

UNIVERSITÉ LILLE1 - SCIENCES ET TECHNOLOGIES

---

Numéro d'ordre : 40459

# Couche logicielle bas niveau efficace pour interface radio à faisceaux agiles

---

*Thèse de doctorat (spécialité Informatique) présentée le 14 décembre 2010*

par THOMAS SOETE

## Composition du jury

*Rapporteurs :* Andrzej DUDA, Professeur, INP-ENSIMAG, Grenoble  
Thomas NOEL, Professeur, Université de Strasbourg

*Examineurs :* Claude CHAUDET, Maître de conférences, Telecom ParisTech  
Gilles GRIMAUD, Professeur, Université Lille1 - Sciences et Technologies  
Nathalie ROLLAND, Professeur, Université Lille1 - Sciences et Technologies

*Directeur :* David SIMPLOT-RYL, Professeur, Université Lille1 - Sciences et Technologies



« Il suffit [...] d'avoir gardé soi-même  
la conscience de vivre dans un monde  
d'énigmes, auquel c'est en énigmes  
aussi qu'il convient le mieux de ré-  
pondre »

**Henri Michaux**



# Remerciements

---

Reprendre le clavier longtemps (voire un peu trop ?) après la rédaction et la soutenance est difficile. Néanmoins je souhaite remercier :

Audrey pour m'avoir soutenu (voire supporté) pendant ces longs et douloureux mois de rédaction. Elle est maintenant prête pour écrire sa propre thèse sur les protocoles de communication sans fils à force de me relire et de me corriger.

Ma fille Léa pour les quelques nuits blanches avant la soutenance.

David SIMPLOT-RYL pour m'avoir accueilli dans l'équipe POPS et m'avoir encadré durant mes années de thèse.

Simon DUQUENNOY, Michaël HAUSPIE et Gilles GRIMAUD, pour nos longues et nombreuses discussions autour des microprocesseurs, du C, de l'assembleur ou d'autres geekeries.

Arnaud FONTAINE pour son aide lors des pires galères avec notre grand ami  $\text{\LaTeX}$ .

Les membres de mon jury pour l'attention qu'ils ont prêtée à mon travail. Merci donc à Andrzej DUDA et Thomas NOEL d'avoir rapporté mon manuscrit. Merci à Claude CHAUDET et Nathalie ROLLAND d'avoir examiné ma thèse ainsi qu'à Gilles GRIMAUD d'avoir présidé le jury.

Les membres de l'équipe POPS et le personnel de l'IUT-A pour la bonne ambiance de travail, les repas, les pauses café et les sorties.

Et tous les autres que j'ai oubliés :-)



# Table des matières

---

<b>1</b>	<b>Introduction</b>	<b>9</b>
1.1	Contexte . . . . .	10
1.2	Antennes directionnelles . . . . .	10
1.3	Bande de fréquence 60 GHz . . . . .	12
1.4	Objectifs de la thèse . . . . .	12
1.5	Organisation du document . . . . .	13
<b>2</b>	<b>État de l'art</b>	<b>15</b>
2.1	Les réseaux sans fil . . . . .	16
2.1.1	Réseaux sans fil à stations de base . . . . .	16
2.1.2	Réseaux maillés . . . . .	16
2.1.3	Réseaux pair-à-pair . . . . .	17
2.2	Pile de protocoles . . . . .	17
2.2.1	Le modèle OSI . . . . .	18
2.2.1.1	Description des couches du modèle OSI . . . . .	19
2.2.2	Couche liaison de données . . . . .	19
2.2.3	Contrôle d'accès au support ( <i>Media Access Control - MAC</i> ) . . . . .	20
2.2.4	Découverte de voisinage . . . . .	21
2.3	Couches MAC pour antennes omnidirectionnelles . . . . .	21
2.3.1	Couches MAC basées sur le canal . . . . .	21
2.3.1.1	FDMA . . . . .	21
2.3.1.2	TDMA . . . . .	22
2.3.1.3	CDMA . . . . .	22
2.3.1.4	PDMA . . . . .	22
2.3.2	Couches MAC basées sur les paquets . . . . .	23
2.3.2.1	ALOHA . . . . .	23
2.3.2.2	Slotted ALOHA . . . . .	23
2.3.2.3	CSMA . . . . .	24
2.3.2.4	CSMA/CA . . . . .	24
2.3.2.5	IEEE 802.11 RTS/CTS . . . . .	25
2.3.2.6	MACAW . . . . .	26
2.4	Couches MAC pour antennes sectorielles . . . . .	26
2.4.1	SDMA . . . . .	26
2.4.2	DMAC . . . . .	27
2.4.3	MMAC . . . . .	27
2.4.4	Circular RTS MAC . . . . .	28
2.4.5	CRCM . . . . .	28
2.4.6	MDA . . . . .	29
2.4.7	DMAC/DA . . . . .	29

2.4.8	DMAC/DA with NPN . . . . .	30
2.4.9	SWAMP . . . . .	31
2.5	Conclusion . . . . .	32
<b>3</b>	<b>Architecture de la pile protocolaire pour antennes à faisceaux agiles</b>	<b>33</b>
3.1	Introduction . . . . .	34
3.2	Couche physique . . . . .	34
3.2.1	Antennes multi-secteur avec temps de commutation . . . . .	34
3.2.2	Canaux radio . . . . .	35
3.3	Découverte du voisinage . . . . .	36
3.3.1	Protocoles existants . . . . .	36
3.3.2	Proposition de protocole . . . . .	36
3.4	Proposition de pile protocolaire efficace . . . . .	38
3.4.1	Architecture globale . . . . .	38
3.4.2	Couche liaison de données . . . . .	39
3.4.2.1	Découverte de voisinage - ( <i>neighborhood discovery</i> ) . . . . .	39
3.4.2.2	Bloc de répartition - ( <i>dispatcher</i> ) . . . . .	41
3.4.2.3	Bloc de contrôle - ( <i>controler</i> ) . . . . .	41
3.4.2.4	Bloc d'ordonnancement - ( <i>scheduler</i> ) . . . . .	41
3.5	Ordonnancement . . . . .	42
3.5.1	Choix du secteur . . . . .	42
3.5.2	Événements extérieurs . . . . .	43
3.6	Conclusion . . . . .	43
<b>4</b>	<b>Couche MAC et agrégation de paquets</b>	<b>45</b>
4.1	Introduction . . . . .	46
4.2	Détails de la couche MAC . . . . .	46
4.2.1	Automate de la couche MAC . . . . .	46
4.2.2	Trames de contrôle . . . . .	47
4.2.2.1	Trame de base . . . . .	47
4.2.2.2	RTS . . . . .	47
4.2.2.3	CTS . . . . .	48
4.2.2.4	ACK . . . . .	48
4.2.2.5	Hello . . . . .	48
4.2.3	Trames de données . . . . .	49
4.3	Canal bas débit . . . . .	49
4.3.1	Trames de contrôle . . . . .	49
4.3.2	Envoi de données . . . . .	50
4.4	Canal haut débit . . . . .	50
4.4.1	Synchronisation . . . . .	50
4.4.2	Taille des trames . . . . .	51
4.4.3	Envoi d'un paquet en utilisant le canal « haut débit » . . . . .	53
4.4.3.1	Envoi classique . . . . .	53
4.4.3.2	Gestion des échecs . . . . .	54
4.4.3.3	Amélioration . . . . .	54
4.5	Algorithme de la couche MAC . . . . .	55
4.5.1	Mode « bas débit » . . . . .	55
4.5.2	Mode « haut débit » . . . . .	57
4.6	Agrégation multi-utilisateurs . . . . .	58
4.7	Conclusion . . . . .	60

---

<b>5</b>	<b>Implémentation</b>	<b>61</b>
5.1	Introduction . . . . .	62
5.2	Description de la plateforme d'expérimentation . . . . .	62
5.2.1	Plateforme Excalibur . . . . .	62
5.2.2	Couche physique . . . . .	62
5.2.3	Couche liaison de données . . . . .	63
5.3	Interface <i>Hard-Soft</i> . . . . .	63
5.4	Expérimentations réalisées . . . . .	64
5.4.1	Description du prototype . . . . .	64
5.4.2	Expérimentations . . . . .	65
5.5	Conclusion . . . . .	65
<b>6</b>	<b>Conclusion et perspectives</b>	<b>67</b>
6.1	Synthèse . . . . .	68
6.2	Perspectives . . . . .	68
	<b>Liste des figures</b>	<b>71</b>
	<b>Liste des sigles et acronymes</b>	<b>73</b>
	<b>Bibliographie</b>	<b>77</b>
<b>A</b>	<b>Algorithmes de la couche MAC</b>	<b>81</b>
<b>B</b>	<b>Structure de la puce <i>Excalibur</i></b>	<b>85</b>



## Premier chapitre

---

# Introduction

### Sommaire

---

1.1	Contexte . . . . .	10
1.2	Antennes directionnelles . . . . .	10
1.3	Bande de fréquence 60 GHz . . . . .	12
1.4	Objectifs de la thèse . . . . .	12
1.5	Organisation du document . . . . .	13

---

Nous proposons dans ce document une solution de couche logicielle bas niveau pour interface radio à faisceaux agiles. Dans ce premier chapitre, nous exposons le contexte des communications sans fil domestiques ainsi que l'intérêt des antennes directionnelles. Nous introduisons ensuite la technologie à 60 GHz utilisée dans le cadre de ces travaux. Nous finirons ce chapitre en détaillant l'organisation du document.

## 1.1 Contexte

Depuis plus de quarante ans, nous raccordons nos ordinateurs en réseau pour pouvoir échanger des données. Arpanet<sup>1</sup> a été le premier réseau informatique du monde avec le premier échange le 29 octobre 1969. D'abord réseau militaire, ce dernier s'est vu scindé en deux en 1980 afin de créer un réseau civil universitaire (CSNet<sup>2</sup> puis NSFNet<sup>3</sup>). À une plus petite échelle, des réseaux dits locaux se sont déployés au sein des universités, des entreprises puis plus récemment chez les particuliers : Ethernet [MB83], par exemple, a été expérimenté en 1976 au PARC<sup>4</sup> avec un débit de 3Mbit/s.

Depuis, les besoins en débit n'ont cessé d'augmenter avec les années. Là où l'on se contentait, il y a une petite dizaine d'années, d'une connexion Internet à 14.4 kb/s ou d'un réseau local à 10 Mb/s, des débits comme 100 Mb/s pour l'accès à Internet ou 1 Gb/s pour le réseau local sont devenus courants.

Durant la dernière décennie, une nouvelle forme de réseau local est apparue avec les réseaux sans fil. Balbutiants avec la technologie Bluetooth, les réseaux locaux sans fil ont commencé à se déployer avec le Wi-Fi en 1999. En France, l'Arcep<sup>5</sup> a libéré la fréquence utilisée par le Wi-Fi (2.4 GHz) en 2003 autorisant le déploiement de réseaux sans fil domestiques sur le territoire français. D'abord intimiste, l'arrivée des « box » des fournisseurs d'accès à Internet français a accéléré la pénétration de la technologie sans fil chez les particuliers ainsi que la création de réseaux domestiques. Avant l'apparition de ces « box », beaucoup d'utilisateurs avaient un ordinateur connecté à Internet avec un simple modem. Celles-ci ont introduit un système de réseau local privé chez les particuliers et un partage de la connexion Internet à tous les ordinateurs.

Enfin, l'arrivée des ordinateurs portables à faible coût a démocratisé le principe de mobilité : connexion au réseau et à Internet, rapide et en tout lieu, disparition presque systématique des câbles réseaux. Malheureusement, malgré des débits théoriques de 300 Mb/s ( $\sim$  100 Mb/s réels), la technologie Wi-Fi n'est actuellement pas adaptée aux communications à très haut débit devenant courantes avec les lecteurs HD<sup>6</sup> domestiques tels que les platines Blu-Ray et les écrans plats HD, ceux-ci nécessitant un transport de flux vidéo haute définition non compressé.

## 1.2 Antennes directionnelles

Encore actuellement, beaucoup de systèmes de communication sans fil (tant grands publics que professionnels) utilisent des antennes omnidirectionnelles. Celles-ci ont pour inconvénient d'utiliser beaucoup d'énergie pour atteindre un destinataire donné et polluent l'espace autour de l'émetteur empêchant d'autres communications et diminuant de ce fait l'efficacité du système. La figure 1.1a montre que si un nœud A veut communiquer avec un nœud D, leur communication occupe une sphère autour de chaque intervenant, empêchant B et C de communiquer.

---

1. Arpanet : *Advanced Research Projects Agency Network*

2. CSNet : *Computer Science Network*

3. NSFNet : *The National Science Foundation Network*

4. PARC : *Palo Alto Research Center*

5. Arcep : Autorité de Régulation des Communications Électroniques et des Postes

6. HD : Haute définition

Avec des antennes dites directionnelles ou sectorielles (figure 1.1b), le volume occupé par une communication représente maintenant un cône et il est ainsi possible d'effectuer plusieurs communications dans un même endroit sans se gêner mutuellement tant que les communications ne se produisent pas dans la même direction.

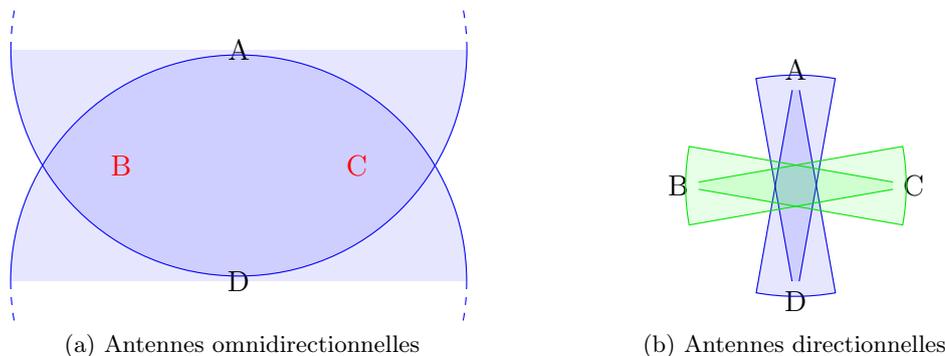


FIGURE 1.1 – Interaction multi-utilisateur

De plus, l'utilisation d'antennes sectorielles permet de diminuer la consommation énergétique. La directivité (et donc son gain dans une direction) d'une antenne est donnée par l'approximation suivante :

$$G \approx 10.\log \left( 4\pi \times \frac{\left(\frac{180}{\pi}\right)^2}{\Theta_{1d} \times \Theta_{2d}} \right) \approx 10.\log \left( \frac{41253}{\Theta_{1d} \times \Theta_{2d}} \right)$$

Source : [http://en.wikipedia.org/wiki/Antenna\\_gain](http://en.wikipedia.org/wiki/Antenna_gain)

$\Theta_{1d}$  et  $\Theta_{2d}$  représentant respectivement l'angle (en degrés) d'ouverture à -3 dB de l'antenne en azimut et en élévation. Plus l'antenne est directive, plus les angles  $\Theta_{1d}$  et  $\Theta_{2d}$  sont petits et plus le gain est grand. Il faut donc moins d'énergie pour réaliser une transmission dans l'axe privilégié de l'antenne.

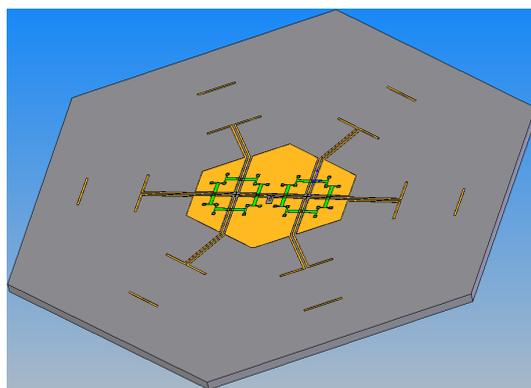


FIGURE 1.2 – Structure d'une antenne à faisceaux agiles

L'IEMN<sup>7</sup>, dans le cadre du projet CPER<sup>8</sup> LOMC<sup>9</sup> a développé une antenne à faisceaux agiles (figure 1.2) en MEMS<sup>10</sup> permettant de « choisir » à l'émission ou à la réception une direction privilégiée (six secteurs couvrant chacun 60°). Il s'agit de six antennes directives simples (type

7. IEMN : Institut d'Électronique, de Microélectronique et de Nanotechnologie [IEM]

8. CPER : Contrat de projets État-région

9. LOMC : Liaison de Proximité Haut Débit entre Objets Mobiles Communicants

10. MEMS : *Microelectromechanical systems*

« yagi ») séparées chacune d'un angle de 60 degrés. Avec l'aide de microcommutateurs à déflexion électrostatique intégrés à l'antenne, il est possible de sélectionner une des six antennes pour émettre ou recevoir dans une direction donnée. De plus, grâce à la nanotechnologie, cette antenne est extrêmement petite.

Cependant, le temps de changement des microcommutateurs n'est pas négligeable et influe directement sur la consommation d'énergie. Plus la tension appliquée est grande, plus ils sont rapides, mais plus ils consomment. Afin de garder une consommation maîtrisée, nous utilisons actuellement un temps de commutation d'une milliseconde. Ce délai non négligeable implique des modifications dans la couche MAC<sup>11</sup> que nous verrons dans la sous-section 3.2.1.

### 1.3 Bande de fréquence 60 GHz

Le spectre électromagnétique est une ressource rare qu'il faut partager entre tous les émetteurs qui sont en fonctionnement. C'est pourquoi l'émission d'onde radio nécessite la plupart du temps une autorisation de la part d'une agence gouvernementale. En France, la planification, la gestion et le contrôle de l'utilisation du spectre électromagnétique sont assurés par l'ANFR<sup>12</sup>. Elle fixe la puissance maximum d'émission, la bande de fréquence utilisable ainsi que son taux d'occupation en fonction du type de transmission.

Cependant, certaines fréquences sont « libres » d'utilisation à condition de respecter une puissance maximum d'émission. C'est le cas des bandes 2400-2483MHz, 5150-5350MHz et 5470-5725MHz en France pour les équipements de transmission de données large bande et systèmes d'accès sans fil [ARC08]. Il est ainsi possible de déployer des réseaux sans fil domestiques sans nécessiter de licence.

Dans ce projet, nous utilisons la bande des 60 GHz récemment autorisée en Europe par l'ETSI<sup>13</sup> [ETS09]. Cette bande de fréquence dispose de quelques avantages :

- Large bande passante : il est possible d'utiliser 4 GHz de bande de fréquence autour de 60 GHz permettant ainsi des communications à très haut débit.
- Fréquence d'absorption de l'oxygène : forte atténuation du signal, idéal pour des communications à courte portée.
- Forte absorption par les matériaux : le signal à 60 GHz ne traverse pas les murs. C'est un point positif pour la sécurité, car les données ne peuvent pas être interceptées depuis l'extérieur d'un bâtiment et ne peuvent pas être brouillées.

### 1.4 Objectifs de la thèse

Dans ce document, nous présentons nos travaux concernant une couche logicielle bas niveau efficace pour les réseaux sans fil ad-hoc disposant d'antennes directionnelles dans la bande de fréquence de 60 GHz. Les nœuds de ces réseaux doivent fonctionner sans infrastructure, c'est-à-dire sans qu'un nœud du réseau soit indispensable à son fonctionnement. Dans la littérature, beaucoup de couches MAC existent déjà pour ce genre de réseaux, mais ne sont pas adaptées au délai introduit par les antennes à commutation de faisceaux présentées dans la section 1.2. Nous présenterons et détaillerons donc une nouvelle pile protocolaire adaptée à ce type d'antennes. L'objectif de cette pile est d'être économe en énergie par l'utilisation d'antennes sectorielles, adaptative grâce à l'utilisation de deux canaux radio (permettant de choisir entre débit, latence et acquittement) et de permettre des liaisons très haut débit.

11. MAC : *Media Access Control* – Contrôle d'accès au support

12. ANFR : Agence Nationale des FRéquences [ANF]

13. ETSI : *European Telecommunications Standards Institute* – Institut européen des normes de télécommunication

## 1.5 Organisation du document

Ce document présente notre étude d'un système de communication haut débit sans fil et plus particulièrement des couches bas niveau (physique, liaison de données) pour antennes à faisceaux agiles. Le chapitre 2 présente un état de l'art concernant les systèmes de communication sans fil. Nous y définissons le modèle OSI<sup>14</sup>, le principe de la couche liaison de données et de sa sous-couche MAC ainsi que les contraintes imposées par l'emploi d'antennes à faisceaux agiles. Pour finir, nous y décrivons les différentes solutions existantes dans le domaine du contrôle d'accès à un canal radio.

Les chapitres 3, 4 et 5 décrivent les contributions apportées au domaine. Le chapitre 3 expose l'architecture de la pile protocolaire que nous proposons. Cette architecture est ensuite détaillée dans le chapitre 4 dans lequel nous présentons la couche MAC et les optimisations possibles. Le chapitre 5 quant à lui s'intéresse à l'implémentation de cette couche MAC sur un prototype.

Enfin, le chapitre 6 conclut nos travaux et présente quelques possibilités d'améliorations et pistes de recherches que nous n'avons pas étudiées dans le présent travail.

---

14. OSI : *Open Systems Interconnection*



## Second chapitre

---

# État de l'art

### Sommaire

---

<b>2.1</b>	<b>Les réseaux sans fil</b>	<b>16</b>
2.1.1	Réseaux sans fil à stations de base	16
2.1.2	Réseaux maillés	16
2.1.3	Réseaux pair-à-pair	17
<b>2.2</b>	<b>Pile de protocoles</b>	<b>17</b>
2.2.1	Le modèle OSI	18
2.2.1.1	Description des couches du modèle OSI	19
2.2.2	Couche liaison de données	19
2.2.3	Contrôle d'accès au support ( <i>Media Access Control - MAC</i> )	20
2.2.4	Découverte de voisinage	21
<b>2.3</b>	<b>Couches MAC pour antennes omnidirectionnelles</b>	<b>21</b>
2.3.1	Couches MAC basées sur le canal	21
2.3.1.1	FDMA	21
2.3.1.2	TDMA	22
2.3.1.3	CDMA	22
2.3.1.4	PDMA	22
2.3.2	Couches MAC basées sur les paquets	23
2.3.2.1	ALOHA	23
2.3.2.2	Slotted ALOHA	23
2.3.2.3	CSMA	24
2.3.2.4	CSMA/CA	24
2.3.2.5	IEEE 802.11 RTS/CTS	25
2.3.2.6	MACAW	26
<b>2.4</b>	<b>Couches MAC pour antennes sectorielles</b>	<b>26</b>
2.4.1	SDMA	26
2.4.2	DMAC	27
2.4.3	MMAC	27
2.4.4	Circular RTS MAC	28
2.4.5	CRCM	28
2.4.6	MDA	29
2.4.7	DMAC/DA	29
2.4.8	DMAC/DA with NPN	30
2.4.9	SWAMP	31
<b>2.5</b>	<b>Conclusion</b>	<b>32</b>

---

Dans ce chapitre, nous présentons dans un premier temps les différents types de réseaux sans fil. Ensuite, nous présentons le principe d'une pile de communication ainsi que le modèle OSI<sup>1</sup>. Nous décrivons ensuite la couche liaison de données puis le contrôle d'accès au support. Nous présenterons finalement différentes approches existantes concernant le contrôle d'accès au support.

## 2.1 Les réseaux sans fil

Un réseau sans fil est un réseau de communication utilisant des ondes radio pour transporter des données. Plusieurs architectures existent dans la littérature définissant la présence ou non d'une infrastructure et la manière dont elle communique.

### 2.1.1 Réseaux sans fil à stations de base

La première architecture de réseau sans fil est l'architecture à stations de base (voir figure 2.1). Chaque nœud utilisateur (en vert) communique avec un des nœuds de l'infrastructure (en bleu) sur lequel il s'est préalablement enregistré. Chaque nœud de l'infrastructure communique avec les autres nœuds de l'infrastructure via une liaison filaire. Toute communication entre deux nœuds utilisateurs se fait donc uniquement via un ou plusieurs nœuds de l'infrastructure.

Cette architecture est très souvent utilisée lorsqu'il existe déjà un réseau filaire déployé dans la zone que l'on souhaite couvrir avec du réseau sans fil. Il suffit de positionner quelques nœuds d'infrastructure et de les connecter au réseau filaire.

On peut citer comme exemple d'infrastructure à stations de base, les réseaux WiFi avec un point d'accès connecté au réseau informatique ou les réseaux de téléphonie mobile comme le GSM<sup>2</sup> ou l'UMTS<sup>3</sup>.

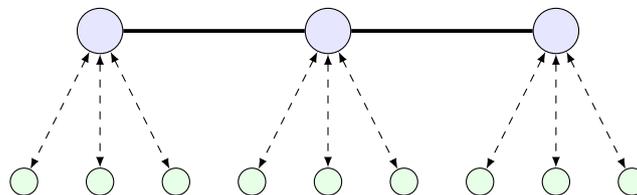


FIGURE 2.1 – Réseau avec trois stations de base et neuf utilisateurs

### 2.1.2 Réseaux maillés

Cette architecture de réseau repose aussi sur des nœuds d'infrastructure tout comme l'architecture précédente. La grande différence est que ces nœuds communiquent aussi sans fil (voir figure 2.2). Ils utilisent généralement un autre canal que celui utilisé par les utilisateurs afin que la bande passante ne soit pas partagée entre les transferts entre nœuds d'infrastructure et les transferts entre l'infrastructure et les utilisateurs.

Les nœuds d'infrastructure peuvent donc être facilement déployés. Ils ne nécessitent pas une infrastructure filaire déjà en place. Très généralement ces nœuds ne sont pas mobiles et le routage avec cette architecture reste identique à l'architecture précédente.

1. OSI : *Open Systems Interconnection*

2. GSM : *Global System for Mobile Communications* – Standard de téléphonie mobile utilisé principalement en Europe et en Asie

3. UMTS : *Universal Mobile Telecommunications System* – Système de téléphonie mobile également appelé 3G (troisième génération) permettant des échanges de données jusqu'à 2 Mbps

Cette architecture peut être utilisée pour couvrir de grandes surfaces. Il est possible d'envisager, par exemple, de fournir une connectivité internet à tout un lotissement. Chaque maison dispose sur son toit d'un nœud d'infrastructure distribuant la connectivité sans fil à l'intérieur de celle-ci. Le trafic remonte de maison en maison jusqu'à un point où les données sont écoulées normalement sur internet. Il existe d'autres études comme le déploiement d'un réseau maillé à Honolulu [Won10].

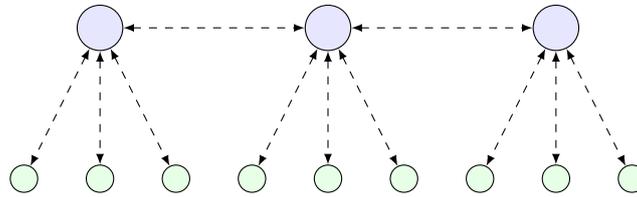


FIGURE 2.2 – Réseau maillé avec trois nœuds d'infrastructure et neuf utilisateurs

### 2.1.3 Réseaux pair-à-pair

Ce type de réseau est dit sans infrastructure. Tous les nœuds du réseau sont égaux et communiquent directement avec leurs voisins les plus proches (voir figure 2.3). Si un nœud souhaite communiquer avec un autre nœud qui n'est pas parmi la liste de ses voisins, il doit utiliser l'un d'eux et transférer les données de proche en proche. On parle de réseau *ad-hoc* ou de MANET<sup>4</sup>.

Ce genre d'infrastructure est souvent utilisé lorsque l'on souhaite créer un réseau spontané. Si plusieurs entités doivent communiquer ensemble, sans matériel supplémentaire il est ainsi possible de créer un réseau sans fil. Il existe néanmoins des réseaux du même style, mais de plus grande envergure. Il faut de ce fait avoir des protocoles de routage adaptés afin de pouvoir trouver son destinataire alors que la topologie du réseau est en perpétuel mouvement.

C'est dans ce contexte que se déroule notre étude de couche bas niveau efficace.

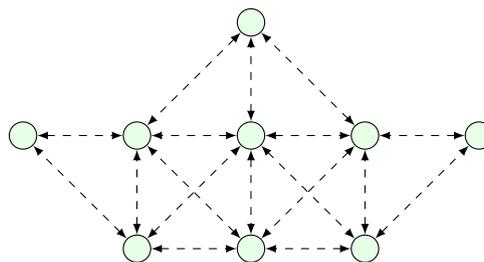


FIGURE 2.3 – Réseau pair-à-pair avec neuf utilisateurs

## 2.2 Pile de protocoles

Une pile de protocoles est une implémentation particulière d'une suite de protocoles de communication réseau. Chaque couche s'appuie sur les protocoles qui sont en dessous afin d'apporter des fonctionnalités : le tout forme ainsi une pile. Le protocole le plus bas définit la

4. MANET : *Mobile Ad-hoc NETWORKs*

connexion physique entre les entités (câble, onde électromagnétique, faisceau laser, ...), alors que le protocole le plus haut fournit l'accès de la pile aux applications. Ce découpage en couches permet aux couches supérieures de s'abstraire des couches inférieures. Ainsi, lors d'une communication entre deux applications, le fait que la liaison physique soit locale, sans fil ou via une dizaine de routeurs à travers Internet est totalement transparent.

Le modèle OSI (décrit dans la sous-section 2.2.1), sur lequel nous nous appuyons dans ce document, est la concrétisation d'un effort pour normaliser les réseaux qui a débuté en 1977 [Zim80] par l'ISO<sup>5</sup> avec l'ITU-T<sup>6</sup>. D'autres modèles existent comme celui de TCP/IP<sup>7</sup> [Bra89] qui peut être aussi décrit par le modèle OSI en agrégeant les trois couches supérieures en une unique couche application.

Notre travail portant sur la couche liaison de données du modèle OSI, nous allons dans un premier temps détailler celui-ci, puis dans un second temps, décrire la couche liaison de données. Enfin, nous détaillerons les mécanismes de la couche MAC<sup>8</sup>, sous-couche de la couche liaison de données sur laquelle est basée une partie de notre travail.

### 2.2.1 Le modèle OSI

Le modèle OSI est décrit en sept couches, de bas en haut, de la liaison physique proprement dite aux applications (figure 2.4a). Chaque couche est une collection de concepts similaires proposant des services à la couche immédiatement supérieure et utilisant les services fournis par la couche inférieure. Ainsi, lors d'une communication, il y a pour chaque couche un protocole qui fournit un service au protocole de la couche supérieure et qui utilise les services de la couche inférieure. Virtuellement, chaque couche communique avec sa couche correspondante au niveau du destinataire indépendamment des couches inférieures (figure 2.4b).

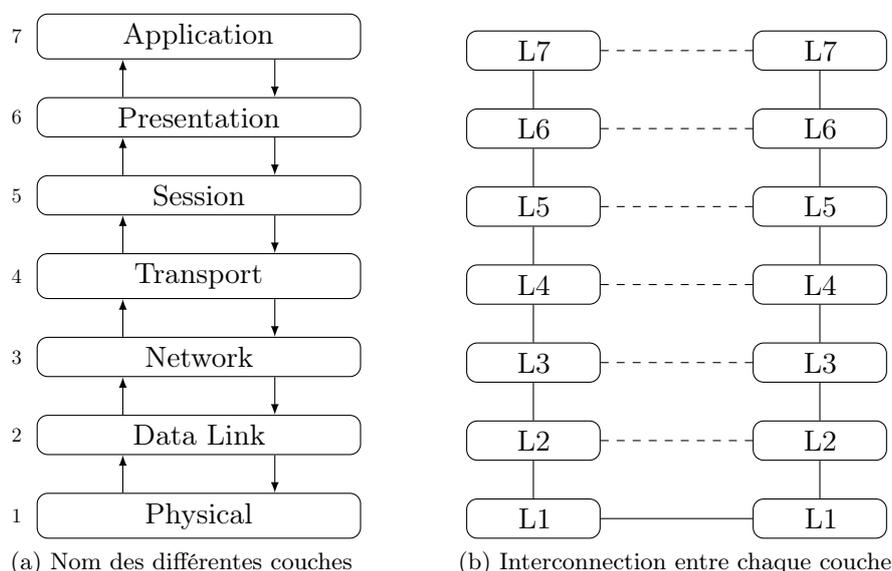


FIGURE 2.4 – Modèle OSI en 7 couches

5. ISO : *International Organization for Standardization* – Organisation Internationale de Normalisation [ISO]

6. ITU-T : *International Telecommunication Union* – Union internationale des télécommunications [ITU]

7. TCP/IP : *Transmission Control Protocol / Internet Protocol* – Protocoles de la couche 3 et 4 du modèle OSI

8. MAC : *Media Access Control* – Contrôle d'accès au support

### 2.2.1.1 Description des couches du modèle OSI

Le modèle OSI est découpé en sept couches :

- 1. Couche physique** La couche physique définit la transmission des bits de façon brute sur un canal de communication. Elle décrit donc comment les informations circulent sur le médium (niveaux des signaux, modulation, ...) ainsi que ses caractéristiques (câble coaxial, paire torsadée, liaison sans fil, fibre optique, ...). Elle assure aussi la résolution des problèmes de contention et le contrôle des flux. Son unité d'information est le bit.
- 2. Couche liaison de données** Cette couche transforme la couche physique en une liaison *a priori* exempte d'erreurs. Pour la couche physique, les données ne sont que des bits se succédant les uns après les autres. La couche liaison de données doit donc reconnaître la frontière entre les trames reçues et commander la couche physique pour l'envoi des données. Son unité d'information est la trame. (Cette couche sera plus amplement développée dans la sous-section 2.2.2.)
- 3. Couche réseau** Elle fournit les moyens de transférer des données de taille variable à partir d'une source et vers une destination à travers un ou plusieurs réseaux tout en maintenant la qualité de service demandée par la couche transport. Elle assure des fonctions de routage réseau et peut gérer la fragmentation et le ré-assemblage des trames. Cette couche introduit un système d'adressage logique et hiérarchique des nœuds dans le réseau. Grâce à ces fonctions, il est possible d'interconnecter plusieurs réseaux (découpage hiérarchique ou couche physique différente) et de les faire communiquer. Son unité d'information est le paquet.
- 4. Couche transport** Cette couche a pour charge de transporter les données sur le réseau de bout en bout. Son rôle principal est de prendre les messages de la couche session, de les découper s'il le faut en unités plus petites et de les passer à la couche réseau. Le transport peut être en mode dit connecté. Dans ce mode, l'expéditeur s'assure que le destinataire peut recevoir les données et s'assure de leur bonne réception. Il peut être aussi en mode non connecté. Dans ce cas les données sont envoyées sans aucun contrôle de la réception. Ce mode peut être utile dans le cas où la retransmission des données n'est pas utile. Pour l'émission en flux de vidéo par exemple, si les données sont réémises, il est généralement trop tard pour que le récepteur puisse les utiliser. Son unité d'information est le message.
- 5. Couche session** Le rôle de cette couche est de fournir aux utilisateurs les outils nécessaires pour gérer l'échange de données, organiser et synchroniser le dialogue. C'est la première couche qui ne s'occupe plus du transfert de données : les services sont orientés application. Elle offre, par exemple, les services suivants : établissement de connexion, synchronisation de l'échange, rapports d'anomalies, ...
- 6. Couche présentation** Cette couche s'occupe de la représentation des données. Elle s'occupe par exemple du codage des caractères permettant ainsi à deux systèmes hétérogènes de communiquer. Elle peut aussi gérer le chiffrement ou la compression des données.
- 7. Couche application** La couche application est le point d'accès des applications aux services réseau. Il existe donc autant de protocoles applicatifs que de services réseau existants. On peut citer par exemple, FTP, HTTP, SMTP, SMB, ...

### 2.2.2 Couche liaison de données

Comme indiqué précédemment, la couche liaison de données est la deuxième couche du modèle OSI. Il s'agit de la couche de protocole qui transfère les données entre les nœuds adjacents dans un réseau sans fil ou entre les nœuds sur le même segment d'un réseau local. Elle peut fournir les

moyens de détecter et éventuellement corriger les erreurs qui peuvent survenir dans la couche physique. Parmi les protocoles les plus utilisés, il y a Ethernet [MB83] pour les réseaux locaux ou encore Point-to-Point Protocol [Sim94], HDLC<sup>9</sup>, ADCCP<sup>10</sup> pour des liaisons point à point.

Les trames de la couche liaison de données ne traversent pas les frontières d'un réseau local. Le routage et l'adressage global sont laissés aux couches supérieures. La couche liaison de données se concentre donc sur la livraison «locale» des trames, l'adressage et l'arbitrage du médium de communication. En effet, si des équipements tentent d'utiliser un médium partagé simultanément, une collision apparaît et la trame est perdue. La couche liaison de données précise donc comment détecter et/ou éviter les collisions.

Afin de préciser le destinataire d'une trame, chaque dispositif est doté d'une adresse unique déterminée par le fabricant. L'entête d'une trame contient donc une adresse source et destinataire qui indique quel dispositif est originaire de la trame et qui est censé la recevoir. Cet adressage n'est pas hiérarchique (contrairement à la couche réseau).

Dans certains protocoles, la couche liaison de données est découpée en deux parties : la couche de contrôle d'accès au support (MAC) et la couche de contrôle de la liaison logique. Celle-ci est la moitié haute de la couche liaison de données qui permet de fiabiliser la couche MAC par un contrôle d'erreur et un contrôle de flux.

Nous allons maintenant détailler la couche de contrôle d'accès au support.

### 2.2.3 Contrôle d'accès au support (*Media Access Control - MAC*)

Certains supports de communication ne permettent pas l'envoi de données par plusieurs utilisateurs simultanément. C'est le cas des topologies en bus, en anneau ou pour les liaisons sans fil par exemple. Si cela devait arriver, dans le meilleur des cas une seule des deux communications serait utilisable. Sinon, généralement, les deux trames envoyées seront brouillées et la communication sera perdue : cela s'appelle une collision. Il va donc falloir éventuellement retransmettre les données, ce qui prend du temps (empêchant d'envoyer de nouvelles données) et consomme de l'énergie inutilement. Il faut donc essayer d'éviter les collisions et de détecter lorsque cela arrive. C'est le rôle de la couche MAC.

Celle-ci est la moitié basse de la couche liaison de données. Elle sert d'interface entre la couche de contrôle de la liaison logique et la couche physique. Son rôle est principalement de :

**Fournir un système d'adressage** Chaque équipement réseau dispose d'une adresse unique déterminée par le fabricant. Cette adresse doit permettre d'identifier de manière unique un équipement dans l'ensemble du réseau. Chaque fabricant disposant d'un sous-ensemble des adresses possibles, ils sont responsables de l'unicité de chaque adresse.

Lors de l'envoi d'un message, la couche MAC rajoute les informations de source et de destination. Lors de la réception, la couche MAC ne transmet les informations à la couche supérieure que si l'adresse de destination du message correspond à l'adresse de l'équipement.

Le système d'adressage peut aussi fournir un moyen d'envoyer un message à tous les destinataires, on parle d'adresse de *broadcast*. Dans ce cas, tous les messages reçus avec cette adresse sont transmis à la couche supérieure. L'adresse fournie par la couche MAC est appelée adresse MAC.

**Contrôler l'accès au support** Lors d'un accès à un support partagé, il faut faire en sorte de ne pas perturber les communications en cours afin d'éviter les collisions. Plusieurs techniques de contrôle existent dans la littérature et sont détaillées dans les sections 2.3 et 2.4.

---

9. HDLC : *High-Level Data Link Control*

10. ADCCP : *Advanced Data Communication Control Procedures*

### 2.2.4 Découverte de voisinage

Les réseaux informatiques peuvent être séparés en deux catégories :

**Avec infrastructure** Le réseau est soit configuré statiquement par un contrôle extérieur (un administrateur par exemple) ou dispose d'une entité centralisée qui prend les décisions. Chaque élément du réseau dépend donc du bon fonctionnement de ceux-ci pour fonctionner.

**Sans infrastructure** Le réseau est dit auto-organisé. Il ne dispose pas d'entité centralisée ou de contrôle extérieur pour l'aider à fonctionner. Uniquement grâce à sa pile protocolaire, il doit être capable de s'adapter à son environnement et aux modifications qu'il peut subir.

C'est dans ce deuxième cas qu'un nœud doit être capable de découvrir son environnement afin de prendre les bonnes décisions. L'absence d'infrastructure permet de déployer rapidement un réseau dans lequel les nœuds peuvent se déplacer librement tout en gardant la connectivité.

La découverte du voisinage est donc un service fondamental pour ce type de réseau. Il permet au nœud de découvrir par lui-même quels sont les autres nœuds présents dans le réseau et contactables directement. Utilisée au niveau de la couche réseau avec les protocoles de routage (OSPF<sup>11</sup> [Moy91], OLSR<sup>12</sup> [CJ03] ou AODV<sup>13</sup> [PBRD03]) permettant de vérifier si le prochain nœud de la route est un de nos voisins, la découverte de voisinage est aussi beaucoup employée par les couches MAC des réseaux sans fil auto-organisés.

Chaque nœud envoie périodiquement un petit message (appelé message « Hello ») indiquant sa présence à ses voisins. Ceux-ci peuvent donc dresser une table contenant la liste des nœuds présents dans leur zone de communication. Il est ainsi possible d'optimiser la couche MAC en fonction de la liste et des états connus des voisins. Cela est d'autant plus critique avec l'utilisation des antennes directionnelles où les nœuds doivent connaître la position géographique des voisins afin de communiquer avec eux.

Certains protocoles sont spécialisés dans l'envoi du message « Hello » comme TAP<sup>14</sup> [IMSR07] qui optimise la fréquence d'envoi du message en fonction de l'évolution du voisinage.

## 2.3 Couches MAC pour antennes omnidirectionnelles

Dans un premier temps, nous allons détailler les couches MAC dans le cas classique des antennes omnidirectionnelles. Le premier moyen d'éviter les collisions est de séparer physiquement les communications afin qu'elles ne puissent pas interférer entre elles. C'est le cas des couches MAC basées sur le canal.

### 2.3.1 Couches MAC basées sur le canal

Dans ce cas, il est fait en sorte que les communications ne puissent physiquement entrer en collision. Il existe quatre méthodes principales :

#### 2.3.1.1 FDMA<sup>15</sup>

La séparation des communications s'effectue en utilisant des bandes de fréquence différentes à chaque utilisateur. Si les fréquences sont suffisamment éloignées, les communications n'interfèrent pas. Bien que très simple à l'utilisation, cette méthode a quelques inconvénients. Elle nécessite

---

11. OSPF : *Open shortest path first*

12. OLSR : *Optimized link state routing protocol*

13. AODV : *Ad hoc On-Demand Distance Vector*

14. TAP : *Turnover based Adaptive HELLO Protocol*

15. FDMA : *Frequency Division Multiple Access* – Accès multiple par répartition en fréquence

des filtres radio de bonne qualité afin de ne pas déborder sur les fréquences adjacentes et est fortement consommatrice en bande de fréquence. De plus, une bande de fréquence étant allouée à un seul utilisateur, si celui-ci ne l'utilise pas elle reste inutilisée. Ce qui diminue la bande passante globalement disponible.

### 2.3.1.2 TDMA <sup>16</sup>

Ici, une partie du temps est allouée à chaque utilisateur. Il peut, s'il en a besoin, émettre un message pendant cet intervalle de temps ou laisser le canal inoccupé (voir figure 2.5). Cette technique ne nécessite aucune modification au niveau de l'interface radio. Par contre, il est nécessaire d'avoir une bonne synchronisation entre tous les utilisateurs du réseau afin d'être sûr de n'émettre que dans son créneau. Cette synchronisation est très difficile à obtenir dès que le réseau augmente en taille. De plus, comme avec le FDMA, si un utilisateur n'a pas besoin d'utiliser son intervalle de temps, il reste inutilisé.

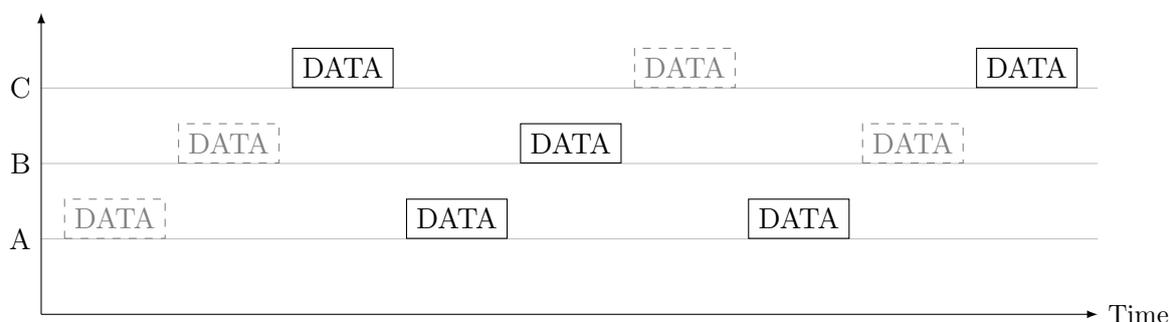


FIGURE 2.5 – TDMA

### 2.3.1.3 CDMA <sup>17</sup>

Le CDMA utilise l'étalement de spectre avec des codes dits orthogonaux. Si deux émissions sont réalisées en même temps avec deux codes orthogonaux, le récepteur, connaissant le code utilisé par l'émetteur, pourra récupérer les informations sans que la deuxième transmission interfère. L'étalement de spectre permet donc à plusieurs utilisateurs de communiquer de manière simultanée, mais consomme plus de bandes de fréquence.

L'inconvénient de cette technique est la lourde quantité de calcul nécessaire tant à l'émission qu'à la réception pour encoder et décoder le signal ainsi que l'attribution des codes entre les utilisateurs. Cette technique fonctionne bien dans le cas des réseaux mobiles par exemple où seule la station de base doit connaître l'intégralité des codes des utilisateurs. Comme elle est le point central du réseau, il lui est aussi possible de facilement attribuer les codes aux utilisateurs en fonction des besoins.

### 2.3.1.4 PDMA <sup>18</sup>

Grâce à la polarisation de l'onde électromagnétique, il est possible d'envoyer des informations qui ne seront reçues que par les récepteurs avec une antenne ayant la même polarisation. On

16. TDMA : *Time Division Multiple Access* – Accès multiple à répartition dans le temps

17. CDMA : *Code Division Multiple Access* – Accès multiple par répartition en code

18. PDMA : *Polarisation Division Multiple Access* – Accès multiple par répartition en polarisation

peut ainsi émettre avec une polarisation horizontale, verticale ou circulaire, créant de ce fait trois canaux virtuels.

Cette méthode est très utilisée dans les liaisons satellites avec la terre, permettant de multiplier ainsi par deux (horizontal et vertical) la bande passante disponible. Elle est par contre difficilement applicable dans un réseau sans fil classique à cause de la spécificité des antennes.

### 2.3.2 Couches MAC basées sur les paquets

Les techniques d'accès au canal précédentes ont presque toutes le même inconvénient : la rigidité. Chacune séparément fonctionne très bien dans un contexte particulier où la topologie du réseau ne change pas régulièrement. Il serait compliqué de réattribuer l'intégralité des fréquences ou les intervalles de temps à chaque fois qu'un utilisateur quitte ou rejoint le réseau.

De plus, certains de ces protocoles ont besoin d'un point central permettant la synchronisation en temps ou la distribution des codes. Nous allons maintenant détailler les couches MAC non plus basées sur le canal, mais sur les paquets. Pour chaque paquet à envoyer, la couche MAC doit décider comment et quand envoyer les données. Ceci complique la couche MAC, mais permet d'être beaucoup plus flexible.

#### 2.3.2.1 ALOHA<sup>19</sup>

ALOHA[Abr70] fait partie des protocoles dits à récupération de collision. Le fonctionnement est assez simple : s'il y a des données à envoyer, on les envoie, s'il y a une collision, on les renverra « plus tard » (voir figure 2.6). Il permet donc de fiabiliser la liaison radio qui ne l'est pas.

Le point clé d'ALOHA est de savoir quand renvoyer les données. Une fois la collision détectée, il faut que chaque station émettrice attende un certain temps avant de renvoyer le message. Afin d'éviter une deuxième collision, chaque station attend un temps aléatoire avant de retransmettre.

Ce protocole a été développé par l'Université d'Hawaii et déployé en 1970. Il est l'un des réseaux informatiques pionnier inspirant certains protocoles encore utilisés aujourd'hui.

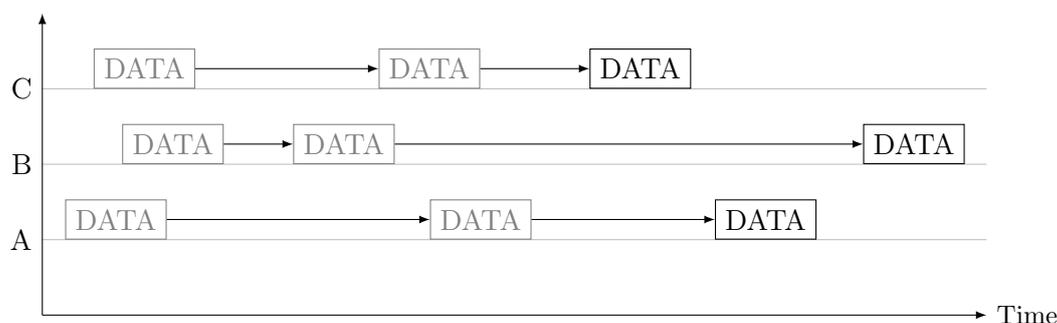


FIGURE 2.6 – Pure ALOHA, les paquets grisés indiquent les collisions

#### 2.3.2.2 Slotted ALOHA

Une amélioration à ALOHA est d'introduire des intervalles de temps discrets : une station ne peut émettre qu'au début de l'intervalle de temps (voir figure 2.7). Comme les paquets sont de taille fixe, les collisions ne peuvent se produire qu'à l'intérieur de l'intervalle de temps réduisant ainsi leur risque.

19. ALOHA : Nom inspiré du mot aloha signifiant en langue hawaïenne bonjour, au revoir

Cette amélioration permet d'augmenter par deux les performances d'ALOHA [Rob75], mais nécessite une synchronisation entre les différents nœuds afin de bien respecter les intervalles de temps.

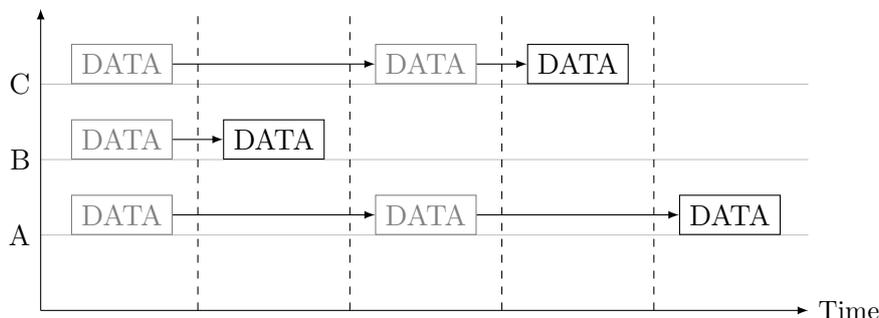


FIGURE 2.7 – Slotted ALOHA, les paquets grisés indiquent les collisions

### 2.3.2.3 CSMA<sup>20</sup>

Avec ce protocole, le nœud vérifie l'absence de trafic avant d'envoyer des données. Si le canal est utilisé, il attend qu'il devienne libre avant d'envoyer, sinon il envoie directement. En cas de collision, il attend un temps aléatoire avant de recommencer le processus.

Il existe certaines variantes comme le *p-persistent CSMA* [KT75] qui, quand le canal devient libre, n'envoie le paquet qu'avec une probabilité  $p$ . Dans le cas où la station décide de ne pas retransmettre, elle attend un certain temps avant de recommencer le processus jusqu'à ce que le paquet soit envoyé. Comme dans le cas précédent (qui est une généralisation de cette variante avec  $p = 1$ ), s'il y a une collision le nœud attend un temps aléatoire avant de réémettre.

### 2.3.2.4 CSMA/CA<sup>21</sup>

Ce protocole est une amélioration de CSMA réduisant la probabilité des collisions. Pour ce faire, si, lors d'une transmission, le nœud détecte que le canal est utilisé, il diffère sa transmission pendant un aléatoire puis recommence le processus. La probabilité que deux nœuds décident de retransmettre en même temps suite à une détection de canal occupé est donc diminuée, ce qui réduit le risque de collision.

Le temps d'attente aléatoire est géré par un algorithme appelé *truncated binary exponential backoff*. Celui-ci est choisi aléatoirement en fonction du nombre de tentatives de retransmission. Après  $i$  collisions, un nombre est choisi aléatoirement entre 0 et  $2^i - 1$ . Ce nombre est ensuite multiplié par un intervalle de temps élémentaire ( $20 \mu s$  dans le cas de IEEE 802.11) pour déterminer le temps d'attente. Si pendant ce laps de temps le canal est occupé, le décompte est arrêté et reprend lorsque le canal redevient libre.

À partir d'un certain nombre d'essais, l'intervalle ne grandit plus (d'où le *truncated*) afin d'éviter les temps d'attente trop longs. Dans le cas d'un réseau surchargé, il se peut que le processus n'arrive jamais à envoyer les données. Pour pallier cette probabilité, si le nombre d'essais excède un certain seuil le processus est abandonné et le paquet est supprimé.

20. CSMA : *Carrier Sense Multiple Access* – Accès multiple par détection de porteuse

21. CSMA/CA : *Carrier Sense Multiple Access with Collision Avoidance* – Accès multiple par détection de porteuse avec évitement des collisions

### 2.3.2.5 IEEE 802.11 RTS/CTS

Ces deux derniers protocoles, reposant sur la mesure d'activité du canal radio, souffrent du problème appelé « terminal caché » [TK75]. Si le nœud *A* communique avec le nœud *B*, le nœud *C*, qui ne peut pas entendre le nœud *A*, pense que le canal est libre et peut envoyer des données à *B* (voir figure 4.8) brouillant ainsi sa réception.

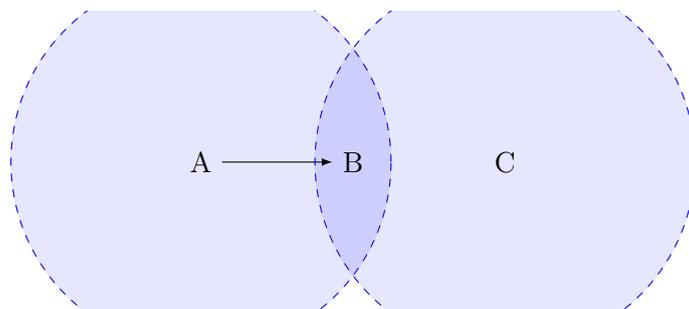


FIGURE 2.8 – Problème du terminal caché

Pour surmonter ce problème, IEEE 802.11 DCF<sup>22</sup> [IEE97] propose une « poignée de main », mise en oeuvre en collaboration avec CSMA/CA. Un nœud voulant émettre un paquet sur le réseau émet d'abord un paquet RTS<sup>23</sup> au destinataire. Si celui-ci reçoit correctement le message et est disponible, il émet un paquet CTS<sup>24</sup> au récepteur. Ces deux paquets, reçus aussi respectivement par les voisins du récepteur et du destinataire, les informent qu'il ne faut pas commencer de transmission sinon ils risquent de brouiller celle qui vient de commencer. Une fois le CTS reçu par l'émetteur, il envoie les données à proprement parler. Le nœud *C* considérera le canal comme occupé (zone grisée sur la figure 2.9) même si pour lui il est libre.

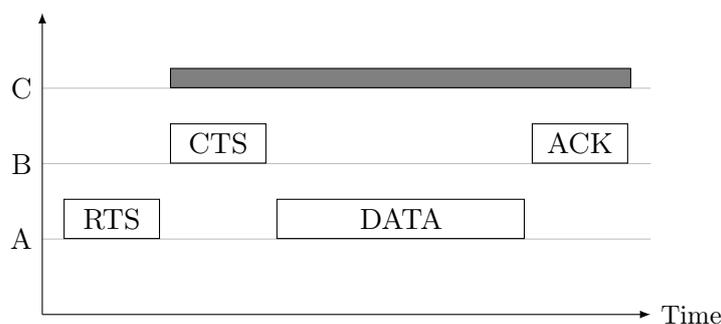


FIGURE 2.9 – RTS/CTS

Ce mécanisme n'est néanmoins pas sans faille, l'envoi des RTS/CTS ajoute du trafic sur le réseau et l'envoi même des RTS peut générer des collisions (par exemple si *A* et *C* envoient leur RTS en même temps). Il se peut aussi que le nœud *C* n'entende pas le CTS du nœud *B* si par exemple il est lui-même en train d'émettre un CTS ou un RTS.

22. DCF : *Distributed Coordination Function*

23. RTS : *Request to Send*

24. CTS : *Clear to Send*

Pour des paquets de petite taille, IEEE 802.11 propose de les envoyer directement sans envoyer de RTS afin de diminuer la charge du réseau, le paquet ayant en effet moins de risque de collisionner qu'une séquence RTS/CTS/DATA/ACK<sup>25</sup>.

### 2.3.2.6 MACAW<sup>26</sup>

Le mécanisme de RTS/CTS de IEEE 802.11 est l'héritier de MACAW[BDSZ94]. Mais celui-ci possède un système d'échange plus complexe pour gérer le problème du terminal caché.

En plus d'utiliser les paquets RTS/CTS, MACAW envoie un paquet DS<sup>27</sup> juste avant d'envoyer un gros paquet de données au destinataire. Ce paquet permet de signaler aux voisins de l'émetteur que l'échange RTS/CTS s'est bien déroulé. Dans le cas de 802.11 un nœud considère que le canal est occupé dès qu'il reçoit un paquet RTS et n'a aucun moyen de savoir s'il a bien été reçu ou pas.

Il se peut qu'un nœud  $B$  reçoive un RTS d'un nœud  $A$  et ne puisse pas lui répondre, car une transmission est en cours dans son voisinage. Dans ce cas, l'expéditeur du RTS considère que son envoi a collisionné et lance son algorithme de *backoff*. Malheureusement, s'il y a beaucoup de trafic dans le voisinage de  $B$ ,  $A$  aura beaucoup de mal à finaliser son échange, car il y a une grande probabilité qu'une transmission soit en cours dans le voisinage de  $B$  lorsque son *backoff* se termine. Pour ce faire, MACAW propose à  $B$  d'envoyer à  $A$  un paquet RRTS<sup>28</sup> dès que le canal se libère.  $A$  sait donc que  $B$  est disponible et lui envoie directement son RTS. Il a ainsi plus de chances de gagner l'accès au canal.

On voit donc que 802.11 est une version simplifiée de MACAW, mais rajoute le principe de CSMA que MACAW ne supporte pas.

## 2.4 Couches MAC pour antennes sectorielles

Il existe beaucoup de protocoles MAC dans la littérature, mais ceux-ci reposent principalement sur l'écoute du canal et le fait que les transmissions sont écoutables dans toutes les directions.

Dès qu'un nœud possède des antennes sectorielles (ou directionnelles), ce principe ne fonctionne plus et ces protocoles deviennent inefficaces. Nous allons détailler quelques protocoles de la littérature dédiés aux antennes sectorielles.

### 2.4.1 SDMA<sup>29</sup>

Utilisée surtout en téléphonie mobile, cette méthode d'accès au canal permet d'optimiser l'énergie dépensée lors des émissions et de réduire les interférences avec des cellules adjacentes utilisant la même fréquence. En effet, grâce à un réseau d'antennes, on peut moduler la directivité du signal en fonction des utilisateurs présent dans la cellule. Il ne sert à rien d'envoyer à pleine puissance dans toutes les directions alors qu'un seul utilisateur est proche de la borne.

De plus, avec un réseau d'antennes on peut aussi focaliser la réception dans une direction, réduisant le bruit et les interférences reçues.

Les mobiles, quant à eux, ne peuvent pas embarquer de réseaux d'antennes à cause de la place réduite à l'intérieur du mobile et des limitations de poids. Ils se contentent donc de moduler leur puissance d'émission. Cela permet aussi de réduire les interférences dans les cellules voisines et d'économiser de l'énergie, augmentant de ce fait la durée de la batterie.

25. ACK : *Acknowledgment code* – Acquiescement des données

26. MACAW : *Multiple Access with Collision Avoidance for Wireless*

27. DS : *Data-Sending*

28. RRTS : *Request for Request to Send*

29. SDMA : *Space-Division Multiple Access* – Accès multiple par division géographique

### 2.4.2 DMAC<sup>30</sup>

Ko *et al.* ont proposé DMAC [KH99]. Tout comme IEEE 802.11 DCF, DMAC utilise des paquets RTS et CTS pour l'accès au canal. Les nœuds utilisant des antennes directionnelles par formation de faisceaux, le paquet RTS est envoyé uniquement dans la direction du destinataire. Par contre, afin que la réception destinataire ne soit pas dérangée par ses voisins, celui-ci renvoie son CTS dans toutes les directions en activant toutes ses antennes (voir figure 2.13a).

DMAC considère chaque antenne comme indépendante. Ainsi, s'il reçoit dans l'une des directions un RTS ou un CTS, il considère la direction comme occupée au même sens que IEEE 802.11 DCF et ne l'utilisera plus jusqu'à la fin de la transmission en cours. Cependant, il lui sera impossible de réaliser les opérations omnidirectionnelles, car il a besoin que toutes les antennes soient disponibles.

De plus, les paquets RTS et CTS doivent contenir les positions géographiques des nœuds émetteurs et récepteurs, mais DMAC ne précise pas comment sont obtenues ces positions. Il n'est donc pas détaillé comment le nœud émetteur fait pour choisir la direction d'envoi du RTS.

### 2.4.3 MMAC<sup>31</sup>

En 2002, DMAC a été étendu et amélioré par MMAC [CYRV02]. Ce protocole considère que le gain des antennes est plus élevé en mode directionnel qu'en mode omnidirectionnel. Un nœud est donc capable de recevoir des messages plus lointains s'il écoute dans une direction particulière. La liste des voisins d'un nœud varie en fonction de la configuration actuelle de leurs antennes.

MMAC définit deux types de voisins :

- *Direction-Omni (DO)* : Un nœud  $B$  est un voisin DO de  $A$  si  $B$  peut recevoir des messages de  $A$  alors qu'il est en mode omnidirectionnel.
- *Direction-Direction (DD)* : Un nœud  $B$  est un voisin DD de  $A$  si  $B$  ne peut recevoir des messages de  $A$  que lorsqu'il écoute dans la direction de  $A$ . Tous les voisins DO sont donc aussi des voisins DD.

Dans l'exemple de la figure 2.10,  $A$  et  $F$  sont des voisins DD alors que  $A$  et  $B$  sont des voisins DO. Si  $A$  veut transmettre des données à  $F$  il faut donc que  $F$  oriente son faisceau dans la direction de  $A$ . Pour ce faire,  $A$  va utiliser les nœuds  $B$  et  $C$  afin de relayer son RTS et attendre que  $F$  lui réponde.

Lorsque  $F$  a reçu le RTS, il peut orienter son faisceau en direction de  $A$  et envoyer un CTS.  $A$  sait ainsi que  $F$  est maintenant directement joignable et peut lui envoyer ses données.

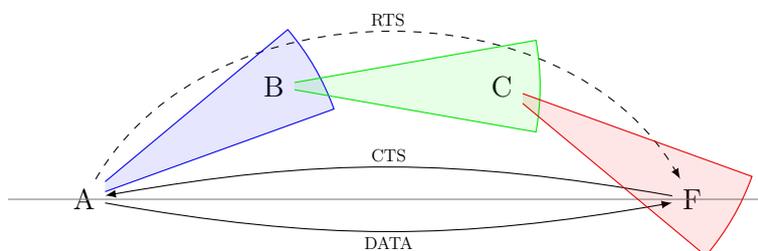


FIGURE 2.10 – Scénario d'envoi d'un paquet avec MMAC

30. DMAC : *Directional MAC*

31. MMAC : *Multi-hop RTS MAC*

Toutes ces opérations respectent l'accès au canal tel que défini par DMAC comme l'écoute du canal avant transmission ou le principe de *backoff* en cas de collision.

Tout comme DMAC, MMAC ne définit pas comment sont obtenues les coordonnées géographiques des nœuds ni comment la route permettant l'envoi du RTS est construite.

#### 2.4.4 Circular RTS MAC

Ce protocole apporte une première réponse au problème de la position géographique des nœuds [KJT03]. Il considère dans un premier temps l'emploi d'antennes directionnelles ayant un nombre de directions fini (voir figure 2.11) [LR99, Rap01].

Lors de l'envoi du RTS, l'émetteur va l'envoyer dans une première direction (faisceau 1 par exemple). Juste après, il utilise la direction suivante puis renvoie le message. Le processus s'arrête une fois que toutes les directions ont été couvertes (envoi sur le faisceau  $M$ ). Ce RTS contient la durée totale attendue de la transmission. Ainsi, les voisins de l'émetteur ayant reçu le RTS peuvent décider de retarder leurs transmissions dans la direction de l'émetteur.

Le destinataire du RTS connaît la direction de l'émetteur en fonction de la puissance reçue sur chacune de ses antennes. Il émet donc son CTS dans la direction supposée de l'émetteur, après avoir attendu que l'envoi du RTS soit terminé pour toutes les directions (voir figure 2.13b). Enfin, l'échange continue comme dans IEEE 802.11 par les données puis un ACK.

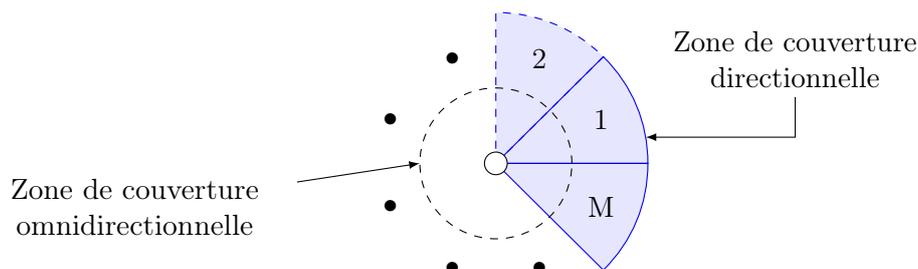


FIGURE 2.11 – Un nœud avec  $M$  faisceaux

#### 2.4.5 CRCM<sup>32</sup>

Le protocole précédent ne traite pas le problème du terminal caché du côté du récepteur. Comme le CTS n'est envoyé que dans la direction de l'émetteur, les nœuds voisins du récepteur ne savent pas qu'une transmission est en cours et peuvent provoquer une collision s'ils envoient des données.

Pour résoudre ce problème, CRCM propose d'envoyer aussi le CTS dans plusieurs directions [JBK<sup>+</sup>05]. Il est par contre inutile d'envoyer le message dans les zones qui ont été couvertes par le RTS. Ainsi, CRCM va faire en sorte de n'envoyer le CTS que dans les directions où des nœuds n'ont pas pu recevoir le RTS (voir figure 2.12). Au final, le CTS n'est envoyé que dans une zone semi-circulaire au maximum disjointe de la zone dans laquelle le RTS a été envoyé plus la direction de l'émetteur (voir figure 2.13c).

Si une des directions requises pour le RTS ou CTS est considérée comme occupée (car un RTS ou un CTS a été reçu dans cette direction), celle-ci est ignorée et l'envoi circulaire continue dans la direction suivante.

32. CRCM : *Circular RTS and CTS MAC*

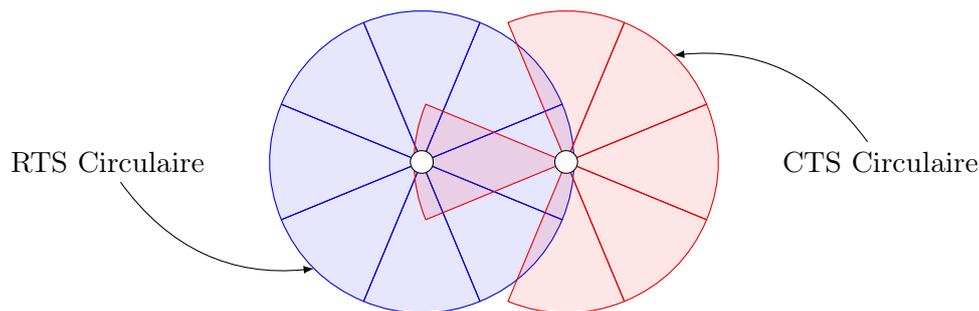


FIGURE 2.12 – Sélection des secteurs pour le CTS par CRCM

### 2.4.6 MDA <sup>33</sup>

Afin d'améliorer les délais induits par l'envoi circulaire des RTS et CTS, MDA propose de découper leur envoi en trois temps [GCA05] :

- L'émetteur envoie son RTS dans la direction du destinataire après avoir vérifié que le canal est libre (en bleu sur la figure 2.13d).
- Une fois reçu, le destinataire envoie juste après un CTS dans la direction de l'émetteur en s'assurant que le canal est libre (en rouge sur la figure 2.13d).
- Une fois le canal alloué par l'échange, l'émetteur et le récepteur vont envoyer simultanément leur RTS et CTS de manière circulaire. Pour ce faire, ils ne doivent pas envoyer de message dans la même zone sous risque de générer une collision. Chaque nœud va donc envoyer son paquet en commençant à l'antenne suivante par rapport à son envoi et couvrir la moitié de la surface. L'autre nœud faisant la même chose et tournant dans la même direction ils vont couvrir une zone qui est diamétralement opposée. Cette procédure est appelée DOD RTS <sup>34</sup> et DOD CTS <sup>35</sup> (en vert sur la figure 2.13d).

Pour économiser plus de temps, les nœuds vont sauter les directions considérées comme occupées. Mais il faut s'assurer que le récepteur ait bien terminé son DOD CTS avant de lui envoyer des données. Le récepteur doit donc générer une liste de secteurs et l'intégrer à son CTS initial. Ainsi, l'émetteur sait combien de temps il doit attendre avant d'envoyer les données.

### 2.4.7 DMAC/DA <sup>36</sup>

Ce protocole, extension de DMAC, tente de diminuer l'*overhead* aussi bien en terme de temps qu'en terme d'occupation du médium généré par l'envoi des RTS et CTS [TBW07]. Le but est de réussir à diminuer le nombre de secteurs dans lesquels on doit envoyer des paquets de contrôle.

En construisant une table contenant la liste des voisins par secteur, un nœud est capable de savoir dans quels secteurs il y a des émetteurs potentiels. De plus, DMAC/DA considère que si un nœud nous a envoyé un paquet récemment, il est fort probable qu'il nous en renvoie un autre. Il sélectionne donc uniquement les secteurs ayant des nœuds nous ayant transmis des données récemment pour l'envoi des paquets de contrôle.

Lors de l'envoi de données, le protocole commence comme MDA. L'émetteur envoie dans un premier temps son RTS dans la direction du destinataire et celui-ci envoie son CTS dans

33. MDA : *MAC protocol for Directional Antennas*

34. DOD RTS : *Diametrically Opposite Directional RTS*

35. DOD CTS : *Diametrically Opposite Directional CTS*

36. DMAC/DA : *Directional MAC with Deafness Avoidance*

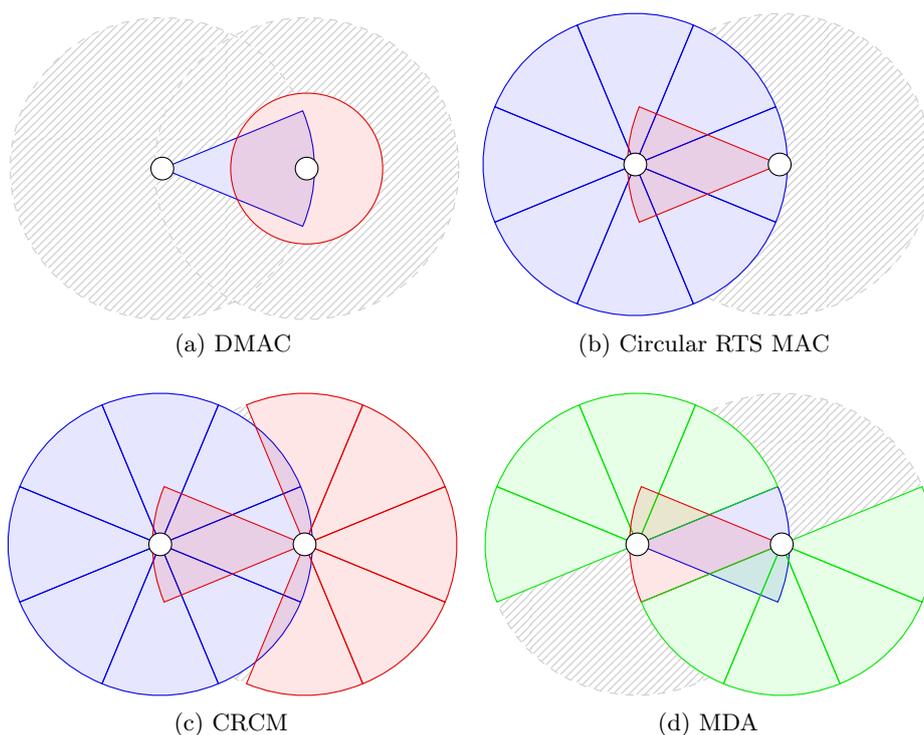


FIGURE 2.13 – Récapitulatif des secteurs utilisés pour l'envoi des RTS/CTS

la direction de l'émetteur. Ensuite, chaque nœud transmet un paquet WTS<sup>37</sup> dans les secteurs précédemment sélectionnés afin de prévenir les potentiels transmetteurs qu'une communication est en cours. Les paquets RTS et CTS initiaux contenant le nombre de secteurs utilisés pour le WTS, chacun des nœuds sait combien de temps attendre avant de respectivement envoyer et recevoir les données. Celles-ci sont envoyées de manière directionnelle, tout comme l'accusé de réception.

De plus, si un nœud reçoit un WTS alors qu'il était dans une procédure de *backoff* avec ce nœud, le compteur est réinitialisé à sa valeur par défaut et recommence son décompte une fois l'échange terminé (la durée d'attente est déduite des informations présentes dans le WTS).

#### 2.4.8 DMAC/DA with NPN<sup>38</sup>

Pour améliorer encore la sélection des secteurs, le protocole précédent est étendu par *DMAC/DA with NPN* [TBW07]. Cette variation propose de rajouter dans l'en-tête du paquet de données, un bit indiquant si le nœud a un paquet pour le même destinataire au sommet de sa pile d'envois. Ainsi, le destinataire sait qu'il va potentiellement bientôt recevoir un autre paquet de l'émetteur et marque celui-ci comme émetteur potentiel. Sinon il est considéré comme simple voisin.

De ce fait, lors du prochain échange effectué par le destinataire, s'il n'y a pas d'autres émetteurs potentiels dans le secteur de l'émetteur, aucun paquet WTS ne sera transmis vers celui-ci.

37. WTS : *Wait To Send*

38. NPN : *Next Packet Notification*

### 2.4.9 SWAMP<sup>39</sup>

Contrairement aux protocoles précédents utilisant des antennes à secteurs fixes (voir figure 2.11), SWAMP [TNW04, TNT<sup>+</sup>06, WMB<sup>+</sup>07] utilise une antenne qui est capable de former un faisceau dans n'importe quelle direction. Cette antenne doit être capable de former quatre types de faisceaux. De plus, l'antenne doit former le faisceau en une durée inférieure à un SIFS<sup>40</sup> de IEEE 802.11 DCF. La figure 2.14 illustre les quatre types de faisceaux nécessaires. Dans cette illustration, deux nœuds peuvent communiquer si le faisceau d'émission et de transmission sont au moins tangents.

Une telle antenne, ESPAR<sup>41</sup>, est proposée par Cheng *et al.* dans [CHIO03]. Cette antenne, composée d'un élément actif au milieu et de six éléments passifs disposés en cercle (rayon de  $0.25\lambda$ ), permet d'envoyer et de recevoir soit dans toutes les directions avec un gain  $G^0$  de -0.83 dBi soit dans une direction particulière avec un gain  $G^d$  de 5.5 dBi. Les éléments passifs permettent par couplage électromagnétique de former un faisceau dans une direction ou de modifier la portée du signal. À 2.5 GHz, le disque occupé par les éléments passifs mesure 6 cm de diamètre pour un volume total de l'antenne de 12 cm × 12 cm × 6 cm.

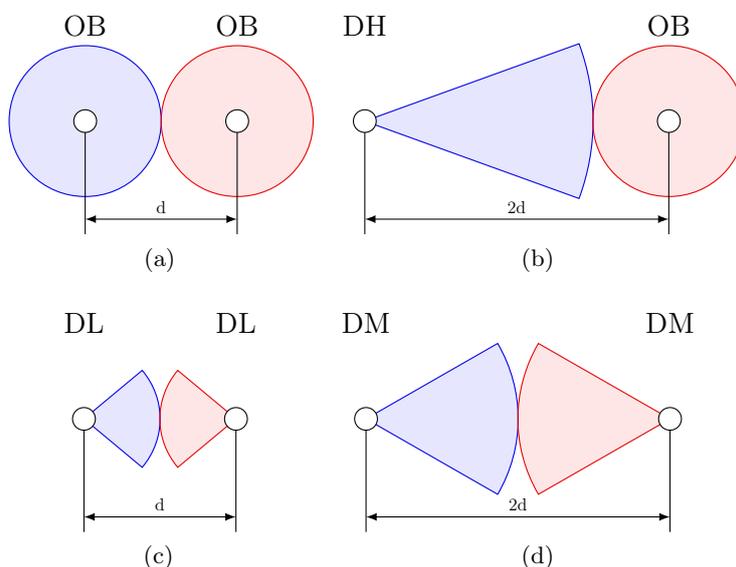


FIGURE 2.14 – Différents faisceaux nécessaires pour SWAMP

Le nœud de gauche représente l'émetteur et celui de droite le récepteur. Les faisceaux OB (omnidirectionnel) et DL (directionnel avec faible gain) sont utilisés dans un mode appelé OC. Ce mode est sélectionné lorsque le récepteur est à portée omnidirectionnelle de l'émetteur ou quand l'émetteur n'a aucune information sur la position du récepteur. Dans un premier temps, l'émetteur envoie un RTS en mode OB avec sa position géographique. Le récepteur envoie, à son tour, un CTS en mode omnidirectionnel avec sa position et la position de l'émetteur. L'émetteur, quant à lui, envoie un paquet SOF<sup>42</sup> en mode OB une fois le CTS reçu. Ainsi, les voisins respectifs de l'émetteur et du récepteur savent qu'une communication est en cours et reçoivent les coordonnées

39. SWAMP : *Smart Antennas Based Wider-range Access MAC Protocol*

40. SIFS : *Short Inter Frame Space* – Temps minimal qui sépare l'envoi de deux paquets sur le réseau dans IEEE 802.11 ( $\sim 10 \mu s$  dans 802.11b)

41. ESPAR : *Electronically Steerable Parasitic Array Radiator* – Antenne orientable électroniquement grâce à un ensemble d'éléments parasites

42. SOF : *Start Of Frame*

des deux nœuds. Ces informations seront utiles pour le mode étendu (EC). S'ensuivent l'échange des données ainsi que l'accusé de réception envoyés chacun en mode DL pour réduire l'occupation spatial de l'échange. Le canal peut ainsi être plus rapidement utilisé par les voisins à condition d'émettre dans d'autres directions.

Si l'émetteur connaît la position géographique du récepteur, il est capable de communiquer avec lui à une distance deux fois plus élevée qu'avec le mode normal (OC), car il peut envoyer son RTS directement dans la bonne direction. Ce mode, appelé mode étendu (EC), utilise le faisceau DH (directionnel avec fort gain) afin d'atteindre le récepteur qui lui écoute en mode OB (voir figure 2.14b). Une fois le RTS reçu avec les coordonnées de l'émetteur, le récepteur peut pointer son antenne dans la direction de celui-ci et la communication utilisera les faisceaux DM (voir figure 2.14d). Ces faisceaux sont utilisés pour le CTS, les données et l'ACK.

Cependant, afin de fournir la position géographique des nœuds du réseau, SWAMP nécessite que chaque nœud soit équipé d'un système de positionnement GPS<sup>43</sup>.

## 2.5 Conclusion

Dans ce chapitre, nous avons présenté le principe d'une pile de protocoles de communication ainsi que différentes couches MAC présentes dans la littérature. Néanmoins, ces différentes couches nécessitent soit du matériel spécifique pour la géolocalisation (GPS), soit des antennes spécifiques et encombrantes (ESPAR), ou bien ne précisent pas comment sont récupérées les coordonnées géographiques.

Notre étude se focalise sur une couche MAC ne demandant pas de système de positionnement précis tout en utilisant une antenne directionnelle très petite (présentée dans la section 1.2).

Nous avons donc développé une pile protocolaire dédiée à ce système présenté dans le chapitre suivant ainsi qu'une couche MAC qui sera détaillée dans le chapitre 4.

---

43. GPS : *Global Positioning System*

## Troisième chapitre

---

# Architecture de la pile protocolaire pour antennes à faisceaux agiles

### Sommaire

---

<b>3.1</b>	<b>Introduction</b>	<b>34</b>
<b>3.2</b>	<b>Couche physique</b>	<b>34</b>
3.2.1	Antennes multi-secteur avec temps de commutation	34
3.2.2	Canaux radio	35
<b>3.3</b>	<b>Découverte du voisinage</b>	<b>36</b>
3.3.1	Protocoles existants	36
3.3.2	Proposition de protocole	36
<b>3.4</b>	<b>Proposition de pile protocolaire efficace</b>	<b>38</b>
3.4.1	Architecture globale	38
3.4.2	Couche liaison de données	39
3.4.2.1	Découverte de voisinage - ( <i>neighborhood discovery</i> )	39
3.4.2.2	Bloc de répartition - ( <i>dispatcher</i> )	41
3.4.2.3	Bloc de contrôle - ( <i>controller</i> )	41
3.4.2.4	Bloc d'ordonnancement - ( <i>scheduler</i> )	41
<b>3.5</b>	<b>Ordonnancement</b>	<b>42</b>
3.5.1	Choix du secteur	42
3.5.2	Événements extérieurs	43
<b>3.6</b>	<b>Conclusion</b>	<b>43</b>

---

### 3.1 Introduction

Les protocoles majoritairement utilisés à l'heure actuelle ne sont généralement pas prévus pour fonctionner avec des antennes sectorielles. Le chapitre précédent a présenté quelques protocoles proposés par la littérature et adaptés à ces antennes, mais ceux-ci nécessitent généralement des équipements spécifiques comme des antennes spéciales (ESPAR<sup>1</sup> [CHIO03] qui est volumineuse) ou l'emploi de GPS<sup>2</sup> sur chaque nœud (comme SWAMP<sup>3</sup> [TNW04, TNT<sup>+</sup>06, WMB<sup>+</sup>07]).

Notre étude se base sur une antenne sectorielle simple à commutation de faisceau. Celle-ci a pour désavantage d'induire un délai à chaque changement de direction. Nous allons détailler la couche physique que nous proposons dans la section 3.2, en prenant en compte l'impact de ce temps de commutation sur les communications réseau dans la sous-section 3.2.1. Finalement, nous détaillerons dans la section 3.4 la pile protocolaire que nous proposons.

### 3.2 Couche physique

#### 3.2.1 Antennes multi-secteur avec temps de commutation

L'antenne sur laquelle se base notre étude a été développée par l'IEMN<sup>4</sup> dans le cadre du projet CPER<sup>5</sup> LOMC<sup>6</sup>. Elle est composée de six antennes individuelles qui peuvent être activées une à une. Il est ainsi possible de sélectionner une direction d'émission ou de réception privilégiée parmi six. Chaque antenne couvre un secteur (appelé faisceau) de 60 degrés. L'ensemble des faisceaux couvre donc 360 degrés. Chaque antenne est commutée grâce à un mécanisme en MEMS<sup>7</sup> (voir figure 3.1). Ces mécanismes (un par antenne) sont des contacteurs qui, sous l'effet d'une tension électrique, relient mécaniquement l'émetteur/récepteur avec une des antennes. De ce fait, ils ne consomment que lorsqu'ils sont actifs.

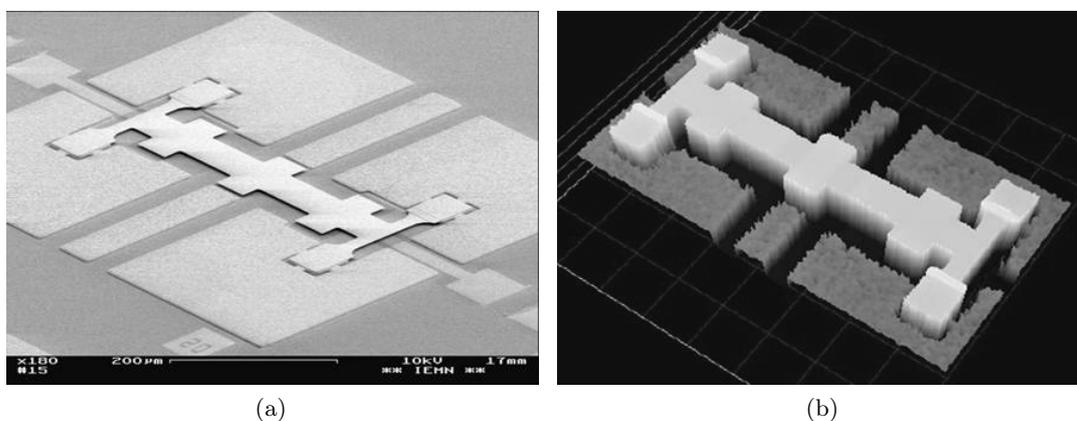


FIGURE 3.1 – Commutateur MEMS

Cette consommation est directement proportionnelle à la tension appliquée au commutateur. Plus la tension est élevée, plus le commutateur s'actionnera rapidement, mais plus celui-ci

1. ESPAR : *Electronically Steerable Parasitic Array Radiator* – Antenne orientable électroniquement grâce à un ensemble d'éléments parasites

2. GPS : *Global Positioning System*

3. SWAMP : *Smart Antennas Based Wider-range Access MAC Protocol*

4. IEMN : Institut d'Électronique, de Microélectronique et de Nanotechnologie [IEM]

5. CPER : Contrat de projets État-région

6. LOMC : Liaison de Proximité Haut Débit entre Objets Mobiles Communicants

7. MEMS : *Microelectromechanical systems*

consommara. Il est donc nécessaire de consentir à un compromis sur le temps de réponse du commutateur si l'on souhaite maîtriser l'énergie consommée.

Ce temps de réponse, avoisinant  $1ms$ , rend incompatibles les protocoles présentés dans le chapitre précédent. En effet ceux-ci considèrent que le coût de la commutation est faible (en terme de temps) et proposent par exemple d'envoyer un RTS<sup>8</sup> dans chaque direction (et un CTS<sup>9</sup> pour certains) avant toute transmission. Avec notre antenne, ces protocoles nécessitent donc au minimum  $10ms$  avant tout envoi de données pour les paquets de contrôle, réduisant ainsi la bande passante utile à moins de 100 paquets de données par seconde soit moins de  $100\text{ ko/s}$  (avec des paquets de  $1ko$ ).

Afin de pouvoir utiliser ce genre d'antennes, il est donc nécessaire de minimiser au maximum le nombre de commutations afin de réduire les pertes de débit. Pour ce faire, la couche liaison de données du modèle OSI (et particulièrement la sous-couche MAC<sup>10</sup>) doit être adaptée. Les modifications que nous proposons à la couche liaison de données sont détaillées dans la section 3.4 et nous détaillerons la couche MAC dans le chapitre suivant.

### 3.2.2 Canaux radio

Le canal radio que nous avons choisi d'utiliser correspond à la bande de fréquence de 60 GHz. Ce canal a plusieurs avantages comme la possibilité d'utiliser une large bande de fréquence (ici 4 GHz) et donc de profiter de débits élevés. En plus, comme les ondes radio à 60 GHz ne traversent pas les murs, ce canal est adapté pour des transmissions intra-bâtiment. D'un point de vue sécurité, il est donc très difficile de s'introduire sur un tel réseau sans fil depuis l'extérieur.

Comme indiqué dans la sous-section 3.2.1, l'antenne que nous utilisons est exclusivement sectorielle. Il n'est possible d'envoyer des données que dans une seule direction, tout comme la réception ne s'effectue qu'en provenance d'une seule direction également. Cela permet d'avoir un gain plus élevé et de diminuer la consommation énergétique de la transmission (même principe que les faisceaux DL et DM de SWAMP, voir figure 2.14c et figure 2.14d page 31).

Néanmoins, cette configuration n'est pas du tout adaptée pour effectuer de la signalisation sur le réseau. Si l'on souhaite prévenir un voisin d'une transmission dans sa direction, il faut que celui-ci oriente son antenne dans notre direction avant de pouvoir recevoir le message. Chose qu'il ne peut pas faire sans don de clairvoyance. Nous avons donc décidé de rajouter une antenne omnidirectionnelle sur chaque nœud afin qu'il puisse recevoir les données de signalisation quelle que soit leur provenance. Dans cette configuration, l'émetteur utilise son antenne directionnelle et le destinataire recevra le message grâce à son antenne omni-directionnelle. On se retrouve dans la même configuration que le mode DH-OB de SWAMP (voir figure 2.14b page 31).

Afin de ne pas occasionner de collisions lors des transferts, nous avons décidé de découper le canal radio en deux bandes de fréquences distinctes. Le premier canal dit « bas débit » est essentiellement utilisé pour la signalisation, la découverte de voisinage et la transmission de petites quantités d'informations. Il utilise l'antenne sectorielle pour l'émission dans un secteur donné et tous les nœuds présents dans ce secteur recevront le message via leur antenne omnidirectionnelle. Ce canal ne devant pas supporter de forts trafics et étant surtout utilisé pour la signalisation (une perte de paquet n'est pas problématique), l'accès au canal est assuré par Pure ALOHA (voir le paragraphe dans la sous-section 2.3.2, page 23) et l'acquiescement des paquets est optionnel. La modulation radio utilisée par ce canal est l'ASK<sup>11</sup> avec une fréquence porteuse de 58 GHz.

Le second canal, dit « haut débit », utilise l'antenne sectorielle tant en émission qu'en réception. Il est donc nécessaire que le récepteur ait orienté son antenne afin de recevoir les

---

8. RTS : *Request to Send*

9. CTS : *Clear to Send*

10. MAC : *Media Access Control* – Contrôle d'accès au support

11. ASK : *Amplitude-Shift Keying* – modulation en amplitude

messages. Ceci implique un échange préalable entre l'émetteur et le récepteur afin de s'assurer qu'il puisse recevoir les données. Ce surcoût est compensé par un débit beaucoup plus élevé et un risque de collision bien inférieur grâce à l'utilisation de l'antenne sectorielle (on reçoit moins d'informations) et au codage canal employé. En effet, nous utilisons sur ce canal une modulation UWB-PPM<sup>12</sup> [DBB<sup>+</sup>05, DLRR08] avec TDMA<sup>13</sup> (voir le paragraphe dans la sous-section 2.3.1) pour l'accès au canal. L'UWB-PPM, de par sa large bande, nous permet d'avoir des débits élevés et une certaine immunité aux bruits. De plus, la modulation en position d'impulsion associée au *Time hopping* du TDMA permet de réduire la probabilité de collision et de construire des détecteurs de faible complexité [FZ05]. Néanmoins, le principe de PPM nécessite une forte synchronisation entre l'émetteur et le récepteur, impliquant un préambule de synchronisation lors du début de la transmission.

Lors de l'envoi de données, il faut donc prendre en compte tous les avantages et inconvénients des deux canaux de communication afin de choisir celui que l'on va utiliser. Ce choix appartient à la couche MAC qui sera détaillée dans le chapitre suivant.

### 3.3 Découverte du voisinage

Dans tout réseau, mais essentiellement dans les réseaux sans fil, il est nécessaire de savoir si le nœud que l'on souhaite joindre est à portée (on peut lui envoyer directement un message) où s'il faut passer par un autre nœud pour le joindre. C'est le rôle du service de découverte du voisinage. Il a pour charge de dresser la liste des nœuds connus avec lesquels on est capable de communiquer. Mais, si l'on utilise des antennes sectorielles, il est en plus primordial de connaître dans quelle direction se trouvent les nœuds afin de sélectionner la bonne antenne. Nous allons présenter rapidement les deux grandes familles de protocoles de découvertes puis notre approche et notre solution à ce problème.

#### 3.3.1 Protocoles existants

Deux grandes familles existent lorsque l'on parle de protocoles cherchant à obtenir une information (un secteur, une route, ...) :

- Protocole réactif : lorsque l'on cherche à obtenir l'information au moment où on en a besoin.
- Protocole proactif : lorsque l'information est obtenue de manière automatique afin qu'elle soit disponible au moment où on en a besoin.

Chaque famille a ses avantages et ses inconvénients. Dans notre cas de figure, un protocole réactif ne peut pas être exploitable. Il faudrait par exemple envoyer un message dans chaque secteur avant chaque envoi afin de savoir où se trouve le destinataire. Cet envoi impliquerait autant de commutations qu'il y a de secteurs, alors que l'on souhaite les éviter au maximum. Il faut donc regarder du côté des protocoles proactifs. Ces protocoles impliquent l'envoi régulier de messages de contrôle afin que les nœuds voisins apprennent l'existence du nœud émetteur. La littérature propose quelques protocoles de découverte de voisinage (voir la sous-section 2.2.4). Néanmoins, ces protocoles ne sont prévus que pour découvrir un voisinage, autrement dit la liste des nœuds avec lesquels un nœud est capable de communiquer. Or il nous est nécessaire de connaître en plus la direction dans laquelle se trouve le nœud afin de savoir dans quel secteur particulier envoyer le message.

12. UWB-PPM : *Ultra WideBand - Pulse-Position Modulation* – Modulation radio à large bande avec codage par position d'impulsions

13. TDMA : *Time Division Multiple Access* – Accès multiple à répartition dans le temps

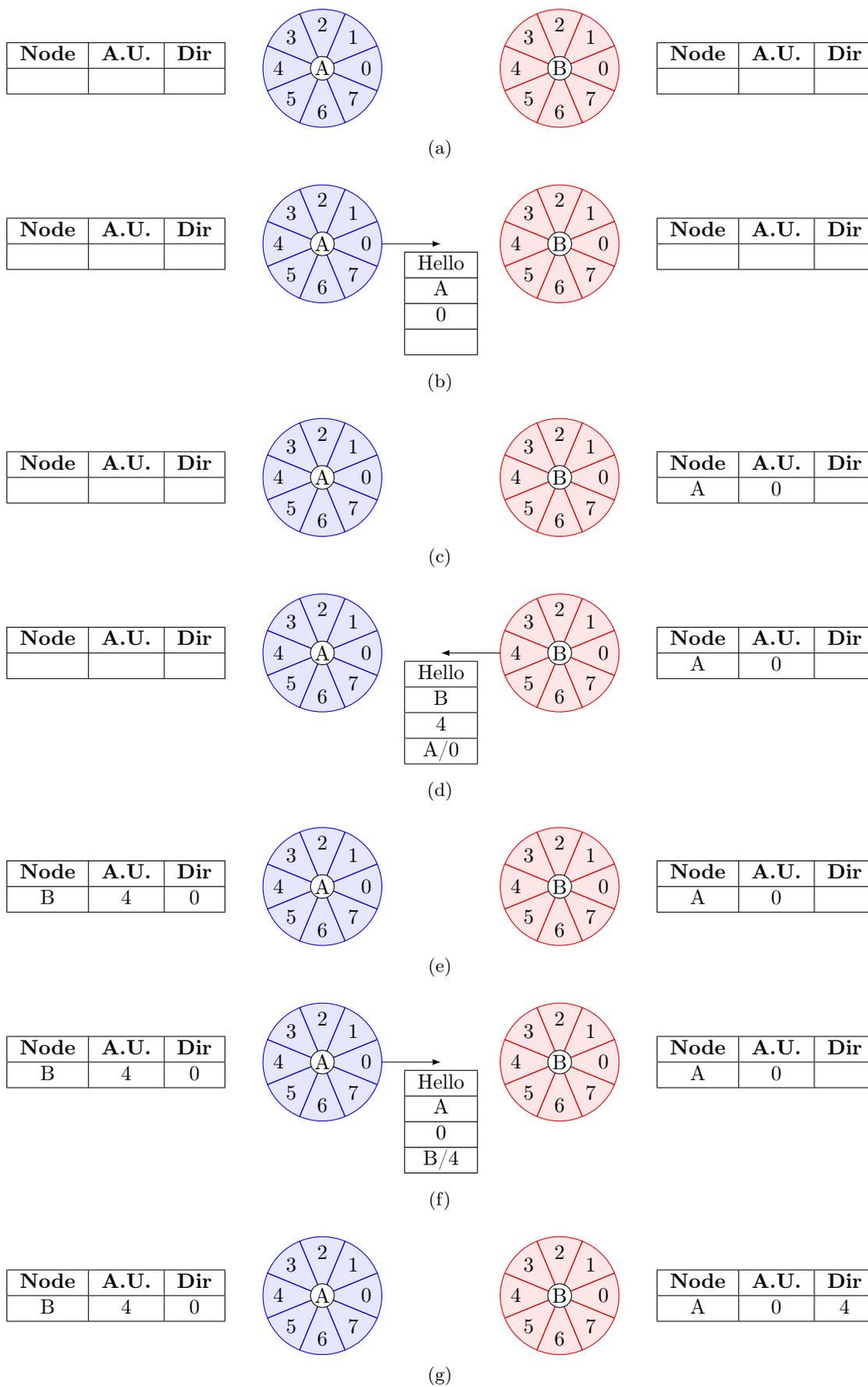


FIGURE 3.2 – Scénario de découverte de voisinage

### 3.3.2 Proposition de protocole

La première approche permettant de savoir dans quel secteur se trouve un nœud serait d'écouter les messages « Hello » de ses voisins et de pouvoir déterminer de quelle direction ces messages proviennent. Malheureusement, notre antenne ne nous permet pas de connaître cette direction.

Si nous ne pouvons pas connaître la direction d'un message, un autre nœud ne peut pas dire non plus dans quelle direction il nous a entendu. Par contre, si l'on rajoute dans nos messages « Hello » l'antenne avec laquelle ce message est émis, nos voisins peuvent nous renvoyer l'information. On saura ainsi qui nous a entendu et dans quelle direction il faut envoyer un message afin qu'ils nous reçoivent.

Il est par contre impensable que chaque voisin réponde à chaque message « Hello » reçu. Avec un réseau un peu dense, il y aurait beaucoup trop de réponses à envoyer. De plus, pour pouvoir répondre il peut être nécessaire de changer la direction de son antenne et donc réaliser une commutation que l'on souhaite éviter. On sait néanmoins que ces voisins vont envoyer régulièrement un message « Hello » dans notre direction. S'ils rajoutent dans leur message « Hello » la liste des nœuds qu'ils ont entendus ainsi que l'information de l'antenne utilisée par ces nœuds lorsqu'ils ont envoyé leur message « Hello », on sera capable de savoir à moindre coût la direction de nos voisins.

Il est donc nécessaire d'avoir une découverte de voisinage proactive afin d'avoir déjà l'information lorsqu'on en a besoin. Au tout début, chaque nœud dispose d'une table de voisinage vide (voir figure 3.2a), *A.U.* correspondant à l'antenne utilisée par le nœud *Node* lorsqu'il a envoyé son message « Hello » et *Dir.* correspondant à la direction à utiliser pour envoyer un message au nœud *Node* si elle est connue. Chaque nœud va envoyer régulièrement un message « Hello » contenant son identifiant et l'antenne qu'il utilise pour envoyer ce message ainsi que l'ensemble des couples identifiant-antenne provenant des messages précédemment reçus (voir figure 3.2b, figure 3.2d et figure 3.2f). Puis lorsqu'ils vont recevoir un message « Hello », ils vont stocker le couple identifiant-antenne utilisé par l'émetteur du message (voir figure 3.2c et figure 3.2e) et, si leur identifiant est présent dans la liste des voisins du message, ils vont stocker dans leur table de voisinage l'antenne à utiliser pour communiquer avec ce nœud (voir figure 3.2e et figure 3.2g).

Grâce à cet échange bi-directionnel, chaque nœud sera en mesure de connaître la direction d'un éventuel voisin lorsqu'il aura un message à lui envoyer.

## 3.4 Proposition de pile protocolaire efficace

Nous proposons dans ce document une nouvelle couche de liaison adaptée aux antennes décrites au début de ce chapitre. Nous allons décrire les principales structures de cette nouvelle couche puis le chapitre 4 détaillera la méthode d'accès au canal.

### 3.4.1 Architecture globale

L'architecture de notre pile protocolaire s'appuie sur le modèle OSI<sup>14</sup> (voir sous-section 2.2.1) et nous nous intéressons plus particulièrement aux couches basses. Nous avons vu dans la section 3.2 la couche physique de notre modèle avec ses deux canaux radio et les différents secteurs d'émission. La couche physique propose à la couche liaison de données six secteurs de communication avec deux modes : « bas débit » et « haut débit ».

Pour la couche liaison de données, deux architectures sont possibles :

- Proposer autant d'interfaces qu'il y a de secteurs à la couche réseau (voir figure 3.3a).

---

14. OSI : *Open Systems Interconnection*

- Ne proposer qu’une interface à la couche réseau (voir figure 3.3b).

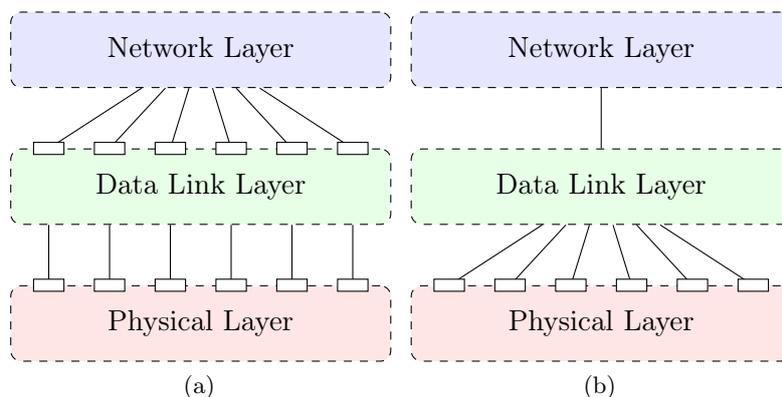


FIGURE 3.3 – Différentes approches possibles pour la couche liaison de données

Le fait de proposer plusieurs interfaces à la couche supérieure permet à la couche liaison de données de déporter la gestion des secteurs aux algorithmes de routage de la couche réseau. Le nœud dispose ainsi de voisins sur chacune de ses interfaces et peut directement dialoguer avec eux comme s’il avait plusieurs cartes réseaux. Néanmoins, la couche liaison de données a aussi besoin de connaître la position des voisins pour pouvoir répondre par exemple à un RTS qui arrive via l’antenne omni-directionnelle. On peut dans ce cas envisager une communication avec la couche supérieure, mais ceci implique de modifier aussi cette couche et casse le modèle OSI. Une autre solution serait de gérer une liste des voisins dans la couche liaison de données, mais ceci rend nul l’intérêt de le déléguer à la couche réseau. Il est donc plus efficace de ne proposer qu’une seule interface à la couche réseau et de s’occuper de la gestion des secteurs directement dans la couche liaison de données.

L’architecture globale de notre pile protocolaire ne spécifie donc que la couche physique et la couche liaison de données du modèle OSI et reste ainsi compatible avec n’importe quelle couche réseau normalement implémentée dans chaque système d’exploitation (voir figure 3.4).

### 3.4.2 Couche liaison de données

Nous avons vu au début du chapitre qu’une commutation d’antenne est pénalisante en temps. Il faut donc essayer de minimiser ce nombre de commutations. Pour cela, les données à envoyer dans une même direction sont regroupées afin d’être envoyées successivement. Afin de stocker ces données, nous avons besoin de files d’attente. Nous avons choisi de créer une file d’attente par secteur radio afin de faciliter l’accès de l’ordonnanceur aux paquets. L’architecture de la couche de liaison est donc centrée autour des files d’attente (voir figure 3.4). Ensuite l’ordonnanceur décide en fonction de l’état des files d’attente le secteur sur lequel il est nécessaire d’envoyer des données. On peut ainsi rajouter de l’intelligence au niveau de la commutation et réduire son nombre.

#### 3.4.2.1 Découverte de voisinage - (*neighborhood discovery*)

La couche liaison de données propose une interface simple à la couche réseau. C’est donc à la couche liaison de données de connaître la position éventuelle de chaque nœud afin de pouvoir lui envoyer un message. Il revient au service de découverte de voisinage de remplir cette tâche.

Nous avons détaillé dans la sous-section 3.3.2 le fonctionnement de notre algorithme de découverte de voisinage. Afin de ne pas solliciter de commutation lors de l’envoi des messages « Hello », ceux-ci sont rajoutés dans les files d’attente comme des messages normaux. Ainsi,

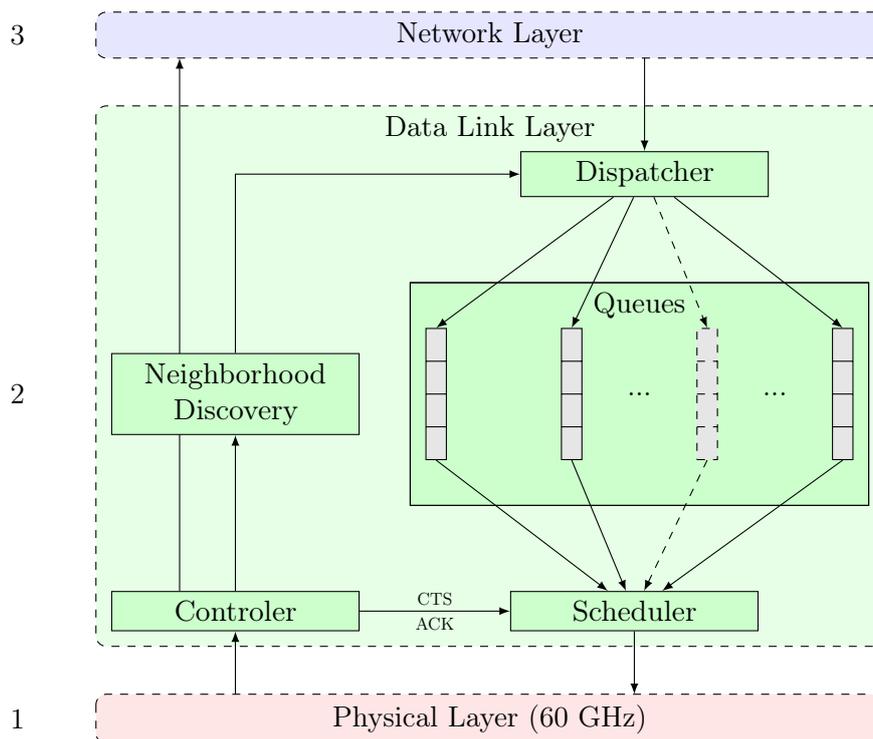


FIGURE 3.4 – Représentation OSI de la pile protocolaire

le message « Hello » d'un secteur ne sera envoyé que lorsque l'ordonnanceur sera en train de s'occuper de la file d'attente correspondant à ce secteur.

Chaque message « Hello » contient l'identifiant du nœud émetteur, le secteur qu'il utilise pour l'envoi ainsi qu'une partie de sa table de voisinage (voir figure 4.6). Mais le service de découverte de voisinage peut permettre aussi de déterminer la puissance d'émission minimale requise pour joindre un nœud. Pour ce faire, chaque nœud peut envoyer ses messages « Hello » à des puissances différentes et les voisins vont reporter lors de leur envoi la plus petite puissance utilisée des messages « Hello » reçus. Si le réseau est assez stable, il est ainsi possible au bout de quelques tours de messages « Hello » de connaître la puissance nécessaire pour communiquer avec chaque voisin.

Si un nœud s'éloigne, il peut être nécessaire d'attendre quelques secondes afin de pouvoir déterminer la nouvelle puissance requise pour communiquer avec celui-ci. Afin d'éviter ce long temps d'attente, la couche MAC n'utilisera cette puissance indicative que lors du premier envoi de message. En cas d'échec, elle pourra utiliser une puissance plus élevée afin d'augmenter les chances de réussite.

### 3.4.2.2 Bloc de répartition - (*dispatcher*)

Grâce au service de découverte de voisinage et à sa table de voisinage, nous sommes capables d'associer pour chaque potentiel voisin un secteur d'émission. Le rôle de ce bloc est d'attribuer pour chaque paquet à envoyer le secteur correspondant.

Pour ce faire, nous avons décidé de créer une file d'attente par secteur d'émission afin de simplifier l'accès aux paquets à l'ordonnanceur. Lorsqu'un paquet doit être émis, il passe d'abord par le répartiteur qui affecte le paquet à la bonne file d'attente. Les paquets à envoyer peuvent provenir soit de la couche supérieure (couche réseau) soit du service de découverte de voisinage. Si le destinataire n'est pas connu à ce stade, le paquet est supprimé.

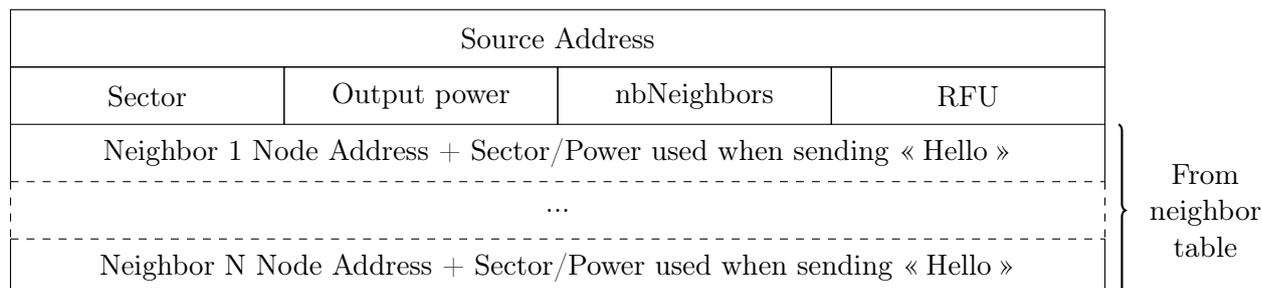


FIGURE 3.5 – Structure d’un paquet « Hello »

Chaque file contient donc les paquets à destination du secteur considéré et sont rangés dans leur ordre d’envoi (du plus vieux au plus récent).

### 3.4.2.3 Bloc de contrôle - (*controler*)

Le principe de ce bloc est de réagir aux paquets reçus par l’interface radio. Plusieurs cas sont possibles selon le type de paquet réceptionné :

- Paquet de données : Il est directement envoyé à la couche supérieure s’il nous est destiné. Si un acquittement est nécessaire, l’ordonnanceur est informé qu’il doit envoyer un ACK <sup>15</sup>.
- Paquet de découverte de voisinage (« paquet Hello ») : Il est dirigé vers le service de découverte de voisinage.
- Paquet de contrôle : En fonction du type de paquet de contrôle reçu, l’ordonnanceur répondra de manière différente :
  - Si le bloc de contrôle reçoit un RTS, il indique à l’ordonnanceur qu’il faut envoyer un CTS.
  - S’il reçoit un CTS, il lui indique qu’il peut envoyer les données.
  - S’il reçoit un ACK, il lui indique que les données ont été correctement reçues par le destinataire.

L’ordonnanceur connaît donc l’état actuel de la couche MAC et sait s’il doit gérer une communication entrante ou s’il peut vider l’une des files d’attente.

### 3.4.2.4 Bloc d’ordonnancement - (*scheduler*)

Ce bloc est le plus important de la couche. Le but de ce bloc est de réduire autant que possible le nombre de commutations de l’antenne afin de maximiser la bande passante disponible tout en gardant une certaine équité dans les flux de communication. Il est essentiel de ne pas privilégier un seul secteur (cas où l’on n’aurait aucun changement de secteur et donc aucun temps perdu par les commutations), ni de gérer les paquets à envoyer en FIFO <sup>16</sup>. Dans ce dernier cas, on diminue la latence dans le réseau (les paquets ne restent pas longtemps dans les files d’attente car le premier arrivé est le premier parti) mais cela induit un grand nombre de commutations et diminue donc la bande passante utile.

Par conséquent, il est nécessaire d’avoir un algorithme dédié prenant en compte les spécificités de l’antenne. Son fonctionnement est détaillé dans la section suivante et les mécanismes d’accès au canal seront étudiés dans le chapitre suivant.

15. ACK : *Acknowledgment code* – Acquittement des données

16. FIFO : *First In, first Out* – « Premier arrivé, premier servi »

### 3.5 Ordonnement

Chaque file d'attente contient les paquets destinés à un secteur d'émission. Le but de l'ordonneur est de vider ces files d'attente le plus rapidement possible en réduisant au maximum le nombre de commutations réalisées par l'antenne, ceci afin d'assurer la bande passante utile maximale pour le nœud.

Soient  $s$  et  $B_{SA}$  respectivement la taille d'un paquet et la bande passante théorique du canal radio. Le temps de latence de l'interface radio est noté  $l$ . Le temps d'émission d'un paquet noté  $t_{em}$  est donné par la formule suivante :

$$t_{em} = l + \frac{s}{B_{SA}}$$

Dans le cas d'utilisation d'un algorithme du type *Round Robin*, l'antenne doit changer de secteur à chaque paquet. Si le temps de commutation est noté  $c$ , le temps d'émission devient :

$$t_{em} = l + \frac{s}{B_{SA}} + c$$

Une autre possibilité, si l'on souhaite minimiser le délai, est d'envoyer le plus vieux paquet en premier (méthode FIFO). Soit  $f$  le nombre de faisceaux du nœud. Si les données sont uniformément réparties entre tous les secteurs, il y a une probabilité de  $\frac{f-1}{f}$  d'effectuer une commutation. Le temps d'émission d'un paquet devient :

$$t_{em} = l + \frac{s}{B_{SA}} + c \times \frac{f-1}{f}$$

Ces deux solutions sont-elles satisfaisantes ? Avec l'algorithme de type *Round Robin*, si l'on prend par exemple un paquet de 1.5ko, un débit d'au moins 10Mbps et un temps de commutation de 1ms, le temps de commutation correspond au moins à la moitié du temps d'émission d'un paquet. Le débit utile est donc au moins divisé par deux. Avec la méthode FIFO, *a priori* un peu plus favorable, le temps de commutation pour six faisceaux représente dans le pire des cas 41% du temps d'émission du paquet. Le débit est donc réduit de 46%.

Ces deux solutions ne sont donc pas intéressantes. L'ordonneur ne doit pas changer de file d'attente (et donc de secteur) à chaque paquet mais il ne doit pas non plus rester trop longtemps sur un secteur car cela va induire un temps de latence pour les paquets présents dans les autres files. Par conséquent, nous avons développé un algorithme propre permettant de gérer efficacement les antennes sectorielles à temps de commutation. L'ordonneur doit prendre en compte plusieurs paramètres :

- Le temps passé sur le secteur courant.
- La longueur des files d'attente.
- La date du plus vieux paquet présent dans les files d'attente.
- La nécessité de répondre aux événements extérieurs.

Il a été démontré dans [GHMSR06] qu'il existe une valeur optimale en fonction de la charge permettant de maximiser la bande passante et de minimiser le délai dans le réseau. Nous faisons donc en sorte que l'ordonneur se maintienne sur un secteur tant qu'il reste des paquets à envoyer et que le temps passé sur ce secteur ne dépasse pas la valeur définie dans TBW<sup>17</sup>. Lorsqu'une de ces deux conditions est remplie, l'ordonneur va élire un nouveau secteur.

#### 3.5.1 Choix du secteur

L'ordonneur doit choisir un secteur sur lequel il va envoyer les messages présents dans une file d'attente. Afin de diminuer le délai dans le réseau, le secteur choisi sera le secteur contenant le paquet le plus vieux. Les paquets étant rangés par ordre d'arrivée dans les files d'attente (FIFO),

17. TBW : *Temporized Beam Window*

il est simple de regarder le premier paquet de chaque pile afin de déterminer celle qui contient le paquet le plus vieux.

Une fois le secteur sélectionné, l'ordonnanceur enverra tous les paquets en conservant leur ordre d'arrivée pendant le temps préconisé par TBW en fonction de la charge actuelle.

### 3.5.2 Événements extérieurs

Un autre paramètre à prendre en compte dans l'algorithme de l'ordonnanceur est l'arrivée de paquets depuis l'interface radio. Ceux-ci peuvent être des paquets de contrôle auxquels il faut répondre rapidement (comme les paquets RTS). Nous avons décidé de répondre à un RTS dans deux cas :

- s'il n'y a plus de paquet à envoyer.
- si le plus vieux paquet à envoyer n'est pas « trop vieux ».

En effet, afin d'éviter qu'un paquet n'arrive trop en retard à sa destination, qu'il soit supprimé et implique donc une réémission, nous avons décidé de ne pas répondre à un RTS si le paquet le plus vieux à envoyer est présent depuis trop longtemps dans les files d'attente. Cela nous laisse le temps d'envoyer le paquet puis de répondre au RTS que l'émetteur réémettra plus tard.

## 3.6 Conclusion

L'antenne sectorielle que nous utilisons induit un délai à chaque commutation de secteur. Ce délai empêche d'utiliser les protocoles usuels de la littérature, car ceux-ci ne prennent pas en compte ce phénomène. Ainsi en réalisant un grand nombre de changements de secteurs, le temps d'émission d'un paquet est grandement augmenté, ce qui diminue le nombre de paquets qu'un nœud peut envoyer.

Dans ce chapitre, nous avons présenté la pile protocolaire que nous avons développée. Celle-ci a été conçue afin de prendre en compte le temps de latence des antennes sectorielles en minimisant le nombre de commutations d'antenne nécessaire. Cela permet de maximiser le débit utile d'un nœud. En adaptant la découverte de voisinage, il est possible de facilement obtenir la direction d'un nœud sans disposer de matériel spécifique comme un récepteur GPS.

Nous allons décrire dans le chapitre suivant les mécanismes d'accès au canal radio pour le canal « bas débit » et le canal « haut débit ».



## Quatrième chapitre

---

# Couche MAC et agrégation de paquets

### Sommaire

---

<b>4.1</b>	<b>Introduction</b>	<b>46</b>
<b>4.2</b>	<b>Détails de la couche MAC</b>	<b>46</b>
4.2.1	Automate de la couche MAC	46
4.2.2	Trames de contrôle	47
4.2.2.1	Trame de base	47
4.2.2.2	RTS	47
4.2.2.3	CTS	48
4.2.2.4	ACK	48
4.2.2.5	Hello	48
4.2.3	Trames de données	49
<b>4.3</b>	<b>Canal bas débit</b>	<b>49</b>
4.3.1	Trames de contrôle	49
4.3.2	Envoi de données	50
<b>4.4</b>	<b>Canal haut débit</b>	<b>50</b>
4.4.1	Synchronisation	50
4.4.2	Taille des trames	51
4.4.3	Envoi d'un paquet en utilisant le canal « haut débit »	53
4.4.3.1	Envoi classique	53
4.4.3.2	Gestion des échecs	54
4.4.3.3	Amélioration	54
<b>4.5</b>	<b>Algorithme de la couche MAC</b>	<b>55</b>
4.5.1	Mode « bas débit »	55
4.5.2	Mode « haut débit »	57
<b>4.6</b>	<b>Agrégation multi-utilisateurs</b>	<b>58</b>
<b>4.7</b>	<b>Conclusion</b>	<b>60</b>

---

## 4.1 Introduction

Après avoir décrit la pile protocolaire que nous proposons, nous allons détailler les mécanismes d'accès au canal radio. La couche MAC<sup>1</sup> est une sous-couche de la couche liaison de données en charge de réguler les émissions sur un support donné, en particulier lorsque plusieurs stations indépendantes sont susceptibles d'émettre à tout moment sur le même support. Il revient également à cette sous-couche de décrire le format des trames, notamment des trames de contrôle.

Dans un premier temps, nous allons détailler le fonctionnement de notre couche MAC ainsi que les différentes trames de contrôle. Bien que proche de IEEE 802.11 par l'emploi de paquets RTS<sup>2</sup>/CTS<sup>3</sup>, notre couche MAC spécifie de nouvelles informations dans ces paquets. Nous allons présenter ensuite les spécificités des deux canaux radio : « bas débit » et « haut débit ». Pour finir, nous apporterons une extension à cette couche en permettant l'envoi d'un message à plusieurs destinataires puis nous conclurons.

## 4.2 Détails de la couche MAC

### 4.2.1 Automate de la couche MAC

Notre couche MAC dispose de deux modes de fonctionnement : « bas débit » et « haut débit ». Le mode de fonctionnement par défaut est le mode « bas débit » noté *BD* dans les différentes figures de ce chapitre (figure 4.1). Si l'ordonnanceur décide d'envoyer un message en utilisant le canal « haut débit », il va quitter le mode « bas débit » le temps de traiter l'envoi. Les différents modes de la couche MAC sont nommés en fonction du type de trame que la couche MAC attend de recevoir :

- *HD<sub>CTS</sub>* : la couche MAC attend de recevoir une trame CTS.
- *HD<sub>ACK</sub>* : la couche MAC attend de recevoir un acquittement.

Une fois l'acquittement reçu, la couche MAC repassera en mode « bas débit ».

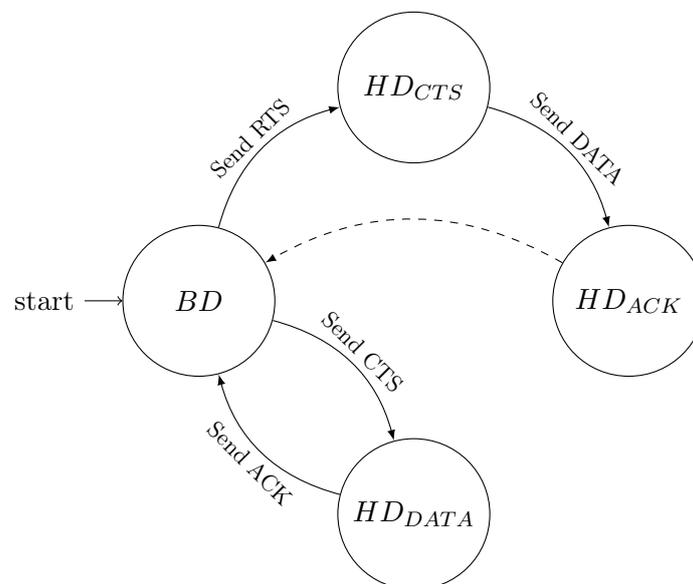


FIGURE 4.1 – Automate de la couche MAC

---

1. MAC : *Media Access Control* – Contrôle d'accès au support  
 2. RTS : *Request to Send*  
 3. CTS : *Clear to Send*

La couche MAC peut aussi quitter le mode « bas débit » si elle décide de répondre à un paquet RTS. La réponse se fait sur le canal « haut débit » par l'envoi d'un message CTS et la couche MAC passe dans l'état  $HD_{DATA}$  en attendant de recevoir les données. Une fois les données reçues, elle enverra un ACK<sup>4</sup> et repassera en mode « bas débit ».

## 4.2.2 Trames de contrôle

### 4.2.2.1 Trame de base

Toute trame contient au moins les informations suivantes : sa nature (le type de trame), les options, sa taille, l'adresse de l'expéditeur et un numéro de séquence permettant de détecter les paquets dupliqués. Le type de trame permet de faire la distinction entre RTS, CTS, ACK, trame de découverte de voisinage ou trame de données (*Basic header*, figure 4.2) et les options permettent par exemple de demander un acquittement des données. Un octet est laissé libre en vue d'ajouter des options dans le futur (RFU<sup>5</sup>).

La trame contient ensuite la partie spécifique de chaque paquet (en fonction de son type, *Extended header* sur la figure 4.2) puis les données utiles de la trame s'il y en a (trame « Hello » ou de données par exemple). Chaque trame est terminée par un code CRC<sup>6</sup> permettant de valider la trame à la réception.

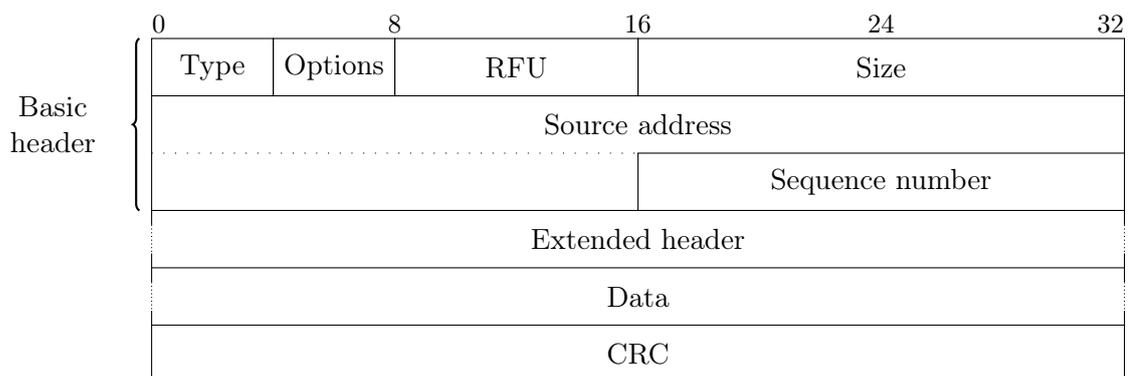


FIGURE 4.2 – Structure d'une trame

### 4.2.2.2 RTS

La trame RTS permet de prévenir un autre nœud du réseau que l'on souhaite lui envoyer des données. Elle doit donc contenir l'adresse du destinataire ainsi que la quantité de données en passe d'être envoyées (figure 4.3). Cette trame ne définit pas de données.

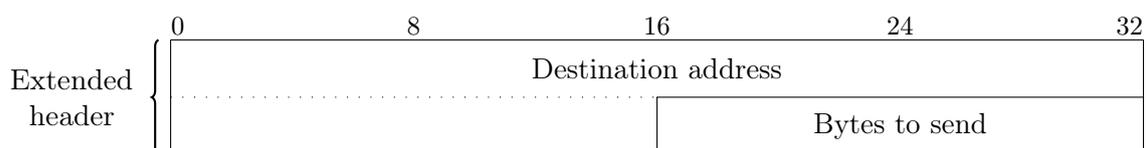


FIGURE 4.3 – Structure de l'entête étendue pour une trame RTS

4. ACK : *Acknowledgment code* – Acquittement des données

5. RFU : *Reserved for Future Use* – Réserve pour un usage futur

6. CRC : *Cyclic Redundancy Check* – Contrôle de redondance cyclique permettant de détecter les erreurs de transmission

### 4.2.2.3 CTS

Suite à la réception d'un RTS, il faut prévenir l'émetteur que l'on est prêt à recevoir les données. C'est le rôle de la trame CTS. L'émetteur n'envoyant qu'un RTS à la fois, il n'est pas nécessaire de préciser dans la trame à quel RTS correspond la réponse. La trame doit néanmoins contenir l'adresse du destinataire (qui est l'émetteur du RTS) afin qu'il sache que c'est bien à lui que l'on s'adresse (figure 4.4).

La trame précise aussi la quantité de mémoire disponible pour la réception des données. L'émetteur pourra ainsi ajuster la quantité de données qu'il souhaite envoyer afin de ne pas saturer le récepteur. Cette trame ne définit pas de données.

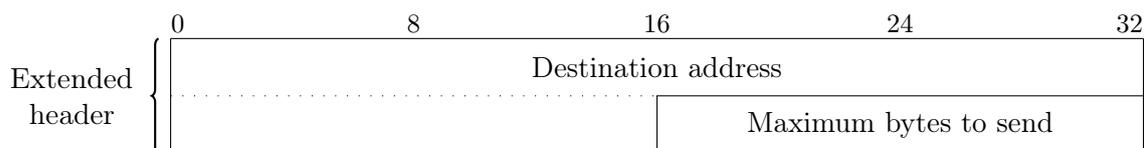


FIGURE 4.4 – Structure de l'entête étendue pour une trame CTS

### 4.2.2.4 ACK

Une fois les données reçues et vérifiées (grâce au code CRC), le nœud enverra à l'émetteur un acquittement appelé ACK. Comme pour la trame CTS, un seul paquet de données est en vol et seule l'adresse du destinataire est nécessaire (figure 4.5). Afin de respecter l'alignement sur 32 bits, 2 octets de bourrage sont insérés après l'adresse de destination. Cette trame ne définit pas de données.

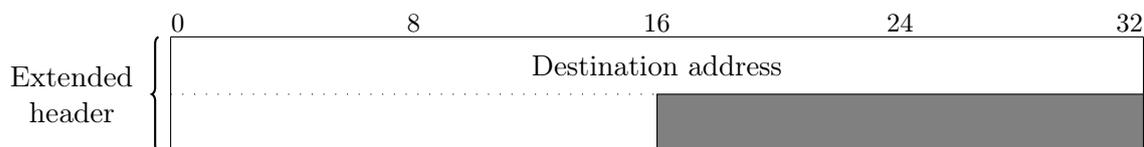


FIGURE 4.5 – Structure de l'entête étendue pour une trame ACK

### 4.2.2.5 Hello

La sous-section 3.4.2 a défini la trame de découverte de voisinage. Cette trame est envoyée régulièrement dans les différents secteurs afin de signaler notre présence et de fournir aux autres nœuds des informations de direction. La trame contient donc en premier les informations propres au nœud émetteur puis un extrait de la table de voisinage du nœud. Lorsque nos voisins recevront ce message, ils connaîtront le secteur qu'ils ont utilisé lors de l'envoi du message « Hello » que nous avons reçu. Cette trame contient aussi des informations de puissance afin de pouvoir économiser l'énergie lors de l'envoi de messages.

Dans la mesure où la trame n'a pas de destinataire désigné, étant destinée à tous les nœuds présents dans la zone de communication, elle ne spécifie pas de destinataire.

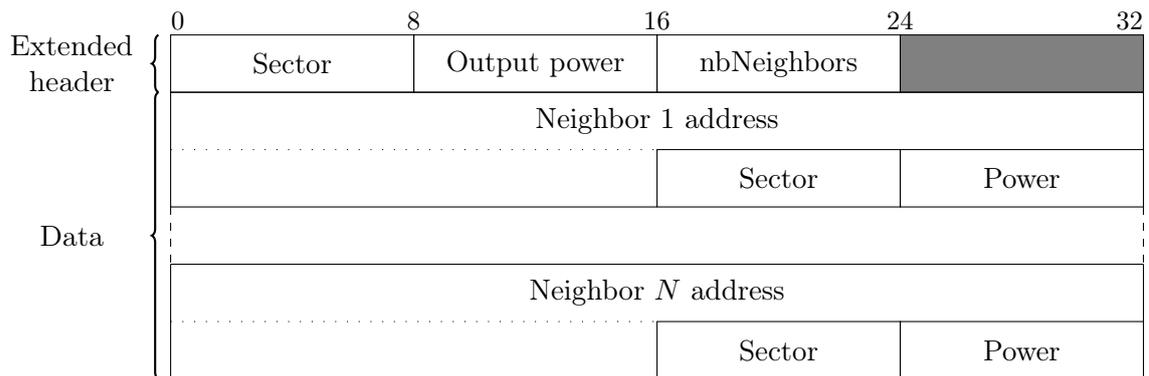


FIGURE 4.6 – Structure de la trame « Hello »

### 4.2.3 Trames de données

Le dernier type de trame que la couche MAC peut envoyer est une trame de données. Celle-ci contient les données de la couche supérieure (couche réseau) à envoyer à un destinataire. L'entête de la trame de données contient l'adresse du destinataire ainsi que la quantité de données utiles présentes dans le paquet (figure 4.7). Pour des raisons d'alignement, il est possible que soient rajoutés quelques octets après les données afin d'aligner le CRC sur 32 bits.

Dans le cadre du canal « bas débit », la taille maximale d'une trame étant de 2048 octets, la taille maximale des données utiles est de 2024 octets. Pour le canal « haut débit », la taille maximale des trames est de 32768 octets, ce qui permet d'envoyer directement 32744 octets à un nœud du réseau.

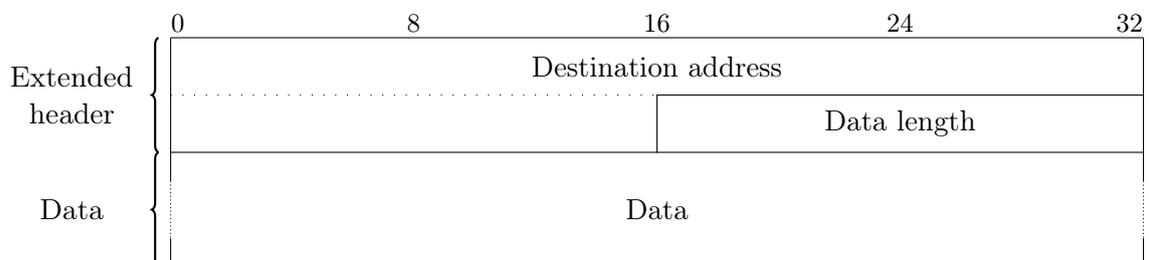


FIGURE 4.7 – Structure de la trame de données

## 4.3 Canal bas débit

### 4.3.1 Trames de contrôle

Le canal « bas débit » est le canal radio le plus simple de la pile protocolaire. Il utilise l'antenne directionnelle pour envoyer les messages, lesquels seront reçus par les nœuds du secteur via leur antenne omni-directionnelle.

Ce canal est prévu pour transporter les trames de contrôle comme les RTS et les messages « Hello ». Il est donc considéré qu'il a un faible trafic. Dans la mesure où nous utilisons des antennes directionnelles, qui amplifient le problème du terminal caché, l'émetteur ne peut pas

savoir si le destinataire est en train de recevoir un message. Si on prend l'exemple de la figure 4.8, le nœud  $S2$  est incapable de savoir si le nœud  $S1$  est en train d'envoyer un message à  $R$ .

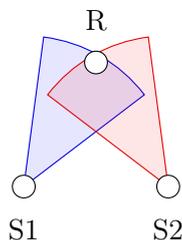


FIGURE 4.8 – Exemple de terminal caché avec des antennes sectorielles

Il est donc impossible d'utiliser des algorithmes du type CSMA<sup>7</sup> car l'information que nous donne le canal ne nous renseigne pas sur l'état du récepteur. Nous avons donc décidé d'utiliser sur ce canal la méthode d'accès au canal la plus simple : Pure ALOHA[Abr70]. En effet, sur un canal à faible trafic le risque de collision est faible et si un RTS est perdu par exemple, il sera réémis.

### 4.3.2 Envoi de données

L'utilisation du canal « haut débit » impliquant un certain surcoût, le canal « bas débit » peut aussi être utilisé pour l'envoi de faibles quantités de données. Dans ce cas, l'entête de la trame contient une option permettant de demander au destinataire l'acquittement des données. La réception sur le canal « bas débit » étant asynchrone avec l'émission (on peut recevoir à tout moment), le destinataire sera peut-être occupé lorsqu'il recevra notre trame.

Celui-ci ne pourra donc nous envoyer un ACK que lorsqu'il sera disponible. Le nœud devant rester au moins sur le même secteur pendant le temps indiqué par TBW<sup>8</sup>, le *timeout*<sup>9</sup> sera fixé à la valeur de TBW. Passé ce délai, la trame sera considérée comme perdue puis réémise.

Pendant ce temps, l'émetteur pourra continuer à envoyer des données (y compris dans d'autres secteurs) car l'envoi de la trame d'acquittement se fera aussi sur le canal « bas débit ».

## 4.4 Canal haut débit

Ce canal est le plus important de notre pile protocolaire. Le but de ce canal, comme son nom l'indique, est d'envoyer les données à très haut débit. Dans cette section, nous allons désigner  $S$  comme le nœud voulant émettre des données utiles au nœud  $R$  qui est considéré comme le destinataire.

### 4.4.1 Synchronisation

Dans ce canal, la couche physique utilise une modulation PPM<sup>10</sup>. Cette modulation envoie chaque bit à une position différente dans l'intervalle de temps en fonction de sa valeur (figure 4.9). Dans notre cas, chaque impulsion dure environ 200ps. Afin de décoder correctement le message, le

7. CSMA : *Carrier Sense Multiple Access* – Accès multiple par détection de porteuse

8. TBW : *Temporized Beam Window*

9. Paramètre réseau indiquant la durée maximum d'une transaction. S'il n'y a aucune réponse à la fin du délai imparti, les données sont considérées comme n'étant jamais arrivées au destinataire

10. PPM : *Pulse-Position Modulation*

récepteur a besoin d'être synchronisé de manière précise avec l'émetteur. Il peut ainsi déterminer la position des impulsions dans le message et retrouver les données d'origine.

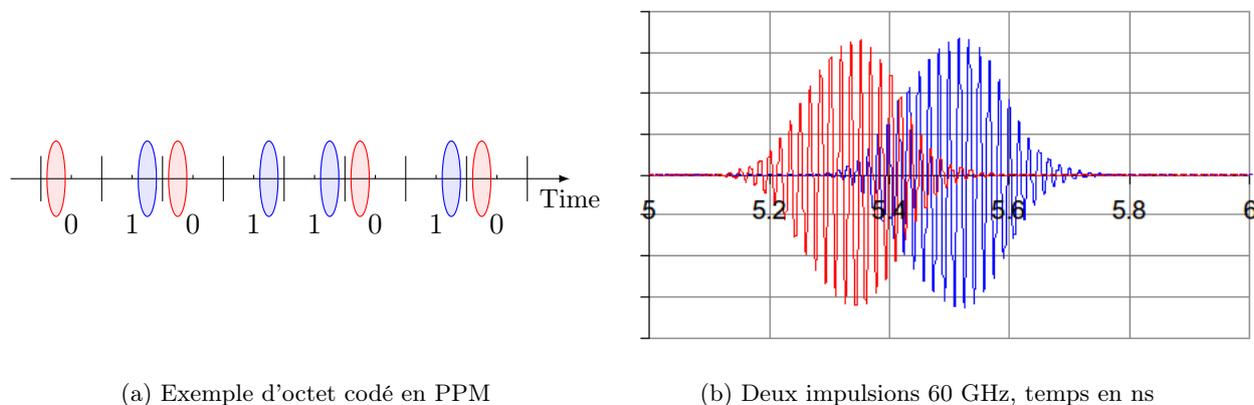


FIGURE 4.9 – Modulation PPM

La couche physique va donc rajouter devant chaque trame de la couche MAC envoyée sur le canal « haut débit » un préambule de synchronisation d'environ  $1ms$  (figure 4.10). Ce préambule permet de synchroniser l'oscillateur local du récepteur avec celui de l'émetteur. Une fois les oscillateurs synchronisés, ceux-ci maintiennent leur synchronisation suffisamment longtemps pour ne pas avoir à rajouter le préambule sur les trames de contrôle suivantes. Ainsi, il n'est adjoint qu'à la première trame envoyée sur le canal « haut débit », à savoir le CTS.

Un mot de fin ajouté à la fin de la trame indique que le message est terminé. La durée de celui-ci est négligeable comparativement au préambule de synchronisation et à l'envoi réel de la trame.

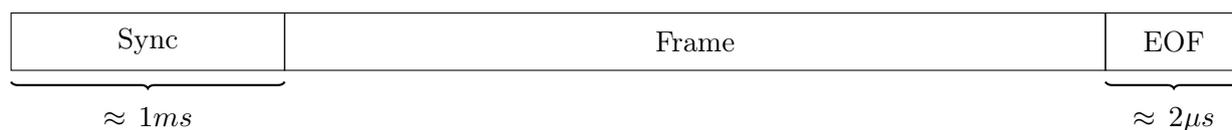


FIGURE 4.10 – Structure de la trame dans les airs sur le canal « haut débit »

#### 4.4.2 Taille des trames

Nous avons vu dans la sous-section précédente qu'un préambule de synchronisation d'environ  $1ms$  était rajouté à la trame. Si l'on rajoute le temps de commutation éventuel de  $1ms$ , on dépasse déjà le temps d'émission d'un paquet sur le canal « bas débit ». L'augmentation du débit de ce canal ne peut donc pas compenser le surcoût de la synchronisation et des trames de contrôle.

Pour diminuer ce surcoût, il est nécessaire de le mutualiser sur plusieurs paquets. Soit  $c$  et  $B_{low}$  respectivement le temps de commutation et le débit du canal « bas débit ». Sur ce canal, le temps d'émission  $t_{low}$  d'un paquet de taille  $S$  est, dans le pire des cas :

$$t_{low}(S) = c + \frac{S}{B_{low}}$$

Sur le canal « haut débit », il faut rajouter le préambule de synchronisation ( $t_{sync}$ ), le temps de commutation du récepteur, ainsi que l'envoi du RTS, CTS et ACK (respectivement  $t_{rts}$ ,  $t_{cts}$  et  $t_{ack}$ ). Le temps d'émission  $t_{high}$  d'un paquet de taille  $S$  devient :

$$t_{high}(S) = c + t_{rts} + c + t_{sync} + t_{cts} + \frac{S}{B_{high}} + t_{ack}$$

Afin d'être plus rapide que le canal « bas débit », il faut envoyer plusieurs paquets en même temps dans une même trame. Si l'on considère que l'on peut envoyer  $N$  paquets dans la même trame, il faut donc que :

$$t_{high}(N \times S) < N \times t_{low}(S)$$

Soit :

$$c + t_{rts} + c + t_{sync} + t_{cts} + \frac{N \times S}{B_{high}} + t_{ack} < N \times \left( c + \frac{S}{B_{low}} \right) \quad (4.1)$$

$$2 \times c + t_{rts} + t_{sync} + t_{cts} + t_{ack} + \frac{N \times S}{B_{high}} < N \times c + \frac{N \times S}{B_{low}} \quad (4.2)$$

$$2 \times c + t_{rts} + t_{sync} + t_{cts} + t_{ack} - N \times c < \frac{N \times S}{B_{low}} - \frac{N \times S}{B_{high}} \quad (4.3)$$

Si l'on considère  $B_{low} = \frac{B_{high}}{10}$  :

$$2 \times c + t_{rts} + t_{sync} + t_{cts} + t_{ack} - N \times c < \frac{10 \times N \times S}{B_{high}} - \frac{N \times S}{B_{high}} \quad (4.4)$$

$$2 \times c + t_{sync} + t_{rts} + t_{cts} + t_{ack} - N \times c < \frac{9 \times N \times S}{B_{high}} \quad (4.5)$$

Avec les valeurs numériques pour les différentes durées on obtient :

$$3 \times 10^{-3}s + 19 \times 10^{-6}s + 2 \times 1.9 \times 10^{-6}s - N \times 10^{-3}s < \frac{9 \times N \times S}{100 \times 10^6 \text{bits.s}^{-1}} \quad (4.6)$$

$$3.0228 \times 10^{-3}s - N \times 10^{-3}s < \frac{9 \times N \times S}{100 \times 10^6 \text{bits.s}^{-1}} \quad (4.7)$$

$$3.0228 \times 10^{-3} - N \times 10^{-3} < \frac{9 \times N \times S}{100 \times 10^6} \quad (4.8)$$

$$3.0228 - N < \frac{9 \times N \times S}{100 \times 10^3} \quad (4.9)$$

$$\frac{9 \times N \times S}{100 \times 10^3} > 3.0228 - N \quad (4.10)$$

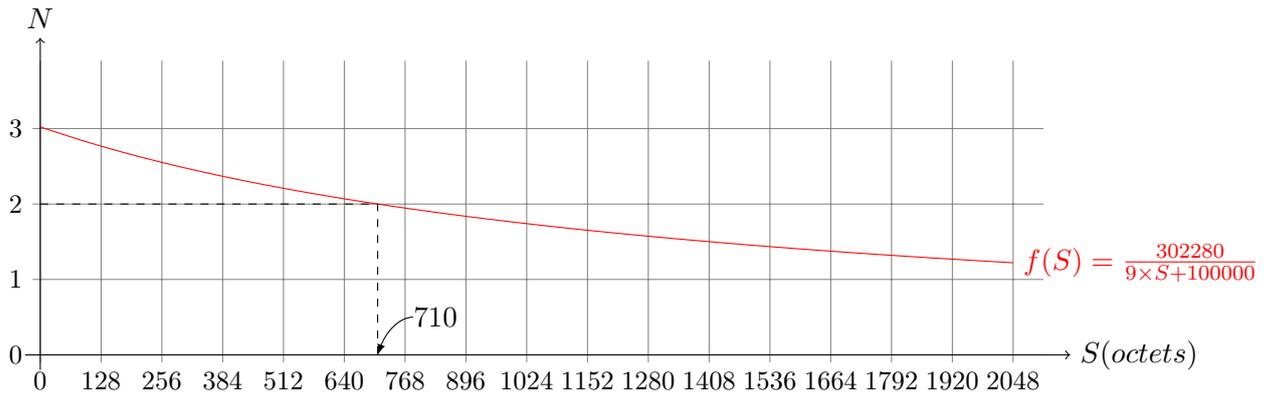
$$\frac{9 \times N \times S}{100 \times 10^3} + N > 3.0228 \quad (4.11)$$

$$\frac{9 \times N \times S + N \times 100 \times 10^3}{100 \times 10^3} > 3.0228 \quad (4.12)$$

$$N \times \frac{9 \times S + 100 \times 10^3}{100 \times 10^3 \text{bits}} > 3.0228 \quad (4.13)$$

$$N > \frac{3.0228 \times 100 \times 10^3}{9 \times S + 100 \times 10^3} \quad (4.14)$$

$$N > \frac{302280}{9 \times S + 100000} \quad (4.15)$$



On s'aperçoit qu'avec des petits paquets ( $< 710$  octets), il faut au minimum trois paquets pour que l'utilisation du canal « haut débit » soit rentable. À partir de 710 octets, deux paquets suffisent pour justifier l'utilisation du canal « haut débit » malgré son surcoût. Dans le cas où l'on considère qu'il n'y a qu'une seule commutation effectuée par le canal « bas débit » (les autres paquets sont envoyés à la suite), l'équation devient :

$$c + t_{rts} + c + t_{sync} + t_{cts} + \frac{N \times s}{B_{high}} + t_{ack} < c + \frac{N \times s}{B_{low}} \quad (4.16)$$

$$t_{rts} + c + t_{sync} + t_{cts} + \frac{N \times s}{B_{high}} + t_{ack} < \frac{N \times s}{B_{low}} \quad (4.17)$$

$$t_{rts} + c + t_{sync} + t_{cts} + t_{ack} < \frac{N \times s}{B_{low}} - \frac{N \times s}{B_{high}} \quad (4.18)$$

$$2.0228ms < \frac{9 \times N \times s}{100 \times 10^6 bits} \quad (4.19)$$

$$2.0228 < \frac{9 \times N \times s}{100000} \quad (4.20)$$

$$N \times s > 22476 \quad (4.21)$$

Il est donc avantageux d'utiliser le canal « haut débit » dès que l'on doit envoyer plus de 22476 bits (2810 octets). Néanmoins, dans ce cas, le temps d'envoi des données représente  $0.22ms$  (2810 octets à 100Mbps) alors que le temps de synchronisation représente  $1ms$ . 81% de la bande passante est donc gâchée avec le préambule de synchronisation. Nous avons donc prévu de pouvoir envoyer beaucoup plus d'informations dans une trame du canal « haut débit » en fixant la taille maximale d'une trame à 32768 octets. Ainsi, le temps d'envoi des données utiles représente 72% du temps d'envoi d'un message.

Les améliorations futures du système de synchronisation permettront de diminuer la période de synchronisation et donc d'augmenter ce taux.

### 4.4.3 Envoi d'un paquet en utilisant le canal « haut débit »

#### 4.4.3.1 Envoi classique

Pour envoyer les données, le canal « haut débit » utilise l'antenne sectorielle tant en émission qu'en réception. Le récepteur doit donc être prévenu que l'on souhaite lui envoyer un message. Lorsque l'ordonnanceur décide d'utiliser le canal « haut débit » pour envoyer des données, un premier message, appelé RTS, est envoyé par  $S$  à destination de  $R$  sur le canal « bas débit » (figure 4.11).

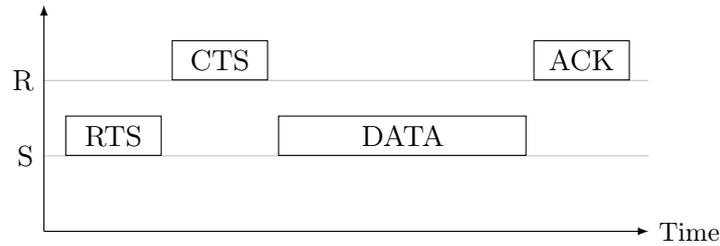


FIGURE 4.11 – Chronogramme d'envoi d'une trame de données simple sur le canal « haut débit »

Ce message contient le nombre d'octets que  $R$  souhaite envoyer à  $S$ . Si  $S$  reçoit correctement le RTS et s'il est disponible (assez de mémoire pour recevoir un message, pas de trop vieux paquet en attente...), il va orienter son antenne dans la direction de  $S$  et envoyer un CTS sur le canal « haut débit ». Comme il est possible d'envoyer plusieurs paquets dans une même trame, il peut arriver que  $R$  n'ait pas assez de mémoire pour tous les stocker. Il indique donc à  $S$  dans le message CTS la quantité de mémoire qu'il lui reste. S'il a assez de mémoire pour stocker toutes les données que  $R$  veut lui envoyer, il indiquera cette valeur dans le message.

Une fois le CTS reçu par  $S$ , celui-ci sélectionne dans la liste des paquets à envoyer à  $R$  les plus anciens, jusqu'à atteindre le seuil fixé par  $R$  dans sa trame CTS. Ces paquets sont placés dans une trame de données et envoyés sur le canal « haut débit ».

Si les données sont reçues correctement par  $R$ , il enverra un ACK sur le canal « haut débit » afin que  $S$  puisse supprimer de la file d'attente les paquets correspondants.

#### 4.4.3.2 Gestion des échecs

Il se peut qu'un nœud ne nous réponde pas suite à une sollicitation via une trame RTS. Les causes les plus naturelles peuvent être la collision de la trame au niveau du récepteur ou tout simplement le fait que le récepteur soit occupé et ne puisse pas recevoir de messages. Dans ces cas-là, il faut renvoyer la trame RTS plus tard.

Il ne faut pas oublier que la trame RTS a été envoyée avec la puissance préconisée par la table de voisinage. Si le nœud s'est éloigné, il est possible qu'il ne puisse plus entendre nos messages. C'est pour cela qu'en cas d'échec à la première tentative, les trames suivantes envoyées à ce nœud seront envoyées à la puissance maximale.

Si au bout de cinq tentatives le nœud ne répond toujours pas, celui-ci est supprimé de la liste des voisins et les paquets lui étant destinés sont supprimés des files d'attente. Il sera de nouveau considéré comme un voisin lorsqu'on recevra un nouveau message « Hello » de sa part.

#### 4.4.3.3 Amélioration

Une fois que les données de  $S$  ont été envoyées à  $R$  et juste avant l'envoi de l'ACK par  $R$ , les deux nœuds sont synchronisés. Il est alors possible d'envoyer des données « gratuitement ». Il serait donc judicieux que  $R$  puisse envoyer des données à  $S$  s'il en a dans ses files d'attente.

Pour ce faire, on rajoute dans la trame RTS la quantité de mémoire disponible pour la réception des messages afin que  $R$  sache combien d'octets il peut envoyer à  $S$ . De plus, on rajoute une option dans la trame CTS indiquant l'envoi ou non de données après l'accusé.

Après la bonne réception des données par  $R$ , celui-ci va envoyer à  $S$  un ACK (comme auparavant) et une trame de données dans la foulée. Il aura sélectionné les plus vieux paquets présents dans les files d'attente à destination de  $S$  tout en ne dépassant pas la quantité fixée par  $S$  dans son RTS (figure 4.12).

Une fois les données correctement reçues par  $S$ , ce dernier enverra un ACK à  $R$  afin de clôturer la communication. Il est ainsi possible dans certains cas d'économiser un préambule de synchronisation et une éventuelle commutation d'antenne.

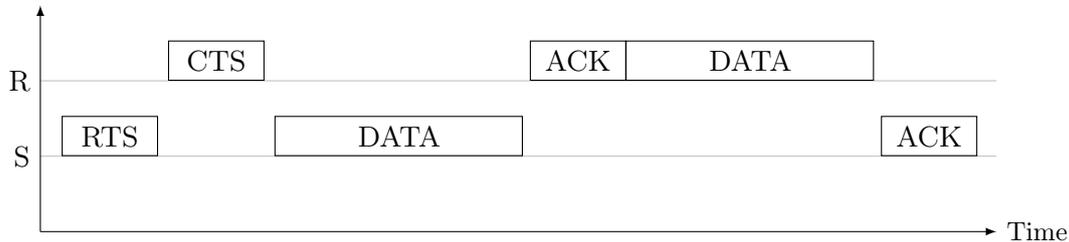


FIGURE 4.12 – Chronogramme d'envoi d'une trame de données bidirectionnelle

## 4.5 Algorithme de la couche MAC

### 4.5.1 Mode « bas débit »

L'algorithme de la couche MAC est donné sur la figure 4.13. Le pseudocode est présent en annexe A, page 81. Il s'agit de la boucle principale de la couche MAC lorsqu'elle est en mode « bas débit ». On note  $SC$  le secteur courant,  $T_{SC}$  le temps passé sur le secteur courant,  $T_{BW}$  le temps maximum qu'il faut rester sur le secteur courant,  $Q_{SC}$  la file d'attente du secteur courant,  $Q_{All}$  l'ensemble des files d'attente,  $T_p$  le temps d'attente du paquet  $P$  dans les files d'attente et  $S_{RTS}$  le secteur du RTS considéré.

La première préoccupation de la couche MAC est de savoir si la file d'attente du secteur courant est vide ou si l'on n'a pas dépassé le temps préconisé par  $T_{BW}$ . Si l'une de ces deux conditions est remplie, la couche MAC s'intéressera au paquet le plus ancien présent dans toutes les files d'attente, sinon elle se concentrera sur le secteur courant.

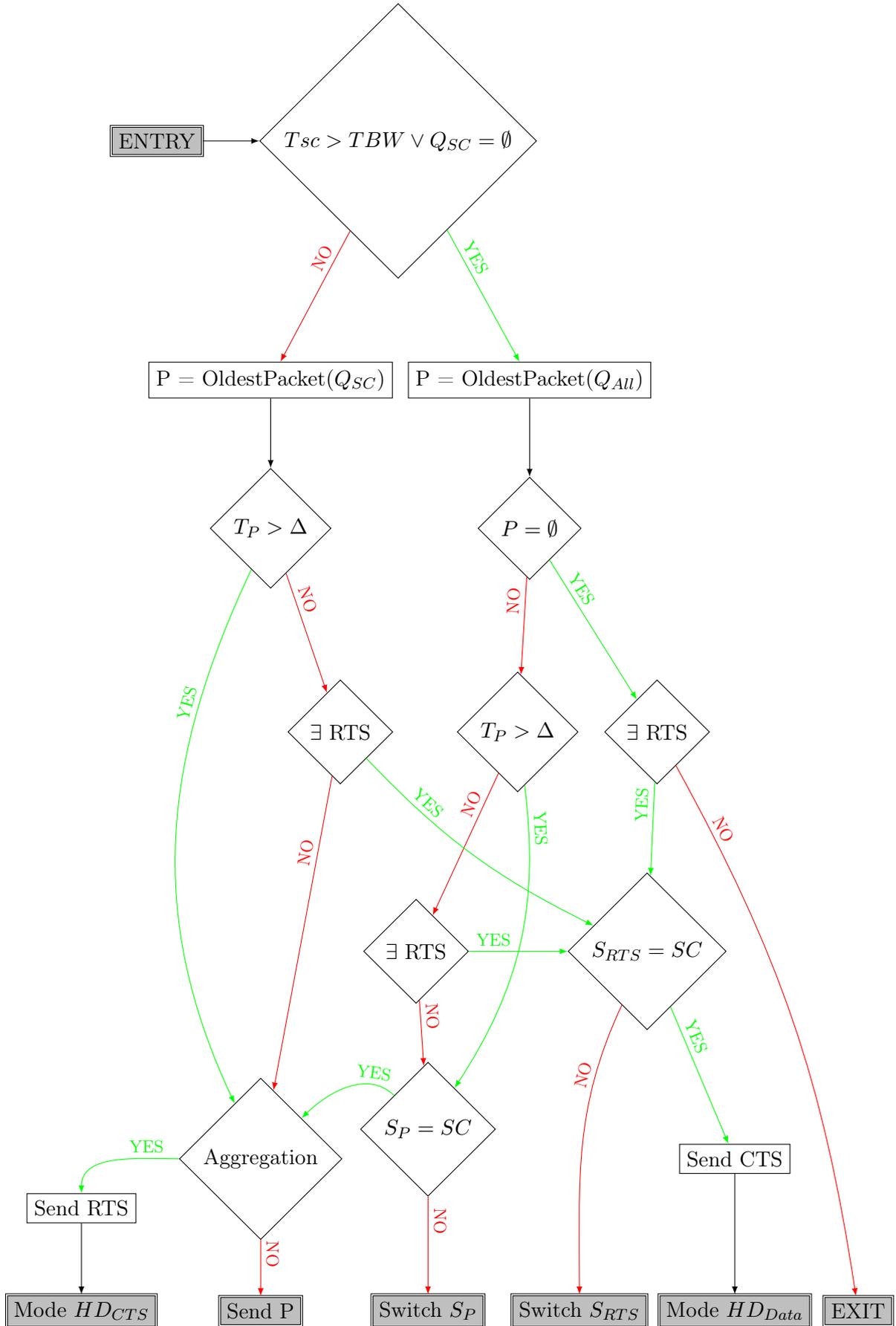
Une fois un paquet sélectionné (le plus vieux de la file d'attente du secteur courant), la couche MAC s'intéresse à l'ancienneté de ce paquet dans la file d'attente. S'il est considéré comme « trop vieux », il sera traité en priorité face aux événements extérieurs. Sinon, la couche MAC s'occupera du dernier RTS arrivé.

Si le paquet est ancien ou s'il n'y a pas de RTS à traiter, la couche MAC regarde la file d'attente pour voir s'il n'y a pas d'autres paquets pour le même destinataire. S'il y en a suffisamment pour justifier une agrégation et utiliser le canal « haut débit », la couche MAC envoie dans la direction du destinataire un RTS et passe en mode  $HD_{CTS}$ , sinon le paquet est envoyé directement sur le canal « bas débit ».

S'il y a un RTS à traiter, la couche MAC commute l'antenne dans la direction du destinataire si nécessaire, sinon elle envoie directement le CTS et passe en mode  $HD_{Data}$ .

Si la file d'attente est vide ou si l'on a passé trop de temps sur le secteur courant, la couche MAC sélectionne le plus vieux paquet parmi toutes les files d'attente. Si ce paquet est considéré comme « trop vieux », il est traité en priorité par rapport à un éventuel RTS. Dans ce cas, si le secteur de ce paquet n'est pas le secteur courant, la couche MAC commute l'antenne dans la bonne direction. Sinon elle exécute la routine d'agrégation qui enverra le paquet soit sur le canal « bas débit » soit sur le canal « haut débit ».

S'il n'y a plus de paquet à traiter, toutes les files d'attente étant vides, la couche MAC regarde s'il y a un RTS à traiter. Dans ce cas soit elle commute l'antenne dans la bonne direction soit elle envoie directement le CTS.

FIGURE 4.13 – Ordonnanceur (Mode *BD*)

### 4.5.2 Mode « haut débit »

Les différents modes « haut débit » ont des algorithmes plus simples. Ceux-ci sont représentés par les figures 4.14, 4.15 et 4.16.

Dans la plupart des cas, il s'agit d'attendre une certaine trame. Si elle est reçue dans les délais, on passe à l'action suivante, sinon on considère l'action comme échouée. Les modes  $HD_{ACK}$  et  $HD_{Data}$  retournent dans tous les cas au mode  $BD$  alors que la couche MAC passera du mode  $HD_{CTS}$  au mode  $HD_{ACK}$  une fois le CTS reçu et les données envoyées.

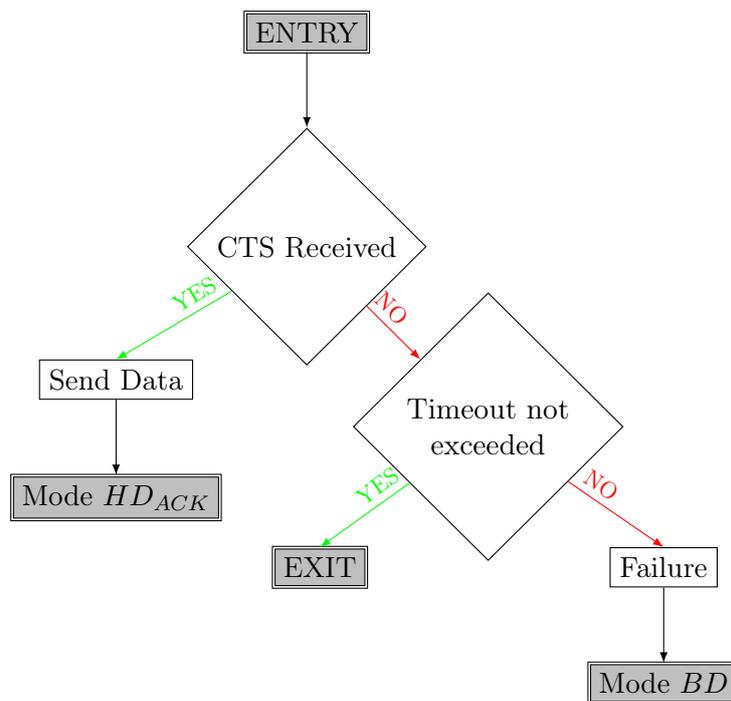


FIGURE 4.14 – Ordonnanceur (Mode  $HD_{CTS}$ )

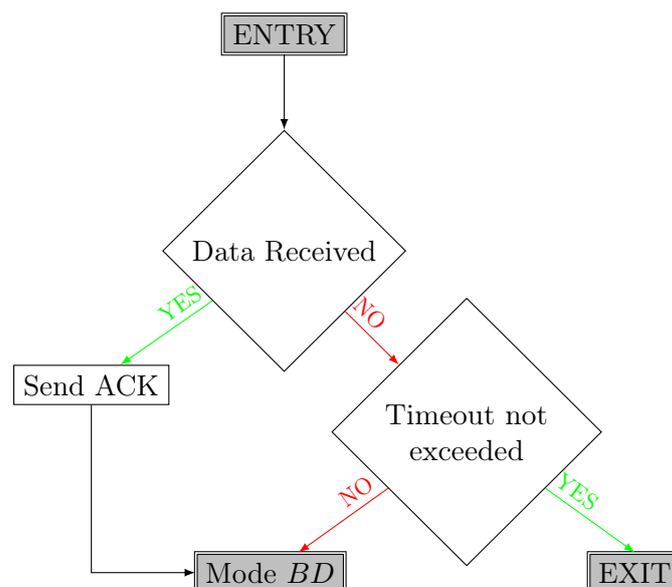
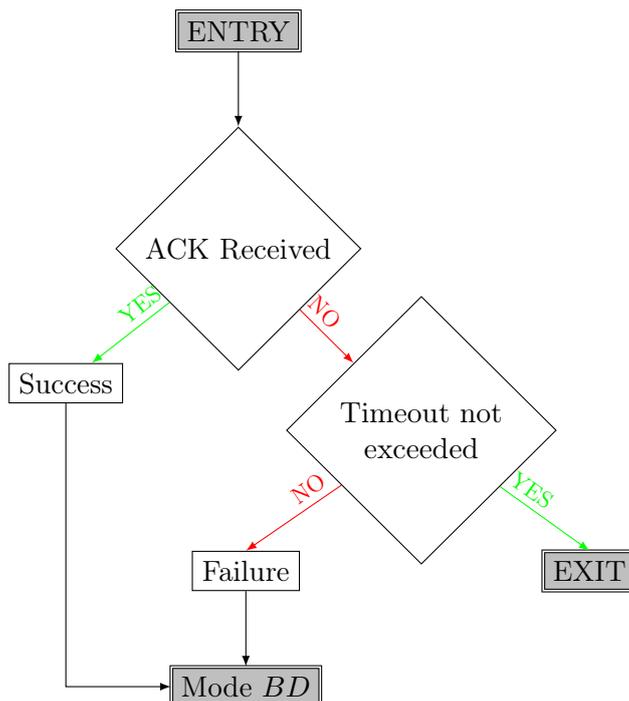


FIGURE 4.15 – Ordonnanceur (Mode  $HD_{Data}$ )

FIGURE 4.16 – Ordonnanceur (Mode  $HD_{ACK}$ )

## 4.6 Agrégation multi-utilisateurs

Dans la section 4.4, nous avons montré qu'il est important de pouvoir envoyer simultanément plusieurs paquets afin de diminuer le surcoût de la synchronisation du canal « haut débit ». Nous avons donc proposé d'agréger plusieurs paquets présents dans la file d'attente pour le même destinataire à l'intérieur d'une même trame de données.

Malheureusement, cela n'est pas forcément toujours possible. Une autre idée serait de pouvoir faire du multicast<sup>11</sup>. En envoyant une même trame à plusieurs destinataires, on pourrait vider rapidement une file d'attente avec tous les plus vieux paquets présents.

Actuellement, la synchronisation entre l'émetteur et le récepteur se fait grâce à l'envoi du CTS par le destinataire. Avec plusieurs destinataires potentiels, cette méthode n'est plus possible. En effet, il est indispensable que ce soit l'émetteur qui envoie la trame de synchronisation afin que tous les destinataires potentiels puissent se synchroniser dessus. Néanmoins, il n'est pas possible d'effectuer la synchronisation avec l'envoi du RTS, première trame envoyée par l'émetteur, car cette trame ne peut être envoyée sur le canal « haut débit ». Il faut donc que l'émetteur envoie la trame de synchronisation sur le canal « haut débit » après l'envoi de son RTS en espérant que les différents destinataires auront commuté leur antenne dans la bonne direction.

Dans un premier temps, il faut modifier le RTS de façon à indiquer plusieurs destinataires. Au lieu de spécifier un couple destination/taille dans l'entête, nous allons spécifier tous les destinataires susceptibles d'être contactés (figure 4.17). Ceux-ci sont choisis et listés en sélectionnant les plus vieux paquets de la file d'attente jusqu'à l'occurrence de 32768 octets. L'identifiant de trame est aussi modifié pour spécifier qu'il s'agit d'un RTS multiple.

Tant que les destinataires n'ont pas eu le temps de traiter le RTS qu'ils viennent de recevoir (figure 4.19), la trame de synchronisation ne doit pas être envoyée. En effet, ceux-ci peuvent être occupés à envoyer un message. Le temps maximum d'envoi d'un message est de  $1.6ms$  sur le

11. Méthode de diffusion de l'information d'un émetteur (source unique) vers un groupe (plusieurs destinataires)

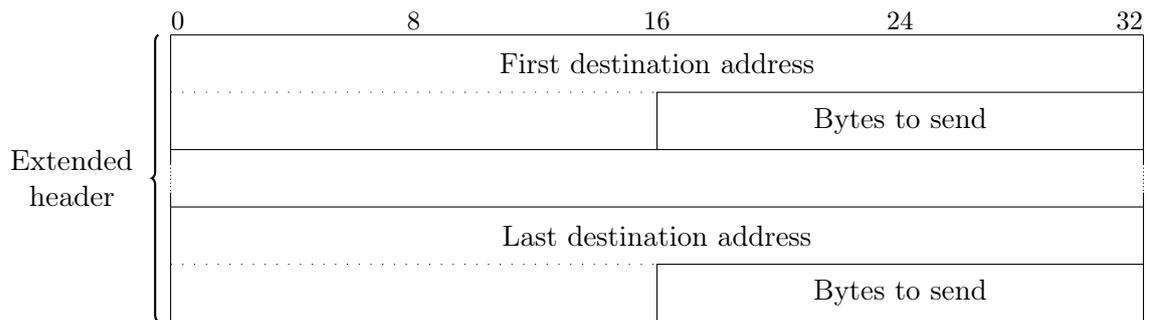


FIGURE 4.17 – Structure de l’entête étendue pour une trame RTS multiple

canal « bas débit » et de  $2.6ms$  sur le canal « haut débit ». Il faut aussi prendre en compte dans le calcul le potentiel temps de commutation de l’antenne des destinataires. En définitive, il faut attendre au moins  $4ms$  pour être sûr de la disponibilité de tous les destinataires.

Une fois ce délai écoulé, il est possible d’envoyer la trame de synchronisation ☒. Celle-ci est reçue par tous les nœuds du secteur disponibles et concernés par le RTS. Ces derniers ont normalement commuté leur antenne ☒ avant le délai imparti. Ils peuvent alors envoyer chacun leur tour leur CTS, selon l’ordre établi dans le RTS multiple. Cela permet d’éviter les collisions à la réception. Si un nœud n’est pas disponible, il n’enverra pas de CTS et l’émetteur saura qu’il ne faut pas envoyer de données pour ce nœud.

Dès que le dernier CTS est reçu, l’émetteur sélectionne les paquets des nœuds ayant répondu pour former une trame de données multicast (figure 4.18). Cette trame est la concaténation de plusieurs trames de données normales encapsulées dans une trame de base. Chaque destinataire recevra la trame complète et pourra extraire les paquets qui lui sont destinés.

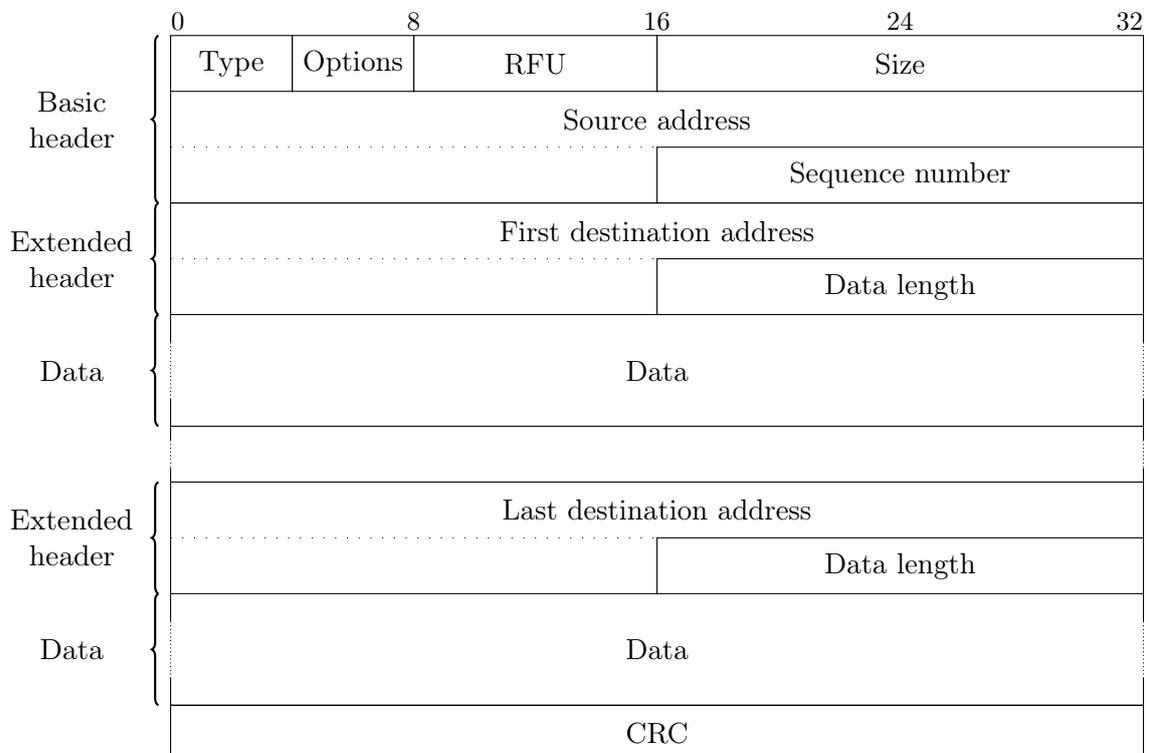


FIGURE 4.18 – Structure complète d’une trame de données multicast

Chaque destinataire peut ensuite envoyer son ACK, toujours dans l'ordre établi à l'origine. L'étude des paquets contenus dans la trame de données leur permet de savoir si l'un des destinataires présents dans le RTS n'a pas répondu. Le cas échéant, il n'est pas nécessaire pour un destinataire d'attendre que son prédécesseur ait envoyé son ACK avant d'envoyer le sien (voir le cas de *R3* et *R4* dans la figure).

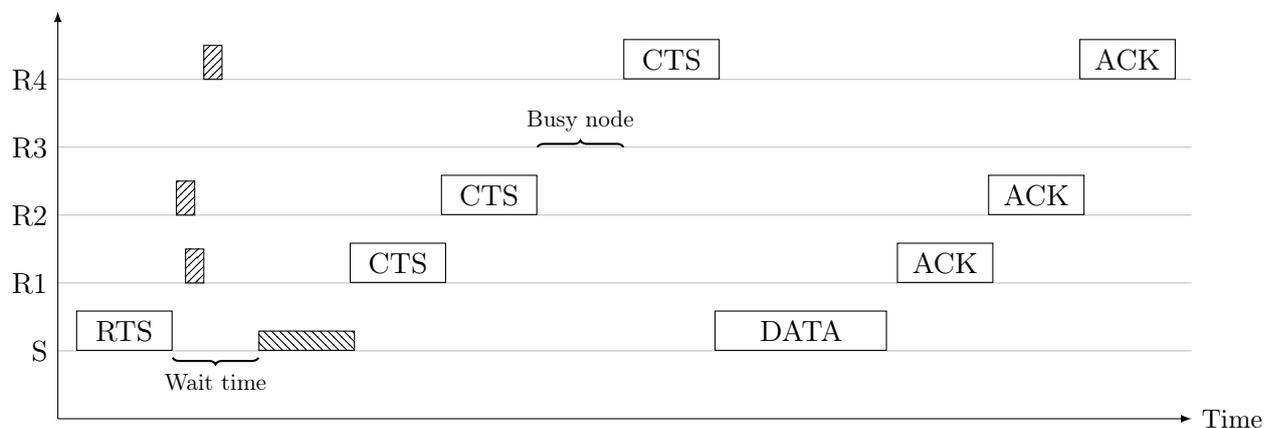


FIGURE 4.19 – Chronogramme de l'agrégation multi-utilisateurs sur le canal « haut débit »

Cette méthode permet de regrouper des paquets de plusieurs destinataires en une seule trame afin d'économiser des commutations et des trames de synchronisation. Sans ce système, on a dans le pire des cas une commutation et une trame de synchronisation pour chaque trame envoyée à un destinataire différent. Ici, la trame de synchronisation et les commutations des destinataires sont mutualisées au prix d'un temps d'attente initial. Il faut remarquer que ce temps d'attente vaut moins de  $4ms$ . On est donc gagnant à partir de trois destinataires.

## 4.7 Conclusion

Dans ce chapitre, nous avons étudié la problématique d'accès au canal au moyen d'antennes sectorielles disposant d'un temps de commutation non négligeable. Les différents protocoles de la littérature ne prenant pas en compte ce temps de commutation, il fallait en créer un nouveau pour répondre à ce problème.

Nous avons dans un premier temps décrit les différentes trames de contrôle et de données de cette nouvelle couche MAC. Nous avons ensuite étudié les avantages et les inconvénients des deux canaux radio mis à notre disposition et décrit dans quel cas il était avantageux d'utiliser le canal « haut débit », plus complexe, plutôt que le canal « bas débit ».

Dans le prochain chapitre, nous allons discuter du prototype que nous avons construit pour implémenter cette couche MAC.

## Cinquième chapitre

---

# Implémentation

### Sommaire

---

<b>5.1</b>	<b>Introduction</b>	<b>62</b>
<b>5.2</b>	<b>Description de la plateforme d'expérimentation</b>	<b>62</b>
5.2.1	Plateforme Excalibur	62
5.2.2	Couche physique	62
5.2.3	Couche liaison de données	63
<b>5.3</b>	<b>Interface <i>Hard-Soft</i></b>	<b>63</b>
<b>5.4</b>	<b>Expérimentations réalisées</b>	<b>64</b>
5.4.1	Description du prototype	64
5.4.2	Expérimentations	65
<b>5.5</b>	<b>Conclusion</b>	<b>65</b>

---

## 5.1 Introduction

Nous allons décrire dans ce chapitre le prototype que nous avons réalisé afin de tester notre pile protocolaire avec des antennes sectorielles à faisceaux agiles.

Dans un premier temps, nous allons détailler la plateforme matérielle que nous avons utilisée ainsi que la couche physique employée. Dans un second temps, nous allons décrire l'interface entre le matériel et le logiciel. Nous présenterons ensuite les expérimentations réalisées avant de conclure.

## 5.2 Description de la plateforme d'expérimentation

### 5.2.1 Plateforme Excalibur

Le prototype a été développé grâce à la plateforme CM922T-XA10 d'ARM Ltd. Cette plateforme est composée de deux cartes : *Core Module* et *Compact Platform*. Le *Core Module* contient de la mémoire volatile et de la mémoire *flash*, ainsi qu'une puce d'Altera nommée *Excalibur*, composée d'un processeur ARM<sup>1</sup> et d'un FPGA<sup>2</sup>. La structure de la puce *Excalibur* est donnée en annexe B, page 85. Cette carte permet de développer la logique du prototype.

La *Compact Platform* est une carte pouvant recevoir différents modèles de *Core Module*. Elle comporte notamment l'étage d'alimentation ainsi qu'une couche physique Ethernet. Elle permet donc d'interfacer le *Core Module* avec un ordinateur via un câble Ethernet.

### 5.2.2 Couche physique

La couche physique est composée de deux parties. La première gère la haute fréquence à 60 GHz. Elle a été développée sur silicium filière BiCMOS<sup>3</sup>. Elle contient pour l'émetteur un oscillateur à 60 GHz ainsi qu'un commutateur permettant de générer soit le signal ASK<sup>4</sup> pour le canal « bas débit » soit les impulsions UWB-PPM<sup>5</sup> pour le canal « haut débit ».

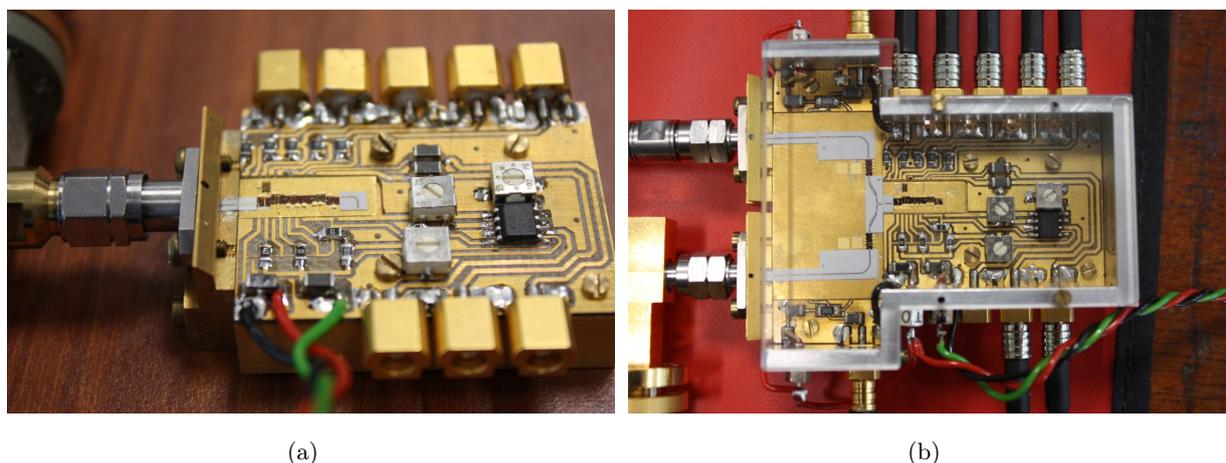


FIGURE 5.1 – Puces 60 GHz

1. ARM : *Advanced Risc Machine* – Processeur RISC 32 bits vendu sous licence par ARM Ltd.
2. FPGA : *Field-Programmable Gate Array* – Circuit logique programmable
3. BiCMOS : Contraction de Bipolar-CMOS : nom d'une technique de circuit intégré alliant les avantages du CMOS et du bipolaire
4. ASK : *Amplitude-Shift Keying* – modulation en amplitude
5. UWB-PPM : *Ultra WideBand - Pulse-Position Modulation* – Modulation radio à large bande avec codage par position d'impulsions

Pour le récepteur, un premier filtre passe-bande permet de séparer le canal « bas débit » du canal « haut débit ». Ensuite, un simple détecteur démodule le signal du canal « bas débit ». La sortie de ce détecteur est directement connectée à une entrée du FPGA afin d'y être traitée. Pour le canal « haut débit », l'échantillonnage se fait directement dans les puces à 60 GHz afin de pouvoir détecter les très courtes impulsions.

La deuxième partie de la couche physique sert essentiellement à faire l'adaptation des niveaux logiques entre le FPGA de la carte ARM et les puces 60 GHz. Une des cartes (figure 5.5) sert à traiter la synchronisation de l'échantillonneur lors de la réception du préambule de synchronisation sur le canal « haut débit ».

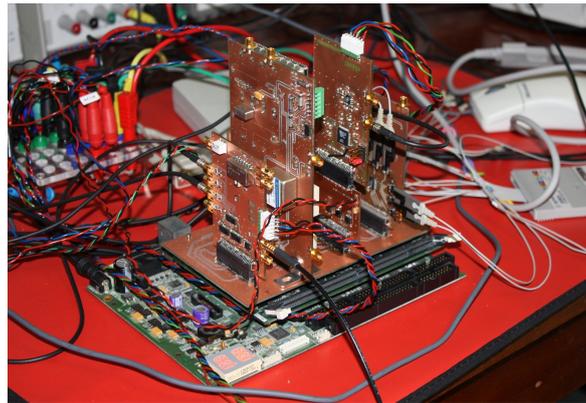


FIGURE 5.2 – Cartes d'adaptation montées sur les cartes ARM

### 5.2.3 Couche liaison de données

La couche liaison de données est implémentée à moitié en logiciel via le processeur de la carte et dans le FPGA. Celui-ci permet de gérer toute la logique simple d'envoi des trames : préambule de synchronisation, fin de trame pour l'émission. Il gère également leur réception : vérification de la taille, vérification du CRC<sup>6</sup> et stockage en mémoire.

Il permet aussi de décharger le processeur pour des tâches de copie mémoire (DMA<sup>7</sup>).

## 5.3 Interface *Hard-Soft*

L'interface entre le logiciel, qui s'exécute dans le CPU<sup>8</sup>, et le matériel présent dans le FPGA se fait à l'aide d'une mémoire spéciale appelée DPRAM<sup>9</sup> (figure 5.4). Cette mémoire, présente dans la puce *Excalibur*, dispose de deux ports d'accès, l'un relié au bus du processeur et l'autre interfaçable dans le FPGA. Il est donc possible d'y accéder depuis les deux mondes.

Nous avons utilisé cette mémoire afin de stocker les trames à émettre ainsi que les trames reçues. Elle est découpée en quatre zones de 32 kioctets chacune permettant de stocker respectivement

---

6. CRC : *Cyclic Redundancy Check* – Contrôle de redondance cyclique permettant de détecter les erreurs de transmission

7. DMA : *Direct Memory Access* – Accès direct à la mémoire, procédé informatique où des données circulant de ou vers un périphérique sont transférées directement par un contrôleur adapté vers la mémoire principale de la machine, sans intervention du microprocesseur si ce n'est pour lancer et conclure le transfert.

8. CPU : *Central Processing Unit* – Processeur

9. DPRAM : *Dual Ported Random Access Memory* – Type de mémoire vive permettant des accès multiples en entrée et en sortie

les trames en émission et en réception pour le canal « bas débit » et « haut débit ». Il est donc possible de stocker directement au niveau du FPGA au minimum seize trames en attente d'émission et seize trames reçues sur le canal « bas débit ». Les deux autres zones de 32 kibioctets permettent de stocker une seule trame du canal « haut débit » tant en émission qu'en réception. Ceci n'est pas un problème. En effet, que ce soit en émission ou en réception, l'accès à la couche physique est cadencée par le logiciel avec l'envoi des trames RTS<sup>10</sup>, CTS<sup>11</sup>, ACK<sup>12</sup>...



FIGURE 5.3 – Organisation matérielle du prototype

Nous avons aussi implémenté directement dans l'espace mémoire du processeur, via le bus AHB<sup>13</sup>, des registres à l'intérieur du FPGA permettant d'obtenir l'état interne des envois. Une DMA a aussi été développée à l'intérieur du FPGA afin d'accélérer l'envoi et la réception de données avec la carte Ethernet.

## 5.4 Expérimentations réalisées

### 5.4.1 Description du prototype

Le prototype que nous avons réalisé fonctionne comme un *bridge* entre sa carte réseau et son interface sans fil. C'est-à-dire que tous les paquets Ethernet reçus par la carte seront envoyés sur la liaison sans fil et réciproquement. La carte ne s'intéresse qu'à la couche 2 (liaison de données) des paquets qu'elle reçoit. Le pont sans fil est donc transparent pour les différents ordinateurs du réseau.

Chaque prototype s'attribue l'adresse MAC<sup>14</sup> de l'ordinateur auquel il est connecté et le service de découverte de voisinage permet de découvrir l'existence et la direction de chaque voisin à portée de communication radio. Ainsi, lorsqu'un paquet arrive de la carte Ethernet, il est analysé puis envoyé dans la bonne file d'attente.

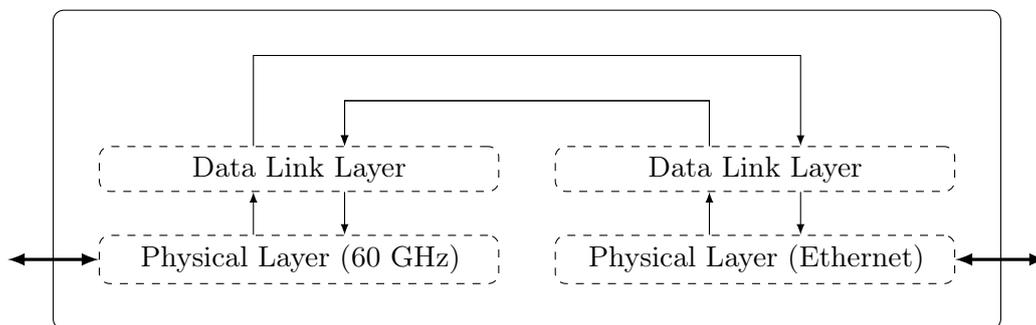


FIGURE 5.4 – Organisation logique du prototype

10. RTS : *Request to Send*

11. CTS : *Clear to Send*

12. ACK : *Acknowledgment code* – Acquiescement des données

13. AHB : *AMBA High-performance Bus* – Bus informatique introduit par ARM Ltd.

14. MAC : *Media Access Control* – Contrôle d'accès au support

### 5.4.2 Expérimentations

Grâce à ce pont, nous avons pu tester la faisabilité de l'implémentation de notre pile protocolaire. Certaines expériences et démonstrations ont été menées, comme par exemple le jour de l'inauguration de l'IRCICA<sup>15</sup>. Durant ces expérimentations, nous avons pu tester et valider le canal « bas débit ».



FIGURE 5.5 – Expérimentation publique réalisée dans le cadre de l'inauguration de l'IRCICA

Malheureusement, à la date de ce jour, les puces 60 GHz pour le canal « haut débit » ainsi que l'antenne MEMS<sup>16</sup> ne sont pas pleinement fonctionnelles. Les contacteurs MEMS prévus pour commuter les différentes antennes restent parfois bloqués, ce qui rend l'antenne inutilisable. Nous n'avons donc pas pu tester ni valider le canal « haut débit ».

## 5.5 Conclusion

Dans ce chapitre, nous avons décrit le prototype qui a été développé dans le cadre de cette nouvelle pile protocolaire. Ce prototype devait nous permettre d'effectuer des mesures précises « dans la vraie vie » au lieu d'utiliser un simulateur.

Malheureusement, le matériel permettant d'établir des communications à très haut débit à 60 GHz a rencontré des difficultés nous empêchant de procéder à nos mesures et validations. Nous espérons dans un bref futur pouvoir terminer ces expérimentations et valider dans la pratique la pile protocolaire.

---

15. IRCICA : Institut de Recherche sur les Composants logiciels et matériels pour l'Information et la Communication Avancée

16. MEMS : *Microelectromechanical systems*



## Sixième chapitre

---

# Conclusion et perspectives

### Sommaire

---

6.1	Synthèse . . . . .	68
6.2	Perspectives . . . . .	68

---

## 6.1 Synthèse

Dans cette thèse nous avons présenté et étudié le canal radio à 60 GHz avec des antennes à faisceaux agiles dans le contexte des réseaux ad hoc. Nous nous sommes intéressés à la problématique de ces antennes directionnelles qui imposent un temps de commutation à chaque changement de direction. Les protocoles de la littérature ne prennent pas en compte ce temps de commutation qui implique une baisse drastique des performances. Le temps de commutation pouvant être proche du temps d'envoi d'un paquet, le débit utile se trouve divisé au moins par deux. D'autre part, certains protocoles utilisent des antennes sectorielles, mais ne résolvent pas le problème de localisation des voisins. Il est difficile d'envoyer un message dans la direction de quelqu'un si on ne sait pas où il se trouve.

Il est donc indispensable d'avoir une pile protocolaire adaptée, qui prenne en compte ce temps de commutation afin d'éviter la perte de débit et résout la problématique de localisation. Pour ce faire, nous proposons dans ce document une solution à ces deux problèmes. La localisation des voisins se fait grâce à un nouveau protocole de découverte de voisinage que nous proposons. Celui-ci permet d'une part de connaître la direction des voisins, mais aussi de déterminer à quelle puissance il faut envoyer les messages afin que lesdits voisins puissent les recevoir. Ce système favorise une économie d'énergie et diminue l'occupation géographique du signal. Il est ainsi possible d'augmenter les performances globales du réseau.

Le temps de commutation est géré grâce à des files d'attente et au regroupement de paquets destinés au même secteur. On oblige le nœud à envoyer les paquets dans un même secteur avant de passer au suivant. Afin d'éviter de trop longs délais, le temps passé dans un secteur est borné. De plus, il est possible d'envoyer dans une même trame, plusieurs paquets pour un même destinataire afin de diminuer le coût des trames de contrôle. Enfin, dans le cas où l'on ne dispose pas d'assez de données pour un même nœud, nous proposons une méthode pour envoyer une trame contenant des paquets pour plusieurs destinataires. Cette méthode est plus complexe, mais permet aussi de diminuer le nombre de commutations et le coût des trames de contrôle.

Enfin, nous présentons un prototype qui nous a permis de valider la faisabilité d'une telle pile protocolaire. Malheureusement, l'absence de matériel fonctionnel pour la partie radio à 60 GHz nous a empêchés d'effectuer des mesures précises et de valider la pile protocolaire.

## 6.2 Perspectives

L'une des premières perspectives est de pouvoir tester « en vrai » la pile protocolaire avec une couche physique fonctionnelle. On pourrait ainsi déterminer son bon fonctionnement et mesurer le gain apporté.

Une autre problématique des réseaux ad hoc est la communication multi-sauts. En effet, dans ces réseaux auto-organisés, un nœud peut servir de relai intermédiaire pour permettre à deux autres nœuds de communiquer. Ce nœud intermédiaire doit donc sans cesse écouter dans une direction pour réémettre dans une autre, ce qui implique un grand nombre de commutations et de trames de contrôle.

Enfin, un aspect qui n'a pas été étudié dans cette thèse est la mobilité des nœuds. Cette mobilité peut être de deux types :

**Mobilité géographique** Surtout dans un réseau ad hoc, les nœuds sont amenés à se déplacer.

Ils peuvent soit s'éloigner, soit changer de secteur. Une proposition à l'éloignement est décrite dans ce mémoire. Nous proposons d'augmenter la puissance d'émission si un nœud ne nous répond pas. Mais si un nœud change de secteur, il devient totalement injoignable. Il peut être possible, par exemple, de modéliser les déplacements de ses voisins afin d'essayer d'anticiper leurs déplacements. Ainsi, en étudiant l'évolution des messages « Hello » de

ses voisins, on peut essayer de déterminer s'ils se rapprochent ou s'éloignent. De plus, en regardant ces messages, on peut s'apercevoir que ces voisins tournent dans un certain sens autour de nous et tenter l'envoi dans un secteur voisin si un nœud est injoignable. Une autre possibilité de déplacement à prendre en compte : le nœud lui même et pas ses voisins.

**Mobilité de rotation** Cette mobilité est la pire de toutes car avec une légère rotation du nœud, toute la table de voisinage devient obsolète. De plus, ce mouvement peut être trop rapide et empêcher toute anticipation avec les messages de découverte de voisinage. Néanmoins, de plus en plus d'objets mobiles communicants embarquent des outils comme des gyroscopes permettant de détecter les mouvements du mobile. Il peut donc être intéressant de prendre en compte ces informations pour en déduire des modifications probables de la table de voisinage.



# Liste des figures

---

1.1	Interaction multi-utilisateur . . . . .	11
1.2	Structure d'une antenne à faisceaux agiles . . . . .	11
2.1	Réseau avec trois stations de base et neuf utilisateurs . . . . .	16
2.2	Réseau maillé avec trois nœuds d'infrastructure et neuf utilisateurs . . . . .	17
2.3	Réseau pair-à-pair avec neuf utilisateurs . . . . .	17
2.4	Modèle OSI en 7 couches . . . . .	18
2.5	TDMA . . . . .	22
2.6	Pure ALOHA, les paquets grisés indiquent les collisions . . . . .	23
2.7	Slotted ALOHA, les paquets grisés indiquent les collisions . . . . .	24
2.8	Problème du terminal caché . . . . .	25
2.9	RTS/CTS . . . . .	25
2.10	Scénario d'envoi d'un paquet avec MMAC . . . . .	27
2.11	Un nœud avec $M$ faisceaux . . . . .	28
2.12	Sélection des secteurs pour le CTS par CRCM . . . . .	29
2.13	Récapitulatif des secteurs utilisés pour l'envoi des RTS/CTS . . . . .	30
2.14	Différents faisceaux nécessaires pour SWAMP . . . . .	31
3.1	Commutateur MEMS . . . . .	34
3.2	Scénario de découverte de voisinage . . . . .	37
3.3	Différentes approches possibles pour la couche liaison de données . . . . .	39
3.4	Représentation OSI de la pile protocolaire . . . . .	40
3.5	Structure d'un paquet « Hello » . . . . .	40
4.1	Automate de la couche MAC . . . . .	46
4.2	Structure d'une trame . . . . .	47
4.3	Structure de l'entête étendue pour une trame RTS . . . . .	47
4.4	Structure de l'entête étendue pour une trame CTS . . . . .	48
4.5	Structure de l'entête étendue pour une trame ACK . . . . .	48
4.6	Structure de la trame « Hello » . . . . .	49
4.7	Structure de la trame de données . . . . .	49
4.8	Exemple de terminal caché avec des antennes sectorielles . . . . .	50
4.9	Modulation PPM . . . . .	51
4.10	Structure de la trame dans les airs sur le canal « haut débit » . . . . .	51
4.11	Chronogramme d'envoi d'une trame de données simple sur le canal « haut débit » . . . . .	54
4.12	Chronogramme d'envoi d'une trame de données bidirectionnelle . . . . .	55
4.13	Ordonnanceur (Mode $BD$ ) . . . . .	56
4.14	Ordonnanceur (Mode $HD_{CTS}$ ) . . . . .	57
4.15	Ordonnanceur (Mode $HD_{Data}$ ) . . . . .	57

---

4.16	Ordonnanceur (Mode $HD_{ACK}$ ) . . . . .	58
4.17	Structure de l'entête étendue pour une trame RTS multiple . . . . .	59
4.18	Structure complète d'une trame de données multicast . . . . .	59
4.19	Chronogramme de l'agrégation multi-utilisateurs sur le canal « haut débit » . . . . .	60
5.1	Puces 60 GHz . . . . .	62
5.2	Cartes d'adaptation montées sur les cartes ARM . . . . .	63
5.3	Organisation matérielle du prototype . . . . .	64
5.4	Organisation logique du prototype . . . . .	64
5.5	Expérimentation publique réalisée dans le cadre de l'inauguration de l'IRCICA . . . . .	65

# Liste des sigles et acronymes

---

<b>Arcep</b>	Autorité de Régulation des Communications Électroniques et des Postes
<b>ACK</b>	<i>Acknowledgment code</i> – Acquittement des données
<b>ADCCP</b>	<i>Advanced Data Communication Control Procedures</i>
<b>AHB</b>	<i>AMBA High-performance Bus</i> – Bus informatique introduit par ARM Ltd.
<b>ARM</b>	<i>Advanced Risc Machine</i> – Processeur RISC 32 bits vendu sous licence par ARM Ltd.
<b>AODV</b>	<i>Ad hoc On-Demand Distance Vector</i>
<b>ALOHA</b>	Nom inspiré du mot aloha signifiant en langue hawaïenne bonjour, au revoir
<b>ASK</b>	<i>Amplitude-Shift Keying</i> – modulation en amplitude
<b>ANFR</b>	Agence Nationale des FRéquences [ANF]
<b>Arpanet</b>	<i>Advanced Research Projects Agency Network</i>
<b>BiCMOS</b>	<i>Contraction de Bipolar-CMOS : nom d'une technique de circuit intégré alliant les avantages du CMOS et du bipolaire</i>
<b>CDMA</b>	Code Division Multiple Access – <i>Accès multiple par répartition en code</i>
<b>CRCM</b>	Circular RTS and CTS MAC
<b>CPER</b>	<i>Contrat de projets État-région</i>
<b>CPU</b>	Central Processing Unit – <i>Processeur</i>
<b>CRC</b>	Cyclic Redundancy Check – <i>Contrôle de redondance cyclique permettant de détecter les erreurs de transmission</i>
<b>CSMA</b>	Carrier Sense Multiple Access – <i>Accès multiple par détection de porteuse</i>
<b>CSMA/CA</b>	Carrier Sense Multiple Access with Collision Avoidance – <i>Accès multiple par détection de porteuse avec évitement des collisions</i>
<b>CSNet</b>	Computer Science Network
<b>CTS</b>	<i>Clear to Send</i>
<b>DCF</b>	<i>Distributed Coordination Function</i>
<b>DMA</b>	<i>Direct Memory Access</i> – Accès direct à la mémoire, procédé informatique où des données circulant de ou vers un périphérique sont transférées directement par un contrôleur adapté vers la mémoire principale de la machine, sans intervention du microprocesseur si ce n'est pour lancer et conclure le transfert.
<b>DMAC</b>	<i>Directional MAC</i>
<b>DMAC/DA</b>	<i>Directional MAC with Deafness Avoidance</i>
<b>DPRAM</b>	<i>Dual Ported Random Access Memory</i> – Type de mémoire vive permettant des accès multiples en entrée et en sortie

<b>DOD CTS</b>	<i>Diametrically Opposite Directional CTS</i>
<b>DOD RTS</b>	<i>Diametrically Opposite Directional RTS</i>
<b>DS</b>	<i>Data-Sending</i>
<b>ESPAR</b>	<i>Electronically Steerable Parasitic Array Radiator</i> – Antenne orientable électroniquement grâce à un ensemble d'éléments parasites
<b>ETSI</b>	<i>European Telecommunications Standards Institute</i> – Institut européen des normes de télécommunication
<b>FDMA</b>	<i>Frequency Division Multiple Access</i> – Accès multiple par répartition en fréquence
<b>FIFO</b>	<i>First In, first Out</i> – « Premier arrivé, premier servi »
<b>FPGA</b>	<i>Field-Programmable Gate Array</i> – Circuit logique programmable
<b>GSM</b>	<i>Global System for Mobile Communications</i> – Standard de téléphonie mobile utilisé principalement en Europe et en Asie
<b>GPS</b>	<i>Global Positioning System</i>
<b>HD</b>	Haute définition
<b>HDLC</b>	<i>High-Level Data Link Control</i>
<b>IRCICA</b>	Institut de Recherche sur les Composants logiciels et matériels pour l'Information et la Communication Avancée
<b>IEEE</b>	<i>Institute of Electrical and Electronics Engineers [IEE]</i> – La plus grande association mondiale dédiée à l'avancement de l'innovation technologique
<b>IEMN</b>	Institut d'Électronique, de Microélectronique et de Nanotechnologie [IEM]
<b>ISO</b>	<i>International Organization for Standardization</i> – Organisation Internationale de Normalisation [ISO]
<b>ITU-T</b>	<i>International Telecommunication Union</i> – Union internationale des télécommunications [ITU]
<b>LOMC</b>	Liaison de Proximité Haut Débit entre Objets Mobiles Communicants
<b>MAC</b>	<i>Media Access Control</i> – Contrôle d'accès au support
<b>MACAW</b>	<i>Multiple Access with Collision Avoidance for Wireless</i>
<b>MANET</b>	<i>Mobile Ad-hoc NETWORKs</i>
<b>MDA</b>	<i>MAC protocol for Directional Antennas</i>
<b>MMAC</b>	<i>Multi-hop RTS MAC</i>
<b>MEMS</b>	<i>Microelectromechanical systems</i>
<b>NSFNet</b>	The National Science Foundation Network
<b>NPN</b>	<i>Next Packet Notification</i>
<b>OLSR</b>	<i>Optimized link state routing protocol</i>
<b>OSI</b>	<i>Open Systems Interconnection</i>
<b>OSPF</b>	<i>Open shortest path first</i>
<b>PARC</b>	<i>Palo Alto Research Center</i>
<b>PDMA</b>	<i>Polarisation Division Multiple Access</i> – Accès multiple par répartition en polarisation
<b>PPM</b>	Pulse-Position Modulation
<b>RRTS</b>	Request for Request to Send

---

<b><i>RTS</i></b>	Request to Send
<b><i>RFU</i></b>	Reserved for Future Use – <i>Réservé pour un usage futur</i>
<b><i>SDMA</i></b>	Space-Division Multiple Access – <i>Accès multiple par division géographique</i>
<b><i>SIFS</i></b>	Short Inter Frame Space – <i>Temps minimal qui sépare l'envoi de deux paquets sur le réseau dans IEEE 802.11 (~ 10 <math>\mu</math>s dans 802.11b)</i>
<b><i>SOF</i></b>	Start Of Frame
<b><i>SWAMP</i></b>	Smart Antennas Based Wider-range Access MAC Protocol
<b><i>TAP</i></b>	Turnover based Adaptive HELLO Protocol
<b><i>TBW</i></b>	Temporized Beam Window
<b><i>TCP/IP</i></b>	Transmission Control Protocol / Internet Protocol – <i>Protocoles de la couche 3 et 4 du modèle OSI</i>
<b><i>TDMA</i></b>	Time Division Multiple Access – <i>Accès multiple à répartition dans le temps</i>
<b><i>UMTS</i></b>	Universal Mobile Telecommunications System – <i>Système de téléphonie mobile également appelé 3G (troisième génération) permettant des échanges de données jusqu'à 2 Mbps</i>
<b><i>UWB-PPM</i></b>	Ultra WideBand - Pulse-Position Modulation – <i>Modulation radio à large bande avec codage par position d'impulsions</i>
<b><i>WTS</i></b>	Wait To Send



# Bibliographie

---

- [Abr70] Norman ABRAMSON : The aloha system : another alternative for computer communications. *Dans Proceedings of the November 17-19, 1970, fall joint computer conference, AFIPS '70 (Fall)*, pages 281–285, New York, NY, USA, 1970. ACM.
- [ANF] Agence nationale des fréquences. <http://www.anfr.fr/>.
- [ARC08] ARCEP<sup>1</sup> : Fréquences utilisables pour certains matériels de faible puissance et de faible portée, 2008. <http://www.anfr.fr/pages/tnrbf/A7.pdf>.
- [BDSZ94] Vaduvur BHARGHAVAN, Alan DEMERS, Scott SHENKER et Lixia ZHANG : Macaw : a media access protocol for wireless lan's. *Dans SIGCOMM '94 : Proceedings of the conference on Communications architectures, protocols and applications*, pages 212–225, New York, NY, USA, 1994. ACM.
- [Bra89] R. T. BRADEN : RFC 1122 : Requirements for Internet hosts — communication layers, octobre 1989. Status : STANDARD. <ftp://ftp.internic.net/rfc/rfc1122.txt>.
- [CHIO03] J. CHENG, M. HASHIGUCHI, K. IIGUSA et T. OHIRA : Electronically steerable parasitic array radiator antenna for omni- and sector pattern forming applications to wireless ad hoc networks. *Microwaves, Antennas and Propagation, IEE Proceedings* -, 150(4) : 203 – 208, 8 2003.
- [CJ03] T. CLAUSEN et P. JACQUET : Optimized Link State Routing Protocol (OLSR). RFC 3626 (Experimental), octobre 2003.
- [CYRV02] Romit Roy CHOUDHURY, Xue YANG, Ram RAMANATHAN et Nitin H. VAIDYA : Using directional antennas for medium access control in ad hoc networks. *Dans MobiCom '02 : Proceedings of the 8th annual international conference on Mobile computing and networking*, pages 59–70, New York, NY, USA, 2002. ACM.
- [DBB<sup>+</sup>05] N. DEPARIS, A. BENDJABALLAH, A. BOE, M. FRYZIEL, C. LOYEZ, L. CLAVIER, N. ROLLAND et P.-A. ROLLAND : Transposition of a baseband uwb signal at 60 ghz for high data rate indoor wlan. *Microwave and Wireless Components Letters, IEEE*, 15(10) : 609–611, octobre 2005.
- [DLRR08] N. DEPARIS, C. LOYEZ, N. ROLLAND et P.-A. ROLLAND : Uwb in millimeter wave band with pulsed ilo. *Circuits and Systems II : Express Briefs, IEEE Transactions on*, 55(4) : 339–343, avril 2008.
- [ETS09] Updated etsi standard promises increased broadband capacity, août 2009. [http://www.etsi.org/Website/NewsandEvents/2009\\_08\\_BROADBAND.aspx](http://www.etsi.org/Website/NewsandEvents/2009_08_BROADBAND.aspx).
- [FZ05] Lu FANG et Huang ZAILU : Low complexity multi-user detectors of th-ppm uwb system. volume 1, pages 320–323, septembre 2005.
- [GCA05] H. GOSSAIN, C. CORDEIRO et D.P. AGRAWAL : Mda : an efficient directional mac scheme for wireless ad hoc networks. *Dans Global Telecommunications Conference, 2005. GLOBECOM '05. IEEE*, volume 6, pages 5 pp. –3637, 2-2 2005.

---

1. Arcep : Autorité de Régulation des Communications Électroniques et des Postes

- [GHMSR06] G. GRIMAUD, A. HONORE, H. MEUNIER et D. SIMPLOT-RYL : Efficient packet scheduler for wireless ad hoc networks with switched beam antennas. volume 1, pages 523 –527, apr. 2006.
- [IEE] Institute of electrical and electronics engineers. <http://www.ieee.org/>.
- [IEE97] 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. *IEEE Std 802.11-1997*, pages i–445, 1997.
- [IEM] Institut d’électronique, de microélectronique et de nanotechnologie. <http://www.iemn.univ-lille1.fr/>.
- [IMSR07] F. INGELREST, N. MITTON et D. SIMPLOT-RYL : A turnover based adaptive hello protocol for mobile ad hoc and sensor networks. *Dans Modeling, Analysis, and Simulation of Computer and Telecommunication Systems, 2007. MASCOTS '07. 15th International Symposium on*, pages 9 –14, 24-26 2007.
- [ISO] Organisation internationale de normalisation. <http://www.iso.org/iso/fr/home.htm>.
- [ITU] Union internationale des télécommunications. <http://www.itu.int/fr/pages/default.aspx>.
- [JBK<sup>+</sup>05] G. JAKLLARI, I. BROUSTIS, T. KORAKIS, S.V. KRISHNAMURTHY et L. TASSIULAS : Handling asymmetry in gain in directional antenna equipped ad hoc networks. *Dans Personal, Indoor and Mobile Radio Communications, 2005. PIMRC 2005. IEEE 16th International Symposium on*, volume 2, pages 1284 –1288 Vol. 2, 11-14 2005.
- [KH99] Young KO et Vaidya N H. : Medium access control protocols using directional antennas in ad hoc networks. Rapport technique, College Station, TX, USA, 1999.
- [KJT03] Thanasis KORAKIS, Gentian JAKLLARI et Leandros TASSIULAS : A mac protocol for full exploitation of directional antennas in ad-hoc wireless networks. *Dans MobiHoc '03 : Proceedings of the 4th ACM international symposium on Mobile ad hoc networking & computing*, pages 98–107, New York, NY, USA, 2003. ACM.
- [KT75] L. KLEINROCK et F. TOBAGI : Packet switching in radio channels : Part i–carrier sense multiple-access modes and their throughput-delay characteristics. *Communications, IEEE Transactions on*, 23(12) : 1400 – 1416, décembre 1975.
- [LR99] Joseph C. LIBERTI et Theodore S. RAPPAPORT : *Smart Antennas for Wireless Communications : IS-95 and Third Generation CDMA Applications*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 1999.
- [MB83] Robert M. METCALFE et David R. BOGGS : Ethernet : distributed packet switching for local computer networks. *Commun. ACM*, 26(1) : 90–95, 1983.
- [Moy91] J. MOY : OSPF Version 2. RFC 1247 (Draft Standard), juillet 1991. Obsoleted by RFC 1583, updated by RFC 1349.
- [PBRD03] C. PERKINS, E. BELDING-ROYER et S. DAS : Ad hoc On-Demand Distance Vector (AODV) Routing. RFC 3561 (Experimental), juillet 2003.
- [Rap01] Theodore RAPPAPORT : *Wireless Communications : Principles and Practice*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 2001.
- [Rob75] Lawrence G. ROBERTS : Aloha packet system with and without slots and capture. *SIGCOMM Comput. Commun. Rev.*, 5(2) : 28–42, 1975.
- [Sim94] W. SIMPSON : The Point-to-Point Protocol (PPP). RFC 1661 (Standard), juillet 1994. Updated by RFC 2153.

- [TBW07] M. TAKATA, M. BANDAI et T. WATANABE : A mac protocol with directional antennas for deafness avoidance in ad hoc networks. *Dans Global Telecommunications Conference, 2007. GLOBECOM '07. IEEE*, pages 620 –625, 26-30 2007.
- [TK75] F. TOBAGI et L. KLEINROCK : Packet switching in radio channels : Part ii—the hidden terminal problem in carrier sense multiple-access and the busy-tone solution. *Communications, IEEE Transactions on*, 23(12) : 1417 – 1433, décembre 1975.
- [TNT<sup>+</sup>06] Y. TAKATSUKA, K. NAGASHIMA, M. TAKATA, M. BANDAI et T. WATANABE : Wsn07-6 : A directional mac protocol for practical smart antennas. *Dans Global Telecommunications Conference, 2006. GLOBECOM '06. IEEE*, pages 1 –6, nov. 2006.
- [TNW04] M. TAKATA, K. NAGASHIMA et T. WATANABE : A dual access mode mac protocol for ad hoc networks using smart antennas. *Dans Communications, 2004 IEEE International Conference on*, volume 7, pages 4182 – 4186 Vol.7, 20-24 2004.
- [WMB<sup>+</sup>07] M. WATANABE, H. MITSUHASHI, M. BANDAI, S. OBANA et T. WATANABE : Empirical discussion on directional mac protocols for ad hoc networks using practice smart antennas. *Dans ICC '07. IEEE International Conference on Communications*, pages 3642 –3647, 24-28 2007.
- [Won10] A. WONG : Case study : Simulated deployment of a mesh network in honolulu. pages 1 –9, jan. 2010.
- [Zim80] H. ZIMMERMANN : Osi reference model—the iso model of architecture for open systems interconnection. *Communications, IEEE Transactions on*, 28(4) : 425–432, avril 1980.



Annexe A

---

# Algorithmes de la couche MAC

```

state = MODE_BD ;
sc = 0 ;
while true do
  switch state do
    case MODE_BD
      if  $T_{SC} > TBW()$  or count(Queue sc) = 0 then
        packet ← OldestPacketOf(AllQueues) ;
        if packet = 0 and isRTSPending() then
          | HandleRTS() ;
        end
        else if packet ≠ 0 then
          if AgeOf(packet) <  $\Delta$  and isRTSPending() then
            | HandleRTS() ;
          else
            if SectorOf(packet) = sc then
              | AggregateAndSend(packet) ;
            else
              | SwitchToSectorOf(packet) ;
              | sc ← SectorOf(packet) ;
            end
          end
        end
      end
    else
      packet ← OldestPacketOf(Queue sc) ;
      if AgeOf( $T_P$ ) <  $\Delta$  and isRTSPending() then
        | HandleRTS() ;
      else
        | AggregateAndSend(packet) ;
      end
    end
  endsw
  case MODE_HD_DATA
    | ModeHDDATA() ;
  endsw
  case MODE_HD_CTS
    | ModeHDCTS() ;
  endsw
  case MODE_HD_ACK
    | ModeHDACK() ;
  endsw
endsw
end

```

Algorithme 1 : Boucle principale de l'ordonnanceur

```

if SectorOf(RTS) = sc then
    | SendCTS();
    | state ← MODE_HD_DATA;
else
    | SwitchToSectorOf(RTS);
    | sc ← SectorOf(RTS);
end

```

**Fonction** HandleRTS

```

forall the packet p of Queue sc do
    | if DestinationOf(p) = DestinationOf(packetToSend) then
    | | nbBytesToSend ← nbBytesToSend + SizeOf(p);
    | | add p to send list;
    | end
end
if nbBytesToSend > threshold then
    | // Send with the high bitrate channel
    | SendRTS();
    | state ← MODE_HD_CTS;
else
    | // Send with the low bitrate channel
    | SendData(packetToSend);
end

```

**Fonction** AggregateAndSend

```

if CTSReceived() then
    | // Send with the high bitrate channel
    | SendData();
    | state ← MODE_HD_ACK;
else
    | if IsTimeoutExceeded() then
    | | Failure();
    | | state ← MODE_BD;
    | end
end

```

**Fonction** ModeHDCTS

```

if ACKReceived() then
    | Success();
    | state ← MODE_BD;
else
    | if IsTimeoutExceeded() then
    | | Failure();
    | | state ← MODE_BD;
    | end
end

```

**Fonction** ModeHDACK

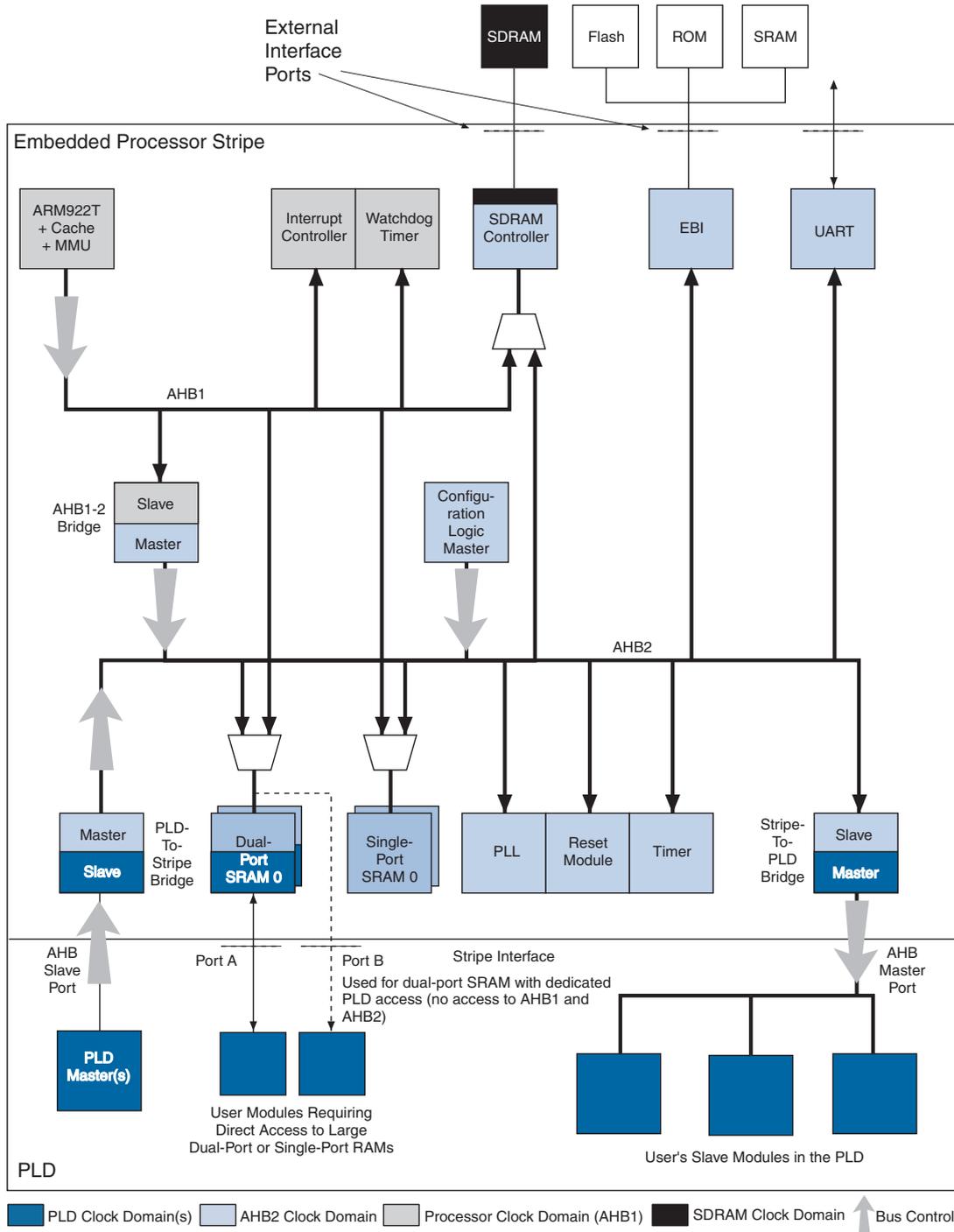
```
if DATAreceived() then
  SendACK();
  state ← MODE_BD;
else
  if IsTimeoutExceeded() then
    Failure();
    state ← MODE_BD;
  end
end
end
```

**Fonction** ModeHDDATA

Annexe B

---

## Structure de la puce *Excalibur*





## Résumé

Les travaux présentés dans ce mémoire se placent dans le contexte des réseaux sans fil sans infrastructure (*Ad Hoc*) impliquant l'utilisation d'antennes sectorielles simples dans le canal radio à 60 GHz. L'antenne est constituée de plusieurs antennes simples, chacune destinée à un secteur particulier et susceptible d'être commutée mécaniquement pour activer un seul secteur. Il est ainsi possible de choisir le secteur d'émission ou de réception des données.

Ce type d'antenne souffre néanmoins d'un problème lié à cette commutation mécanique : elle ne peut pas se faire instantanément. À chaque changement de secteur, une petite quantité de temps est perdue, ce qui diminue l'efficacité de l'ensemble. Ce problème est d'autant plus important lorsque le temps de commutation avoisine le temps d'émission d'un paquet : plus de la moitié de la bande passante peut être gâchée. Les solutions de l'état de l'art permettant de supporter les antennes sectorielles ne prennent généralement pas en compte ce temps de commutation et ne sont donc pas adaptées au problème.

La thèse défendue est qu'en prenant en compte ce temps de commutation, il est possible de produire une pile protocolaire performante et adaptée à ce type d'antennes. Nous proposons donc une nouvelle pile protocolaire optimisée comprenant les deux couches basses du modèle OSI. Elle reste donc interopérable avec les systèmes actuels.

