

Mémoire de fin d'études
ENS Louis-Lumière



Étude de la possibilité du transport sans fil d'un réseau
audio-numérique

Léo GRISLIN

Directeur interne : Franck JOUANNY

Directeur externe : Jémérie KOKOT

Rapporteur : Frank GILLARDEAUX

Session de Juin 2015

Résumé

Le développement des réseaux informatiques appliqués à l'audiovisuel, tel qu'à travers les technologies Dante et Ravenna, a permis de simplifier certaines installations, notamment grâce aux possibilités et à la simplicité du routage du signal entre les appareils audio. Cependant, le maintien des contraintes de câblage laisse ces réseaux, dans la plupart des cas, en concurrence avec d'autres technologies tout aussi légères dans leur déploiement. Bien que cela soit déjà une évolution majeure dans le transport du signal audio, proposer une solution sans fil serait un moyen d'élargir encore les possibilités, et d'ouvrir l'utilisation d'un réseau informatique dédié à l'audio à de nouvelles applications.

Ce mémoire étudie la possibilité du transport d'un réseau audionumérique à travers un réseau informatique sans fil, et traite en particulier de la problématique du transport de données audionumériques non compressées en temps réel. Cette discussion se base sur l'étude des standards en vigueur, menée en parallèle de tests pratiques.

Mots-clés

Réseaux audionumériques, AES67, Dante, Ravenna, RTP, IEEE 1733, Réseaux sans fil, IEEE 802.11, Wi-Fi

Abstract

The use of computer networks applied to pro-audio, such as Dante and Ravenna technologies, has already simplified some audio deployments, thanks to the signal routing possibilities and its simplicity. However, if it is a major step-up in audio transport, wiring is still a constraint and a wireless solution would be a great improvement, allowing the use of such technologies in a wider and maybe whole new area of application.

This Masters dissertation studies the possibility of transporting a digital audio network. Specifically, it discusses the problem of the real-time transportation of uncompressed digital audio data, through studies of standards and practical tests.

Keywords

Audio networking, AES67, Dante, Ravenna, RTP, IEEE 1733, Wireless networking, IEEE 802.11, Wi-Fi

*« Ecco Dite », dicendo, « ed ecco il loco
ove convien che di fortezza t'armi ».*

DANTE ALIGHIERI,
Commedia, Inferno, Canto XXXIV

Remerciements

J'aimerais remercier toutes les personnes qui ont contribué de manière directe ou non à l'élaboration de ce mémoire.

Je souhaite tout particulièrement remercier William LEVEUGLE et Léo ROSSIROTH, pour m'avoir intégré à leur « groupe de recherche », pour tout le temps qu'ils m'ont consacré pendant cette période et pour leurs relectures avisées.

Je remercie infiniment Jérémie KOKOT, pour sa forte implication et pour les nombreuses personnes qu'il m'a fait rencontrer.

Je remercie également toute l'équipe de DE PRÉFÉRENCE de m'avoir mis à disposition leur matériel et leur savoir ; merci pour votre accueil, vos conseils et votre gentillesse. Merci aussi à l'équipe de YASTA.

Je souhaite aussi remercier Micha RIVAULT pour son expertise sur les réseaux sans fil et sa collaboration.

Merci aussi à Franck JOUANNY pour les problématiques qu'il a su soulever lors de nos discussions.

Merci à Paul PAYEN DE LA GARANDERIE de m'avoir gracieusement offert un ordinateur pour le transformer en CARTON.

Merci à mes oncles informaticiens Didier GREINER et Rémi PORCEDDA, pour m'avoir aidé à initier mes recherches.

Merci Frank GILLARDEAUX de m'avoir fait rencontrer Jérémie KOKOT et l'équipe de DE PRÉFÉRENCE.

Je remercie Gérard PELÉ et Laurent MILLOT pour leurs conseils.

Merci enfin à tous mes camarades de classe, en compagnie desquels j'ai passé trois superbes années, et qui ont, sans le savoir, contribué à la rédaction de ce mémoire.

Table des matières

Liste des acronymes	xi
Introduction	xiii
Avant-propos	xv
1 Évolutions depuis les précédents travaux	1
1.1 Les réseaux audio-numériques et le sans-fil	2
1.1.1 Développement de l’AES67	2
1.1.2 Aperçu des nouveaux systèmes de diffusion sans fil grand public	3
1.2 Exploration de l’IEEE 802.11	5
1.2.1 Objectifs de la nouvelle norme IEEE 802.11ac	6
1.2.2 Débits proposés	16
1.2.3 L’IEEE 802.11 et le multimédia	17
2 Caractérisation des liaisons existantes et normalisation des performances minimales requises pour qu’une liaison soit apte à passer un flux AES67	19
2.1 Horloge	20
2.1.1 Installations traditionnelles – rappel	20
2.1.2 Synchronisation ou syntonisation	21
2.1.3 Qualité de la synchronisation à travers une liaison sans fil	22
2.1.4 Syntonisation externe	29
2.2 Besoins d’un réseau AES67	31
2.2.1 Paquet RTP – conteneur de l’audio	31

2.2.2	Calcul de débit	34
2.2.3	Latence de lecture – Mémoire tampon de réception	34
2.2.4	Résumé	36
2.3	netem : simulateur de liaisons	36
2.3.1	Présentation de netem	37
2.3.2	Construction d’une NETEM BOX	39
2.3.3	Premiers tests avec netem	44
2.4	Première liaison sans fil avec syntonisation externe	51
3	Optimisation des réseaux sans fil – améliorations possibles de la couche physique	55
3.1	Optimisation des réseaux sans fil	56
3.1.1	Rappel des mécanismes problématiques pour notre application .	57
3.1.2	Calcul du temps de parcours d’un paquet	59
3.1.3	Calcul de débit effectif de données	66
3.1.4	Trames agrégées	68
3.2	Liaison <i>full-duplex</i> , une alternative au CSMA/CA	73
3.2.1	Présentation de la solution <i>full-duplex</i>	74
3.2.2	Antennes utilisées pour la liaison	74
3.2.3	Le protocole Nstreme dual	76
3.2.4	Mise en place de la liaison pour les tests	77
3.2.5	Paramètre d’agrégation	79
3.2.6	À la recherche de la cause d’instabilité	84
3.3	Transport sans fil de la synchronisation	87
3.3.1	Transport sur le réseau sans fil	87
3.3.2	Dispositif de syntonisation externe sans fil	87
A	Algorithme simplifié du CSMA/CA	90
B	Mise en place d’une méthode de contrôle de la qualité d’une transmission audio	91

C	Fiche technique des antennes	93
D	Capture du spectre effectuée avec l’outil AirView : local de De Préférence	94
	Conclusion	97
	Bibliographie	98

Liste des acronymes

A-MSDU Aggregation Mac Service Data Unit
AES Audio Engineering Society
AVB Audio Video Bridge
CSMA Carrier Sense Multiple Access
DCF Distributed Coordination Function
DIFS Distributed Inter-Frame Space
GPS Global Positioning System
gPTP generalized Precision Time Protocol
IEEE Institute of Electrical and Electronics Engineers
IP Internet Protocol
LAN Local Area Network
MAC Media Access Control
MADI Multichannel Audio Digital Interface
MCS Modulation and Coding Scheme
MIMO Multiple Inputs Multiple Outputs (6.7.2, [14])
MPDU Mac Protocol Data Unit
NAV Network Allocation Vector
OSI Open Systems Interconnection
PLCP Physical Layer Convergeance Protocol
PTP Precision Time Protocol
RTP Real-time Transport Protocol
SIFS Short Interframe Space
TCP Transfert Control Protocol
UDP User Datagram Protocol
WAN Wide Area Network
WLAN Wireless LAN

Introduction

Depuis leur apparition, les technologies des réseaux informatiques dédiés au transport de l'audio, que l'on appelle plus communément « réseaux audionumériques » ne cessent de faire de nouveaux adeptes grâce à leur flexibilité de déploiement, leur fiabilité et leurs performances.

De nombreuses technologies de réseaux audionumériques sont apparues au fil des années depuis l'apparition de COBRANET en 1996. Très majoritairement propriétaires, et toujours incompatibles entre elles, 2013 marque un tournant avec la publication de l'AES67, qui a la volonté de standardiser le transport de l'audio sur un réseau IP. La standardisation d'une technologie accélère la recherche et le développement, par exemple en mettant à contribution des associations de fabricants. Le cas des réseaux audionumériques est particulier dans la mesure où il fédère à la fois les fabricants de matériel audio et les fabricants de matériel réseau. On peut alors aussi s'attendre à bénéficier des innovations venant de ces derniers concernant les méthodes de transport et les protocoles de transmission. Pour ces raisons, il semblerait bien que nous ne soyons qu'à l'aube des réseaux audionumériques.

Dans cette optique, la prochaine amélioration pourrait être le transport sans fil des réseaux audionumériques. Cette avancée serait en effet un moyen de démarquer de manière évidente cette nouvelle technologie, car, bien que les technologies audionumériques traditionnelles (comme le MADI) soient moins flexibles dans leur déploiement, elles n'en demeurent pas significativement plus lourdes en câblage qu'un réseau AES67 filaire¹. L'utilisation d'un réseau AES67 sans fil serait d'une grande légèreté, et pourrait dans certains cas appuyer le fait d'avoir recours à un réseau audionumérique.

1. le MADI permet de faire passer 64 canaux audio dans un câble coaxial

Dans la microphonie sans fil, il se peut que les constructeurs s'orientent aussi vers les réseaux audionumériques. Le nombre de canaux de transmission radio disponibles étant de plus en plus restreint, ces systèmes de microphones commencent déjà à investir les bandes ISM². L'utilisation de ces bandes requiert une excellente gestion du partage entre les différents utilisateurs, toujours plus nombreux. Une solution que l'on pourrait envisager alors serait l'utilisation d'un réseau audionumérique pour multiplexer les canaux dans le cas des microphones sans fil.

Ce mémoire se propose d'étudier si les solutions sans fil existantes pour les réseaux informatiques sont capables ou non de transporter un réseau audionumérique. Plus précisément, les réseaux IEEE 802.11 répondent-ils aux besoins d'un réseau audionumérique sur IP ?

Cette question a déjà été soulevée à deux reprises dans les travaux de William LEVEUGLE et Léo ROSSI-ROTH. Ce mémoire se propose d'en être la continuité. Nous nous intéresserons en particulier à la capacité des réseaux IEEE 802.11 à supporter le transfert de données audionumériques non compressées en temps réel.

Dans la première partie, nous examinerons les récentes évolutions technologiques et normatives, ce qui sera aussi l'occasion d'effectuer quelques rappels sur certains termes techniques. Dans la deuxième partie nous essayerons de caractériser les besoins en performance d'une liaison pour transporter un réseau audionumérique, et nous étudierons le cas d'une liaison sans fil standard IEEE 802.11. Enfin nous envisagerons dans quelle mesure nous pouvons optimiser une liaison sans fil pour qu'elle s'adapte au mieux à nos besoins.

2. bandes « Industriel, Scientifique et Médical » ; ce sont les bandes 2,4GHz et 5GHz, communément utilisées par le « Wi-Fi »

Avant-propos

Ce mémoire prend la suite de deux autres mémoires effectués à l'ENS Louis-Lumière. Il s'agit des travaux de William LEVEUGLE³ et de Léo ROSSI-ROTH⁴. Nous bénéficions donc des résultats obtenus dans les précédentes études pour mener à bien nos recherches.

Cet avant-propos a pour but de situer notre démarche par rapport à celles de William LEVEUGLE et de Léo ROSSI-ROTH, afin de tirer le meilleur parti des avancées de leurs travaux.

Cadre de la recherche

Dans le but d'une application dans le secteur professionnel, nos exigences pour les réseaux audionumériques sans fil sont élevées. Dans le meilleur des cas il faudrait que les performances soient identiques à celles proposées par les réseaux câblés, c'est-à-dire :

- faible latence
- audio non compressé
- transport d'un grand nombre de canaux audio

Nous détaillerons les caractéristiques envisagées dans le cadre de ce mémoire dans la deuxième partie.

3. W. LEVEUGLE, "Des possibilités d'évolution vers le sans-fil des réseaux audionumériques pour la sonorisation - Cas des Wireless Lan", mém.de maîtr., École Nationale Supérieure Louis-Lumière, 2012

4. L. ROSSI-ROTH, "Étude des possibilités d'utilisation d'un réseau sans-fil pour le transport d'un réseau audionumérique", mém.de maîtr., École Nationale Supérieure Louis-Lumière, 2014

Historique des précédents travaux – résultats acquis

En 2012, William LEVEUGLE était le premier à étudier en profondeur la compatibilité des réseaux audionumériques avec les réseaux sans fil. Dans un premier temps, il présente de manière détaillée le fonctionnement des réseaux informatiques classiques, les protocoles décrits dans les normes IEEE 802⁵ et les différents périphériques composant un réseau. Il s’agit d’abord de donner au lecteur les bases pour la compréhension des enjeux techniques. William LEVEUGLE passe ensuite en revue les différents réseaux audionumériques répandus à l’époque, avant de choisir, par défaut, la technologie DANTE pour ses tests. DANTE est alors la seule technologie entièrement compatible avec le protocole IP (troisième couche du modèle OSI) et donc *a priori* compatible avec un réseau IEEE 802.11.

Au cours de ses tests, William LEVEUGLE se heurte à des problèmes de synchronisation d’horloge entre les périphériques du réseau ce qui lui empêche d’obtenir une liaison audio stable sur la durée.

C’est en grande partie ce problème que Léo ROSSI-ROTH tente de résoudre dans son mémoire. Ce dernier entreprend une étude plus approfondie du protocole de synchronisation utilisé (PTP : *Precision Time Protocol*) en se basant sur la toute récente technologie RAVENNA (sous licence libre contrairement à DANTE). Aussi, le fait que RAVENNA soit la première technologie compatible AES67 permet à Léo ROSSI-ROTH d’étudier ses mécanismes avec une grande transparence. La première partie de son mémoire est plus vouée à l’appréhension de l’Horus (appareil RAVENNA de MERGING) et des différents drivers dont le fonctionnement peut s’avérer obscur. Il s’agit dans un premier temps de confirmer les hypothèses faites par William LEVEUGLE.

La partie centrale de son travail porte sur la question de la synchronisation, et de sa mise en œuvre dans les réseaux audio sans fil. Il analyse en détail les protocoles PTPv2 et gPTP, et leur mise en œuvre à travers les réseaux sans fils.

Ses conclusions ne lui permettent pas d’acheminer correctement les signaux de synchronisation sur un réseau sans fil classique (norme IEEE 802.11), mais il propose

5. normes qui décrivent les deux couches les plus basses du modèle OSI

un certain nombre de solutions alternatives. Parmi ces solutions, la liaison *full-duplex* fait l'objet de quelques tests et porte ses fruits, Léo ROSSI-ROTH parvient à plusieurs reprises à transmettre un signal stéréo sans erreur⁶. Il évoque également dans son mémoire la possibilité d'utiliser une synchronisation par GPS ; méthode sans fil et suffisamment précise pour notre application.

Plusieurs résultats ont été obtenus par William LEVEUGLE et Léo ROSSI-ROTH dans leur mémoire respectif. Nous n'hésiterons pas à y faire référence en dirigeant le lecteur vers ces travaux pour des explications plus détaillées.

Les tests précédemment effectués nous permettent de partir des assertions suivantes :

- le protocole PTPv2 préconisé dans l'AES67 pour la synchronisation n'est pas compatible avec les implémentations actuelles de la norme IEEE 802.11
- le protocole gPTP (décrit dans les normes AVB), également mentionné dans le standard AES67, est *a priori*, compatible avec la norme IEEE 802.11. Cependant, il n'est pas supporté matériellement à l'heure actuelle

Il demeure quelques inconnues auxquelles nous tenterons de répondre dans ce mémoire :

- en mettant le problème de la synchronisation de côté, la liaison IEEE 802.11 est-elle suffisamment robuste et fiable pour permettre le transport en temps réel de plusieurs canaux audio ?
- est-il possible d'opérer sur les couches physiques de la norme IEEE 802.11 pour les rendre compatibles à l'utilisation du PTP ?
- est-il possible de transmettre les données de synchronisation par un autre biais, comme avec plusieurs récepteurs GPS ?

6. liaison réalisée lors de sa soutenance le 17 juin 2014, avec le morceau *Summertime* de Big Brother & The Holding Company

Utilisation de la technologie Dante

Tous les tests que nous avons réalisés dans le cadre de ce mémoire, se sont déroulés dans les locaux de la société DE PRÉFÉRENCE qui a bien voulu nous accueillir pendant toute la durée du mois d'avril. Nous avons à disposition une grande quantité de matériel DANTE et de différentes générations.

L'utilisation de matériel DANTE pour nos tests est en fait une contrainte historique à laquelle nous sommes confrontés. Bien que DANTE s'avère être une « boîte noire » (Audinate communique peu sur les mécanismes utilisés), il s'agit de la technologie de réseau audionumérique la plus répandue actuellement. Ce que l'on sait du fonctionnement de DANTE nous permettra quand même de faire certains rapprochements avec les recommandations de l'AES67.

Nous commençons cependant notre étude, en prenant en considération que DANTE diffère du standard AES67 quant au protocole de synchronisation ; DANTE utilise le protocole PTPv1, tandis que l'AES67 recommande l'utilisation du PTPv2.

Chapitre 1

Évolutions depuis les précédents travaux

Plusieurs éléments se sont développés depuis les travaux effectués par William Leveugle en 2012 et ceux de Léo ROSSI-ROTH en 2014. Les normes, les technologies, le matériel évoluent très rapidement, même en l'espace d'un an, donc à l'échelle de temps d'un mémoire . . . Dans les précédents travaux sur la question, les objectifs que nous souhaitons atteindre se sont avérés gourmands en performances et proche des limites imposées par les technologies de réseaux sans fil actuelles. La quête de WLAN¹ plus performant concerne de nombreuses applications et fait l'objet des recherches menées par le groupe de travail 802.11 très actif à l'heure actuelle².

Les avancées technologiques concernent également l'évolution du matériel réseau qui, rappelons-le, n'est pas obligatoirement conforme aux normes et recommandations du groupe 802.11. Des technologies propriétaires voient donc régulièrement le jour comme par exemple les solutions proposées par UBIQUITI ou MIKROTIK, dont certaines ont déjà été expérimentées les années précédentes.

Il convient donc dans cette partie de rendre compte des dernières évolutions normatives et matérielles dont nous tiendrons compte dans la suite du mémoire, tout en sachant que cette base risque d'être dépassée entre temps.

1. WLAN : *Wireless Local Area Network*, réseau local sans fil

2. <http://www.ieee802.org/11/>

1.1 Les réseaux audio-numériques et le sans-fil

1.1.1 Développement de l’AES67

L’arrivée du standard AES67 semble bousculer les acteurs du monde audio professionnel. De plus en plus de constructeurs intègrent des modules compatibles AES67 dans leurs produits, certaines technologies préexistantes se rendent même compatibles avec le standard AES67, à l’instar de DANTE ayant annoncé sa prochaine compatibilité le 4 février 2014³.

Parmi les avantages liés à une standardisation des réseaux audionumériques, on compte l’interopérabilité entre constructeurs qui a fait ses preuves lors de la *AES67 Interoperability PlugFest* qui s’est déroulée à Munich du 27 au 30 octobre 2014. Le principe de cet événement était de déterminer dans quelle mesure des appareils de constructeurs différents conformes au standard AES67 étaient capables de communiquer entre eux. L’expérience s’est avérée concluante, puisque seules 6 configurations émetteur/récepteur des 225 testées n’ont pas fonctionné.

Il est difficile de donner un chiffre du nombre de produits compatibles AES67 actuellement en développement. Nous pouvons cependant déclarer que la *PlugFest* a réuni dix fabricants de matériel, venus tester l’interopérabilité de seize produits :

- ALC NetworX GmbH
- Archwave AG
- Axia Audio
- Digigram SA
- DirectOut GmbH
- Georg Neumann GmbH
- Lawo AG
- Merging Technologies S.A.
- SOUND4
- Telos Systems Inc.

Cette interopérabilité a pour conséquence de stimuler le développement des réseaux audionumériques, puisqu’elle permet en quelque sorte de rassurer les constructeurs de matériel audio, en assurant le fait que leurs produits trouveront facilement leur place sur le *réseau*.

3. <https://www.audinate.com/article/audinate-announces-support-aes67-standard>

1.1.2 Aperçu des nouveaux systèmes de diffusion sans fil grand public

Il nous paraît intéressant de mentionner dans cette partie les nouveautés concernant les dispositifs de transmission audio sans fil domestiques. D'autant plus que certaines technologies s'apparentent à de véritables réseaux, objet de ce mémoire.

1.1.2.1 Enceintes et écouteurs sans fil

Très répandus dans notre quotidien depuis l'apparition des téléphones portables puis des smartphones, les appareils de diffusions de la voix (oreillettes Bluetooth™) puis de contenu multimedia (casques audio sans fil, enceintes portables) connaissent de perpétuelles évolutions. Les mises à jour régulières du standard bluetooth visent à améliorer la portée, la consommation d'énergie et la qualité en augmentant les débits de transmission.

Bien que la technologie du bluetooth soit plébiscitée pour les systèmes audio domestiques il convient de rappeler que cette solution n'est pas envisageable, comme l'avait expliqué William LEVEUGLE ([14], 6.1.1), du fait de sa faible portée (pas plus de 20m) et de sa faible bande passante empêchant le transfert de signaux audionumériques non compressés.

1.1.2.2 Enceintes « Wi-Fi »

De plus en plus d'enceintes nomades grand public se basent sur des protocoles compatibles IEEE 802.11, communément appelés Wi-Fi. On peut citer les produits AIRPLAY⁴ de APPLE (depuis 2010) ou plus récemment ceux de SONOS⁵ qui propose des services s'apparentant à de véritables réseaux audionumériques pour permettre la diffusion de différents flux audio dans plusieurs pièces d'une maison simultanément.

SONOS innove en proposant à l'utilisateur de créer chez lui un réseau sans fil dédié à l'audio, et piloté avec un appareil mobile (smartphone ou tablette). Pour cela il

4. <https://www.apple.com/fr/airplay/>

5. <http://www.sonos.com/fr-fr>

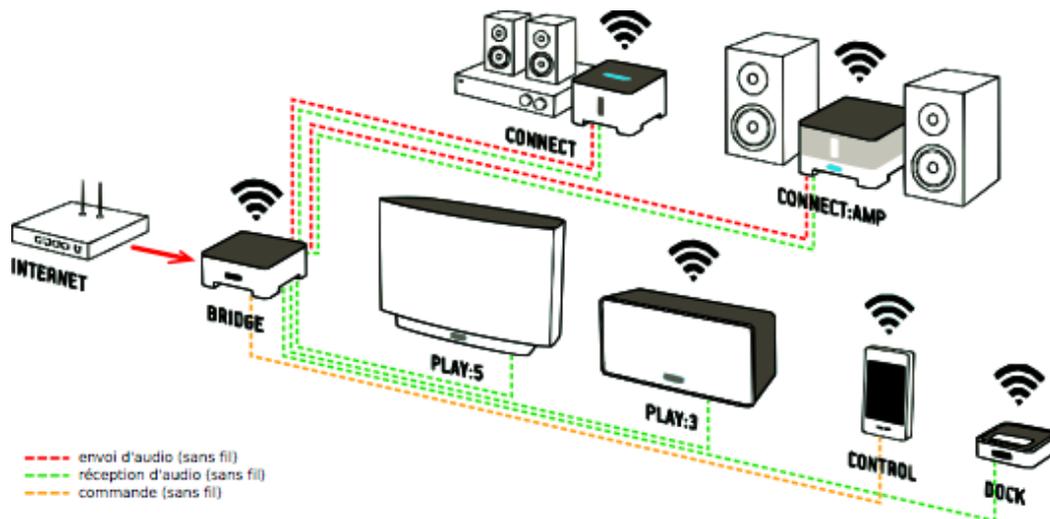


FIGURE 1.1 – Synoptique d’une installation SONOS

devra faire l’acquisition du “BRIDGE” qui servira de point d’accès sans fil. Ce point d’accès est relié au routeur domestique qui lui procure un accès à Internet pour lire du contenu en ligne. Les appareils de la gamme connectés au BRIDGE sont alors capables de lire les flux audio qui transitent sur le réseau.

Le système est suffisamment précis pour permettre à l’utilisateur d’associer plusieurs enceintes pour créer un unique système de diffusion multicanal.

Ce système utilise la flexibilité de configuration offerte par les réseaux informatiques. Les connexions audio entre les appareils se font de manière logicielle, depuis un téléphone portable ou une tablette connectés au réseau sans fil domestique (voir fig. 1.1) sur lequel est installé un logiciel de contrôle et de gestion des flux.

La simplicité de mise en œuvre et la qualité proposée par SONOS ont un coût ; ce système assure certes un délai nul entre deux enceintes d’un même système (paire stéréo, home cinéma), mais ne prévient pas de la latence entre la numérisation d’un signal audio (avec les appareils CONNECT) et sa diffusion sur les enceintes du réseau, qui est avec ce système d’environ 70 millisecondes.

Cette latence est en grande partie introduite par un *buffer* (mémoire tampon) qui a pour rôle d’amortir l’impact des erreurs de transmission. Cette marge de sécurité prise par SONOS pour leur protocole de transmission propriétaire est justifiée par le fait que pour une application grand public, une latence de 70ms est tout à fait acceptable.

Il faut noter que ces dernières évolutions des technologies grand public restent très éloignées de l'utilisation que nous souhaitons faire des réseaux audionumériques sans fil en terme de performance. Ces dispositifs permettent seulement le transport de 1 ou 2 canaux (pour chaque enceinte) avec une latence avoisinant les 100ms, alors que notre objectif en terme d'audio professionnel est de l'ordre de 1ms pour du *live*.

1.2 Exploration de l'IEEE 802.11

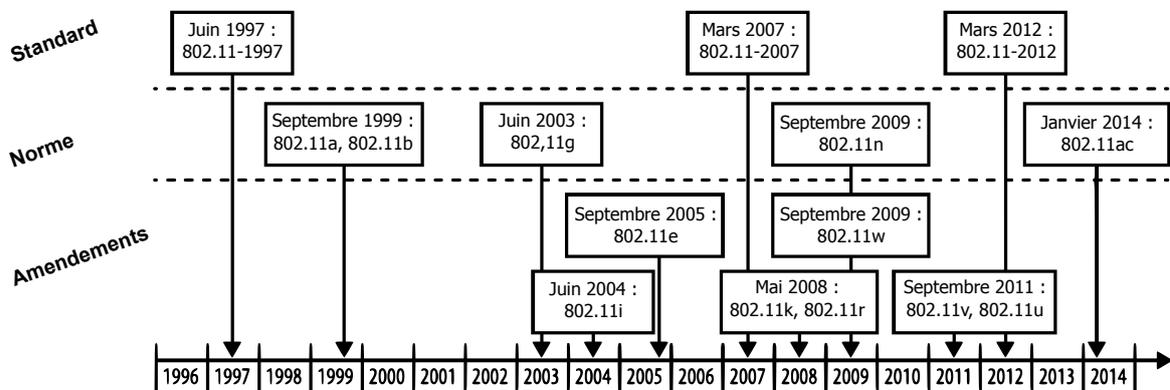


FIGURE 1.2 – Évolutions du groupe de travail 802.11 – source : [11]

L'IEEE 802.11 est un standard de transmission sans fil développé par le groupe de travail 802.11⁶ et normalisé par l'IEEE dont la première version date de 1997. Il s'agit du principal standard de communication sans fil pour les WLAN (Wireless Local Area Network, réseau local sans fil) que l'on a l'habitude de nommer Wi-Fi. Ce standard comprend une description de la couche physique (PHY) et des méthodes d'accès à la couche média (MAC : Media Access Control). Il concerne donc les couches 1 et 2 du modèle OSI (fig. 1.3). Le dernier amendement en date, l'IEEE 802.11ac, augmente de manière significative les débits pour répondre aux exigences actuelles.



FIGURE 1.3 – Le modèle OSI – source [14]

6. l'activité du groupe de travail 802.11 est visible sur <http://www.ieee802.org/11/>

Dans la suite de cette partie, nous allons énoncer les changements qui interviennent dans la norme IEEE 802.11ac, par rapport à la dernière standardisation effectuée en 2012 (IEEE 802.11-2012). La plupart des documents que nous présenterons dans cette partie, sont issu de : [11] M. GAST, *802.11ac : A Survival Guide*. O'Reilly Media, 2013. Pour obtenir des explications complètes sur les mécanismes de transmission décrit dans le standard IEEE 802.11-2012, il faut se référer au mémoire de William LEVEUGLE ([14], I.6).

1.2.1 Objectifs de la nouvelle norme IEEE 802.11ac

Afin de répondre à la demande de performances de plus en plus fortes, le groupe de travail 802.11 a mis en place un groupe d'étude « VHT » pour *Very High Throughput* (très haut débit) dont le but serait d'apporter un certain nombre de modifications au standard pour lui permettre de dépasser la vitesse de 1 Gbits/s. Parmi les applications gourmandes en débit on peut citer la diffusion de flux vidéo en haute définition ou le partage d'écran.

Ce groupe d'études est parvenu à dépasser les débits permis par la norme 802.11n (600Mbit/s) en proposant la norme 802.11ac qui permet des débits théoriques de 1,6Gbit/s. Ces performances reposent sur un certain nombre d'améliorations qui ont été effectuées et qui sont résumées dans ce tableau.

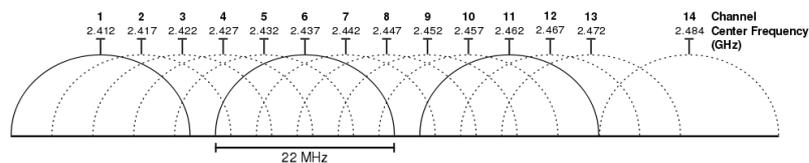
802.11n	802.11ac
<ul style="list-style-type: none"> - compatible avec des canaux de 20 et 40 MHz de largeur de bande - compatible avec les bandes de fréquences 2,4 et 5 GHz - utilise les modulations BPSK, QPSK, 16-QAM, et 64-QAM - supporte beaucoup de types de <i>beamforming</i> (voir 1.2.1.6) - supporte jusqu'à 4 flux spaciaux (MIMO 4) 	<ul style="list-style-type: none"> - ajout de canaux de 80 et 160 MHz - uniquement 5 GHz - ajout de la 256-QAM - uniquement compatible avec le <i>null data packet</i> (NDP) <i>beamforming</i> (voir 1.2.1.6) - supporte jusqu'à 8 flux spaciaux (MIMO 8) pour les points d'accès et 4 (MIMO 4) pour les clients - ajout du MU-MIMO (<i>multi-user multiple in multiple out</i>)

TABLE 1.1 – Améliorations significatives apportées par la norme 802.11ac – source [11]

1.2.1.1 Bande et largeur de bande des canaux

La bande choisie et la largeur de bande des canaux d'émission jouent un rôle important dans la transmission des données. Il faut rappeler qu'un nombre conséquent d'appareils sont conçus pour émettre et recevoir sur les bandes ISM, il convient donc de les séparer en plusieurs canaux d'émission de largeur réduite pour simplifier leur partage. Les utilisateurs choisissent de manière manuelle ou automatique le canal qu'ils vont utiliser, en évitant d'en choisir un perturbé ou déjà utilisé.

Comme le montre le schéma ci-dessous, la norme IEEE 802.11b définit les canaux utilisés dans la bande des 2,4GHz. On constate que les canaux utilisent une plage de fréquence de 22MHz, bien qu'ils soient à peine distants de 5MHz. Ce chevauchement entraîne des perturbations entre les canaux adjacents, ce qui rend les conditions de transmission assez mauvaises dans cette bande. Sur la bande des 2,4GHz on peut utiliser au plus 4 canaux qui ne se chevauchent pas : 1 – 6 – 11 – 14



La norme IEEE 802.11ac abandonne la bande des 2,4GHz pour se concentrer sur la bande des 5GHz, moins sujette aux pollutions électromagnétiques car moins utilisée. Cette bande est composée de deux sous-bandes adont l'utilisation est autorisée en France : 5150 – 5350 MHz et 5470 – 5725 MHz. Alors que la bande des 2,4 GHz dont la largeur totale est de 83MHz (2400 – 2483 MHz) ne supportait que des canaux d'une largeur de 20MHz ou de 40MHz, la bande des 5GHz, beaucoup plus large supporte des canaux étendus sur 80 voire 160MHz. Cette évolution permet d'augmenter considérablement le débit grâce à l'utilisation d'un plus grand nombre de sous-porteuses, ce qui augmente la complexité de la modulation OFDM (voir parite 1.2.1.2).

Il existe des mécanismes qui permettent de sélectionner la largeur des canaux de manière automatique. L'utilisateur doit néanmoins définir des canaux « primaires » et « secondaires ». Les protocoles de la couche MAC vont ensuite sélectionner de

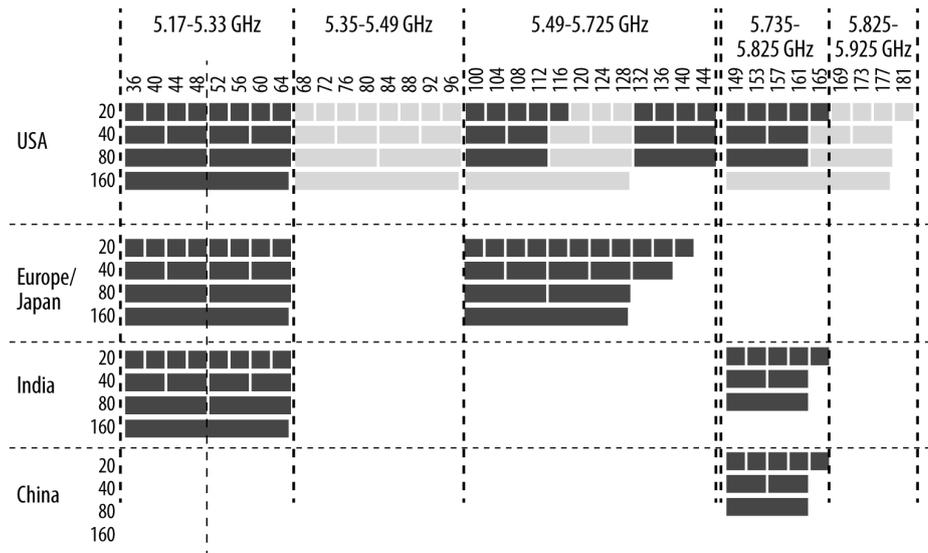


FIGURE 1.4 – Répartition des canaux de la bande des 5GHz dans le monde – source [11]

manière dynamique la bande la plus large possible, centrée sur le canal primaire (de 20MHz) choisi. Cela permet notamment à plusieurs appareils de se partager une large bande qu'ils n'utiliseront qu'en cas de besoin, tant que leurs bandes primaires ne se chevauchent pas.

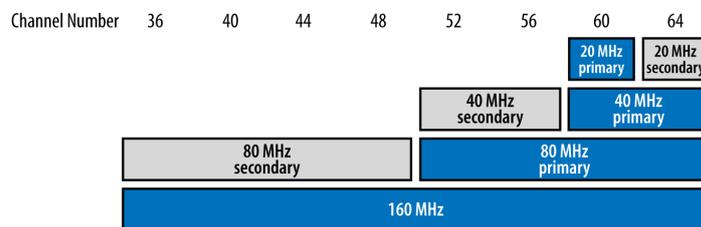


FIGURE 1.5 – Canal primaire / canal secondaire – source [11]

La bande des 5GHz présente cependant l'inconvénient de réduire la portée des émetteurs.

1.2.1.2 Utilisation des canaux – modulation OFDM

Comme ses prédécesseurs, la norme IEEE 802.11ac utilise une modulation OFDM (*Orthogonal Frequency Division Multiplexing*) pour se servir de la bande passante. Les canaux (20, 40, 80 ou 160MHz) sont divisés en plusieurs sous-porteuses (voir [14], p.27) distantes de 312,5kHz, soit 64 sous-porteuses pour un canal de 20MHz. Notons

cependant que dans ces 64 sous-porteuses, seules 52 sont utilisables pour transmettre des informations. En effet on ne se sert pas des six premières et cinq dernières sous-porteuses pour limiter les interférences avec les canaux adjacents. Le canal central n'est pas non plus utilisé en modulation OFDM.

La modulation OFDM répartit les données entrantes entre les différentes sous-porteuses.

Parmi ces sous-porteuses, certaines appelées « *pilot* » servent à calibrer la transmission entre l'émetteur et le récepteur. Elles sont donc inutilisables pour transporter des données. En utilisant des largeurs de bande plus importantes, on peut se passer de certains *pilots* et leur proportion diminue, comme on peut le visualiser sur le schéma ci-dessous (les *pilots* sont représentés par les creux dans la bande passante).

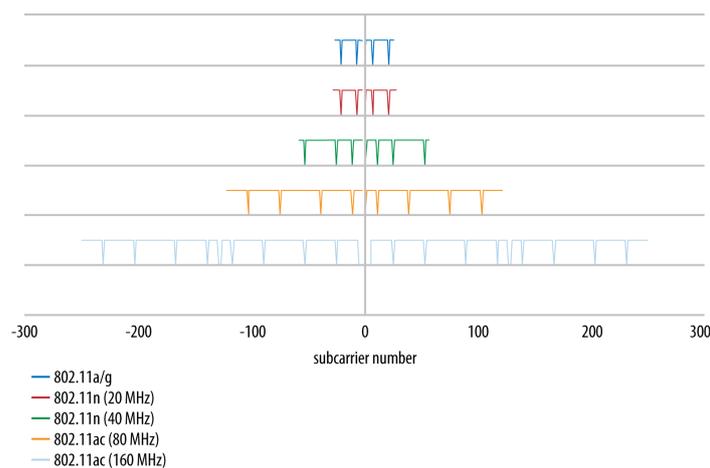


FIGURE 1.6 – Répartition des *pilots* sur la bande – source [11]

La diminution du nombre de *pilot* sur des bandes plus larges permet une augmentation relative du débit :

Standard PHY	Nombre de <i>pilots</i>	Capacité relative à une bande de 20MHz
802.11n/802.11ac, 20MHz	7%	×1,0
802.11n/802.11ac, 40MHz	5%	×2,1
802.11ac, 80MHz	3%	×4,5
802.11ac, 160MHz	3%	×9,0

1.2.1.3 Trame VHT (*Very High Throughput*, très haut débit)

Rappelons qu'avant d'être émises sous forme d'ondes électromagnétiques, les données sont regroupées dans des trames qui indiqueront dans leur entête des métadonnées renseignant notamment sur le destinataire et le type de modulation.

La conception des trames pour la norme IEEE 802.11ac s'est fait selon le critère de rétrocompatibilité imposé par le groupe de recherche 802.11. Lorsqu'un émetteur conforme IEEE 802.11ac émet une trame, les appareils des normes IEEE 802.11a et IEEE 802.11n doivent être capables de la voir et de ne pas transmettre pendant la durée d'émission. Pour répondre à ce critère, la norme IEEE 802.11ac utilise une trame mixte, qui reprend le préambule utilisé dans les normes précédentes.

Les trames doivent comporter des informations supplémentaires relatives aux spécificités de la couche PHY utilisée dans la norme IEEE 802.11ac. Elles doivent notamment renseigner sur le canal utilisé, la largeur de bande et l'utilisateur auquel elles s'adressent dans le cas d'une configuration MU-MIMO (cf. 1.2.1.7).

La partie data des trames, qui supporte les données des couches supérieures (paquets IP) est de taille variable et est modulée selon la valeur de MCS choisie (voir tableau 1.2).

Depuis la norme IEEE 802.11n, les trames de la couches MAC peuvent être agrégées pour ne former qu'une seule « trame agrégée ». Comme nous le verrons dans la partie portant sur l'optimisation des réseaux sans fil (plus précisément dans la partie 3.1.4 p.68), cela permet d'améliorer l'efficacité du réseau. Comme pour beaucoup de protocoles réseaux, la plus grande source de ralentissement vient du fait qu'il faut acquérir le droit d'émettre sur le support de communication – l'air dans notre cas – pour transmettre des données. L'agrégation de trame profite de la disponibilité du canal pour envoyer un nombre plus important de trames et ainsi augmenter le *payload* de la transmission, c'est-à-dire la partie données utiles.

L'IEEE 802.11 définit deux modes d'agrégation de trames. D'une part le mode A-MSDU (*Aggregate - MAC Service Data Unit*) qui consiste à regrouper les trames

derrière un unique entête MAC commun à toutes les trames. Celles-ci doivent obligatoirement adressées au même destinataire. D'autre part le mode A-MPDU (*Aggregate - MAC Protocol Data Unit*) dont l'implémentation est la plus répandue et dont le principe de fonctionnement est simplement de mettre bout-à-bout des trames MPDU avec un champ *delimiter* pour les délimiter (voir figure 3.4). Lorsqu'il acquiert le canal pour émettre, l'émetteur va donc envoyer un maximum de trames qu'il a en file d'attente en une seule fois. L'émetteur n'attend pas d'avoir le nombre requis de trame avant d'envoyer une trame agrégée.

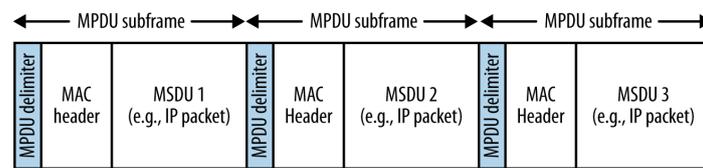


FIGURE 1.7 – Trame A-MPDU pouvant contenir une ou plusieurs trames MPDU – source [11]

Cette notion d'agrégation de trames a été améliorée dans la norme IEEE 802.11ac en ne faisant plus la distinction entre une trame agrégée et une trame seule. Une trame, qu'elle soit seule ou non, est systématiquement transmise dans une trame agrégée, ce qui simplifie les mécanismes des couches MAC et PHY au prix d'une petite augmentation de la taille des entêtes.

Dans la suite de cet exposé, nous ferons référence au mode d'agrégation A-MPDU en parlant de trames agrégées.

1.2.1.4 Modulation 256-QAM et différentes MCS (*Modulation and Coding Scheme*)

La modulation consiste à faire varier les paramètres (phase et amplitude) d'une oscillation électrique (onde porteuse) dans le but de transmettre un signal. En modulation QAM, on associe des séquences de bits (appelés « mots ») à des couples (phase, amplitude) donnés. Chaque sous-porteuse OFDM sera donc modulée en phase et en amplitude pour former un *symbole* contenant un mot de bits.

On appelle « intervalle de garde » le temps qui sépare deux symboles. Depuis la norme IEEE 802.11n, celui-ci peut être long (800ns) ou court (400ns), sachant qu'un intervalle court augmente le débit, mais n'est parfois pas suffisant pour distinguer clairement deux symboles consécutifs.

Le rapport signal/bruit obtenu en changeant de bande de fréquence nous permet d'utiliser une modulation plus complexe pour parvenir à des débits plus importants.

La norme IEEE 802.11ac introduit la 256-QAM avec 16 valeurs possibles de phase et d'amplitude. Dans ce type de modulation, il y a 256 symboles possibles, soit $2^8 = 8$ bits. Chaque symbole représenté par un point sur le diagramme de constellation ci-dessous représente 8 bits (figure 1.8 b), contre 6 bits par symboles pour une 64-QAM (figure 1.8 a). Cela entraîne une augmentation de 33% du débit.

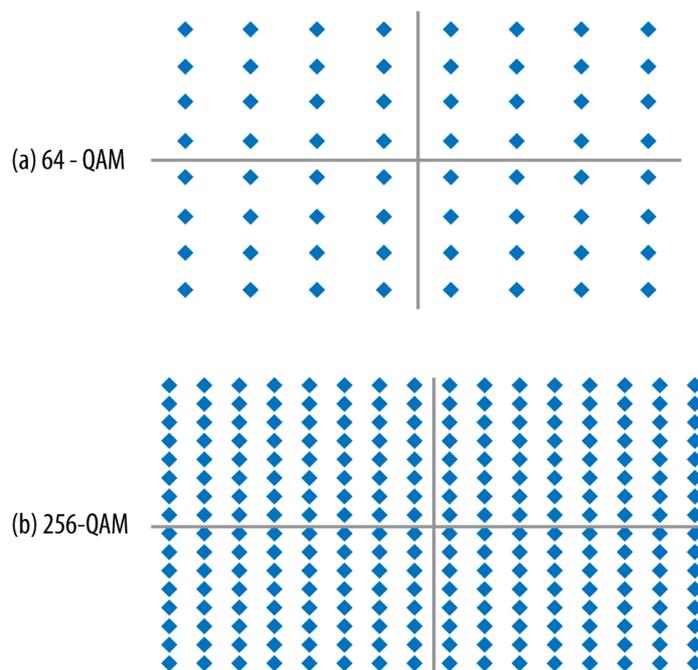


FIGURE 1.8 – Diagrammes de constellation : (a) 64-QAM, (b) 256-QAM – source [11]

Une telle modulation est tout de même très sensible aux perturbations (interférences et dysfonctionnement matériel) car l'écart entre les symboles est très faible. L'utilisation de la 256-QAM nécessite un rapport signal/bruit 5dB supérieur à celui nécessaire pour une 64-QAM ...

La norme IEEE 802.11ac définit aussi d'autres modes de modulation (MCS : *Modulation and Coding Scheme*), plus ou moins sensibles aux perturbations. Plusieurs moyens sont utiles pour réduire cette sensibilité : l'émetteur peut choisir une modulation plus « large », où les symboles seront plus facilement interprétés, ou augmenter la redondance dans la transmission (noté « taux de codage » dans le tableau ci-dessous). Ces choix de MCS se font généralement de manière automatique.

Les MCS de la norme IEEE 802.11ac sont au nombre de 10, dont les 8 premières sont mandataires, c'est-à-dire qu'elles doivent obligatoirement être supportées.

MCS index	Modulation	Taux de codage
0	BPSK	1/2
1	QPSK	1/2
2	QPSK	3/4
3	16-QAM	1/2
4	16-QAM	3/4
5	64-QAM	2/3
6	64-QAM	3/4
7	64-QAM	5/6
8	256-QAM	3/4
9	256-QAM	5/6

TABLE 1.2 – MCS disponibles en 802.11ac – source [11]

1.2.1.5 MIMO – Augmentation du nombre de flux spatiaux

Le MIMO (Multiple In Multiple Out) utilise un maillage d'antennes côté émetteur et côté récepteur. Les antennes doivent être indépendantes physiquement, et posséder leur propre circuit radio.

Lorsqu'une antenne émet un flux, il est réceptionné par chaque antenne de l'appareil récepteur d'une manière différente. Le signal reçu par les antennes réceptrices tient compte de l'empreinte de l'environnement, et de la position géographique de l'émetteur et du récepteur. Grâce à de puissants calculs, et en prenant en compte les signaux de toutes les antennes réunies, le récepteur parvient à isoler le signal émis par chaque antenne. On peut alors imaginer émettre différents flux de données avec deux, trois, . . . , huit antennes simultanément et sur la même fréquence, en parvenant à décoder le signal de chacune indépendamment.

Le nombre de flux spatiaux utilisable pour une liaison est majoré par le nombre d'antennes disponibles côté émetteur et récepteur. Un nombre d'antennes supérieur au nombre de flux spatiaux ne fait qu'améliorer le rapport signal/bruit de la transmission. Pour plus de détails concernant le MIMO, nous vous invitons à vous référer à [14].

Tandis que la norme IEEE 802.11n limitait une liaison à 4 flux spatiaux, la norme IEEE 802.11ac permet d'aller jusqu'à 8.

Le nombre d'antennes généralement limité sur les appareils nomades comme les téléphones portables ou les ordinateurs portables restreint les possibilités offertes par les points d'accès disposant d'un grand nombre d'antennes. Le groupe de travail 802.11ac a donc imaginé une nouvelle utilisation du MIMO, le MU-MIMO (Multi User MIMO) qui permet de partager les flux spatiaux disponibles d'un point d'accès entre différents utilisateurs.

1.2.1.6 NDP *beamforming* (modelage spatial de lobe avec paquets sans données)

Le *beamforming* est une technique de traitement du signal qui permet de « diriger » les ondes électromagnétiques dans une zone privilégiée de l'espace. Cela est rendu possible en combinant les antennes émettrices de telle manière à créer des interférences constructives dans la direction voulue.

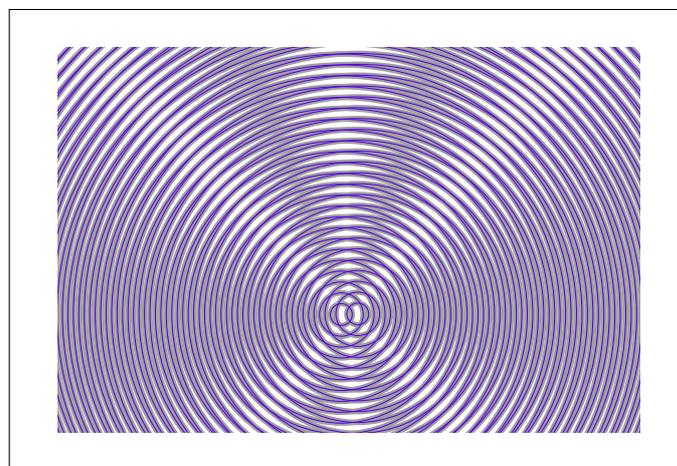


FIGURE 1.9 – Exemple d'interférence pour deux antennes émettrices

La méthode de *beamforming* employée dans la norme IEEE 802.11ac repose sur un échange de paquets appelés *Null Data Packet*.

Le point d'accès et le client partagent une phase d'écoute qui sert à mesurer la matrice de direction à appliquer pour diriger les fronts d'ondes dans la bonne direction. Pour calculer le signal propre à chaque antenne émettrice, le point d'accès et le client échangent des informations de mesure de l'environnement par l'intermédiaire de paquets ne contenant pas de données (*Null Data Packet*). Cela implique que les deux appareils doivent supporter le beamforming. L'émetteur ajuste l'amplitude et la phase de chaque antenne afin de transmettre le flux au bon endroit.

1.2.1.7 MU-MIMO

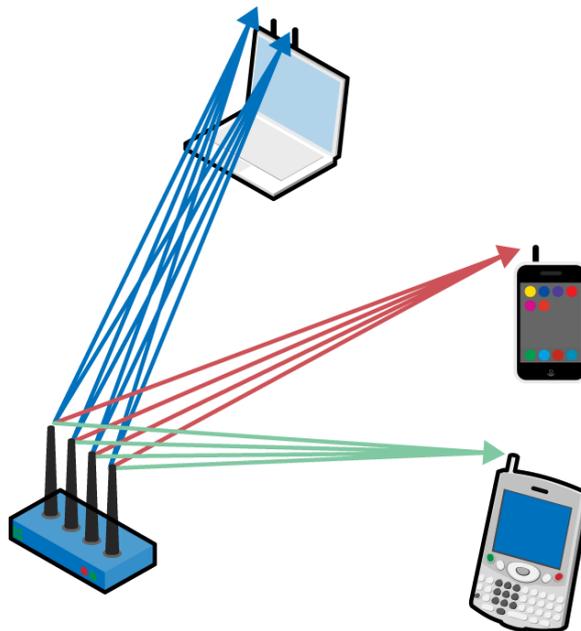


FIGURE 1.10 – Principe du MU-MIMO – source [11]

L'utilisation standard du MIMO est limitée par le nombre d'antennes des appareils indépendantes et ayant leur propre circuit radio. Par exemple dans le cas d'une liaison entre un point d'accès possédant 5 antennes indépendantes, et un client n'en possédant que deux, la transmission sera limitée à deux flux spatiaux (contre 5 normalement supportés par le point d'accès). Cette situation est courante puisque les appareils mobiles n'embarquent que très rarement un grand nombre d'antennes et de circuits

radio. Le MU-MIMO rentabilise la présence de flux non utilisés dans la plupart des configurations en les partageant entre plusieurs utilisateurs.

Le MU-MIMO (pour *Multi User MIMO*) utilise le *beamforming* pour pouvoir adresser des données simultanément à deux utilisateurs sans qu’elles interfèrent.

Le MU-MIMO tire profit de tous les flux spatiaux mis à disposition par un point d’accès seulement dans le cas où il s’adresse à des utilisateurs distants les uns des autres. Cela permet d’allouer de la bande passante supplémentaire à chaque appareil, alors que la norme IEEE 802.11n divisait la bande passante totale entre les utilisateurs.

1.2.2 Débits proposés

Une méthode proposée dans [11] pour calculer le débit offert par une liaison IEEE 802.11ac en fonction de la MCS choisie, est de partir du débit avec une largeur de bande de 20 MHz, un *short* GI (intervalle de garde court), un unique SS (spatial stream flux spatial), puis de multiplier par les facteurs représentés dans le tableau 1.3.

MCS index	débit pour 20MHz (1 SS, short GI*)	coeff. par flux spatial	coeff. par largeur de bande	débit maximal, bande de 160 MHz
MCS 0	7,2Mbit/s	×2 pour 2 flux	×1,0 pour 20MHz	520 Mbit/s
MCS 1	14,4Mbit/s	×3 pour 3 flux	×2,1 pour 40MHz	1040 Mbit/s
MCS 2	21,7Mbit/s	×4 pour 4 flux	×4,5 pour 80MHz	1560 Mbit/s
MCS 3	28,9Mbit/s	×5 pour 5 flux	×9,0 pour 160MHz	2080 Mbit/s
MCS 4	43,3Mbit/s	×6 pour 6 flux		3120 Mbit/s
MCS 5	57,8Mbit/s	×7 pour 7 flux		4160 Mbit/s
MCS 6	65,0Mbit/s	×8 pour 8 flux		4680 Mbit/s
MCS 7	72,2Mbit/s			5200 Mbit/s
MCS 8	86,7Mbit/s			6240 Mbit/s
MCS 9	96,3Mbit/s			6933,3 Mbit/s

* 1 flux spatial, intervalle de garde court)

TABLE 1.3 – Tableau de calcul de débit – source [11]

Par exemple, avec une MCS 9, 4 flux spatiaux, et une largeur de bande de 80 MHz, on obtient un débit de :

$$D = 96,3 \times 4 \times 4,5 = 1733,5\text{Mbit/s}$$

Les nouveautés apportées par la norme IEEE 802.11ac visent toutes à augmenter les débits de transmission, et à conserver un débit maximum malgré la multiplication

des clients. Le principal enjeu de cette évolution était de passer au-dessus du gigabit. Pour notre utilisation, les débits proposés par la précédente norme IEEE 802.11n sont largement suffisants, tout du moins pour les premiers tests que nous envisageons. L'autre grande nouveauté de la norme IEEE 802.11ac est le MU-MIMO. Cette technologie qui permet de mieux gérer plusieurs clients connectés à un même point d'accès pourrait s'avérer intéressante dans certaines configurations. On pourrait par exemple l'utiliser dans le cas où plusieurs consoles son connectées à un même boîtier de scène, situation courante en concert. Le MU-MIMO inclut également le beamforming qui permet de « diriger » les ondes radio dans la direction du récepteur afin de limiter les interférences et augmenter le rapport signal/bruit, ce qui est utile dans toute circonstance.

1.2.3 L'IEEE 802.11 et le multimédia

Le support explicite de l'application que nous souhaitons faire des réseaux sans fil est décrit dans d'autres *amendements* du standard IEEE 802.11.

Le groupe de travail 802.11 est divisé en plusieurs groupes d'étude. Ces derniers ont pour rôle d'explorer un sujet précis pour répondre à une problématique donnée. De leurs travaux découlent ce que l'on appelle des amendements, qui seront regroupés à terme dans une norme.

Parmi ces groupes d'étude, certains s'attaquent à la question du multimédia. On peut notamment citer le groupe 802.11aa, créé en 2008, qui travaille sur les améliorations à apporter à la norme IEEE 802.11-2012 pour permettre le *streaming* robuste de l'audio et la vidéo sur les réseaux sans fils. Le groupe 802.11aa travaille en collaboration avec le groupe de travail TSN (*Time Sensitive Networking*, ex groupe AVB *Audio Video Bridging*) qui développe le standard AVB permettant le *streaming* de l'audio et la vidéo synchronisés et à faible latence sur les réseaux câblés et sans fil. Le standard AVB contient plusieurs normes relatives à la synchronisation et à la qualité de service que doivent implémenter les *switch* Ethernet estampillés AVB et des solutions de transfert de données multimédia.

Le groupe de travail 802.11aa s'attaque de manière globale au *streaming* de la

vidéo et de l'audio sur les réseaux sans-fil, ce qui dépasse notre cadre d'étude limité à l'audio. Un certain nombre de problèmes soulevés par le groupe 802.11aa ne nous concernent donc pas. De plus, ils ne considèrent que le cas d'une diffusion *multicast* sans fil, c'est-à-dire le cas où un émetteur émet le même flux de données à plusieurs clients. Dans une première approche, ce cas ne fera pas l'objet de notre recherche.

Le groupe 802.11aa règle notamment la question de la synchronisation des appareils en faisant appel à la norme IEEE 802.1AS dont il est question dans le standard AVB.

Comme Léo ROSSI-ROTH l'a écrit dans son mémoire, la norme IEEE 802.1AS, qui définit le protocole de synchronisation gPTP (*generalized PTP*), a prévu une compatibilité avec les réseaux sans-fil. L'utilisation du gPTP sur un réseau IEEE 802.11 nécessite l'implémentation de primitives MLME-TIMINGMSMT qui permettent un horodatage le plus proche possible de la couche radio (voir [18]). Bien que quelques produits compatibles IEEE 802.1AS aient déjà vu le jour, à notre connaissance il n'existe à ce jour aucun routeur sans fil implémentant le standard gPTP.

De manière générale, il semble que l'enjeu actuel des constructeurs de matériel "Wi-Fi" pour le multimedia soit davantage axé sur le maintien des performances actuelles, même pour un grand nombre d'utilisateurs connectés sur un même point d'accès. De fait le groupe 802.11aa s'attache à la diffusion de contenu multimedia en *multicast* (donc plusieurs utilisateurs sur le même contenu). Hormis les augmentations de débit offertes par la nouvelle norme 802.11ac, le MU-MIMO démontre aussi la volonté d'offrir de meilleures performances lors de l'augmentation du nombre d'utilisateurs.

Si les derniers amendements de la norme IEEE 802.11 apportent des perspectives intéressantes pour l'utilisation dans nos domaines, il reste encore à résoudre des points cruciaux pour des applications audio-professionnelles (comme la latence). Nous allons entre autre nous attacher dans la suite de ce mémoire à savoir si dans l'état actuel, une utilisation de produits IEEE 802.11 permet l'établissement de liaisons audio.

Chapitre 2

Caractérisation des liaisons existantes et normalisation des performances minimales requises pour qu'une liaison soit apte à passer un flux AES67

Comme nous l'avons vu dans la première partie de ce mémoire, de plus en plus de matériel répond aux spécifications du standard AES67. Parmi ces spécifications, on note les méthodes de transport des paquets audio ainsi que des données de synchronisation.

Dans son mémoire, Léo ROSSI-ROTH décrit en détail le fonctionnement et le transport du protocole PTPv2 préconisé pour la synchronisation des appareils audio dans les réseaux AES67. Il explique notamment en quoi il n'est pas possible d'utiliser une liaison IEEE 802.11 sans modifications matérielles et logicielles pour synchroniser les appareils audio sur le réseau.

Lors de sa soutenance, il était tout de même parvenu à obtenir la synchronisation de deux Horus (matériel compatible RAVENNA) à travers une liaison *full-duplex*, chose qui s'avérait impossible avec une liaison IEEE 802.11 standard. Cette réussite

confirme qu’il existe des liaisons sans fil alternatives dont les caractéristiques sont plus favorables au transport de la synchronisation.

Dans la première partie de ce chapitre, nous avons souhaité rappeler et mettre en avant certains éléments concernant la gestion de la synchronisation dans un appareil audio possédant une interface type AES67. Il ne s’agira pas de refaire l’étude du fonctionnement du protocole PTP sur un réseau, mais d’apporter une vision plus globale de la synchronisation dans une installation audio numérique.

Cette partie aura aussi pour ambition de s’attacher plus précisément au transport des données audio, afin de déterminer si, à défaut d’être incompatible pour la synchronisation, un lien IEEE 802.11 s’avère être un bon candidat pour le transport des données audio.

Pour cela nous étudierons la nature des données audio échangées sur un flux AES67, avant de mettre un réseau audionumérique à l’épreuve grâce au simulateur de réseau *netem*.

2.1 Horloge

Comme dans toute installation audionumérique, le transport et la gestion de la synchronisation sont au moins aussi importants que ceux des données audio, et une attention particulière doit être apportée à la manière dont se partage cette référence temporelle entre les appareils.

2.1.1 Installations traditionnelles – rappel

Dans une installation classique les appareils utilisent une référence commune pour se synchroniser. Par ordre de préférence, on utilise :

- Word Clock externe en étoile (fig. 2.1.a)
- Word Clock externe en *daisy chain* (fig. 2.1.b)
- entrées/sorties numérique (AES/EBU, MADI)

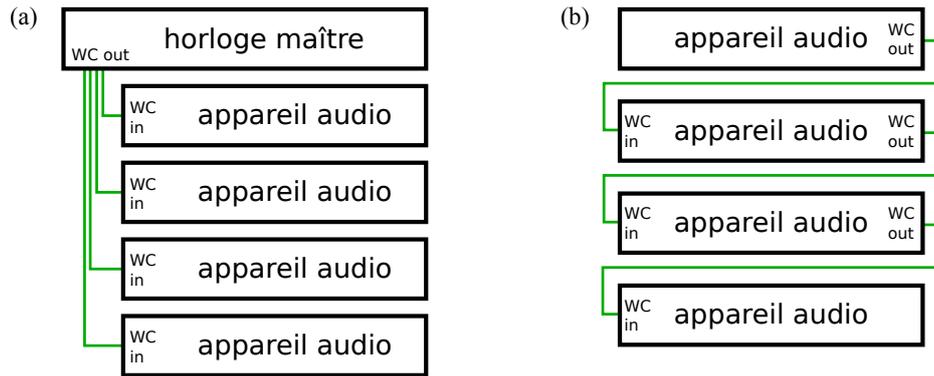


FIGURE 2.1 – (a) topologie en étoile, (b) topologie en *daisy chain*

La transmission de l’horloge se fait dans par une relation hiérarchique ; l’horloge **esclave** suit l’horloge **maître** .

L’utilisation de deux horloges différentes pour deux appareils qui communiquent de l’audio entraîne notamment les fameux clics de synchronisation.

2.1.2 Synchronisation ou syntonisation

Il est important de préciser le terme de synchronisation, utilisé de manière générale pour parler aussi bien de synchronisation que de syntonisation.

Synchronisation, subst. fém., [...] MÉCAN., PHYS. „Égalisation des vitesses et mise en concordance de phase de deux ou plusieurs machines synchrones identiques mécaniquement indépendantes“ (Siz. 1968). [...]

Syntonisation, subst. fém.,phys., radio-électr. Action d’accorder plusieurs circuits sur la même fréquence de résonance ; résultat de cette action.[...]

*Centre National de Ressources Textuelles et Lexicales,
www.cnrtl.fr*

La définition du terme « synchronisation » reste assez large. Dans la suite de ce mémoire, nous utiliserons les définitions suivantes tirées de la norme IEEE 802.1AS [5] définissant le protocole de synchronisation gPTP (voir [18], p.43).

3.20 **synchronized** time-aware systems : Two time-aware systems are synchronized to a specified uncertainty if they have the same epoch and their measurements of the time of a single event at an arbitrary time differ by no more than that uncertainty. [...]

3.21 **syntonized** time-aware systems : Two time-aware systems are syntonized if the duration of the second is the same on both, which means the time measured by each advances at the same rate. They can but need not share the same epoch.

IEEE 802.1AS [5], p.6

Nous parlerons donc de synchronisation pour l'action de partager une fréquence de lecture **et une date** entre toutes les machines. Il s'agit en fait de la même définition utilisée dans le cinéma lorsqu'on parle de synchronisation entre l'image et le son.

Cette définition plus précise nous permet de dire que dans une installation classique, dite « synchronisée en *word clock* », il faudrait plutôt parler de syntonisation dans la mesure où le signal de *word clock* ne véhicule aucune information de date.

Cette distinction sera utile par la suite étant donné que pour transférer des données audio entre deux machines à travers une liaison AES67, celles-ci devront être rigoureusement synchronisées. L'utilisation de la synchronisation en plus de la syntonisation dans les réseaux audionumérique s'explique par l'utilisation du protocole UDP qui ne garantit pas le fait que les paquets seront reçus dans l'ordre d'émission. Il est alors nécessaire d'apposer sur chaque paquet une date (*timestamp*) qui, une fois analysée par l'appareil récepteur, puis comparée à sa propre horloge, l'informerá sur l'heure à laquelle il doit jouer les données audio.

2.1.3 Qualité de la synchronisation à travers une liaison sans fil

La précédentes études ont montré qu'il était possible de synchroniser des appareils RAVENNA (utilisant le protocole PTPv2) à travers une liaison *full-duplex* pour leur faire échanger deux canaux audio dans chaque sens.

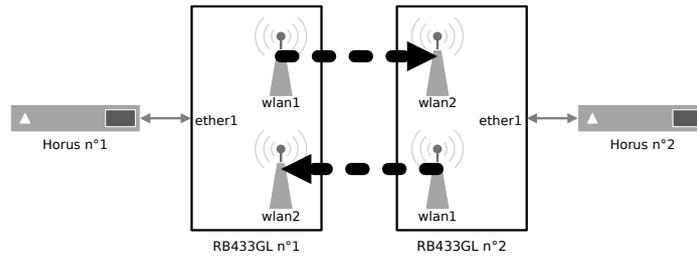


FIGURE 2.2 – Liaison réalisée lors de la soutenance de Léo ROSSI-ROTH

Dans notre volonté de poursuivre les travaux ayant été menés par William LEVEUGLE puis Léo ROSSI-ROTH, nous avons souhaité reprendre les tests au stade où ils avaient été laissés, c'est-à-dire commencer par mettre en place une liaison similaire à celle réalisée lors de la soutenance de Léo ROSSI-ROTH en juin 2014.

Le but de ces premiers tests, sera de voir si nous sommes en mesure de réaliser à nouveau la liaison de la figure 2.2 avec la technologie Dante. Si possible, nous souhaiterions ensuite mesurer la gigue d'horloge résultant de la transmission sans fil.

2.1.3.1 Test synchronisation 1

Matériel utilisé :

- 2× YAMAHA 01V96i équipées de cartes MY16AUD (carte DANTE)
- 2× MIKROTIK RB433GL + antennes
- ordinateur avec Dante Controller
- switch TP-LINK SG3210

Synoptique de l'installation :

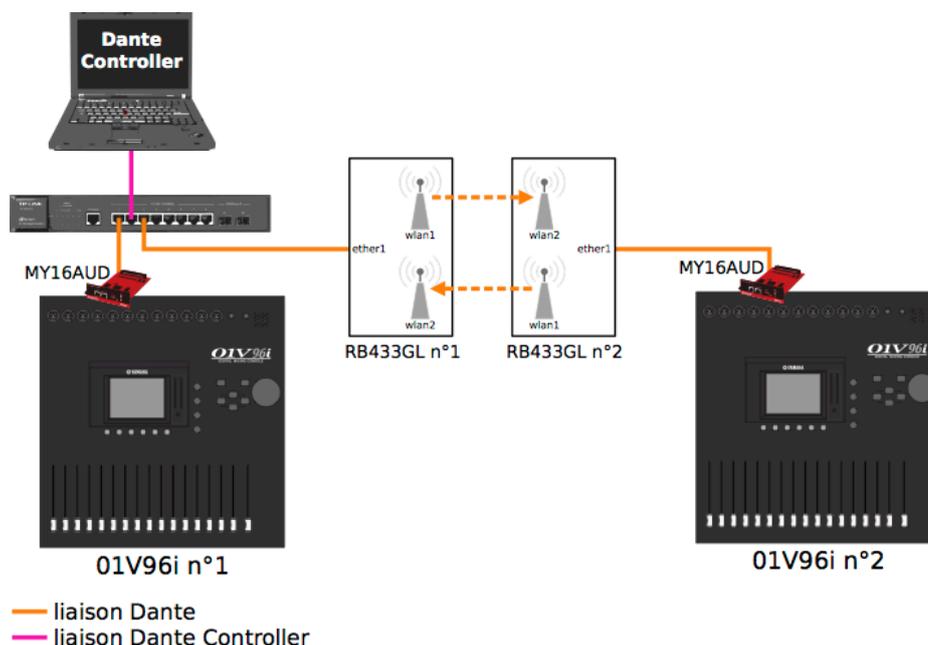


FIGURE 2.3 – Synoptique du test de synchronisation 1

Configuration matérielle :

Les 01V96i sont configurées pour suivre la synchronisation du réseau DANTE¹. La carte MY16AUD de la 01V96i n° 1 est déclarée maître, et celle de la 01V96i n° 2 esclave sur le réseau. Les routeurs MIKROTIK sont configurés en mode *bridge*, c'est-à-dire que toute information arrivant sur l'interface filaire est envoyée sur l'interface sans fil, et vice versa.

Nous avons connecté, via le Dante Controller, deux canaux dans le sens console n° 1 → console n° 2 et deux autres dans le sens console n° 2 → console n° 1.

Résultats du test

Nous ne sommes pas parvenus à réaliser la liaison attendue. À la connection des deux 01V96i à travers le lien sans fil, il est possible de bien visualiser bien leur apparition dans la fenêtre principale du Dante Controller, cependant, elles ne parviennent pas à se synchroniser l'une avec l'autre.

1. nous donnerons plus de détail sur la dissociation de l'horloge audio et de l'horloge du réseau dans la partie suivante

On peut émettre trois hypothèses sur le dysfonctionnement observé. La **première hypothèse** est que le protocole de synchronisation PTPv1 utilisé par DANTE ne permette pas d'obtenir une synchronisation, alors que le PTPv2 utilisé par RAVENNA fonctionnait. La **deuxième hypothèse** est le fait que DANTE regroupe les canaux par *flows* (flux) de quatre canaux², cela signifie qu'une liaison de 2x2 canaux équivaut à une liaison de 4x4 canaux en terme de débit, ce qui est le double de la liaison effectuée par Léo ROSSI-ROTH avec la technologie RAVENNA. La **troisième hypothèse** est que les cartes MY16AUD installées dans les bacs d'extension des 01V96i sont trop anciennes et ne bénéficient pas des mêmes évolutions techniques que les Horus utilisées auparavant.

La seule hypothèse que nous sommes en mesure de vérifier est la dernière en remplaçant une des 01V96i par un autre périphérique DANTE, plus récent.

2.1.3.2 Test synchronisation 2

Matériel utilisé :

- 2× YAMAHA CL5 (console avec interface DANTE 64 canaux)
- 2× MIKROTIK RB433GL + antennes
- ordinateur avec Dante Controller
- switch TP-LINK SG3210

² *Dante Controller User Guide*, version 1.6, AUD-MAN-DanteController-3.5.x-v1.6.pdf, Audinate, 2014, p.75

Synoptique de l'installation :

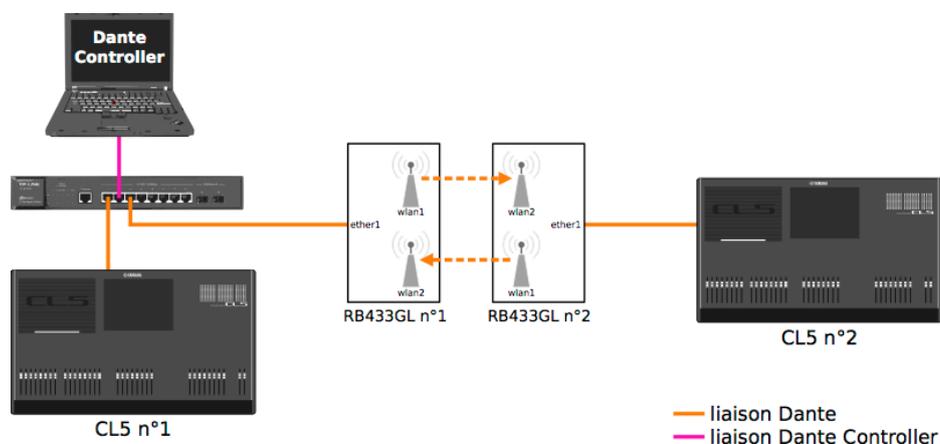


FIGURE 2.4 – Synoptique du test de synchronisation 2

Configuration matérielle :

Nous sommes exactement dans la même configuration que dans le test précédent, à la différence près que nous avons remplacé les 01V96i par des CL5 qui intègrent directement des modules DANTE.

Résultats du test

Le résultat de ce test est positif dans le sens où nous sommes parvenus à établir une liaison DANTE entre deux CL5 à travers le pont sans fil. L'outil de contrôle du réseau (Dante Controller) affiche bel et bien un témoin vert ne signalant aucune erreur de synchronisation.

Nous avons fait transiter des canaux audio dans les deux sens, sans erreur apparente.

On peut donc en déduire que le précédent test n'a pas fonctionné à cause de l'ancienneté des modules DANTE utilisés avec les 01V96i. Cela signifie qu'au sein même de la technologie DANTE, nous devons faire la distinction entre les différentes générations de matériel. Dans la suite des tests, nous utiliserons les consoles CL5.

2.1.3.3 Test synchronisation 2 bis

Bien que l'audio semble effectuer son aller-retour sans erreur, rien ne nous permet de certifier que le transport de la synchronisation est aussi précis qu'il peut l'être en filaire. Comme nous le verrons plus tard (partie 2.1.4 p.29) dans le cadre de ce test, l'horloge interne de la console est synchronisée sur l'horloge du module DANTE qu'elle intègre. Une méthode pour mesurer la gigue d'horloge résultant de la synchronisation par le protocole PTP, serait alors de comparer, à l'aide d'un oscilloscope, l'horloge des consoles elles-mêmes³. Le module DANTE de la CL5 n° 1 étant « maître PTP » sur le réseau, l'horloge de la CL5 n° 1 servira de référence, et le module DANTE de la CL5 n° 2 étant « esclave PTP » sur le réseau, on observera sa gigue d'horloge en analysant l'horloge de la CL5 n° 2.

Configuration du test :

On se place dans la même configuration que dans le test de synchronisation 2. On place un oscilloscope dont l'entrée *CH-A* est connectée à la sortie *word clock* de la CL5 n° 1, et l'entrée *CH-B* à la sortie *word clock* de la CL5 n° 2.

Protocole :

On réalise dans un premier temps un test avec une liaison filaire.

On remplace ensuite la liaison filaire par la liaison sans fil.

Résultat du test :

La mise en place d'un oscilloscope pour analyser l'état de syntonisation du dispositif a permis d'observer que la CL5 n° 2 subissait une gigue d'horloge très importante par rapport à celle observée dans le cas d'une liaison filaire.

Bien que l'oscilloscope utilisé ne permettait pas de mesurer les décalages de forme d'onde que nous observions, nous pouvons donner les valeurs approximatives de la gigue d'horloge :

liaison filaire : gigue $< \frac{1}{20}$ échantillons (= 1µs)

liaison MIKROTIK : gigue > 5 échantillons (= 104µs)

3. ce qui est possible en analysant les sorties d'horloge *WC out* des consoles

Ces résultats médiocres étaient attendus étant donné que la précision de synchronisation du PTP est déterminée par la qualité du réseau en terme de variation de temps de parcours des paquets.

Cependant des alternatives pour synchroniser et/ou syntoniser des appareils en sans fil existent. Nous verrons ces solutions dans la partie 3.3 (p.87) de ce mémoire.

En attendant, il serait intéressant de pouvoir faire des tests sur le transfert de l'audio à proprement parler. Pour cela il est avant tout nécessaire de tirer un trait sur le problème de la synchronisation, qui est actuellement l'élément qui nous limite dans nos expérimentations. Dans la même optique, Léo ROSSI-ROTH avait essayé de mettre en place une liaison hybride en séparant les paquets PTP (synchronisation) et les paquets RTP (audio), pour acheminer la synchronisation en filaire, et l'audio en sans fil. Cette idée n'avait pas abouti, étant donné que la RFC 1112 n'autorise pas le routage d'adresses appartenant à la plage d'adresse IP 224.0.0.0/24⁴ dont font partie les paquets PTP.

La documentation du Dante Controller⁵ mentionne dans un paragraphe à propos de la synchronisation externe des modules DANTE, le cas, dit « non-usuel » dans lequel l'utilisateur choisirait de synchroniser plusieurs appareils du réseau sur une source externe de synchronisation. Audinate signale que dans ce cas, les appareils doivent partager la même source d'horloge pour fonctionner ensemble. Cette remarque, nous permet de croire que la technologie DANTE permettrait d'avoir plusieurs appareils synchronisés en *word clock* externe, et qu'ils n'utiliseraient peut-être plus le protocole PTP pour se synchroniser.

En « externalisant » la synchronisation, nous serions peut-être en mesure de réaliser les tests sur le transport de l'audio que nous envisageons.

4. adresses de 224.0.0.0 à 224.0.0.255

5. *Dante Controller User Guide*, version 1.6, AUD-MAN-DanteController-3.5.x-v1.6.pdf, Audinate, 2014, p.37

2.1.4 Syntonisation externe

Rappelons brièvement que le fonctionnement du protocole PTP repose sur un échange de paquets horodatés. Le périphérique esclave calcule le délai de son compteur par rapport à celui du master et corrige sa phase (synchronisation), en répétant l'opération, il en déduit la correction à appliquer à sa propre horloge (syntonisation).

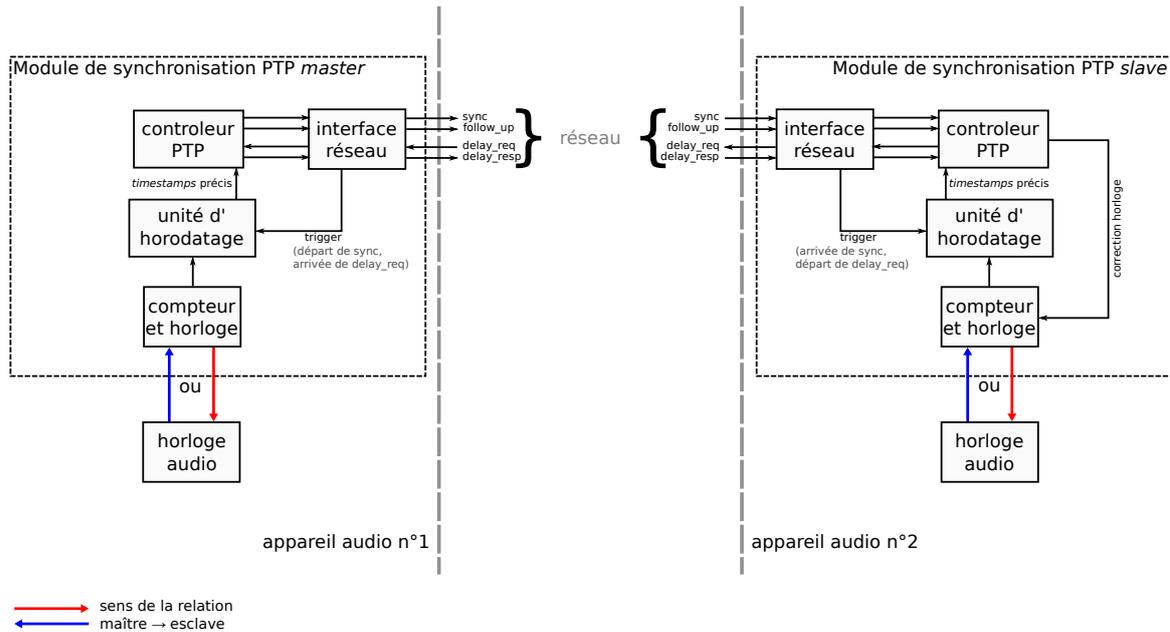


FIGURE 2.5 – Schéma de fonctionnement de synchronisation de deux module PTP

Dans la figure 2.5, on note qu'il y a une dissociation entre l'horloge des modules PTP et l'horloge de l'appareil audio. Cela transparaît notamment dans le menu de configuration d'horloge des CL5, dans lequel nous avons le choix de synchroniser la console sur une source externe d'horloge, sur sa propre horloge interne, ou sur le réseau DANTE. Si cette dernière option est choisie, la console syntonisera son horloge en fonction de celle de son module PTP (un module DANTE dans le cas d'une CL5).

La configuration classique consiste à mettre tous les appareils audio dans cette dernière configuration (flèches rouges) pour ainsi laisser le protocole PTP gérer la synchronisation. Inversement, on peut aussi configurer l'horloge des modules PTP pour qu'elles se syntonisent sur l'horloge audio des appareils. Pour un périphérique DANTE, cette option est disponible en cochant la case « sync to external » dans le Dante Controller.

La configuration « non-usuelle » décrite dans le guide d'utilisateur du Dante Controller ([9], p 37) consiste à placer plusieurs périphériques DANTE en synchronisation externe (voir figure 2.6 (b) et (c)).

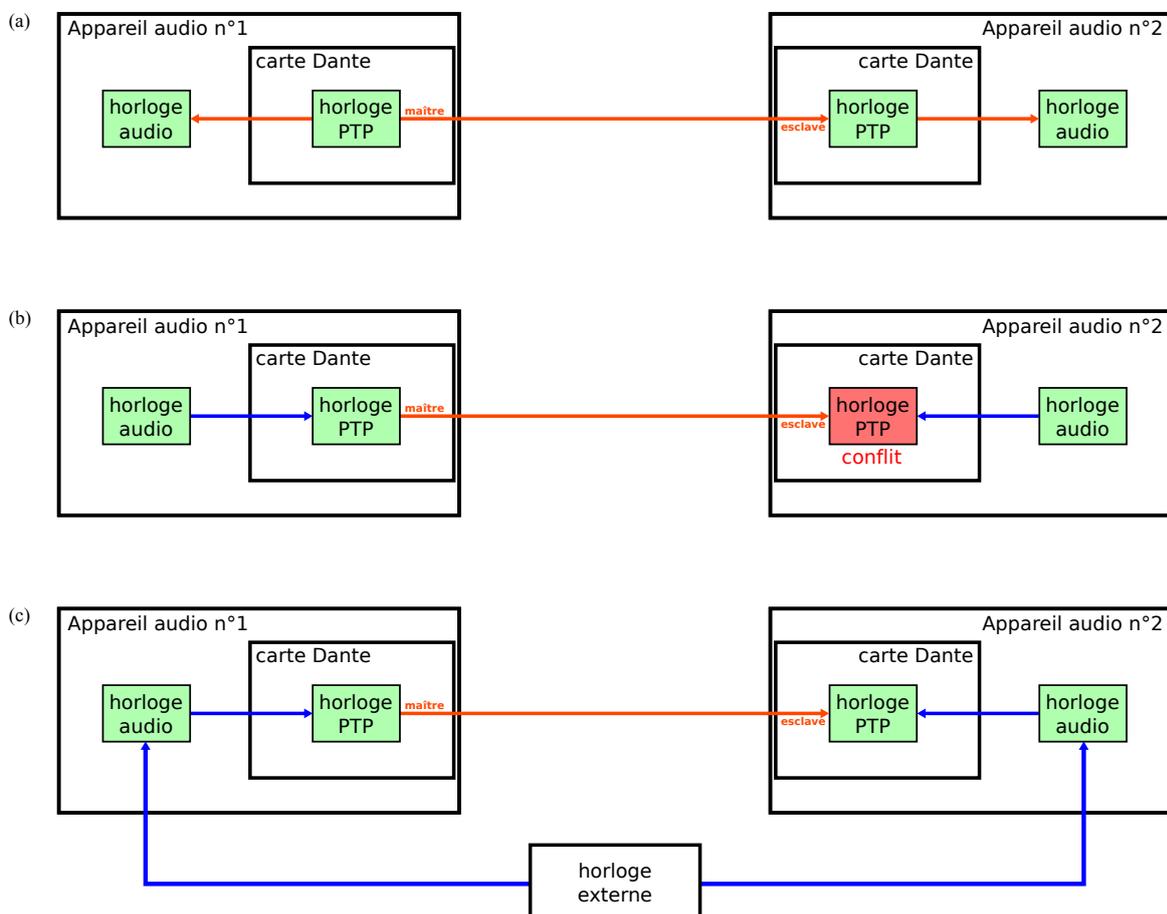


FIGURE 2.6 – (a) situation standard, (b) synchronisation externe avec horloges audio non-identiques, (c) synchronisation externe avec horloge audio identiques

Lorsque l'on configure plusieurs périphériques DANTE en externe, il faut veiller à ce que l'on ait toujours **une unique** source d'horloge. Dans le schéma 2.6 (b), où l'on n'a pas mis en place de synchronisation externe, les deux informations différentes d'horloges entrent en conflit au niveau de la carte DANTE n°2; celle-ci reçoit deux renseignements discordants, et ne pourra donc pas se synchroniser.

La solution serait de synchroniser les horloges audio sur une même référence, pour que chaque élément de l'installation distribue à son tour la même valeur d'horloge.

Afin de vérifier le bon fonctionnement de ce type d'installation avec horloge externe, nous avons effectué des essais avec le simulateur *netem*. La présentation de

netem et le déroulé des tests seront abordés dans la partie 2.3 p.36.

2.2 Besoins d'un réseau AES67

Pour les raisons évoquées précédemment, les tests que nous avons effectués dans le cadre de ce mémoire ont été exclusivement réalisés avec du matériel DANTE. Par conséquent, les applications numériques et expériences contenues dans cette partie seront généralement conformes au matériel DANTE.

Dans la mesure où les mécanismes de transport décrits dans l'AES67 sont similaires à ceux utilisés par la technologie DANTE, nous pourrions tout de même tirer des conclusions conformes au standard AES67.

2.2.1 Paquet RTP – conteneur de l'audio

Lors d'une transmission de données avec le protocole RTP, plusieurs canaux audio peuvent être regroupés dans le même paquet RTP. Ces canaux forment alors ce que l'on appelle un *stream* (flux) audio.

Un paquet RTP est encapsulé dans un paquet UDP puis dans un paquet IP. Avant d'être transporté sur le réseau.

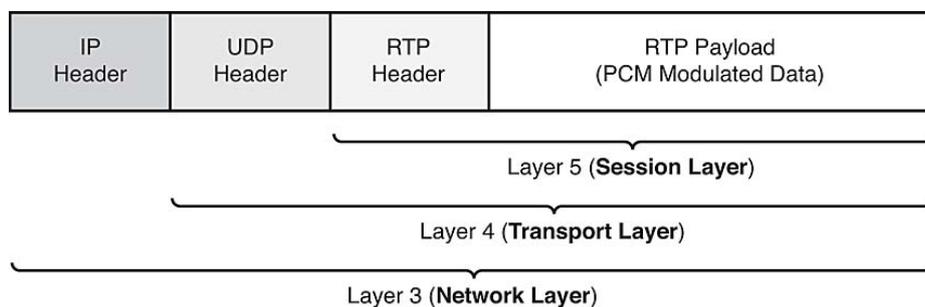


FIGURE 2.7 – Encapsulation d'un paquet RTP – source [12]

On appelle *payload* la partie données utiles d'un paquet et *header* le préambule de ce paquet.

La taille du paquet vaut :

$$S_{paquet} = S_{header} + S_{pay}$$

avec S_{header} la taille du *header* (IP header + UDP header + RTP header) et S_{pay} la taille du *payload*

La taille totale du *header* S_{header} est de 40 octets. La taille du *payload* est variable. Pour optimiser le débit utile de transmission, il faut avoir un *payload* le plus grand possible.

Taille de paquet et *packet time* (temps de paquet)

Lors d'une transmission de données informatiques, le MTU (maximum transmission unit unité maximale de transmission) désigne la taille maximale qu'une interface peut transmettre sans fragmentation. Cette valeur est réglable directement par le pilote de l'interface, et vaut généralement 1500 octets. L'AES67, qui veut garantir une compatibilité avec les réseaux informatiques standards indique de ne pas dépasser une taille de *payload* de 1440 octets pour éviter toute éventuelle fragmentation des données audio⁶. De plus, avec un *payload* important, on optimise le débit puisque la taille du header, elle, reste la même.

L'utilisateur a donc la possibilité de transmettre jusqu'à 1440 octets de données audio, soit 480 échantillons à 24bits répartis sur le nombre de canaux contenus dans le stream.

$$N_{e/p} = \frac{S_{pay}}{Q}$$

avec $N_{e/p}$, le nombre d'échantillons par paquet, S_{pay} la taille du *payload* et Q la quantification de l'audio.

6. "Fragmentation is undesirable and under this standard receivers are not required to perform reassembly." (La fragmentation est indésirable, le standard ne requiert pas des récepteurs qu'ils soient en mesure de réassembler les paquets.) [7], p.16

Vient alors la question de temps de paquet, ou le temps d'audio nécessaire pour remplir un paquet RTP. La latence minimale du système sera minorée par cette valeur.

Le temps de paquet T_p est fonction de la taille en octets du payload, du nombre de canaux par stream $n_{c/s}$, de la fréquence d'échantillonnage f_{ech} et de la quantification. Il est donné par la formule suivante :

$$T_p = \frac{N_{e/p}}{n_{c/s} \cdot f_{ech}}$$

$$= \frac{S_{pay}}{n_{c/s} \cdot f_{ech} \cdot Q}$$

Le temps de paquet doit être choisi en tenant compte de la latence escomptée pour le système. Pour des latences plus élevées on peut se permettre un temps de paquet « long », tandis que dans une application à faible latence, il faudra choisir un temps de paquet très court.

L'AES67 conseille quelques temps de paquets répondant à diverses applications. Pour être compatible avec le standard, il est précisé que les appareils émetteurs et récepteurs doivent obligatoirement supporter un temps de paquet de 1ms, les autres temps supportés sont optionnels.

Temps paquet	n_{ech} 48kHz	n_{ech} 96kHz	n_{ech} 44,1kHz	Commentaires
125µs	6	12	6	très haute performances
250µs	12	24	12	haute performances
333µs	16	32	16	opérations faible latences
1ms	48	96	48	temps de paquet requis pour le matériel compatible avec le standard
4ms	192	n/a	192	compatible avec le transport AVB de classe A

TABLE 2.1 – Temps de paquets requis et conseillés par l'AES67

Lors de nos tests avec du DANTE, nous avons mesuré un temps de paquet de 667µs à 48kHz, ce qui correspond à 32 échantillons.

On notera de plus qu'un temps de paquet de 4ms à 96kHz n'est pas supporté car la MTU limiterait le stream à un seul canal. Pour deux canaux :

$$S_{pay} = T_p \cdot n_{c/s} \cdot f_{ech} \cdot Q = 4 \cdot 10^{-3} \times 2 \times 96 \cdot 10^3 \times 3 = 2304 \text{ octets} > \text{MTU}$$

2.2.2 Calcul de débit

L'émetteur envoie autant de paquet RTP qu'il y a de *streams* à un intervalle de temps T_p .

Le débit D peut donc être calculé avec la formule suivante :

$$D = N_{stream} \cdot \frac{S_{pay} + S_{header}}{T_p}$$

$$\text{or } S_{pay} = T_p \cdot n_{c/s} \cdot f_{ech} \cdot Q$$

$$D = N_{stream} \left(n_{c/s} \cdot f_{ech} \cdot Q + \frac{S_{header}}{T_p} \right)$$

La technologie DANTE regroupe 4 canaux par *stream* ce qui nous permet de calculer le débit pour 4 canaux audio, à 48kHz, 24bits et avec un temps de paquet de 667 μ s :

$$D = 1 \times \left(4 \times 48 \cdot 10^3 \times 24 + \frac{40 \times 8}{667 \cdot 10^{-6}} \right) = 5,09 \text{Mbit/s}$$

2.2.3 Latence de lecture – Mémoire tampon de réception

Dans une installation audionumérique « traditionnelle », les échantillons audio arrivent en continu et sont directement lus à leur arrivée. Ils arrivent à destination avec une telle précision, que les appareils audio sont même en mesure d'en déduire

un signal de synchronisation. Dans le cas d'un réseau audionumérique, une mémoire tampon est nécessaire côté récepteur pour pouvoir absorber les variations de temps de transmission introduites par la mise en paquet, la capacité du réseau, et les contrôleurs réseaux. L'AES67 préconise d'utiliser une mémoire tampon au moins égale à trois fois la valeur du temps de paquet. Les échantillons audio émis à l'instant t seront donc lus à l'instant $t + \mathcal{T}$, où \mathcal{T} correspond à la valeur de latence sélectionnée. Cette valeur devra être supérieure à la somme du temps de paquet et du temps de transfert sur le réseau. Ce dernier étant variable, il faudra choisir une latence suffisamment grande pour faire face aux éventuelles fluctuations de temps de transmission.

On peut définir deux types de latence, la latence dite « traditionnelle » due aux conversions numériques/analogiques, aux tampons d'entrée/sortie et aux traitements divers, et la latence dite « de réseau⁷ » ajoutée par l'utilisation d'un réseau audionumérique, égale à la taille du tampon sélectionné pour la réception. La latence totale de notre système sera donc égale à la somme de ces deux latences.

Une des caractéristiques de la mémoire tampon de réception, est qu'elle est configurable par l'utilisateur. Dans le cas d'un réseau DANTE, la configuration se fait via *Dante Controller* qui propose différentes valeurs selon la complexité du réseau (c.f. fig. 2.8). Dans le cadre de nos tests nous avons configuré cette valeur sur 5ms (mode dit « sécurité »).

Plus la valeur du tampon est élevée, plus le système est capable d'absorber des grandes latences de transmission. Cela a aussi pour conséquence d'augmenter la latence de lecture, ce qui n'est pas acceptable dans certains cas.

Le temps de paquet est en revanche non indiqué par *Dante Controller*. Il est automatiquement déduit de la valeur de tampon choisie. On peut le mesurer avec un outil de capture réseau comme *wireshark*. C'est avec cette technique que nous avons mesuré un temps de paquet de 667µs pour 4 canaux à 48kHz 24 bits, valeur utilisée pour nos calculs précédents.

7. à ne pas confondre avec le délai de parcours (défini dans la partie 3.1.2) qui correspond au temps de que met un paquet réseau à parcourir le réseau

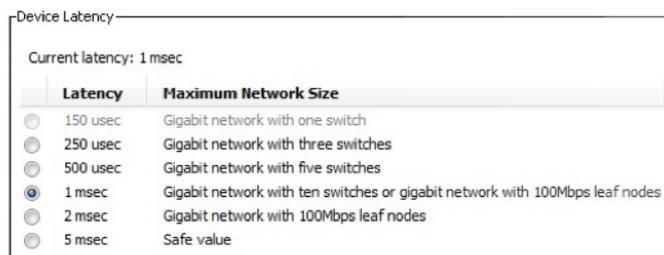


FIGURE 2.8 – Sélection de la latence dans Dante Controller

2.2.4 Résumé

Si l'on résume le transport d'un réseau audio numérique au transport de deux types de données sensibles, l'audio et la synchronisation, on peut en déduire les caractéristiques minimales requises pour les liaisons utilisées.

	Audio (RTP)	Synchronisation (PTP)	Commentaires
latence	$\ll 5\text{ms}$ (\ll tampon)	/	la latence n'a théoriquement pas d'influence sur la synchronisation, tant que celle-ci est fixe
gigue	/	$< 10\mu\text{s}$ * ($\frac{1}{2}$ échantillon à 48kHz)	la gigue n'a <i>a priori</i> pas d'influence sur les données audio, tant que la latence totale reste sous la valeur du tampon
débit	$> 20\text{Mbit/s}$ (pour 16 canaux)	13,6kbit/s	le débit nécessaire pour le transport du PTP a été évalué par [14], p.93

* $10\mu\text{s}$: cette valeur ne devrait pas être dépassée par souci de qualité de restitution sonore, cependant DANTE semble supporter des valeurs bien plus élevées (voir p.27)

TABLE 2.2 – Caractéristiques minimales requises pour transporter un réseau audionumérique

On constate qu'entre les données audio et celles de la synchronisation, les exigences en performances ne sont pas les mêmes. Une stratégie à avancer pour réaliser notre objectif serait de dissocier le transport de l'audio et celui de la synchronisation afin de répondre de manière adaptée aux exigences de chacun.

2.3 netem : simulateur de liaisons

Pour simuler l'influence qu'aurait le transport d'un réseau audionumérique sur un lien sans fil, nous avons utilisé netem, un simulateur de réseau. Cet outil est conçu pour permettre à des développeurs de tester le fonctionnement de leurs programmes

sur différentes qualités de liaison, en y ajoutant latence, gigue, perte ou corruption de paquets ...

Il s'agira dans cette partie de chercher les limites de fonctionnement d'un réseau audionumérique pour voir dans quelle mesure un lien sans fil pourrait remplacer un lien filaire.

2.3.1 Présentation de netem

netem est un simulateur réseau faisant partie de la banque d'outils Traffic Control (tc) présente dans le package GNU/netem iproute. Ce simulateur permet de modifier les caractéristiques du réseau en ajoutant du délai, de la latence, de la gigue, et de la perte de paquets.

Pour comprendre à quel niveau ce simulateur agit et prévoir les éventuels biais qu'il pourrait introduire, il convient d'observer brièvement le chemin suivi par un paquet dans le noyau GNU/Linux.

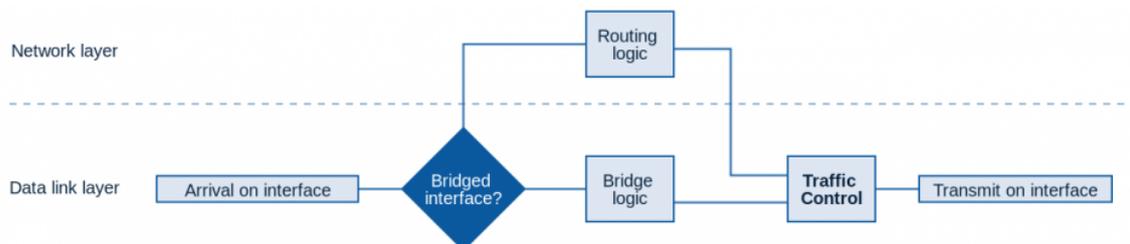


FIGURE 2.9 – Chemin parcouru par un paquet réseau dans le noyau GNU/Linux – source [8]

Lorsqu'un paquet réseau arrive sur une interface réseau, il est analysé par le noyau qui le dirige vers son destinataire s'il est connu. Il est acheminé soit par la couche réseau (OSI 3) soit par la couche liaison (OSI 2) si les interfaces sont pontées, ce qui sera le cas de notre configuration. Avant de sortir sur une interface réseau, le paquet passe à travers le module tc, en couche 2 du modèle OSI.

tc travaille sur les paquets sortant du noyau. Le module tc est le scheduler (ordonnancement) de paquets constamment actif dans un système GNU/Linux. Par défaut, tc

maintien une file d'attente similaire à un FIFO (First In First Out) dans laquelle le premier paquet arrivé est le premier paquet sorti.

L'élément de base de **tc** est la **queueing discipline** (**qdisc**), ou méthode de mise en file d'attente. **tc** propose plusieurs **qdisc** permettant par exemple de limiter le débit (**qdisc tbf**), de prioriser les flux (**qdisc prio**) ou d'effectuer des simulations de réseau (**qdisc netem**).

tc regroupe deux types de **qdisc**, les **qdisc classful** et les **qdisc classless** :

- **qdisc classful** : permettent de répartir les paquets sur différentes files d'attente traitées par ordre de priorité. Elles s'avèrent utiles pour prioriser des paquets par rapport à d'autres, ou pour limiter le débit de certains paquets.
- **qdisc classless** : travaillent sur une unique file d'attente.

L'outil permettant de répartir les paquets entre les différentes files d'attente est **tc filter**. Il peut par exemple classer les paquets en fonction de l'adresse ip ou du port de source ou de destination.

Exemple de configuration de **tc** (fig. 2.10) :

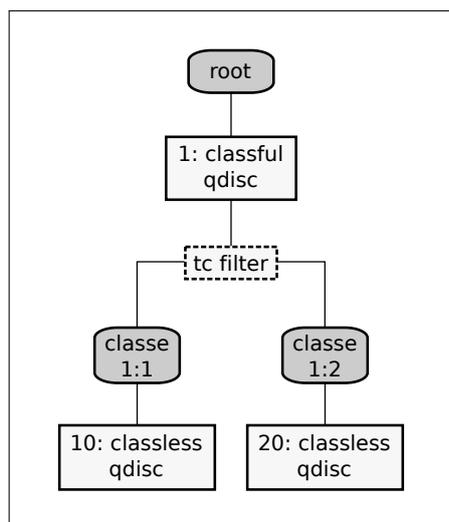


FIGURE 2.10 – Exemple de *queueing discipline*

Lorsque des paquets arrivent dans le module **tc**, ils passent par défaut par la classe “root”. Ils sont immédiatement mis en file d'attente dans la “qdisc 1:” puis au filtre qui les oriente vers la classe “1:1” ou “1:2”. Selon l'aiguillage, les paquets passeront ensuite par la qdisc “10:” ou “20:”. **tc** construit ensuite un flux à partir des données sortant des qdisc “10:” et “20:” en priorisant la “10:” car son indice est plus faible.

Les noms de qdisc sont de la forme $x:$, les noms de classe sont de la forme $x:y$ avec x le nom de la qdisc parente.

netem est une qdisc *classless*. Elle peut donc fonctionner directement après la classe root (figure 2.11.a). Cela constituerait une première configuration simple pour simuler une liaison sans fil.

L'utilisation d'un filtre pour choisir sur quel type de paquet on intervient pourrait s'avérer utile dans nos tests par exemple pour simuler de la latence sur les paquets audio uniquement. Pour ce faire il faut mettre une qdisc *classful* en amont telle que la "qdisc prio". On bénéficie alors de ses sous-classes et par conséquent de sous-files d'attente (fig. 2.11.b).

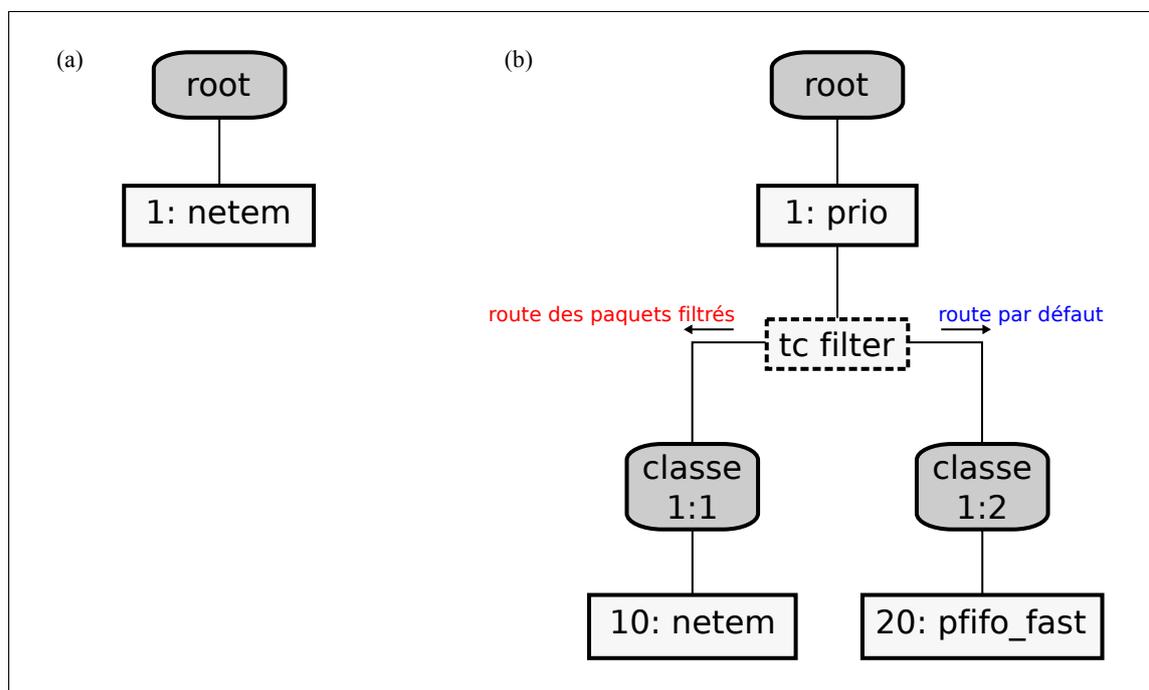


FIGURE 2.11 – Configuration de netem : (a) simple, (b) avec classe

2.3.2 Construction d'une Netem Box

2.3.2.1 Montage de l'ordinateur

Pour les besoins de nos tests nous avons construit notre propre NETEM BOX dans un ordinateur que nous avons appelé CARTON. Il s'agit simplement d'un ordinateur tournant sous GNU/Linux, et possédant deux interfaces réseau gigabit.

Spécifications techniques de CARTON :

- 2x Intel PRO/1000 CT Desktop Adapter Single Port – EXPI9301CTBLK
- CPU
- Carte mère
- Ubuntu 14.04.2LTS

2.3.2.2 Configuration du système

Nous avons configuré les cartes réseau en mode *bridge*. C'est-à-dire que toutes les données arrivant sur l'une, sont immédiatement envoyées sur l'autre. Pour reprendre le schéma de la figure 2.9, les paquets n'ont pas à passer par la couche réseau pour transiter d'une interface à l'autre, le lien se fait directement par la couche liaison.

Notre NETEM BOX possède donc trois interfaces physiques : **eth0** la carte réseau intégrée à la carte mère, **eth1** et **eth2**, les deux interfaces Intel PRO/1000. Ainsi que deux interfaces logiques : la *loopback*⁸, et **br0** le *bridge* que nous venons de créer entre **eth1** et **eth2**.

Le simulateur **netem** agira sur les flux sortant (*egress*) des interfaces physiques **eth1** et **eth2**. Nous aurons donc la possibilité de modifier séparément les caractéristiques des sens console 1 vers console 2, et console 2 vers console 1 (fig. 2.12).

2.3.2.3 Forme des instructions données à netem

Comme nous l'avons vu plus haut, **netem** est un type de **qdisc**. La mise en place d'un module **netem** nécessite la création d'une **qdisc** avec l'utilitaire GNU/Linux **tc**.

Par exemple, la configuration de deux modules **netem** (un sur chaque interface physique) pour obtenir une latence de 1ms avec une gigue de 1ms sera la suivante :

8. présente par défaut sur toutes les machines GNU/Linux, qui leur permet de s'adresser à elles-mêmes sans passer par une interface externe

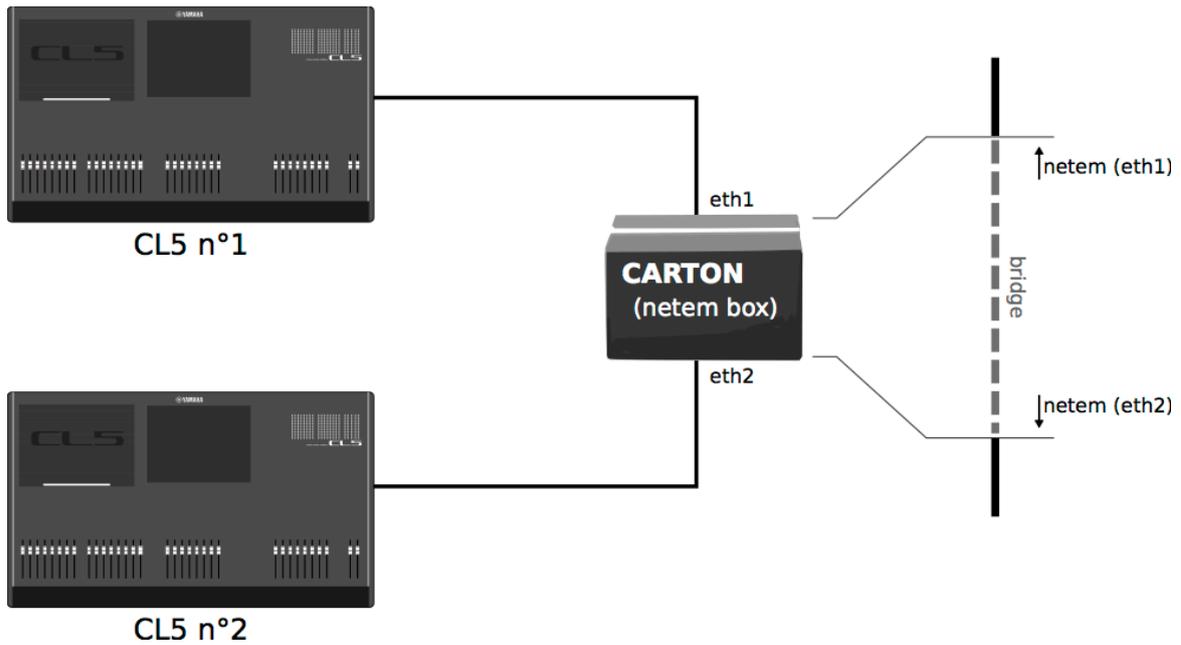


FIGURE 2.12 – Synoptique d'utilisation de la NETEM BOX

```

1 >tc qdisc add dev eth1 root handle 1: netem delay 1ms 1ms
2 >tc qdisc add dev eth2 root handle 1: netem delay 1ms 1ms

```

On peut donner les instructions suivantes à `netem` :

`delay x ms y ms` pour avoir un délai de $x(\pm y)$ ms
`loss n %` pour avoir une perte de paquet de n %
`corrupt k %` pour une corruption de k % des données.

2.3.2.4 Caractérisation de la Netem Box

Avant d'utiliser notre NETEM BOX, il est nécessaire de caractériser ses performances, et notamment de déterminer dans quelle mesure le *bridge* entre les deux interfaces et le module `netem` sont transparent.

Pour cela nous avons effectué des tests avec deux modules `netem` configuré pour introduire 0ms de délai :

```

1 >tc qdisc add dev eth1 root handle 1: netem delay 0ms

```

```
2 >tc qdisc add dev eth2 root handle 1: netem delay 0ms
```

La configuration d'un module `netem` avec comme paramètre un délai de 0ms permet d'effectuer une mesure étalon qui tiendra compte de l'influence du passage à travers le `bridge`, puis l'émulateur.

Protocole de test :

Nous avons effectué l'installation décrite dans la figure 2.12, avec le logiciel `Wireshark` lancé en tâche de fond sur `CARTON`. `Wireshark` nous permet d'intercepter les paquets qui transitent par les interfaces `eth1` et `eth2`.

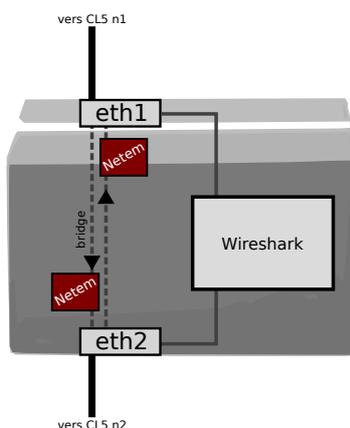


FIGURE 2.13 – Schéma de fonctionnement de CARTON

Après avoir lancé la capture `Wireshark`, nous avons connecté virtuellement à l'aide du `Dante Controller`, 1 canal audio de la `CL5 n° 2` à la `CL5 n° 1`, de telle manière à obtenir un échange de paquet.

Nous avons ensuite analysé la capture `Wireshark`, et calculé la différence de temps entre l'arrivée (par l'interface `eth2`) et le départ (par `eth1`) des paquets. Les résultats sont tracés dans le graphique ci-dessous :

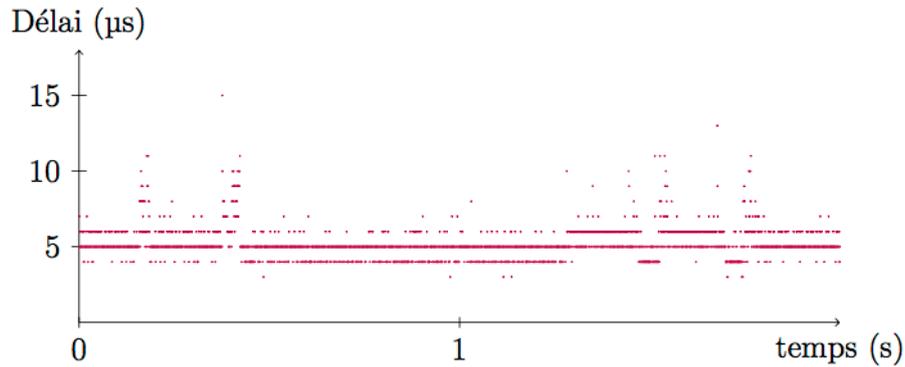


FIGURE 2.14 – Retard mesuré à travers CARTON, en fonction du temps

On constate que, sans délai introduit par `netem`, CARTON introduit un délai de quelques microsecondes, jusqu'à 15µs sur la fenêtre de mesure de 2 secondes.

Cela signifie que les paramètres de latence et de gigue que nous entrerons dans `netem` se feront avec une précision de 15µs environ. Étant donné qu'aucun matériel de réseau sans fil existant à ce jour ne garantit une telle précision, nous serons en mesure d'effectuer un grand nombre de tests dont les résultats seront acceptables dans notre application.

Contrôle du délai simulé par `netem` :

Avant de procéder aux simulations, nous avons voulu vérifier l'efficacité de `netem`. Pour cela nous l'avons configuré pour qu'il ajoute exactement 1ms de latence et 1ms de gigue, puis nous avons effectué une capture `Wireshark`. Cela nous permet de s'assurer que `netem` ajoute bel et bien la latence et la gigue indiquée.

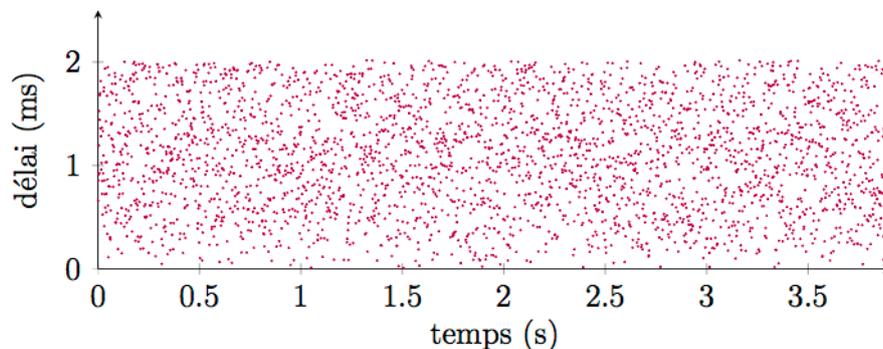


FIGURE 2.15 – Capture de délai effectuée à travers CARTON – avec `netem` 1 ± 1 ms de latence

La mesure du délai relatif entre deux paquets émis successivement, montre que **netem** procède également à un réordonnancement de paquets ; un paquet émis plus tard peut très bien « dépasser » les paquets suivants, pour peu qu’il bénéficie d’un délai très court par rapport aux autres. Cela signifie aussi que la file d’attente **netem** n’est pas une FIFO (*First In First Out*) où les premiers paquets entrant seraient les premiers paquets sortis. Ces paquets arrivés plus tôt sont symbolisés dans le schéma suivant par les valeurs de délai relatif négatives.

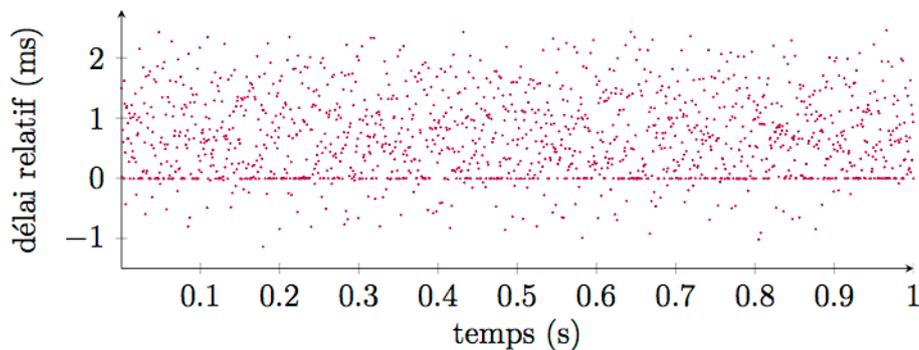


FIGURE 2.16 – Délai mesuré par rapport au paquet contenant les échantillons audio antérieurs – avec **netem** 1 ± 1 ms de latence

2.3.3 Premiers tests avec **netem**

Le but des tests que nous souhaitons effectuer avec **netem** est de déterminer jusqu’à quel point un réseau audionumérique gère la dégradation d’une liaison. Nous allons tenter dans un premier temps d’évaluer l’efficacité de la syntonisation externe décrite dans la partie 2.1.4. Ensuite, nous verrons quelle latence et quelle gigue les appareils audio sont capable de gérer.

2.3.3.1 Test de syntonisation externe

Il s’agit pour cette série de tests de vérifier l’efficacité de la syntonisation des périphériques DANTE sur une *word clock* externe.

Matériel utilisé :

- 2× YAMAHA CL5 (console avec interface DANTE 64 canaux)

- CARTON avec netem et Wireshark
- ordinateur avec Dante Controller
- switch TP-LINK SG3210
- Mutec iCLOCK (horloge externe)

Synoptique de l'installation :

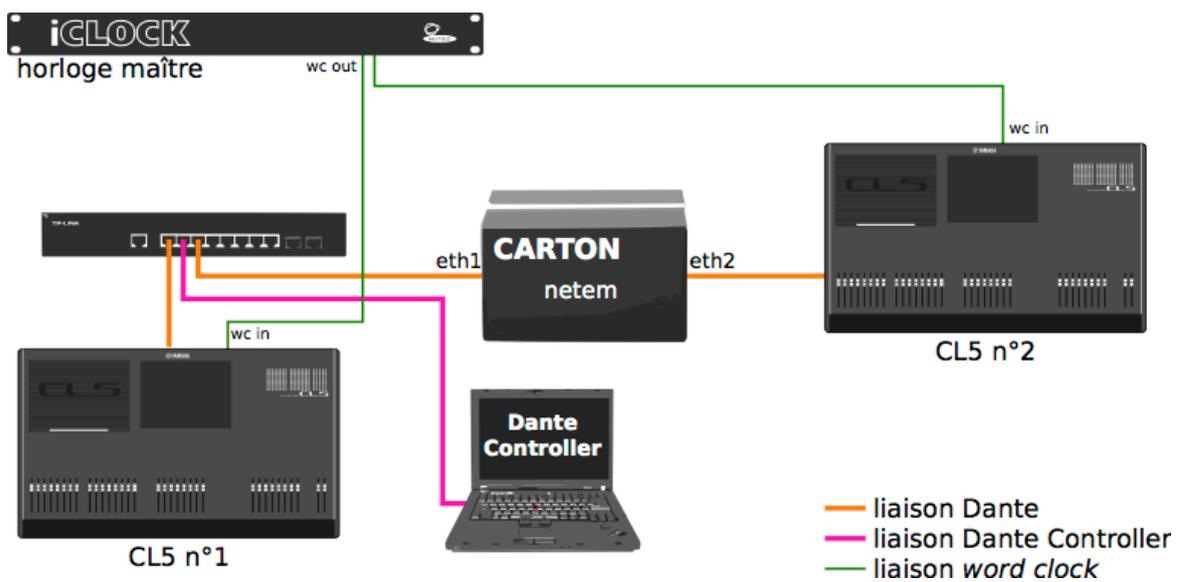


FIGURE 2.17 – Synoptique du test de synchronisation externe

Configuration des périphériques Dante et de la Netem Box :

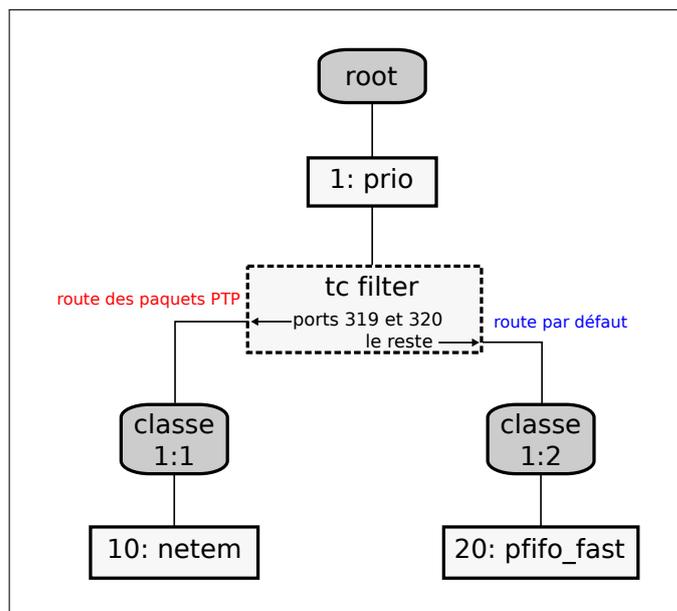


FIGURE 2.18 – Configuration de netem pour le test de synchronisation externe

Les cartes DANTE sont configurées sur une mémoire tampon de 5ms. Nous avons connectés 4 canaux audios dans chaque sens afin d’observer une éventuelle dégradation de l’audio.

netem est configuré de telle manière à pouvoir n’agir que sur les paquets de synchronisation. Pour ce faire nous avons séparé les paquets PTP des autres paquets en nous basant sur le fait qu’ils utilisent les ports UDP 319 et 320. Nous avons ensuite chargé une instance de netem sur cette nouvelle file d’attente (cf. figure 2.18).

Protocole expérimental :

L’opération consistera d’abord à connecter les appareils audio en DANTE à travers la NETEM BOX sans aucune dégradation du flux de synchronisation PTP afin de garantir l’état de synchronisation du système. Nous introduirons ensuite une perte de 100% sur les paquets PTP, pour annihiler la synchronisation via le protocole PTP. Nous devrions alors observer le maintien de la synchronisation dans le cas où le système reste synchronisé en *word clock* externe.

Première manipulation :

Nous allons d’abord voir si, dans le cas où il n’y a **pas de syntonisation externe**,

la destruction du flux de paquet PTP entraîne bien une perte de synchronisation. Pour cette première étape il faut donc mettre les consoles en configuration standard, c'est-à-dire en mode esclave par rapport à leur carte DANTE.

Immédiatement après la suppression des paquets PTP sur le réseau, nous avons observé l'apparition de deux horloges maîtres dans la fenêtre de synchronisation du Dante Controller. S'ensuit une détérioration très forte de la qualité audio de la transmission, qui s'accroît dans le temps par paliers.

L'apparition de deux périphériques maîtres en synchronisation dans le Dante Controller quoique surprenante, s'explique assez facilement ; chaque périphérique ne voyant plus dans son environnement PTP de meilleure horloge PTP que lui-même, il passe automatiquement maître en synchronisation. Cela n'a pas d'influence sur l'autre, puisqu'il n'est pas en mesure de voir qu'un autre maître se trouve de l'autre côté de la NETEM BOX.

La détérioration de l'audio montre que la suppression de la synchronisation PTP a bien des conséquences sur l'état de synchronisation de l'installation.

Deuxième manipulation :

Maintenant que nous sommes sûrs de pouvoir détruire la synchronisation PTP de manière efficace, nous allons refaire le test **avec une syntonisation externe** et comparer les résultats.

Après suppression de la synchronisation, l'outil Dante Controller voit à nouveau l'apparition de deux périphériques maîtres PTP sur son réseau. Contrairement à tout à l'heure, aucune détérioration audio ne se fait ressentir. Après 10 minutes de test sans aucun problème de synchronisation, nous pouvons avancer que la syntonisation externe a une influence positive sur le maintien de l'état de synchronisation du système.

Troisième manipulation :

La troisième manipulation vise à déterminer si la cohabitation entre une syntonisation externe correcte, et une synchronisation PTP éronnée est possible. Au lieu de

configurer *netem* pour qu'il perde 100% des paquets, nous lui avons fait simuler une latence de 1ms accompagnée de 1ms de gigue. Ces caractéristiques auront pour conséquence de maintenir le lien PTP entre les deux machines qui effectueront cependant de fortes erreurs d'interprétation de la synchronisation PTP.

Nous ne sommes pas parvenu à maintenir un état de synchronisation dans cette configuration. La CL5 n° 2 (définie comme esclave sur réseau DANTE) synchronisée en *word clock* externe affiche par un voyant clignotant qu'elle détecte une mauvaise synchronisation sur le réseau DANTE. Cela correspond à l'information donnée par Dante Controller, qui signale aussi que la CL5 n° 2 s'est mise en mode *mute* (muet). On peut expliquer ce phénomène par le fait que la carte DANTE présente dans la CL5 n° 2, esclave à la fois sur le réseau DANTE et de la console elle-même, reçoit deux informations contradictoires de synchronisation.

Résultats et conclusions :

Ces manipulations ont permis de montrer que l'on pouvait utiliser une syntonisation externe pour conserver l'état de synchronisation des modules DANTE. Nous avons cependant remarqué que le maintien d'une synchronisation PTP erronée bousculait cet état. Dans les prochains tests il conviendra de supprimer l'échange de paquets PTP lorsqu'on utilise une syntonisation externe avec un pare-feu⁹ par exemple.

Dans la mesure où la syntonisation permet seulement de conserver un état de synchronisation, il faudra qu'au moment où l'on décide d'interrompre le transit des paquets PTP, les appareils partagent le même état de synchronisation. Cela se fait simplement en connectant les appareils (syntonisés en externe) en filaire le temps qu'ils se synchronisent.

9. cet outil implémenté dans la plupart des routeurs permet notamment de bloquer les paquets provenant d'une adresse ou d'un port donné

Voici la procédure que nous suivrons pour initialiser la synchronisation :

Procédure de synchronisation par syntonisation externe :

- 1- connecter les appareils audio à une même référence d'horloge, les synchroniser à cette source externe
- 2- configurer les périphériques DANTE pour qu'ils suivent le mode de synchronisation externe
- 3- connecter les périphériques en filaire
- 4- une fois synchronisés, on peut déconnecter la liaison filaire
- 5- les périphériques ont une référence de date commune qui est incrémentée par une syntonisation externe commune ; ils sont synchronisés

Cette procédure étant compliquée à mettre en œuvre, et se reposant sur un détournement des mécanismes utilisés par DANTE (filtrage des ports UDP 319 et 320), nous ne la jugerons acceptable que dans un cadre expérimental.

2.3.3.2 Test de simulation

Il s'agit pour cette série de tests de mesurer l'impact de la dégradation des liaisons d'un réseau audionumérique sur l'audio. Cela a pour but de vérifier les données inscrites dans le tableau 2.2.

Matériel utilisé :

- 2× YAMAHA CL5 (console avec interface DANTE 64 canaux)
- CARTON avec netem et Wireshark
- ordinateur avec Dante Controller
- switch TP-LINK SG3210
- Mutec iCLOCK (horloge externe)

Synoptique de l'installation :

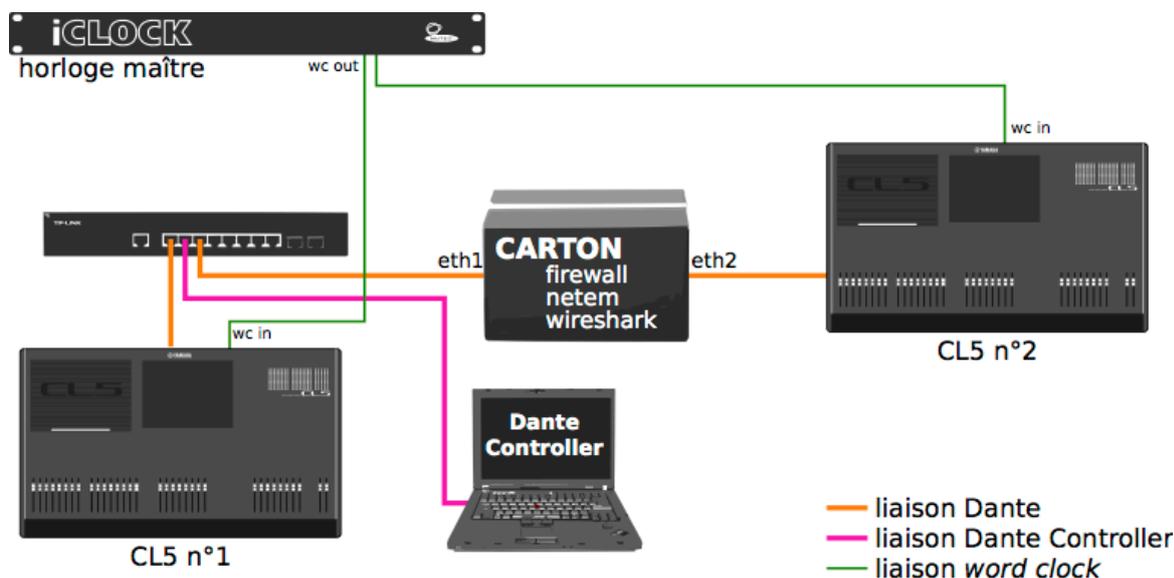


FIGURE 2.19 – Synoptique du test de simulation

Configuration des appareils et de la Netem Box :

Les consoles sont configurées en mode esclave par rapport à leur *word clock* externe, les cartes DANTE sont toutes les deux configurées sur *external sync* dans le Dante Controller. On utilise aussi des mémoires tampons de 5ms.

Nous avons configuré un pare-feu sur CARTON afin de bloquer les paquets de synchronisation, et mis en place deux instances de **netem** (une sur chaque interface) qui nous permettront d'entrer les paramètres à simuler.

Le logiciel **Wireshark** est également lancé sur l'ordinateur afin d'enregistrer les paquets passant par les interfaces physiques **eth1** et **eth2**.

Protocole expérimental

Nous souhaitons mesurer les dégradations subies par nos signaux audio après avoir parcouru le réseau. Nous avons donc mis en place le dispositif expliqué en annexe B qui nous permet de voir et d'entendre en direct, la moindre erreur de transmission.

Nous incrémenterons peu à peu la valeur du délai et celle de la gigue, jusqu'à obtention des premières erreurs.

Résultats

Ces tests ont permis de montrer que conformément à ce qui est annoncé, le tampon permet bien d'amortir des latences, des pics de latence tant qu'ils restent inférieurs à la valeur du tampon.

Ligne de commande entrée dans `netem`, pour une latence de $X \pm 1\text{ms}$:

```
1 >tc qdisc add dev eth1 root handle 1: netem delay Xms 1ms
```

X	1	2	3	3,2	3,3
Délai max (ms)	2	3	4	4,2	4,3
Transmission	ok	ok	ok	ok	erreurs

TABLE 2.3 – Résultats du premier test

On note que l'on peut monter précisément jusqu'à des pics de latence de 4,3ms, ce qui correspond à : tampon – temps de paquet = 5ms – 0,7ms = **4,3ms**.

Nous pouvons aussi déduire de cette expérience que les périphériques DANTE ne semblent pas sensibles au réordonnancement de paquet, ce qui est une bonne chose puisque les réseaux sans fil ne sont pas toujours transparent quant à l'ordre de transmission des paquets.

2.4 Première liaison sans fil avec syntonisation externe

Maintenant que nous avons défini une méthode pour nous affranchir des problèmes de synchronisation que nous connaissions jusqu'alors, nous allons effectuer les premiers essais de liaison audio sans fil avec syntonisation externe. Ainsi pourrions-nous déterminer si la synchronisation est le seul point qui fait problème dans le transport d'un réseau audionumérique à travers une liaison IEEE 802.11.

Pour ce test nous avons mis en place une liaison IEEE 802.11n sur laquelle nous allons tenter de transporter un réseau DANTE.

Matériel utilisé :

- 2× YAMAHA CL5 (console avec interface DANTE 64 canaux)
- ordinateur avec Dante Controller
- switch TP-LINK SG3210
- MUTECH ICLOCK
- 2× UBIQUITI NANOBRIDGE M5

Synoptique de l'installation :

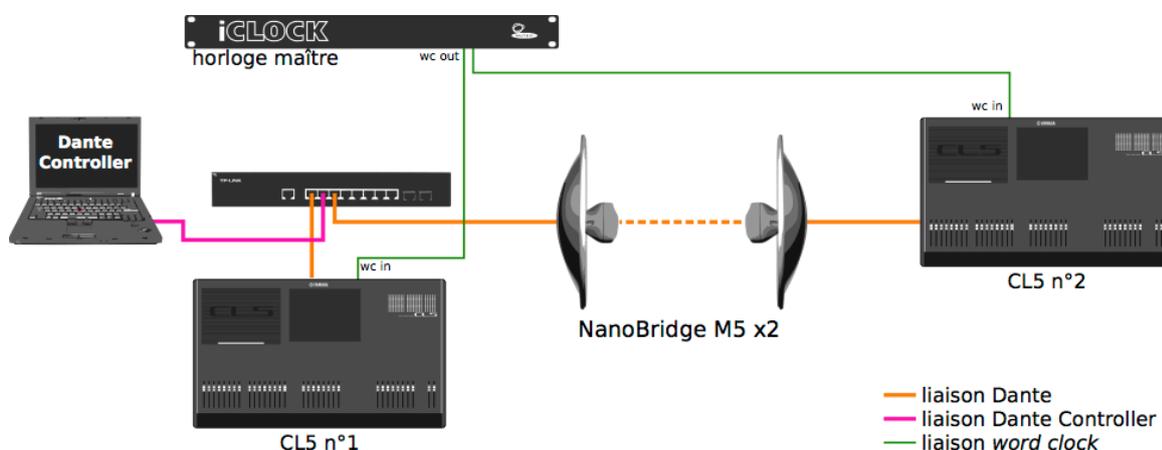


FIGURE 2.20 – Synoptique de la liaison sans fil avec syntonisation externe

Configuration des consoles et des antennes :

Les antennes sont placées à 10m l'une de l'autre en vision directe. La fréquence de transmission a été déterminée compte tenu de l'encombrement spectral mesuré sur place (cf. annexe D). Paramètres de la liaison :

- MCS12 (qui correspond à la valeur automatiquement sélectionnée comme étant la plus appropriée)
- 2x2 MIMO
- largeur de bande de 40MHz
- intervalle de garde court.

Nous sommes dans la configuration d'horloge décrite pour fonctionner en syntonisation externe. C'est-à-dire que les cartes DANTE sont syntonisées sur l'horloge des consoles qui sont elles-mêmes esclaves par rapport à l'horloge maître (iCLOCK). On a

de plus configuré un pare-feu sur les routeurs NANOBIDGE M5 interdisant les ports 319 et 320.

Protocole :

On contrôle en permanence la bonne transmission du son, selon la méthode décrite en annexe B.

La première étape sera de synchroniser les périphériques DANTE entre eux, conformément à la procédure décrite p.49. Nous incrémenterons ensuite le nombre de canaux jusqu'à apparition des premières erreurs sur l'audio. Le nombre de canaux connectés sera toujours un multiple de 4, puisque le DANTE regroupe les canaux par 4.

Résultats :

La première colonne décrit le nombre de canaux connectés de cette manière : $x \times y$ signifie x entrées connectées dans la CL5 n° 1 et y dans la la CL5 n° 2.

nb. canaux	état de l'audio
4 × 4	quelques erreurs
4 × 8	beaucoup d'erreurs
8 × 8	perte totale de l'audio

Nous avons également effectué un relevé des latences indiquées par Dante Controller dans l'onglet « latency » des périphériques. Les valeurs indiquées sont sous la forme « moyenne/crête ». Nous rappelons que dans notre cas des latences supérieures à 5ms (taille de la mémoire tampon pour la lecture de l'audio) s'avèreront problématiques.

nb. canaux	CL5 n° 2 → CL5 n° 1 (µs)	CL5 n° 1 → CL5 n° 2 (µs)
4 × 4	3,7/3,9	2,4/2,6
4 × 8	IMP*	IMP*

* la valeur de la latence est si grande qu'elle n'est pas affichée

Ces premiers résultats sont quelque peu décevants. De fait, on aurait pu s'attendre à obtenir une liaison fonctionnelle avec un nombre réduit de canaux.

Ce test était sans doute encore un peu précoce. Pour comprendre les éléments qui posent problème, nous avons décidé d'étudier plus en détail le fonctionnement d'une liaison IEEE 802.11, notamment ses caractéristiques en terme de débit et de latence.

Chapitre 3

Optimisation des réseaux sans fil – améliorations possibles de la couche physique

Pour être applicables dans le monde de l'audio professionnel, les transmissions doivent se faire dans un temps très court pour garantir un minimum de latence. Une autre condition à réaliser pour permettre le transport de la synchronisation est d'avoir une liaison « temps réel », condition sinequanone au transport de l'audio.

Traitement en temps réel, mode de fonctionnement d'un système informatique dans lequel l'ordinateur élabore [...] des informations de commande [...] dans un intervalle de temps très faible vis-à-vis de l'évolution du processus avec lequel il est en relation.

*Dictionnaire Larousse en ligne,
<http://www.larousse.fr/dictionnaires/francais/reel/67387/>*

Pour un traitement audionumérique, on peut dire qu'il est « temps réel » s'il existe un réel positif l , tel que, pour tout réel positif τ , la durée de traitement des échantillons audio représentant τ secondes soit inférieur à $\tau + l$, et si l , qui correspond à la latence introduite par le système, est suffisamment petit (les valeurs de l acceptables étant définies par l'application). Autrement dit, à n'importe quel instant, on peut prédire qu'un échantillon arrivé au temps t , ressorte au temps $t + l$. Dans notre cas où le traitement correspond au transport de l'audio, dans une application *live*, l devra

être de l'ordre de la milliseconde, et strictement inférieur à 10ms (valeur retenue dans l'AES67).

Étant initialement conçu pour des applications informatiques qui n'exigent pas d'envoyer ou de recevoir en un temps précisément défini, on peut se demander à quel point les liaisons IEEE 802.11 répond à cette définition du temps réel.

Ce chapitre a pour but d'analyser dans quelle mesure nous pourrions tirer parti des capacités d'un réseau sans fil IEEE 802.11 pour transporter de l'audio en optimisant au maximum les échanges de données entre deux stations.

Nous verrons aussi quelles mesures peuvent être prises pour acheminer correctement le signal de synchronisation sans fil.

3.1 Optimisation des réseaux sans fil

Dans cette partie nous nous concentrerons sur l'optimisation d'une liaison point à point IEEE 802.11 pour le transport de données audio (paquets RTP).

De prime abord une liaison sans fil semble être un bon candidat pour le transport de données audio à grande vitesse. La norme 802.11n proposant des débits pouvant atteindre les 600 Mbit/s¹ et la norme 802.11ac passant la barre du gigabit (jusqu'à près de 7 Gbit/s dans les meilleures conditions²), autant dire que nos données audio ne représentent pas une lourde charge pour un réseau sans fil de dernière génération.

Ces valeurs de débit ne doivent cependant pas être prises telles quelles. Il s'agit du débit pratiqué par l'émetteur une fois qu'il a acquis le droit d'émettre sur sa bande de fréquence. Les plages de fréquence dites « des 2,4 GHz » et « des 5 GHz » sont partagées par de nombreux appareils (bien que la plage des 5 GHz compte moins d'utilisateurs à l'heure actuelle). Des mécanismes de vérification de disponibilité du médium de transport (la bande de fréquence dans notre cas) sont donc nécessaires pour éviter toute interférence entre émetteurs.

1. en MCS 31 avec un intervalle de garde court

2. voir tableau 1.3

3.1.1 Rappel des mécanismes problématiques pour notre application

*Distributed coordination function*³ (DCF) est le mécanisme fondamental utilisé par l'IEEE 802.11. Il repose sur l'utilisation du protocole CSMA/CA, *Carrier Sense Multiple Access with Collision Avoidance*⁴ qui consiste à écouter le support avant d'émettre afin de savoir s'il est disponible. . Avant d'envoyer une trame, l'émetteur vérifie que le médium est libre. S'il ne l'est pas, il attendra un temps aléatoire (*Backoff Time*) avant de retenter sa chance.

La présence de ce mécanisme nous rappelle que les émissions entre deux stations se font nécessairement à tour de rôle ; deux stations ne peuvent pas émettre l'une vers l'autre de manière simultanée⁵, ce qui signifie que la liaison est *half duplex*.

Un deuxième mécanisme vient améliorer le CSMA. Il s'agit de l'échange RTS/CTS (*Ready To Send / Clear To Send*, « prêt à envoyer / la voie est libre »). Ce mécanisme a pour but d'anéantir le problème des noeuds cachés, situation courante dans laquelle deux stations communiquent avec un même point d'accès mais ne se voient pas. Si l'une d'entre-elles commence à émettre vers le point d'accès, l'autre n'en a pas l'information et risque d'émettre au même moment, ce qui a pour conséquence d'interférer au niveau du point d'accès. Pour éviter cette situation, la station qui souhaite émettre envoie un message RTS *Request To Send* (demande pour émettre) qui contient des informations sur le volume des données et sa vitesse de transmission. Le récepteur répond alors un *Clear To Send* (signifiant que le champ est libre pour émettre), puis la station commence l'émission des données. Les autres stations recevant aussi le message CTS sauront qu'une station est en train d'émettre. Elles activent alors leur *Network Allocation Vector* (vecteur d'allocation réseau) pendant le temps de la transmission pour simuler l'occupation de leur propre radio.

Un message ACK (*Acknowledgment*) est envoyé par le récepteur pour signaler à l'émetteur que la transmission s'est bien passée.

3. fonction de coordination distribuée

4. écoute d'un support à accès multiple avec évitement de collision

5. avec du matériel conforme à la norme 802.11

Le fonctionnement de ce mécanisme témoigne de l'application envisagée pour les réseaux 802.11 : l'intégrité des données reçues prime sur leur temps de parcours. Le schéma ci-dessous met en évidence son déroulé au cours du temps.

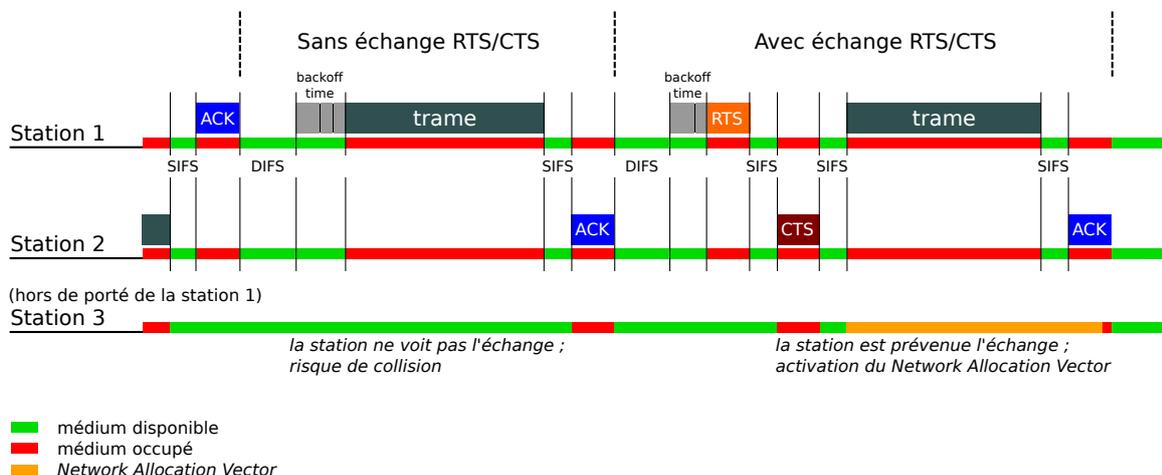


FIGURE 3.1 – Déroulé simplifié des échanges entre deux stations

Différents temps intervenant dans l'échange :

SIFS (*Short Interframe Space*) est le temps requis par une interface sans fil pour traiter les trames reçues et répondre avec une autre trame. Celui-ci vaut $16\mu\text{s}$ ⁶.

DIFS (*DCF Interframe Space*) est le temps au bout duquel si le médium de transport est libre, la station le considèrera comme tel et pourra envisager d'émettre. Il vaut $34\mu\text{s}$ ⁶.

Backoff time il s'agit d'un temps aléatoire divisé en *slot* de $9\mu\text{s}$. Ce temps a pour but d'éviter l'envoi simultané de deux trames par deux stations différentes. Chacune d'elle aura un temps de *backoff* différent, et celle qui aura choisi le plus court aura le droit d'émettre la première. Le *backoff time* dure $n \times 9\mu\text{s}$ ⁶ avec n un nombre entier aléatoire le plus faible possible. Dans la suite des calculs, on considèrera ce temps en moyenne égal à $36\mu\text{s}$.

Les messages **ACK** (14 octets), **RTS**(20 octets) et **CTS**(14 octets) durent environ $50\mu\text{s}$ chacun⁷.

6. pour les normes 802.11n (à 5GHz) et 802.11ac

7. B. MILLER, *A Voice Of Reason On Voice Over WiFi*, blog, 2015. adresse : <http://www.sniffwifi.com/2015/02/the-voice-qos-category-of-no-reason-to.html>

On décompte donc les temps que l'on appellera temps moyen de négociation T_{nego} (nécessaire à l'envoi de chaque trame de données) :

- **sans RTS/CTS** : $T_{CSMA} = T_{DIFS} + T_{backoff} + T_{SIFS} + T_{ACK} = 136\mu\text{s}$
- **avec RTS/CTS** : $T_{RTS/CTS} = T_{DIFS} + T_{backoff} + T_{RTS} + 3 \times T_{SIFS} + T_{CTS} + T_{ACK}$
 $= 268\mu\text{s}$

Un mécanisme alternatif propose que la station émettrice s'envoie à elle-même un message CTS. Il s'agit du *CTS-to-self*. Cela permet de simplifier les échanges et de gagner du temps. Le *CTS-to-self* ne règle pas le problème des noeuds cachés, il s'agit cependant d'une protection utile dans un environnement mixte, c'est-à-dire un environnement comprenant aussi des stations utilisant des anciennes normes comme 802.11a.

Il existe également la possibilité de désactiver l'accusé de réception ACK.

Ces deux dernières options permettent de réduire le temps moyen à :

- **CTS-to-self/sans ACK** : $T_{C2S,noACK} = T_{DIFS} + T_{backoff} + T_{CTS} + T_{SIFS} = 136\mu\text{s}$

Le problème majeur que nous rencontrons dans le temps de négociation, est la présence du temps aléatoire $T_{backoff}$, qui empêche notamment l'utilisation du protocole PTP pour la synchronisation.

3.1.2 Calcul du temps de parcours d'un paquet

Dans notre cas, on entend par temps de parcours, le temps que met un paquet à parcourir le lien sans fil. Il est déterminé par différents facteurs ; le temps de propagation de l'onde dans l'air, le temps d'attente avant que le médium ne soit libre, le temps de négociation avant d'émettre, ainsi que le temps de modulation des données (celui-ci dépend du débit de la liaison). Dans notre cas nous négligerons le temps de

propagation des ondes dans l'air (0,33µs pour 100m).

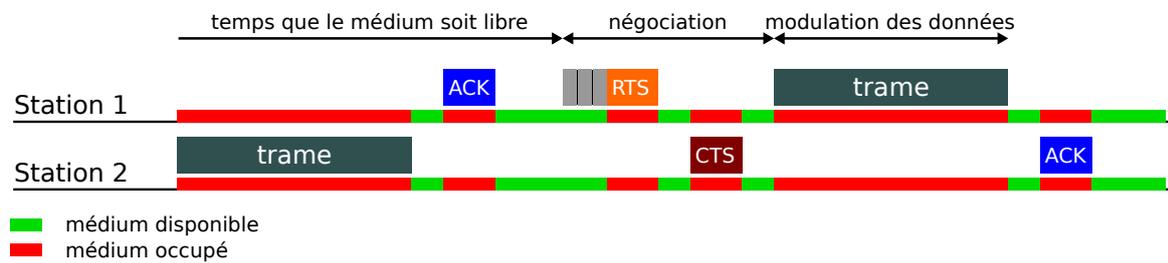


FIGURE 3.2 – Temps de transmission d'un paquet

3.1.2.1 Durée d'une trame PLCP

Les trames de la couche PHY (représentées en vert foncé dans la figure 3.2) contiennent de nombreux champs d'information que nous n'étudierons pas en détail, mais dont les temps sont indiqués dans la figure ci-dessous.

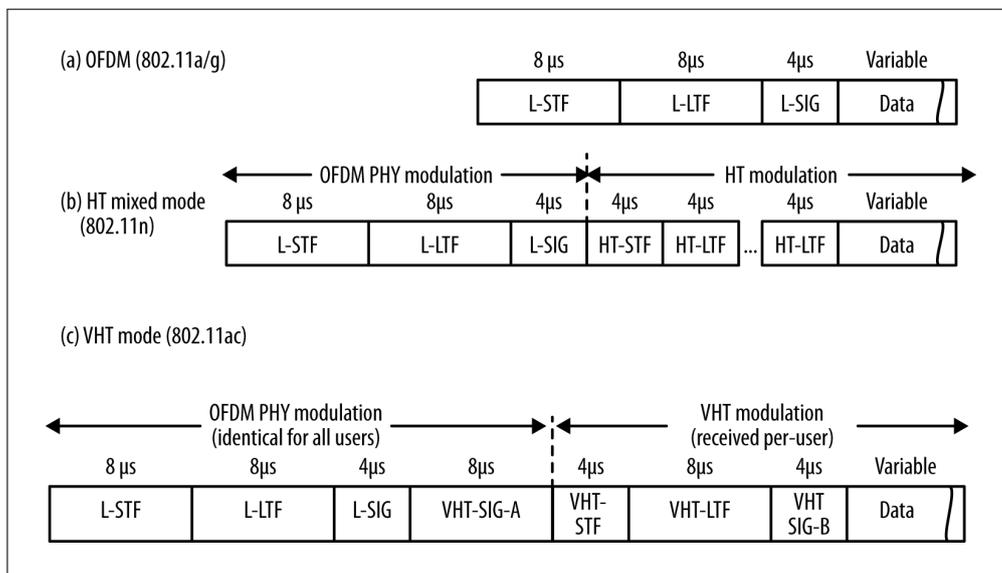


FIGURE 3.3 – Format de trame de la couche PHY pour les normes 802.11n et 802.11ac – source [11]

- **Partie préambule**

On retiendra de ce schéma la durée $T_{preambule}$ du préambule des trames de la couche PHY :

- 802.11n : 28µs (pour 1 flux spatial), 40µs (pour 4)
(avec 1 champ « HT-LTF » par flux spatial)

- 802.11ac : 44µs

• **Partie *data* (données)**

La partie *data* représentée figure 3.3 contient un ou plusieurs paquets IP (selon le paramètre d'agrégation choisi) successivement encapsulés dans une trame de la couche MAC appelée MPDU⁸, puis agrégés dans une A-MPDU (*Aggregate-MPDU*). Le schéma ci-dessous représente le détail de cette partie *data* :

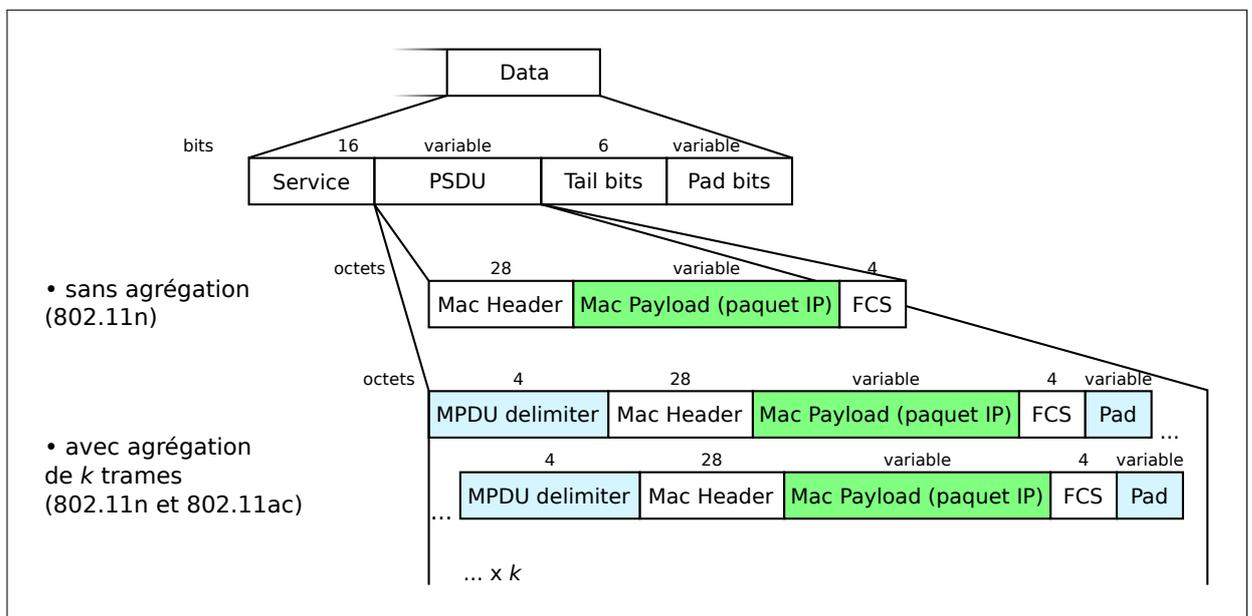


FIGURE 3.4 – Format de la MPDU (ou A-MPDU pour des trames agrégées) - schéma simplifié

Le champ « service » sert à initialiser le scrambler⁹.

« Tail bit » : utilisé pour les algorithmes de vérification d'erreur.

« FCS » : somme de contrôle pour vérifier l'intégrité de chaque trame ou sous-trame indépendamment.

Les champs « pad » servent à ajuster les sous-frames pour les faire correspondre à des valeurs entières de symboles.

8. MPDU : *MAC Protocol Data Unit* ; unité de données du protocol MAC

9. voir [14], p.36

Le « MAC Header » contient des informations notamment sur la durée de la trame, sa source et sa destination.

Le header de la couche PHY vaut donc :

$$S_{headerPHY} = S_{service} + S_{tailbit} + S_{pad} \quad (3.1)$$

$$= 16 + 6 + S_{pad} \quad (3.2)$$

$$\Rightarrow \begin{cases} 22 + S_{pad} \\ \text{avec } S_{pad} \text{ (en bit)} / S_{pad} + S_{PSDU} + 22 \equiv 0[N] \end{cases} \quad 10$$

$$\Rightarrow 22 \leq S_{headerPHY} < 22 + \frac{N}{8} \quad [\text{octets}]$$

$$\Rightarrow 176 \leq S_{headerPHY} < 176 + N \quad [\text{bits}] \quad (3.3)$$

où N est le nombre de bits par symbole, et S_{PSDU} la taille du PSDU.

Pour les calculs, nous utiliserons la plus grande valeur possible du header, c'est-à-dire $S_{headerPHY} = 175 + N$ bits.

On décompte aussi une taille de header pour une trame MPDU de :

$$\begin{aligned} S_{headerMPDU} &= S_{headerMAC} + S_{FCS} \quad [+S_{delimiter} + S_{pad}] \\ &\quad \text{si trames agrégées} \\ &= 28 + 4 \quad [+4 + S_{pad}] \end{aligned}$$

$$\text{si trame non-agrégée} \Rightarrow S_{headerMPDU} = 32 \text{ octets (= 256 bits)} \quad (3.4)$$

$$\text{si trame agrégée} \Rightarrow S_{headerMPDU} = \begin{cases} 36 + S_{pad} \\ \text{avec } S_{pad} \text{ (en bit)} / S_{pad} + S_{paquet} + 36 \equiv 0[N] \end{cases}$$

$$\Rightarrow 36 \leq S_{headerMPDU} < 36 + \frac{N}{8} \quad [\text{octets}]$$

$$\Rightarrow 288 \leq S_{headerMPDU} < 288 + N \quad [\text{bits}] \quad (3.5)$$

où N est le nombre de bit par symbole, et S_{paquet} la taille du paquet IP définie p.32.

Comme au dessus, pour les calculs nous simplifierons en utilisant la plus grande valeur

10. Puisqu'on utilise une modulation OFDM (voir partie 1.2.1.2) chaque trame est codée en symboles, représentant chacun N bits. La taille du champ « pad » est alors calculé de telle manière à ce que la taille totale de la trame soit un multiple de N .

possible, soit $S_{\text{headerMPDU}} = 287 + N$ bits pour des trames agrégées, et $S_{\text{headerMPDU}} = 256$ bits pour des trames seules (en norme 802.11n¹¹). Par souci de simplification, nous ne considèrerons que le cas où les trames sont transférées comme trames agrégées.

À partir de ces tailles de *header*, on peut calculer le temps qu'il faut pour les transmettre en fonction du débit DR (*data rate*) de notre liaison :

$$\begin{aligned} T_{\text{headerMAC/PHY}} &= \frac{S_{\text{headerPHY}} + k.S_{\text{headerMPDU}}}{DR} & (3.6) \\ &= \frac{175 + N(k + 1) + 257k}{DR} \quad \text{avec } k \text{ le nb de trames agrégées} \end{aligned}$$

Application numérique : pour l'envoi d'une seule trame en MCS4 (16-QAM, *code rate* = 3/4)¹², 2 flux spatiaux, sur une bande de 40MHz.

$$DR = 43,3 \times 2 \times 2,1 = 181,86 \text{ Mbit/s}$$

en 16-QAM on a 4 bits/symbole avec un taux de 3/4, il reste $N = 3$

on peut donc calculer le temps de header MAC et PHY :

$$T_{\text{headerMAC/PHY}} = \frac{175 + 3 \times 2 + 257}{181,86 \cdot 10^6} = 2,41 \cdot 10^{-6} \text{ s} = 2,41 \mu\text{s}$$

3.1.2.2 Calcul du temps de transmission pour un paquet de 424 octets

Dans le cadre de cette application numérique, nous considèrerons un réseau standard 802.11n, utilisant le mécanisme RTS/CTS, utilisant systématiquement des trames agrégées, et dont les paramètres de débit sont les mêmes que dans l'application numérique juste au-dessus.

11. rappelons que la norme 802.11ac ne considère plus que des trames agrégées, même lorsqu'une seule trame est transmise

12. voir tableau 1.2 p.13

Soit un paquet RTP/IP de 424 octets, similaire à ceux utilisés par le DANTE.

$$\begin{aligned}
 T_{transmission} &= T_{nego} + T_{preamble} + T_{headerMAC/PHY} + \frac{S_{paquet}}{DR} & (3.7) \\
 &= 268 + 32 + 2,41 + \frac{424 \times 8}{181,86 \cdot 10^6} \\
 &= 321 \mu s
 \end{aligned}$$

Il est à remarquer que la partie négociation CSMA/CA avec message RTS/CTS contribue à presque 85% au temps de transmission.

3.1.2.3 Temps avant que le médium ne se libère

Pour terminer de calculer le temps de parcours moyen d'un paquet sur le réseau, il faut encore considérer le temps nécessaire pour que le médium se libère. Étant donné que ces temps sont en grande partie déterminés par des évènements contingents, au mieux nous pourrions encadrer la valeur du temps de parcours pour une situation donnée.

On peut envisager différents scénarios (toutes les unités sont des μs) :

1er scénario : le paquet arrive alors que l'appareil est déjà en train d'émettre :

Le paquet devra alors attendre que la transmission se termine pour pouvoir être transmis à son tour. Supposons que le paquet précédent soit de même nature que celui-ci ^a, s'ajoute donc à son temps de transmission, un temps d'attente $T_{attente} \in]0; T_{trans}]$. On a donc :

$$\begin{aligned}
 T_{parcours} &= T_{trans} + T_{attente} \\
 \Rightarrow \mathbf{T}_{parcours} &\in]\mathbf{T}_{trans}; \mathbf{2T}_{trans}] \\
 \Rightarrow T_{parcours} &\in]321; 642]
 \end{aligned}$$

a. avec un temps de transmission T_{trans} identique

2ème scénario : le paquet arrive alors que l'appareil a déjà p paquets en file d'attente (y compris celui en cours d'émission) :

Dans le cas où il s'agit de paquets de même nature que celui qui arrive, le même raisonnement qu'au-dessus nous permet de donner une fourchette pour le temps d'attente du paquet $T_{attente} \in](p-1) \cdot T_{trans}; p \cdot T_{trans}]$. On a donc :

$$\Rightarrow \mathbf{T}_{\text{parcours}} \in]\mathbf{p} \cdot \mathbf{T}_{\text{trans}}; (\mathbf{p} + \mathbf{1}) \cdot \mathbf{T}_{\text{trans}}]$$

3ème scénario : le paquet arrive alors qu'une autre station est en train d'émettre :

Dans notre cas, il s'agit de la situation dans laquelle la liaison 802.11 est utilisée dans les deux sens. Étant donné qu'une liaison 802.11 standard est *half duplex*, il faut attendre que le médium ne soit plus utilisé par une autre station pour émettre. Cela revient donc à la situation du premier scénario ; le temps d'attente $T_{attente}$ est égal au temps restant pour la transmission en cours. Pour un paquet de même nature, $T_{attente} \in]0; T_{trans}]$. D'où :

$$\Rightarrow \mathbf{T}_{\text{parcours}} \in]\mathbf{T}_{\text{trans}}; \mathbf{2T}_{\text{trans}}]$$

4ème scénario : le paquet arrive alors que plusieurs stations ont des paquets en file d'attente :

Il s'agit de la situation la plus problématique. Les stations devront écouler tous leurs paquets avant de pouvoir émettre le dernier arrivant. L'utilisation du mécanisme CSMA/CA ne garantit pas le fait que les stations écouleront leur file d'attente à tour de rôle. À chaque fois, ce sera celle avec le $T_{backoff}$ le plus court qui remportera la transmission.

Imaginons que la station 1 (sur laquelle arrive le paquet) ait déjà 1 paquet en cours d'émission, et la station 2, 2 paquets en file d'attente. Le pire des cas pour notre paquet, serait de devoir attendre l'émission du paquet en cours, et celle des deux paquets de la station 2. On a donc $T_{attente} \in]2; 3T_{trans}]$, soit :

$$\Rightarrow T_{\text{parcours}} \in]3T_{\text{trans}}; 4T_{\text{trans}}] =]\mathbf{963}; \mathbf{1284}]$$

On remarque que le temps de parcours d'un paquet à travers la liaison dépasse rapidement la valeur du temps de paquet (qui vaut $667\mu\text{s}$ dans nos expériences avec du DANTE). Lorsqu'on transmet les paquets *un à un*, un temps de parcours supérieur au temps de paquet (*i.e.* à la période d'émission d'un paquet) entraîne une rapide accumulation de paquets en file d'attente des routeurs, qui ne seront pas distribués à temps, et finira irrémédiablement par provoquer des pertes de données audio. Cela signifie donc que le débit n'est pas suffisant.

3.1.3 Calcul de débit effectif de données

Les calculs et l'expérience menée à la fin du chapitre précédent ont montré que la latence de parcours apportée par une liaison 802.11 était intrinsèquement liée à la charge du réseau. Elle est due en grande partie aux temps de négociation entraînés par les mécanismes anti-collision de la couche PHY. Ces temps de parcours ne sont pas sans conséquence sur le débit effectif de la liaison (*throughput* en anglais).

Nous entendons par débit effectif la quantité en bits de paquets IP (de taille S_{paq}) réellement transmise par unité de temps.

Les calculs qui suivent tenteront de quantifier l'impact du temps de parcours sur ce débit effectif. Nous verrons également en quoi l'agrégation de trame permet d'augmenter le débit effectif en rentabilisant les temps de négociation et de préambule.

Dans un premier temps, rappelons la définition du débit de transmission :

$$D_{eff} = \frac{k \cdot S_{paq}}{T_{trans}}$$

nous introduisons ici le coefficient k , le nombre de trames agrégées qui s'avèrera utile à la fin du calcul. Le temps T_{trans} est donc le temps de transmission de la trame agrégée contenant k trames.

D'après les équations (3.7) et (3.6),

$$T_{trans} = T_{nego} + T_{preamb} + \frac{S_{headerPHY} + k \cdot S_{headerMPDU} + S_{paq}}{DR}$$

On a donc :

$$\begin{aligned}
 D_{eff} &= \frac{k.S_{paq}}{T_{nego} + T_{preamb} + \frac{S_{headerPHY} + k(S_{headerMPDU} + S_{paq})}{DR}} \\
 &= DR \times \frac{k.S_{paq}}{S_{headerPHY} + k(S_{headerMPDU} + S_{paq}) + DR(T_{nego} + T_{preamb})} \quad (3.8)
 \end{aligned}$$

Application numérique : toujours dans les mêmes conditions que dans les calculs précédents (MCS4, 2 flux spatiaux, intervalle de garde court, 40MHz, avec RTS/CTS).

dans ces conditions, on rappelle que le nombre de bits (utiles) par symbole vaut $N = 3$,

$$S_{headerPHY} = 175 + N = 178 \text{ bit} \text{ et } S_{headerMPDU} = 257 + 3 = 300 \text{ bit}$$

on a donc :

$$D_{eff} = DR \times \frac{k.S_{paq}}{178 + k(300 + S_{paq}) + DR(268 + 32) \cdot 10^{-6}}$$

ce qui nous permet de tracer le graphique suivant :

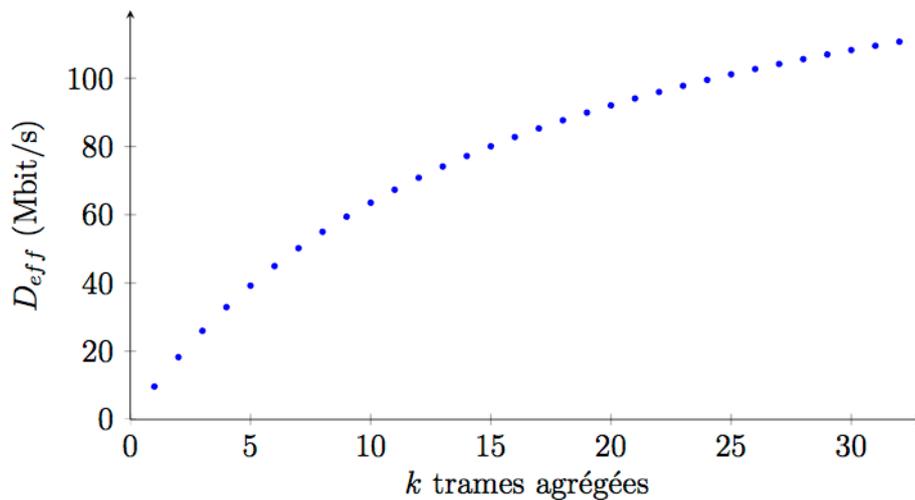


FIGURE 3.5 – Évolution du débit efficace en fonction du nombre de trames agrégées, pour des paquets de 424 octets

On se rend compte très nettement que les débits effectifs dont est capable une liaison 802.11 sont bien inférieurs au débit de la liaison. Cela est tout à fait normal puisqu'il s'agit de deux mesures différentes. En anglais on parle de *data rate* et de *throughput*.

Le *throughput* calculé ci-dessus doit être partagé dans le cas d'un échange *half duplex*. Cela signifie par exemple que sans agrégation de trame avec un débit effectif partagé entre deux stations (c'est le cas dans lequel nous avons réalisé le dernier test), nous avons à disposition un débit effectif de $D_{eff} = \frac{9,66}{2} = 4,83\text{Mbit/s}$ ce qui est inférieur au débit requis par la liaison audio calculé p.34 qui valait 5,09Mbit/s.

3.1.4 Trames agrégées

L'utilisation de trames agrégées semble obligatoire étant donné les débits requis par un réseau AES67 (de l'ordre de 20Mbit/s par sens de communication pour une liaison bidirectionnelle de 16 canaux audio).

Les NanoBridge M5 offrent la possibilité d'agréger les trames selon le mode d'agrégation A-MPDU. L'utilisateur indique au routeur IEEE 802.11 le nombre maximal de trames qu'il souhaite agréger dans une même trame A-MPDU. Les NanoBridge M5 permettent d'agréger jusqu'à 32 trames. Cela ne signifie pas que l'émetteur attende d'avoir 32 trames pour les transmettre d'un seul coup, mais qu'à chaque occasion d'émettre celui-ci transmettra au plus 32 trames simultanément.

Afin de mesurer l'influence de mise en agrégation de trames, nous avons réalisé le même test que celui effectué pour mesurer la latence p.53, avec un nombre maximal de 32 trames par trame agrégée.

Résultats :

Valeurs indiquées par Dante Controller (moyenne/crête) :

nb. canaux	CL5 n° 2→CL5 n° 1 (ms)	CL5 n° 1→CL5 n° 2 (ms)
0 × 4	n.a.	1,9/2,3
0 × 8	n.a.	2,3/2,8
0 × 12	n.a.	2,4/2,8
0 × 16	n.a.	2,4/2,8
4 × 4	4/4,9	2,3/3
4 × 8	3,7/4,1	2,7/3,4
4 × 12	4/4,2	2,9/3,2
4 × 16	4,2/5	3,2/4,3
8 × 8	4,6/4,9	3,1/3,7
8 × 12	4,7/5,5	3,3/4,1
8 × 16	4,9/6	3,6/4,9
12 × 12	5,2/5,4	4/4,8
12 × 16	5,1/5,3	4,3/5
16 × 16	6,1/6,8	5/5,9

Le premier élément à noter à la vue des résultats obtenus en activant l'agrégation de trames, est que certaines configurations de canaux, qui étaient impossibles lors du précédent test, se sont avérées possibles.

Lorsque l'on compare les nouvelles valeurs de latence avec celles obtenues sans agrégation de trame, on constate qu'elles ont certes diminué, mais qu'elles restent particulièrement élevées compte tenu de l'étude que l'on vient de mener dans ce chapitre.

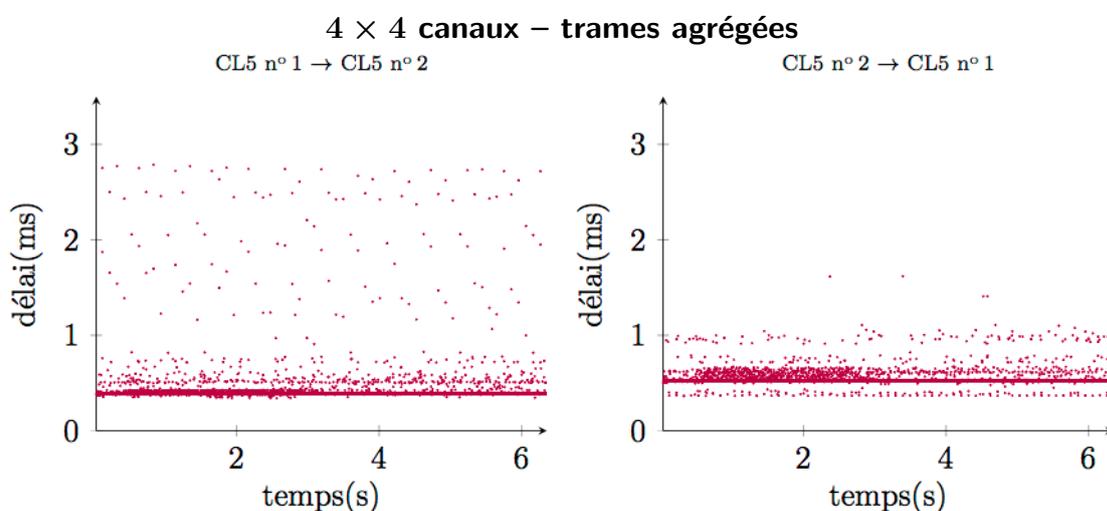
Prenons l'exemple de la latence mesurée pour 4 × 4 canaux. Le paramètre d'agrégation sélectionné permettant de regrouper l'envoi de 32 trames MPDU (donc 32 paquets IP), on peut supposer qu'à aucun moment, un paquet n'aura à attendre l'envoi d'autres paquets pour être transmis à son tour (hormis ceux déjà en cours de transmission). On peut comparer cette situation à celle d'un usager des transports en commun qui ne serait jamais contraint de laisser passer de bus avant de pouvoir monter à bord puisqu'ils seraient surdimensionnés.

Le temps de parcours auquel on aurait pu s'attendre serait donc celui obtenu dans

le cas du « scénario 3 » décrit précédemment (p.65) qui correspond au cas où le paquet doit attendre la fin de la transmission de l'autre station sans fil. Dans ce cas on a un temps de parcours de l'ordre de 700µs (valeurs arbitrairement majorées). Après avoir ajouté le temps de paquet égal à 667µs, on obtient une latence de l'ordre de 1,8ms; valeur 2 à 3 fois inférieur au temps mesurés.

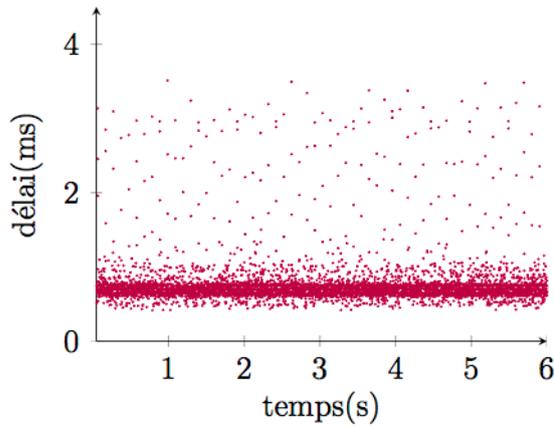
Afin d'avoir une vision chronologique de la latence, nous avons également effectué quelques captures de part et d'autre de la liaison sans fil avec l'outil WIRESHARK. En comparant les dates de capture d'un même paquet RTP avant et après le pont sans fil, on calcule sa valeur de la latence de parcours sur le lien sans fil.

Pour être comparées avec les latences indiquées par Dante Controller, ces valeurs de délai doivent être majorée du temps de paquet. Nous signalons cependant que par souci pratique, les valeurs de latence relevées dans Dante Controller, n'ont pas été effectuées au même instant que les captures WIRESHARK présentées ici. Il se peut donc que les résultats ne soient pas rigoureusement identiques.

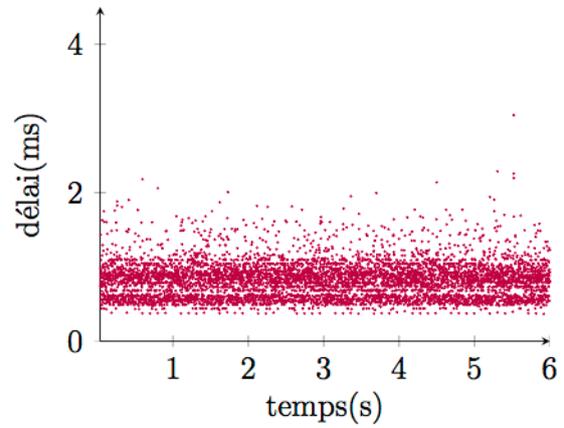


8 × 8 canaux – trames agrégées

CL5 n° 1 → CL5 n° 2

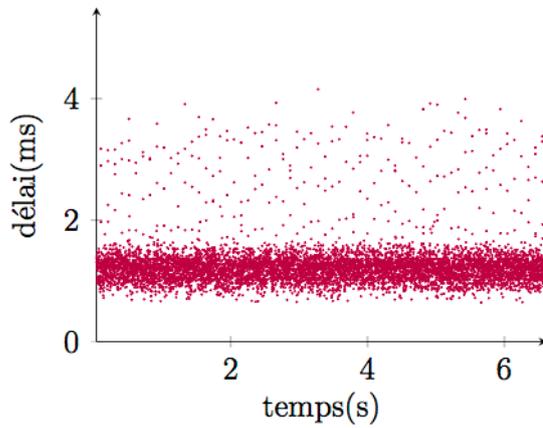


CL5 n° 2 → CL5 n° 1

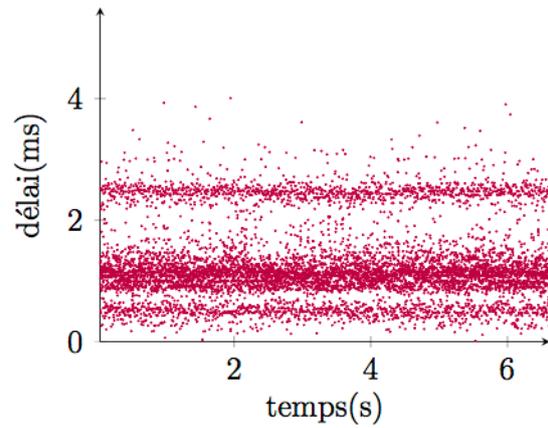


12 × 12 canaux – trames agrégées

CL5 n° 1 → CL5 n° 2

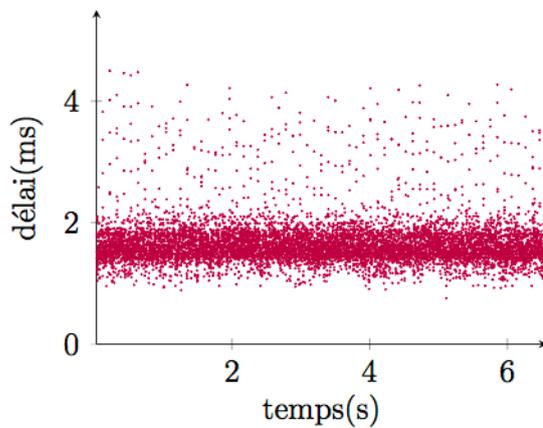


CL5 n° 2 → CL5 n° 1

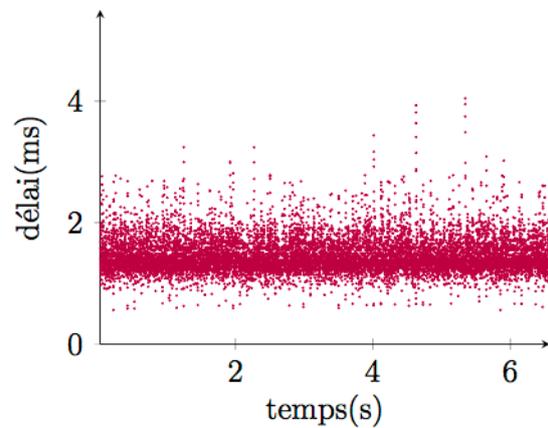


16 × 16 canaux – trames agrégées

CL5 n° 1 → CL5 n° 2



CL5 n° 2 → CL5 n° 1



Dans le but de simplifier l'interprétation des graphiques de latence obtenus, nous allons tenter de réaliser une analogie avec les déplacements en avion.

Imaginons que les passagers représentent des paquets RTP qui arrivent par groupes à l'aéroport (station sans fil) à intervalle de temps régulier (temps de paquet).

Un groupe de passagers arrive à l'aéroport pour prendre l'avion. Soucieux d'arriver à destination le plus vite possible, ils se précipitent dans la file d'attente pour se faire enregistrer. Une fois arrivé au guichet, le dernier passager du groupe apprend avec regret que bien que l'avion n'ait pas encore décollé, et qu'il reste de nombreuses places, la procédure d'enregistrement est close¹³ ; il devra attendre le prochain vol ... Incapable de dire au voyageur combien de temps il devra attendre, on lui explique au guichet que la mauvaise visibilité dans l'air, empêche les avions de décoller puisqu'on ne peut être sûr qu'aucun autre avion n'est actuellement en train de circuler¹⁴. Ainsi, ne sachant jamais quelle sera la fréquentation dans les aéroports, la compagnie aérienne décide qu'à chaque fois que la voie sera libre, elle tirera au sort l'aéroport qui enverra le prochain avion¹⁵. Autant dire que personne ne peut vraiment prévoir quand aura lieu le prochain départ.

Nous proposons maintenant d'interpréter les graphiques obtenus en considérant que le délai représente le temps qui s'est écoulé entre l'arrivée du passager à l'aéroport et son arrivée à destination. Les passagers les moins chanceux seront ceux qui auront attendu le plus longtemps.

On observe une asymétrie entre les deux sens de la liaison. Cela s'explique par le fait qu'une des stations émet des trames de manière plus fréquente ; elle a tendance à avoir un *backoff time* inférieur à l'autre.

La présence d'amas sur les graphiques témoigne du fait qu'il existe des temps de délai plus fréquents que d'autres. Cela signifie qu'il existe une concordance entre l'arrivée des paquets et le départ des trames. Cependant les écarts importants par rapport à la valeur moyenne témoignent, pour reprendre notre analogie, d'une mauvaise gestion des horaires et de la répartition des vols allers et retours.

13. début de la négociation (mécanisme CSMA/CA et RTS-CTS)

14. transmission *half duplex*

15. grâce au *backoff time* de la station

Dans le cas de 4 canaux aller-retour, on constate que les latences mesurées semblent permettre une liaison sans erreur de transmission. Nous avons réalisé une capture de 5 minutes dans cette situation qui n'a révélé aucune erreur sur l'audio. Les configurations comptant plus de canaux n'ont pas fait l'objet de tests. Cela dit, le tableau de valeurs de latences mesurées avec **Dante Controller** permet d'avoir une idée des configurations possibles.

Des mécanismes alternatifs à ceux utilisés par l'IEEE 802.11 permettraient peut-être d'avoir une gestion plus adaptées du *duplex* de la liaison (*i.e.* sa capacité à gérer les deux sens de transmission).

Des protocoles alternatifs de transmission existent et ont déjà fait leurs preuves dans les travaux précédents. Il serait de fait intéressant d'effectuer de nouveaux tests avec le matériel MIKROTIK (technologie *full-duplex*) et une syntonisation externe.

3.2 Liaison *full-duplex*, une alternative au CSMA/CA

La partie précédente traitait le cas de l'utilisation d'une liaison conforme au standard IEEE 802.11. Nous avons notamment évoqué l'impact qu'avait l'utilisation de ses mécanismes de gestion/partage du médium de transport (le CSMA/CA et éventuellement de l'échange RTS-CTS) sur le débit effectif et la latence de la liaison.

Nous avons aussi remarqué lors de nos tests, que dans le cas d'un échange bidirectionnel, nous étions rapidement limités par cette gestion inadaptée des temps de paroles qui entraînait notamment des pics de latence indésirables.

Dans le cas d'une liaison point à point, la méthode la plus efficace permettant de partager le médium de transport entre les stations est l'utilisation de deux fréquences de transmission ; l'une pour le Rx (réception) et l'autre pour le Tx (émission), il s'agit de la seule solution véritablement *full-duplex* ; c'est-à-dire permettant l'envoi et la réception de données simultanément.

3.2.1 Présentation de la solution *full-duplex*

Il existe peu de produits sur le marché utilisant une technologie sans fil *full-duplex*. Cela vient du fait qu'en informatique, peu d'applications nécessitent des débits aussi élevés et une latence aussi faible que ce que proposent les liaisons *full-duplex*.

On entend par *full-duplex* le partage du médium de transport qui utilise la technique du FDD (*Frequency Division Duplex*) dont le principe repose sur l'utilisation de deux fréquences de transmission : l'une pour l'émission (Rx), l'autre pour la réception (Rx).

Parmi le matériel réseau « grand public » permettant de faire une liaison sans fil *full-duplex*, on trouve facilement les produits de la gamme AIRFIBER d'UBIQUITI, et les produits MIKROTIK implémentant le protocole propriétaire Nstreme dual. S'agissant de la seule solution qui pouvait entrer dans le budget d'un mémoire, c'est elle qui avait fait l'objet du choix de Léo ROSSI-ROTH pour ses tests.

Les tests de transmission menés auparavant avec le matériel *full-duplex* MIKROTIK se sont avérés fructueux puisqu'ils ont permis de réaliser une première liaison fonctionnelle de deux canaux audio aller-retour avec le protocole RAVENNA.

Dans le deuxième chapitre de ce mémoire, nous avons tenté de réaliser une liaison similaire avec du matériel DANTE (voir partie 2.1.3.2, p.25), qui s'est avérée peu stable d'un point de vue de la synchronisation ; nous avons constaté une grande gigue d'horloge.

Nous allons tenter dans cette partie de combiner la solution *full-duplex* à l'utilisation d'une syntonisation externe.

3.2.2 Antennes utilisées pour la liaison

Une attention particulière doit être portée au choix des antennes. Celui-ci doit être fait en tenant compte de l'environnement, et de la cohabitation qu'il peut y avoir avec

d'autres émetteurs et récepteurs radio. De fait, même si deux couples émetteur/récepteur communiquent sur des fréquences différentes, les interférences entre eux restent possibles.

Dans le cadre de nos expérimentations, nous avons utilisé quatre antennes dites « secteurs » (une paire par sens de transmission). Ce type d'antennes que l'on retrouve surtout dans le déploiement des réseaux GSM, a une directivité particulière. Sur le plan horizontal (H-PLANE), elles ouvrent sur une portion de cercle (*secteur* doit donc être pris au sens géométrique du terme), et sur le plan vertical, sa directivité est très serrée (E-PLANE).

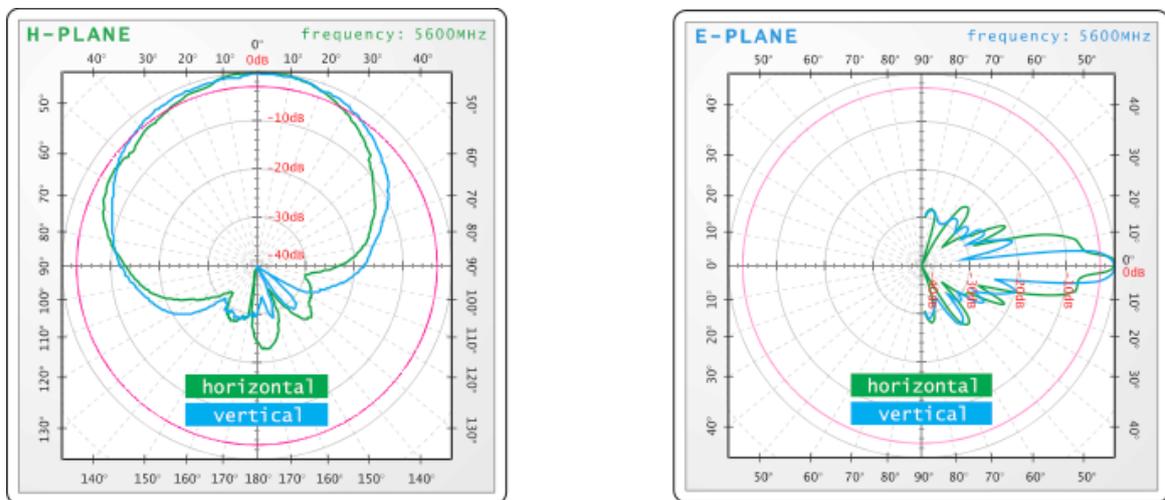


FIGURE 3.6 – Diagrammes de directivité des antennes – voir annexe C

Ce type d'antennes est particulièrement adapté pour couvrir une zone délimitée.

Chaque antenne dispose d'une double polarisation, ce qui nous permettra d'utiliser le 2x2 MIMO des cartes réseau sans fil installées dans les routeurs.

Distance entre les antennes

Afin de limiter les interférences entre les antennes d'émission et de réception, nous les avons montées sur deux pieds différents, le plus éloigné possible compte tenu de la longueur des câbles dont nous disposons (c'est-à-dire à 2m environ l'une de l'autre).

Choix des canaux de transmission

Le choix des fréquences de transmission s'est fait au regard des disponibilités mesurées avec l'analyseur AirView (annexe D). Le manuel d'utilisation des routeurs MIKROTIK indique aussi que les canaux utilisés pour le Rx et le Tx doivent être au minimum distants de 200MHz¹⁶.

Pour optimiser au mieux le choix des fréquences de transmission, il est préférable de regarder les courbes de gain des antennes.

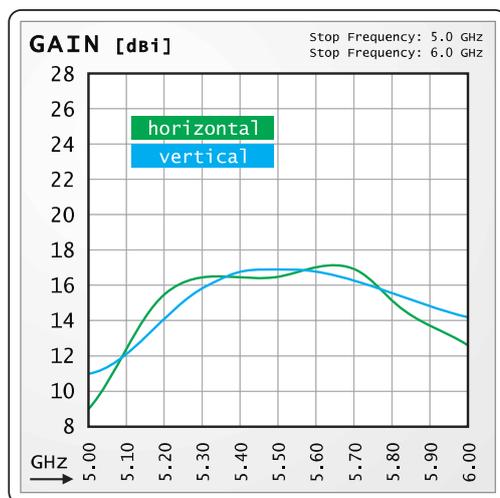


FIGURE 3.7 – Courbe de gain des antennes – voir annexe C

3.2.3 Le protocole Nstreme dual

Il s'agit d'un protocole entièrement propriétaire. Par conséquent nous ne pourrions qu'émettre des hypothèses quant à son fonctionnement.

Sa mise en service décrite dans [18] requiert l'utilisation de deux interfaces physiques IEEE 802.11n par routeur ; une utilisée uniquement pour la transmission et l'autre pour la réception de données. Ces deux interfaces sont dites « esclaves » d'une interface virtuelle nommée `nstreme1`.

Ce principe de fonctionnement nous laisse penser que Nstreme dual utilise peut-être les mécanismes de la couche PHY du standard IEEE 802.11. Chaque lien IEEE 802.11

16. <http://wiki.mikrotik.com/wiki/Manual:Interface/Wireless>

serait alors dans la situation particulière où une seule station utiliserait le canal de transmission.

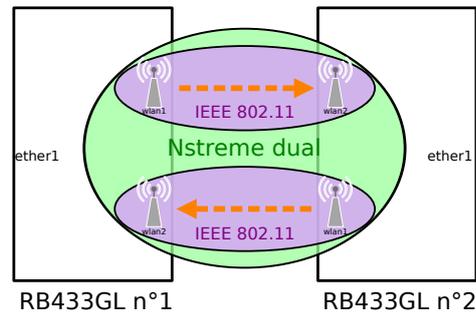


FIGURE 3.8 – Hypothèse sur les mécanismes du Nstreme dual

La présence du paramètre « *disable CSMA* » (désactiver CSMA) dans la fenêtre de configuration de l'interface `nstreme1` soutient cette hypothèse ; par défaut les interfaces physiques utiliseraient le CSMA (défini dans le standard IEEE 802.11) comme méthode d'accès au support. L'utilisation du CSMA en *full duplex* prend son sens uniquement si d'autres réseaux IEEE 802.11 sont proches et utilisent l'une des deux fréquences déjà occupées par la liaison. Dans le cas inverse, la désactivation du CSMA prend son sens et entraîne un gain en performances.

Étant donné que nous ne pourrions pas aller plus loin dans la compréhension précise des mécanismes utilisés, nous avons choisi une approche empirique qui consistera à expérimenter l'influence des différents paramètres accessibles via l'interface utilisateur des routeurs.

3.2.4 Mise en place de la liaison pour les tests

Dans un premier temps, nous allons mettre en place la liaison *full-duplex* avec syntonisation externe, sans porter d'attention particulière au paramétrage avancé des routeurs MIKROTIK.

Les seuls réglages que nous avons effectués consistent simplement à choisir les canaux sans fil de manière rigoureuse, et une MCS adaptée.

Synoptique

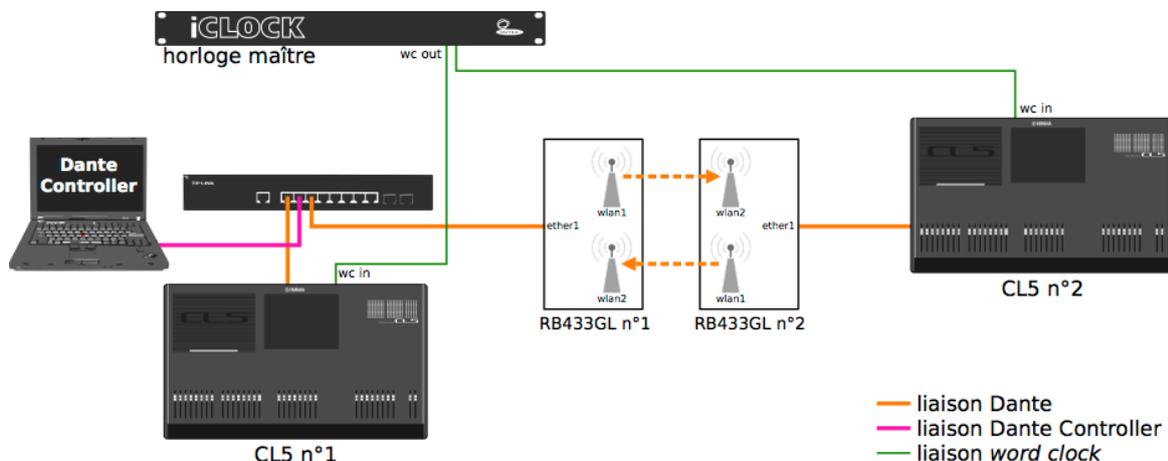


FIGURE 3.9 – Synoptique de la liaison MIKROTIK avec syntonisation externe

Configuration

Les consoles sont configurées de manière à suivre la méthode de vérification de transmission que nous avons établie dans l'annexe B.

Les cartes sans fil utilisées par les routeurs MIKROTIK sont configurées pour utiliser les fréquences 5580MHz et 5320MHz, des largeurs de bande de 40MHz et une « HT4¹⁷ » ce qui équivaut à une MCS11 (débit de 108Mbps avec un intervalle de garde long, et une largeur de canal de 40MHz).

Protocole

Le principe de la manipulation est de monter progressivement le nombre de canaux de la liaison, afin de noter une éventuelle apparition d'erreurs.

Résultats

Les résultats de ce test sont prometteurs dans la mesure où nous arrivons à augmenter de manière significative le nombre de canaux par rapport à celui que nous obtenions lorsque nous utilisons une liaison standard IEEE 802.11 et ceci sans que l'on note un nombre important d'erreurs.

17. les valeurs « HT » sont l'équivalent MIKROTIK des MCS défini dans le standard IEEE 802.11

Pour cette première expérimentation du dispositif *full-duplex* avec syntonisation externe, nous n'avons pas souhaité aller au-delà de 16 canaux dans chaque sens. Nous estimons de fait que dans le cadre de ce mémoire, une transmission fonctionnelle de 16 canaux dans les deux sens de transmission serait déjà une grande avancée. Dans la suite des tests, nous nous sommes fixé pour objectif d'atteindre ce nombre.

3.2.5 Paramètre d'agrégation

L'interface de configuration du *Nstreme dual* donne accès au paramètre d'agrégation de trames de la liaison. La valeur à entrer est en octets et correspond à la quantité maximale d'octets que pourra contenir une trame agrégée. D'après la documentation, cette valeur doit être comprise entre 64 et 4000. Un deuxième paramètre auquel on a accès est le *framer policy* (méthode de mise en trame) qui peut prendre trois valeurs :

- **none** (pas d'agrégation)
- **best fit** (les trames seront remplies de manière à approcher au plus près la taille maximale sélectionnée précédemment, sans fragmentation des trames MPDU)
- **exact size** (les trames feront exactement la taille indiquée précédemment, au risque de fragmenter les trames MPDU).

Dans notre cas, et conformément aux recommandations de l'AES67 ([7], p.16), nous utiliserons le paramètre **best fit** pour éviter de fragmenter les paquets audio.

Un parallèle avec la partie précédente nous permet de supposer que la modification de ce paramètre pourrait optimiser la liaison.

Le prochain test visera donc à déterminer l'influence du paramètre d'agrégation de trames.

Pour ce faire, on fixe le nombre de canaux transmis à 16 canaux DANTE dans chaque sens, et on augmente par paliers le paramètre d'agrégation. On espère alors observer, une diminution du nombre d'erreur.

Nous avons mis en place la méthode de contrôle de qualité de transmission décrite

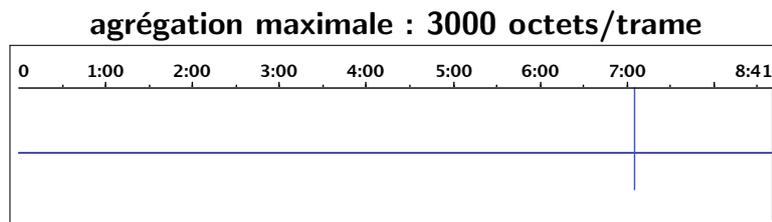
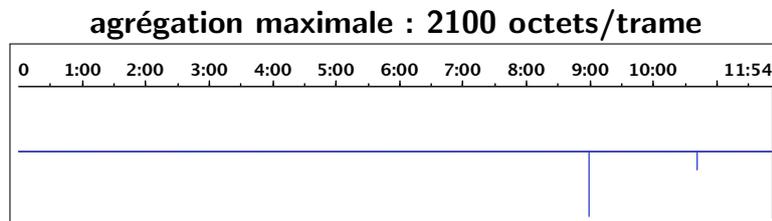
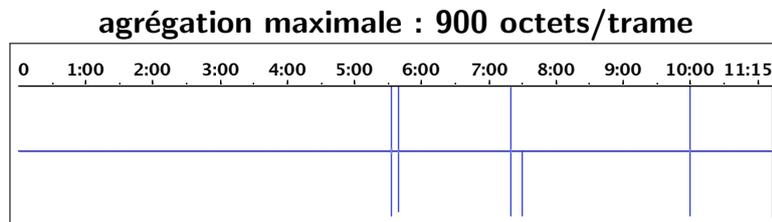
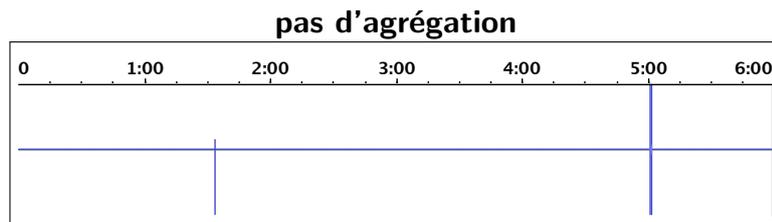
dans l'annexe B, et réalisé un enregistrement des erreurs à l'aide de l'enregistreur interne à la CL5 n° 1.

L'outil de capture WIRESHARK analyse les paquets de part et d'autre du pont sans fil, ce qui nous permettra après exploitation des captures, d'apprécier les évolutions de la latence au cours du temps.

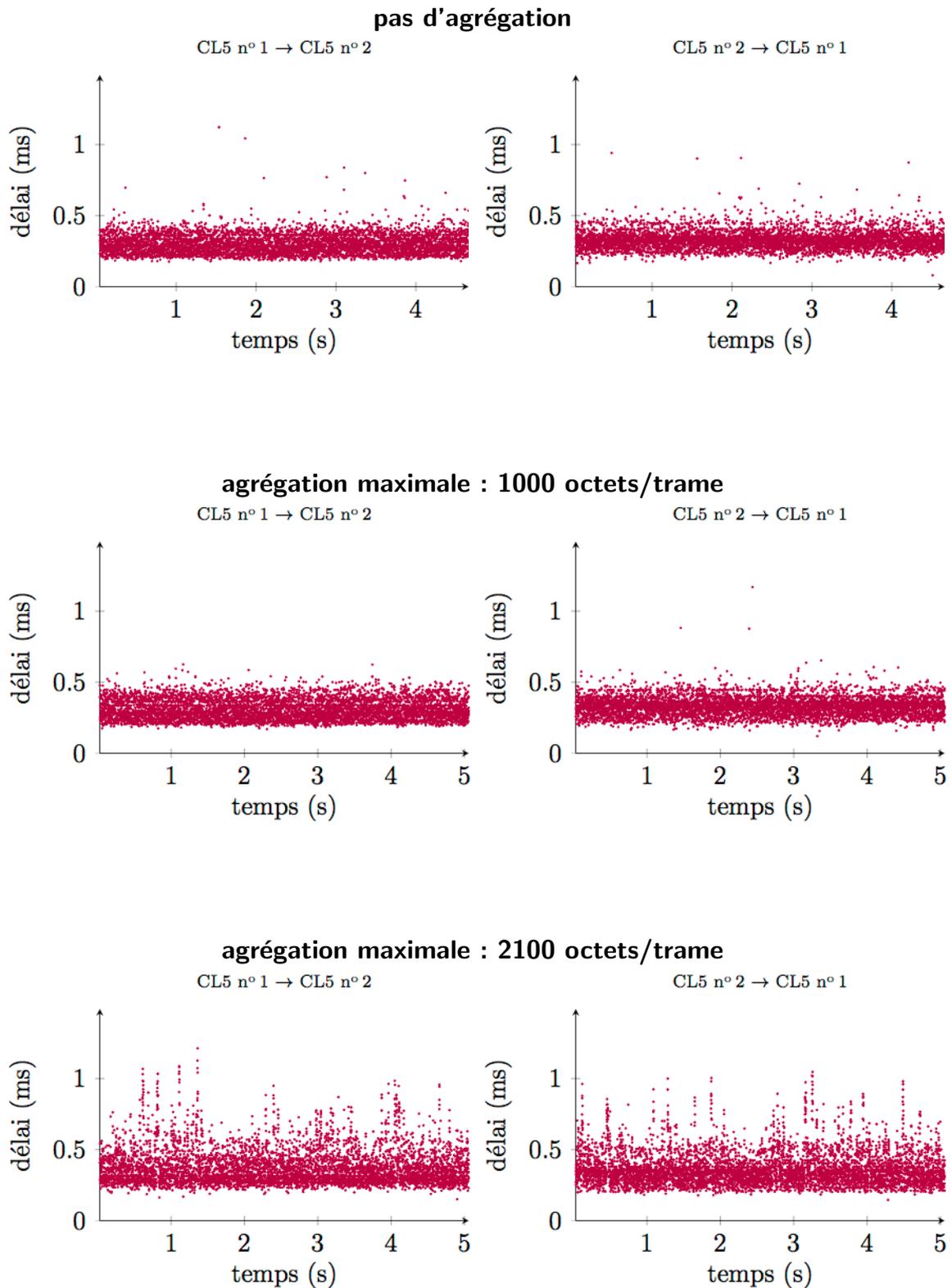
N.B. : Pour ces tests, les captures WIRESHARK et les enregistrements des erreurs n'ont pas été réalisés en de manière synchrone.

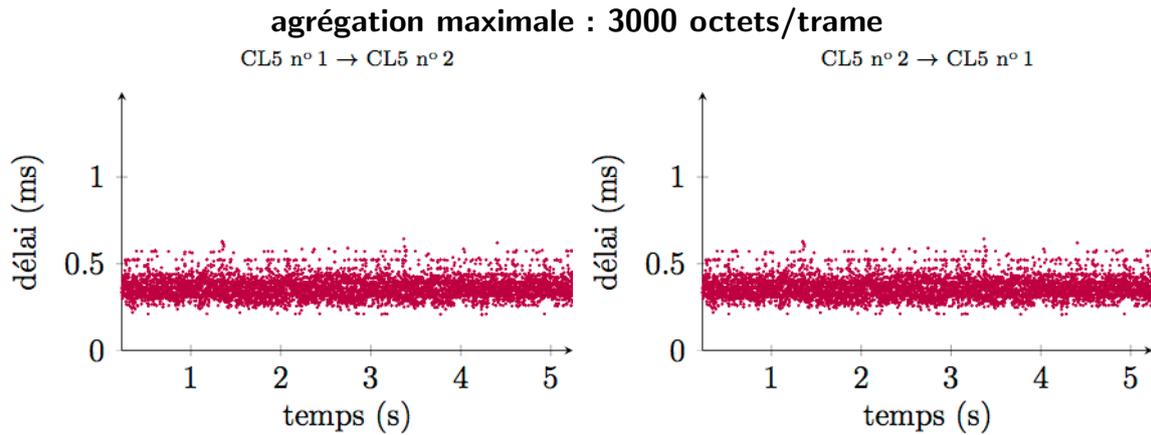
Analyse des résultats

On observe une légère diminution du nombre d'erreurs au cours du temps lorsqu'on augmente le paramètre d'agrégation, comme en témoigne les captures d'opposition de phase suivantes, où les erreurs de transmission sont représentées par des pics :



Comme nous l'avons déjà expérimenté précédemment, les captures effectuées par l'outil WIRESHARK permettent de tracer des graphiques de latence de parcours sur le lien sans fil.





La première observation que l'on peut faire par rapport aux résultats qu'on avait obtenus avec la liaison standard IEEE 802.11n, est que les temps de latence du parcours sans fil sont bien inférieurs avec la liaison *full-duplex*. On observe également que l'agrégation de trames permet de diminuer les retards accidentels de paquets.

Le graphique obtenu pour un paramètre d'agrégation maximal de 2100 octets reste cependant énigmatique. On observe des pics de latence survenant sur plusieurs paquets d'affilée. Cela peut s'expliquer par le fait qu'au moment où nous avons effectué cette capture, de nombreuses tentatives de transmission ont échoué au niveau radio. Les routeurs MIKROTIK possèdent un outil permettant de visualiser l'état de la liaison, et notamment le nombre de trames qu'il a fallu réémettre pour qu'elles arrivent à destination. Le compteur, appelé `tx retries lost` (tentative d'émission échouée) s'incrémente plus ou moins rapidement selon la qualité de la transmission. Dans notre cas, ce compteur n'est jamais resté à 0, ce qui devrait être le cas pour une liaison fiable.

Cet accident ne démentit pas le fait que l'agrégation de trame soit bénéfique dans notre utilisation, mais témoigne d'une possible instabilité des routeurs MIKROTIK.

Test sur la durée

Afin d'avoir une impression du nombre d'erreurs pouvant survenir sur une longue durée, nous avons réalisé une transmission de 16 canaux bidirectionnels dans la confi-

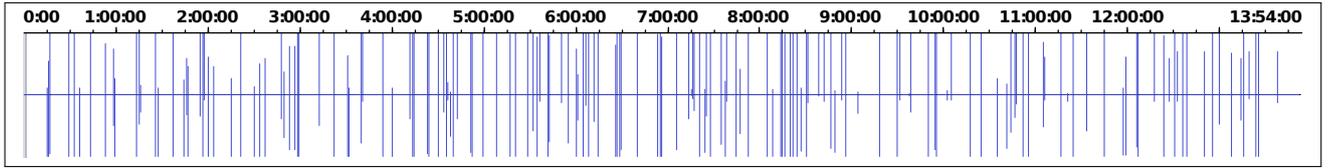


FIGURE 3.10 – Test de 14 heures – captures des erreurs

guration utilisant un paramètre d'agrégation de 2100 octets, pendant près de 14 heures.

Bien qu'on ait enregistré un grand nombre d'erreurs lors de ce test (179), il faut relativiser. Les résultats de ce tests sont plutôt prometteurs. D'abord, il convient de rappeler l'échelle de temps de la capture ci-dessus. Par souci de représentation, l'épaisseur des traits représentant les erreurs de transmissions est supérieure à la durée de celle-ci ... Pour quantifier le nombre d'erreur de transmission, nous avons « découpé » à l'aide d'un logiciel de montage, toutes les erreurs enregistrées en prenant des marges de début et de fin (voir fig. 3.11). En mettant bout à bout toutes les erreurs, on obtient un peu moins de 2 secondes d'erreurs, soit 0,004% de la durée totale de l'enregistrement, en ayant pris des marges de calcul.



FIGURE 3.11 – Erreur prélevée de l'enregistrement de 14 heures – durée de l'extrait = 19ms

L'autre point positif est que le système ne semble pas avoir subi de désynchronisation sur toute la durée du test.

3.2.6 À la recherche de la cause d'instabilité

La liaison *full-duplex* MIKROTIK ne semble pas très éloignée d'une solution fonctionnelle qui permettrait le transport d'un réseau audionumérique. En tout cas, pour remplir notre objectif de 16 canaux bidirectionnels, le débit effectif est suffisant et les mesures de latence ont donné des résultats prometteurs.

Malgré ces éléments positifs, on constate de nombreuses erreurs de transmission sur des longues durées dont nous avons souhaité connaître la cause.

Nous pouvons émettre au moins deux hypothèses quant à l'origine de l'instabilité constatée. Il pourrait d'abord s'agir d'un problème radio venant de la présence d'interférences. L'autre hypothèse est qu'une mauvaise gestion de l'agrégation de trame serait à l'origine d'un pic de retard, ce qui entraînerait la perte de données audio. Ces deux hypothèses ne sont pas nécessairement incompatibles, et on peut penser que le concours des deux est à l'origine des erreurs que l'on observe.

Afin de vérifier la première hypothèse, nous avons modifié les valeurs de MCS qui, trop élevées, pourraient rendre la liaison plus sensible aux interférences.

Des valeurs plus basses n'ont pas arrangé la transmission, ni même diminué l'occurrence de erreurs. Nous en déduisons donc que le problème proviendrait surtout d'une mauvaise gestion des données à l'intérieur des routeurs.

Pour visualiser les événements de latence qui seraient à l'origine des erreurs audio observées, nous avons réalisé une capture WIRESHARK, doublée d'un enregistrement audio des erreurs. De cette manière, nous serons peut-être en mesure de déterminer ce qui aura provoqué les erreurs enregistrées.

La difficulté de cette opération de double capture réside dans le traitement de données qu'il faudra faire *a posteriori*. Les captures WIRESHARK comprennent un nombre colossal de données ; à titre d'information, la capture de 42 secondes que nous avons effectuée dans le cadre de ce test pèse près de 473 Moctets, autant de données qu'il faudra ensuite traiter pour tracer le graphique de latence présenté.

Dans la mesure où les erreurs de transmission restent peu courantes, il faudra avoir la « chance » d'en observer une sur la durée de la capture ...

Présentation des résultats

Lorsque l'on effectue l'analyse des temps de délai mesurés aux alentours de l'erreur enregistrée (figure 3.12), il semblerait qu'on en ait trouvé la cause.

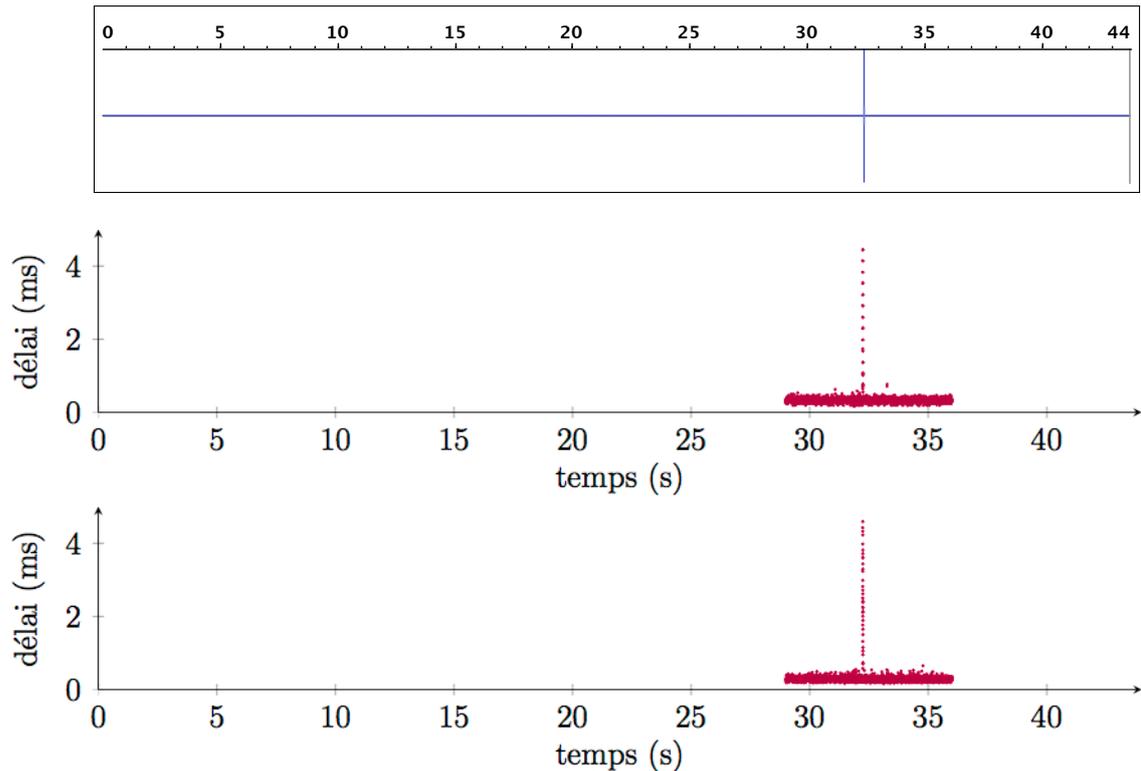


FIGURE 3.12 – De haut en bas : capture audio des erreurs de transmission par opposition de phase, délai mesuré dans le sens CL5 n° 1 → CL5 n° 2 et délai mesuré dans le sens CL5 n° 2 → CL5 n° 1

N.B. : les temps en indiqués abscisse des graphiques (figure 3.12) sont relatifs à la date de lancement de la capture en question. On ne peut donc pas interpréter de manière rigoureuse le décalage qu'il y a entre le pic de délai et l'erreur de transmission enregistrée. Les deux captures WIRESHARK quant à elles sont parfaitement synchrones.

En regardant de plus près les captures de délai (figure 3.13) on constate un pic de délai survenu à 32,238 secondes, dans les deux sens de transmission. Cet événement est quelque peu troublant puisqu'il intervient simultanément dans les deux sens. Cela signifie qu'il ne s'agit pas *a priori* d'un problème radio étant donné que chaque sens de transmission est assuré par un couple d'antennes/cartes réseau sans fil différent.

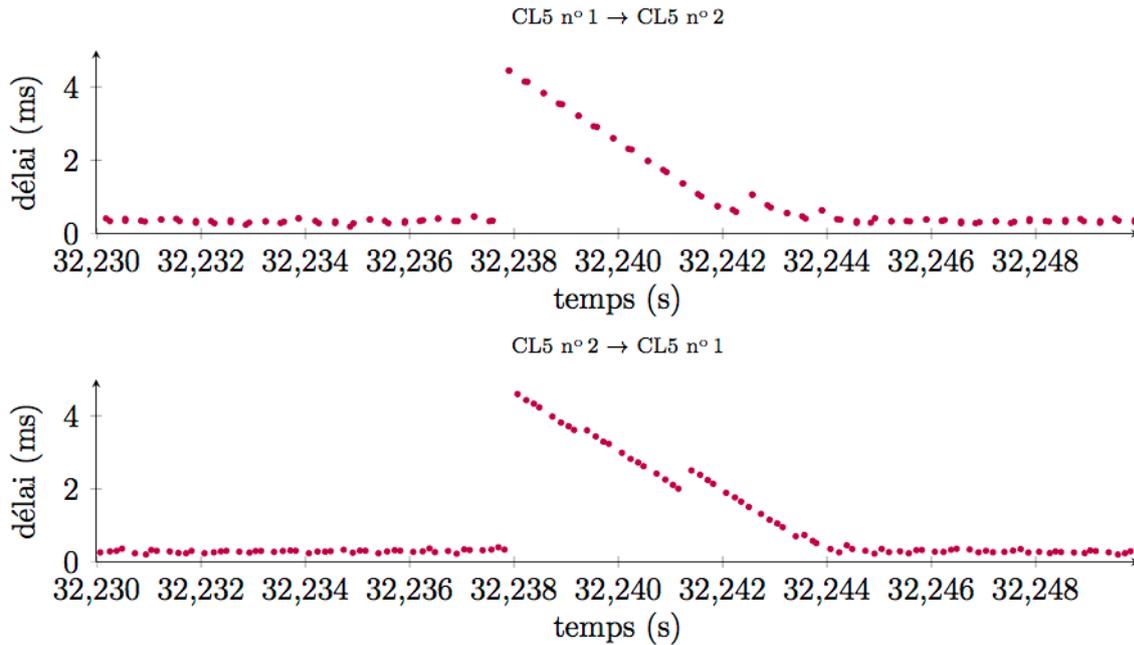


FIGURE 3.13 – Captures de délai autour du pic de délai observé

Nous pouvons en revanche incriminer les routeurs. De fait, une augmentation du temps de traitement des paquets pourrait expliquer les pics de latence observés.

Nos recherches nous ont conduit au fait que les routeurs étaient dotés d'un mode de gestion énergétique configuré par défaut sur `power-save` (sauvegarde d'énergie). Ce mode peut plonger le processeur du routeur dans des périodes de micro-sommeil aux moments de faible activité. Cela permet de consommer moins d'énergie lorsqu'on utilise une batterie comme source de courant, ce qui n'est pas notre cas.

Nous avons essayé à plusieurs reprises de le modifier, sans succès ; les routeurs refusent de changer de mode.

Bien que les résultats obtenus ne sont pas parfait, les tests que nous avons effectués révèlent une nette amélioration de la transmission lorsqu'on utilise une liaison *full duplex*. En utilisant deux canaux radio séparés pour l'émission et la réception, ainsi qu'en se passant des mécanismes de contrôle du support (CSMA), nous sommes parvenu à augmenter le débit effectif de la liaison, et diminuer la latence de parcours des paquets. Théoriquement, cela permet d'augmenter le nombre de canaux transmis, et permettrait de diminuer la mémoire tampon de réception sélectionnée dans Dante

Controller, et par conséquent la latence de lecture du système.

3.3 Transport sans fil de la synchronisation

Dans les tests que nous avons réalisés jusqu'ici, nous ne nous sommes pas attachés au problème du transport de la synchronisation en sans fil. Lors de nos tests, nous sommes d'ailleurs un peu éloigné du concept de base de la transmission sans fil, en tirant un câble coaxial entre les appareils audio pour nous défaire des problèmes liés à la synchronisation. Nous allons aborder dans cette partie les différents moyens sans fil permettant d'acheminer la synchronisation.

3.3.1 Transport sur le réseau sans fil

Dans le meilleur des cas, nous serions en mesure de conduire les données de synchronisations et les données audio par la même voie de transmission. Cette situation est permise dans le cas des réseaux câblés, grâce au protocole PTP. Il s'agit d'un protocole de synchronisation sur IP capable de partager le même réseau que celui utilisé pour l'échange de données audio.

L'application de ce type de synchronisation aux réseaux sans fil, question déjà abordée et en partie résolue par Léo ROSSI-ROTH dans son mémoire, nécessite soit l'implémentation du gPTP (*generalized PTP*) dans un routeur sans fil, soit l'ajout d'un dispositif d'horodatage au niveau de la couche PHY.

Quelle que soit la solution choisie pour le transport de la synchro sur le réseau sans fil, elle ne pourra pas se réaliser sans support des fabricants de matériel réseau.

3.3.2 Dispositif de syntonisation externe sans fil

Une autre solution que nous avons envisagé lors des tests que nous avons effectué, serait de simplement remplacer le câble utilisé pour la syntonisation externe par un dispositif sans fil.

Cela revient donc simplement à résoudre le problème du transport sans fil d'une *word clock* (signal analogique carré de fréquence fondamentale égale à 48kHz). On pourrait par exemple utiliser une modulation-démodulation très haute fréquence pour transmettre ce signal d'horloge comme on ferait pour transmettre la radio analogique. Cette solution présente l'avantage d'être très bon marché, mais n'est pas réellement compatible avec la diminution actuelle des bandes de fréquences utilisables sans autorisation. La tendance actuelle est plutôt de se partager les bandes ISM (selon les pays d'utilisation) en multiplexant les signaux pour améliorer le partage des fréquences.

Une recherche de matériel permettant le transport de la synchronisation en sans fil nous ont mené au matériel utilisé dans le cadre des tournages au cinéma. Le constructeur AMBIENT¹⁸ a récemment sorti le Lockit - ACL 204, nouveau dispositif de partage de synchronisation sur un plateau de tournage.



FIGURE 3.14 – AMBIENT Lockit - ACL 204

Les produits de la gamme Lockit utilisent un réseau propriétaire ACN (*Ambient Clockit Network*) fonctionnant dans la bande des 2,4GHz pour partager les informations de synchronisation.

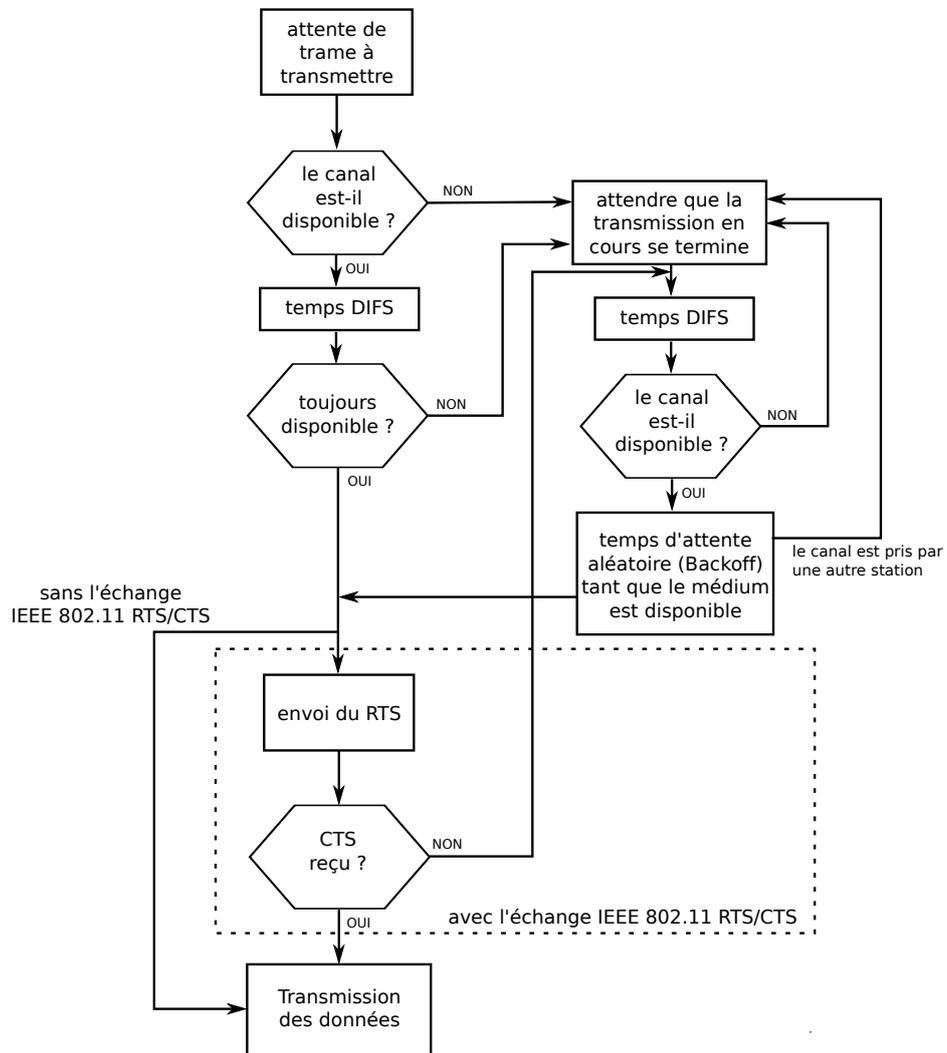
La précision de cette technologie le rend tout à fait utilisable dans notre cas. De fait AMBIENT annonce une précision de 10µs de décalage entre les horloges synchronisées sur le réseau. Cette précision (qui représente 1/2 échantillon à 48kHz) est tout à fait acceptable pour synchroniser deux appareils audio entre eux.

18. <http://www.ambient.de/>

L'existence de matériel de cette nature fonctionnant sur la bande des 2,4GHz, témoigne du fait qu'il est tout à fait envisageable d'un point de vue technologique de synchroniser des appareils sur un réseau sans fil. Étant donné que nous ne savons pas quel protocole (sans doute propriétaire) est utilisé pour la synchronisation sur le réseau ACN, nous ne pouvons pas faire de parallèle avec l'implémentation de la norme IEEE 1588 sur les réseaux IEEE 802.11.

L'utilisation d'un procédé de syntonisation externe est le seul moyen actuel permettant de réaliser une liaison 100% sans fil. Cela dit, il ne s'agit pas d'une solution viable étant donné que sa mise en service doit se faire en suivant la procédure décrite page 49, qui nécessite à un moment donné de connecter puis déconnecter les appareils par l'intermédiaire d'un câble.

Annexe A - Algorithme simplifié du CS-MA/CA



Annexe B - Mise en place d'une méthode de contrôle de la qualité d'une transmission audio

Lors des tests réalisés dans ce mémoire, nous devons en permanence surveiller la qualité de la transmission audio. Nous avons donc mis en place une méthode de vérification qui nous permettra d'avoir un retour immédiat en cas d'erreur.

De plus, celle-ci devra nous permettre de contrôler une liaison bidirectionnelle de plusieurs canaux audio.

La méthode la plus simple pour s'assurer qu'une transmission a lieu sans erreur consiste à comparer par opposition de phase le signal émis avec celui ayant traversé la liaison. Nous avons donc choisi d'effectuer un aller-retour sur un nombre donné de canaux audio, puis d'écouter en direct l'opposition de phase entre le signal à mesurer et le signal témoin.

Pour ce faire, autant de tranches « envoi » de la console n° 1 qu'il existe de canaux à tester sont assignées à la sortie du générateur de bruit interne à la console. Les sorties directes (*direct out*) sont envoyées sur des canaux du réseau audionumérique, puis récupérées dans des tranches de la console n° 2. Cette dernière les renvoie immédiatement en utilisant les sorties directes des tranches sur le réseau audionumérique. La console n° 1 reçoit ces canaux dans de nouvelles tranches « retour ». Après avoir ajusté le délai sur les tranches « envoi » (délai *post* sorties directes) et inversé la phase des tranches « retour », chaque tranche « retour » devrait annuler une tranche « envoi » dans le bus *master*.

Si une erreur intervient sur l'un des canaux audio du réseau audionumérique, cela se manifestera par un bruit sur le bus *master* provoqué par une opposition de phase imparfaite.

Cette méthode a l'avantage de pouvoir contrôler la transmission sur un grand nombre de canaux, de manière simple. En cas d'erreur, il ne sera en revanche pas possible de connaître le canal sur lequel elle a eu lieu.

La CL5 intègre un enregistreur stéréo sur support USB. Cela permettra au besoin

d'enregistrer un historique des erreurs de transmission.

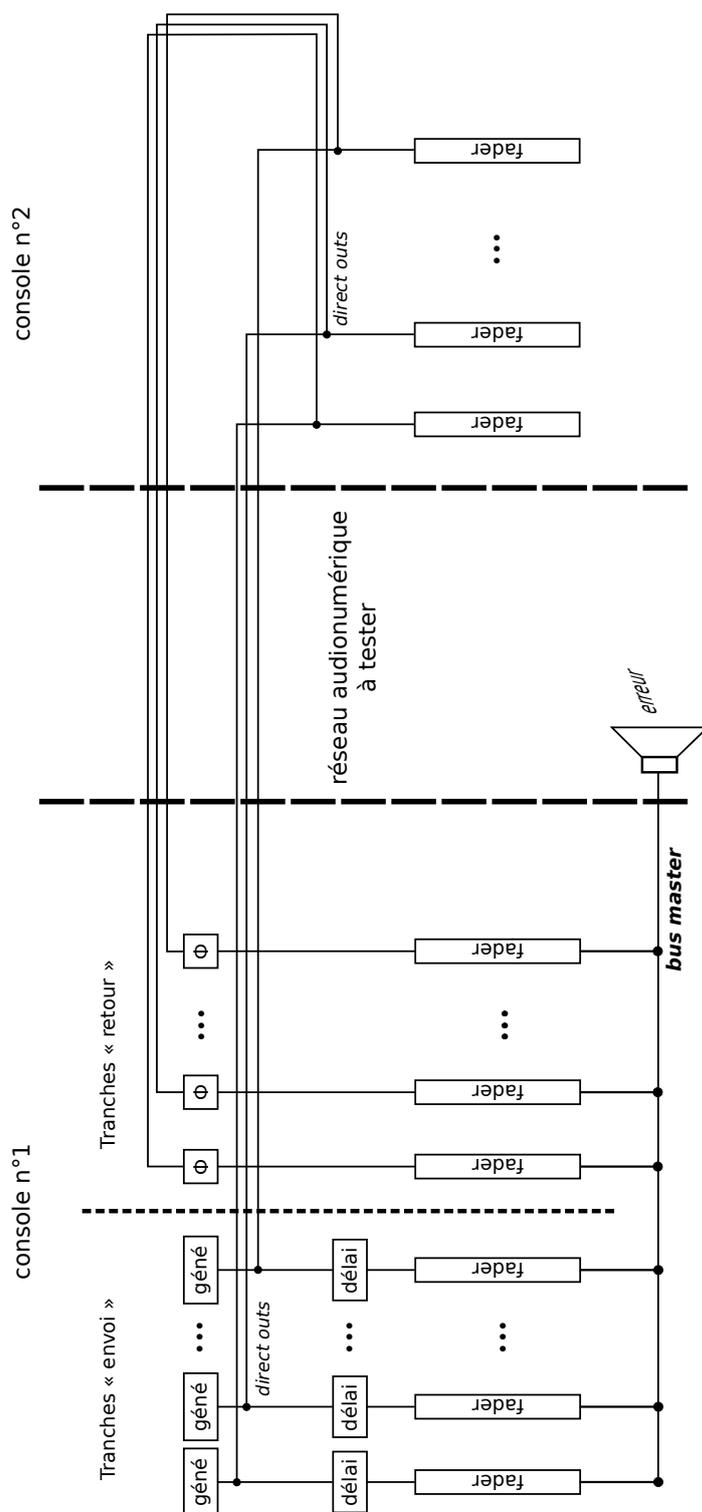


FIGURE B.1 – Synoptique du dispositif de vérification utilisé pour les tests

Annexe C - Fiche technique des antennes



SECTOR MIMO HV 17dBi/5GHz

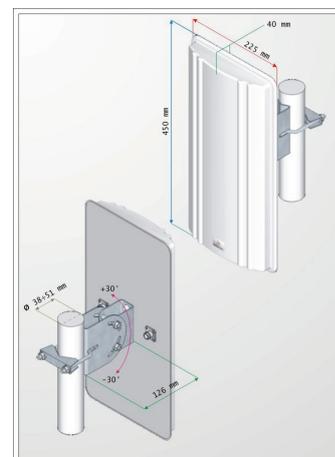
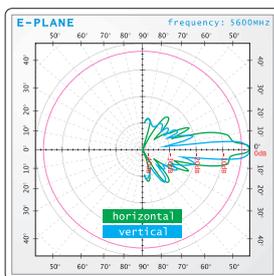
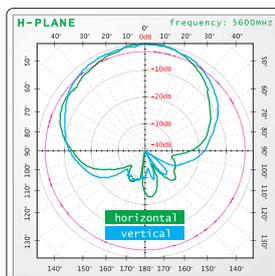
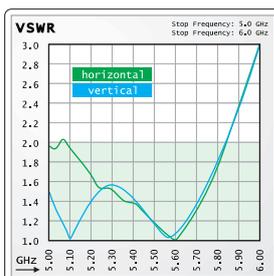
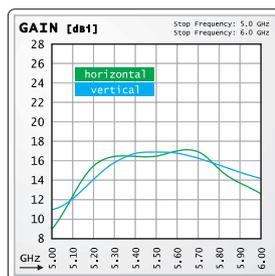
New sector antennas for base stations in the MIMO technology developed for use with radio equipment operating in the 802.11n standard that use MIMO technology.

Key Features

- 2 integrated antennas: allow to use 802.11n to increase the theoretical bandwidth up to 300Mbps
- patch matrix lacquered: reliable performance in all weather conditions

Technical Data

product code	INT-SEC-17/5X-HV
frequency	5150-5850MHz
impedance	500Hm
VSWR (max)	2.0
polarization	horizontal and vertical
gain	17dBi
front to back ratio	> 26dBi
port separation	> 44dBi
lighting protection	shorted for DC
dimensions (without mounting)	450x225x40mm
weight (with mounting)	1.5kg
connector	N female
mounting diameter	38...51mm



Annexe D - Capture du spectre effectuée avec l'outil **AirView** : local de De Préférence

Les NANOBRIDGE M5 de UBIQUITI intègrent l'outil AirView qui permet de réaliser une capture du spectre. Cela permet de visualiser l'encombrement spectral des autres émetteurs de la zone de test.

La première partie de la fenêtre AirView est un spectrogramme sur la gamme de fréquence analysée.

Les deux autres graphiques représentent la puissance mesurées en fonction de la fréquence.

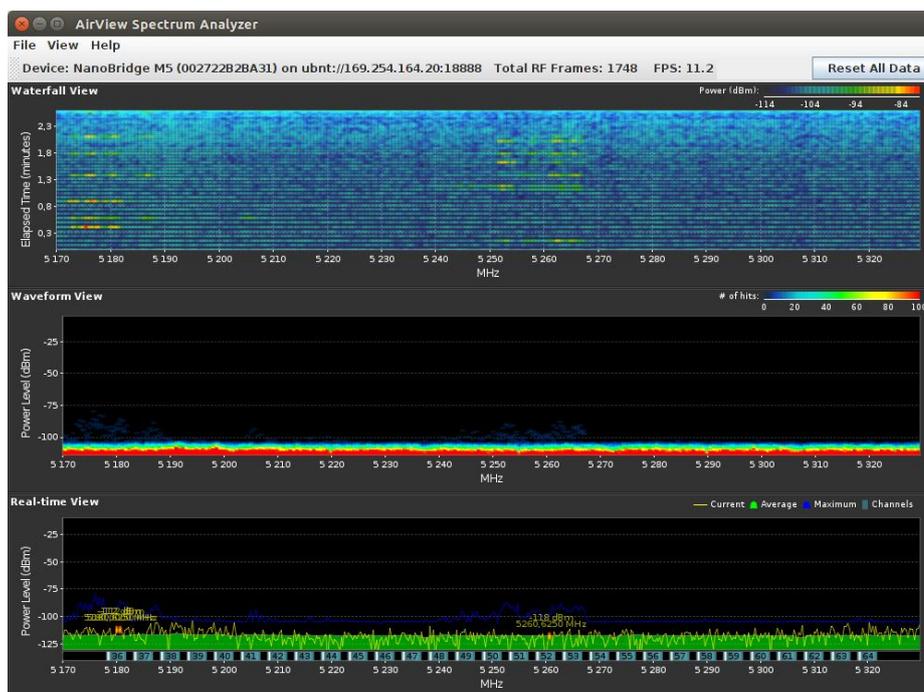


FIGURE D.1 – 5170MHz - 5330MHz

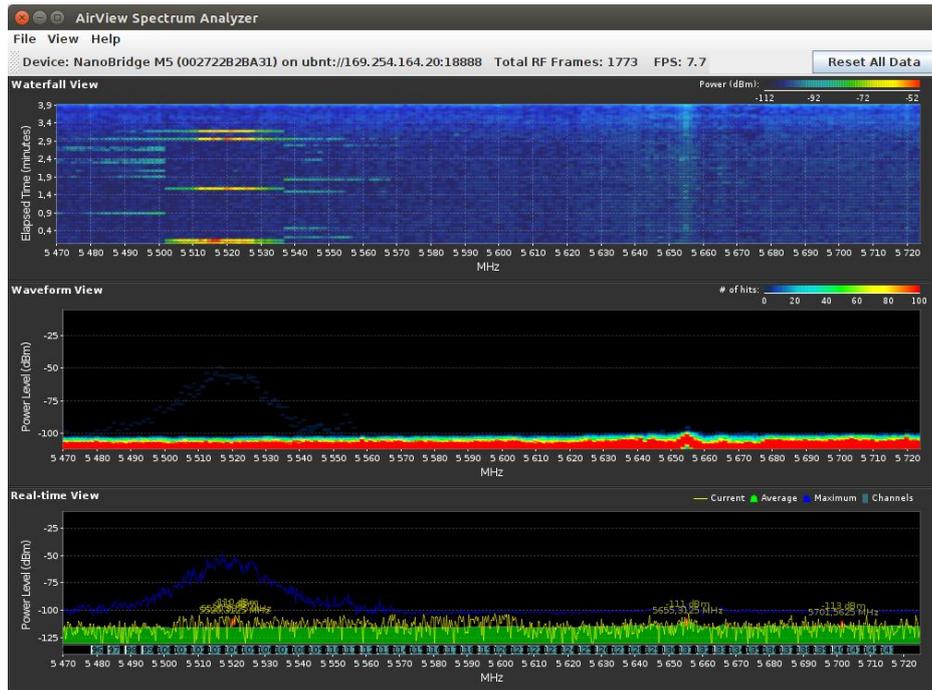


FIGURE D.2 – 5470MHz - 5725MHz

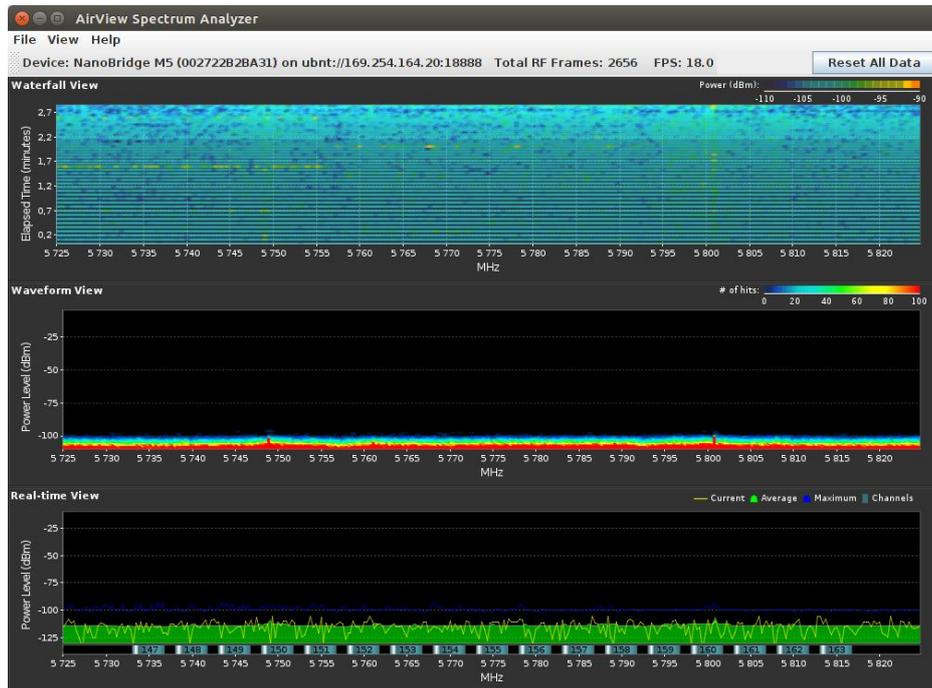


FIGURE D.3 – 5725MHz - 5830MHz

Conclusion

Dans cette étude, notre objectif était de caractériser de manière précise une liaison sans fil qui permettrait le transport d'un réseau audionumérique. Plus précisément, d'évaluer dans quelle mesure une liaison « Wi-Fi » (standard IEEE 802.11) permet d'acheminer les données audio d'un réseau audionumérique.

Les travaux de Léo Rossi-Roth qui portaient, eux, sur le transport des données de synchronisation, avaient révélé que les fluctuations de temps de parcours à travers une liaison sans fil ne permettaient pas de synchroniser les appareils audio entre eux à travers un réseau sans fil standard. Les appareils ne communiquant pas d'audio s'ils ne sont pas synchronisés, nous avons donc opté, dans le cadre de nos tests, pour une synchronisation externe *filaire*, afin de pouvoir étudier le transport des données audio à proprement parler.

Les résultats que nous avons obtenus sont prometteurs, puisque nous sommes parvenus à faire transiter sans erreur quatre canaux aller et quatre canaux retour avec une latence de cinq millisecondes sur une liaison standard IEEE 802.11. En utilisant une liaison *full duplex* (solution optimisée pour les échanges bidirectionnels), nous avons pu échanger seize canaux dans chaque sens avec quelques erreurs.

Les analyses ont montré que les principales erreurs étaient provoquées par des retards trop importants de paquets contenant des échantillons audio, au moment de leur transmission sans fil. Une manière de résoudre le problème serait d'utiliser du matériel sans fil plus performant. Une autre solution serait d'augmenter la valeur du tampon de réception des appareils afin qu'ils acceptent des retards de paquet plus importants.

Cette deuxième solution trouverait sa place dans le cas d'applications qui tolèrent

de plus grandes latences. Il convient de rappeler que bien que nous avons étudié le cas d'une utilisation *live* de la solution sans fil, l'utilisation des réseaux audionumériques sans fil ne se limite pas à ce cas et on peut par exemple s'autoriser une latence beaucoup plus importante pour transmettre les canaux audio d'un concert *live* à un car régie, cas habituel des retransmissions en direct.

Le problème de la synchronisation des appareils audio à travers un réseau sans fil, qui rend actuellement impossible le transport sans fil des réseaux audionumériques reste entier. Nous avons évoqué une alternative à la synchronisation PTP qui consisterait à utiliser une *word clock* externe sans fil. Cette solution n'est cependant pas applicable telle quelle à long terme, car pour la mettre en œuvre, nous exploitons pour le moment une faille de fonctionnement des réseaux audionumériques.

Rappelons aussi que la bonne synchronisation des périphériques à travers un réseau sans fil peut se faire avec l'utilisation de protocoles prévus à cet effet, comme le gPTP décrit dans la norme IEEE 802.1AS. Cela nécessite néanmoins un support matériel.

En somme comme nous l'avons étudié pour le transport des données audio ou des données de synchronisation seule l'utilisation de matériel réseau dédié, c'est-à-dire garantissant une faible latence de transmission (très inférieure à la latence de lecture souhaitée pour le système) et implémentant un protocole de synchronisation permettra d'acheminer correctement un réseau audionumérique.

Même si ce produit n'existe pas à l'heure actuelle, il semble n'y avoir aucun obstacle majeur à sa réalisation. On peut alors raisonnablement penser qu'une entreprise souhaitant le développer y parvienne rapidement.

En outre, et contrairement à ce que décrivait William Leveugle il y a trois ans, nous avons pu lors de l'élaboration de ce mémoire rencontrer et échanger avec de nombreux professionnels conscients du potentiel et des possibilités qu'offrirait un réseau audionumérique sans fil ; il y a déjà une attente et un marché pour cet éventuel produit.

Bibliographie

- [1] *IEEE Std 1733™-2011 - IEEE Standard for Layer 3 Transport Protocol for Time-Sensitive Applications in Local Area Networks*, Institute of Electrical et Electronics Engineers, 2011.
- [2] *IEEE Std 802.11™-2012 - IEEE Standard for Information technology–Telecommunications and information exchange between systems Local and metropolitan area networks–Specific requirements Part 11 : Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, Institute of Electrical et Electronics Engineers, 2012.
- [3] *IEEE Std 802.11aa™-2012 - IEEE Standard for Information technology–Telecommunications and information exchange between systems Local and metropolitan area networks–Specific requirements Part 11 : Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 2 : MAC Enhancements for Robust Audio Video Streaming*, Institute of Electrical et Electronics Engineers, 2012.
- [4] *IEEE Std 802.11ac™-2013 - IEEE Standard for Information technology–Telecommunications and information exchange between systems Local and metropolitan area networks– Specific requirements–Part 11 : Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications–Amendment 4 : Enhancements for Very High Throughput for Operation in Bands below 6 GHz*, Institute of Electrical et Electronics Engineers, 2013.
- [5] *IEEE Std 802.1AS™-2011 - IEEE Standard for Local and metropolitan area networks - Timing and Synchronization for Time-Sensitive Applications in Bridged Local Area Networks*, Institute of Electrical et Electronics Engineers, 2011.
- [6] *IEEE Std 802.1BA™-2011 - IEEE Standard for Local and metropolitan area networks — Audio Video Bridging (AVB) Systems*, Institute of Electrical et Electronics Engineers, 2011.
- [7] *AES67-2013 : AES standard for audio applications of networks - High-performance streaming audio-over-IP interoperability*, Audio Engineering Society, Inc., 2008.

- [8] T. DE BACKER, *Use Linux Traffic Control as impairment node in a test environment*, blog, 2014. adresse : <https://www.excentis.com/blog/use-linux-traffic-control-impairment-node-test-environment-part-1>.
- [9] *Dante Controller User Guide*, version 1.6, AUD-MAN-DanteController-3.5.x-v1.6.pdf, Audinate, 2014.
- [10] M. GAST, *802.11n : A Survival Guide*. O'Reilly Media, 2012.
- [11] —, *802.11ac : A Survival Guide*. O'Reilly Media, 2013.
- [12] D. HANES et G. SALGUEIRO, *Chapter 4 : Passthrough*, 2008. adresse : <http://www.networkworld.com/article/2274805/lan-wan/chapter-4--passthrough.html>.
- [13] B. HUBERT, T. GRAF, M. GREGORY, R. van MOOK, M. van OOSTERHOUT, P. B. SCHROEDER, J. SPAANS et P. LARROY, *Linux Advanced Routing & Traffic Control HOWTO*, 2012. adresse : <http://lartc.org/howto/>.
- [14] W. LEVEUGLE, “Des possibilités d’évolution vers le sans-fil des réseaux audio-numériques pour la sonorisation - Cas des Wireless Lan”, mém.de maîtr., École Nationale Supérieure Louis-Lumière, 2012.
- [15] B. MILLER, *A Voice Of Reason On Voice Over WiFi*, blog, 2015. adresse : <http://www.sniffwifi.com/2015/02/the-voice-qos-category-of-no-reason-to.html>.
- [16] *IEEE Std 1588TM-2008 - IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems*, Institute of Electrical et Electronics Enginners, 2008.
- [17] *Technology Overview*, ALC NetworX, 2013. adresse : <http://ravenna.alcnetworx.com/technology/technology-overview.html>.
- [18] L. ROSSI-ROTH, “Étude des possibilités d’utilisation d’un réseau sans-fil pour le transport d’un réseau audionumérique”, mém.de maîtr., École Nationale Supérieure Louis-Lumière, 2014.
- [19] J. TOURRILHES, *The Linux Wireless LAN Howto*, “5 The MAC level (link layer)”, 2000. adresse : http://wiki.linuxwall.info/doku.php/fr:ressources:dossiers:networking:traffic_control#traffic_control_les_bases.
- [20] J. VEHENT, *Voyage au centre du noyau : Traffic Control, la QoS*, wiki, 2011. adresse : http://wiki.linuxwall.info/doku.php/fr:ressources:dossiers:networking:traffic_control#traffic_control_les_bases.

Table des figures

1.1	Synoptique d'une installation SONOS	4
1.2	Évolutions du groupe de travail 802.11 – source : [11]	5
1.3	Le modèle OSI – source [14]	5
1.4	Répartition des canaux de la bande des 5GHz dans le monde – source [11]	8
1.5	Canal primaire / canal secondaire – source [11]	8
1.6	Répartition des <i>pilots</i> sur la bande – source [11]	9
1.7	Trame A-MPDU pouvant contenir une ou plusieurs trames MPDU – source [11]	11
1.8	Diagrammes de constellation : (a) 64-QAM, (b) 256-QAM – source [11]	12
1.9	Exemple d'interférence pour deux antennes émettrices	14
1.10	Principe du MU-MIMO – source [11]	15
2.1	(a) topologie en étoile, (b) topologie en <i>daisy chain</i>	21
2.2	Liaison réalisée lors de la soutenance de Léo ROSSI-ROTH	23
2.3	Synoptique du test de synchronisation 1	24
2.4	Synoptique du test de synchronisation 2	26
2.5	Schéma de fonctionnement de synchronisation de deux module PTP . .	29
2.6	(a) situation standard, (b) synchronisation externe avec horloges audio non-identiques, (c) synchronisation externe avec horloge audio identiques	30
2.7	Encapsulation d'un paquet RTP – source [12]	31
2.8	Sélection de la latence dans Dante Controller	36
2.9	Chemin parcouru par un paquet réseau dans le noyau GNU/Linux – source [8]	37

2.10	Exemple de <i>queuing discipline</i>	38
2.11	Configuration de netem : (a) simple, (b) avec classe	39
2.12	Synoptique d'utilisation de la NETEM BOX	41
2.13	Schéma de fonctionnement de CARTON	42
2.14	Retard mesuré à travers CARTON, en fonction du temps	43
2.15	Capture de délai effectuée à travers CARTON – avec netem 1 ± 1 ms de latence	43
2.16	Délai mesuré par rapport au paquet contenant les échantillons audio antérieurs – avec netem 1 ± 1 ms de latence	44
2.17	Synoptique du test de syntonisation externe	45
2.18	Configuration de netem pour le test de synchronisation externe	46
2.19	Synoptique du test de simulation	50
2.20	Synoptique de la liaison sans fil avec syntonisation externe	52
3.1	Déroulé simplifié des échanges entre deux stations	58
3.2	Temps de transmission d'un paquet	60
3.3	Format de trame de la couche PHY pour les normes 802.11n et 802.11ac – <i>source [11]</i>	60
3.4	Format de la MPDU (ou A-MDPU pour des trames agrégées) - schéma simplifié	61
3.5	Évolution du débit efficace en fonction du nombre de trames agrégées, pour des paquets de 424 octets	67
3.6	Diagrammes de directivité des antennes – voir annexe C	75
3.7	Courbe de gain des antennes – voir annexe C	76
3.8	Hypothèse sur les mécanismes du Nstreme dual	77
3.9	Synoptique de la liaison MIKROTIK avec syntonisation externe	78
3.10	Test de 14 heures – captures des erreurs	83
3.11	Erreur prélevée de l'enregistrement de 14 heures – durée de l'extrait = 19ms	83
3.12	De haut en bas : capture audio des erreurs de transmission par opposition de phase, délai mesuré dans le sens CL5 n° 1 → CL5 n° 2 et délai mesuré dans le sens CL5 n° 2 → CL5 n° 1	85

3.13	Captures de délai autour du pic de délai observé	86
3.14	AMBIENT Lockit - ACL 204	88
B.1	Synoptique du dispositif de vérification utilisé pour les tests	92
D.1	5170MHz - 5330MHz	94
D.2	5470MHz - 5725MHz	95
D.3	5725MHz - 5830MHz	95