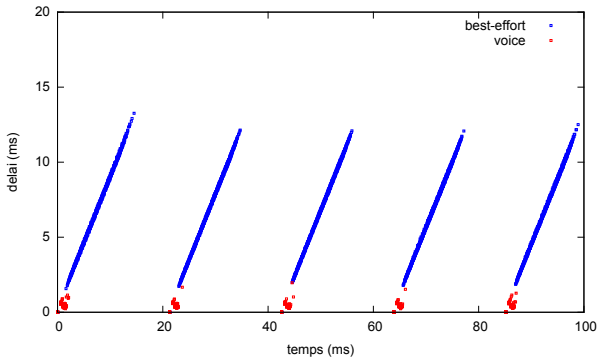
(a) Au seuil bas (b) Au seuil haut

Figure 3.52 – Évolution du délai lorsque le flux *best-effort* est agrégé ($e125\upsilon_{e40}$)

Les valeurs de délai représentées en figure 3.51 sont visiblement différentes entre les deux flux. Cette écart est fortement réduit lorsque le flux *best-effort* est agrégé (figure 3.52) puisque son délai moyen est réduit de 55 % et son pire délai de 41 %. Augmenter le seuil porte la réduction moyenne à 62 % comparé à des flux non agrégés (figure 3.52b). Tout comme pour l'arriéré de traitement, cette amélioration du délai est possible parce qu'après transmission des paquets du flux *voice*, la durée lors de laquelle le medium est libre est suffisante pour que le flux *best-effort* traite son arriéré. On notera que les délais subis par les paquets *voice* sont légèrement augmentés en moyenne (de l'ordre de 5 % en moyenne mais le pire cas augmente de

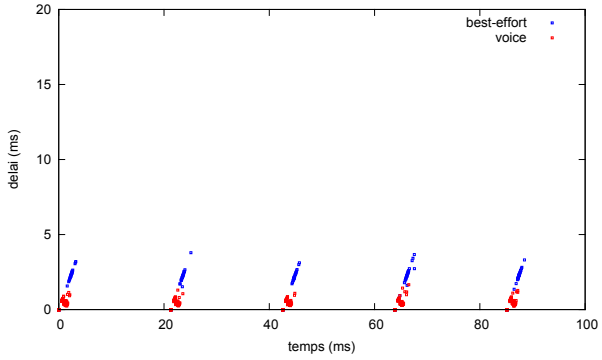


(a) Au seuil bas

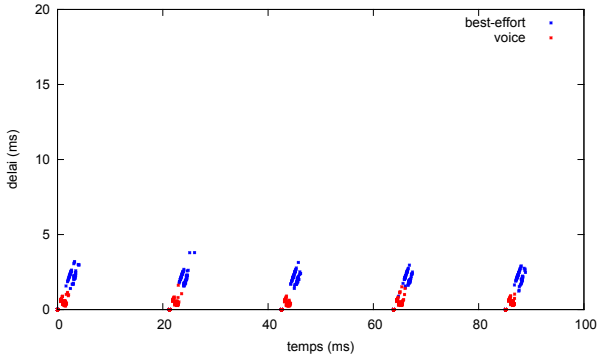


(b) Au seuil haut

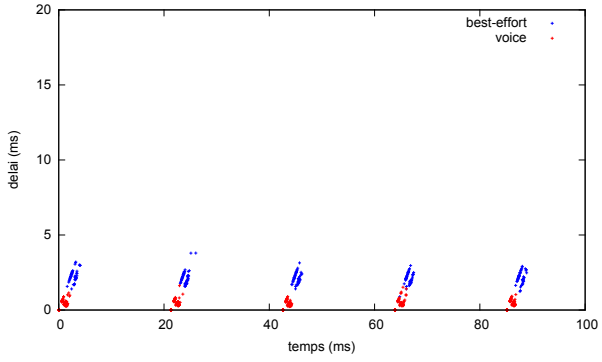
Figure 3.53 – Évolution du délai lorsque le flux *voice* est agrégé ($e125v_{e40}$)



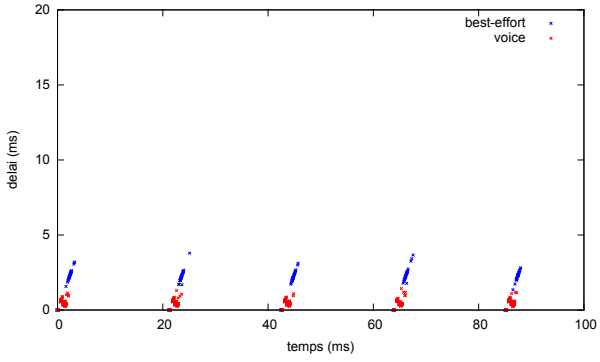
(a) Deux flux au seuil haut



(b) Flux *voice* au seuil haut



(c) Deux flux au seuil bas



(d) Flux *best-effort* au seuil haut

Figure 3.54 – Évolution du délai lorsque les deux flux sont agrégés ($e125v_{e40}$)

21 %). Ce résultat est causé par l'augmentation de la taille des trames transmises par le flux *best-effort*. Auparavant composées d'un paquet, elles en encapsulent maintenant une multitude (un agrégat) et lorsque le flux *best-effort* gagne l'accès au medium avant que toutes les trames du flux *voice* aient été transmises le délai payé par les paquets *voice* augmente en conséquence.

Observons maintenant les délais des paquets lorsque le flux *voice* est agrégé (figure 3.53). Maintenant agrégé, ses délais sont réduits de 86 % en moyenne ce qui constitue une forte amélioration. Cette importante

réduction empêche complètement le flux *best-effort* de transmettre avant que son concurrent ait vidé sa mémoire. D'ailleurs le fait de transmettre des agrégats diminue mécaniquement le nombre de fois où le medium est remis en jeu ce qui laisse très peu de place au flux *best-effort* pour transmettre ses trames. Malgré cela, son délai moyen est réduit de près de 45 % juste parce que le flux *voice* (son concurrent) est agrégé. Le fait de changer pour le seuil haut ne change en rien les performances observées puisque ce flux ne consomme pas assez de bande passante à lui seul pour forcer le système à former des agrégats de taille maximale.

Enfin, les derniers cas présentent tout simplement les plus fortes améliorations et ce pour les deux flux (figure 3.54) puisque lorsqu'ils sont agrégés les délais sont réduits de plus de 85 % dans tous les cas. Les valeurs sont extrêmement proches quel que soient les seuils considérés (2 % d'écart au maximum).

Nous pouvons faire un premier parallèle entre le comportement du délai sur ce cas et le comportement de l'arriéré étudié en section précédente sur le même flux. Il semble que ces deux indicateurs mènent aux mêmes conclusions, ce que nous allons tenter de confirmer en analysant les autres flux déjà considérés lors de l'étude de l'arriéré de traitement.

3.4.2.b Avec un flux de 250 octets / 25 μs ($e250v_{e50}$)

La figure 3.55 indique clairement que le trafic considéré est bien plus grand qu'auparavant. Le délai du flux *best-effort* monte jusque 356 *ms* (parce que la simulation s'arrête après cinq rafales), celui du flux *voice* atteint les 112 *ms* . Agréger le flux *best-effort* (figure 3.56) peut donc être utile pour les quelques trames qui passeront entre deux trames *voice* mais le gain obtenu semble marginal. Comme recommandé auparavant, c'est alors le flux *voice* qu'il faut agréger en priorité (figure **??**). Le medium étant libre pour le flux *best-effort* entre deux rafales, ses délais seront réduits de 55 % alors que lorsqu'il est lui-même agrégé la réduction apportée n'est que de 35 % (et sous réserve que l'arrivée cesse). De plus, le flux *voice* bénéficie dans ce cas de délais 95 % inférieurs.

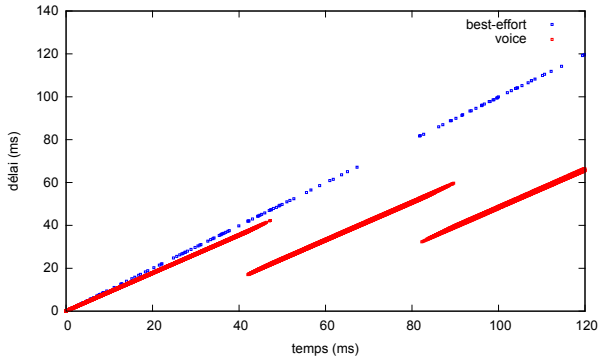
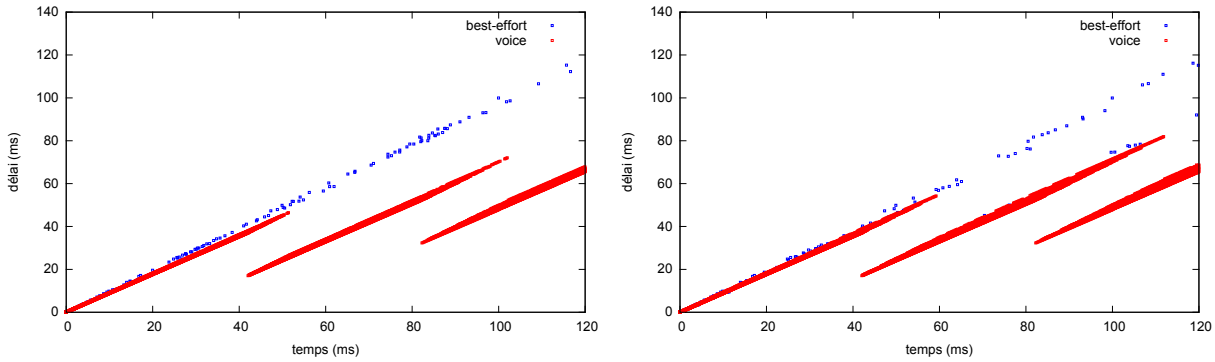


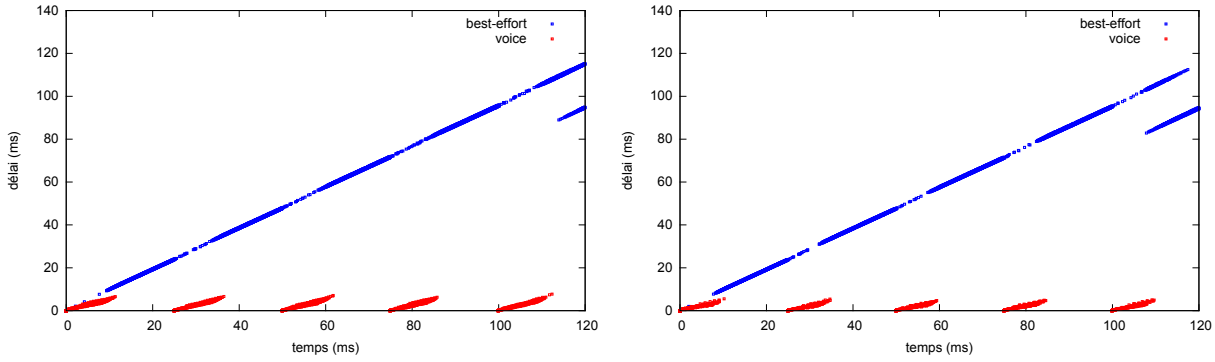
Figure 3.55 – Évolution du délai lorsque aucun flux n'est agrégé ($e250v_{e50}$)

En combinant les bénéfices (deux flux agrégés), il est possible de réduire d'environ 95 % les délais pour tous les flux considérés. Encore une fois que l'on choisisse le seuil bas ou haut, les résultats sont très similaires (moins de 1 % d'écart de performances). Toutes ces observations confirment bien qu'un parallèle peut être établi entre l'évolution de l'arriéré et du délai.



(a) Au seuil bas (b) Au seuil haut

Figure 3.56 – Évolution du délai lorsque le flux *best-effort* est agrégé ($e250v_{e50}$)



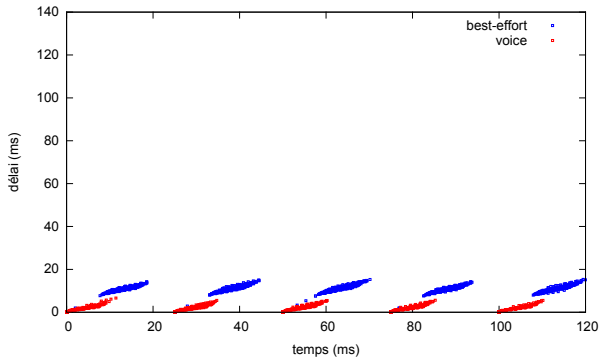
(a) Au seuil bas (b) Au seuil haut

Figure 3.57 – Évolution du délai lorsque le flux *voice* est agrégé ($e250v_{e50}$)

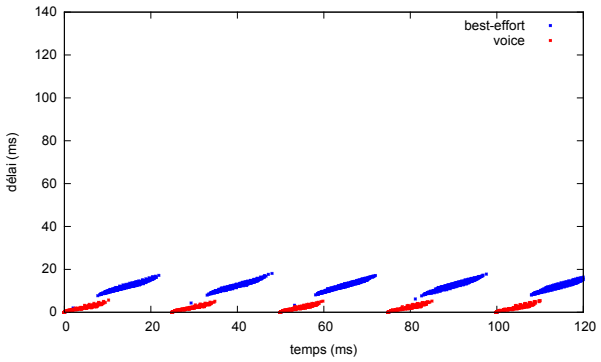
3.4.2.c Avec un flux de 500 octets / 50 μs ($e500v_{e50}$)

Les tracés étant similaires au cas précédent, étudions les valeurs de délais mesurées. Agréger le flux *voice* (figure 3.61) apporte toujours une réduction plus importante des délais du flux *best-effort* (50 % contre 45 % lorsqu'il est agrégé) mais l'écart entre ces deux choix est réduit (5 %) comparé aux résultats obtenus avec le flux considéré dans la section précédente (20 % d'écart). Lorsque seul le flux *best-effort* est agrégé (figure 3.60), les délais moyens du flux *voice* vont augmenter légèrement (de l'ordre de 8 %). Agréger les deux flux (figure 3.62) reste le plus efficace avec une réduction de 91 % pour le flux *best-effort* et de 70 à 80 % en fonction du seuil de taille choisi pour le flux *voice*.

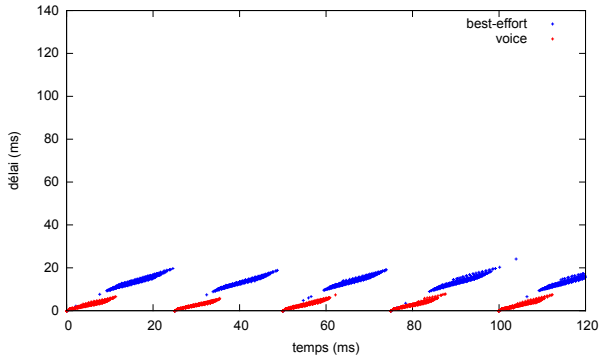
Ici le choix du seuil a une influence significative. Il est possible de gagner jusque 10 % sur le délai en augmentant le seuil pour le flux *voice*. Réduire la taille des agrégats du flux *best-effort* peut également améliorer le délai du flux *voice* selon s'il est lui même agrégé (3 %) ou non (40 %). On remarque alors que pour un flux prioritaire, l'agrégation constitue une protection contre des augmentations de délai dues à des changements de la taille des paquets en concurrence même s'il sont moins prioritaires. Encore une fois les observations sont similaires pour le délai et l'arriéré.



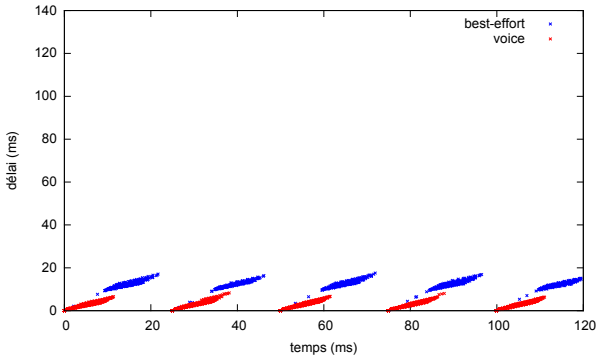
(a) Deux flux au seuil haut



(b) Flux *voice* au seuil haut



(c) Deux flux au seuil bas



(d) Flux *best-effort* au seuil haut

Figure 3.58 – Évolution du délai lorsque les deux flux sont agrégés ($e250v_{e50}$)

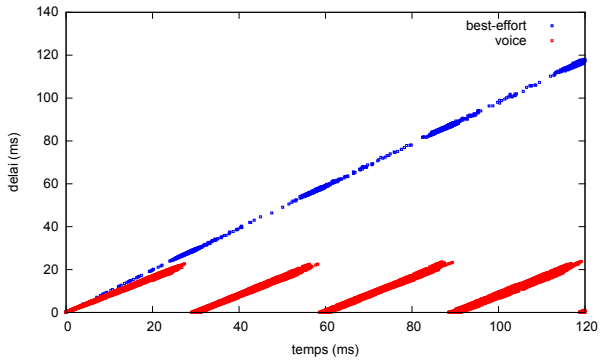
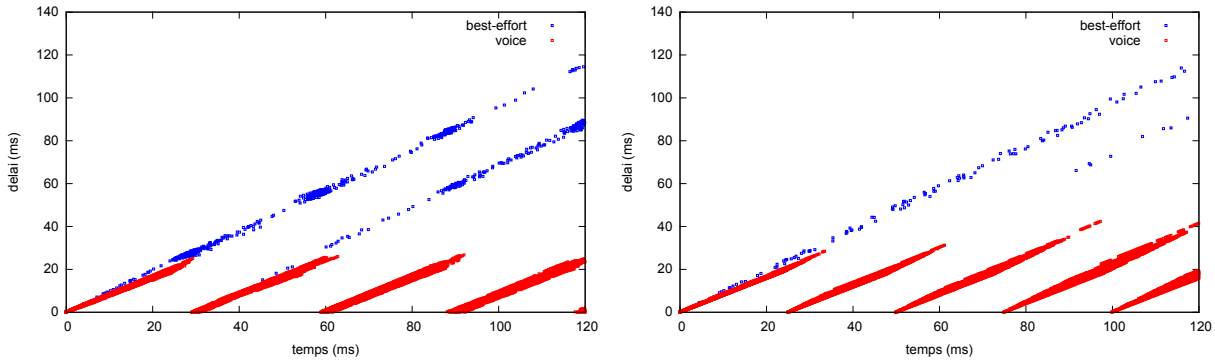


Figure 3.59 – Évolution du délai lorsque aucun flux n'est agrégé ($e500v_{e50}$)

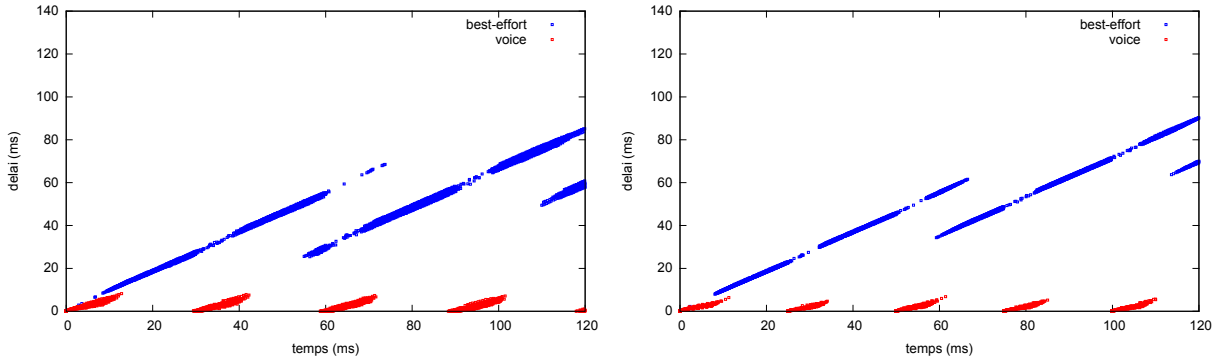
3.4.2.d Avec un flux de 1500 octets / 150 μs ($e1500v_{e150}$)

Bien que les comportements du délai représentés soient encore une fois similaires, une différence apparaît clairement dans le pourcentage de réduction des délais vis à vis du cas sans agrégation faisant office de référence. Lorsque les deux flux sont agrégés (figure 3.66) le gain apporté est de 80 % au lieu des 95 % atteints auparavant pour le flux *best-effort*. Quand au flux *voice* ses performances descendent à 36 % d'amélioration. L'explication donnée lors de l'étude de l'arriéré s'applique donc également pour le délai. Ensuite, le seuil de taille a une influence significative puisqu'il permet au flux *voice* de réduire de 60 % les délais subis par ses paquets (au lieu des 36 % évoqués). Lorsqu'ils sont agrégés seuls, les délais des flux



(a) Au seuil bas (b) Au seuil haut

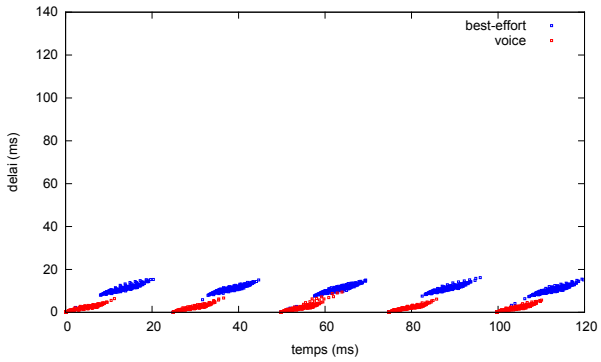
Figure 3.60 – Évolution du délai lorsque le flux *best-effort* est agrégé ($e500v_{e50}$)



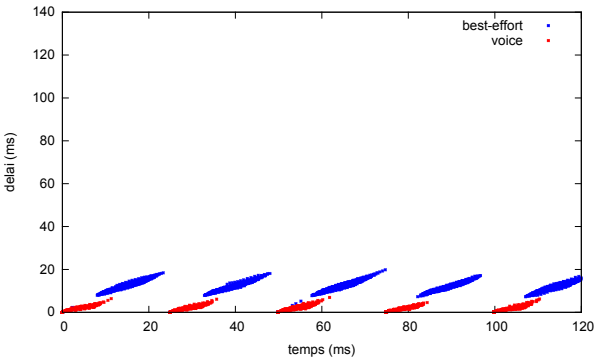
(a) Au seuil bas (b) Au seuil haut

Figure 3.61 – Évolution du délai lorsque le flux *voice* est agrégé ($e500v_{e50}$)

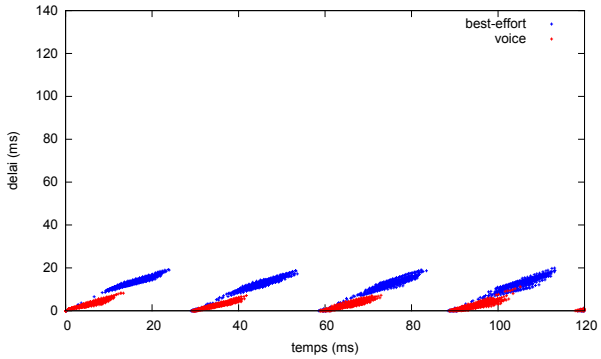
best-effort et *voice* peuvent être respectivement réduits de 14 et 19 % en utilisant plutôt le seuil haut. Ces résultats s'expliquent encore une fois par l'augmentation de la taille des paquets en entrée du système. Sur tous les cas considérés les effets des différentes configurations sur le délai et l'arriéré sont semblables ce qui nous permet d'éditer les recommandations posées auparavant en indiquant leur effet sur le délai.



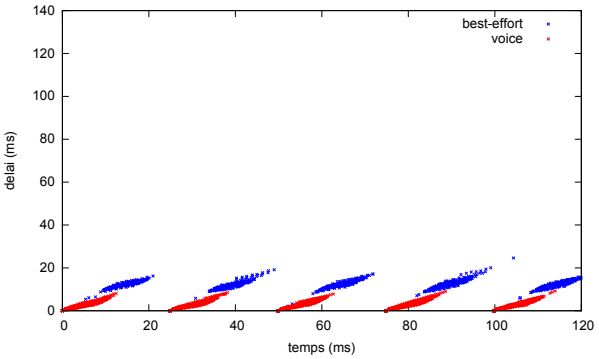
(a) Deux flux au seuil haut



(b) Flux *voice* au seuil haut



(c) Deux flux au seuil bas



(d) Flux *best-effort* au seuil haut

Figure 3.62 – Évolution du délai lorsque les deux flux sont agrégés ($e500v_{e50}$)

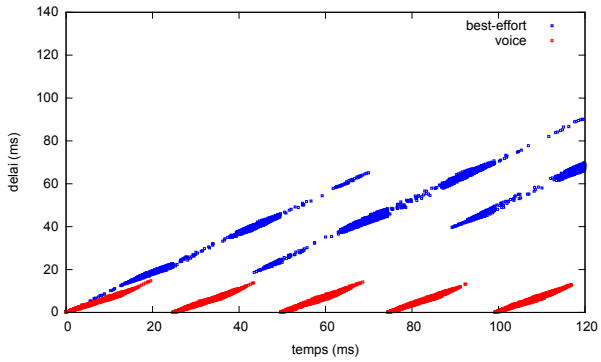
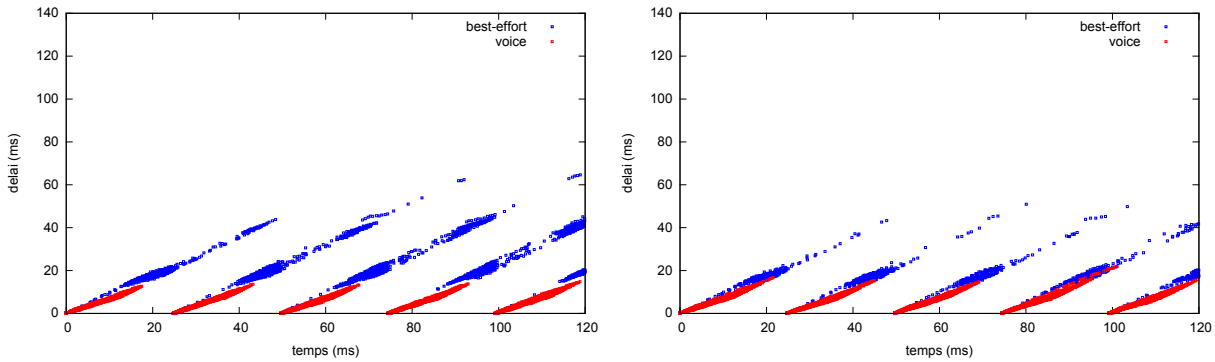
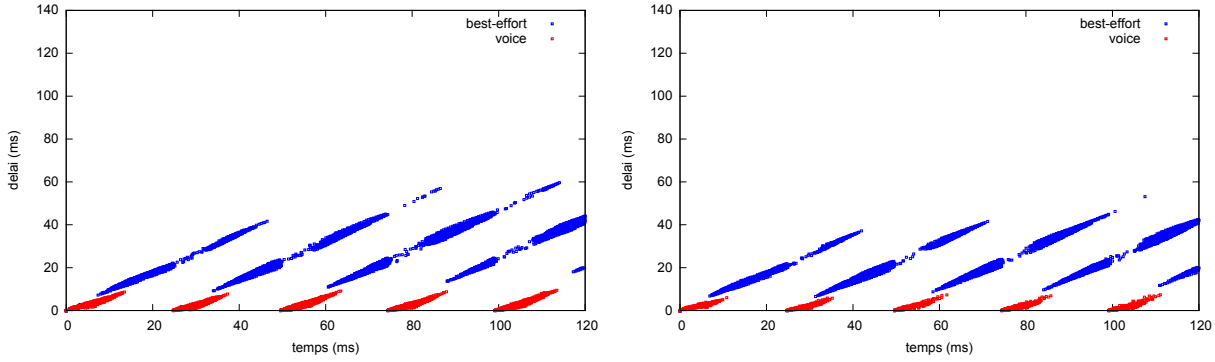


Figure 3.63 – Évolution du délai lorsque aucun flux n'est agrégé ($e1500v_{e150}$)



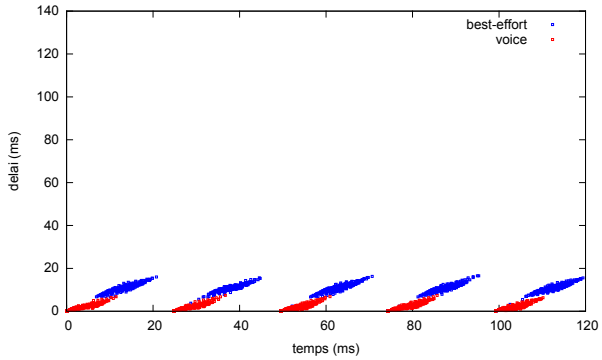
(a) Au seuil bas (b) Au seuil haut

Figure 3.64 – Évolution du délai lorsque le flux *best-effort* est agrégé ($e1500v_{e150}$)

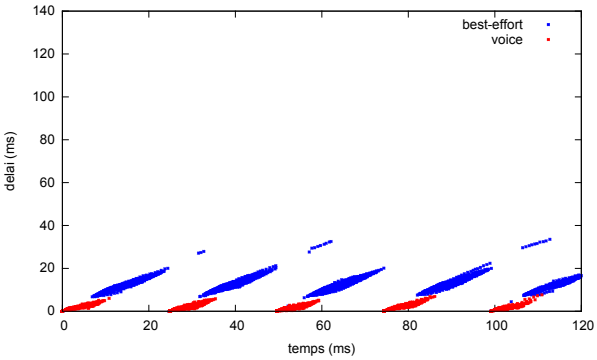


(a) Au seuil bas (b) Au seuil haut

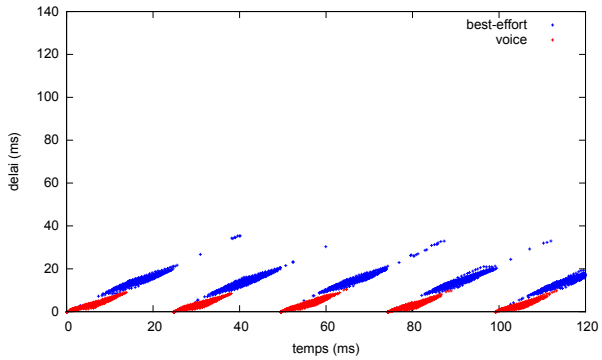
Figure 3.65 – Évolution du délai lorsque le flux *voice* est agrégé ($e1500v_{e150}$)



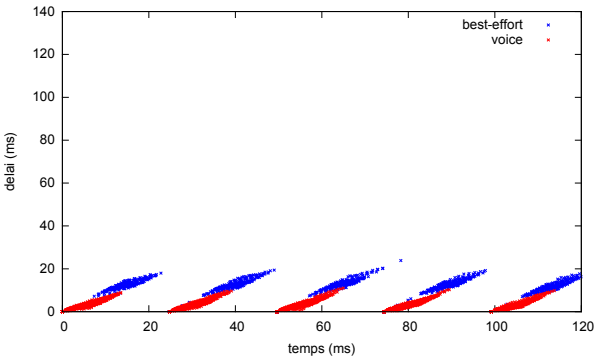
(a) Deux flux au seuil haut



(b) Flux *voice* au seuil haut



(c) Deux flux au seuil bas



(d) Flux *best-effort* au seuil haut

Figure 3.66 – Évolution du délai lorsque les deux flux sont agrégés ($e1500v_{e150}$)

Recommandation 1 : *Lorsqu'un flux de classe inférieure est dans une situation proche de la famine OU lorsque les rafales des flux en concurrence pour le medium sont synchronisées (par définition de l'application par exemple), les flux de classes supérieures doivent être agrégés en priorité pour réduire l'arriéré de traitement maximal et le délai moyen.*

Recommandation 2 : *S'il n'est pas souhaitable d'agréger les flux concurrents de classes supérieures (environnement bruité, incompatibilité technologique, ...) ou si les délais et arriérés mesurés ne sont toujours pas satisfaisants, le flux considéré doit être agrégé pour réduire son arriéré de traitement maximal et son délai moyen. Cette mesure peut être inefficace si les flux de classes supérieures saturent le système.*

Recommandation 3 : *Lorsque les paquets sont de petite taille (moins de 500 octets) il est inutile de modifier le seuil de taille des agrégats pour réduire l'arriéré de traitement maximal et les délais.*

Recommandation 4 : *Lorsque l'arriéré de traitement d'un flux de classe supérieure augmente jusqu'à atteindre la limite mémoire, l'agrégation doit être désactivée sur les flux de classe inférieure pour réduire l'arriéré de traitement et le délai moyen du flux de classe supérieure.*

Recommandation 5 : *Lorsque les paquets sont de grande taille (1500 octets et plus) les flux doivent être agrégés au seuil maximal pour réduire l'arriéré de traitement maximal et les délais.*

3.4.3 Sur les pertes de paquets

Jusqu'ici seuls l'arriéré de traitement et les délais moyen et maximum ont été considérés. Or une quantification des pertes dues aux débordements des mémoires peut compléter les résultats obtenus précédemment. Une fois la taille mémoire maximale atteinte, l'arriéré n'indique pas cette valeur et même si certains paquets arrivant trop tard peuvent être considérés comme perdus par l'application. Nous ne pouvons pas conclure à ce sujet puisque nous n'avons pas encore posé d'exigences.

La section suivante étudie donc l'impact de l'agrégation et du choix du seuil de taille sur les pertes de paquets suivant les mêmes profils de trafic que précédemment. Pour représenter les pertes de paquets nous avons choisi de les sommer sur chaque tirage et de les tracer en fonction du délai maximum subi par les paquets lors de ce tirage. Autrement dit, chaque point sur ces courbes correspond à un tirage pour un flux et une configuration du système. Il a pour abscisse le délai maximum subi par les paquets et pour ordonnée le nombre de paquets perdus sur la durée de la simulation pour un tirage.

Cependant, comme certains flux ne sont pas assez importants pour produire de telles pertes, nous étudierons uniquement le délai maximal pour les flux $e125\upsilon_{e40}$ et $e1500\upsilon_{e150}$. Concernant les graphes sans pertes, le délai maximal est en abscisse et l'ordonnée est arbitraire et ne sert qu'à départager les flux sur la figure.

3.4.3.a Avec un flux de 125 octets / 40 μs ($e125\upsilon_{e40}$)

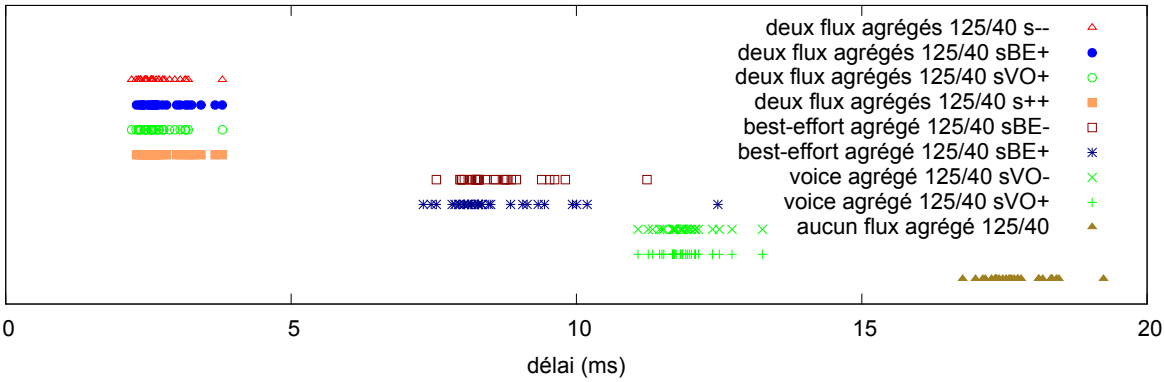


Figure 3.67 – Délai maximum pour le flux *best-effort* ($e125\upsilon_{e40}$)

Les caractéristiques du flux choisi ici permettent d'observer l'influence de l'agrégation lorsque le système n'est pas saturé. Les deux indicateurs choisis dans cette partie sont le délai maximum de traversée de la couche MAC d'une part et la somme des pertes de paquets d'autre part. Les deux sont tracés pour chaque tirage. Ces dernières étant nulles (système non saturé et aucune perte de propagation) les tracés qui suivent montreront uniquement les délais. En figure 3.67, on retrouve les neuf scénarios listés dans l'introduction de cette section. Pour chacun de ces scénarios les délais maximum de 30 tirages sont relevés.

Débutons par l'étude du flux *best-effort*. Agréger ce dernier ainsi que son concurrent permet en moyenne de diviser ses délais par six. Agréger seulement ce flux les divise par deux alors qu'agréger son concurrent les divise par un et demi. Agréger le flux *voice* améliore donc significativement les performances obtenues par le flux *best-effort* considéré ici. Il est cependant plus efficace d'agréger ce dernier. Enfin, le changement pour un seuil de taille presque deux fois supérieur ne change pas significativement le délai maximal mesuré.

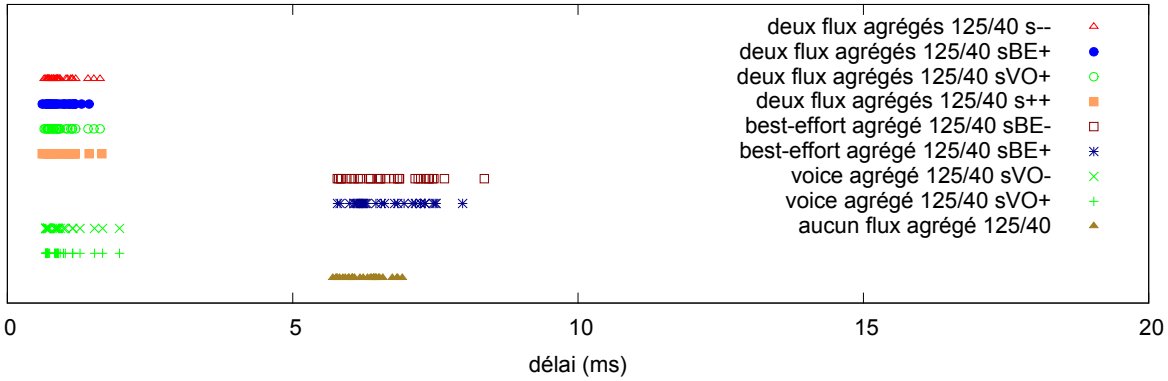


Figure 3.68 – Délai maximal pour le flux *voice* ($e125\upsilon_{e40}$)

Nous pouvons faire des observations identiques concernant le flux *voice* (délais en figure 3.68) et on retrouve sur le délai maximal l'observation établie en observant le délai moyen : les délais du flux *voice* sont plus important lorsque le flux *best-effort* est agrégé que sans agrégation (pour une augmentation de (7 % à 40 %). L'influence d'un changement de seuil est ici négligeable comme vu lors des autres études.

3.4.3.b Avec un flux de 250 octets / 25 μs ($e250\upsilon_{e25}$)

Commençons avec un flux de 250 octets / 25 μs et observons les résultats obtenus par le flux *best-effort*. Ce dernier subit des pertes dans toutes les configurations d'agrégation excepté celles où deux flux sont agrégés. (figure 3.69).

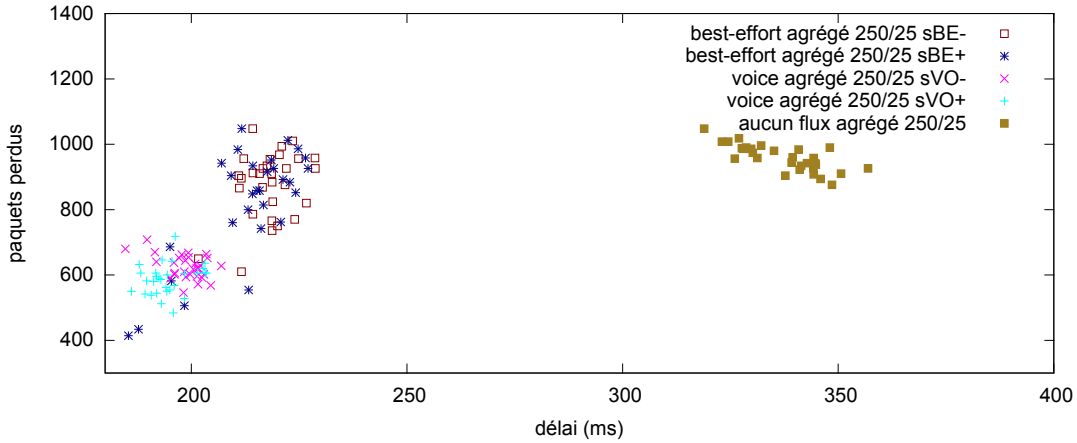


Figure 3.69 – Délai maximum, flux *best-effort*, $e250\upsilon_{e25}$ (avec pertes)

À quelques rares exceptions, il est plus avantageux ici d'agréger le flux *voice* (voir figure 3.69). En effet, cela permet de réduire le délai maximal de 9 % en moyenne et jusque 14 % (moyenne réalisée sur les tirages). Les pertes sont quand à elles réduites de 28 % en moyenne. On notera tout de même en figure 3.69 quelques tirages dans lesquels les performances sont bien meilleures en cas d'agrégation du flux *best-effort* en particulier lorsque le seuil de taille est au maximum. Excepté pour ces cas, le changement pour un seuil plus grand n'offre pas d'amélioration significative lorsqu'on agrège le flux *best-effort*. Lorsqu'on agrège son concurrent, utiliser le seuil haut modifie le nombre de paquets perdus. Les pertes sont très mitigées, 6 % en

moyenne mais la plage de valeur évolue d'une détérioration de 10 % à une amélioration de 20 % en fonction des tirages. L'évolution des délais offre un constat identique, 2 % d'amélioration pour une plage entre -1,5 % et 4 %.

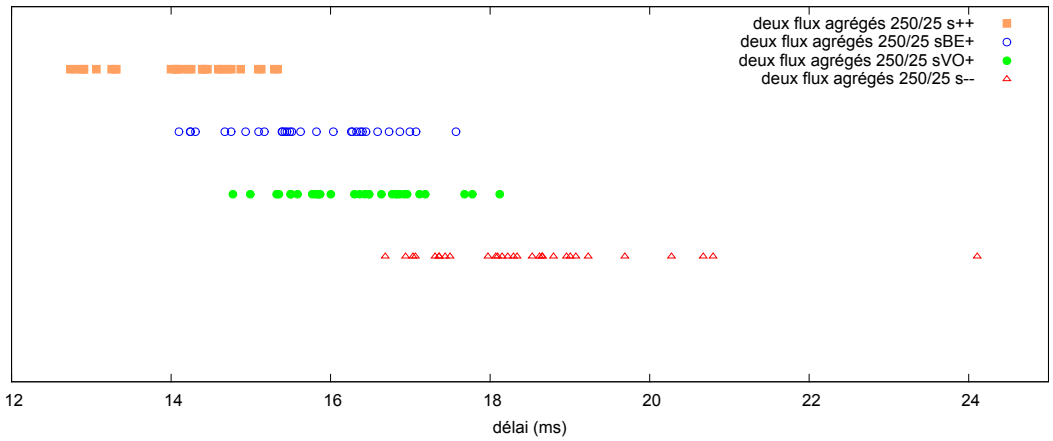


Figure 3.70 – Délai maximum, flux *best-effort*, $e250v_{e25}$ (sans pertes)

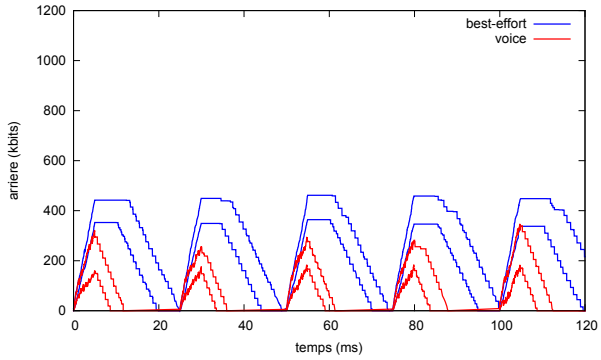


Figure 3.71 – Arriéré deux flux agrégés $e250v_{e25}$

Concentrons nous maintenant sur les scénarios sans pertes où les deux flux sont agrégés (figure 3.70). Au delà de l'abolition pure et simple des pertes, agréger ces deux flux réduit les délais de 95 % en moyenne. Cependant, le système reste légèrement saturé comme le montre la figure 3.71 où un léger arriéré de traitement subsiste au fil des rafales pour le flux *best-effort*. Si la simulation se prolongeait, il y aurait des pertes dues aux débordements des mémoires.

On remarque ici une influence significative du seuil de taille. Le classement des délais est le suivant (les améliorations sont calculées par rapport au cas dans lequel les deux seuils bas sont utilisés) :

1. utiliser le seuil haut pour les deux flux réduit le délai maximal de 23,6 % en moyenne (de 10,8 % dans le pire cas et jusque 46,5 % dans le meilleur).
2. utiliser le seuil haut sur le flux considéré (ici *best-effort*) réduit le délai maximal de 15 % (d'une détérioration de 3 % à une amélioration de 32 %).
3. utiliser le seuil haut sur le flux concurrent (ici *voice*) avec une réduction de 11,5 % en moyenne (de -9 % à 30 %).
4. utiliser les deux seuils bas

Comme toujours, même si seul le flux *voice* augmente la taille de ses agrégats le flux *best-effort* que nous observons ici verra ses délais réduits. Enfin, contrairement aux scénarios avec pertes il est préférable ici d'agréger le flux *best-effort* considéré plutôt que son concurrent.

Observons maintenant les résultats de simulation du flux *voice* (figure 3.72). Les nuages de points sont fortement entremêlés. Néanmoins, agréger le flux *best-effort* augmente les délais du flux *voice* de 3 % en moyenne et jusque 24,6 % au seuil minimal. De plus, l'augmentation au seuil maximal pour le flux *best-effort* empire encore les performances du flux *voice* de 8,8 % en moyenne et jusque 29,4 % dans le pire cas. Ce comportement se reproduit sur quelques tirages pour les configurations d'agrégation permettant d'éviter les pertes (figure 3.73) ce qui confirme qu'agréger le flux *best-effort* peut augmenter les délais du flux *voice* même si celui-ci est prioritaire et agrégé.

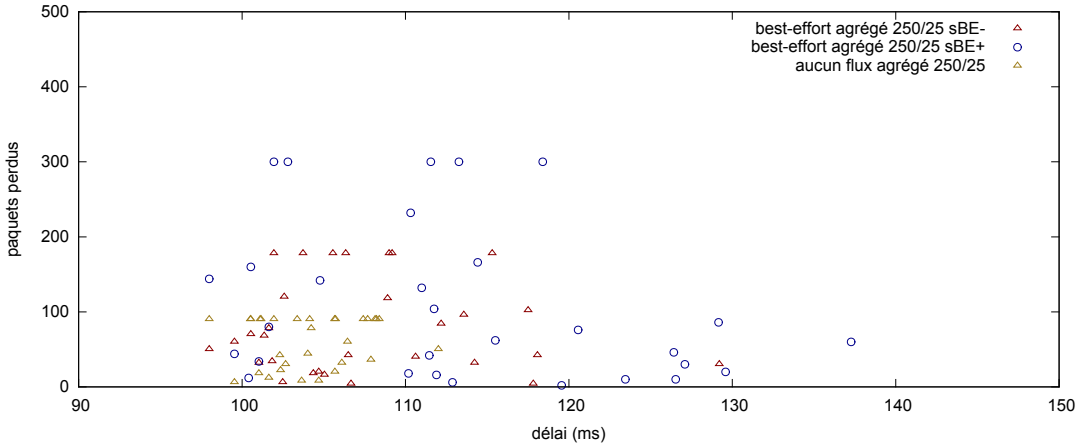


Figure 3.72 – Délai maximum, flux *voice*, $e250v_{e25}$ (avec pertes)

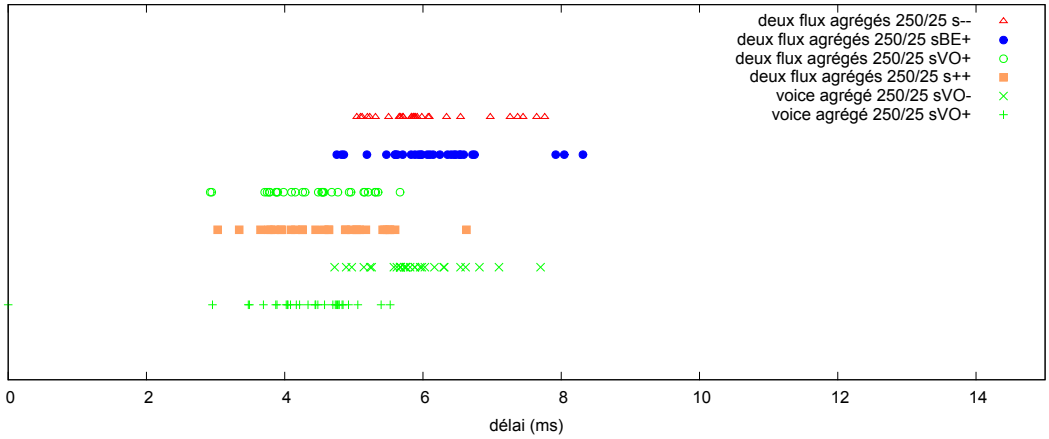


Figure 3.73 – Délai maximum, flux *voice*, $e250v_{e25}$ (sans pertes)

Jusqu'ici les délais, l'arriéré et les pertes subies pas le flux *voice* ont quelquefois été influencés par la configuration du système pour le flux *best-effort* et nous en voyons un nouvel exemple ici. L'influence du flux *best-effort* est plus importante lorsque le flux *voice* n'est pas agrégé. Nous avons vu que l'agrégation du flux *voice* peut le protéger dans une certaine mesure de l'influence du flux *best-effort* (figure 3.73) mais ici les résultats sur le délai maximal sont mitigés. En fonction des configurations, le flux *best-effort* peut

détériorer les délais du flux *voice* jusque 42,5 %. Une bonne protection à cela est d'utiliser le seuil haut sur le flux *voice*.

Avec un peu de recul sur la figure 3.67 on remarque que lorsque seul le flux *best-effort* est agrégé les pertes ne sont pas fortement réduites contrairement au délai maximal. C'est seulement lorsque le flux *voice* est agrégé que les pertes sont réduites (le délai maximal l'est de nouveau néanmoins). Enfin agréger les deux flux permet d'éviter les pertes et de réduire très fortement le délai maximal subi par les paquets (divisé par 10).

3.4.3.c Avec un flux de 500 octets / 50 μs ($e500v_{e50}$)

Concernant les configurations du système d'agrégation produisant des pertes (figure 3.74), agréger seulement le flux *voice* plutôt que le flux *best-effort* offre toujours de meilleures performances. Les délais sont réduits de 9,7 % en moyenne et 17,9 % au maximum avec le seuil bas. L'amélioration est d'autant plus significative que le seuil haut est utilisé : 16,6 % en moyenne et jusqu'à 23,8 %. Le nombre de paquets perdus est fortement réduit en adoptant ces mesures d'agrégation :

— pour le seuil bas : une réduction de 67,7 % en moyenne (de 33,6 à 93,5 %).

— pour le seuil haut : une réduction de 63,8 % en moyenne (de 21 à 80,7 %).

Contrairement au cas précédent, choisir le seuil haut limite les bénéfices apportés au flux *best-effort* par l'agrégation. Même si augmenter la taille du seuil peut dans quelques cas réduire les pertes de 30 %, celles-ci sont multipliées par 2,5 en moyenne et par 11 dans le pire cas. Le délai maximum est également augmenté de 7 % en moyenne et jusque 33,5 %. Le choix du seuil influence significativement les résultats obtenus et un seuil plus important signifie ici des délais maximum et des pertes plus importante.

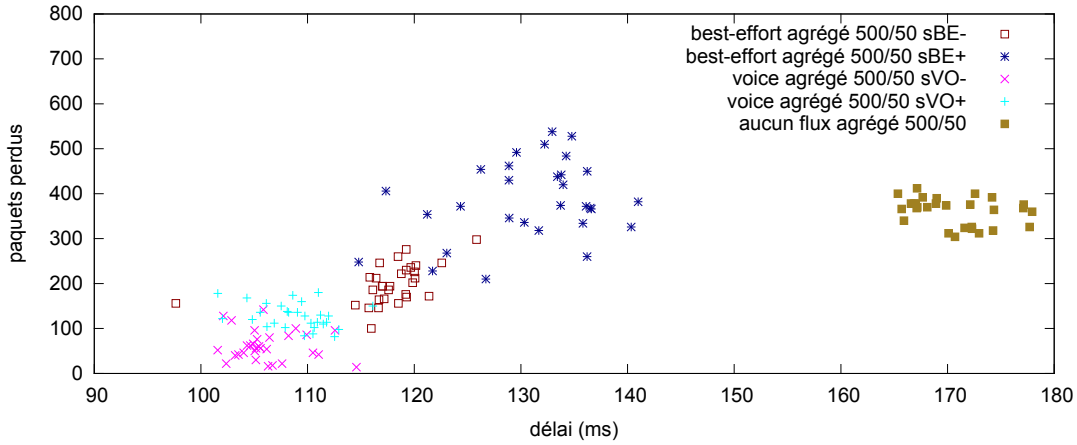


Figure 3.74 – Délai maximum, flux *best-effort*, $e500v_{e50}$ (avec pertes)

Concernant les scénarios sans pertes (figure 3.75) on retrouve un comportement similaire à celui observé pour le flux précédent. Voici le classement des configurations de seuils en fonction des délais maximaux mesurés :

1. utiliser les deux seuils hauts avec une amélioration de 21,8 % en moyenne (de 4 % à 37,9 %).
2. utiliser le seuil haut sur le flux considéré (ici *best-effort*) avec une amélioration de 10 % (de -41,6 % à 26 %).

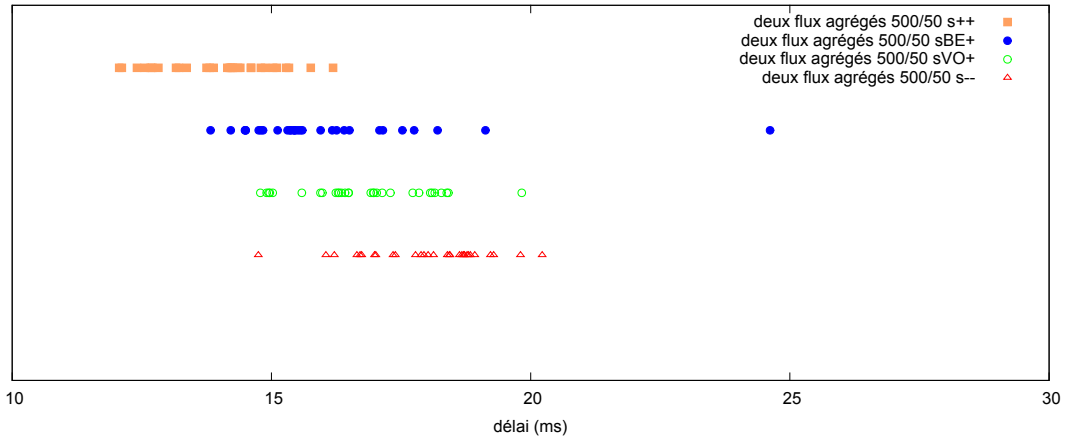


Figure 3.75 – Délai maximum, flux *best-effort*, $e500v_{e50}$ (sans pertes)

3. utiliser le seuil haut sur le flux concurrent (ici *voice*) avec une amélioration de 6,2 % en moyenne (de -14,3 % à 21,9 %).
4. utiliser les deux seuils bas.

On notera qu'agréger le flux *best-effort* est plus risqué qu'agréger son concurrent puisqu'il est possible d'augmenter le délai maximal de 41,6% dans le pire cas.

D'autre part, les gains apportés par l'agrégation sont ici moindres comparé au flux de 250 octets / 25 μs considéré précédemment. Même si les flux génèrent le même taux d'arrivée, la différence entre la taille moyenne des paquets reçus par le système influence le degré des bénéfices de l'agrégation.

Des paquets de petite taille arrivant à de courts intervalles augmenteront les chances d'agréger et permettront un gain beaucoup plus important au niveau des en-têtes que de gros paquets. Les délais subis par les paquets même sans agrégation sont bien plus importants avec un flux de 250 octets / 25 μs qu'avec un flux de 500 octets / 50 μs (en moyenne 337,24 et 171,43 ms respectivement). Enfin, des paquets de plus petite taille facilitent l'optimisation du remplissage d'un agrégat.

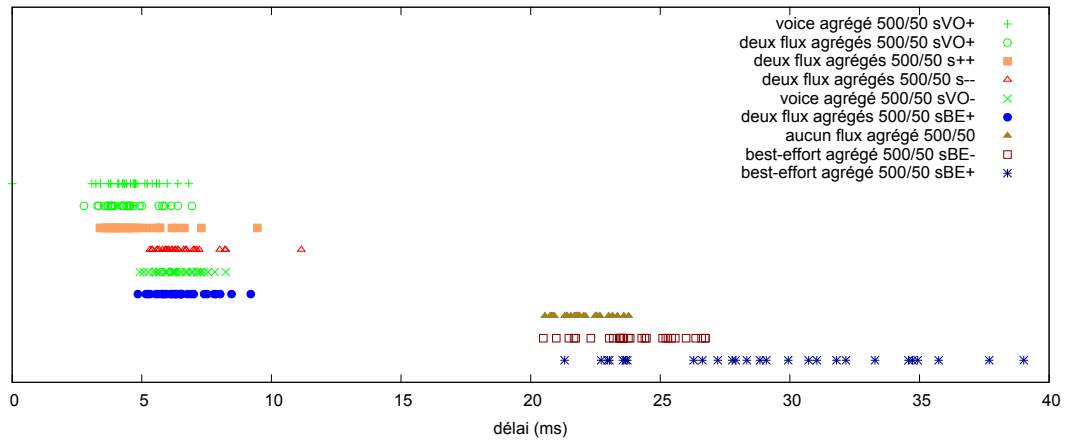


Figure 3.76 – Délai maximum, flux *voice*, $e500v_{e50}$ (sans pertes)

Observons les résultats mesurés pour le flux *voice*. Celui-ci ne souffre d'aucune perte dans cette configuration. Nous tracerons donc uniquement la figure 3.76 qui illustre les délais. Comme ce flux est prioritaire

il est sensé être moins sensible aux changements de configuration de son concurrent (agrégation ou pas, changement de seuil). On observe cependant des variations non négligeables en particulier lorsqu'on compare le scénario sans agrégation et les scénarios ou le flux *best-effort* est agrégé. Comme pour les cas précédents, le flux *voice* voit ses performances diminuées lorsque son concurrent est agrégé, notamment avec le seuil d'agrégation maximal (-39,5 % en moyenne et jusque -107 %).

Cependant, lorsque le flux *voice* est agrégé, l'influence des modifications de configuration du système pour le flux *best-effort* est fortement réduite. En effet, comparé au cas avec seul le flux *voice* agrégé, agréger le flux *best-effort* réduit les délais de 3,8 % en moyenne. Comparé à ce dernier cas, augmenter le seuil de taille pour le flux *best-effort* réduit les délais de 2 % en moyenne. Dans les autres cas, une augmentation du seuil de taille pour le flux *voice* permet d'améliorer les délais de 30 % en moyenne et jusque 60 % dans le meilleur cas. Comme dans le cas précédent, en figure 3.74 on note que le délai maximal est d'abord réduit avant que les pertes le soient par l'utilisation des différentes configurations de l'agrégation.

3.4.3.d Avec un flux de 1500 octets / 150 μs ($e1500v_{e150}$)

Comme dans les cas précédents, le flux *best-effort* (figure 3.77) obtient les meilleures performances lorsque les deux flux sont agrégés (70 % de moins sur les délai maximal comparé au cas sans agrégation). S'il est impossible d'agréger les deux flux (environnement trop bruité, incompatibilité technologique, ...) le plus rentable pour ce flux sera encore une fois d'être agrégé (délai réduit de 46 % en moyenne) même si agréger le flux concurrent apporte des performances similaires (38 % d'amélioration). Dans chacun de ces cas augmenter la taille des agrégats réduit les délais de 14 % si appliqué sur le flux *voice* et de 25 % si appliqué sur le flux *best-effort* étudié ici. Ces résultats sont similaires à ceux obtenus avec les autres profils de trafic dans les cas sans pertes.

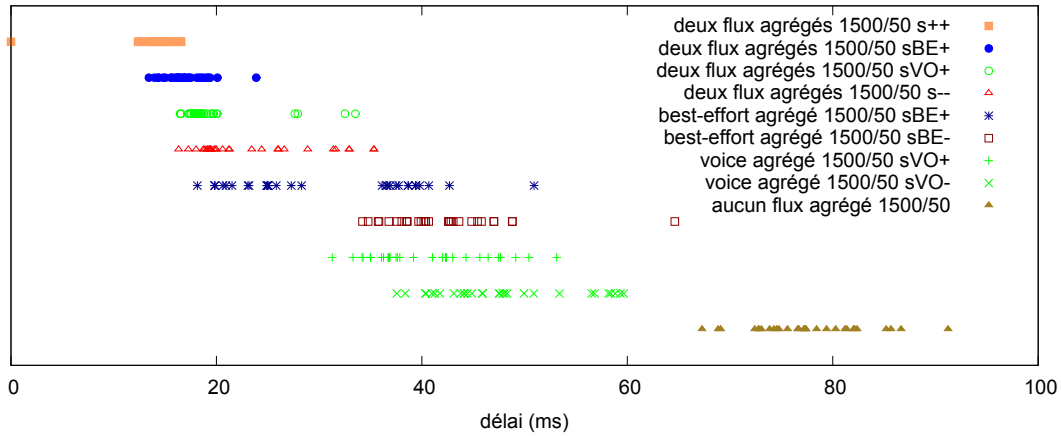


Figure 3.77 – Délai maximum, flux *best-effort*, $e1500v_{e150}$ (sans pertes)

Concernant le flux *voice* (figure 3.78), il est toujours préférable de l'agréger au seuil maximal. D'ailleurs si le flux *best-effort* n'est pas agrégé au seuil maximal il est possible de gagner entre 3 et 5 % supplémentaires. Ensuite viennent tous les cas où le flux *voice* est agrégé, peu importe la configuration de son concurrent les différences ne sont pas significatives (moins de 1 % d'écart). Enfin le délai maximum subi par le flux *voice* est augmenté de 10 % en moyenne si le flux *best-effort* est agrégé (et que le flux *voice* ne l'est pas). Le délai pour le flux *voice* est également augmenté de 10 % lorsque le flux *best-effort* utilise le seuil haut au

lieu du seuil bas et ce même si le flux *voice* est agrégé.

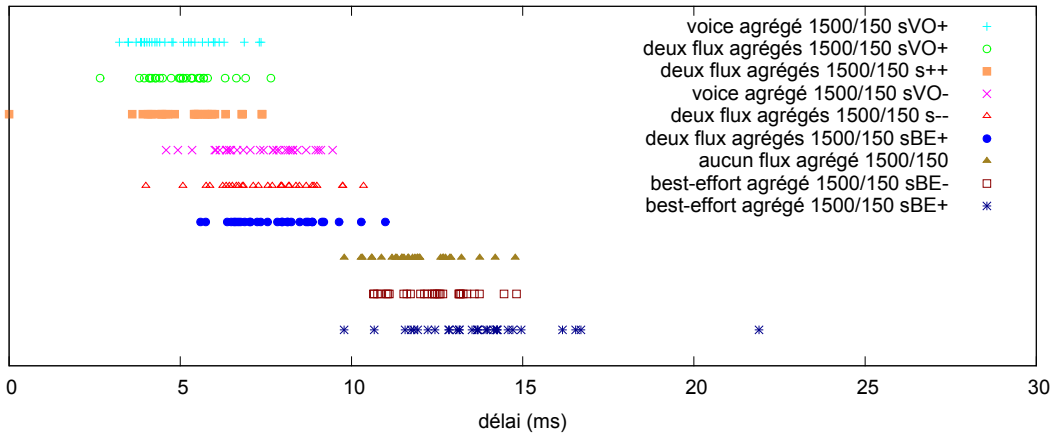


Figure 3.78 – Délai maximum, flux *voice*, $e1500v_{e150}$ (sans pertes)

À débit constant, l'augmentation de la taille des paquets réduit délais et pertes (si l'on compare les cas sans agrégation) mais également les bénéfices apportés par l'agrégation. Les résultats obtenus ici confirment ceux de l'étude du flux de 500 octets / 50 μs dans laquelle agréger améliorait les délais et pertes dans une moindre mesure comparée à l'étude avec un flux de 250 octets / 25 μs. Nous pouvons encore une fois conclure que le bénéfice de l'agrégation diminue avec l'augmentation de la taille des paquets.

3.4.3.e Avec un flux best-effort de 1500 octets / 50 μs ($e1500v_{e50}$) et un flux voice de 500 octets / 50 μs ($e500v_{e50}$)

Commençons par affecter le flux le plus important à la classe *best-effort* moins prioritaire. Pour ce flux (figure 3.79) des pertes se produisent dans chaque scénario. L'amélioration offerte par l'agrégation de ce flux est très inférieure à celle obtenue lorsqu'on agrège son concurrent (9,5 % contre 53,4 % en moyenne). Augmenter le seuil permet d'améliorer encore les performances de 2 à 5 %. Les délais et pertes les plus faibles sont obtenus en agrégeant les deux flux comme toujours. Dans ces scénarios il est plus avantageux d'agréger le flux *best-effort* au seuil maximal plutôt que son concurrent, 18 % et 9 % de réduction respectivement. Cependant et comme pour les autres flux considérés, les meilleures performances sont obtenues avec agrégation sur les deux flux et au seuil maximal.

Le flux *voice* (figure 3.80) ne subit ici aucune perte. Nous obtenons des résultats très proches (éloignés de quelques %) de ceux obtenus précédemment lorsque ce flux *voice* avait les même caractéristiques mais que le flux *best-effort* transmettait des paquets trois fois moins importants (500 octets). Comme il est prioritaire, le flux *voice* sera peu sensible aux changements de caractéristiques du flux *best-effort*.

Encore une fois, le délai maximum est le premier à être réduit (figure 3.77) avant que les pertes soient diminuées par l'utilisation de l'agrégation.

3.4.3.f Avec un flux best-effort de 500 octets / 50 μs et un flux voice de 1500 octets / 50 μs

Affectons maintenant le flux de 1500 octets / 50 μs à la classe *voice*. Nous commençons par étudier les performances obtenues dans cette configuration par le flux *best-effort* (figure 3.81). Comme pour les autres

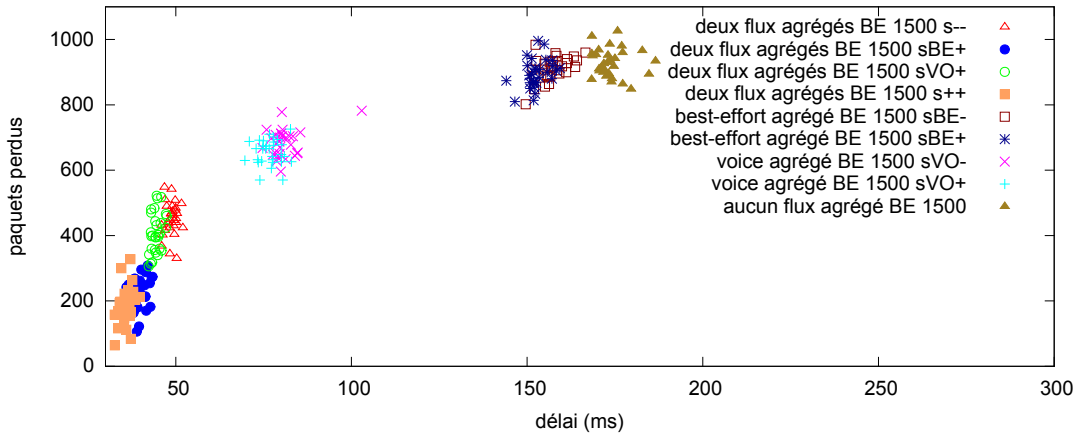


Figure 3.79 – Délai maximum, flux *best-effort*, flux différents (*best-effort* le plus important)

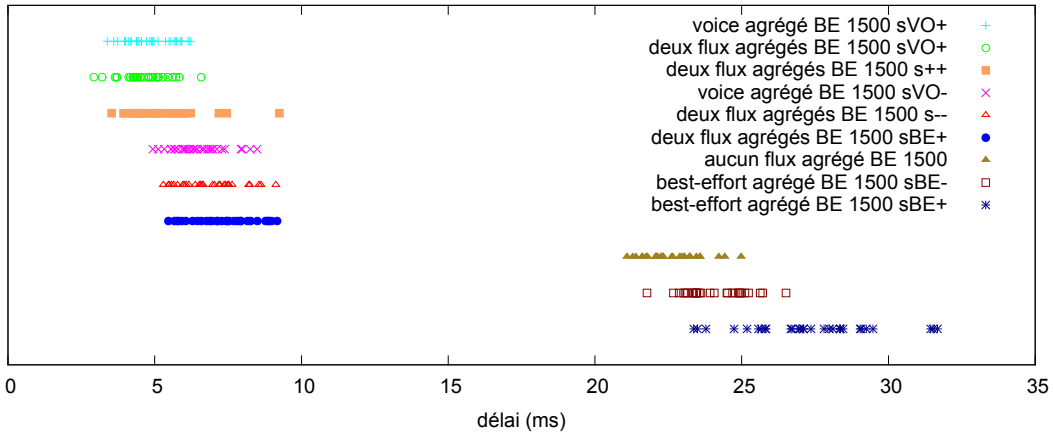


Figure 3.80 – Délai maximum, flux *voice*, flux différents (*best-effort* le plus important)

cas, les meilleures performances sont obtenues avec les deux flux agrégés. Si cette configuration est impossible, agréger seulement le flux *voice* semble moins performant qu'agréger seulement le flux *best-effort*(gain de 10 % contre 16 % respectivement). Cependant l'augmentation du seuil provoque une amélioration notable des performances lorsque le flux *voice* est agrégé (gain de 22 % comparé au cas sans agrégation) contrairement à l'autre configuration (gain de 20 % comparé au cas sans agrégation). Ces chiffres sont parfaitement illustrés en figure 3.81 malgré quelques tirages exceptionnels lors de l'agrégation du flux *best-effort*.

Le flux *voice* (figure 3.82) subit également des pertes dans presque tous les cas. Agréger uniquement son concurrent détériore toujours ses performances (entre 3 et 6 % ici). Les délais et pertes sont les plus faibles lorsque ce flux est agrégé avec le seuil maximal. Comme précédemment moins les trames transmises par le concurrent sont importantes, meilleures sont les performances pour le flux *voice* (gains calculés par rapport au cas ou les deux flux sont agrégés aux seuils maximaux) :

— gain de 2,7 % et jusque 14,7 % si le flux concurrent est agrégé mais pas au seuil maximal.

— gain de 3,6 % et jusque 15,5 % si le flux concurrent n'est pas agrégé.

Contrairement à son concurrent, le flux prioritaire bénéficie de meilleures performances lorsqu'il est seul à agréger au seuil maximal. Il est même possible d'améliorer très légèrement ses performances en agrégeant pas du tout le flux *best-effort*.

Entre les extrêmes cités ci-dessus on retrouve les scénarios ou le flux *voice* est agrégé avec le seuil le

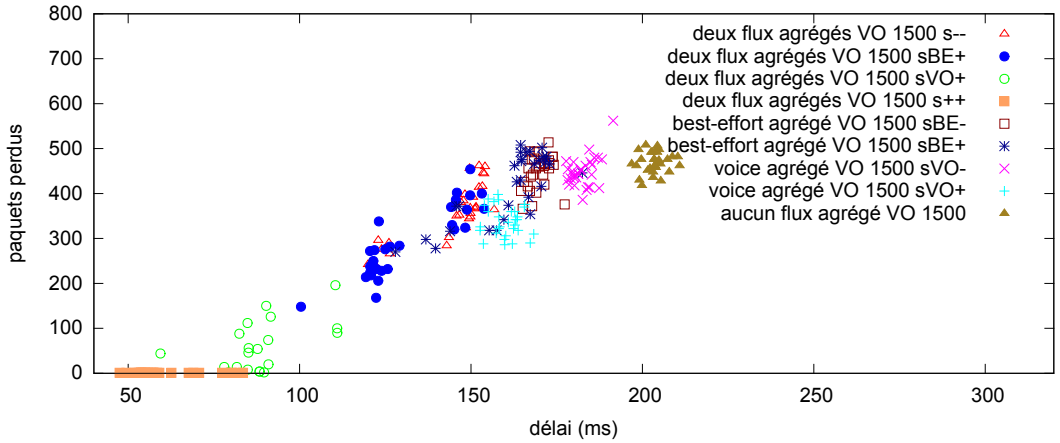


Figure 3.81 – Délai maximum, flux *best-effort*, flux différents (*voice* le plus important)

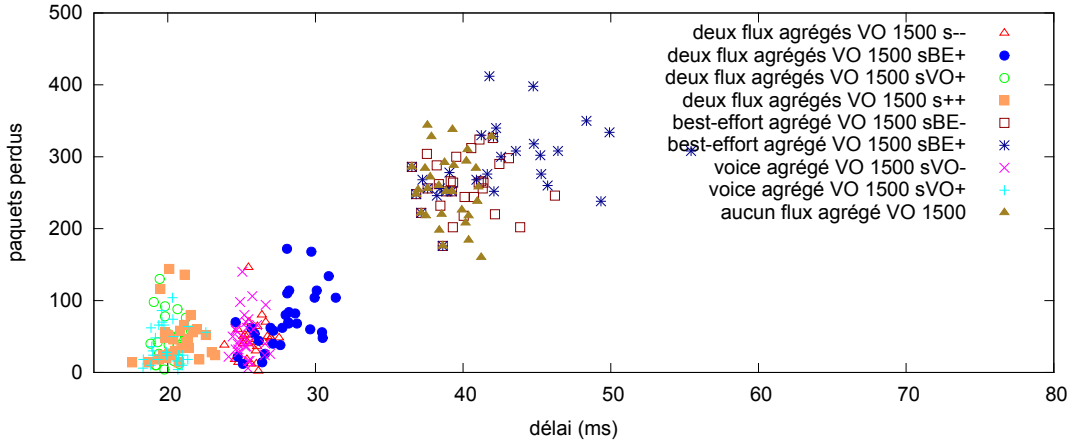


Figure 3.82 – Délai maximum, flux *voice*, flux différents (*voice* le plus important)

plus bas. Dans ces scénarios les performances seront dégradées si le flux *best-effort* agrège au seuil maximal (de 8.7 % en moyenne et jusque 20,8 %).

Cette étude des pertes et du délai maximal subis par les paquets nous permet de confirmer toutes les observations et recommandations proposées auparavant. Elle nous a permis d'observer que lorsque on utilise l'agrégation dans ses différentes configurations, le délai maximal est le premier indicateur à être amélioré, ensuite suivent les pertes de paquets, puis l'arriéré maximal et le délai moyen (confirmé figure 3.79 à nouveau).

3.4.4 Discussion

Cette section résume les différentes observations, les résultats chiffrés ainsi que les recommandations tirés des simulations réalisées. Une discussion sera également menée autour de ces résultats et de leurs implications vis à vis de l'intérêt du client et du fournisseur d'accès.

Commençons par les résultats chiffrés des modifications apportées sur nos indicateurs par les différentes configurations d'agrégation. Ces résultats sont exprimés en pourcentage de réduction pour l'arriéré de traitement maximal (tableau 3.5) et moyen (tableau 3.6) et pour le délai maximal (tableau 3.7) et moyen (tableau 3.8). Chaque pourcentage est calculé en fonction de l'écart avec le cas sans agrégation. Notons

qu'une valeur négative traduit une augmentation de l'arriéré ou du délai.

		best-effort agrégé		**voice agrégé**		**best-effort et voice agrégés**			
		s_B	s_H	s_B	s_H	be=s_B vo=s_B	be=s_H vo=s_B	be=s_B vo=s_H	be=s_H vo=s_H
best-effort	$e125_{e40}$	10,6	14,6	5,1	5,1	3,8	11,7	3,8	11,7
	$e250_{e25}$	0,0	0,0	0,0	0,0	54,9	53,0	56,2	51,3
	$e500_{e50}$	0,0	0,0	0,0	0,0	50,0	49,1	48,2	51,9
	$e1500_{e150}$	8,3	23,4	12,2	15,8	41,7	44,3	31,9	45,2
	be=$e1500_{e50}$ vo=$e500_{e50}$	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0
	be=$e500_{e50}$ vo=$e1500_{e50}$	0,0	0,0	0,0	0,0	0,0	0,0	0,0	2,6
voice	$e125_{e40}$	-9,6	-9,6	-7,8	-7,8	-13,9	-9,5	-13,9	-9,5
	$e250_{e25}$	0,0	0,0	67,4	66,8	66,1	68,7	65,7	66,6
	$e500_{e50}$	-0,6	-44,1	32,7	30,5	29,8	17,2	25,5	16,0
	$e1500_{e150}$	7,5	-4,1	10,1	10,6	0,4	-1,2	9,9	-0,4
	vo=$e500_{e50}$ be=$e1500_{e50}$	-2,0	-19,8	11,7	6,4	10,1	7,9	4,4	6,7
	vo=$e1500_{e50}$ be=$e500_{e50}$	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0

Tableau 3.5 – Arriéré de traitement maximal, améliorations (en pourcentage) pour différentes configurations du système d'agrégation

Les valeurs de l'arriéré de traitement maximal (tableau 3.5) permettent de déterminer si le système subira des pertes. Toutes les valeurs nulles indiquent que le système est saturé lorsque aucun flux n'était agrégé (la limite mémoire est atteinte) et agréger ne change pas cet état de fait (amélioration de 0%). Par exemple, le flux *best-effort* est saturé dans presque tous les cas dès qu'un des deux flux considérés vaut $e1500_{e50}$. Seule une configuration avec les deux flux agrégés au seuil haut lorsque le flux *voice* vaut $e1500_{e50}$ permet au flux *best-effort* d'éviter la saturation sur la durée de la simulation. Même dans ce cas, le flux *voice* sera saturé (voir la dernière ligne du tableau 3.5).

Lorsque le flux *voice* suit $e125_{e40}$, on remarque que l'arriéré de traitement maximal de ce flux *voice* augmente dans tous les cas de 7,8 à 13,9 %. Ces résultats ne trouvent aucune explication avec le comportement du système. Ils sont donc dus à des variations des trafics entrants. Cette hypothèse est confirmée par les résultats obtenus pour l'arriéré moyen qui vont à l'encontre de résultats obtenus pour l'arriéré maximal. Comme la valeur maximale de l'arriéré est variable selon les tirages obtenus, nous choisissons d'ignorer ici les améliorations ou détériorations inférieures à 15 % exception faite des valeurs à 0 % qui indiquent clairement que le système était saturé et que la situation n'a pas changé.

Il est également intéressant de noter les valeurs identiques comme lorsque le seuil change et que la valeur obtenue est identique (comme lorsque le flux *best-effort* suit $e125_{e40}$: si le flux *voice* est agrégé peut importe à quel seuil).

Nous pouvons donc maintenant conclure que pour le flux $e125_{e40}$ les améliorations apportées par l'agrégation sur l'arriéré maximal ne sont pas significatives. Dans les autres cas, la classe *best-effort* réduit sont arriéré de traitement en agrégeant uniquement si les deux flux sont agrégés excepté lorsque le flux *voice* est trop gourmand. On note également qu'entre les flux $e250_{e25}$, $e500_{e50}$ et $e1500_{e150}$ les améliorations apportées par l'agrégation diminuent alors que la taille initiale des paquets augmente.

De son côté, la classe *voice* bénéficie d'un arriéré maximal réduit que les deux flux soient agrégés ou qu'elle soit la seule à être agrégée. Cependant les améliorations ne sont significatives que pour les flux $e250_{e25}$

		best-effort agrégé		voice agrégé		best-effort et voice agrégés			
		s_B	s_H	s_B	s_H	be=s_B vo=s_B	be=s_H vo=s_B	be=s_B vo=s_H	be=s_H vo=s_H
best-effort	$e125_{e40}$	-4,2	-14,4	1,3	1,5	-6,1	-21,0	-6,3	-20,2
	$e250_{e25}$	-13,9	-23,3	-10,0	-10,1	60,7	60,0	60,9	60,4
	$e500_{e50}$	-13,2	-19,9	-11,1	-6,2	59,8	60,0	60,0	59,5
	$e1500_{e150}$	42,1	54,7	31,5	38,8	60,9	62,2	60,7	62,4
	be=$e1500_{e50}$ vo=$e500_{e50}$	-8,4	-9,8	-22,5	-22,6	-10,8	6,0	-8,6	9,2
	be=$e500_{e50}$ vo=$e1500_{e50}$	-4,3	-5,2	-2,1	-8,7	-9,4	-13,4	0,4	28,3
voice	$e125_{e40}$	-1,6	-2,1	16,5	16,1	15,2	15,6	15,0	14,0
	$e250_{e25}$	-2,5	-3,6	77,3	78,7	75,8	75,3	76,4	76,2
	$e500_{e50}$	-9,9	-30,7	34,4	41,8	30,2	28,7	30,6	29,8
	$e1500_{e150}$	1,4	-3,1	20,1	23,0	10,1	7,1	11,8	14,3
	vo=$e500_{e50}$ be=$e1500_{e50}$	-2,7	-7,9	30,3	29,7	25,7	26,2	28,7	27,8
	vo=$e1500_{e50}$ be=$e500_{e50}$	-0,6	-1,4	19,4	21,4	19,1	17,8	19,7	19,7

Tableau 3.6 – Arriéré de traitement moyen, améliorations (en pourcentage) pour différentes configurations du système d'agrégation

et $e500_{e50}$. Cela laisse supposer que le flux $e125_{e40}$ est trop faible et que les paquets du flux $e1500_{e150}$ sont trop grands pour que le flux prioritaire forme de grands agrégats.

On note pour le flux $e500_{e50}$ un grand écart dans les performances subies par le flux *voice* lorsque la classe *best-effort* est agrégés au seuil bas ou au seuil haut. L'arriéré de traitement maximal du flux *voice* augmente de plus de 44 % lorsque la classe *best-effort* agrège au seuil haut. Ce phénomène s'explique par une augmentation conséquente de la taille des trames transmises par le flux *best-effort*.

Les résultats obtenus pour l'arriéré de traitement maximal sont confirmés lorsqu'on observe l'arriéré de traitement moyen (tableau 3.6) à quelques exceptions près. Nous avons déjà discuté le cas du flux $e125_{e40}$: la seconde exception concerne les cas dans lesquels l'un des flux est gourmand ($e1500_{e50}$). Dans ces cas on n'observait aucune amélioration de l'arriéré de traitement maximal puisque le système était saturé dans presque tous les cas. Ici on note que l'arriéré moyen sera dans la plupart des cas inchangé ou détérioré pour le flux *best-effort*. Le flux *voice* bénéficiera d'améliorations s'il est agrégé ce qui est identique au comportement observé pour l'arriéré moyen ou maximal pour les profils $e250_{e25}$ et $e500_{e50}$. On note que seule une configuration avec deux flux agrégés au seuil haut permet aux deux flux d'obtenir des améliorations significatives.

Observons maintenant les améliorations apportées par l'agrégation sur le délai. Comme pour l'arriéré nous avons relevé le délai maximal (tableau 3.7) et le délai moyen (tableau 3.8). Que ce soit pour le délai maximal ou moyen, agréger réduit significativement les délais dans tous les cas pour le flux *best-effort*. La situation est plus mitigée pour le flux *voice* qui profite de ces améliorations lorsqu'il est agrégé mais qui subit de légères détériorations lorsque seul le flux *best-effort* l'est. On note que lorsque le flux *best-effort* est agrégé au seuil haut pour le profil $e500_{e50}$ le délai maximal subi par le flux *best-effort* est multiplié par 2. Cette augmentation plus importante que les autres du délai maximal rejoint l'observation faite précédemment concernant l'arriéré de traitement.

Les résultats commentés ici sont détaillés et confirmés par l'étude menée dans les sections précédentes dont les figures et les conclusions associées sont rappelées en tableau 3.9.

		best-effort agrégé		voice agrégé		best-effort et voice agrégés			
		s_B	s_H	s_B	s_H	be=s_B vo=s_B	be=s_H vo=s_B	be=s_B vo=s_H	be=s_H vo=s_H
best-effort	$e125_{e40}$	41,6	35,1	31,1	31,1	80,3	80,3	80,3	80,3
	$e250_{e25}$	35,9	36,4	42,0	43,0	93,2	95,1	94,9	95,7
	$e500_{e50}$	34,9	27,1	40,7	39,9	89,5	87,3	89,7	91,6
	$e1500_{e150}$	29,1	44,2	34,6	41,8	61,2	73,8	63,2	81,9
	be=$e1500_{e50}$ vo=$e500_{e50}$	10,7	14,6	44,8	55,5	72,1	76,7	74,6	78,7
	be=$e500_{e50}$ vo=$e1500_{e50}$	15,9	13,4	9,2	20,2	25,6	27,0	47,4	60,4
voice	$e125_{e40}$	-20,7	-15,3	71,6	71,6	76,5	79,3	76,5	76,1
	$e250_{e25}$	-15,4	-22,6	93,1	95,1	93,1	92,6	94,9	94,1
	$e500_{e50}$	-12,5	-97,9	65,4	71,4	53,1	61,3	70,8	60,2
	$e1500_{e150}$	-0,2	-48,2	36,1	50,2	30,0	25,7	48,3	50,0
	vo=$e500_{e50}$ be=$e1500_{e50}$	-6,1	-26,8	66,0	75,1	63,5	63,3	73,6	63,0
	vo=$e1500_{e50}$ be=$e500_{e50}$	-10,2	-32,2	35,8	46,1	34,3	25,2	48,5	44,6

Tableau 3.7 – Délai maximum, améliorations (en pourcentage) pour différentes configurations du système d'agrégation

		best-effort agrégé		voice agrégé		best-effort et voice agrégés			
		s_B	s_H	s_B	s_H	be=s_B vo=s_B	be=s_H vo=s_B	be=s_B vo=s_H	be=s_H vo=s_H
best-effort	$e125_{e40}$	55,0	62,2	44,5	44,6	85,9	87,4	85,9	87,4
	$e250_{e25}$	34,7	42,0	54,2	55,7	95,2	95,8	95,7	96,3
	$e500_{e50}$	45,0	32,2	50,5	49,4	91,6	91,8	91,6	92,7
	$e1500_{e150}$	53,0	66,5	43,8	54,2	73,4	77,7	76,6	81,0
	be=$e1500_{e50}$ vo=$e500_{e50}$	-16,2	26,2	56,5	59,2	75,3	83,6	78,1	85,8
	be=$e500_{e50}$ vo=$e1500_{e50}$	24,1	33,4	13,6	31,5	42,7	51,9	69,9	79,8
voice	$e125_{e40}$	-4,6	-2,6	85,9	85,9	85,9	85,7	85,9	85,5
	$e250_{e25}$	-5,4	-10,9	95,2	96,9	94,9	94,7	96,3	96,3
	$e500_{e50}$	-7,6	-47,5	76,0	84,1	75,5	72,8	80,9	80,3
	$e1500_{e150}$	-4,3	-12,4	47,1	65,9	39,4	36,2	59,4	57,6
	vo=$e500_{e50}$ be=$e1500_{e50}$	-5,4	-17,4	76,4	89,3	74,2	72,7	83,6	82,4
	vo=$e1500_{e50}$ be=$e500_{e50}$	-2,4	-5,7	49,8	62,5	48,9	45,2	60,6	60,0

Tableau 3.8 – Délai moyen, améliorations (en pourcentage) pour différentes configurations du système d'agrégation

Après avoir étudié plus de 15000 simulations, nous savons qu'agréger permet toujours d'améliorer le débit utile pour le flux qui est agrégé ce qui peut réduire l'arriéré de traitement et donc éviter des pertes de paquets et réduire les délais. Cependant, lorsqu'on observe l'influence d'une telle agrégation sur un concurrent du flux agrégé, les résultats sont bien souvent en sa défaveur. Ainsi, lorsqu'un flux de moindre priorité est agrégé, il transmettra des agrégats en lieu et place de paquets. Au vu de la taille importante d'un agrégat vis à vis d'un paquet, nous comprenons mieux pourquoi certains paquets des flux de plus haute priorité voient leurs délais augmenter.

Pour se figurer ce comportement, imaginez-vous sur une route en voiture. Devant vous une voiture s'insère depuis une route moins prioritaire marquée d'un stop, elle a tout juste le temps de passer et vous fait ralentir. Maintenant remplacez la voiture par un convoi exceptionnel et vous aurez une juste mesure du

Observations	**Figures concernées**
L'agrégation augmente le débit utile.	3.27, 3.28, 3.32, 3.33 et 3.35 à 3.50
Cette augmentation réduit le délai moyen subi par les paquets.	3.29, 3.30, 3.34, 3.51 à 3.66
Agréger réduit également le délai maximal subi par les paquets et permet d'éviter des pertes	3.67 à 3.82
Lorsqu'il n'y a plus de pertes, agréger réduit l'arriéré de traitement maximal (et donc l'occupation mémoire).	3.27, 3.28, 3.32, 3.33 et 3.35 à 3.50
Le choix des files sur lesquelles activer l'agrégation modifie significativement les délais moyens et maximal subis par les paquets.	3.29, 3.30, 3.34, 3.51 à 3.66
Le choix du seuil auquel sont agrégés les flux modifie significativement le délai maximal subi par les paquets.	3.29, 3.30, 3.34, 3.51 à 3.66
Un flux en concurrence avec un flux prioritaire et agrégé voit ses chances de remporter l'accès au medium diminuer.	3.29, 3.30, 3.34, 3.51 à 3.66
Un flux en concurrence avec un flux moins prioritaire agrégé voit le délai de certaines trames augmenter.	3.29, 3.30, 3.34, 3.51 à 3.66
Parfois la station moins prioritaire obtiendra l'accès au médium alors que la file prioritaire n'est pas vide grâce au mécanisme de conservation du *backoff*.	3.15 à 3.19
Les comportements observés pour le délai et l'arriéré de traitement ne sont pas modifiés par l'utilisation de lois différentes (constante ou exponentielle sur 30 tirages différents).	3.24
Des paquets de taille importante (1500 octets par exemple) limitent les bénéfices de l'agrégation (similaire à une agrégation à la source).	3.47 à 3.50, 3.63 à 3.66

Tableau 3.9 – Synthèse des résultats de simulation

rapport de taille entre paquet et agrégat.

Prenons le cas inverse, des flux de haute priorité sont agrégés au détriment de leurs concurrents. Au détriment, en effet, puisque les chances d'accéder au medium sont réduites par l'agrégation pour les flux moins prioritaires. Le mécanisme permettant à ce flux d'améliorer ses chances d'accès au medium est celui de la conservation du *backoff*, si des agrégats sont transmis par le concurrent, le medium est moins souvent remis en jeu, ce qui réduit la vitesse de décrémentation du *backoff* restant.

Cette réduction vient donc s'ajouter au mécanismes de 802.11e qui dégradaient déjà les probabilités d'accès au médium pour les classes de moindre priorité (*background* et *best-effort*). Or, certaines applications sur Internet n'utilisent toujours pas la classification de services ce qui peut conduire des paquets de type voix à être classés dans la classe *best-effort*. Activer l'agrégation dégraderait donc les performances pour ces applications pourtant critiques.

Excepté pour les deux situations décrites ci-avant, agréger améliore les performances. Si le flux prioritaire est agrégé, son concurrent pourra également en profiter car le medium sera libre plus tôt. S'il est lui même agrégé, ce flux transmettra plus de données lorsqu'il aura accès au medium. Ce raisonnement conduit à la conclusion suivante, il est toujours meilleur d'un point de vue global d'activer un tel mécanisme d'agrégation sur toutes les files possibles. En effet, le flux prioritaire traite son arriéré plus rapidement, ce qui libère de l'espace pour le flux moins prioritaire qui lui même traite son arriéré plus vite.

Sur certains réseaux il peut être impossible d'agréger sur toutes les stations. Les réseaux sans-fil peuvent être très hétérogènes et donc certaines stations peuvent ne pas être compatibles avec ces mécanismes

d'agrégation. Nous avons vu que si les autres flux sont agrégés et importants, cette station verra sa qualité de service réduite.

Les clients dont les flux sont en concurrence pour l'accès au medium peuvent donc se retrouver dans différentes situations selon leur rapport de priorité. Pour un client moins prioritaire, les clients prioritaires sont des ennemis qui occupent le medium mais quand ils sont agrégés ces ennemis sont moins imposants et lui laissent plus d'espace. Un client prioritaire considère également les clients moins prioritaires comme des ennemis car même s'il a l'avantage la plupart du temps, il lui arrive de perde la compétition pour l'accès au medium. À cet instant, si le client concurrent transmet un agrégat au lieu d'un paquet, le client prioritaire perd en terme de délai et d'arriéré. Agréger ne permet cependant pas de réduire les inégalités entre clients si le client prioritaire est trop gourmand. Lorsqu'il sature complètement le système, agréger le flux prioritaire peut être insuffisant pour libérer de la bande passante pour d'autres flux moins prioritaires.

Les différents indicateurs choisis se rejoignent sur les grandes tendances mais parfois l'un apportera des informations que l'autre ne donne pas (comme le délai ou l'arriéré moyen vis à vis de l'arriéré maximal). Que ce soit le délai ou les pertes, ils sont importants à la fois pour le client et pour le fournisseur d'accès. Ce dernier observera ainsi l'arriéré de traitement afin de vérifier que ses équipements sont correctement dimensionnés en terme de mémoire. Comme une comparaison entre l'arriéré de traitement instantané et la taille mémoire permet de prévenir les pertes dues aux débordements mémoire, nous nous contenterons dans le chapitre suivant d'utiliser l'arriéré de traitement ainsi que le délai comme indicateurs pour notre algorithme de configuration de l'agrégation. Pour les autres pertes, elles sont contenues dans le délai puisque dans notre cas d'étude (802.11n) nous utilisons le délai en couche MAC et un paquet n'est supprimé de la mémoire que lorsqu'il est acquitté. Dans d'autres cas d'étude et avec d'autres technologies, il sera intéressant d'ajouter cet indicateur afin d'ajuster la configuration de l'agrégation et ainsi respecter au mieux les exigences client et fournisseur d'accès.

A partir des simulations, nous avons pu faire de nombreuses observations qui nous ont permis de proposer un certains nombre de recommandations qui sont ici discutées et parfois reformulées afin de les adapter à la suite de nos travaux.

Recommandation 1 : *Lorsqu'un flux de classe inférieure est dans une situation proche de la famine les flux de classes supérieures doivent être agrégés en priorité pour réduire l'arriéré de traitement maximal et le délai moyen.*

Cette première recommandation introduisait une seconde condition à propos de la synchronisation des flux entrants dans le système. Comme nous considérons qu'ils sont inconnus, cette partie a été supprimée

Recommandation 2 : *S'il n'est pas souhaitable d'agréger les flux concurrents de classes supérieures ou si les délais et arriérés mesurés ne sont toujours pas satisfaisants, le flux considéré doit être agrégé pour réduire son arriéré de traitement maximal et son délai moyen.*

La deuxième recommandation tient compte du contexte dans lequel l'agrégation est envisagée. En effet selon le niveau de bruit ou selon les caractéristiques des stations sur le BSS il peut être impossible d'agréger. Notons que cette mesure peut être inefficace si les flux de classes supérieures occupent la totalité de la bande passante disponible.

Recommandation 3 : *Lorsque l'arriéré de traitement d'un flux de classe supérieure augmente jusqu'à atteindre la limite mémoire, l'agrégation doit être désactivée sur les flux de classe inférieure pour réduire l'arriéré de traitement et le délai moyen du flux de classe supérieure.*

En troisième recommandation, on retrouve une originalité de nos travaux, le choix de désactiver l'agrégation peut améliorer le service offert aux concurrents.

Recommandation 4 : *Lorsque les paquets sont de petite taille (moins de 500 octets) il est inutile de modifier le seuil de taille des agrégats pour réduire l'arriéré de traitement maximal et les délais.*

Recommandation 5 : *Lorsque les paquets sont de grande taille (1500 octets et plus) les flux doivent être agrégés au seuil maximal pour réduire l'arriéré de traitement maximal et les délais.*

Les deux dernières recommandations concernent les caractéristiques des flux et offrent des informations intéressantes quant à l'impact du seuil de taille sur les performances du système.

Notons que ces recommandations sont issues de simulations menées dans un cadre restreint et ne prennent pas en compte les exigences des clients ou du fournisseur d'accès. C'est pourquoi dans le chapitre suivant nous prendrons soin de définir quelques exigences client et fournisseur avant d'établir un algorithme permettant de reconfigurer dynamiquement le système d'agrégation pour remplir au mieux ces exigences.

Dans le Chapitre 4, nous utilisons toutes ces recommandations et les simulations effectuées ici pour établir un algorithme de contrôle des paramètres d'agrégation (activation ou non par classe, valeur du seuil). Son comportement dynamique évolue au fil des mesures du délai de traversée de la couche MAC et de l'arriéré de traitement. Après comparaison aux exigences posées à la fois par les clients et par le fournisseur d'accès, il pourra prendre différentes décisions, comme l'activation ou non de l'agrégation pour une classe donnée.

Chapitre 4

Configuration du système vis à vis des exigences client et fournisseur

Après avoir étudié l'influence des différents paramètres d'agrégation sur le système (la couche MAC d'une station 802.11n), nous pouvons proposer ici un algorithme permettant de configurer ce système dynamiquement (comme illustré en figure 4.1). Cette configuration repose sur les exigences fournies d'une part par le fournisseur d'accès et d'autre part par le client. C'est pourquoi nous définirons dans un premier temps les exigences qu'ils peuvent exprimer.

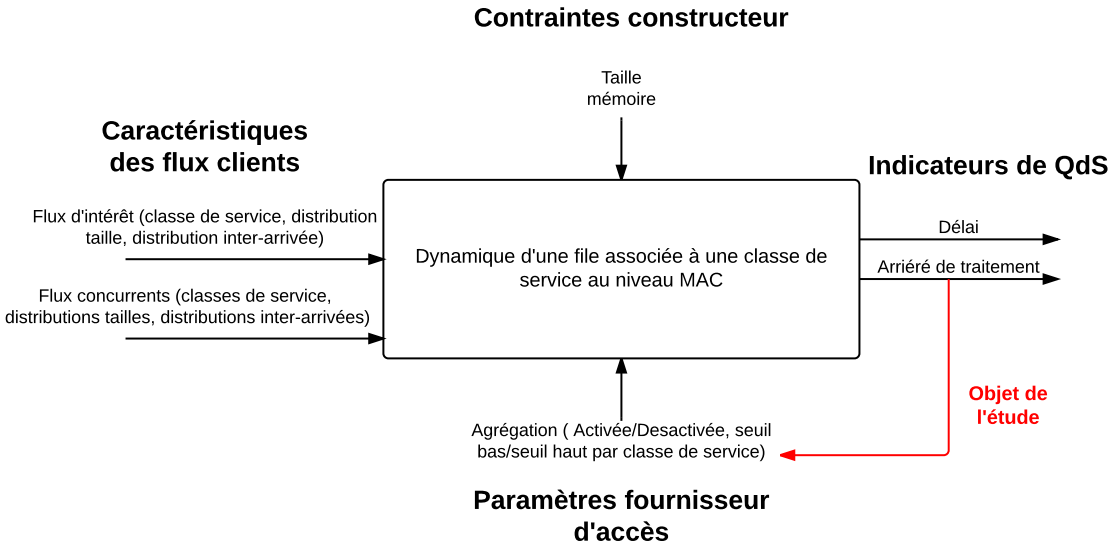


Figure 4.1 – Représentation du système sous forme de boîte noire et positionnement des différentes contraintes, indicateurs et paramètres du système

La figure 4.1 représente le schéma boîte noire du système légèrement modifié par rapport au chapitre précédent puisque l'objectif de ce chapitre est maintenant de proposer un boucle de retour sur les paramètres de l'agrégation.

4.1 Définition des exigences

Afin de définir des exigences possibles nous nous plaçons d'abord du point de vue du client et ensuite du point de vue du fournisseur d'accès. Cette section se termine sur une discussion des éventuelles oppositions entre les exigences des différentes parties prenantes.

4.1.1 Du point de vue d'un client

Le besoin d'un client se résume à obtenir le meilleur service possible pour le prix le moins élevé (rapport qualité/prix). Comme évoqué en chapitre 1, nous pouvons distinguer deux types de clients :

— le client final
— le fournisseur de services

Nous pouvons situer chacun de ces clients d'un bout à l'autre de l'architecture. À la source le client final (l'utilisateur des services) souhaite que les performances du réseau lui permettent de profiter au mieux des services fournis. Cette notion de satisfaction est très difficile à exprimer de façon spécifique et surtout mesurable sans connaître les services souhaités par l'utilisateur (téléphonie sur IP, visio-conférence, VOD, TV HD, ...).

Pour simplifier nous considérons les classes de services définies dans 802.11e et y associons quelques exigences selon les services souhaités par le client :

Classe	**Exemple d'exigences utilisateur**	**Traduction pour le réseau**
background	aucune	aucune
best-effort	échange de fichiers le plus rapide possible	taux de pertes nul et débit utile équivalent à la capacité du medium
voice	pas de distorsion de la voix pas de blancs au milieu des phrases	taux de pertes nul,gigue nulle, faible délai
video	pas de coupure de la vidéo, image et son de haute qualité	taux de pertes nul et débit utile équivalent au minimum au flux vidéo

Tableau 4.1 – Traduction des exigences client sur les indicateurs de performances pour chaque classe de service

Comme montré au chapitre 3, agréger est généralement profitable aux clients à condition que les concurrents n'agrègent pas afin d'éviter une baisse de performances. À l'autre bout du réseau, le fournisseur de services exprime des exigences similaires à celles du tableau 4.1. L'unique différence réside dans son poids économique plus important qui lui permet de négocier des garanties dans son SLA. Cela ne l'empêche pas d'être en concurrence avec d'autres fournisseurs. En effet, la tendance est à l'utilisation de data-centers dans lesquels plusieurs fournisseurs de service peuvent être hébergés. Cette concurrence peut conduire au même problème de baisse des performances si les concurrents agrègent.

Qu'ils soient utilisateurs particuliers ou fournisseurs de service, les exigences des clients sont de nature identique et font appel aux mêmes indicateurs. Seule leur influence sur le fournisseur d'accès change selon leur poids économique.

4.1.2 Du point de vue du fournisseur d'accès

Rappelons que le fournisseur d'accès est défini comme l'opérateur du réseau. Il est en charge de son bon fonctionnement, des garanties fournies sur les SLA et de la configuration du réseau. Le fournisseur d'accès assure simplement l'interface entre les clients finaux et les fournisseurs de service. Notons qu'en réalité les fournisseurs d'accès remplissent souvent le rôle de fournisseur de service mais nous séparons ces rôles afin d'exprimer au mieux les exigences qui en découlent.

Alors que les clients souhaitent tous obtenir le meilleur service, le fournisseur d'accès veut maximiser le rendement économique de son réseau. Que ce soit avec un seul fournisseur de service ou avec plusieurs clients, l'idée est d'utiliser le système au maximum de ses capacités. En ce sens, l'agrégation de paquets est un mécanisme qui rend service au fournisseur d'accès puisqu'elle lui permet d'augmenter le débit utile offert au client et donc d'augmenter sa rentabilité sans pour autant modifier les équipements utilisés.

De plus, un fournisseur d'accès sera intéressé par une réduction des besoins en mémoire des équipements réseau. Le but est de réduire le coût à l'achat de ces équipements et ensuite de leur permettre de suivre à long terme les besoins croissants des clients en termes de bande passante.

Ces besoins se traduisent sur le réseau par les exigences suivantes :

— utiliser au maximum la bande passante offerte
— minimiser l'arriéré de traitement dans le système
— offrir un service équitable au sens des priorités (classes de service ici) établies

Les deux premières exigences rejoignent le souhait des clients de profiter d'un délai réduit puisque le fait de minimiser l'arriéré de traitement ou de maximiser l'utilisation de bande passante minimise également le délai de traversée du système. C'est à propos de la dernière exigence qu'une divergence potentielle existe avec les exigences exprimées par les clients.

Les exigences définies ici reflètent simplement notre idée. Il serait intéressant à l'avenir d'utiliser de réelles exigences issues d'une discussion avec un panel de clients et de fournisseurs d'accès.

4.1.3 Oppositions entre exigences

Chaque client souhaite obtenir le meilleur service possible alors que le fournisseur d'accès veut rentabiliser son système. L'opposition entre ces exigences provient du point de vue adopté par chacun, pour un client les concurrents ne devraient pas exister alors qu'ils sont indispensables au fournisseur pour rentabiliser son système. Ce dernier se voit alors contraint de trouver un équilibre pour satisfaire un maximum de clients. Cet équilibre passe par la bonne configuration du système d'agrégation qui peut influer significativement sur les performances offertes aux différents clients surtout avec la classification de service. Nous ne parlons pas ici d'égalité mais bien d'équité puisque chaque client devrait obtenir le service pour lequel il paye et obtenir un meilleur service s'il paie plus (cas du fournisseur de service).

C'est ce qui limite le poids des particuliers ou des petites entreprises excepté lorsqu'ils sont nombreux. Cependant même s'ils sont nombreux à souhaiter la même chose, ils n'en restent pas moins des concurrents et leurs exigences s'opposent naturellement. Cette relation ami/ennemi conduit le fournisseur d'accès à trouver un équilibre pour satisfaire un maximum de clients.

4.1.4 Autour de la neutralité du Net

Comme expliqué au Chapitre 1, le poids économique énorme de certains de ces fournisseurs de services ébranle aujourd'hui les fondements de ce qu'on appelle "la neutralité du net". Puisque les fournisseurs de service les plus importants peuvent réserver la bande passante pour leurs clients, les autres fournisseurs risquent de voir leur qualité de service fortement dégradée. En pratique, le fournisseur d'accès peut décider d'éviter une dégradation du service en augmentant la capacité du medium et ainsi la bande passante disponible pour tous. Cette augmentation de la capacité du medium peut être très coûteuse, il est donc logique que les plus intéressés les financent. Ainsi les capacités augmentent chez les clients pour leur donner accès à un service supplémentaire payant. La boucle est bouclée : le client paye le fournisseur de service qui investit dans l'infrastructure du fournisseur d'accès. À mon avis, ce cercle est plus équitable que celui qui consiste à facturer un surplus au client sur son abonnement pour rembourser des évolutions d'infrastructure (dont parfois il ne profite pas) avant de lui faire payer une seconde fois le service.

Ces problèmes nécessitant une remise en cause de l'infrastructure ne peuvent être résolus par le seul mécanisme d'agrégation de paquets. Celui-ci permet uniquement de profiter au maximum de la capacité offerte par le medium, non de l'augmenter. C'est pourquoi la première exigence est en réalité la contrainte de stabilité suivante : $R << C$ (avec R le trafic entrant dans le système et C la capacité du médium). Celle-ci doit toujours être respectée avant de considérer l'utilisation de l'agrégation.

4.2 Adaptation des paramètres d'agrégation

4.2.1 Positionnement scientifique

Au sein de la communauté 802.11, les auteurs se sont intéressés à la création d'algorithmes que ce soit pour l'ordonnancement des paquets en vue d'optimiser les performances de l'agrégation ou pour le mécanisme d'agrégation lui même. Par exemple, (Bhanage *et al.*, 2011) propose un algorithme d'agrégation basé sur l'arriéré de traitement présent dans le système. Le principe clé de cet algorithme est : *"agréger tant qu'il y a des trames en mémoire pouvant être transmises dans une seule TXOP"*.

Le comportement de l'algorithme se résume comme suit : lorsqu'une station gagne l'accès au medium, elle place ses paquets en mémoire de transmission mais ne transmet pas tout de suite. Elle reste dans cet état tant que la taille de l'agrégat formé ne dépasse pas le seuil et tant que la durée de transmission respecte la durée de la TXOP. Dès que l'une de ces conditions est dépassée, la station transmet le contenu de la mémoire de transmission.

Le fait de respecter la contrainte de TXOP imposée par 802.11e assure une certaine équité entre stations concurrentes. L'évaluation de l'algorithme est basée sur des traces réelles de trafic audio et vidéo et utilise une plate-forme expérimentale (ORBIT) avec des équipements embarquant un driver MadWifi modifié. Ces expériences révèlent des améliorations significatives en termes de délai moyen, de débit et de gigue.

Par ailleurs, (Selvam et Srikanth, 2010) proposent une approche différente basée sur l'échéance d'arrivée d'un paquet, un seuil de taille et une taille optimale des agrégats. Cette taille optimale est calculée à partir du BER. L'algorithme qu'ils proposent teste d'abord si l'échéance est atteinte, et ensuite si le seuil de taille l'est. Que l'un ou l'autre de ces critères soit rempli suffit à passer à l'étape suivante qui consiste à tester si la quantité d'arriéré est supérieure à la taille optimale. Si ce n'est pas le cas, une A-MSDU est

construite normalement. Si cette condition est vraie, les paquets sont triés par taille croissante, des A-MSDU sont construites et encapsulées dans une A-MPDU. Ce mécanisme permet d'améliorer sensiblement les performances de l'agrégation en cas de non-saturation du medium d'après les résultats obtenus sur un simulateur développé par leurs soins en langage C.

Une méthode différente est utilisée par (Saif *et al.*, 2011), qui propose de réduire les en-têtes utilisée au sein des agrégats de MSDU pour passer de 10 octets par sous-trames à 4 octets. Les bénéfices sont significatifs pour des petits paquets : par exemple, avec des paquets de 128 octets, l'amélioration est de 30 % par rapport aux agrégations classiques (d'après une simulation réalisée à l'aide de ns-2).

(Dely *et al.*, n.d.) utilise pour sa part un système de contrôle flou pour paramétrer le système d'agrégation. En particulier il essaie de contrôler le délai artificiellement ajouté (le seuil de temps) en fonction d'informations sur le délai MAC et le débit. Ce contrôle utilise plusieurs indicateurs pour tenter de placer le système dans un état de débit élevé pour un délai faible.

Ces études se concentrent sur la mise en œuvre d'un algorithme d'optimisation du fonctionnement propre de l'agrégation comme une modification des en-têtes, une adaptation de la taille de l'agrégat, etc. Les propositions et les évaluations associées s'inscrivent du point de vue d'une communication entre deux stations. Lorsque la concurrence entre stations est étudiée seules des mesures globales sont prises comme le délai des agrégats ou encore le pourcentage d'utilisation du canal. Rappelons que nous cherchons à étudier un flux particulier ce qui nous conduit à une approche différente.

De plus, nous choisissons de modifier au minimum le système prévu par la norme. Contrairement aux travaux de la littérature, le format des trames proposé dans la norme n'est pas modifié (pas de réduction des en-têtes). Aucun seuil temporel n'est considéré puisqu'il provoquerait une rétention des agrégats (qu'il soit arbitrairement fixé ou basé sur la TXOP) et donc une augmentation potentielle des délais. Nous souhaitons conserver la philosophie opportuniste du système d'agrégation.

Nous avons vu que l'agrégation ne peut être que bénéfique pour la station virtuelle (la classe de service) considérée. Il en est alors de même pour tout algorithme d'optimisation proposé. Or l'agrégation peut avoir des effets négatifs sur les concurrents. C'est pourquoi nous nous intéresserons en particulier aux moyens d'améliorer le service offert aux stations en fonction de leurs exigences, de celles de leurs concurrents et des priorités définies par 802.11e. Nous évaluerons l'algorithme proposé sur les indicateurs définis pour le chapitre 3 que sont le délai de traversée de la couche MAC et l'arriéré de traitement de cette même couche.

4.2.2 Proposition d'un algorithme de configuration de l'agrégation

Pour établir un algorithme de configuration du système d'agrégation, nous utilisons les recommandations établies au chapitre précédent et rappelées ci-dessous.

Recommandation 1 : *Lorsqu'un flux de classe inférieure est dans une situation proche de la famine les flux de classes supérieures doivent être agrégés en priorité pour réduire l'arriéré de traitement maximal et le délai moyen.*

Recommandation 2 : *S'il n'est pas souhaitable d'agréger les flux concurrents de classes supérieures ou si les délais et arriérés mesurés ne sont toujours pas satisfaisants, le flux considéré doit être agrégé pour réduire son arriéré de traitement maximal et son délai moyen.*

Recommandation 3 : *Lorsque l'arriéré de traitement d'un flux de classe supérieure augmente jusqu'à*

atteindre la limite mémoire, l'agrégation doit être désactivée sur les flux de classe inférieure pour réduire l'arriéré de traitement et le délai moyen du flux de classe supérieure.

Recommandation 4 : *Lorsque les paquets sont de petite taille (moins de 500 octets) il est inutile de modifier le seuil de taille des agrégats pour réduire l'arriéré de traitement maximal et les délais.*

Recommandation 5 : *Lorsque les paquets sont de grande taille (1500 octets et plus) les flux doivent être agrégés au seuil maximal pour réduire l'arriéré de traitement maximal et les délais.*

La première recommandation nous renseigne sur la conduite à tenir lorsque la valeur de l'arriéré de traitement devient critique. Nous débuterons donc par établir un algorithme basé sur ce seul indicateur.

Posons maintenant b_{ac} l'arriéré de traitement mesuré pour une classe de service (*Access Category, AC*). Nous nous plaçons du point de vue d'un système d'agrégation sur une station ou un point d'accès 802.11n qui possède quatre classes de service à gérer.

En respectant la standardisation, chaque classe peut être configurée dans trois états d'agrégation, pas d'agrégation, agrégation au seuil bas, agrégation au seuil haut. Avec quatre classes nous arrivons à un total de quatre-vingt une combinaisons. Afin d'éviter l'explosion combinatoire inévitable si le nombre de stations augmente, nous établissons simplement une liste de règles qui évoluent en fonction de la valeur de l'arriéré.

Cette valeur d'arriéré mesurée est comparée à l'exigence (notée $\widetilde{b_{ac}}$) et permet de décider des actions à mettre en œuvre. Il est souhaitable d'utiliser un pourcentage de la valeur cible pour laisser au système le temps de réagir et ainsi éviter un dépassement de l'exigence.

Données : exigences sur l'arriéré pour chaque classe $\widetilde{b_{ac}}$; mesures de l'arriéré b_{ac} ($ac \in 1, 2, 3, 4$ avec 1 la priorité la plus élevée) ; facteur d'anticipation p.
Résultat : état de l'agrégation $E_{agg}[ac]$ pour toutes les classes ($E_{agg}[ac]$ prend les valeurs $noagg$, agg_{Sb} et agg_{Sh} pour sans agrégation, agrégation seuil bas et agrégation seuil haut respectivement).
initialisation $E_{agg} \leftarrow [agg_{Sh}, agg_{Sh}, agg_{Sh}, agg_{Sh}]$;
début
  si $b_{ac} \geq \widetilde{b_{ac}}$ **alors**
    pour chaque $i \in [1, ac]$ **faire** $E_{agg}[i] \leftarrow agg_{Sh}$;
    pour chaque $i \in]ac, 4]$ **faire** $E_{agg}[i] \leftarrow noagg$;
  sinon si $b_{ac} < \widetilde{b_{ac}}$ **alors**
    pour chaque $i \in]ac, 4]$ **faire** $E_{agg}[i] \leftarrow agg_{Sb}$;

Algorithme 2 : Ébauche d'un algorithme de configuration de l'agrégation

L'algorithme 2 est issu d'une lecture directe des recommandations du chapitre 3. Ainsi, en suivant les recommandations 1 et 2, l'agrégation sera activée sur la classe considérée et sur les classes qui lui sont prioritaires lorsque son exigence anticipée est atteinte. De plus, la recommandation 3 implique de désactiver l'agrégation sur les classes de plus faible priorité dans ce même cas. Ensuite, les recommandations 4 et 5 indiquent que le seuil haut doit être utilisé pour de gros paquets mais qu'il est inutile si les paquets sont petits. Ici, inutile ne signifie pas pénalisant. Nous choisissons donc d'utiliser un état initial qui consiste à agréger toutes les classes au seuil haut.

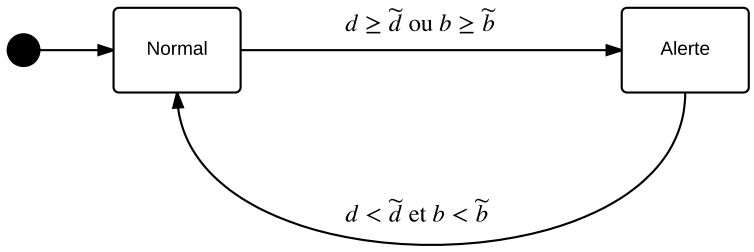
Le retour à la normale (sous l'exigence d'arriéré) des classes de priorités hautes implique que les classes de priorité basse peuvent agréger à nouveau. Une telle modification peut conduire à une augmentation franche

de l'arriéré sur les classes de plus haute priorité. Or si l'arriéré dépasse l'exigence anticipée, les classes de plus faible priorité seront à nouveau privées d'agrégation. Afin d'atténuer cette potentielle instabilité du système, nous choisissons d'affecter le seuil bas aux classes de moindre priorité tant qu'elles respectent leur exigence d'arriéré.

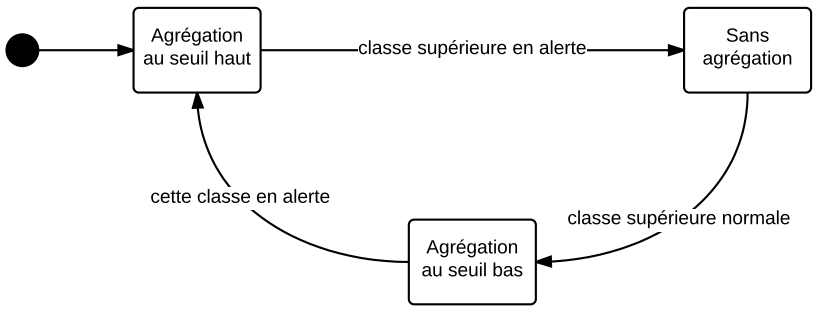
Ce premier algorithme établi, il nous faut décider du moment auquel il sera actif. L'idéal serait bien sûr de disposer de ressources illimitées et de l'exécuter de façon distribuée sur chaque classe et en parallèle, ce qui n'est pas réaliste. La question est donc la suivante : à quel moment de telles modifications sur le système d'agrégation seront efficaces ?

Pour pousser la réflexion, rappelons que le système d'agrégation est implémenté en entrée de la couche MAC et ajoute chaque paquet entrant dans un agrégat en formation. Il apparaît donc logique de placer ce système à l'entrée de la couche MAC. Cependant, le nombre de paquets reçus est potentiellement très important et l'exécution d'un algorithme à ce moment peut modifier de manière exagérée le temps mis par les paquets pour traverser la couche MAC.

La seconde solution est de prendre une décision quant à la configuration du système d'agrégation juste avant que les agrégats soient transmis, soit lorsqu'une station obtient l'accès au medium. À ce niveau, la décision prise en fonction de l'algorithme proposé ci-dessus est partiellement inutile puisque chaque classe lorsqu'elle gagne l'accès au medium décidera d'agréger au seuil haut si elle est en alerte. Or, le comportement souhaité pour l'algorithme suit plutôt les modalités spécifiées par les machines à états en figure 4.2.



(a) Observation



(b) Configuration

Figure 4.2 – Machines à état pour la configuration de l'agrégation sur une classe de service

La première machine (figure 4.2a) observe l'état du système qui évolue entre deux états suivant une mesure de l'arriéré b et du délai d. Ces mesures sont comparées à leurs exigences respectives $\widetilde{b}$ pour l'arriéré

et $\widetilde{d}$ pour le délai :

— $d < \widetilde{d}$ et $b < \widetilde{b}$ (état normal)
— $d \geq \widetilde{d}$ ou $d \geq \widetilde{d}$ (état d'alerte)

La seconde machine à état configure cet état en fonction d'une vision globale de l'état des classes (figure 4.2b).

Dans l'état initial, nous choisissons toujours d'activer l'agrégation sur un maximum de classes, toutes si possible et au seuil haut. Ensuite lorsqu'une ou plusieurs classes entrent en état d'alerte, l'agrégation est désactivée sur les classes de priorité inférieures qui sont à l'état normal. Si la classe retourne à l'état normal, l'agrégation peut être activée à nouveau, d'abord avec le seuil bas puis avec le seuil haut si celle-ci entre en alerte.

Pour définir à quel moment les stations doivent agir sur le système d'agrégation nous nous sommes demandé comment l'implémentation en était réalisée au sein de *Modeler*. Nous avons découvert qu'une partie de l'agrégation est bien réalisée à chaque réception d'un paquet mais une seconde partie est effectuée au moment de transmettre. La première est l'ajout des en-têtes spécifiques aux agrégats la seconde est l'agrégation à proprement parler. Nous savons que l'agrégation est toujours bénéfique au flux qui agrège. Nous savons également que ce n'est pas toujours vrai pour les stations concurrentes. C'est pourquoi en recommandation 3 nous préconisons de désactiver l'agrégation sur les classes moins prioritaires. L'algorithme que nous proposons ici est donc exécuté sur une station à chaque fois qu'elle obtient l'accès au medium.

Chaque classe adapte donc son comportement (agrégation ou non et seuil de taille) à chaque fois qu'elle gagne l'accès au medium et ce en fonction de l'état de toutes les classes de la station (normal ou alerte). C'est à cet endroit que les décisions prises auront le plus d'impact. Une modification de l'algorithme est maintenant nécessaire.

Données : exigences sur l'arriéré pour chaque classe $\widetilde{b_{ac}}$; mesures de l'arriéré b_{ac} ($ac \in 1, 2, 3, 4$ avec 1 la priorité la plus élevée) ; facteur d'anticipation p ;état des classes $alerte[ac]$
Résultat : état de l'agrégation $E_{agg}[ac]$ pour la classe ($E_{agg}[ac]$ prend les valeurs $noagg$, agg_Sb et agg_Sh pour sans agrégation, agrégation seuil bas et agrégation seuil haut respectivement).

ac **gagne l'accès au medium**
  pour chaque $i \in [1, ac[$ **faire**
    si $b_i \geq p\widetilde{b_i}$ **alors** $alerte[i] \leftarrow 1$;
    sinon si $b_i < p\widetilde{b_i}$ **alors** $alerte[i] \leftarrow 0$;
    $\sum_{alerte} \leftarrow \sum_{alerte} + alerte[i]$;
  si $\sum_{alerte} > 0$ **alors** $E_{agg}[ac] \leftarrow noagg$;
  sinon si $E_{agg}[ac] = noagg$ **alors** $E_{agg}[ac] \leftarrow agg_Sb$;
  sinon si $E_{agg}[ac] = agg_Sb$ **et** $b_ac \geq \widetilde{b_{ac}}$ **alors** $E_{agg}[ac] \leftarrow agg_Sh$;

Algorithme 3 : Proposition d'un algorithme de configuration de l'agrégation

Ce second algorithme cherche dans un premier temps à compter le nombre de classes de priorité supérieure (par rapport à la classe qui gagne l'accès au medium) qui sont en état d'alerte. Il configure ensuite l'agrégation pour cette station avant toute transmission. Contrairement au cas précédent, ici le comportement d'une classe est dicté juste avant son émission par l'état des classes de plus haute priorité.

4.3 Évaluation de l'algorithme proposé

Pour évaluer cet algorithme nous avons d'abord écrit un programme qui reproduit le comportement modélisé lors de l'analyse de la norme effectuée au chapitre 2. Ensuite cet algorithme a été implémenté dans le simulateur Riverbed modeler afin d'évaluer ses performances sur un outil indépendant.

4.3.1 Avec notre propre programme

Nous utiliserons dans un premier temps l'un des scénario utilisé dans les chapitres 2 et 3 avec deux classes de service (*best-effort* et *voice*). La valeur du *backoff* est constante et fixée à CW_{min}. Les valeurs de CW_{min} et de l'AIFS sont spécifiées en tableau 4.2.

	best-effort	**voice**
tranche canal	9 μs	9 μs
AIFS (en tranche canal)	3	1
CW_{min} **(en tranche canal)**	31	7

Tableau 4.2 – Valeurs d'AIFS et de *backoff* pour une première évaluation de l'algorithme proposé

Les caractéristiques physiques du point d'accès considéré sont identiques à celles posées dans les chapitres précédents avec un débit au niveau physique de 65 Mb/s.

La structure de la mémoire interne d'une station peut théoriquement être composée d'une seule mémoire partagée par toutes les classes ou de plusieurs mémoires affectées à chaque classe. Nous supposons ici comme en pratique que la mémoire est partagée par toutes les classes mais qu'il est possible de mesurer l'arriéré de traitement de manière différenciée pour chaque classe.

Comme la mémoire est partagée, nous divisons sa limite unique en quatre pour affecter cette exigence à chaque classe prévue dans la norme même si nous n'en utilisons que deux ici. À partir de la valeur maximale proposée dans le simulateur Riverbed Modeler (128000 octets) nous obtenons 32000 octets d'arriéré maximal exigé pour chaque classe. Il ne manque plus que le facteur d'anticipation qui est fixé à 40 % de la valeur maximale (soit 12800 octets) car dans cet exemple l'algorithme n'aurait aucun effet au dessus de ce seuil.

Avec tous ces paramètres nous obtenons les résultats exprimés en tableau 4.3 qui représentent les différences de délai et d'arriéré pour chaque classe.

	Sans algorithme	**Avec algorithme (facteur 0,4)**	**différence (en %)**
arriéré max BE	42500	49500	16
arriéré max VO	26000	20000	-23
arriéré total	67000	69500	4
délai max BE	0,011959	0,012583	5
délai max VO	0,004403	0,003515	-20
délai moyen BE	0,006514	0,006982	7
délai moyen VO	0,002461	0,002017	-18

Tableau 4.3 – Évaluation de l'algorithme BE 500 50 VO 500 50

Les résultats obtenus sont encourageant puisque l'implémentation de l'algorithme permet de réduire de 23 % l'arriéré maximal de la classe *voice* en augmentant de 16 % celui de la classe *best-effort*. L'arriéré total

en octets est cependant augmenté de 4 % ce qui est logique puisque la classe *best-effort* arrête d'agréger pendant un moment.

Dans cette étude nous avons également relevé le délai de traversée de la couche MAC du point d'accès 802.11n. Ce délai, pour la classe *voice*, est amélioré de 16 % en moyenne et de 20 % au maximum. Au contraire le délai subi par les paquets de la classe *best-effort* est augmenté de 3 à 5 %. L'amélioration offerte à la classe *voice* est donc plus importante que la détérioration subie par la classe *best-effort* en proportions. Cependant, les données brutes sont une réduction de presque 900 μs pour la classe *voice* contre une augmentation de plus de 600 μs pour la classe *best-effort*. Cette différence plus importante en pourcentage est expliquée par des délais beaucoup plus importants pour la classe *best-effort* puisqu'elle est de moindre priorité.

Il est possible de configurer notre algorithme à partir d'échéances temporelles à respecter pour les paquets. Cependant la légitimité d'une telle implémentation est à discuter. En effet pour certaines applications comme la téléphonie, il est habituel de considérer une échéance sur le délai de bout en bout d'une application. Or, seul le délai de traversée de la couche MAC d'un équipement réseau (ici le point d'accès) est amélioré. Poser une échéance sur ce délai de traversée suppose un découpage du délai de bout en bout. Découpage qui permet d'affecter un temps borné à chaque équipement pour effectuer sa mission. De plus, influer sur le mécanisme d'agrégation ne permet en aucun cas d'apporter des garanties, simplement d'améliorer le service offert à certains paquets prioritaires. Cette dernière remarque est également valable pour l'arriéré de traitement.

Ce premier exemple d'application nous a permis d'évaluer les performances de notre algorithme sur un cas simple. Passons à un second scénario avec des flux différents :

— un flux *best-effort* avec 300 octets / 90 μs

— un flux *video* avec 500 octets / 50 μs

— un flux *voice* avec 100 octets / 125 μs

Le *backoff* est fixé à 15 tranches canal et l'AIFS à 2 tranches canal pour le flux *video*. Le reste des paramètres est identique à l'exemple précédent.

Concernant l'exigence sur l'arriéré de traitement, celle-ci est ici exprimée en octets puisque la taille des paquets est variable entre les stations soit une limite à 32 000 octets. Le facteur d'anticipation prendra deux valeurs ici 0,5 et 0,1 et ce pour toutes les classes. Ces nouveaux paramètres fournissent les résultats listés en tableaux 4.4 et 4.5.

	Sans algorithme	**Avec algorithme (facteur 0,5)**	**Écarts (en %)**
arriéré max BE	17100	29400	72
arriéré max VI	34500	29500	-14
arriéré max VO	1900	1500	-21
arriéré total	43100	47000	9
délai max BE	0,008415	0,012005	43
délai max VI	0,010016	0,007413	-26
délai max VO	0,002867	0,002139	-25
délai moyen BE	0,004028	0,006195	56
délai moyen VI	0,004459	0,003314	-26
délai moyen VO	0,001287	0,001069	-17

Tableau 4.4 – Évaluation de l'algorithme BE 300 90 VI 500 50 VO 100 125 (facteur 0,5)

Avec un facteur d'anticipation de 50%, les performances offertes aux classes *video* et *voice* sont améliorées de 14 à 26 % à la fois en termes de délai et d'arriéré de traitement (voir tableau 4.4). Néanmoins, cette amélioration coûte cher à la classe *best-effort* qui voit son arriéré augmenter de plus de 70% et ses délais d'environ 50%. Encore une fois l'algorithme mis en œuvre provoque une augmentation de l'arriéré total dans la station. Cela confirme que l'algorithme proposé ici est en faveur d'une partie des clients mais en défaveur d'une part des autres clients et d'autre part du fournisseur dans le cas ou il souhaite minimiser l'arriéré de traitement de ses passerelles et si ces dernières utilisent une mémoire partagée.

	Avec algorithme (facteur 0,1)	**Écarts / Sans algorithme (en %)**	**Écarts / Avec algorithme (facteur 0.5) (en %)**
arriéré max BE	29400	72	0
arriéré max VI	29500	-14	0
arriéré max VO	1200	-37	-20
arriéré total	47000	9	0
délai max BE	0,012522	49	4
délai max VI	0,006945	-31	-6
délai max VO	0,001681	-41	-21
délai moyen BE	0,006639	65	7
délai moyen VI	0,002443	-45	-26
délai moyen VO	0,000935	-27	-12

Tableau 4.5 – Évaluation de l'algorithme BE 300 90 VI 500 50 VO 100 125 (facteur 0,1)

En tableau 4.5, l'utilisation d'un facteur d'anticipation de 10% fourni des résultats intéressants puisque les performances sont améliorées jusque 45% soit parfois plus de 20% meilleur qu'avec un facteur à 50%. Le fait de réduire ce facteur permet d'activer l'algorithme plus souvent qu'auparavant. Ses effets bénéfiques peuvent donc se cumuler pour donner de tels résultats. De plus, l'arriéré total ainsi que l'arriéré du flux *best-effort* restent inchangés. Seuls les délais pour ce flux *best-effort* sont légèrement dégradés (de l'ordre de 7%).

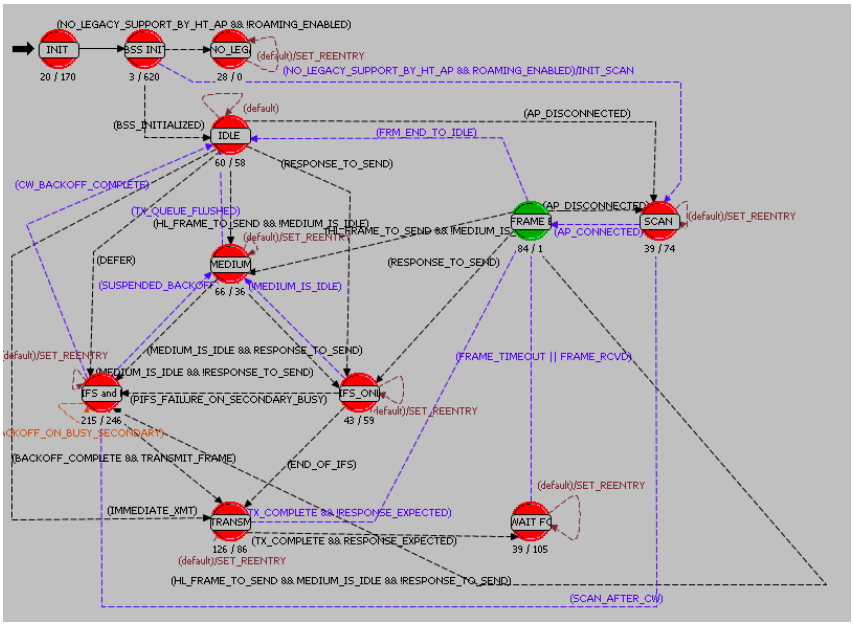
Ces résultats montrent une influence significative de l'exigence anticipée sur les délais et l'arriéré des classes de haute priorité. Cette influence est ici limitée à une répartition différente de l'arriéré entre les classes et ne modifie pas l'arriéré global. Concernant le délai, les inégalités entre classes sont d'autant plus fortes que le facteur d'anticipation est élevé. Cela signifie que plus les exigences sur les classes de priorités hautes sont importantes, plus les délais subis par les paquets de basse priorité seront importants.

4.3.2 Avec Riverbed Modeler

4.3.2.a Description de l'implémentation

L'algorithme proposé à été inclus dans un nouveau modèle de station sans fil au sein de Modeler. Pour cela, il a été nécessaire d'étudier les machines à état qui composent la couche MAC d'une station 802.11 dans ce simulateur (figure 4.3).

La figure 4.3a représente l'automate à états finis de la couche MAC de 802.11 implémentée dans une station Modeler. Le code proposé est illustré figure 4.3b et intervient à l'entrée de l'état *transmit* de l'automate. Sa localisation précise se situe donc au moment où la station obtient l'accès au medium



(a) Place dans l'automate de la partie HCF de la sous-couche MAC d'une station sans fil

```
    // Algorithm
if (DB_ALGO) {
    for (i=0;i<WLANC_HCF_AC_COUNT;i++) {
        if (is_ac_amsdu_shndl_reg[i])
            op_stat_write (ac_amsdu_limit_b_shndl_arr[i], max_amsdu_sizes[i]);
    }

    for (i=cur_tx_ac+1;i<WLANC_HCF_AC_COUNT;i++) {
        if (ac_backlog[i]>WLANC_AC_BACKLOG_LIMIT) // 128000 bits
            are_higher_ac_full=OPC_TRUE;
        if (ac_delays[i]>WLANC_AC_DEADLINES[i])
            are_higher_ac_late=OPC_TRUE;
    }
    if ((are_higher_ac_full) || (DEADLINES_EXIST && (are_higher_ac_late))) {
        // Do not aggregate for the AC
        switch(cur_tx_ac) {
            case WlanC_AC_VO:
                {
                ht_params_ptr->amsdu_bitmap=ht_params_ptr->amsdu_bitmap & 0x003F;
                break;
                }
            case WlanC_AC_VI:
                {
                ht_params_ptr->amsdu_bitmap=ht_params_ptr->amsdu_bitmap & 0x00CF;
                break;
                }
            case WlanC_AC_BE:
                {
                ht_params_ptr->amsdu_bitmap=ht_params_ptr->amsdu_bitmap & 0x00F6;
                break;
                }
            case WlanC_AC_BK:
                {
                ht_params_ptr->amsdu_bitmap=ht_params_ptr->amsdu_bitmap & 0x00F9;
                break;
                }
        }
        // Update AC A-MSDU Limit
        max_amsdu_sizes[cur_tx_ac]=0;
    } else if (!IS_AMSDU_ENABLED(ht_params_ptr, hld_ptr->up)) {
        // Aggregate for the AC
        ht_params_ptr->amsdu_bitmap=ht_params_ptr->amsdu_bitmap | WLANC_AC_TO_UP_AGGR_BIT
        // Update AC A-MSDU Limit at the low level
        max_amsdu_sizes[cur_tx_ac]=30712; // 3839 bytes
    } else if ((max_amsdu_sizes[cur_tx_ac]==30712) && ((ac_backlog[cur_tx_ac]>WLANC_AC_BA
        // Aggregate for the AC
        ht_params_ptr->amsdu_bitmap=ht_params_ptr->amsdu_bitmap | WLANC_AC_TO_UP_AGGR_BIT
        // Update AC A-MSDU Limit at the high level
        max_amsdu_sizes[cur_tx_ac]=63480; // 7935 bytes
    }

    for (i=0;i<WLANC_HCF_AC_COUNT;i++) {
        if (is_ac_amsdu_shndl_reg[i])
            op_stat_write (ac_amsdu_limit_b_shndl_arr[i], max_amsdu_sizes[i]);
    }
```

(b) Extrait du code implémenté

Figure 4.3 – Implémentation de l'algorithme établi dans Riverbed Modeler

et juste avant qu'elle ne transmette une trame. Cette localisation ne concerne que le cœur de l'algorithme puisqu'il est nécessaire de modifier des fonctions disséminées dans le programme.

L'algorithme établi impose deux modifications au modèle générique implémenté. D'abord l'agrégation est désactivée ou activée dynamiquement lors d'une simulation. Auparavant, ce paramètre était statique et configuré avant toute simulation. Ensuite, le seuil de taille varie dynamiquement et une valeur différente peut être affectée sur chaque classe. Dans le modèle générique Modeler, ce seuil était identique pour toutes les classes d'une station et comme pour le paramètre précédent, configuré avant l'exécution d'une simulation.

Afin de suivre l'évolution de l'état de chaque classe nous avons développé deux nouvelles sondes. La première recueille la taille maximale des agrégats pour chaque classe de façon dynamique. Cette statistique nous renseigne sur l'état de l'agrégation pour une classe à un instant donnée. Ainsi nous pouvons suivre les actions effectuées par l'algorithme implémenté. La seconde nous permet de suivre l'arriéré par classe en bits, alors que la sonde existante proposée dans Modeler n'était exprimée qu'en paquets.

4.3.2.b Résultats avec un scénario étudié dans les chapitres précédents

Si nous cherchons à utiliser l'un des scénarios proposé au chapitre 3 nous obtenons les résultats présentés en figure 4.4.

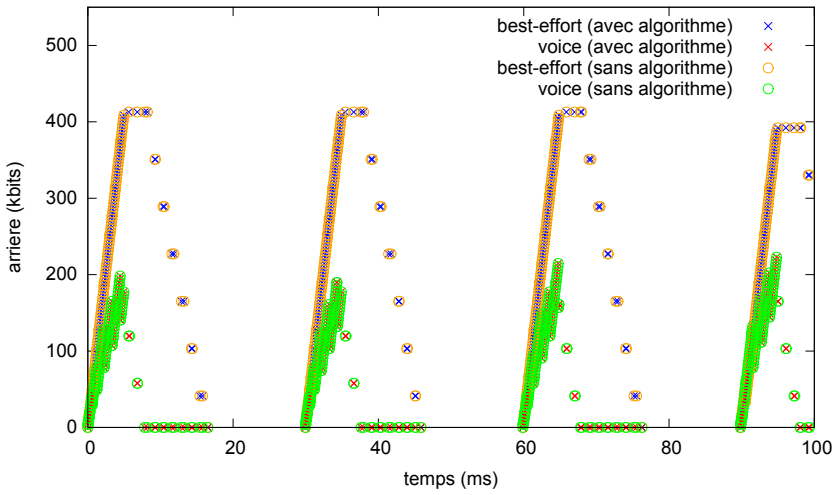
Nous avons vu au chapitre 3 que dans les scénarios proposés la concurrence se résume quasiment à une priorité stricte. Cet ordonnancement conduit notre algorithme à ne proposer aucune modification efficace de l'agrégation. Le système est néanmoins placé dans la meilleure configuration, soit agréger au seuil haut sur toutes les files.

La figure 4.4 illustre parfaitement ce propos puisque les résultats obtenus avec l'algorithme ou avec l'agrégation active sur toutes les classes au seuil maximal sont identiques. Le manque de concurrence est dû à des valeurs trop différentes de *backoff* entre les stations d'une part et d'autre part aux caractéristiques des flux choisis puisqu'ils consomment une grande quantité de bande passante.

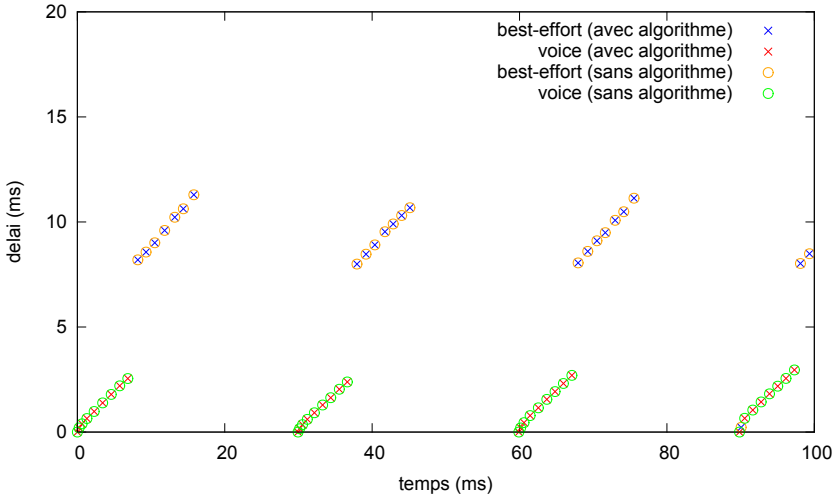
4.3.2.c Résultats avec un scénario comparable avec notre programme

Afin de nous situer dans un cas comparable à l'étude précédente avec notre programme, nous utilisons bien sûr la même configuration physique mais également des valeurs similaires de *backoff*. Similaires seulement puisqu'il est impossible d'obtenir une valeur fixe de *backoff* différente de 0 dans Modeler (comme vu au chapitre 3). Comme le tirage du *backoff* reste aléatoire nous choisissons d'effectuer des simulations sur plusieurs tirages (comme au chapitre 3). Avec 30 tirages nous pourrons obtenir des résultats représentatifs puisque l'influence de la variation du *backoff* sera moindre. Nous choisissons donc de fixer des valeurs de CW_{min} et CW_{max} au double de celles utilisées dans notre programme. Ainsi, nous obtenons des valeurs similaires en moyenne puisque le tirage du *backoff* suit une loi uniforme.

En tableau 4.6 on note des améliorations pour le flux *voice* similaires à celles observées précédemment. La dégradation du service subie par le flux *best-effort* est au contraire beaucoup plus importante ici. Après analyse du comportement du simulateur, cette dégradation est due à la différence entre le moment où la décision d'agréger ou non est prise et celui où notre algorithme intervient. En effet, l'algorithme est activé un instant avant la transmission d'un paquet alors que le système implémenté dans Modeler décide de former un agrégat dès l'instant ou un paquet est reçu. Ce comportement conduit à une accumulation de paquets



(a) Arriéré



(b) Délai

Figure 4.4 – Résultats de simulation de notre algorithme dans Riverbed Modeler sur un scénario issu de l'étude de performances précédente

non agrégés en mémoire qui seront transmis sans être agrégés même si l'algorithme les y autorise. C'est pourquoi les valeurs de délai et d'arriéré sont aussi élevées pour la classe *best-effort*.

4.4 Discussion et perspectives

4.4.1 Limites et compromis client/fournisseur

Lorsque la concurrence entre classes se résume à un simple ordonnancement à priorité stricte, cet algorithme active l'agrégation sur toutes les classes du système, et ce, au seuil haut. Aucun changement de l'agrégation sur les classes de priorités basses n'est alors efficace. Les classes de priorité inférieure n'ont tout simplement pas accès au medium tant que les autres n'ont pas vidé leur mémoire. Comme ces classes

	Écart moyen (en %)	Écart maximal (en %)
arriéré BE	80	150
arriéré VO	-5	-15
délai BE	317	905
délai VO	-9	-29

Tableau 4.6 – Évaluation de l'algorithme BE 500 VO 500 (facteur 0,4)

prioritaires ne dépasseront aucune exigence avec une mémoire vidée, les classes moins prioritaires pourront transmettre des agrégats.

Du point de vue du fournisseur, le meilleur cas (celui qui minimise l'arriéré) est de rester dans cet état avec toutes les classes qui agrègent indépendamment des exigences clients (si la mémoire est partagée). Une des améliorations possibles consiste à trouver un compromis entre les exigences des clients et du fournisseur. Comme les exigences clients sont déjà prises en considération, il est possible d'ajouter une condition à cet algorithme pour forcer l'agrégation sur toutes les classes si une exigence sur l'arriéré global est atteinte. Reste à gérer son incompatibilité potentielle avec les exigences client. Cette exigence peut prendre le pas sur les autres ou encore être incluse dans une recherche du minimum entre toutes les exigences à respecter.

Dans le cas d'une mémoire distincte par classe les possibilités sont multipliées pour la configuration du système. Il est tout à fait possible d'avoir des exigences en termes de pertes et d'arriéré différentes pour chaque classe. Chacune possédant sa propre mémoire, l'une peut déborder sans produire de pertes pour les autres. Chose impossible dans notre scénario avec une mémoire partagée puisque si une classe commence à occuper plus d'espace mémoire les autres qui risquent de subir des pertes également. Le respect des exigences est alors simplifié lorsque les mémoires sont distinctes.

L'utilisation de notre algorithme se révèle en particulier pertinente lorsque les exigences sont lâches sur les classes basses et restrictives sur les classes hautes. Par exemple si les délais et l'arriéré sont fortement contraints sur une classe *voice* alors que la classe *best-effort* accepte tous les délais et même quelques pertes de paquets.

4.4.2 Modalités d'implémentation de l'algorithme

4.4.2.a Distribué sur plusieurs stations physiques

L'algorithme proposé est prévu et évalué sur une seule station en tenant compte uniquement des données locales fournies par les classes de service. Il est néanmoins possible de distribuer cet algorithme sur plusieurs stations (une proposition d'algorithme est d'ailleurs fournie en annexe B). Il devient alors nécessaire de définir un protocole de communication entre les stations et même entre les stations virtuelles (chaque station équivaut à quatre stations virtuelles qui sont ses classes de service). À mon avis, la problématique de la résolution de conflit entre classes de même priorité doit également être adressée (sachant qu'elle peut être raffinée une première fois à l'aide des huit niveaux de priorité proposés par les couches supérieures).

Le comportement d'un algorithme avec des notifications entre stations peut produire une réaction en chaîne dans laquelle les stations en alerte désactivent l'agrégation sur d'autres qui entrent à leur tour en alerte sur leurs exigences. Les requêtes de changement de configuration peuvent alors être contradictoires et conduire à un système instable.

Une solution peut être de prendre des décisions égoïstes en fonction de son propre état en priorité car rappelons nous que ces stations sont en concurrence pour l'accès au medium. Quand au problème d'instabilité il peut être en partie réglé par l'utilisation de deux seuils sur les exigences (par exemple un à 70% et l'autre à 90%). L'un de ces seuils est utilisé pour déclencher la désactivation de l'agrégation et l'autre pour l'activer à nouveau (l'hystérésis bien connu des automaticiens).

4.4.2.b Vers une implémentation centralisée

Une autre solution pour régler les conflits entre stations est d'implémenter un algorithme centralisé basé sur une vue globale de l'état du réseau et de ses équipements. Cette solution est proposée en particulier dans les Software Defined Networks (SDN) où un contrôleur central s'assure de la bonne configuration du réseau. Cette solution offre également des possibilités d'amélioration de l'algorithme en choisissant par exemple d'agir sur les stations ou classes au trafic le plus important pour augmenter la réactivité du système et l'efficacité des actions menées tout en limitant leur impact négatif. Cela permettrait d'ajouter des indicateurs comme le nombre de clients sur le réseau sans fil.

Lorsque de nombreuses stations partagent le même réseau, utiliser un point de vue global permet de sélectionner les stations dont les paramètres seront modifiés. Ainsi, des modifications trop brutales et généralisées à l'ensemble des stations peuvent être évitées. Le problème de stabilité peut également être adressé plus simplement dès que le contrôleur de l'agrégation sur le réseau possède une vue entière de l'état des stations.

4.4.3 Un algorithme à adapter

Nous avons vu une implémentation possible de cet algorithme en prenant l'exemple des classes de services (stations virtuelles de 802.11n). Néanmoins, il est tout à fait possible de généraliser cet algorithme en remplaçant les classes par des clients différents ayant un rapport de priorité mesuré par exemple par la criticité de leur application (comme pour les classes 802.11e) ou encore définie par leur SLA. Dans ce cas une étude de faisabilité effectuée sur chaque technologie doit tenir compte des moyens pour le système de différencier les clients.

Nous avons exprimé ici une seule représentation de l'équité en cherchant d'une part à satisfaire les exigences du plus grand nombre de clients et d'autre part à respecter la classification définie dans 802.11e. Selon la volonté du fournisseur d'accès (qui a la mainmise sur l'équipement à configurer) la configuration du système peut être radicalement différente et l'algorithme également.

Il peut au choix laisser l'agrégation active pour tout le monde afin de minimiser l'arriéré dans ses équipement, activer l'agrégation uniquement pour des clients ayant souscrit un SLA plus exigeant, ou utiliser un algorithme comme le nôtre pour s'adapter aux exigences de chacun au travers de l'agrégation. L'algorithme alors établi pourra être basé sur nos travaux et peut être modifié en fonction de la finalité du fournisseur d'accès suivant les modalités précisées dans ce chapitre.

4.5 Conclusion

Après avoir défini un certain nombre d'exigences client et fournisseur, nous avons établi un algorithme de configuration des paramètres d'agrégation. Il est basé sur les recommandations issues de l'évaluation de performances du Chapitre 3. Cet algorithme se distingue de la littérature puisqu'il respecte la norme 802.11n en agissant uniquement sur des leviers prévus par celle-ci (activation/désactivation de l'agrégation et seuil de taille proposés).

L'originalité de cet algorithme est de proposer de désactiver l'agrégation en se basant sur des informations issues des concurrents. Il ajoute une part de collaboration entre les classes de service (ou les stations) puisqu'elles font des efforts les unes pour les autres.

L'algorithme proposé ici permet d'améliorer les performances des classes de priorités hautes de façon significative. Cependant, ce gain est obtenu au détriment des classes de priorité basse et dans certains cas les délais et arriérés peuvent augmenter fortement. L'exigence choisie est importante puisqu'elle conditionne l'efficacité de l'algorithme. Plus cette exigence sera restrictive, meilleures seront les performances pour les flux de haute priorité. En d'autres mots, il ne faut pas oublier que cette amélioration se paye cher sur les classes de priorité basse. Enfin, lorsque la mémoire est partagée, une augmentation excessive de l'arriéré pour une classe signifie plus de risques de pertes de paquets pour toutes les classes de la station.

Nous sommes conscients que la proposition effectuée dans ce chapitre mériterait d'être étoffée et évaluée sur un ensemble de scénarios plus exhaustif. Nous arrivons au terme de trois années de travaux qui nous ont permis de construire ce document et de mener un nombre important de simulations. Un approfondissement de l'évaluation de l'algorithme proposé sera ainsi amené à se poursuivre au même titre que d'autres perspectives comme celles développées dans la conclusion ci-après.

Conclusion et perspectives générales

Les travaux menés dans cette thèse permettent de comprendre à la fois la nature et les implications d'un système d'agrégation de paquets. En particulier, son influence sur la qualité de service offerte à une application est quantifiée. Lorsque l'accès au medium est concurrentiel, agréger signifie profiter au maximum de l'obtention du medium ce qui réduit la fréquence de remise en jeu de celui-ci. La conséquence potentielle est une augmentation provisoire des arriérés de traitement et des délais pour les stations concurrentes. Cette thèse a toutefois montré que l'augmentation est largement compensée par une amélioration de rendement puisqu'à la prochaine transmission les concurrents pourront réduire plus rapidement leur arriéré s'ils agrègent.

C'est à fin d'améliorer la fréquence de remise en jeu du medium que cette thèse propose en dernier lieu un algorithme de configuration dynamique des paramètres d'agrégation basé sur des exigences de qualité de service.

En Chapitre 1, nous avons proposé un recueil des différentes agrégations possibles dans le domaine des réseaux de télécommunications. Il nous permet de positionner l'agrégation considérée vis à vis des définitions variées que ce terme recouvre.

Au Chapitre 2, le système formalisé a été modélisé à l'aide de la théorie du calcul réseau ce qui nous permet d'étudier la concurrence entre deux paquets pour l'espace au sein d'un agrégat. Le pessimisme de ce modèle est évalué grâce à notre analyse approfondie de la norme 802.11n et grâce au simulateur Riverbed Modeler.

Dans le Chapitre 3, l'utilisation intensive de Riverbed Modeler nous permet d'étendre l'évaluation de performances à de nombreux scénarios. En particulier, nous nous intéressons ici à la concurrence pour l'accès au medium et à l'influence de l'agrégation dans ce contexte. Ainsi nous proposons un certain nombre de recommandations pour une configuration efficace du système d'agrégation.

Au Chapitre 4, nous proposons un algorithme de configuration dynamique des paramètres d'agrégation pour chaque classe de service en fonction d'exigences de qualité de service fournies par les classes concurrentes. Cette proposition modifie dans un sens le rapport entre les classes puisqu'elles peuvent collaborer.

En termes de perspectives à court terme, une évaluation approfondie de l'algorithme proposé semble nécessaire. En particulier, la stabilité du système vis à vis de modifications des mesures de délai et d'arriéré effectuée pourra être qualifiée. La proposition de créer une hystérésis qui permettrait d'assurer une certaine stabilité du système a été évoquée au chapitre 4. D'autre part, le comportement de cet algorithme pourrait être modifié afin d'intégrer un seuil temporel (chien de garde défini entre autre dans 802.15.3c). Il serait intéressant d'observer les modifications introduites sur le comportement du système avec ce seuil temporel.

La réalisation d'expérimentations était auparavant compliquée puisque basée sur des drivers spécifiques à implémenter sur des plate-formes de test. La technologie 802.11n ayant atteint une certaine maturité

aujourd'hui, les points d'accès sont répandus. Or certains paramètres intéressants apparaissent, par exemple, dans les *box* louées par les fournisseurs d'accès. On retrouve quelquefois ainsi la possibilité de configurer le seuil de taille de l'agrégation. Des résultats d'expérimentations permettraient d'une part de compléter notre analyse des performances de l'agrégation et d'autre part de comparer le comportement des systèmes d'agrégation implémentés avec ceux définis dans la norme et ceux implémentés dans Modeler.

Le réseau Internet abandonne progressivement ses spécificités, héritage d'un passé où les télécommunications étaient spécialisées pour un type de donnée (voix, vidéo ou fichiers). Il tend aujourd'hui vers plus de flexibilité. Une nouvelle architecture d'Internet semble émerger et des reconfigurations dynamiques des équipements seront nécessaires pour lui donner tout son sens. Ces configurations seront potentiellement assurées par des contrôleurs *Software Defined Networks*. Les travaux menés dans cette thèse s'intègrent parfaitement à ce point de vue puisque nous proposons une reconfiguration dynamique des paramètres des systèmes d'agrégation à l'interface des réseaux d'accès (sans fil) et de distribution. Cet algorithme de reconfiguration pourrait être piloté par des contrôleurs dispersés sur le réseau.

Le travail à effectuer pour implémenter notre algorithme de façon distribuée sur plusieurs contrôleurs SDN dépasse le cadre de cette thèse mais nous avons posé un certain nombre de problématiques à adresser et offert quelques pistes de réflexions. En ce sens notre but est atteint puisqu'il s'agissait principalement dans cette thèse d'apporter une expertise du système d'agrégation de paquets et de son influence sur le comportement du système dans lequel il est implémenté.

Liste des publications

Articles publiés dans des revues nationales

D. Breck, J.-P. Georges, T. Divoux. *Évaluation des performances de trafics agrégés 802.11n.* Journal Européen des Systèmes Automatisés, Hermès, 2013, 47 (4-8), pp.447-466.

Communications dans des conférences internationales avec actes

D. Breck, J.-P. Georges, T. Divoux. *Delay and backlog bounds for an aggregation system in wireless networks*. 19th IFAC World Congress, IFAC'14, Aug 2014, Cape Town, South Africa.

Communications dans des conférences nationales avec actes

D. Breck, J.-P. Georges, T. Divoux. *Étude du système d'agrégation de IEEE 802.11n et évaluation de son implémentation dans le simulateur Opnet*. 10ème Conférence Francophone de Modélisation, Optimisation et Simulation, MOSIM'14, Nov 2014, Nancy, France.

D. Breck, J.-P. Georges, T. Divoux. *Évaluation des performances de trafics agrégés 802.11n.* 5e Journées Doctorales / Journées Nationales MACS, JD-JN-MACS 2013, Jul 2013, Strasbourg, France.

Communications nationales sans actes

D. Breck, J.-P. Georges, T. Divoux. *Modélisation et évaluation des performances de la fonction d'agrégation 802.11n.* 6ème Journée du GT ARC (Automatique et Réseaux de Communication) du GdR MACS, Mar 2013, Paris, France.

D. Breck, J.-P. Georges, T. Divoux. *Évaluation des performances du système d'agrégation de 802.11n.* Réunion RGE (Réseau Grand Est) (GdR ASR), Oct 2013, Besançon, France.

Annexe A

Définitions et acronymes 802.11n

A.1 Paramètres nécessaires au calcul du temps d'émission

T_{L_PRE} : durée du préambule non-HT (rétrocompatibilité).

T_{L_SIG} : durée de la signalisation non-HT (rétrocompatibilité).

T_{HT_PRE} : durée du préambule HT (pour les stations 802.11n).

T_{HT_SIG} : durée de la signalisation HT (pour les stations 802.11n).

T_{SYM} : intervalle de temps entre deux symboles.

N_{SYM} : nombre de symboles OFDM.

$SigExt$: durée des extensions ajoutées au signal.

m : indique l'utilisation ou non de STBC (vaut 2 si utilisé et 1 sinon).

l : longueur de la trame.

N_{ES} : nombre d'encodeurs BCC pour les données.

N_{DBPS} : nombre de bits de données par symbole (varie en fonction du MCS et donc de la modulation entre autre).

T_{L_STF} : durée de la séquence courte du préambule non-HT.

T_{L_LTF} : durée de la séquence longue du préambule non-HT.

T_{PRE} : durée du préambule PLCP pour OFDM.

T_{SIG} : durée de la signalisation BPSK-OFDM.

A.2 Définitions, abréviations et acronymes

Cette annexe a été inspirée par la section 3 de (IEEE-802.11n, 2012).

AIFS : arbitration interframe space, le temps minimal suivant un medium occupé et précédant l'émission d'une nouvelle trame (utilisé dans HCF).

A-MPDU : aggregated-MPDU, une structure contenant de multiples MPDU transportées dans une seule PPDU.

A-MSDU : aggregated-MSDU, une structure contenant de multiples MSDU transportées dans une seule MPDU.

BCC : binary convolutional code, un algorithme de codage des données utilisé dans plusieurs modulations.

BSS : basic service set, un ensemble de stations qui sont en compétition pour le même medium.

CWmin et **CWmax** : les limites basse et haute de l'intervalle de tirage du backoff.

DCF : distributed coordination function, la plus connue des méthodes d'accès de 802.11.

DIFS : distributed (coordination function) interframe space, le temps d'attente minimum dans DCF.

HCF EDCA : hybrid coordination function (HCF) utilisant enhanced distributed control access (EDCA) comme méthode d'accès au medium. EDCA fournit une qualité de service en séparant les paquets dans des classes de service. Chaque classe de service possède des valeurs propres pour AIFS, CWmin et CWmax.

HT : high-throughput, un terme relatif à la nouvelle couche physique de 802.11n.

HT-mixed : high-throughput mixed, un format de PPDU compatible avec les précédentes versions de 802.11.

MCS : modulation and coding scheme, une spécification des paramètres de la couche physique de 802.11n (HT) qui consiste en une modulation et un taux de codage.

MSDU : medium access control (MAC) service data unit, l'unité d'information au sein de la couche MAC.

MPDU : MAC protocol data unit, l'unité de donnée échangée entre deux entités MAC.

non-HT : non high-throughput, un format de PPDU qui correspond à une couche physique d'une version antérieure à 802.11n émise par une station 802.11n.

OFDM : orthogonal frequency division multiplexing, une modulation utilisée entre autre dans 802.11n.

PPDU : physical layer convergence procedure (PLCP) protocol data unit, l'unité de donnée échangée entre deux entités physiques.

SIFS : short interframe space, le temps nominal requis par les couches MAC et physique pour débuter la réponse à une trame reçue.

STBC : space-time block coding, permet des débits plus élevés en utilisant de multiples chaînes de transmission côté émetteur et/ou côté récepteur.

TXOP : transmission opportunity, un intervalle de temps (dans HCF) au cours duquel une station a le droit d'initier une séquence d'échange de trames sur le medium.

Annexe B

Distribution de l'algorithme proposé sur plusieurs stations physiques

Données : exigences délai et arriéré $\widetilde{d}, \widetilde{b}$; mesures délai et arriéré d, b ; facteur d'anticipation p ; priorité de cette station virtuelle x ; notification d'alerte d'une autre station virtuelle $n_{x'}$; avec $x, x' \in 1, 2, 3, 4$ la priorité (1 étant le plus élevé).

Résultat : état de l'agrégation E_{agg} ; notification d'alerte aux autres stations virtuelles n_x

initialisation ($E_{agg} \leftarrow agg_{SH}, n_x \leftarrow faux$);

début

- **si** *($n_{x'}$=vrai) et ($x' > x$)* **alors**
  - $E_{agg} \leftarrow noagg$;
  - $c_{x'} \leftarrow c_{x'} + 1$;
  - reprendre l'algorithme du début ;
- **si** *($n_{x'}$=faux) et ($x' > x$)* **alors**
  - $c_{x'} \leftarrow c_{x'} - 1$;
  - **si** $(c_{x'} = 0)$ **alors** $E_{agg} \leftarrow agg_{SB}$;
  - reprendre l'algorithme du début;
- **si** $(n_x = vrai)$ *et* $(c_{x'} = 0)$ **alors**
  - $E_{agg} \leftarrow agg_{SH}$;
  - reprendre l'algorithme du début;
- `/* Observateur avec notifications */`
- **si** $(d \geq p\widetilde{d})$ *ou* $(b \geq p\widetilde{b})$ **alors**
  - $n_x = vrai$;
  - broadcast n_x;
  - reprendre l'algorithme du début;
- **si** $(d < p\widetilde{d})$ *et* $(b < p\widetilde{b})$ *et* $(n_x = vrai)$ **alors**
  - $n_x = faux$;
  - broadcast n_x;
  - reprendre l'algorithme du début;

Bibliographie

An, Xueli, Zhou Lan, R.V. Prasad, R. Hekmat, H. Harada et I. Niemegeers (2009). Performance analysis of the frame aggregation mechanisms in ieee 802.15.3c. In : *IEEE 20th International Symposium on Personal, Indoor and Mobile Radio Communications*. pp. 2095–2100.

Bellalta, B., J. Barcelo, D. Staehle, A. Vinel et M. Oliver (2012). On the Performance of Packet Aggregation in IEEE 802.11ac MU-MIMO WLANs. *IEEE Communications Letters*, **16**(10), 1588–1591.

Bhanage, G., D. Raychaudhuri et I. Seskar (2011). Backlogged queue based MAC frame aggregation. *Pervasive and Mobile Computing*, **7**(4), 449 – 466.

Bhanage, G. et al. (2009). Implication of mac frame aggregation on empirical wireless experimentation. IEEE Global Telecommunications Conference, GLOBECOM. pp. 1–7.

Bianchi, G. (2000). Performance analysis of the ieee 802.11 distributed coordination function. *IEEE Journal on Selected Areas in Communications*, **18**(3), 535–547.

Bourguiba, Manel, Kamel Haddadou et Guy Pujolle (2012). Packet aggregation based network i/o virtualization for cloud computing. *Computer Communications*, **35**(3), 309 – 319.

Breck, Damien, Jean-Philippe Georges et Thierry Divoux (2014). Étude du système d'agrégation de IEEE 802.11n et évaluation de son implémentation dans le simulateur Opnet. MOSIM, Nancy.

Bredel, M. et M. Fidler (2009). Understanding fairness and its impact on quality of service in ieee 802.11. In : *INFOCOM 2009, IEEE*. pp. 1098–1106.

Breuer, D., E. Weis, S. Gosselin, T. Mamouni et J. Torrijos (2013). Unified access and aggregation network allowing fixed and mobile networks to converge. In : *Optical Fiber Communication Conference and Exposition and the National Fiber Optic Engineers Conference (OFC/NFOEC), 2013*. pp. 1–3.

Castineyra, I., N. Chiappa et M. Steenstrup (1996). The Nimrod Routing Architecture. RFC 1992 (Informational).

Chakraborty, S., S. Kunzli et L. Thiele (2000). A general framework for analysing system properties in platform-based embedded system designs. IEEE Comput. Soc. pp. 190–195.

Chang, Cheng-Shang (2000). *Performance guarantees in communication networks*. Telecommunication networks and computer systems. Springer. London ; New York.

Chebrolu, K. et R.R. Rao (2006). Bandwidth aggregation for real-time applications in heterogeneous wireless networks. *IEEE Transactions on Mobile Computing*, **5**(4), 388–403.

Choudhury, R, Ye Chen et Steve Emeott (2006). Performance analysis of data aggregation techniques for wireless lan. In : *Proceedings of IEEE Globecom*. Citeseer. pp. 1958–1962.

Cruz, R.L. (1991). A calculus for network delay. 1 Network elements in isolation. *IEEE Transactions on Information Theory*, **37**(1), 114–131.

Dely, P., A. Kassler, N. Bayer, H.-J. Einsiedler et D. Sivchenko (n.d.). In : *Seventh International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)*.

Duran-Faundez, Cristian (2009). Energy-efficient image transmission for wireless camera sensor networks. PhD thesis. Université Henri Poincaré - Nancy I.

Erman, Bilgehan et Z.John Zhao (2011). Quantifying qos benefits in the aggregation network for internet video services. *Bell Labs Technical Journal*, **16**(2), 63–77.

ETS 300 147 (1992). Transmission and Multiplexing (TM) ; Synchronous Digital Hierarchy (SDH) Multiplexing structure.

Fang, Hao et E.W. Zegura (2000). On scalable QoS routing : performance evaluation of topology aggregation. In : *INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies.*. Vol. 1. pp. 147–156.

Ginzburg, Boris et Alex Kesselman (2007). Performance analysis of a-MPDU and a-MSDU aggregation in IEEE 802.11n. IEEE Sarnoff Symposium. pp. 1–5.

Hong, Jung Ha et K. Sohraby (2010). On modeling, analysis, and optimization of packet aggregation systems. *IEEE Transactions on Communications*, **58**(2), 660–668.

Hui, J. et P. Thubert (2011). The Nimrod Routing Architecture. RFC 6282 (Standards Track).

IEEE-802.11n (2012). 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.

IEEE-802.15.3c (2009). Standard for information technology. Telecommunications and information exchange between systems local and metropolitan area networks. specific requirements part 15.3 : Wireless medium access control (MAC) and physical layer (PHY) specifications for high rate wireless personal area networks (WPANs) amendement 2 : Millimeter-wave-based alternative physical layer extension.

Intanagonwiwat, C., D. Estrin, R. Govindan et J. Heidemann (2002). Impact of network density on data aggregation in wireless sensor networks. In : *22nd International Conference on Distributed Computing Systems*. pp. 457–458.

IUT-R-M.1645 (2003). Series M : Mobile, radiodetermination, amateur and related satellite services. Framework and overall objectives of the future development of IMT-2000 and systems beyond IMT-2000.

IUT-T-G.707/Y.1322 (2007). Series G : transmission systems and media, digital systems and networks. Series Y : Global information infrsatructure, Internet protocol aspqects and next-generation networks. Network node interface for the synchronous digital hierarchy (SDH).

IUT-T-I.361 (1999). Series I : Integrated services digital networks. B-ISDN ATM layer specification.

Iwamura, M., K. Etemad, Mo-Han Fong, R. Nory et R. Love (2010). Carrier aggregation framework in 3gpp lte-advanced [wimax/lte update]. *IEEE Communications Magazine*, **48**(8), 60–67.

Iwata, A., H. Suzuki, R. Izmailov et B. Sengupta (1998). Qos aggregation algorithms in hierarchical atm networks. In : *IEEE International Conference on Communications*. Vol. 1. pp. 243–248 vol.1.

Johansson, K., J. Bergman, D. Gerstenberger, M. Blomgren et A. Wallen (2009). Multi-carrier hspa evolution. In : *IEEE 69th Vehicular Technology Conference (VTC)*. pp. 1–5.

Khattab, Tamer, Amr Mohamed, Ayman Kaheel et Hussein Alnuweiri (2002). Optical packet switching with packet aggregation. In : *IEEE International Conference on Software, Telecommunications, and Computer Networks (SOFTCOM)*.

Kim, Youngsoo, Sunghyun Choi, Kyunghun Jang et Hyosun Hwang (2004). Throughput enhancement of ieee 802.11 wlan via frame aggregation. In : *IEEE 60th Vehicular Technology Conference*. Vol. 4. pp. 3030–3034 Vol. 4.

Korkmaz, Turgay et Marwan Krunz (2000). Source-oriented topology aggregation with multiple qos parameters in hierarchical networks. *ACM Trans. Model. Comput. Simul.*, **10**(4), 295–325.

Krishnamachari, B., D. Estrin et S. Wicker (2002). The impact of data aggregation in wireless sensor networks. In : *22nd International Conference Workshops on Distributed Computing Systems*. pp. 575–578.

Kuppa, Srikant et G.R. Dattatreya (2006). Modeling and analysis of frame aggregation in unsaturated WLANs with finite buffer stations. IEEE International Conference on Communications. pp. 967–972.

Le Boudec, Jean-Yves et Patrick Thiran (2001). *Network calculus : a theory of deterministic queuing systems for the Internet*. number 2050 In : *Lecture notes in computer science*. Springer. Berlin ; New York.

Lee, W.C. (1995). Spanning tree method for link state aggregation in large communication networks. In : *INFOCOM '95*. pp. 297–302 vol.1.

Li, Tianji, Qiang Ni, David Malone, Douglas Leith, Yang Xiao et Thierry Turletti (2009). Aggregation with fragment retransmission for very high-speed wlans. *IEEE/ACM Trans. Netw.*, **17**(2), 591–604.

Lin, Yuxia et Vincent W. S. Wong (2006). WSN01-1 : frame aggregation and optimal frame size adaptation for IEEE 802.11n WLANs. IEEE Global Communication Conference. pp. 1–6.

Lu, Kejie, Dapeng Wu, Yi Qian, Yuguang Fang et R.C. Qiu (2007). Performance of an aggregation-based mac protocol for high-data-rate ultrawideband ad hoc networks. *IEEE Transactions on Vehicular Technology*, **56**(1), 312–321.

Lui, King-Shan, K. Nahrstedt et Shigang Chen (2004). Routing with topology aggregation in delay-bandwidth sensitive networks. *IEEE/ACM Transactions on Networking*, **12**(1), 17–29.

Mao, Shiwen et Shivendra S. Panwar (2006). A survey of envelope processes and their applications in quality of service provisioning. *Commun. Surveys Tuts.*, **8**(3), 2–20.

Maqhat, B., M.D. Baba et R.A. Rahman (2012). A-msdu real time traffic scheduler for ieee802.11n wlans. In : *IEEE Symposium on Wireless Technology and Applications (ISWTA)*. pp. 286–290.

McCreary, S. et k. claffy (2000). Trends in wide area IP traffic patterns - A view from Ames Internet Exchange. In : *ITC Specialist Seminar*. Monterey, CA.

Oliveira, T., S. Mahadevan et D.P. Agrawal (2011). Handling network uncertainty in heterogeneous wireless networks. In : *IEEE INFOCOM*. pp. 2390–2398.

Pedersen, K.I., F. Frederiksen, C. Rosa, H. Nguyen, L.G.U. Garcia et Yuanye Wang (2011). Carrier aggregation for LTE-advanced : functionality and performance aspects. *IEEE Communications Magazine*, **49**(6), 89–95.

Perrig, Adrian, Robert Szewczyk, J. D. Tygar, Victor Wen et David E. Culler (2002). Spins : Security protocols for sensor networks. *Wirel. Netw.*, **8**(5), 521–534.

Petrovic, D., R.C. Shah, K. Ramchandran et J. Rabaey (2003). Data funneling : routing with aggregation and compression for wireless sensor networks. In : *IEEE International Workshop on Sensor Network Protocols and Applications*. pp. 156–162.

PNNI 1.0 (1996). Private Network-to-Network Interface Specification Version 1.0.

Rajagopalan, R. et P.K. Varshney (2006). Data-aggregation techniques in sensor networks : A survey. *IEEE Communications Surveys Tutorials*, **8**(4), 48–63.

Ramaboli, Allen L., Olabisi E. Falowo et Anthony H. Chan (2012). Bandwidth aggregation in heterogeneous wireless networks : A survey of current approaches and issues. *Journal of Network and Computer Applications*, **35**(6), 1674 – 1690.

Rekhter, Y., T. Li et S. Hares (2006). A Border Gateway Protocol 4 (BGP-4). RFC 4271 (Draft Standard). Updated by RFCs 6286, 6608, 6793.

Riverbed (2014). Modeler 17.5. `http://www.riverbed.com`.

Saif, Anwar et al. (2011). An enhanced a-MSDU frame aggregation scheme for 802.11n wireless networks. *Wireless Personal Communications*, **66**(4), 683–706.

Saif, Anwar et Mohamed Othman (2013). Sra-msdu : Enhanced a-msdu frame aggregation with selective retransmission in 802.11n wireless networks. *Journal of Network and Computer Applications*, **36**(4), 1219 – 1229.

Schmitt, J. B., F. A. Zdarsky et M. Fidler (2008). Delay Bounds under Arbitrary Multiplexing : When Network Calculus Leaves You in the Lurch.... IEEE. pp. 1669–1677.

Selvam, T et S Srikanth (2010). A frame aggregation scheduler for IEEE 802.11n. IEEE. pp. 1–5.

Sinha, Rishi, Christos Papadopoulos et John Heidemann (2007). Internet packet size distributions : Some observations. Technical Report ISI-TR-2007-643. USC/Information Sciences Institute. Orignally released October 2005 as web page `http://netweb.usc.edu/~rsinha/pkt-sizes/`.

Skordoulis, D. et al. (2008). IEEE 802.11n MAC frame aggregation mechanisms for next-generation high-throughput WLANs. *IEEE Wireless Communications*, **15**(1), 40–47.

Thiele, L., S. Chakraborty et M. Naedele (2000). Real-time calculus for scheduling hard real-time systems. Vol. 4. Presses Polytech. Univ. Romandes. pp. 101–104.

UIT-T-G.708 (1999). Series G : Systèmes et supports de transmission, systèmes et réseaux numériques. Interface de noeud de réseau infra STM-0 pour la hierarchie numérique synchrone (SDH).

Uludag, Suleyman, King-Shan Lui, Klara Nahrstedt et Gregory Brewster (2007). Analysis of topology aggregation techniques for qos routing. *ACM Comput. Surv.*

Wang, Chih-Yu et Hung-Yu Wei (2009). IEEE 802.11n MAC enhancement and performance evaluation. *Mobile Networks and Applications*, **14**(6), 760–771.

Wu, Jiyan, Jingqi Yang et Junliang Chen (2013). Loss tolerant bandwidth aggregation for multihomed video streaming over heterogeneous wireless networks. In : *IEEE Global Communications Conference (GLOBECOM)*. pp. 2956–2962.

Xie, Jing et Yuming Jiang (2010). A network calculus approach to delay evaluation of ieee 802.11 dcf. In : *IEEE 35th Conference on Local Computer Networks (LCN)*. pp. 560–567.

Yuan, Guangxiang, Xiang Zhang, Wenbo Wang et Yang Yang (2010). Carrier aggregation for lte-advanced mobile communication systems. *IEEE Communications Magazine*, **48**(2), 88–93.

Zhang, Qi, V.B. Iversen et F.H.P. Fitzek (2008). Throughput and delay performance analysis of packet aggregation scheme for prma. In : *IEEE Wireless Communications and Networking Conference*. pp. 1380–1384.