HAL
open science

# Analyse des performances d'un réseau de capteurs exploitant le standard IEEE 802.15.4

Mohamed Nazim Abdeddaim

## To cite this version:

Mohamed Nazim Abdeddaim. Analyse des performances d'un réseau de capteurs exploitant le standard IEEE 802.15.4. Autre [cs.OH]. Université de Grenoble, 2012. Français. NNT : 2012GRENM067 . tel-00767883

**HAL Id: tel-00767883**

**https://theses.hal.science/tel-00767883v1**

Submitted on 20 Dec 2012

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

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

UNIVERSITÉ DE GRENOBLE

# THÈSE

Pour obtenir le grade de

## DOCTEUR DE L'UNIVERSITÉ DE GRENOBLE

Spécialité : **Informatique**

Arrêté ministériel : 7 aout 2006

Présentée par


## Nazim Abdeddaim


Thèse dirigée par **M. Andrzej Duda**
et codirigée par **M. Fabrice Theoleyre**

préparée au sein **Laboratoire d'Informatique de Grenoble**
et de **Ecole Doctorale Mathématiques, Sciences et Technologies de l'Information, Informatique**

# Analyse des performances d'un réseau de capteurs exploitant le standard IEEE 802.15.4

Thèse soutenue publiquement le **05 Octobre 2012**,
devant le jury composé de :

**Professeur Vivien Quéma**
Grenoble INP–Ensimag, Examinateur
**Docteur Nathalie Mitton**
Lille INRIA-FUN, Rapporteur
**Professeur André-Luc Beylot**
Toulouse IRIT–ENSEEIHT, Rapporteur
**Professeur Andrzej Duda**
Grenoble INP–Ensimag, Directeur de thèse
**Docteur Fabrice Theoleyre**
Chargé de Recherche CNRS–LSIIT, Co-Directeur de thèse
**Docteur Franck Rousseau**
Grenoble INP–Ensimag, Examinateur

Tout d'abord, je remercie mon directeur de thèse Andrzej Duda pour ses précieux conseils, son aide et sa patience durant ma thèse. Je tiens également à exprimer ma profonde gratitude à Fabrice Thèoleyre pour sa disponibilité, ses idées et conseils, ainsi que pour son soutien tout au long de mon travail de thèse. Merci enfin à Monsieur Benoit Ponsard qui a dirigé ma première année de thèse.

Je tiens aussi à remercier Mme Nathalie Mitton et M. André Luc-Beylot d'avoir accepté de rapporter mon travail de thèse, ainsi que M. Vivien Queama d'avoir présidé le jury.

J'adresse mes remerciements à l'ensemble des collègues du bâtiment D et spécialement les squatteurs de la passerelle (Fabien, Benoit, Laurent) ainsi qu'aux membres de l'équipe Drakkar et de l'équipe administrative particulièrement Pascale et Ghislaine. Merci à tous pour toute l'aide et les conseils que vous m'avez apportés.

Martin, Oliver et Franck, ont toujours fait preuve, envers moi, de beaucoup de disponibilité du point de vue professionnel et social. Je les en remercie vivement.

Un grand merci également aux anciens de Drakkar qui sont très vite devenus de très proches amis et avec qui j'ai passé mes meilleurs moments à Grenoble : Vincent, Yan, Noha, Benoit et Christelle.

Un merci particulier à Maru et Michel pour leur amitié, leur disponibilité, leur gentillesse et l'aide qu'il m'ont apporté au moment où j'en avais le plus besoin. J'espère que l'on gardera contact.

Merci à mes colocataires du bureau D317 Isa, Ana, Bogdan : on aura bien déliré dans ce bureau, n'oubliez pas votre rendez-vous de 10h du matin :) et bon courage pour la suite.

Merci à mes voisins du bureau D 326 : mes amis brésiliens Carina et Reinaldo pour leur joie de vivre, rendez-vous très vite à Fortaléza et merci à Maciej pour tous les bons moments passés ensemble.

Je n'oublie pas mes très chers amis Spido et Anso qui ont toujours été présents à mes côtés ainsi que Didine et Sofia mes complices Algériens.

Enfin je tiens à remercier ma famille qui m'a plus que soutenu, mes parents qui ont tout donné pour que je puisse atteindre cet objectif, mon frère et ma belle-soeur pour leur soutien et leur aide depuis mon arrivée en France et enfin ma soeur pour son soutien et ses encouragements durant toutes ces années.

Et pour finir un grand merci à Nat, je ne sais comment t'exprimer ma gratitude. Merci de m'avoir soutenu durant toutes ces années, merci d'y avoir cru aux moments où j'ai le plus douté et de m'avoir permis d'y croire. Je pense que sans toi je n'aurais pas pu aller jusqu'au bout.

# Table des matières

**1 Introduction générale** **1**

**I Etat de l'art** **7**

**2 Couches MAC pour les réseaux de capteurs** **11**
2.1 Introduction . . . . . 11
2.2 Les protocoles synchronisés . . . . . 12
2.3 Les protocoles à préambule . . . . . 14
2.4 Les protocoles hybrides . . . . . 17
2.5 Les protocoles en multi-canal . . . . . 18
2.5.1 Les protocoles avec attribution fixe de canaux . . . . . 18
2.5.2 Les protocoles avec attribution semi-dynamique de canaux . . . 20
2.5.3 Les protocoles avec attribution dynamique de canaux . . . . . 21

**3 Standardisation** **23**
3.1 Introduction . . . . . 23
3.2 IEEE 802.15.4 . . . . . 24
3.3 Architecture . . . . . 25
3.4 La couche Physique dans IEEE 802.15.4 . . . . . 27
3.5 La couche MAC dans IEEE 802.15.4 . . . . . 28
3.5.1 Les trames dans IEEE 802.15.4 . . . . . 28
3.5.2 Modes de fonctionnement dans IEEE 802.15.4 . . . . . 29
3.5.3 Transfert de données dans IEEE 802.15.4 . . . . . 30
3.5.4 Définition de la supertrame dans IEEE 802.15.4 . . . . . 30
3.5.5 Découverte de voisinage et construction de topologie . . . . . 32
3.5.6 Méthodes d'accès au canal . . . . . 35
3.6 Limites dans la norme . . . . . 38
3.6.1 Topologies . . . . . 38
3.6.2 Collisions de Beacons . . . . . 38
3.7 ZigBee . . . . . 44
3.7.1 Adressage . . . . . 44

3.7.2 Routage . . . . . 45

**II Contributions** **47**

**4 MCCT – MULTI-CHANNEL CLUSTER TREE** **49**
4.1 Introduction . . . . . 49
4.2 Etat de l'art sur les protocoles multi-canaux dans IEEE 802.11 . . . . 51
4.3 Problématiques et motivations . . . . . 55
4.4 Propriétés . . . . . 56
4.5 MCCT . . . . . 57
4.5.1 Construction du cluster-tree . . . . . 58
4.5.2 Association et réduction des collisions inter-cluster . . . . . 61
4.5.3 Choix du coordinateur et réduction des collisions intra-cluster . 62
4.5.4 Choix du canal et du slot . . . . . 64
4.5.5 Maintenance du réseau . . . . . 66
4.6 Evaluation des performances . . . . . 66
4.6.1 Topologie aléatoire . . . . . 67
4.6.2 Topologie en grille . . . . . 69
4.6.3 Scalabilité . . . . . 71
4.6.4 Impact de la densité sur les performances . . . . . 71
4.7 Conclusion . . . . . 74

**5 Etude du mécanisme d'accès au canal proposé dans le standard IEEE 802.15.4** **77**
5.1 Introduction . . . . . 77
5.2 Influence des paramètre du CSMAC/CA sur les performances de IEEE 802.15.4 . . . . . 78
5.3 Etude de l'impact des différents paramètres utilisés dans le mode Slotted CSMA/CA sur les performances . . . . . 80
5.3.1 macMaxCSMABackoffs . . . . . 82
5.3.2 macMaxFrameRetries . . . . . 84
5.3.3 BE (Backoff Exponent) . . . . . 84
5.4 Conclusion . . . . . 86

**6 Auto-adaptation des paramètres du CSMA/CA** **89**
6.1 Introduction . . . . . 89
6.2 Auto-adaptation des paramètres du CSMA/CA dans la littérature . . . 92
6.3 Première proposition . . . . . 94
6.3.1 Evaluation de la proposition . . . . . 95
6.3.2 Comportement dynamique de l'auto configuration . . . . . 95

6.3.3 Limites de la proposition . . . . . 98
6.4 Deuxième proposition: SABE (Self Adapting BE) . . . . . 100
6.4.1 Analyse . . . . . 100
6.4.2 Validation par simulations . . . . . 102
6.4.3 SABE (Self Adapting BE) . . . . . 103
6.4.4 Adaptation pour les rafales (bursts) de trafic . . . . . 105
6.4.5 Adaptation des CCAs . . . . . 106
6.4.6 Longest remaining time first (LRTF) vs Short remaining time first (SRTF) . . . . . 107
6.4.7 Evaluation de la proposition . . . . . 107
6.5 Conclusion . . . . . 111

**III Conclusion et perpectives 113**

**7 Conclusion et perspectives 115**
7.1 Conclusions . . . . . 115
7.2 Perspectives . . . . . 116
7.2.1 Proportional fair sharing . . . . . 116
7.2.2 Routage et crosslayer . . . . . 117
7.2.3 Energie sporadique . . . . . 118

**IV Bibliographie 119**

**V Liste des publications scientifiques et brevets 131**

**List of Publications 133**

# Liste des figures


2.1 S-MAC . . . . . 13
2.2 T-MAC . . . . . 14
2.3 D-MAC . . . . . 14
2.4 B-MAC . . . . . 16
2.5 WiseMAC . . . . . 16
2.6 XMAC . . . . . 17

3.1 Topologies supportées dans IEEE 802.15.4 . . . . . 26
3.2 Structure de la supertrame dans IEEE 802.15.4 . . . . . 29
3.3 CSMA/CA SLotted et Unslotted dans IEEE 802.15.4 . . . . . 37
3.4 L'incoming et l'outgoing supertrame dans IEEE 802.15.4 . . . . . 39
3.5 Collisions de Beacon directes et indirectes . . . . . 39
3.6 MeshMAC . . . . . 42
3.7 Beacon Only Period (BOP) . . . . . 43

4.1 Nœud caché, exposé et sourd . . . . . 51
4.2 Architecture du réseau multi-canal . . . . . 59
4.3 Découverte de voisinage . . . . . 61
4.4 Packet Delivery Ratio dans un réseau en étoile IEEE 802.15.4 . . . . 63
4.5 Packet Delivery Ratio dans une topologie aléatoire de 60 nœuds . . . . 67
4.6 Jain Index dans une topologie aléatoire de 60 nœuds . . . . . 68
4.7 Délai dans une topologie aléatoire de 60 nœuds (échelle logarithmique) . 69
4.8 Packet Delivery Ratio pour une topologie en grille . . . . . 69
4.9 delai pour une topologie en grille (échelle logarithmique) . . . . . 70
4.10 Scalabilité . . . . . 70
4.11 Exemple des topologies en grille, en ligne et en croix . . . . . 71
4.12 Impact de la densité de coordinateurs sur le Packet Delivery Ratio . . . 72
4.13 Impact de la densité de coordinateurs dans une topologie en croix . . . 73
4.14 Impact de la densité de coordinateurs dans une topologie en grille . . . 73
4.15 Impact de la densité de coordinateurs dans une topologie en ligne . . . 73

5.1 Impact du trafic sur le PDR . . . . . 81
5.2 Impact du paramètre macMaxCSMABackoffs sur le PDR . . . . . 81

5.3 Nombre d'abandons de transmission dus aux paramètres macMaxCSMABackoffs et macMaxFrameRetries . . . . . 83
5.4 Impact du paramètre macMaxFrameRetries sur le PDR . . . . . 83
5.5 Impact du paramètre BE sur le Packet Delivery Ratio . . . . . 84
5.6 Impact du nombre de nœuds sur le débit . . . . . 85
5.7 Impact du paramètre macMinBE sur le débit . . . . . 86

6.1 Fonctionnement du CSMA/CA dans IEEE 802.11 et IEEE 802.15.4 . . 91
6.2 Débit dans un réseau en étoile en régime de trafic saturé . . . . . 97
6.3 Debit dans un contexte de réseau multisaut avec un trafic CBR . . . . 98
6.4 PDR dans un contexte de réseau multisaut avec un trafic CBR . . . . 99
6.5 Nombre de slots libres en fonction de la taille de la fenêtre de contention 102
6.6 Probabilité de collision en fonction de la taille de la fenêtre de contention 103
6.7 Débit en réception dans un contexte de trafic saturé . . . . . 106
6.8 Débit dans un cluster de 4 nœuds autour du coordinateur un trafic non uniforme . . . . . 109
6.9 Débit dans un cluster de 16 nœuds autour du coordinateur un trafic non uniforme . . . . . 109
6.10 Débit dans un cluster de 32 nœuds autour du coordinateur un trafic non uniforme . . . . . 109
6.11 Délai avec un trafic non uniforme un trafic non uniforme . . . . . 110
6.12 Délai dans un cluster de 20 nœuds autour du coordinateur un trafic poissonien . . . . . 110
6.13 Délai dans un cluster de 20 nœuds autour du coordinateur un trafic poissonien . . . . . 111

# Liste des tableaux

3.1 Récapitulatif de la couche physique dans IEEE 802.15.4. . . . . 28
3.2 Taille des Supertrames et des slots selon le choix des paramètres BO et SO 33

4.1 Contenu de la trame `hello` . . . . . 63
4.2 Paramètres de simulations . . . . . 66

6.1 Paramètres de la simulation . . . . . 97
6.2 Récapitulatif des probabilités calculées . . . . . 101

# CHAPITRE 1

# INTRODUCTION GÉNÉRALE

### Description

L'avancée technologique et la miniaturisation des composants électroniques ont facilité l'émergence des réseaux de capteurs [1]. Les capteurs (nœuds) formant ce genre de réseaux ont comme particularité leur petite taille ce qui implique des ressources limitées en termes d'énergie et de calcul comparé à un ordinateur par exemple.

Les capteurs ont pour finalité d'être déployés en grand nombre dans un lieu donné afin d'automatiser des mesures environnementales comme la température ou la pression notamment, ou afin de contrôler l'environnement dans des applications de maisons intelligentes par exemple (réglage de la température, de la luminosité, etc.).

Ces derniers sont composés entre autres d'un module de communication et de captage ayant pour rôle de collecter des informations et de les transmettre vers le nœud qui sera chargé de les traiter (serveur central, contrôleur, etc.).

### Applications

Le coût de production de ces dispositifs étant faible, un déploiement massif des réseaux de capteurs devrait être observé dans les années à venir. De nombreuses applications ont déjà été définies pour ce genre de réseaux, parmi lesquelles nous pouvons citer:

- La surveillance de l'environnement (météo, agriculture, etc.) [2–4]
- Les applications militaires [5]
- Les applications liées au bâtiment (maison intelligente, sécurité, etc.) [6, 7]
- Les applications médicales [8, 9]
- Les applications industrielles [10, 11]

### La contrainte énergétique dans les réseaux de capteurs

Comme cité ci-dessus, ces réseaux ont un spectre très large en terme d'applications [12], chacune ayant des attentes différentes en terme de performances, ce qui implique des contraintes très différentes en terme de réactivité (latence tolérée), d'intensité de trafic, de sécurité, de densité etc. Le dénominateur commun entre toutes ces applications est la nécessité d'avoir une bonne fiabilité tout en minimisant les coûts énergétiques et maximisant la durée de vie du réseau.

Sachant que dans la plupart de ces applications les nœuds n'auront pas une quantité inépuisable d'énergie, ils seront dans la majorité des cas alimentés sur batterie ou équipés de modules de récupération d'énergie (vibration, lumière, chaleur ...). Néanmoins la quantité d'énergie récupérée ne sera pas suffisante et de ce fait des efforts doivent être faits au niveau protocolaire afin d'économiser cette ressource [13].

L'économie d'énergie sera donc l'enjeu le plus important pour l'émergence et la popularisation de cette technologie [14].

L'apparition de ces réseaux a changé la manière de penser les réseaux: l'objectif principal n'est plus de maximiser le débit, mais de maximiser la durée de vie du réseau grâce à des protocoles capables de fonctionner avec de faibles consommations énergétiques et de calcul tout en restant fiables.

Afin de prendre en compte cette contrainte il faut adapter les protocoles existants ou en développer de nouveaux. Dans les réseaux de capteurs le design des protocoles est intimement lié aux contraintes applicatives [15]. Les applications pour ce genre de réseaux sont tellement nombreuses qu'il faudra en tenir compte dans la conception de ces protocoles et la couche MAC aura un impact des plus importants. Etant chargée de l'accès au canal, elle aura pour rôle de contrôler le module radio, et donc d'en réduire au maximum l'usage, puisqu'il représente l'une des principales sources de dépense énergétique [16, 17].

La couche MAC dans un réseau de capteurs devra donc être développée dans l'objectif de réduire au maximum le coût énergétique. Une couche MAC idéale, adaptée à toutes les applications citées ci-dessus, sera difficile à réaliser. Ainsi il faudra faire des compromis sur les performances que peuvent offrir ce genre de réseaux (réactivité, délais, débit, durée de vie, etc.).

Comme facteurs de consommation énergétique devant être réduits, ou idéalement évités dans les réseaux de capteurs, on dénombre :

- Les collisions qui surviennent lorsque deux stations ou plus, se trouvant à portée radio les unes des autres, émettent des paquets au même moment, impliquant la perte des paquets et leur réémission

- L'overhearing qui peut être défini comme la réception par les nœuds de paquets ne leur étant pas destinés

- L'overhead représenté par la quantité de trafic de contrôle échangés dans le réseau.
- L'écoute passive (idle listening) qui représente le temps où un nœud écoute le canal sans qu'il y ait de paquet à son adresse
- L'over-emitting que l'on peut définir comme étant l'émission de paquets vers un nœud ayant sa radio éteinte

### Les types de trafic dans les réseaux de capteurs

Dans un réseau de capteurs, différents types de trafic peuvent être rencontrés et sont divisés en trois catégories:

- La première catégorie, qui peut être considérée comme le trafic le plus répandu dans les réseaux de capteurs est le trafic dit convergecast (Multi-point to Point): les données sont générées par les nœuds dans le réseau et envoyées vers un nœud central du réseau appelé puits (Sink) afin d'y être traitées
- La deuxième catégorie est celle où le trafic est généré par le puits et envoyé aux nœuds du réseau (Point to Multi-point)
- Et pour finir, la dernière catégorie est la catégorie Point to Point qu'on peut résumer comme l'échange de données entre deux nœuds quelconques du réseau

Dans ces différentes catégories le trafic peut être de type périodique ou non périodique. Dans les réseaux de collecte d'informations par exemple, le type de trafic le plus répandu sera le trafic périodique: chaque nœud après mesure périodique, mesure de la consommation électrique dans un bâtiment par exemple, doit envoyer un paquet. Un autre type de trafic assez représentatif des réseaux de capteurs est le trafic dit événementiel. Nous le retrouvons par exemple dans des applications liées à la surveillance d'une zone donnée. Les capteurs ne transmettent les données qu'après détection d'un événement, par exemple la détection de fuites dans une centrale nucléaire ou pétrolière. Ce trafic sera sous forme de burst généré à la détection d'un événement. Le troisième type est le trafic à la demande, où un nœud n'envoie les informations qu'à la demande (réception de requête) : nous pouvons citer par exemple les applications de domotique ou de télé surveillance.

### Motivations de ce travail de thèse

Dans ce travail de thèse nous nous focaliserons principalement sur la couche MAC du standard IEEE 802.15.4. Nous sommes partis du constat qu'un grand nombre de travaux ont été effectués dans le domaine des couches MAC des réseaux de capteurs, des centaines d'articles ont été publiés, des centaines de nouveaux protocoles ont vu le jour, parmi lesquels on dénombre:

- Des protocoles se basant sur la synchronisation
- Des protocoles évitant la synchronisation et utilisant la technique d'échantillonnage de préambule
- Des protocoles mixant les deux techniques de synchronisation et d'échantillonnage de préambule appelés les protocoles hybrides
- Des protocoles utilisant la technique de FDMA (Frequency Division Multiple Access)

Il est vrai que ces protocoles ont tous apporté à leur manière des solutions aux problèmes imposés par les contraintes liées aux réseaux de capteurs, mais le constat que nous faisons ici est que malgré toutes ces propositions plus originales les unes que les autres, aucun consensus n'en est sorti. Notre opinion est que le standard IEEE 802.15.4 représente le meilleur compromis pour les réseaux de capteurs. Néanmoins la couche MAC de ce standard n'est pas parfaite, bien au contraire.

Ce standard a été proposé afin de gérer des réseaux tels que les réseaux de capteurs. Dans cette thèse nous nous sommes proposés d'étudier en profondeur le standard IEEE 802.15.4 et plus exactement la couche MAC proposée dans ce dernier. De notre point de vue d'importants efforts doivent être fait dans les couches basses du modèle OSI, car sans une couche physique et une couche MAC fiables, toutes les optimisations proposées pour les couches supérieures resteront vaines.

L'auto-organisation et l'adaptabilité aux contraintes inhérentes aux capteurs est plus que primordiale, sachant que ces réseaux ont comme vocation d'être autonomes et de fonctionner plusieurs années sans intervention externe.

Le but de cette thèse est de proposer des solutions permettant l'auto-adaptation et l'auto-organisation des nœuds formant le réseau aux conditions imposées par leur environnement telles que la densité et les interférences. Les optimisations apportées dans ce travail de thèse ont comme finalité de permettre une bonne gestion des ressources disponibles tant au niveau du spectre radio que de l'utilisation efficace du canal de communication.

### Organisation du manuscrit

Le manuscrit est organisé en une introduction, deux parties découpées en six chapitres et une conclusion.

La première partie nommée état de l'art se compose de deux chapitres: le premier chapitre a pour but d'introduire les plus importantes couches MAC proposées pour les réseaux de capteurs. Dans le deuxième chapitre nous présenterons le standard IEEE 802.15.4, ainsi que les principales problématiques posées par la couche MAC proposée et une bref description de la norme ZibBee.

La seconde partie de ce manuscrit concerne les contributions apportées dans ce travail de thèse aux différents problèmes rencontrés dans les réseaux de capteurs actuels utilisant le standard IEEE 802.15.4: la gestion des collisions de supertrame et le mécanisme d'accès au canal.

Le chapitre 4 présente l'une de nos contributions. Nous proposons une nouvelle méthode de construction de topologie et d'ordonnancement de Beacons se basant sur l'utilisation des différents canaux mis à disposition par le standard IEEE 802.15.4.

Dans le chapitre 5 nous étudierons en détail la méthode d'accès proposée dans le standard IEEE 802.15.4: nous montrerons que cette méthode d'accès n'est pas adaptée aux réseaux de capteurs à moyenne et forte densité.

Dans le chapitre 6 nous proposerons deux algorithmes faisant évoluer la méthode d'accès CSMA/CA du standard permettant une adaptation aux conditions du réseau en terme de densité de voisinage et d'intensité de trafic offrant des performances optimales.

Enfin dans le dernier chapitre il sera question de tirer les conclusions de ce travail et d'évoquer les perspectives futures en relation avec les développements des technologies sans fil.

# Partie I

# Etat de l'art

## Introduction de l'état de l'art

Les réseaux de capteurs sans fil, aussi appelés WSN (Wirless Sensor Network), sont souvent assimilés aux réseaux Ad Hoc. Il est vrai que dans un certain sens nous retrouverons toutes les problématiques déjà étudiées dans ces derniers. Néanmoins avec des problématiques presque identiques les solutions retenues pour les uns ne peuvent pas être appliquées d'une manière optimale pour les autres [18].

Comme contraintes communes entre les réseaux ad hoc et les réseaux de capteurs on peut citer: la non présence d'infrastructure, la non fiabilité des liens radio ainsi que le problème de l'énergie. Cependant, dans les réseaux de capteurs sans fil, ces contraintes sont plus critiques, du fait de la taille des capteurs. Les réseaux de capteurs souffrent plus de la contrainte énergétique due à la taille de leur batterie, mais aussi des capacités limitées en terme de mémoire et de calculs.

La consommation énergétique dans les réseaux de capteurs représente l'enjeu majeur en termes de recherche et d'innovation. La problématique est donc de réussir à maximiser la durée de vie du réseau tout en offrant une bonne fiabilité.

En plus de la contrainte énergétique, il existe d'autres facteurs tout aussi importants, tels que le faible débit, la portée, la capacité de calcul et la mémoire limitée [19].

Les réseaux de capteurs ont attiré de nombreux chercheurs et industriels ces dernières années. En effet l'apparition de ces réseaux a fait apparaître un nouveau paradigme dans la recherche en réseau. Le changement réside dans le fait que la course au débit n'est plus de mise. On revient à un problème simple de prime abord: comment faire des réseaux fiables avec peu de ressources.

Dans cette partie nous présentons les principales propositions ayant été faites afin de répondre aux problématiques posées par les contraintes citées ci-dessus.

# CHAPITRE 2

# COUCHES MAC POUR LES RÉSEAUX DE CAPTEURS

**Sommaire**


2.1 Introduction . . . . . **11**

2.2 Les protocoles synchronisés . . . . . **12**

2.3 Les protocoles à préambule . . . . . **14**

2.4 Les protocoles hybrides . . . . . **17**

2.5 Les protocoles en multi-canal . . . . . **18**

2.5.1 Les protocoles avec attribution fixe de canaux . . . . . 18

2.5.2 Les protocoles avec attribution semi-dynamique de canaux . 20

2.5.3 Les protocoles avec attribution dynamique de canaux . . . . 21


## 2.1 Introduction

Un grand nombre d'études ont traité les problématiques posées par les réseaux de capteurs et les solutions pouvant être proposées au niveau MAC afin de les résoudre. Les protocoles MAC proposés ont tous eu comme objectif d'apporter des solutions au problème d'énergie dans les réseaux de capteurs se basent tous sur le même principe de duty cycle dans le but de réduire le coût énergétique de fonctionnement du réseau en minimisant l'utilisation de la radio.

La différence entre les propositions que nous allons décrire réside dans la manière de faire communiquer les nœuds et de maintenir le réseau connecté:

- Les protocoles synchronisés où les nœuds ont des périodes de réveils synchronisées

- Les protocoles à préambule où les périodes de réveil ne sont pas synchronisées et utilisent l'envoi de préambule avant l'émission de trame afin d'avertir le voisinage d'une future transmission.
- La famille des protocoles hybrides qui est un mélange entre les deux techniques de protocole synchronisé et à préambule
- Les protocoles utilisant différents canaux afin de paralléliser les communications

## 2.2 Les protocoles synchronisés

Les protocoles synchronisés ont pour but de faire coordonner les phases de réveil des nœuds afin qu'ils se réveillent en même temps et ainsi puissent s'échanger leurs données sans problème de connectivité tout en économisant leur énergie. Plusieurs protocoles utilisant la technique de synchronisation ont été proposés. Les plus importants d'entre eux sont décrits ci-après.

Le protocole S-MAC (Sensor MAC) [20] est l'un des premiers protocoles synchronisés proposés pour les réseaux de capteurs. Tous les nœuds d'un même voisinage partagent la même période de réveil afin d'éviter le problème de surdité. Dans le protocole S-MAC le nœud a deux états possibles (cf. Fig. 2.1).

- Période d'activité (Radio allumée): les nœuds se réveillent afin de s'envoyer dans l'ordre les paquets de synchronisation appelés SYNC puis les trames de données
- Période d'inactivité (Radio éteinte): les nœud éteignent la radio pour économiser la batterie

A l'initialisation les nœuds commencent par écouter le canal pendant une durée déterminée en attente de paquets de synchronisation SYNC.

- Si aucun paquet n'est reçu, le nœud choisit un ordonnancement et le transmet (dans un SYNC) afin de le déclarer à son voisinage direct. Ce paquet contient le temps d'éveil du nœud ainsi que le temps jusqu'au prochain réveil.
- Si au contraire pendant la durée d'écoute, le nœud reçoit un paquet SYNC d'un autre nœud, il suit l'ordonnancement de son voisin et le retransmet à son tour dans un SYNC.
- Si un nœud a déjà adopté un ordonnancement et qu'il reçoit un paquet SYNC d'un autre nœud avec un autre ordonnancement, alors il doit adopter les deux ordonnancements, puis transmettre un paquet SYNC avec ses deux périodes de réveil aux autres nœuds.

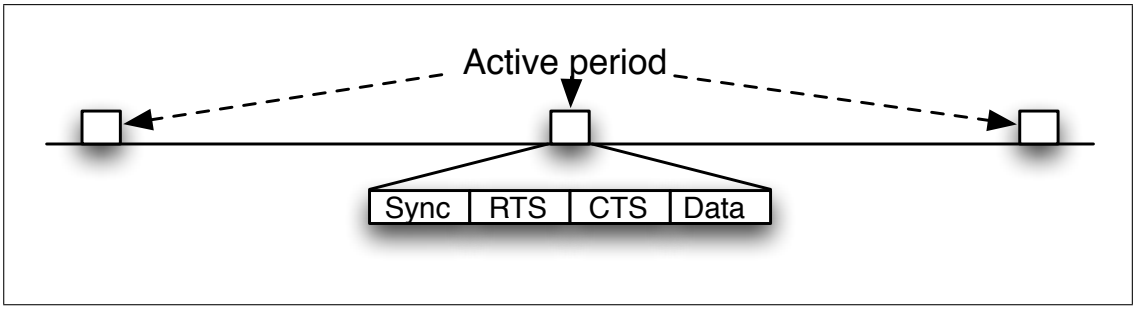


**Figure 2.1:** S-MAC

Le protocole SMAC utilise le mécanisme RTS/CTS afin d'éviter le problème de stations cachées, et le mécanisme CSMA/CA du standard 802.11 [21] est utilisé pour éviter les collisions.

Un mécanisme d'adaption de duty cycle au besoin de communication a été proposé dans DSMAC (Dynamic SMAC) [22] qui est une extension du protocole S-MAC permettant aux nœuds de rajouter des périodes d'activité supplémentaires et ainsi augmenter leur duty cycle. Une autre proposition d'extension de S-MAC a été proposée dans [23] afin de prendre en charge les réseaux de capteurs mobiles. Les périodes d'activité dans ce protocole sont adaptées à la mobilité du voisinage en se basant sur la variation de la puissance du signal des paquets SYNC reçus.

Le protocole T-MAC (Timeout MAC) [24] fonctionne de la même façon que S-MAC avec des périodes d'éveil/endormissement négociées entre les nœuds par le biais des paquets SYNC. La différence entre ces deux protocoles réside dans le fait que le protocole T-MAC prévoit un mécanisme d'adaptation au trafic et ceci dans le but de pallier la rigidité de fonctionnement du protocole S-MAC. T-MAC a pour but de permettre aux nœuds d'augmenter la taille de la période active en cas de besoin, à l'encontre de DS-MAC [22] qui rajoute des périodes d'activités supplémentaires. T-MAC a aussi pour objectif d'éviter de garder la radio allumée alors qu'aucun trafic n'existe sur le canal (cf. Fig. 2.2), ainsi dans T-MAC un nœud pourra éteindre sa radio avant la fin de sa période d'activité s'il n'est pas impliqué dans des communications au bout d'un temps TA (cf. Fig. 2.2).

Le protocole D-MAC [25] a été proposé afin de pallier le problème de latence engendré par l'utilisation des protocoles S-MAC et T-MAC, tout en ayant toujours comme objectif la diminution de la consommation énergétique. Ce protocole a principalement été conçu pour des applications avec un trafic de type convergecast.

Le principe de D-MAC représenté en figure 2.3 est de faire concorder les périodes de réveil le long de l'arbre afin que les données transitent le plus rapidement possible vers le puits. Chaque nœud se voit attribué un slot temporel découpé en deux parties: la première pour la réception de données des fils, et la deuxième pour l'envoi des paquets

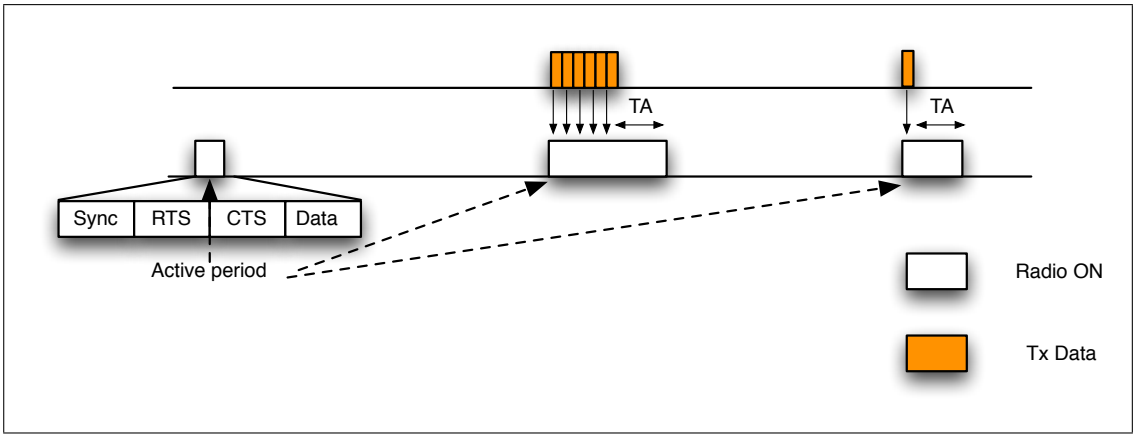


**Figure 2.2:** T-MAC

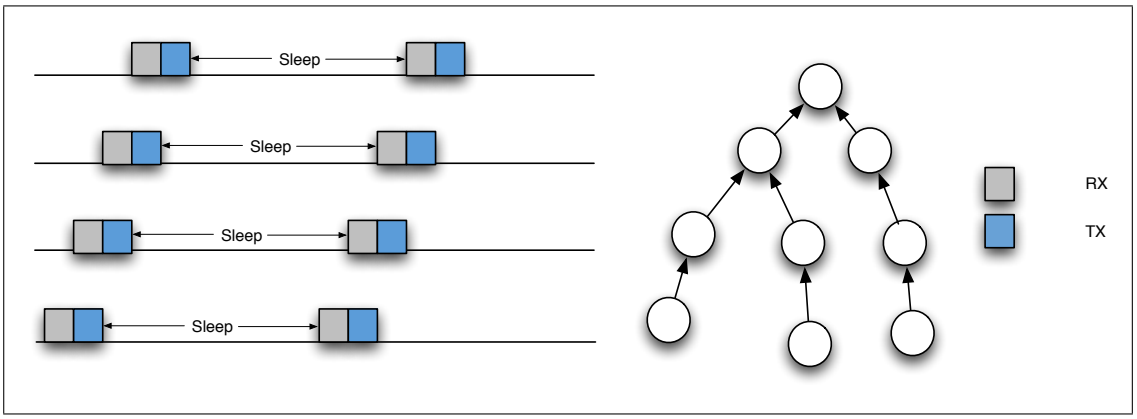


**Figure 2.3:** D-MAC

vers le père. L'avantage de ce protocole réside dans la remontée rapide des données vers le puits, mais aucun mécanisme afin d'éviter les collisions n'a été introduit dans le cas où deux nœuds de différentes branches de l'arbre choisissent le même ordonnancement. Un des inconvénients de ce protocole est qu'en cas de changement de topologie les périodes de communications doivent être recalculées.

## 2.3 Les protocoles à préambule

Les protocoles à préambule ont un mode de fonctionnement complètement différent par rapport aux protocoles synchronisés. Les nœuds utilisant ce genre de protocole choisissent leur temps de réveil indépendamment de leur voisinage, mais avec des périodes

inter-réveil de la même taille. Les nœuds dorment durant la majeure partie de leur cycle et se réveillent périodiquement pour un petit intervalle de temps afin de voir l'état du canal et ainsi prendre la décision de rester éveillé ou de se rendormir. Ayant des périodes de réveil asynchrone afin de rendre la communication possible, les nœuds doivent émettre avant la trame de données un préambule d'une longueur assez grande pour couvrir au moins un cycle complet (réveil/sommeil) et ceci dans le but de s'assurer que le récepteur puisse entendre le préambule et ainsi réceptionner la trame envoyée.

Les inconvénients de ce genre de protocoles sont d'une part un surcoût énergétique dû notamment à l'émission du préambule et d'autre part la consommation énergétique engendrée par l'obligation des nœuds entendant un préambule de rester jusqu'à la fin de ce dernier afin de vérifier si la trame allant être émise leur est adressée.

Plusieurs protocoles ont été proposés afin de pallier le problème du surcoût énergétique imposé par le préambule, tout en optimisant la taille de ce dernier afin de réduire l'attente des nœuds avant réception des paquets. Nous citerons ci-dessous les protocoles plus importants.

Le protocole B-MAC (Berkeley MAC) [26] représente le premier protocole à préambule. Un nœud voulant émettre une trame envoie un préambule signifiant a son voisinage qu'il a des données à transmettre. Le préambule dans B-MAC peut être assimilé à du bruit et ne véhicule aucune information. Même en l'absence de données à émettre, les nœuds se réveillent périodiquement afin de voir s'il n'y a pas de nœuds voulant communiquer. A chaque période de réveil, les nœuds scrutent le canal à la recherche du préambule. S'il n'y a pas de bruit sur le canal, ils éteignent leur radio à la fin de la période de réveil en attendant la prochaine période. Dans le cas contraire s'ils détectent la présence d'un préambule ils devront attendre la fin de ce dernier pour vérifier si la trame annoncée par le préambule leur est destinée. Si le paquet émis ne leur est pas destiné les nœuds se mettront en mode sommeil afin d'économiser leur batterie, sinon ils resteront éveillés jusqu'à la fin de la réception.

L'avantage de B-MAC est qu'il ne nécessite pas de synchronisation entre les nœuds. Le désavantage de ce protocole est que les nœuds émettent un long préambule afin qu'il soit détecté par les autres nœuds, ce qui augmente la latence pour les applications multi-saut et les applications dense en terme de trafic ou de topologie. Un autre problème dans ce protocole est que les récepteurs n'ont aucun moyen de savoir si la trame qui sera émise après le préambule leur est destinée, impliquant que tout nœud détectant le préambule doit rester éveillé pendant toute cette période (overhearing).

Le protocole WiseMAC [27] a été proposé afin de réduire le surcoût énergétique imposé pour chaque émission de trame dû à la longueur des préambules. Ce protocole a été proposé spécialement pour les applications de type trafic descendant.

Dans WiseMac il y a deux genres de nœuds: les point d'accès (Access Point) et les nœuds simples (node). Le rôle des points d'accès est d'apprendre et d'enregistrer les temps de réveil de tout le voisinage, ceci dans le but de réduire la taille des préambules.

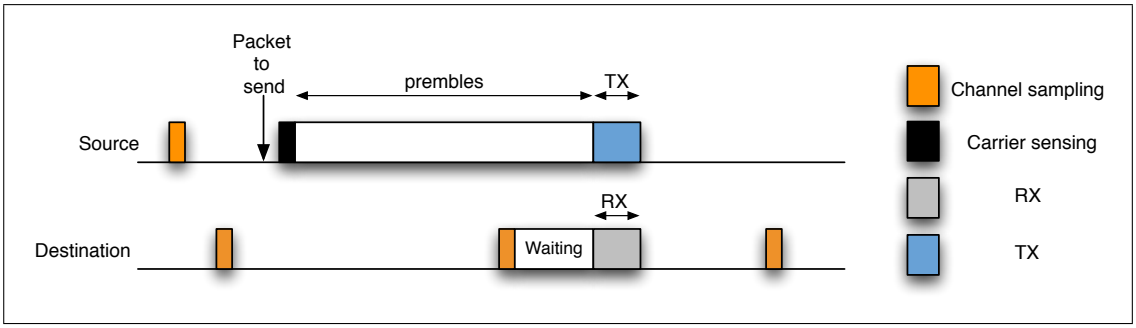


**Figure 2.4:** B-MAC

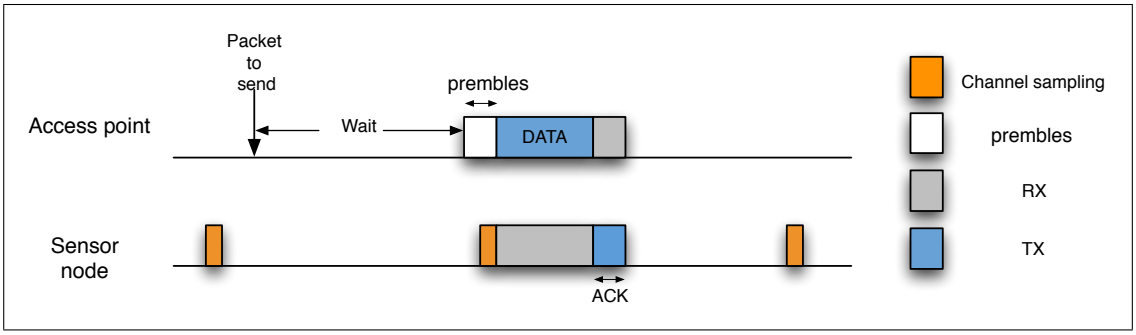


**Figure 2.5:** WiseMAC

En effet si un nœud connaît le moment de réveil du nœud avec qui il veut communiquer il n'aura plus qu'à envoyer un préambule d'une taille réduite servant uniquement à éviter les dérives d'horloges.

Les points d'accès passent pour cela par une phase d'apprentissage du réseau. Au début, la taille du préambule est assez longue pour englober un cycle complet de réveil/-sommeil comme dans B-MAC. Après l'échange avec succès de la trame, le récepteur (nœud destinataire) envoie un acquittement avec son ordonnancement, permettant aux points d'accès d'adapter leur taille de préambule pour les communications futures, connaissant désormais le temps de leur prochain réveil.

D'autres protocoles ont été proposés afin de réduire la consommation énergétique induite par l'envoi et la réception du préambule. Le principe de ces propositions est de découper le préambule en une suite de micro préambules. Plusieurs auteurs ont proposé des protocoles utilisant cette technique [28–30]. Afin d'illustrer le fonctionnement de ces protocoles nous présentons le protocole X-MAC [31], où la solution retenue est l'émission de courts préambules séparés par des pauses assez longues pour permettre à un nœud d'envoyer un acquittement. Dans chacun de ces micro préambules est inscrite l'adresse

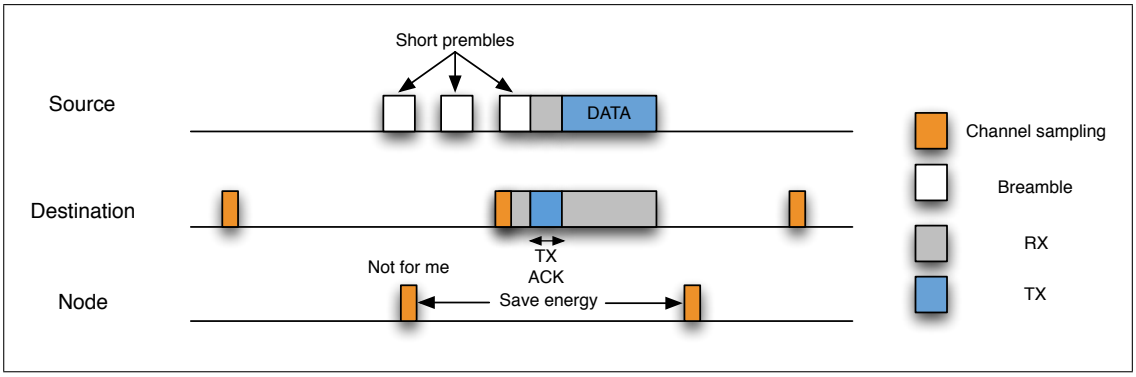


**Figure 2.6:** XMAC

du nœud devant réceptionner les données.

L'intérêt de cette technique est de réduire la consommation énergétique puisque les nœuds n'étant pas concernés par la transmission pourront rapidement éteindre leur radio sans attendre la fin du préambule. La latence sera également réduite grâce à la possibilité pour les nœuds détectant un préambule d'interrompre son émission en envoyant un acquittement signifiant à l'initiateur qu'ils sont prêts à recevoir les données.

## 2.4 Les protocoles hybrides

Les protocoles hybrides sont une combinaison des deux techniques de protocoles synchronisés et à préambule.

Dans SCP [32] le principe de fonctionnement est le même que dans

- B-MAC du point de vue de l'utilisation du préambule
- S-MAC pour la synchronisation

Les nœuds dans SCP ont des périodes de réveil synchronisées, permettant l'utilisation de courts préambules afin d'éviter les dérives d'horloges. Un mécanisme d'adaptation dynamique aux conditions trafic a aussi été rajouté à ce protocole.

Afin de réduire le coût de la synchronisation et d'éviter un overhead trop important dans des applications denses en termes de trafic et de nombre de nœuds, il est prévu que la synchronisation pourra être envoyée dans les paquets de données (piggyback synchronization).

Les auteurs du protocole SCP proposent un mécanisme intéressant pour s'adapter au trafic. Si le nœud détecte une augmentation du trafic, ce dernier ajoutera automatiquement des périodes d'éveil additionnelles durant le cycle en cours. Ainsi, si le trafic doit être relayé et en multi-saut, ce mécanisme permettra d'établir tout le long du réseau

jusqu'à la destination une suite de périodes de réveil des nœuds intermédiaires dans le réseau afin de réduire la latence.

AS-MAC [33] est un autre protocole hybride dans le but de faire de l'économie d'énergie, de réduire la contention et de diminuer le délai.

Durant la période d'association au réseau les nœuds scrutent le canal pendant un cycle complet réveil/sommeil et enregistrent tous les paquets Hello entendus contenant les temps de réveils de leurs voisins. Après s'être associé, le nœud choisit un temps pour sa période de réveil et commence à son tour à distribuer cette information dans ses propres paquets Hello.

Les nœuds comme dans B-MAC se réveillent d'une manière non synchronisée. Mais grâce aux informations contenues dans les Hello les nœuds savent à quel moment ils doivent se positionner afin d'envoyer leur trame au voisin souhaité, et de ce fait n'utiliser que des préambules de courte durée.

## 2.5 Les protocoles en multi-canal

Un certain nombre de protocoles utilisant la technique de multi-fréquence ont été proposés pour les réseaux de capteurs. Ces protocoles peuvent être classifiés en trois grandes catégories: les protocoles avec attribution fixe de canaux, les protocoles avec attribution semi-dynamique de canaux et les protocoles avec attribution dynamique de canaux [34, 35].

Ces protocoles ont été proposés comme solutions non seulement aux collisions mais aussi aux problèmes de réduction de capacité du réseau et dans le but de diminuer les délais.

### 2.5.1 Les protocoles avec attribution fixe de canaux

Le premier protocole multi-canal n'utilisant qu'une interface radio ayant été proposé pour ce genre de réseaux est le protocole MMSN [36]. Chaque nœud du réseau se verra attribuer un canal qui lui servira à la réception des données, les nœuds dans ce protocole sont synchronisés et utilisent un slot de communication divisé en deux partie: la première partie pour le broadcast et la deuxième pour les communications unicast. Ce protocole propose quatre approches pour l'allocation de canaux:

- Attribution des fréquences exclusives (Exclusive Frequency Assignment): les nœuds doivent, afin de choisir leur canal, connaître les identifiants (ID) des nœuds dans leur voisinage à deux sauts. Ces identifiants sont diffusés (brodcast) dans le voisinage en deux étapes. Durant la première phase de brodcast, les nœuds envoient seulement leur ID. Durant la seconde phase ce sont les IDs de leurs voisins qui seront diffusés. L'assignation de canal se fait en fonction des informations recueillies (ID à 2 sauts): le nœud ayant l'ID le plus petit choisira le premier canal libre

et ainsi de suite. Tous les nœuds choisiront les canaux libres dans leur voisinage selon un ordre croissant d'IDs.

- Even-Selection: Si tous les canaux ont déjà été attribués dans un voisinage donné alors qu'il reste encore des nœuds n'ayant pas de canal assigné, ces derniers doivent en choisir un, aléatoirement, dans la liste des canaux les moins utilisés.

- Eavesdropping: Proposée afin de réduire le temps d'attente qu'un nœud doit passer en écoute afin de choisir un canal, cette solution repose sur le tirage d'un temps aléatoire (Backoff) durant lequel les nœuds écoutent le canal, construisent une liste de canaux non utilisés dans leur voisinage à un saut, et choisissent un canal aléatoirement dans cette liste. A la fin de ce temps ils envoient leur décision concernant le canal choisi. Cette technique est moins performante étant donné que les nœuds n'ont aucune information sur le choix de leurs voisins à deux sauts.

- Implicit-Consensus: Comme dans la première stratégie, les nœuds doivent connaître les IDs de leur voisinage à deux sauts. Cette technique d'allocation repose sur un algorithme utilisant un générateur pseudo aléatoire proposé dans [37] qui garantit à chaque nœud l'allocation d'un canal unique si tous le IDs des voisins ont bien été reçus par ce dernier.

Le problème majeur de ce protocole est sa rigidité: les canaux étant attribués après la construction du réseau il faudra recommencer le processus d'attribution des canaux afin d'inclure un nouveau nœud dans le réseau.

Le protocole HyMAC (Hybrid MAC) [38] comme le protocole MC-LMAC combine les deux techniques d'ordonnancement temporel et fréquentiel (TDMA et FDMA). Ce protocole a été développé pour des applications avec trafic convergecast. Le temps dans HyMAC est découpé en trames, découpées elles-mêmes en slots. L'attribution de canal et de slot se base sur l'algorithme BFS (Breadth First Search) et est accomplie par les stations de base. Pour ce faire la station de base recueille les Hello des nœuds voisins et forme un cluster. Les communications dans HyMAC se font toujours entre le nœud et sa station de base sur le canal et le slot lui ayant été assigné. L'avantage de ce protocole est l'utilisation d'un algorithme simple pour l'assignation des slots et canaux (BFS), mais aucun mécanisme de prévention des collisions ni de synchronisation n'est abordé dans HyMAC, ni même l'association d'un nœud au réseau.

Ce protocole a comme avantage d'éviter les collisions, mais en contrepartie il génère un overhead important, sachant que pour l'envoi d'un paquet il faudra toujours que l'émetteur envoie deux paquets de contrôle: l'un durant la CF et le deuxième durant la Split Phase.

Dans [39] Wu *et al.* proposent qu'après sa formation le réseau soit découpé en plusieurs sous arbres. Ces derniers sont tous reliés au puits, et chacun de ces sous arbres se verra attribuer un canal unique afin de réduire les collisions dans le réseau. Le

protocole TMCP (Tree-based Multi-Channel Protocol) utilise un algorithme centralisé: il faut d'abord une phase d'initialisation où l'algorithme Breadth-First search (BFS) est utilisé afin de construire l'arbre. Les nœuds dans cet arbre pourront avoir plusieurs parents. Après la construction de l'arbre, commence la phase d'attribution des canaux. Ceci est fait en commençant par la racine du réseau, et chaque nœud se verra assigné à un sous arbre, en n'autorisant qu'un seul père par nœud tout en s'assurant que chaque nœud sera assigné au sous arbre où il créera le moins d'interférences. A chaque sous arbre sera attribué un canal afin de minimiser les interférences. Le puits devra avoir plusieurs interfaces afin de pouvoir gérer en parallèle les communications de ses fils.

Cependant, le désavantage de ce protocole est qu'il n'est pas distribué et n'apporte des solutions qu'aux collisions que l'on peut appeler les interférences inter-cluster (entre les différents clusters des différents sous arbres). En résumé, il ne solutionnera que les collisions qui s'opèrent entre les fils du coordinateur central du réseau (puits), laissant les collisions intra-cluster et inter-cluster dans une même branche sans solution, puisque chaque sous arbre utilise le même canal de communication. Ce qui nous ramène aux mêmes problèmes de collisions que la version originale de IEEE 802.15.4, comme nous le verrons ultérieurement (c.f 3.6).

### 2.5.2 Les protocoles avec attribution semi-dynamique de canaux

MC-LMAC (Multi-Channel Lightweight MAC) [40] allie les deux techniques TDMA et FDMA. Chaque nœud devra choisir un slot temporel ainsi qu'un canal pour communiquer. Ce protocole se base sur le protocole LMAC [41] conçu pour l'ordonnancement de slot temporel dans les réseaux de capteurs utilisant un canal unique. Chaque nœud dans MC-LMAC aura un slot temporel qu'il occupera exclusivement dans son voisinage à deux sauts. Chaque time slot dans ce protocole est divisé en deux parties. La première partie est dédiée à la signalisation, où tous les nœuds seront invités à positionner la radio sur le canal de contrôle. Un nœud ayant des données à émettre doit inviter le nœud destinataire à le rejoindre sur son canal de communication afin de se communiquer les données, la deuxième partie du time slot est alors utilisée pour le transfert des trames. Même si un nœud n'a pas de donnée à envoyer il devra quand même se mettre sur le canal de contrôle pour voir s'il n'y a pas de nœud voulant communiquer avec lui.

Ce protocole a pour désavantage d'introduire un important overhead: la transmission de chaque trame doit être précédée d'une trame de contrôle.

CMAC [42] est un protocole désynchronisé multi-canal, dont la particularité est l'utilisation de deux interfaces radio: une radio basse consommation appelée LR (Low-power Radio) toujours positionnée sur le canal de contrôle, et servant uniquement à la signalisation (messages de contrôle), et une radio half-duplex utilisée pour l'échange des données, et positionnée sur le canal de communication du nœud. Un nœud voulant communiquer doit utiliser sont interface LR pour inviter la destination à le rejoindre sur son canal de communication afin de lui envoyer ses paquets. L'utilisation des deux

interfaces a pour avantage de résoudre le problème de surdité. Ce protocole présente de très bonnes performances en terme d'économie d'énergie vu que les nœuds n'ont que la radio basse consommation allumée, et ne doivent allumer la deuxième radio que pour échanger des données. Le problème de ce protocole est qu'il a été pensé pour un genre particulier de capteur et ne fonctionne pas avec les capteurs à interface unique. Un autre bémol à ce protocole est que le problème d'allocation des canaux de communication n'a pas été abordé.

### 2.5.3 Les protocoles avec attribution dynamique de canaux

Une autre façon de faire du multi-canal dans les réseaux de capteurs est de ne pas attribuer un canal fixe aux nœuds.

Les auteurs, dans [43], proposent une solution pour les topologies cluster-tree avec le protocole MCMAC: chaque tête de cluster appelé coordinateur aura pour rôle d'attribuer un canal à tous ses nœuds fils voulant communiquer.

Les auteurs ont introduit une supertrame afin de gérer les communications dans le réseau, avec une partie active elle-même découpée en quatre parties et une partie inactive dédiée à l'économie d'énergie des nœuds. La partie active commence toujours par un Beacon envoyé par le coordinateur afin de synchroniser ses filss, la deuxième partie se compose d'autant de time slot que de nœuds fils dans un cluster donné. A chaque nœud est attribué exclusivement un slot, durant lequel tout nœud voulant communiquer devra obligatoirement envoyer une requête de demande de canal à son coordinateur, en utilisant le canal de contrôle. Durant cette période le coordinateur sera en attente des requêtes de ses filss demandant l'attribution d'un canal dédié à l'envoi de leurs paquets. Dans cette requête les nœuds spécifieront entre autres le nœud auquel sont destinés les paquets qui devront être envoyés. Le coordinateur recueillera toutes les requêtes venant de ses fils et si les ressources le permettent (nombre de canaux disponibles) il attribuera un canal à chaque paire source/destination afin qu'ils s'échangent leurs données. Tous les nœuds du cluster doivent écouter le canal de contrôle durant la troisième partie de la supertrame où le coordinateur envoie sous forme de broadcast un paquet contenant la liste des nœuds et canaux attribués. Les nœuds ayant demandé un canal ainsi que les nœuds destinataires recevront le paquet channel assignment et pourront ainsi changer de canal afin de transmettre/recevoir les données sans collision.

Le problème avec MCMAC est que, pour chaque émission de paquet, chaque nœud doit d'abord envoyer une demande d'attribution de canal au coordinateur. Ceci a un double désavantage: le premier est l'augmentation de l'overhead, et le second est le coût énergétique. Afin d'envoyer les requêtes pour les canaux, les nœuds se voient attribuer un slot. Plus il y aura de nœuds dans le cluster plus il y aura de slots réduisant la taille de la période de communication et augmentant les délais.

Le protocole MuChMAC [44] proposé par Borms *et al.* utilise la technique de saut de fréquence (Channel hopping). Le temps est découpé en slot et tous les nœuds changent

de fréquence au début de chaque slot temporel. Afin de calculer le prochain canal que le nœud doit utiliser les auteurs proposent d'utiliser un générateur pseudo aléatoire prenant en entrée l'ID du nœud et le numéro du slot en cours d'utilisation. Afin de réduire la consommation énergétique dans MuchMAC les slots sont à leur tour découpés en sous slot, les nœuds dans ce cas peuvent se réveiller uniquement durant un sous slot.

Y-MAC proposé dans [45] présente deux modes de fonctionnement selon l'intensité de trafic: le mode TDMA si le trafic est faible et le mode FTDMA en introduisant du saut de fréquence pour faire face à des trafics plus soutenus.

La supertrame prévue dans ce protocole est découpée en deux parties: une pour le Broadcast et l'autre pour l'unicast. Chaque nœud se voit attribuer un slot pour la réception et un pour l'émission. Les slots sont attribués de façon à ce que chaque nœud soit le seul qui l'utilise dans un voisinage à deux sauts. L'attribution de slot se fait de la même manière que dans MMSN, avec un échange de paquets dans le voisinage à deux sauts pour savoir quel slot est libre. Cette phase de négociation se fait durant la période de Broadcast.

Pour communiquer ses paquets de données l'émetteur doit attendre le slot de la destination et utiliser le mécanisme CSMA/CA pour se prémunir des collisions. La destination devra obligatoirement être en écoute à chacun de ses slots de réception afin de ne pas rater une hypothétique trame lui étant destinée.

L'inconvénient dans ce protocole est qu'il faut autant de slots que de nœuds dans un voisinage à deux sauts. Le calcul du nombre de slot sera très important: trop de slots augmente la latence, alors que s'il n'y en a pas assez cela augmente le taux de collisions. Afin que les communications soient possibles, tous les nœuds doivent connaître les slots de réception de tous leurs voisins mais aussi la séquence de saut de fréquence du destinataire des paquets (channel hopping) vu que l'émetteur devra suivre cette séquence.

# CHAPITRE 3

# STANDARDISATION

**Sommaire**


3.1 Introduction . . . . . **23**
3.2 IEEE 802.15.4 . . . . . **24**
3.3 Architecture . . . . . **25**
3.4 La couche Physique dans IEEE 802.15.4 . . . . . **27**
3.5 La couche MAC dans IEEE 802.15.4 . . . . . **28**
3.5.1 Les trames dans IEEE 802.15.4 . . . . . 28
3.5.2 Modes de fonctionnement dans IEEE 802.15.4 . . . . . 29
3.5.3 Transfert de données dans IEEE 802.15.4 . . . . . 30
3.5.4 Définition de la supertrame dans IEEE 802.15.4 . . . . . 30
3.5.5 Découverte de voisinage et construction de topologie . . . . 32
3.5.6 Méthodes d'accès au canal . . . . . 35
3.6 Limites dans la norme . . . . . **38**
3.6.1 Topologies . . . . . 38
3.6.2 Collisions de Beacons . . . . . 38
3.7 ZigBee . . . . . **44**
3.7.1 Adressage . . . . . 44
3.7.2 Routage . . . . . 45


## 3.1 Introduction

Depuis leur apparition les WPAN (Wireless Personal Area Network) ont suscité un intérêt croissant de la part du monde de la recherche et de l'industrie, ce qui a donné

lieu à un nombre conséquent de propositions. Les solutions présentées dans le chapitre précédent représentent des solutions intéressantes aux problèmes posés par les contraintes dans les réseaux de capteurs et surtout au problème de la consommation énergétique. Bien que chacune de ces contributions apporte une solution innovante à un problème donné, le fait qu'elles ne soient pas interopérables freine le déploiement de ces réseaux.

Une solution au problème d'interopérabilité dans les nouvelles technologies peut être la standardisation qui permet de créer une base commune de travail pour tout intervenant dans le domaine afin de pouvoir produire du matériel et logiciel compatible et interopérable.

Selon l'article 1 du décret n$^o$2009-697 du 16 juin 2009 relatif à la normalisation : "La normalisation est une activité d'intérêt général qui a pour objet de fournir des documents de référence élaborés de manière consensuelle par toutes les parties intéressées, portant sur des règles, des caractéristiques, des recommandations ou des exemples de bonnes pratiques, relatives à des produits, à des services, à des méthodes, à des processus ou à des organisations. Elle vise à encourager le développement économique et l'innovation tout en prenant en compte des objectifs de développement durable."

Le monde des réseaux et de l'internet compte deux organismes de standardisations très influents: L'IEEE et IETF. Ces deux organismes ont tous deux comme objectif de standardiser:

- les couches basses du modèle OSI (physique et MAC) pour l'IEEE
- Les couches hautes pour l'IETF (réseau, transport, etc.)

Le travail de standardisation a déjà porté ses fruits avec l'internet et la téléphonie qui ont pu se propager et atteindre leur maturité.

Depuis l'apparition des réseaux de capteurs il y a eu plusieurs initiatives de standardisation, propriétaire comme l'alliance ZigBee ou Wavenis, ou non propriétaire comme le standard IEEE 802.15.4 et les RFC de RPL et 6LowPAN.

## 3.2 IEEE 802.15.4

Le groupe IEEE 802 fondé en février 1980, a eu pour tâche de standardiser les couches basses des réseaux locaux (LAN) et métropolitains (MAN) par le biais de liaisons filaires ou sans fil.

Ce groupe a, entre autres, produit des standards incontournables, comme:

- 802.3 : Ethernet
- 802.11 : Wifi
- 802.15.1 et 802.15.3 : Bluetooth

Les deux derniers standards cités ci-dessus traitent des réseaux sans fil, mais comme dit précédemment, avec l'apparition des réseaux Personnel WPAN les contraintes et les besoins ont changé, le principal but n'étant plus d'atteindre des débits toujours plus élevés mais de maximiser la durée de vie des réseaux. De ce fait ces technologies ne peuvent être utilisées pour les réseaux WPAN.

C'est pour cette raison que le standard IEEE 802.15.4 a été proposé. Ce dernier ne traite que des deux premières couches du modèle OSI dans les WPANs.

Ces réseaux ont pour particularité d'être des réseaux sans fil, caractérisés par un faible débit et une faible consommation énergétique. Ils ne couvrent que des distances réduites de l'ordre des dizaines de mètres et ne nécessitent pas ou très peu d'infrastructures.

## 3.3 Architecture

Un WPAN qu'on appellera par la suite PAN, est composé d'un ensemble de nœuds ayant pour rôle de collecter et de transmettre l'information vers un organe central du réseau, qui sera en charge de traiter ces informations ou de jouer le rôle de passerelle (gateway). Ce dernier a pour nom puits (PAN coordinateur) dans la littérature.

Le standard IEEE 802.15.4 définit deux types de nœuds: FFD (Full Fonction Device) et RFD (Reduced Fonction Device) pouvant jouer le rôle de:

- PAN Coordinateur qui est le nœud initiateur du réseau. Il a comme attribution d'initier, de gérer l'ensemble du réseau et de jouer le rôle de passerelle vers l'extérieur

- Coordinateur qui est par définition un FFD et qui aura pour tâche de gérer une zone donnée du réseau (cluster) et de router les paquets de ses fils vers le PAN coordinateur

- Nœuds terminaux qui sont de simples nœuds périphériques (actionneurs, capteurs).

On peut définir les FFD et RFD comme suit:

**Les FFD (Full-Function Devices)** peuvent communiquer avec des FFD et des RFD, et peuvent ainsi jouer le rôle de coordinateur de PAN (puits), de routeur (Coordinateur) ou de simple périphérique (capteur, interrupteur).

**Les RFD (Reduced-Function Devices)** sont des nœuds terminaux utilisés dans des contextes simples comme les interrupteurs ou de simples capteurs. Ces nœuds ne peuvent communiquer qu'avec des FFD (coordinateur) et ne peuvent participer au routage dans le réseau. Ils ne sont ainsi que des nœuds terminaux dans le réseau avec des ressources moindre en comparaison avec des FFD.

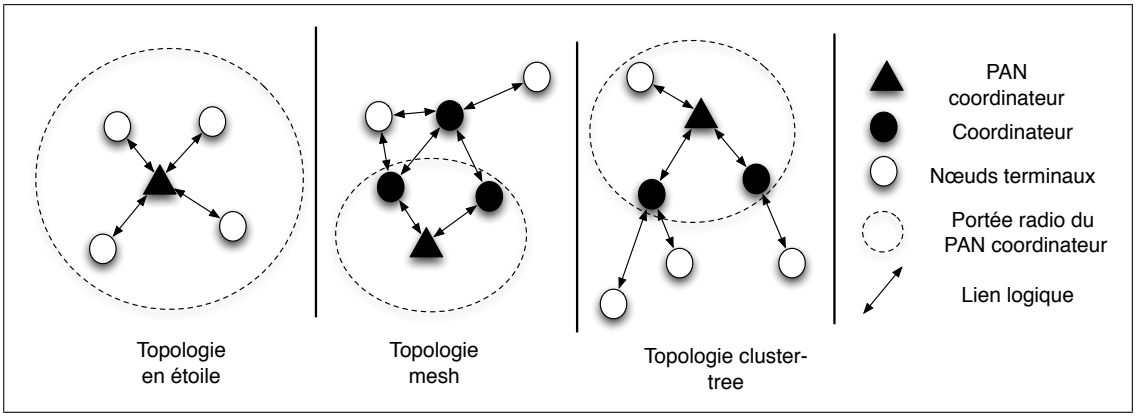


**Figure 3.1:** Topologies supportées dans IEEE 802.15.4

Le standard IEEE 802.15.4 supporte plusieurs types de topologies (figure3.1): la topologie en étoile, la topologie point à point (peer-to-peer) et la topologie en arbre (cluster-tree). Il est possible d'avoir plusieurs réseaux PAN opérant dans la même zone mais fonctionnant indépendamment les uns des autres, chacun utilisant un identifiant de PAN différent.

La topologie en étoile impose des communications s'établissant directement et uniquement entre le nœud qui gère le réseau défini précédemment (PAN coordinateur) et les nœuds se trouvant dans sa portée radio. Le PAN coordinateur sera le nœud initiateur du réseau et tout le trafic devra passer par ce nœud.

Dans la topologie point à point, chaque nœud pourra communiquer avec n'importe quel autre nœud du réseau grâce à la collaboration des nœuds intermédiaires (Coordinateur) sollicités afin de relayer les paquets jusqu'à la destination. Cette topologie permet de réaliser des réseaux beaucoup plus étendus, mais il faudra pour cela ajouter un protocole de routage.

La topologie cluster-tree n'est qu'un cas particulier de la topologie peer-to-peer. La seule différence avec cette dernière est que le réseau sera hiérarchisé avec un PAN coordinateur, des coordinateurs (têtes de cluster) gérant les nœuds qui leur sont connectés, ainsi que des nœuds terminaux communiquant uniquement avec leur coordinateur.

Le standard IEEE 802.15.4 ne définit pas de méthode pour la création de topologie, il définit uniquement les topologies pouvant être utilisées. La méthode de création de topologie est laissée à la libre interprétation du développeur. Néanmoins la méthode d'association des nœuds au réseau a été spécifiée. Cuomo *et al.* dans [46,47] étudient les propriétés (nombre de fils par père, profondeur de l'arbre, taux d'association au réseau, énergie consommée pour l'association etc.) des cluster-trees obtenus avec le mécanisme d'association au réseau fourni dans le standard IEEE 802.15.4. Dans [48] Zheng *et al.* ont étudié en détail le mécanisme d'association au réseau proposé par le standard, et

en ont démontré l'efficacité. Les auteurs ont entre autres étudié l'impact du nombre de coordinateurs ainsi que de la fréquence d'envoi des Beacons sur la formation du réseau.

## 3.4 La couche Physique dans IEEE 802.15.4

L'étude en détails de cette couche étant en dehors du spectre de cette thèse, nous ne détaillerons ici que ses aspects les plus importants. La couche physique a pour rôle de gérer le support physique sur lequel seront faites les transmissions. Elle définit les techniques par lesquelles les bits seront transformés en signaux analogiques et inversement.

La couche physique dans IEEE 802.15.4 offre les fonctionnalités suivantes :

- Gestion de l'activation et de la désactivation du module radio, la radio pourra prendre trois états différents: réception, émission et éteinte.
- La détection de la puissance du signal sur le canal (Energy Detection (ED)).
- La possibilité de remonter à la couche supérieure des informations sur la qualité du lien radio après réception d'une trame : LQI (Link Quality Indication)
- La détection de l'occupation ou non du médium : CCA (Clear Channel Assessment)
- La sélection d'un canal de transmission parmi les différents canaux disponibles

Le standard IEEE 802.15.4 [49] propose quatre différentes couches physiques exploitant chacune des techniques de modulation, d'étalement de spectre ou des bandes de fréquences différentes comme présenté ci-dessous:

- 868/915 MHz utilisant Direct Sequence Spread Spectrum (DSSS) comme technique d'étalement de spectre et la modulation Binary Phase-Shift Keying (BPSK)
- 868/915 MHz utilisant aussi la technique d'étalement de spectre DSSS mais combinée avec une modulation Offset Quadrature Phase-Shift Keying (O-QPSK)
- 868/915 MHz utilisant l'étalement de spectre Parallel Sequence Spread Spectrum (PSSS) et combinant les deux techniques de modulation BPSK et Amplitude Shift Keying (ASK)
- 2450 MHz utilisant DSSS pour l'étalement de spectre et la O-QPSK comme modulation

Le standard offre au total 27 canaux de communication. Selon le choix de la bande de fréquence et de la couche physique on aura des débits allant de 20 à 250 kbit/s ( 3.1)

| Bande de fréquence | Chiprate (kchip/s) | Modulation | Kbit/s | ksymboles/s | Symboles |
|---|---|---|---|---|---|
| 868–868.6 | 300 | BPSK | 20 | 20 | Binary |
| 902–928 | 600 | BPSK | 40 | 40 | Binary |
| 868–868.6 | 400 | ASK | 250 | 12.5 | 20-bit PSSS |
| 902–928 | 1600 | ASK | 250 | 50 | 5-bit PSSS |
| 868–868.6 | 400 | O-QPSK | 100 | 25 | 16-ary Orthogonal |
| 902–928 | 1000 | O-QPSK | 250 | 62.5 | 16-ary Orthogonal |
| 2400–2483.5 | 2000 | O-QPSK | 250 | 62.5 | 16-ary Orthogonal |

**Table 3.1:** Récapitulatif de la couche physique dans IEEE 802.15.4.

## 3.5 La couche MAC dans IEEE 802.15.4

La couche MAC (Medium Acces Control), comme son nom l'indique, aura pour rôle de gérer l'accès au canal avec le mécanisme CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance). Cette couche offrira un mécanisme d'économie d'énergie, une des fonctionnalités les plus importantes dans les réseaux de capteurs sans fil. Nous pouvons résumer ce mécanisme d'une manière très simple: l'utilisation de la radio représentant le plus grand facteur de dépense énergétique doit être limitée au strict minimum en la gardant éteinte la majorité du temps.

La couche MAC offre aussi d'autres fonctionnalités de contrôle liées à:

- La gestion du réseau,
- La gestion des acquittements;
- La gestion de l'adressage (adresse IEEE 802.15.4 codé sur 64 *et* 16 *bits*);
- L'allocation et la gestion des Slots dédiés;
- Le mécanisme de découverte de voisinage;

### 3.5.1 Les trames dans IEEE 802.15.4

Le standard IEEE 802.15.4 propose quatre types de trames pour les échanges entre les nœuds:

- Les trames Beacon (Beacon frame) qui sont envoyées uniquement par les nœuds coordinateurs et servent à administrer le réseau;
- Les trames de données (Data frame) servent au transfert des données utiles entre les nœuds;
- Les trames d'acquittement (Acknowledgment frame) servent à notifier la bonne réception des trames;

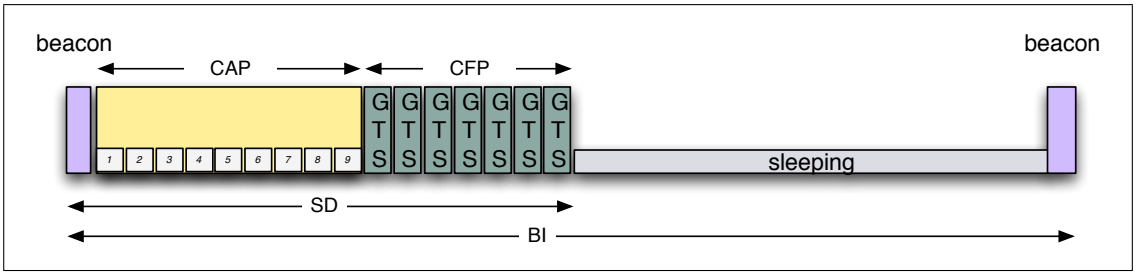


**Figure 3.2:** Structure de la supertrame dans IEEE 802.15.4

- Les trames de contrôle (Command frame) servent à effectuer des demandes spécifiques comme par. exemple l'association au réseau

### 3.5.2 Modes de fonctionnement dans IEEE 802.15.4

Le standard IEEE 802.15.4 offre deux modes de fonctionnement. Le premier est le mode dit non balisé (non Beacon Enabled mode) aussi appelé Unslotted mode où aucun mécanisme de duty cycle (réveil périodique) n'a été spécifié en l'état. Ce mode est plus adapté à des applications où les nœuds n'ont aucune contrainte énergétique (alimenté sur secteur). Dans ce mode il y a absence de synchronisation entre les nœuds du réseau. Afin que ces derniers puissent communiquer entre eux, ils doivent laisser leur radio allumée ou se réveiller périodiquement afin d'interroger le coordinateur pour savoir s'il y a des messages en attente, ce mode étant laissé ouvert par le standard on peut imaginer l'utilisation soit de la technique de préambule soit d'une méthode de synchronisation. Dans ce mode le mécanisme pour se protéger des collisions sera le CSMA/CA Unslotted.

Le deuxième mode appelé mode balisé (Beacon Enabled mode) ou Slotted mode, étant le mode le plus important dans le standard, ses performances en terme de débit, consommation énergétique, packet delivery ratio et fiabilité ont fait l'objet d'un grand nombre d'études [50–53]. Le mode Beacon Enabled est un mode dit synchronisé. Les nœuds doivent suivre une structure périodique appelée supertrame (SuperFrame) représentée sur la figure 3.2.

Cette structure périodique commence toujours par l'envoi d'une trame par le coordinateur appelée Beacon. Cette trame Beacon a pour rôle de synchroniser les nœuds avec leur coordinateur, et fournir toutes les informations vitales au fonctionnement du réseau, comme par exemple l'identifiant du réseau, les données en attente au niveau du coordinateur (communication indirecte), le temps où le prochain Beacon est envoyé et la durée de la partie active de la supertrame, etc.

### 3.5.3 Transfert de données dans IEEE 802.15.4

Au sein des deux modes cités dans la section précédente (non Beacon Enabled et Bacon Enabled), et selon la topologie utilisée, le standard définit trois modes de transfert de données.

**Transmission de données des fils vers le coordinateur (transmission directe)**: Les données sont envoyées de la part des fils vers leur coordinateur. Ce mode est utilisé dans la topologie cluster-tree et la topologie en étoile, si c'est le mode Beacon Enabled qui est utilisé, les nœuds doivent d'abord réceptionner le Beacon du coordinateur avant d'envoyer leurs trames en utilisent le CSMA/CA Slotted durant la partie active de la supertrame pour leur transmission. Si au contraire c'est le mode Non Beacon Enabled qui est utilisé, les nœuds envoient directement leurs données en passant par le CSMA/CA Unslotted. Dans ce cas les coordinateurs devront rester éveillés dans l'impossibilité d'économiser leurs ressources.

**Transmission de données du coordinateur vers ses fils (indirecte)**: Les données sont envoyées par le coordinateur vers ses fils. Ce mode de transmission est aussi utilisé si le réseau est hiérarchisé (topologie cluster-tree ou étoile). Si le réseau utilise le mode Beacon Enabled, le coordinateur doit indiquer qu'il y a des trames en attente pour ses fils dans son Beacon, ces derniers savent alors à la réception de ce dernier qu'ils ont une donnée en attente auprès de leur coordinateur et peuvent ainsi en solliciter la transmission. En mode non Beacon Enabled, en l'absence d'envoi périodique des Beacon, les nœuds demandent périodiquement à leur coordinateur s'il y a des données en attente (polling).

**Transmission de données Pair-à-pair**: Si le réseau est un réseau de type pair-à-pair les nœuds communiquent entre eux en utilisant le CSMA/CA Unslotted. Dans ce cas soit les nœuds restent éveillés tout le temps pour d'éventuelles transmissions, soit ils doivent se synchroniser.

### 3.5.4 Définition de la supertrame dans IEEE 802.15.4

La supertrame est l'un des concepts les plus importants du standard. En résumé la supertrame est une structure temporelle périodique, divisée en unités de temps élémentaires de vingt symboles appelées aUnitbackoff-period. La taille de la supertrame n'est pas figée et selon le choix de certains paramètres on pourra en modifier la taille. La supertrame quelle que soit sa longueur est divisée en 16 slots de taille identique. Ces slots temporels sont composés d'un certain nombre de Unitbackoff-period, la taille de ces slots augmentera avec l'augmentation de la taille de la supertrame 3.2.

La supertrame revient périodiquement afin de fournir un mécanisme de synchronisation aux nœuds qui doivent passer la majeure partie de leur temps la radio éteinte, et ceci pour réduire le coût de consommation énergétique tout en évitant les problèmes

de connectivité liés aux réveils décalés des nœuds. La supertrame se compose des trois parties suivantes (cf. Fig. 3.2):

- Un Beacon (Balise)
- La partie active de la supertrame où les nœuds peuvent avoir la radio allumée
- La partie inactive où les nœuds devront éteindre leur radio afin de faire des économies d'énergie

A chaque réception d'une balise, tous les dispositifs sont informés de la durée de la supertrame. L'envoi régulier de balises permet d'informer l'ensemble des dispositifs sur d'éventuels changements dans la supertrame et compenser les dérives d'horloges. La partie active de la supertrame est divisée, quant à elle en deux sous-parties :

- La CAP (Contention Access Period) où les nœuds communiquent avec leur coordinateur en utilisant le mécanisme d'accès au canal CSMA/CA slotté
- La CFP (Contention Free Period) étant une partie optionnelle de la supertrame où un nœud pourra demander l'attribution exclusive d'un slot appelé GTS

Les communications entre nœuds et coordinateurs se font pendant la CAP en utilisant le mécanisme d'accès au canal CSMA/CA Slotté (Slotted-CSMA/CA).

Durant la CFP, si un nœud s'est déjà vu attribuer un slot il n'aura aucun besoin de CSMA/CA puisque ce slot lui est attribué d'une manière unique. Les GTS ont été proposés afin de permettre à des nœuds ayant des contraintes temporelles de réserver un slot [54] où il pourra envoyer sa donnée vers le coordinateur sans utiliser le CSMA/CA. Plusieurs GTS peuvent être attribués par nœud. Le nombre de GTS par supertrame ne pourra néanmoins dépasser sept slots qui sont réservés en partant du dernier slot de la partie active de la supertrame.

Afin de se voir allouer un GTS par son coordinateur un nœud doit envoyer une requête GTS Request pendant la CAP en utilisant le CSMA/CA Slotted. Le nœud recevra la réponse à sa demande de GTS dans le Beacon. Le coordinateur devra aussi mettre à jour la taille de la CAP dans le Beacon en indiquant le dernier slot afin que les autres nœuds sachent qu'un GTS a été attribué. Comme présenté précédemment un coordinateur peut allouer au maximum sept GTS de la supertrame (sept slots). Un nœud demandant un GTS se verra allouer l'ensemble du slot et cela même si la longueur de ce dernier est supérieure à ses besoins. Les GTS dépendant de la taille des slots (cf. fig. 3.2) peuvent être une source de gaspillage de bande passante, c'est pourquoi Koubaa *et al.* proposent l'algorithme IGAME (Implicit GTS Allocation Mechanism) [55], qui a pour but de permettre l'utilisation d'un slot GTS par plusieurs nœuds afin d'éviter que les slots ne soient sous exploités.

Le taux de consommation énergétique dépendra des paramètres choisis par les coordinateurs. Cette information est envoyée dans la balise et définit la durée d'éveil et d'endormissement des nœuds. Plus le rapport cyclique est petit plus il y aura un gain d'énergie important. Comme dit précédemment la supertrame commence obligatoirement par l'envoi du Beacon contenant les informations conditionnant la taille de la supertrame: Beacon Order (BO) et Superframe Order (SO). Ces deux paramètres appartiennent à l'intervalle $0 \leq SO \leq BO \leq 14$ pour le mode dit Beacon Enabled ou égal à 15 pour le mode non Beacon .

Avec ces deux paramètres BO et SO envoyés dans le Beacon les nœuds pourront calculer la taille du:

- Beacon Interval (BI : Intervalle inter-balise), représentant la longueur totale de la supertrame ( partie active et partie inactive) et donné par l'équation suivante :

$$BI = aBaseSuperframeDuration * 2^{BO} symbols \tag{3.1}$$

- Superframe Duration représentant la taille de la partie active et calculée comme suit :

$$SD = aBaseSuperframeDuration * 2^{SO} symbols \tag{3.2}$$

Avec :
$aBaseSuperframeDuration = abaseSlotDuration * aNumSuperFrameSlots$
$abaseSlotDuration = 60\ Symbols$
$aNumSuperFrameSlots = 16\ Slots$

### 3.5.5 Découverte de voisinage et construction de topologie

Afin qu'un nœud puisse communiquer avec d'autres nœuds dans un réseau IEEE 802.15.4, ce dernier doit obligatoirement être associé au réseau pour plusieurs raisons:

- Construction de la table de fils par les coordinateurs permettant ainsi l'utilisation de la transmission indirecte des trames;
- Pour des raison de sécurité;
- Pour les besoins de filtrage: chaque réseau aura un identifiant unique codé sur 16 bits (macPANId) utilisé comme règle de filtrage de trames, un nœud recevant un Beacon venant d'un autre réseau pourra l'ignorer grâce au PANId, la réception par un coordinateur d'une trame de données provenant d'un nœud non associé sera ignorée.

| *BO ou SO* | *BI ou SD* | *Slot* |
|---|---|---|
| 0 | 15.36 ms | 0.96 ms |
| 1 | 30.72 ms | 1.92 ms |
| 2 | 61.44 ms | 3.84 ms |
| 3 | 122.9 ms | 7.68 ms |
| 4 | 245.7 ms | 15.36 ms |
| 5 | 491.5 ms | 30.72 ms |
| 6 | 983 ms | 61.44 ms |
| 7 | 1.96 s | 122.88 ms |
| 8 | 3.93 s | 245.7 ms |
| 9 | 7.86 s | 491.5 ms |
| 10 | 15.7 s | 983 ms |
| 11 | 31.4 s | 1.966 s |
| 12 | 62.9 s | 3.932 s |
| 13 | 125.8 s | 7.864 s |
| 14 | 251.6 s | 15.728 s |

**Table 3.2:** Taille des Supertrames et des slots selon le choix des paramètres BO et SO

Ainsi pour qu'un nœud s'associe au réseau il doit d'abord passer par la phase de Scan dans le but de récupérer les informations sur son environnement véhiculées dans les Beacons envoyés par les coordinateurs. Après récupération de ces informations le nœud devra décider du coordinateur ou du PAN à rejoindre, puis passer à la phase d'association proprement dite qui consiste à l'échange entre le nœud et le coordinateur sélectionné de requêtes de type requête d'association (Association Request) et réponse d'association (Association Response).

Quand un coordinateur reçoit une requête d'association il doit d'abord acquitter la bonne réception de cette dernière. Le coordinateur notifie son acceptation de l'association du nœud au PAN en générant une adresse courte codée sur 16 bits puis la renvoie dans la requête Association Response. Le nœud doit attendre la réponse du coordinateur jusqu'à 491,52 ms après l'envoi de sa requête, et ce processus d'association peut de ce fait être étalé sur plusieurs supertrames. Si au terme de ce délai aucune réponse n'a été reçue le nœud pourra faire une nouvelle tentative [56].

Dans le cas où le coordinateur ne peut accepter l'association du nœud, il devra quand même envoyer une Association Response indiquant l'impossibilité de l'association.

L'Association Request ne peut être envoyée qu'en mode indirect. Le processus d'association est terminé uniquement quand le nœud reçoit l'Association Response en réponse à sa demande explicite en utilisant la requête Data Request. La requête Disassociation Request est utilisée dans le cas où un nœud veut quitter le réseau ou qu'un coordinateur veut se séparer d'un de ses fils.

Le standard IEEE 802.15.4 propose plusieurs mécanismes permettant à un nœud de créer ou de rejoindre un réseau existant sous forme de balayage (Scan). Quatre types de

Scan ont été proposés afin de permettre:

- à un futur coordinateur de trouver un canal adéquat pour former son réseau (PAN)
- à des nœuds non associés de rejoindre le réseau
- à un nœud ayant perdu la connectivité avec le réseau de se rattacher

Afin que les nœuds puissent connaître la présence d'un réseau dans leur voisinage, dans IEEE 802.15.4 les coordinateurs associés devront annoncer leur présence en envoyant des trames Beacon. Tous les nœuds (FFD et RFD) seront capables de faire des scans d'orphelins et passifs, mais seuls les FFDs seront habilités à faire, en plus, le scan à détection d'énergie et le scan actif. La durée des Scan à détection d'énergie, actif et passif est de $(aBaseSuperframeDuratioon * (2^N + 1))$ symboles où $N$ appartient à l'intervalle (0 à 14).

Scan à détection d'énergie (ED channel scan): Le Scan à détection d'énergie permet à un nœud de mesurer l'énergie maximale dans chaque canal. Ce scan sera effectué par un PAN Coordinateur potentiel cherchant un canal afin de créer un nouveau PAN.

Scan actif (Active channel scan): Un Scan actif permet à un nœud de trouver un coordinateur transmettant des Beacons. Ce type de scan permet à un futur coordinateur de répertorier les PANID dans le voisinage afin d'en choisir un à son tour pour créer un nouveau PAN. Ce scan peut aussi servir à l'association d'un nœud au réseau PAN, et consiste à envoyer une trame de commande Beacon Request sur chaque canal. Si le coordinateur réceptionnant cette requête utilise le mode Beacon-enabled, il doit ignorer la commande Beacon Request et continuer d'envoyer les Beacons périodiquement au début de sa supertrame. Au contraire, si le coordinateur est dans un mode non Beacon enabled ce dernier devra alors répondre par l'envoi d'une trame Beacon en utilisant CSMA-CA, ce qui fait que cette methode de Scan soit la plus adaptée pour la découverte de voisinage dans le mode non Beacon enabled.

Scan passif (Passive channel scan): Le scan passif a le même but que le Scan actif mais dans ce Scan les nœuds n'auront pas besoin d'émettre de trame Beacon Request. Ce type de Scan servira exclusivement à l'association au réseau et non à la création d'un PAN. Son principe est que le nœud écoutera passivement le canal sélectionné pendant toute la durée du Scan afin de récupérer le maximum de Beacons pour pouvoir prendre une décision sur son éventuelle association au PAN, cette technique est la plus appropriée dans la découverte de voisinage si le mode Beacon Enabled est utilisé.

Scan d'orphelin (Orphan channel scan): Un Scan d'orphelin permet à un nœud ayant été associé à un réseau de se rattacher à ce dernier après perte de la connectivité. Son principe est d'envoyer une trame appelée Orphan Notification sur un ou plusieurs canaux et d'attendre la réception d'une trame Coordinator Realignment, envoyée par le coordinateur reconnaissant le nœud comme un de ses fils.

### 3.5.6 Méthodes d'accès au canal

La Couche MAC de IEEE 802.15.4 propose l'utilisation du mécanisme CSMA/CA comme méthode d'accès au canal afin de se prémunir des collisions. Ce mécanisme d'accès au canal est étudié en profondeur dans le chapitre six.

Le CSMA/CA a pour principe de désynchroniser le temps d'envoi des trames des différents nœuds en utilisant un tirage de temps aléatoire dans un intervalle conditionné par le Backoff Exponent. Après ce temps d'attente les nœuds se doivent de vérifier l'état du canal avant de transmettre.

Dans IEEE 802.15.4 deux versions de CSMA/CA ont été proposées fonctionnant de la même façon à quelques différences près: CSAMA/CA Slotté (Slotted CSMA/CA) utilisée dans le mode Beacon, et CSMA/CA non-slotté (Unslotted CSMA/CA) utilisée dans le mode non Beacon illustré dans la figure 3.3.

Avant d'entamer la description du fonctionnement du CSMA/CA nous devons tout d'abord définir la terminologie utilisée dans IEEE 802.15.4:

- Unit Backoff Period: Unité de temps égale à 20 symboles
- BE: Backoff Exponent initialisé à MacMinBE=3 par défaut
- Backoff: Temps d'attente aléatoire tiré dans l'intervalle :

$$[0, 2^{BE} - 1] * \text{aUnitBackoffperiod} \tag{3.3}$$

- CW (Contention Window): Ce paramètre n'est utilisé que dans le mode Slotted et représente le nombre d'unités de temps après le temps Backoff où le canal doit être libre avant transmission
- NB: Nombre d'essais d'accès au canal pour la transmission en cours, initialisé à zéro
- macMaxCSMABackoffs: Le nombre de fois où le canal est trouvé occupé avant d'abandonner l'émission de la trame, initialisé à 3
- macMaxFrameRetries: Le nombre de retransmissions maximum autorisé pour chaque trame, initialisé à 4

Le Fonctionnement du CSMA/CA slotted est le suivant. Avant tout essai de transmission d'une nouvelle trame les nœuds doivent effectuer les étapes suivantes :

1. Initialisation des paramètres à leur valeur par défaut (CW, NB, BE)
2. Tirage d'un temps aléatoire (Backoff) afin de désynchroniser les transmissions.
3. Les nœuds alignent leur début de Unit Backoff Period sur ceux de la supertrame, tous les nœuds doivent s'assurer qu'il reste assez de temps avant la fin de la supertrame pour le Backoff, les deux CW, l'envoi de la trame, et l'acquittement si ce dernier est demandé.
4. A la fin du temps d'attente aléatoire (Backoff) les nœuds doivent tester l'état du canal en effectuant ce qu'on appelle un CCA (Clear Channel Assessment) qui dure 8 Symboles de temps:
   - Si le canal est détecté occupé, le nœud incrémente toutes les constantes de l'algorithme (NB=NB+1, BE=min(BE+1,macMaxBE) et CW=2).
   - Si NB > macMaxCSMABackoffs e.g le nombre maximum d'essais permis pour la transmission d'une trame est dépassé, le nœud abandonne l'envoi de la trame en cours. Dans le cas contraire le nœud retire un nouveau Backoff afin de retenter l'envoi de la trame (étape 2).
   - Si au contraire le canal est détecté libre le nœud décrémente la valeur de CW (CW=CW-1). Si CW=0 la trame est envoyée au début du prochain Unit Backoff Period, sinon le nœud exécute un nouveau CCA après un Unit Backoff Period

Au contraire, si le non Beacon Enabled mode est utilisé c'est le CSMA/CA Unsllotted qui sera alors utilisé par les nœuds afin d'accéder au canal. Comme pour le CSMA/CA Slotted, les nœuds passeront par les deux premières étapes citées ci-dessus: intialisation de (CW, NB, BE) et tirage du temps de backoff aléatoire. Les seules différences avec le Slotted CSMA/CA résident dans les deux points suivants:

- Les périodes de décrémentation des aUnitBackoffPeriod entre les nœuds en concurrence pour l'accès au médium sont à l'encontre du Slotted CSMA/CA indépendantes dans le temps i.e la décrémentation du backoff commence immédiatement après le temps d'attente aléatoire sans synchronisation sur le unit-backoff-period puisqu'il y a absence de supertrame
- Avant de pouvoir transmettre, l'émetteur doit sonder le canal afin de savoir s'il est libre. En mode Unslotted ceci n'est fait qu'une seule fois, alors qu'en mode Slotted CSMA/CA le canal devra être libre durant deux aUnitBackoffPeriod avant l'émission de la trame

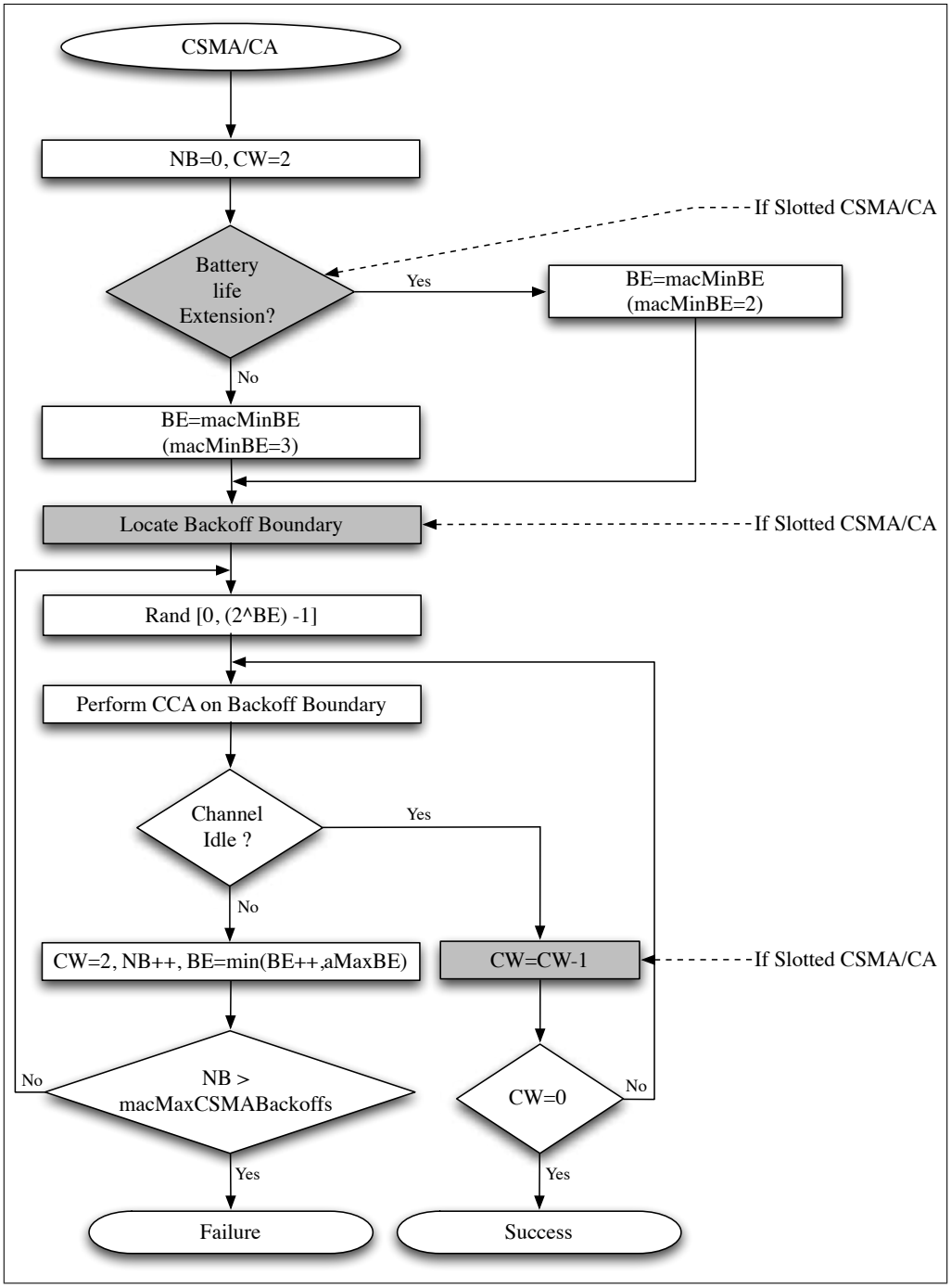


**Figure 3.3:** CSMA/CA SLotted et Unslotted dans IEEE 802.15.4

## 3.6 Limites dans la norme

Toutes les solutions proposées dans le standard IEEE 802.15.4 tendent dans l'absolu à faire des économies d'énergie tout en offrant de bonnes performances. La supertrame a été proposée afin de gérer les communications tout en garantissant aux nœuds d'économiser l'énergie, néanmoins cette structure temporelle introduit plusieurs problématiques relatives au fonctionnement du réseau dont nous discutons ci-dessous.

### 3.6.1 Topologies

Le standard supporte plusieurs types de topologies, mais seule la topologie en étoile peut fonctionner correctement si le réseau a une faible densité en terme de nœuds. Pour ce qui est des topologies peer-to-peer et cluter-tree il faudrait rajouter un certain nombre de mécanismes, tels que la construction de la topologie et la gestion des collisions de Beacons i.e la gestion des interférences entre les différents coordinateurs du réseau.

La solution afin de prendre en compte des topologies en multi-saut a été proposée dans la deuxième édition du standard IEEE 802.15.4 paru en 2006 sous forme d'une nouvelle structure pour la supertrame.

La supertrame d'origine a été découpée en deux parties avec pour seule condition que les deux parties de la supertrame soient séparées par un temps appelé StartTime, et que les valeurs de Beacon Order et Superframe Order soient les mêmes dans tout le réseau. Cette nouvelle supertrame est représentée sur la figure 3.4 et se compose:

- d'une première partie appelée incoming superframe, où le nœud reçoit le Beacon de son coordinateur afin de communiquer avec lui;
- d'une deuxième partie qui s'appelle outgoing superframe où le nœud pourra envoyer son propre Beacon afin de communiquer avec ses fils.

Cette structure permet d'avoir du multi-saut en mode Beacon, mais introduit une nouvelle problématique concernant l'ordonnancement des Beacons des différents coordinateurs du réseau laissée ouverte par le standard.

### 3.6.2 Collisions de Beacons

Le standard dans sa première version (2003) [57] n'offrait aucune solution concrète pour l'exploitation de topologies autres que la topologie étoile. Aucun mécanisme de gestion des collisions de Beacons n'ayant été spécifié à cet effet, en l'état les supertrames des différents coordinateurs pouvaient se superposer [58], collisionner et ainsi paralyser l'ensemble du réseau si celui-ci était un réseau multisaut. Dans la deuxième édition du standard publiée en 2006 [49] a été introduite l'incomming/outgoing superframe pour prendre en compte les topologies avec plusieurs coordinateurs devant collaborer afin de permettre la construction de topologies plus complexes et plus étendues que la topologie

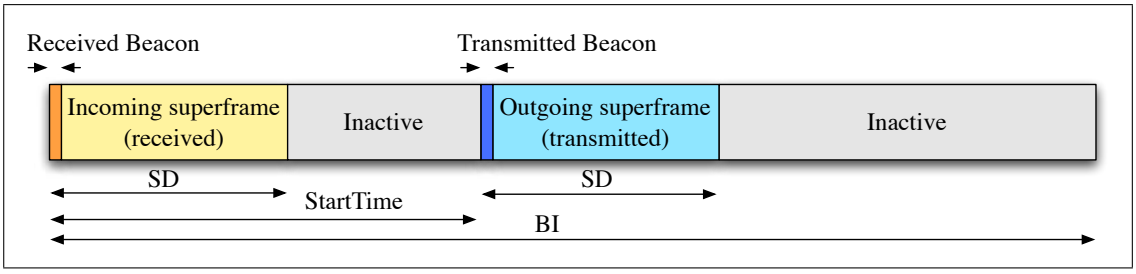


**Figure 3.4:** L'incoming et l'outgoing supertrame dans IEEE 802.15.4

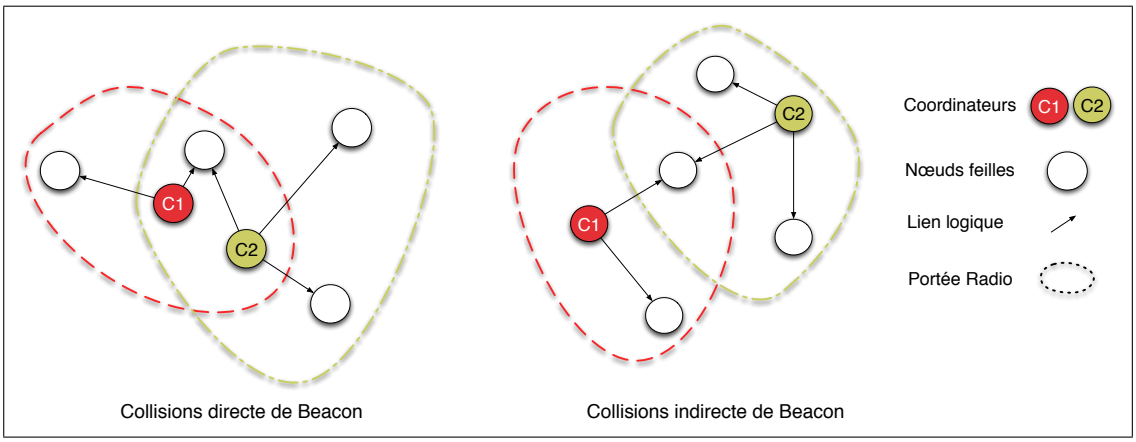


**Figure 3.5:** Collisions de Beacon directes et indirectes

en étoile. Les deux parties de la supertrame sont séparées par le paramètre StartTime qui doit être calculé localement par les coordinateurs, dans le but d'éviter la superposition de supertrames. Cette proposition pose juste un cadre général mais n'apporte pas de solution proprement dite étant donné que le calcul du paramètre StartTime, servant à empêcher les chevauchements des périodes d'activité des cellules adjacentes, n'est pas traité par la norme.

Or, il est facile de noter que l'envoi de balises sans ordonnancement particulier peut avoir comme conséquences des collisions inter-balises. Ainsi les nœuds se trouvant dans la portée radio de plusieurs coordinateurs resteront isolés du réseau.

Deux types de collisions de balises ont été identifiés par le groupe de travail 15.4b illustrés dans la figure 3.5: les collisions de balises dites directes et les collisions dites indirectes. Les Beacons étant envoyés directement, sans mécanisme de protection contre les collisions, si un coordinateur maintient sa supertrame en même temps qu'un coordinateur voisin, il se peut qu'il y ait des collisions entre les Beacons et les trames émises dans les supertrames adjacentes.

La solution la plus naïve à ce problème serait d'avoir un plan de déploiement des capteurs, et d'attribuer les ordonnancements en fonction. Cette solution devrait être écartée dans la majorité des cas à cause des contraintes qu'elle implique: d'un coté il faut utiliser des algorithmes centralisés pour calculer l'ordonnancement, ce qui n'est pas conseillé dans des réseaux distribués, d'un autre coté il faudra recalculer tous les ordonnancements pour ajouter un nouveau nœud dans le réseau. Avoir des algorithmes distribués serait beaucoup plus flexible pour le déploiement et l'ajout de nœuds dans le réseau.

#### Collisions directes de Beacon

On parle de collisions directes de Beacons en présence de plusieurs coordinateurs dans la même portée radio et émettant simultanément les Beacons, impliquant la perte de synchronisation et l'isolation des nœuds fils de ses coordinateurs.

Deux approches ont été retenues par le groupe de travail 15.4b afin d'apporter des solutions aux collisions directes de Beacons.

- The time division approach: son principe est que les coordinateurs étant dans le même voisinage ne partagent pas les même périodes de réveils. Les périodes de réveils sont ordonnancées de telle sorte qu'un coordinateur ne peut avoir sa partie active que pendant la période inactive de ses voisins.

- Beacon only period approach: son principe est d'allouer un intervalle de temps divisé en slots au début de chaque supertrame où chacun des coordinateurs devra choisir un slot libre dans son voisinage afin d'émettre son Beacon sans provoquer de collisions.

#### Collisions indirectes de Beacon

Les collisions indirectes peuvent être assimilées au problème des stations cachées dans les réseaux sans fil. Ces collisions se produisent entre les Beacon émis par des coordinateurs n'étant pas dans la même portée radio les uns des autres mais ayant des nœuds fils dans la portée des différents coordinateurs.

Ce problème est un peu plus délicat à régler parce que les coordinateurs n'ont aucun moyen de savoir qu'il y a d'autres émetteurs de Beacon dans leur voisinage. Deux approches ont été retenues par le groupe de travail 15.4b afin de régler les collisions indirectes, selon si cela est effectué durant l'association au réseau ou après.

- L'approche proactive: Les coordinateurs potentiels tentent d'avoir une connaissance assez complète de leur voisinage à deux sauts pour choisir un temps de transmission de Beacon vide. Pour cela tous les coordinateurs associés au réseau relayent leur temps d'envoi de leur Beacon mais aussi celui de leur père.

- L'approche réactive: Les nœuds s'associent normalement au réseau sans tenter de trouver un ordonnancement particulier pour leur Beacon. Ce dernier sera calculé en fonction des Beacons émis dans le voisinage et à la détection de collisions.

Les propositions introduites par le groupe 15.4b liées aux collisions directes et indirectes de Beacons ne représentent qu'une réflexion sur les possibles solutions et ne donnent que des directives sous la forme d'approches à suivre afin d'éviter ce problème.

Dans la littérature il y a eu beaucoup de travaux consacrés à ce problème. Parmi les solutions proposées certaines suivent les recommandations établies par le groupe 15.4b. Ces propositions se basent toutes sur le même principe d'ordonnancement temporel. On trouvera deux types d'ordonnancement: les ordonnancements temporels de supertrames et les ordonnancements temporels de Beacons.

Koubaa *et al.* proposent un algorithme d'ordonnancement de supertrame [59] afin d'éviter les collisions directes et indirectes de supertrame. Cet algorithme se base sur la technique time division approach citée précédemment, où les supertrames seront ordonnancées afin qu'il n'y ait pas de superposition des parties actives des différents coordinateurs. Ceci évitera les collisions de Beacons mais aussi les collisions des données entre les différents clusters appartenant au même voisinage.

Les auteurs proposent Superframe Duration Scheduling (SDS) et Different Superframe Duration (DSD): deux algorithmes permettant l'ordonnancement des parties actives des supertrames respectivement si les parties actives sont de la même taille ou de taille différente. Ces algorithmes sont centralisés: le PAN coordinateur doit connaître la localisation de tous les nœuds dans le réseau afin de construire le graphe d'interférence. Les auteurs proposent un algorithme glouton pour ordonnancer les supertrames. Un nouveau coordinateur se rattachant au réseau devra d'abord demander un ordonnancement afin de pouvoir émettre ces Beacons.

Dans [60] Burda *et al.* proposent un protocole d'ordonnancement de supertrame: chaque nœud voulant rejoindre le réseau doit comme pour le standard IEEE 802.15.4 envoyer une requête d'association au coordinateur dont il a entendu le Beacon, ce dernier devra retransmettre cette requête vers son coordinateur et ceci de façon à ce qu'elle parvienne au PAN coordinateur. Si le PAN coordinateur accepte l'association il enverra une Association Response qui sera elle aussi transmise tout le long du réseau jusqu'à la destination qui utilisera le numéro de séquence de ce paquet afin de calculer aléatoirement le numéro de slot qui sera utilisé pour ces futures supertrames. Afin de s'assurer que le slot choisi est bien libre une phase de vérification appelée Integrity Phase a été prévue .

Muthukumaran *et al.* proposent MeshMAC [61] qui est un algorithme totalement distribué. Les nœuds définiront leur ordonnancement sur les informations recueillies dans leur voisinage.

Dans MeshMac, la partie active de la supertrame SD et la supertrame BI seront de la même taille pour tous les nœuds dans le réseau. L'ordonnancement est effectué afin que chaque nœud puisse avoir le temps nécessaire de positionner la partie active

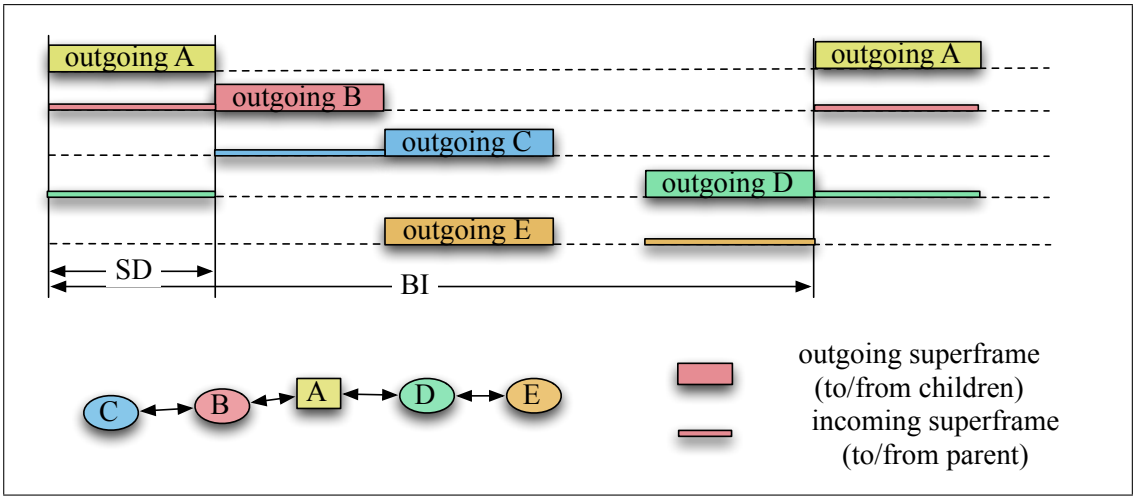


**Figure 3.6:** MeshMAC

de sa supertrame sans superposition dans un voisinage à deux sauts. Dans MeshMac, le choix de la valeur des paramètres de la supertrame seront très importants, puisque du choix de BO (Beacon Order) et SO (Superframe Order) dépend le nombre de supertrames (noter N dans l'équation 3.4) qu'il sera possible d'ordonnancer. Afin que les nœuds puissent choisir un slot pour ordonnancer la partie active de supertrame, un slot temporel est réservé au broadcast où chaque coordinateur devra d'abord recueillir toutes les informations concernant son voisinage à un saut (ordonnancement des supertrames du voisinage) puis recueillir les mêmes informations sur son voisinage à 2 sauts en envoyant une requête demandant à ses voisins de lui envoyer les ordonnancements de leurs voisins. Avec toutes ces informations le nœud pourra aisément trouver un slot libre pour ordonnancer sa supertrame.

$$N = 2^{BO-SO} - 1 \tag{3.4}$$

Jeon *et al.* [62], proposent une autre solution au problème de collision de Beacons. Cette proposition représente la deuxième approche proposée par le groupe de travail, qui préconise non plus l'ordonnancement des supertrames mais seulement des Beacons. Les auteurs proposent d'allouer une partie du début de la supertrame pour l'envoi des Beacons des différents coordinateurs. Cette partie au début de la supertrame est appelée BOP (Beacon Only Period) et est découpée en slots de taille égale réservés pour l'ordonnancement sans collision des Beacons. Les auteurs préfèrent faire un ordonnancement de Beacons au lieu des supertrames à cause du délai. L'algorithme d'ordonnancement de Beacon choisi par les auteurs est simple. Le premier BOP slot sera attribué au PAN coordinateur, le deuxième au premier nœud ayant rejoint le réseau

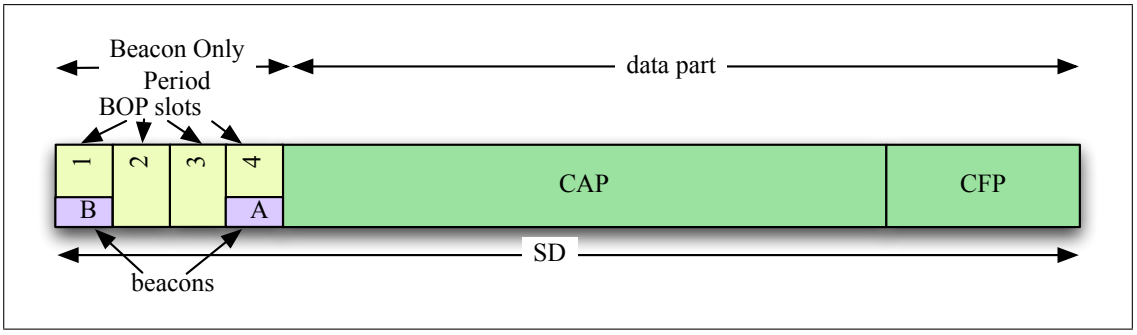


**Figure 3.7:** Beacon Only Period (BOP)

et ainsi de suite. Dans BOP (Beacon Only Period), les nœuds appartenant au même voisinage, même s'ils n'ont pas les mêmes parents partagent la même supertrame. Le fait que les supertrames des différents coordinateurs soient superposées peut en effet diminuer les délais mais à condition que soit réglé le problème des nœuds cachés. En résumé ce protocole ne sera adapté que pour des applications à faible débit et où les réseaux n'ont que des densités réduites [63]. Une autre contrainte à l'utilisation de BOP est que le choix du nombre de slots réservés pour l'ordonnancement des Beacons est très important. Avec trop de BOP slot il y aura un coût à payer sur les délais et en terme d'énergie à cause de l'overhearing induit par ce dernier. Mais s'il n'y en a pas assez, cela engendrera des collisions de Beacons. Aucun mécanisme n'a été prévu pour évaluer le nombre de slots qui doivent être réservés en fonction des conditions du réseau et aucune adaptation dynamique de leur nombre n'a été prévue.

D'autres propositions ont été faites pour améliorer le protocole BOP. Kim *et al.* proposent [64] un algorithme centralisé qui a pour but de réduire la consommation énergétique imposée par l'écoute des Beacons durant toute la durée de la BOP. Cet algorithme a pour but d'adapter dynamiquement la longueur de la BOP grâce au nœud central appelé TMC (Topology Management Center) qui a comme rôles de répertorier toutes les informations sur le réseau, construire le graphe d'interférence, définir le nombre de BOP slots indispensables pour ordonnancer les Beacons sans collision, attribuer les slots et notifier les nœuds.

Villaverde *et al.* proposent DBOP (Distrubuted BOP): une autre manière de réaliser la technique de Beacon Only Approach. Ces derniers proposent cette fois-ci un algorithme distribué pour l'attribution des BOPs slot avec une adaptation dynamique de la longueur de la partie Beacon Only Period. Un nœud choisit un slot BOP libre, mais avant de se l'attribuer il doit aviser son voisinage pendant la période active. Ses voisins devront alors répercuter cette information dans leur voisinage, et de ce fait le slot sera attribué au nœud et aucun autre coordinateur dans un voisinage à 2 sauts ne pourra le choisir. Si tous les slots sont utilisés le nœud augmente la taille de la BOP et procède

de la même façon que précédemment.

## 3.7 ZigBee

L'alliance ZigBee formée par un consortium d'entreprises a proposé un standard propriétaire qui offre des fonctionnalités allant de la couche réseau à la couche applicative. Le standard ZigBee repose sur les couches basses du standard IEEE 802.15.4.

Dans Zigbee, ce que l'on a appelé précédemment PAN Coordinateur, Coordinateur et nœuds terminaux vont s'appeler respectivement le ZigBee Coordinateur (ZC), ZigBee Router (ZR) et ZigBee End Device (ZED) qui ne peut communiquer qu'avec son coordinateur (ZR).

Zigbee ou plus précisément la couche réseau du standard Zigbee a comme tâches les opérations suivantes:

- Découverte de voisinage;
- Création de topologie;
- Adressage;
- Routage.

La couche application de ZigBee propose différents profils applicatifs selon l'utilisation voulue du réseau. Nous citerons par exemple:

- Domotique: éclairage, système d'alarme, chauffage, etc;
- ZigBee Smart Energy;
- Santé: suivi et monitoring de patients;
- Contrôle industriel: englobant la domotique et le contrôle de production;
- ZigBee Telecommunication Applications.

### 3.7.1 Adressage

À l'association d'un nœud au réseau, le coordinateur doit lui attribuer une adresse. L'attribution d'adresses dans ZigBee peut s'effectuer de deux manières différentes:

- Les adresses sont allouées d'une manière aléatoire et unique par la couche application jusqu'à l'épuisement total de ces dernières
- Les adresses sont attribuées hiérarchiquement

L'avantage de cette dernière est d'être une technique distribuée qui garantit l'unicité des adresses allouées dans le réseau. Elle est utilisée dans les réseaux de type cluster-tree où chaque ZigBee routeur se verra attribuer une plage d'adresse unique dans le réseau à allouer à ses futurs fils.

Le coordinateur ZigBee fixera, lors de la formation du réseau, le nombre maximal de fils par parent (Cm), le nombre maximal de routeurs parmi ces fils (Rm) et la profondeur maximale du réseau (depth i.e. Lm). Avec ces trois valeurs il est possible de calculer la fonction Cskip(d) eq. 3.6 représentant la taille du sous-bloc d'adresses pouvant être allouées à un parent se trouvant à la profondeur d. Si un nœud a une valeur de Cskip(d) égale à zéro il ne pourra accepter de fils et se comportera comme un nœud feuille. Le calcul de l'adresse du nouveau fils se fait par le biais de l'équation 3.5.

$$An = Aparent + Cskip(d) * Rm + n \tag{3.5}$$

$$Cskip(d) = \begin{cases} 1 + Cm * (Lm - d - 1) & \text{si} \quad Rm = 1 \\ \frac{1+Cm-Rm-Cm*Rm^{Lm-d-1}}{1-Rm} & \text{sinon} \end{cases} \tag{3.6}$$

Cette technique d'adressage a comme désavantage le fait qu'à chaque coordinateur sera attribuée une plage d'adresses à son association pour ses futures fils, si aucun noeud ne s'associe à ce coordinateur, ces adresses peuvent être considérées comme perdues.

### 3.7.2 Routage

Afin que les routeurs puissent relayer les paquets vers les bonnes destinations, Zigbee offre deux possibilités pour le routage. Si le réseau est un cluter-tree, la technique de routage utilisée sera le routage hiérarchique proposé initialement par MOTOROLA [65]. Les nœuds ayant la capacité de router des paquets auront besoin de garder uniquement les adresses de leurs fils et de leur père. Les nœuds terminaux n'ayant pas de capacité de routage relayeront directement les paquets à leur père et de ce fait n'auront qu'une adresse à garder en mémoire. Puisque les adresses sont attribuées d'une manière hiérarchique il est simple de déduire par quel chemin router le paquet.

Si la topologie est de type peer to peer, Zigbee propose d'utiliser un algorithme de routage à la demande (on demand) qui est une version allégée du protocole AODV [66] proposé initialement pour les réseaux Ad Hoc. L'établissement d'une route dans ce protocole ne se fait qu'à la demande du nœud.

Le principe est simple: un nœud voulant envoyer un paquet devra d'abord vérifier sa table de routage. Si l'adresse y figure, le nœud envoie directement le paquet. Dans le cas contraire où l'entrée cherchée n'existe pas, le nœud devra alors faire une demande de route (Route Request) vers ses voisins. Ces derniers relayeront cette requête en broadcast dans tout le réseau jusqu'à la destination qui répondra à son tour par un paquet Route Reply en unicast vers l'initiateur de la requête de route.

## Conclusions de la partie état de l'art

Dans ce chapitre nous avons essayé de définir ce qu'est un réseau de capteurs, ses particularités, ainsi que les principales propositions ayant été faites pour ce qui est des couches MAC.

Nous avons voulu être les plus exhaustifs possibles sur les différentes couches MAC proposées dans la littérature ainsi que sur la couche MAC du standard IEEE 802.15.4.

Nous avons exposé le fonctionnement de la méthode d'accès CSMA/CA utilisée dans IEEE 802.15.4, et brièvement expliqué que cette méthode d'accès présente des problèmes de performances principalement dus à la concentration du trafic sur de petits intervalles de temps imposés par le duty cycle.

Nous avons aussi développé la problématique de collisions de Beacons non traitée dans le standard et exposé les plus importantes propositions ayant cherché à résoudre ce problème. Dans le chapitre suivant nous exposerons la solution que nous proposons pour éviter les interférences inter-cluster (les collisions de Beacons), et les collisions intra-cluster ainsi qu'un mécanisme pour la construction de topologie dans IEEE 802.15.4

# Partie II

# Contributions

# CHAPITRE 4

# MCCT – MULTI-CHANNEL CLUSTER TREE

**Sommaire**


4.1 Introduction . . . . . **49**
4.2 Etat de l'art sur les protocoles multi-canaux dans IEEE 802.11 . . **51**
4.3 Problématiques et motivations . . . . . **55**
4.4 Propriétés . . . . . **56**
4.5 MCCT . . . . . **57**
4.5.1 Construction du cluster-tree . . . . . 58
4.5.2 Association et réduction des collisions inter-cluster . . . . . 61
4.5.3 Choix du coordinateur et réduction des collisions intra-cluster 62
4.5.4 Choix du canal et du slot . . . . . 64
4.5.5 Maintenance du réseau . . . . . 66
4.6 Evaluation des performances . . . . . **66**
4.6.1 Topologie aléatoire . . . . . 67
4.6.2 Topologie en grille . . . . . 69
4.6.3 Scalabilité . . . . . 71
4.6.4 Impact de la densité sur les performances . . . . . 71
4.7 Conclusion . . . . . **74**


## 4.1 Introduction

Les réseaux de capteurs sont considérés par nature comme étant des réseaux à forte densité, dépourvus d'infrastructure dans la majorité des cas et utilisant, de la même

manière que les réseaux Ad Hoc, les nœuds formant le réseau comme relais dans le but d'acheminer leurs paquets vers la destination.

Les réseaux de capteurs sont souvent assimilés aux réseaux Ad Hoc. Nous pouvons dire d'une manière plus précise que les réseaux de capteurs sont des réseaux Ad Hoc mais plus fortement contraints. Ces deux types de réseaux présentent les mêmes contraintes liées à l'absence d'infrastructure, l'obligation d'économiser l'énergie ainsi que la non fiabilité des liens radio. Cependant ces contraintes sont amplifiées dans les réseaux de capteurs à cause d'un côté de la taille des batteries et d'un autre côté des capacités limitées en terme de mémoire et de calculs.

Le standard IEEE 802.15.4 représente l'une des meilleures propositions pour les couches basses dans les réseaux de capteurs. Les propositions apportées dans ce standard tant au niveau de la couche physique que de la couche MAC ont été pensées afin de réduire la consommation énergétique grâce à notamment:

- L'utilisation de radio n'émettant qu'à de faibles puissances permet des communications de l'ordre de 10 à 20 mètres au maximum
- L'utilisation de la technique de duty cycle
- La proposition d'une nouvelle version du CSMA/CA où les nœuds éteignent leur radio durant le backoff.

L'augmentation de la densité des coordinateurs dans un réseau utilisant le standard IEEE 802.15.4 augmentera la difficulté d'ordonnancement des supertrames impliquant une réduction de la capacité du réseau en terme de débit. Si par exemple le protocole BOP est utilisé, il faudra augmenter la taille de la période Beacon Only Period afin de pouvoir ordonnancer les Becons des différents coordinateurs, ce qui réduira la taille de la partie active où les nœuds pourront communiquer.

Cette densité aura aussi pour effet l'augmentation de la contention sur la supertrame partagée par les nœuds du même voisinage (même s'ils ne partagent pas le même coordinateur) impliquant un très grand nombre de collisions. Un autre exemple peut être donné concernant l'utilisation de protocoles employant la technique d'ordonnancement de supertrame où le choix des paramètres Beacon Order et SuperFrame Order est très important puisqu'il conditionne le nombre de coordinateurs qu'il est possible d'ordonnancer dans un voisinage donné: plus la densité de coordinateurs augmentera plus il faudra favoriser de grandes valeurs de BO et réduire au maximum la valeur de SO, ce qui impliquera une augmentation des délais, et augmentera potentiellement le nombre de paquets devant être envoyés durant la supertrame.

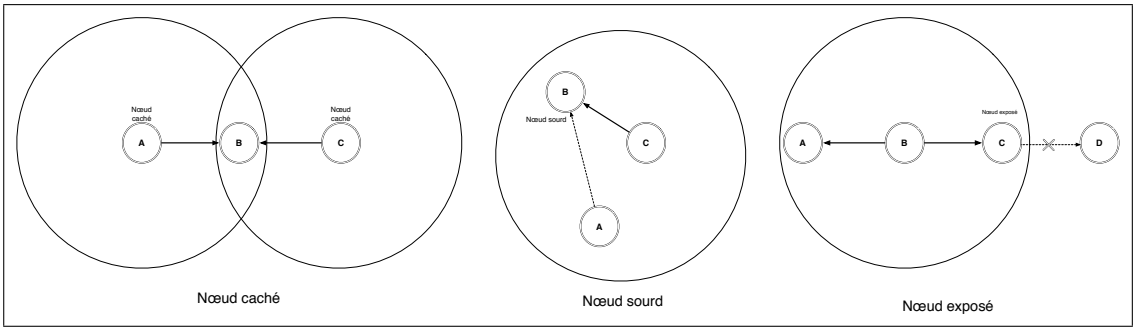


**Figure 4.1:** Nœud caché, exposé et sourd

## 4.2 Etat de l'art sur les protocoles multi-canaux dans IEEE 802.11

Le standard 802.11 a été proposé principalement pour relier des machines (ordinateurs, téléphones, tablettes...) à Internet en utilisant le mode infrastructure. Dans ce mode le point d'accès est responsable de coordonner les communications des différents nœuds. Néanmoins avec l'apparition des réseaux Ad Hoc qui sont des réseaux sans infrastructure et des réseaux maillés (Mesh), les communications ne doivent pas transiter obligatoirement par le point d'accès, mais par différents nœuds ou routeurs du réseau afin d'arriver à la destination (multi-saut). Dans ce mode de fonctionnement, c'est-à-dire dans un contexte multi-saut, la couche MAC de IEEE 802.11 devant faire face au problème :

- des nœuds cachés où plusieurs stations qui ne sont pas à portée radio les unes des autres envoient des trames vers une même station;
- des nœuds sourds où un nœud étant déjà engagé ou juste exposé à une communication n'arrive pas à recevoir les trames d'une station tierce cherchant à communiquer avec lui;
- des nœuds exposés où un nœud ne pourra pas communiquer à cause de la présence de nœuds communiquant dans son voisinage.

Dans ce genre de situation la couche MAC du standard IEEE n'est pas aussi performante que dans le mode infrastructure comme ont pu le démontrer différentes études menées sur le sujet [67, 68]. La figure 4.1 résume ces trois problèmes.

Nous nous intéressons dans cette section au standard IEEE 802.11. Ce dernier a été étudié en profondeur et un grand nombre de protocoles utilisant la technique de multi-canal ont été proposés afin de gérer les problèmes présentés ci-dessus dans les réseaux maillés et Ad Hoc. Cette technique maximise le débit dans des applications en multi-saut en permettant aux nœuds et routeurs de paralléliser les communications.

Pour faire des réseaux où sont utilisés différents canaux il faut gérer le mécanisme d'assignation de canaux et éviter d'avoir les mêmes problèmes cités précédemment (nœuds cachés, sourds, ou exposés).

On peut classifier les protocoles multi-canaux en quatre catégories [69, 70].

**Canal de contrôle dédié (Dedicated Control Channel)** : L'assignation de canaux dans ce genre de protocoles se base sur l'utilisation d'un canal de contrôle. Le but étant de séparer les messages de contrôle (signalisation) et le trafic utile (paquets de données). Avant l'envoi de paquets de données, les nœuds doivent se mettre d'accord sur le futur canal utilisé pour l'échange des paquets, et ceci sur le canal de contrôle.

Jain *et al.* [71] proposent le protocole Multichannel CSMA MAC, dans lequel un nœud voulant envoyer un paquet doit d'abord scanner l'ensemble des canaux mesurant le niveau de bruit afin de construire une liste de canaux potentiels. Une fois la liste construite le nœud envoie un paquet RTS à la destination sur le canal de contrôle. La destination recevant le CTS scanne à son tour l'ensemble des canaux, et construit de même une liste. Les deux listes sont comparées, et si aucun canal commun n'y figure, le nœud destinataire ne renvoie pas de paquet, ce qui impliquera une nouvelle tentative de la part de l'émetteur suivant les mêmes étapes. Si au contraire les deux listes contiennent des canaux en commun, le meilleur canal sera choisi par le récepteur et renvoyé dans un paquet CTS. Ce canal sera alors utilisé pour l'échange de paquets.

Ji *et al.* [72] proposent aussi qu'une liste de canaux soit constituée par les nœuds, mais en sélectionnant cette fois des canaux en fonction de leur état (libre ou occupé). Cette liste est construite par l'émetteur et sera envoyée au récepteur sur le canal de contrôle dans un paquet RTS, le récepteur choisira alors un canal dans cette liste et informera l'émetteur par l'envoi d'un CTS.

Wu *et al.* [73] proposent le protocole DCA (Dynamic Channel Assignment), qui utilise la même technique d'assignation de canal avec un canal de contrôle. Cette technique se base aussi sur l'état du canal et la négociation entre nœuds se fait avec des RTS/CTS. La différence avec le protocole cité ci-dessus est que les auteurs préconisent l'utilisation de deux interfaces radio. Une interface sera dédiée aux messages de contrôle, et une autre servira pour les échanges de paquets de données. Cette dernière doit dès lors se positionner sur le canal choisi après la négociation sur le canal de contrôle.

En résumé afin de permettre les communications en multi-canal ces trois protocoles proposent de scanner l'ensemble des canaux avant l'envoi d'une trame, puis de s'échanger des trames de contrôle afin de négocier le canal le plus intéressant à utiliser.

Dans les réseaux de capteurs cette technique doit être exclue à cause du coût énergétique imposé d'un côté par le scan des canaux à chaque envoi de trame et d'un autre côté par l'overhead imposé par la négociation du canal entre l'émetteur et le récepteur.

**Saut de fréquence (Channel Hopping)** : Les nœuds changent de canaux suivant des séquences de saut afin de s'échanger leurs données. A la différence des protocoles utilisant un canal de contrôle, dans les protocoles à saut de fréquence tous les canaux sont alloués à la communication.

Bahl *et al.* [74] proposent le protocole SSCH (Slotted Seeded Channel Hopping), dans lequel le temps est découpé en slots de 10 ms. Chaque nœud choisit et suit une séquence pseudo aléatoire de sauts de fréquence afin d'éviter les collisions dans son voisinage. Le saut d'une fréquence à l'autre se fait toujours au début du slot. Deux nœuds pourront communiquer entre eux quand ils seront en même temps sur le même canal. Afin d'éviter que les nœuds n'arrivent pas à communiquer parce que les canaux/slots ne se superposent pas il a été prévu un slot additionnel (parity slot), durant lequel les nœuds envoient des informations sur la séquence de saut afin que les nœuds voisins puissent communiquer avec eux au besoin.

So *et al.* proposent McMAC (Multichannel MAC) [75] un protocole utilisant comme le protocole SSCH la technique de saut de fréquence. Chaque nœud génère une séquence pseudo aléatoire pour les sauts de fréquence, à la différence que dans ce protocole aucun slot n'a été réservé pour la découverte du voisinage comme dans SSCH. Afin de faire connaître leur séquence de sauts les nœuds rajoutent cette information dans les paquets échangés.

La technique de saut de fréquence peut être intéressante pour les réseaux de capteurs, néanmoins il faut éviter les dérives d'horloge en synchronisant les nœuds.

**Partage de phase (split phase)** : dans ce genre de protocoles le temps est divisé en deux parties: La première où les nœuds s'échangent les paquets de contrôle afin de décider du canal qui sera utilisé pour les communications futures qui seront effectuées durant la deuxième partie.

So *et al.* [76] proposent MMAC (Multichannel MAC), un protocole synchronisé, utilisant une seule interface. Chaque nœud maintient une liste appelée PCL (Preferable Channel List) où il classe les canaux par ordre de préférence (High, Medium ou Low préférence). Comme pour le standard IEEE 802.15.4 MMAC utilise une supertrame d'une longueur d'un Beacon Interval représentant le temps s'écoulant entre l'envoi de deux Beacons successifs de l'ordre de 100 ms. Chaque Beacon Interval commence par un intervalle de temps appelé ATIM (Announcement Traffic Indication Message) qui dure 20 ms et où les nœuds se mettent sur un canal prédéfini utilisé comme canal de contrôle pour négocier un canal afin de

transmettre leurs paquets. Après la phase ATIM le canal utilisé durant cette phase sera aussi utilisé pour les communications.

Maheshwari *et al.* [77] proposent le protocole LCM (Local Coordination-based Multichannel MAC), où les nœuds partagent leur temps entre deux phases. Dans la première s'appelant Transmission Schedule, un nœud voulant envoyer un paquet doit d'abord envoyer un RTS sur le canal de contrôle. Ce paquet contient la liste des canaux libres et propose un temps pour l'envoi des paquets. Le récepteur répond alors par un CTS contenant le canal choisi. A la réception du CTS le nœud initiateur de la communication doit envoyer un autre paquet (RES) pour réserver le canal choisi et confirmer le schedule pour l'envoi des paquets. De ce fait tous les voisins dans un voisinage à un saut seront au courant du canal et schedule choisis, leur interdisant son utilisation durant le temps de la communication. Après cette négociation les nœuds changent ce canal afin de s'échanger leur données.

Zhang *et al.* [78] proposent le protocole TMMAC (TDMA Multi-channel MAC), un protocole synchronisé, dans lequel, comme dans MMAC chaque période de temps commence par un Beacon, suivi de la période appelée ATIM et de la période de communication. Mais à la différence de MMAC le temps durant cette phase est lui aussi découpé en slots, chacun de taille suffisante pour l'envoi d'un seul paquet. Durant la période ATIM les nœuds s'échangent des paquets spécifiques définis dans ce protocole (ATIM control packets) afin de négocier le canal et le slot temporel qui seront utilisés durant la phase de communication.

Dans des réseaux contraints énergétiquement et avec l'utilisation de la technique du duty cycle, les nœuds n'auront qu'un court intervalle de temps afin d'envoyer et de recevoir les trames. Ceci implique que la technique de négociation de canal proposée dans les trois protocoles cités ci-dessus doit être évitée dans les réseaux de capteurs à cause du temps limité pour les échanges de données. D'un autre côté afin de réduire la consommation énergétique il faut limiter au maximum l'envoi de paquets de contrôle quand cela est possible.

**Canal dédié à la réception** : dans ce genre de protocole, chaque nœud se verra attribué un canal dédié à la réception des données, et un émetteur devra se positionner sur le canal de réception du destinataire pour le transfert de ses paquets.

Shacham *et al.* [79] proposent une architecture pour réseaux maillés en multi-canal RDT (Receiver Directed Transmission), dans laquelle chaque nœud se voit attribuer un canal utilisé pour la réception des paquets de données et où les nœuds seront toujours en écoute. Un nœud voulant émettre un paquet devra changer de canal et se positionner sur le canal du récepteur. L'allocation de canal et la connaissance du canal de la destination ne font pas partie de la proposition car les auteurs considèrent que cela doit être fait dans une couche plus haute.

Extended RDT (xRDT) proposé par Maheshwari *et al.* [77], représente une extension de l'approche RDT. Cette extension a pour but de pallier les problèmes de nœuds cachés et sourds pouvant survenir dans des réseaux utilisant différents canaux pour communiquer. Pour cela les auteurs préconisent de rajouter une interface appelée TI (Tone Interface). Un nœud recevant sur un canal enverra un signal pour annoncer qu'il est occupé en réception sur son interface TI (l'interface de signalisation).

L'utilisation d'un canal de réception pour chaque nœud peut être intéressante à condition de rajouter un mécanisme d'attribution de canaux et de découverte du voisinage ainsi qu'un mécanisme pour éviter les nœuds cachés et sourd pouvant survenir dans les réseaux en multi-canal et ceci sans l'utilisation de plusieurs interfaces.

L'utilisation de ces protocoles est peu appropriée pour les réseaux de capteurs et cela pour des raisons simples: dans les réseaux de capteurs la contrainte énergétique fait qu'il faut diminuer au maximum les échanges de paquets de contrôles, et de ce fait les mécanismes tels que RTS/CTS ou autres mécanismes de réservation de canaux présentés précédemment reposant sur l'envoi multiple de paquets doivent être abolis dans les réseaux de capteurs. De même que l'utilisation de plusieurs interfaces gérant plusieurs canaux simultanément n'est pas conseillée dans les réseaux de capteurs, d'une part pour le coût en terme d'énergie qui sera proportionnel au nombre d'interfaces, et d'autre part pour des raisons économiques puisque que cela coûtera plus cher d'avoir plusieurs antennes vu que la majorité des capteurs disponibles sur le marché ne sont équipés que d'une seule interface.

## 4.3 Problématiques et motivations

Dans la section précédente nous avons exposé les différentes solutions proposées afin de construire des réseaux fonctionnant sur différents canaux dans IEEE 802.11. Les propositions citées sont impossibles à mettre en œuvre dans les réseaux de capteurs car ils n'ont pas été pensés dans une optique d'économie d'énergie:

- Utilisation de multiples interfaces radio
- Multiples échanges de paquets de contrôle (overhead)

Comme expliqué précédemment le standard IEEE 802.15.4 ne propose aucune solution aux problèmes suivants:

- Les collisions de Beacons (interférences inter-cluster)
- La construction de la topologie

- Les interférences intra-cluster
- La réduction du temps d'association des nœuds au réseau

Un certain nombre de solutions utilisant l'ordonnancement temporel de Beacons ou de supertrames dans le but de pallier le problème de collisions de Beacons dans le standard IEEE 802.15.4 ont été proposées dans la littérature. Néanmoins ces solutions se concentrent uniquement sur les techniques d'ordonnancement et occultent les problèmes pouvant survenir avec ce genre de solutions, tels que la réduction de la capacité du réseau, les interférences inter-cluster, sachant que ces protocoles ne s'intéressent pas à la construction de la topologie.

La majeure partie des protocoles [36, 38, 40, 42, 44, 45] proposés utilisant la technique de multi-canal dans les réseaux de capteurs ne traite pas les problèmes du standard IEEE 802.15.4 cités dans la section 3.6 de l'état de l'art. Le protocole MCMAC [43] propose de gérer les réseaux en multi-canal dans IEEE 802.15.4 mais comme nous l'avons déjà expliqué ce protocole génère un overhead trop important dû aux demandes d'attribution des slots de communication. TMCP [39] proposé afin de construire un ensemble de cluster-tree fonctionnant sur plusieurs canaux peut aussi être utilisé dans le standard IEEE 802.15.4 mais cette solution laisse ouverte le problème des collisions directes et indirectes de Beacons.

## 4.4 Propriétés

Notre proposition a pour but de répondre aux problèmes suivants:

- L'allocation de canaux
- La construction de la topologie i.e trouver le meilleur coordinateur pour l'association du nœud au réseau
- La réduction du temps d'association des nœuds au réseau par l'accélération de la découverte de voisinage
- L'élimination des collisions de Beacons (inter-cluster)
- La réduction des collisions intra-cluster
- L'augmentation de la capacité du réseau
- Accélération de la reconstruction du réseau: dans le cas où une partie du réseau est isolée à cause de la perte de la synchronisation d'un coordinateur, le sous arbre pourra être rattaché au réseau en un BI (Beacon Interval) maximum
- Réduction de la consommation énergétique: accélérer le temps d'association revient à diminuer le temps que les nœuds passent à scruter le canal

A l'encontre des différentes propositions [36, 38, 40, 42, 44, 45], notre proposition a été pensée pour le standard IEEE 802.15.4 et tend à respecter son fonctionnement. Dans MCCT (Multi-Channel Cluster Tree) le seul changement en comparaison au mode Beacon Enabled est l'ajout d'un paquet servant uniquement à la découverte de voisinage.

## 4.5 MCCT

Nous présentons MCCT (Multi-Channel Cluster Tree), qui est un protocole pour la construction de topologie utilisant le mode Beacon-Enabled du standard IEEE 802.15.4 dans un contexte multi-canal. Le but de ce protocole est d'éviter les collisions de Beacons qui représentent un des problèmes majeurs du standard IEEE 802.15.4. Le protocole MCCT utilise l'ensemble des canaux fournis par le standard IEEE 802.15.4 dans le but de construire un réseau fonctionnant sur différents canaux.

L'utilisation de différents canaux dans MCCT aura pour but d'éviter les collisions inter-cluster en ordonnançant les supertrames sur différents canaux. Et afin de réduire les collisions intra-cluster nous avons rajouté un mécanisme de limitation de nombre de fils par coordinateur, ceci étant la solution la plus triviale permettant au mécanisme CSMA/CA de produire de bonnes performances. Le chapitre suivant sera exclusivement consacré à l'étude du mécanisme CSMA/CA à travers les différents paramètres.

Le standard offre les outils nécessaires pour construire la topologie soit:

- Mécanisme de découverte de voisinage (Scan)
- Les requêtes d'association (association request et association response)

Mais ne donne aucune directive à cet effet.

Notre protocole MCCT aura pour tâche de gérer l'association des nœuds au réseau, de définir sur quelles bases un nœud choisira un coordinateur comme père, ainsi que d'ordonnancer les supertrames dans le but d'en réduire au maximum les collisions. Les informations à partir desquelles un nœud choisira un parent lui seront fournies par le biais d'un nouveau paquet que nous avons défini. En s'associant au réseau comme dans le standard IEEE 802.15.4, si le nœud est un RFD, il se comportera comme une feuille n'échangeant des paquets qu'avec son parent. Au contraire si le nœud nouvellement associé est un FFD, ce dernier devra choisir un canal, et un ordonnancement pour ses futures supertrames. Tout coordinateur devra éviter la superposition de sa supertrame et celle de son père, et utiliser un canal différent des canaux utilisés dans son voisinage, afin d'éviter les collisions de Beacons. En résumé à l'encontre de ce qui est fait dans le standard IEEE 802.15.4 nous proposons de construire un réseau en cluster-tree où chaque cluster utilise une fréquence différente de celle qu'utilisent ses voisins.

La phase d'association au réseau dans un réseau IEEE 802.15.4 peut s'avérer très longue et gourmande en terme d'énergie. Le temps que le nœud cherchant à s'associer passe à scanner les différents canaux est conditionné par la longueur des supertrames

et le nombre de canaux. Plus la supertrame est longue plus le temps d'association des nœuds au réseau croît et plus la consommation énergétique augmente. Dans le standard IEEE 802.15.4 par exemple il faudra qu'un nœud scrute les 16 canaux pour au moins une longueur max de supertrame. Ceci implique un temps de convergence très long (70 minutes de scan par nœud pour un BO=14).

Afin de réduire au maximum le temps que prend l'association au réseau, nous proposons dans MCCT d'utiliser un canal de contrôle qui sera dédié uniquement à l'envoi du paquet Hello servant essentiellement à la construction du cluster-tree. En résumé, au lieu de scanner tous les canaux à la recherche de Beacons pour s'associer comme dans IEEE 802.15.4, dans MCCT un nœud aura à scanner uniquement le canal de contrôle réduisant ainsi le coût en terme d'énergie et de délai d'association au réseau.

En effet en utilisant un canal de contrôle, un nœud n'aura à scanner le canal que pendant un cycle au maximum (256 secondes dans le cas où BO=14) réduisant le temps que les nœuds passent à scanner le canal et réduisant ainsi le temps de formation du réseau.

En résumé nous présentons un nouveau mécanisme de découverte de voisinage ayant les objectifs suivants:

- L'accélération de la découverte de voisinage (construction et reconstruction du réseau)
- Réduction de l'énergie consommée durant la phase de construction
- Communication des canaux utilisés dans le voisinage

Nous conservons la même structure de supertrame fig. 3.4 proposée dans IEEE 802.15.4, mais en permettant que l'incoming et l'outgoing superframe soient sur différents canaux. La figure 4.2 représente l'architecture du réseau construit par MCCT: le PAN coordinateur représenté par le nœud A dans la figure communique avec ses fils (B,C,D,E) sur le canal 1; les filss du PAN coordinateur qui sont eux-mêmes coordinateurs utiliseront des canaux différents afin de communiquer avec leurs filss; les nœuds feuilles participeront seulement à la supertrame de leur père et n'auront aucun Beacon à transmettre aux nœuds K,L,M par exemple.

Nous allons maintenant décrire plus en détails les différents mécanismes que nous proposons dans notre protocole.

### 4.5.1 Construction du cluster-tree

Dans le standard IEEE 802.15.4 afin de rejoindre le réseau les nœuds doivent scanner dans l'ordre croissant l'ensemble des canaux disponibles, ceci dans le but de collecter les Beacons envoyés par les coordinateurs dans leur voisinage et ainsi recueillir le maximum d'informations.

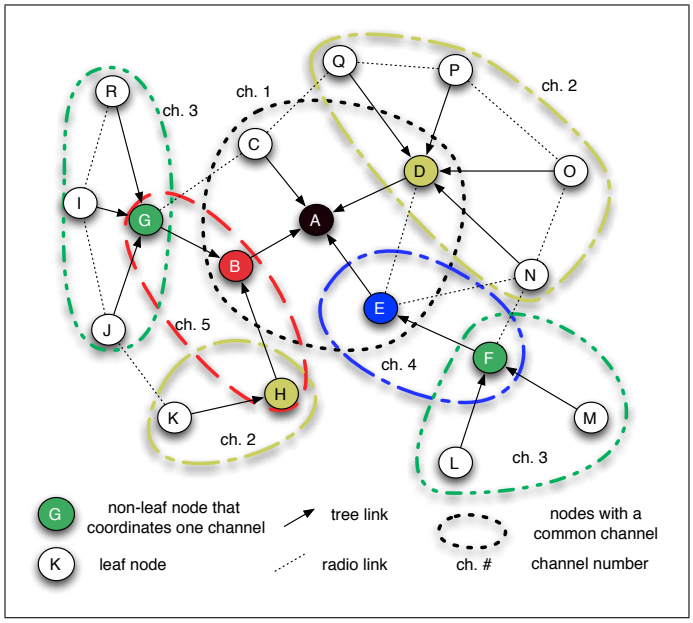


**Figure 4.2:** Architecture du réseau multi-canal

Un nœud est assuré de répertorier tous les Beacons envoyés dans leur voisinage que s'il passe un temps suffisant sur chaque canal. Le problème est que scanner tous les canaux a un double inconvénient: non seulement cela engendre une forte consommation en terme d'énergie, mais en plus, cela prend un temps non négligeable, impliquant une longue durée en écoute (overhearing). Ce temps passé à scanner chacun des canaux se répercutera ainsi sur le temps nécessaire pour la construction du réseau et sur sa reconstruction en cas de perte de la synchronisation.

Comme expliqué dans la section 3.5.5, les nœuds ont le choix de faire un scan actif en demandant explicitement des Beacons (Beacon request) ou un scan passif.

Le scan actif n'est prévu que pour des réseaux fonctionnant en mode Non-Becon-Enabled. Si cette technique est néanmoins utilisée dans le mode Beacon-enabled les nœuds n'ont aucune garantie que les coordinateurs aient la radio allumée lors de l'envoi de la requête. Ce scan introduit aussi un overhead inutile dû à l'envoi des requêtes que les coordinateurs ignorent (envoi du Beacon est périodique).

La deuxième méthode à la disposition des nœuds est le scan passif où les nœuds scrutent l'ensemble des canaux à la recherche des Beacons dans leur voisinage. Les nœuds n'ayant aucune information sur les longueurs des supertrames des différents coordinateurs, doivent afin d'être sûr de ne pas manquer un Beacon, scanner chaque canal durant une période d'au moins la longueur maximum d'une supertrame:

$$(aBaseSuperframeDuratioon * (2^N + 1))symboles \tag{4.1}$$

Où $N$ appartient à l'intervalle [0,14]. Comme dit précédemment afin de ne pas manquer les Beacons et en l'absence d'information sur la supertrame le nœud utilisera la valeur maximale du paramètre $N$ i.e 14, ce qui impliquera un scan de 251,67 secondes par canal.

Pour éviter l'overhead induit par le scan actif (Beacon request) et réduire le temps passé à scruter tous les canaux à la recherche de Beacons dans le scan passif, nous proposons d'utiliser dans MCCT un canal qui sera dédié à la seule découverte de voisinage. Sur ce canal sera envoyé uniquement le paquet Hello sur lequel se base notre méthode de découverte du voisinage.

Tout nœud cherchant à s'associer devra d'abord récupérer les Hellos dans son voisinage. Pour cela il devra scanner uniquement le canal de contrôle où sont envoyés les Hellos, au lieu de scanner tous les canaux à la recherche des Beacons comme cela est fait dans le standard IEEE 802.15.4.

Comme déjà énoncé dans MCCT la trame Hello sera recherchée par les nœuds afin de savoir s'il y a des coordinateurs dans leur voisinage réduisant le temps de convergence grâce à l'envoi de cette trame par tout coordinateur associé au réseau à chaque BI sur le canal de contrôle.

Ce paquet n'est pas envoyé périodiquement mais de manière aléatoire durant la période inactive du coordinateur ($schedule_rand()$) dans la figure 4.3). Si deux Hello collisionnent ils auront une très faible probabilité d'être envoyés au même moment durant de futurs envois. Ceci peut être expliqué par la taille des paquets Hellos comparés au BI. Nous avons choisi un envoi aléatoire afin d'éviter de rajouter une négociation entre les coordinateurs pour l'envoi de ce paquet.

L'utilisation d'un canal de contrôle fait que le nœud n'aura pas à scruter l'ensemble des canaux à la recherche de Beacons avant de s'associer au réseau, un nœud a seulement à scanner le canal de contrôle durant une période maximum de supertrame réduisant le temps de scan à 251,67 secondes (au lieu de 16*251,67 secondes dans IEEE 802.15.4), accélérant ainsi le processus d'association et de construction du réseau.

Dans MCCT, nous dissocions le rôle du Beacon et du Hello. La procédure de synchronisation est la même que celle utilisée dans le standard IEEE 802.15.4 employant le Beacon qui servira aussi au maintien de la supertrame i.e. coordonner les nœuds appartenant au cluster (Synchronisation, GTS, Data pendig list , etc.), alors que le Hello aura uniquement comme rôle la découverte du voisinage. Dans ce dernier sera rajouté le temps du prochain Beacon du coordinateur choisi.

La figue 4.3 illustre la découverte de voisinage que l'on propose dans MCCT. Le PAN coordinateur représenté par le nœud A commence à envoyer le paquet Hello sur le canal zéro, le nœud B qui est en train de scanner ce même canal reçoit le Hello de A. Ainsi la fin de sa période de scan qui sera d'un BI au maximum, le nœud B s'associera au nœud A et devient ainsi un de ses fils. La procédure d'association est la même que celle utilisée dans le standard IEEE 802.15.4 (Association Request/ Association Response), sauf que le nœud après sa période de scan devra attendre le moment où débute la supertrame du

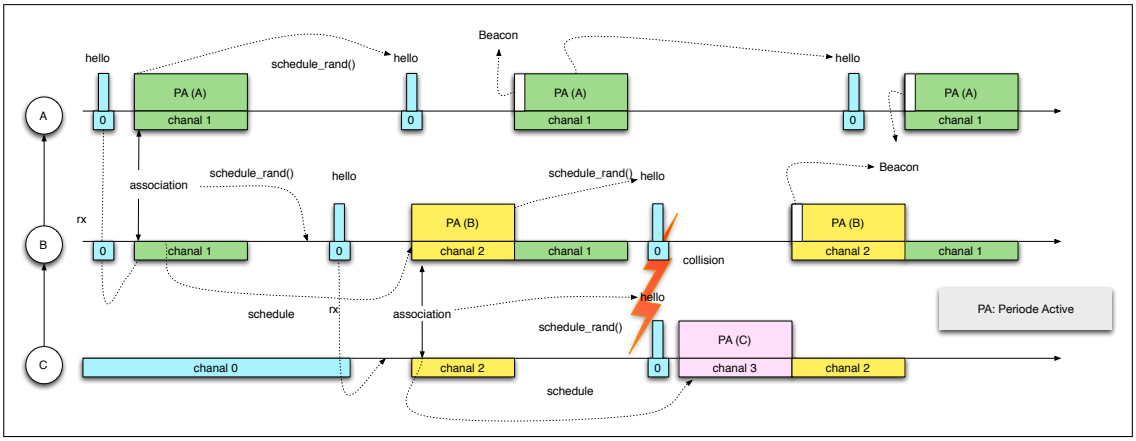


**Figure 4.3:** Découverte de voisinage

coordinateur qu'il a choisi comme père pour aller le rejoindre sur son canal afin d'entamer le processus d'association. Une fois associé, le nœud B envoie à son tour des Hellos pour déclarer sa présence et inviter d'autres nœuds à rejoindre le réseau. Le nœud C étant à la recherche de Hello s'associera à B. La formation du réseau se fera de cette manière. Les critères de choix du coordinateur comme futur parent seront expliqués ultérieurement

### 4.5.2 Association et réduction des collisions inter-cluster

A part le mécanisme d'association au réseau le standard IEEE 802.15.4 ne définit pas de mécanisme de création de topologie i.e sur quelle base un nœud choisit un parent à la place d'un autre. Dans cette section nous présentons la méthode de construction de cluster-tree utilisée dans MCCT.

Dans MCCT la construction du cluster-tree est toujours initialisée par le PAN coordinateur qui enverra des paquets Hello sur le canal de contrôle afin de manifester sa présence. Le contenu d'un paquet Hello est présenté dans la table 4.1. Un nœud voulant s'associer au réseau, scanne uniquement ce canal pour une durée d'un BI au maximum. Pendant cette période, les nœuds enregistrent les paquets Hello reçus afin de construire une table de voisinage. A la fin de la période de scan ils devront prendre une décision sur le parent à choisir. La décision ayant été prise, le nœud n'aura plus qu'à aller sur le canal de son futur père et d'exécuter l'association (cf. section 3.5.5). A son tour, si le nœud est un FFD il choisira un canal et un slot pour communiquer avec ses futurs fils, et commencera à envoyer ses Hello sur le canal de contrôle.

Dans le standard IEEE 802.15.4, dès qu'un FFD rejoint le réseau, ce dernier devient coordinateur et a comme rôle d'envoyer des Beacons afin de gérer son cluster et d'annoncer sa présence. Dans des réseaux où il y a un grand nombre de coordinateurs cohabitant dans le même voisinage il peut être dangereux de permettre que tous les

coordinateurs envoient des Beacons, et ceci pour deux raisons: la première est une raison énergétique i.e un coordinateur émettant un Beacon sera dans l'obligation de rester éveillé pendant toute sa partie active (SD) dans l'attente d'éventuels paquets de la part de ses fils, ou d'association de la part de nouveaux nœuds voulant rejoindre son cluster; la deuxième raison est qu'un coordinateur émettant des Beacons alors qu'il n'a pas de fils peut collisionner avec les Beacons de ses voisins, ou juste occuper un slot (Temporel ou fréquentiel) et en interdire son utilisation à d'autres coordinateurs.

Afin de réduire la pression sur le canal et le nombre de Beacons envoyés dans le réseau nous essayons dans MCCT de maximiser le nombre de nœuds feuilles. Pour cela nous définissons deux types de coordinateurs:

- Les coordinateurs actifs: ce sont des coordinateurs associés au réseau et ayant des fils. Ils enverront en début de chaque supertrame, comme dans le standard IEEE 802.15.4, un Beacon, et continueront d'annoncer leur présence sur le canal de contrôle sous forme de Hello.

- Les coordinateurs passifs: ce sont les coordinateurs associés au réseau mais n'ayant pas encore d'fils. Ils ne seront autorisés à envoyer que les Hello sur le canal de contrôle, mais n'enverront pas de Beacons. Ils ne resteront réveillés que pendant une portion du temps de la partie active pour recevoir une éventuelle requête d'association de la part de fils potentiels. Si aucune requête n'est reçue ils s'endormiront en attendant l'envoi du Hello et le Beacon de la prochaine supertrame.

Un FFD se rattachant au réseau dans notre proposition sera considéré comme un coordinateur passif jusqu'au moment d'avoir des fils, et ceci afin de réduire l'envoi de Beacons, d'économiser les canaux utilisés dans le réseau et l'énergie consommée pour le maintien de supertrame. Les RFD pourront seulement participer au cluster en tant que nœud feuille.

### 4.5.3 Choix du coordinateur et réduction des collisions intra-cluster

Le but dans MCCT est de maximiser le nombre de nœuds feuilles. Ceci est réalisé en s'associant de préférence à un coordinateur ayant déjà des filss (coordinateur actif).

En outre, en choisissant les parents avec le plus petit nombre de fils les nœuds auront tendance à équilibrer la charge des coordinateurs en terme de nombre de fils. MCCT aura pour rôle, comme dit précédemment, d'éviter les collisions inter-cluster en parallélisant les communications sur différents canaux. Ceci aura un impact positif sur les performances mais n'est pas suffisant.

Notre protocole aura aussi pour rôle de réduire les collisions intra-cluster. Il faut souligner que le standard IEEE 802.15.4, même s'il a été pensé pour des réseaux qui sont par définition des réseaux à forte densité, n'offre pas de bonnes performances dans

| | |
|---|---|
| **coordinator ID** | Identifiant du coordinateur |
| **Nombre de fils** | |
| **Slot utilisé** | Le temps avant le début de la supertrame |
| **Canal du cluster** | Le canal utilisé pour le fonctionnement du cluster |
| **Numéro de séquence** | |
| **La profndeur dans l'arbre** | |
| $< \boldsymbol{canal}, \boldsymbol{Slot\ utilisé} >$ | Les paires canal/slot utilisé dans le voisinage |

**Table 4.1:** Contenu de la trame `hello`

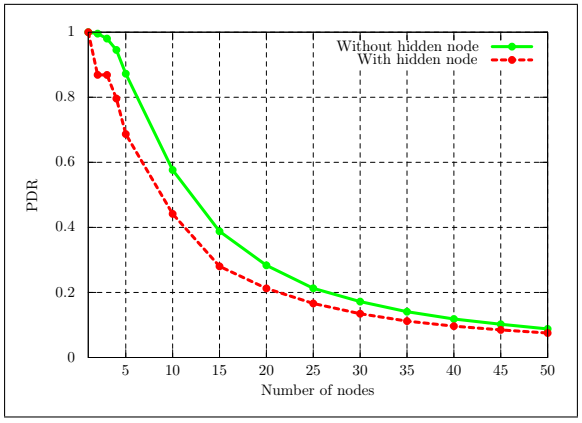


**Figure 4.4:** Packet Delivery Ratio dans un réseau en étoile IEEE 802.15.4

ce genre de contexte (densité en terme de nœuds et de trafic). La cause vient du fait que le mécanisme d'accès au canal (CSMA/CA) proposé dans le standard IEEE 802.15.4 n'est efficace qu'en présence d'un trafic et d'une densité faible.

Nous avons étudié l'impact du nombre de fils sur les performances du CSMA/CA slotted. La figure 4.4 représente le PDR (Packet Delivery Ratio) pouvant être offert par la couche MAC IEEE 802.15.4 en mode Beacon-Enabled. Nous avons simulé un réseau en étoile où nous avons augmenté progressivement le nombre de filss du coordinateur, chaque nœud ayant un paquet à émettre par supertrame.

On peut déduire aisément à partir de la courbe 4.4 que cette méthode d'accès ne peut offrir de bonnes performances que si le coordinateur a un nombre limité de filss, et cela quelque soient les valeurs choisies de BO et SO. Cette courbe montre que si on dépasse

un nombre de 5 fils par coordinateur et cela dans des conditions optimales (sans nœud caché), le coordinateur en question ne pourra recevoir plus de 85% des paquets émis. Avec dix nœuds cette valeur chutera aux alentours de 60%. Cette courbe ne présente que les performances à un saut, mais sur plusieurs sauts les performances seront beaucoup plus faibles comme montré dans l'étude menée par Liu *et al.* [80].

Nous avons estimé qu'à partir de cinq nœuds dans un cluster un nœud devrait s'associer à un coordinateur passif au lieu de charger le cluster ayant atteint le nombre de 5 fils. Si aucun coordinateur ayant moins de fils que le seuil autorisé (5 fils) ne se trouve dans le voisinage, le nœud, pour ne pas rester isolé, pourra s'associer avec ce coordinateur sans tenir compte du seuil, mais cette décision ne devra être prise qu'en dernier recours. Les conditions que nous avons posées ont pour but de maximiser le nombre de nœuds feuilles dans le réseau (nœuds terminaux et coordinateurs passifs) mais pas au détriment de la connectivité du réseau. Dépassant le seuil maximum du nombre d'e fils par cluster, le nœud devra, si cela est possible, choisir un coordinateur passif, dans le but aussi de répartir la charge sur différents coordinateurs, et si ce n'est pas possible il devra se rattacher à un coordinateur avec un nombre de fils dépassant le seuil.

En résumé, dans MCCT un nœud voulant s'associer au réseau, scanne le canal de contrôle à la recherche des HELLO, construit sa table de voisinage, et prend la décision de s'associer au réseau. Le nœud, en fonction des informations contenues dans les Hellos, classe les coordinateurs suivant le nombre de fils de ces derniers, et devra choisir comme parent le coordinateur avec le moins de fils, à condition que ce dernier ne soit pas un coordinateur passif. Ceci nous permettra de minimiser le nombre de coordinateurs actifs et en même temps d'économiser canaux et énergie. Si des coordinateurs ont le même nombre de fils, le nœud choisira le coordinateur le plus proche du puits en terme de sauts. Si le nombre d'e fils et la profondeur dans l'arbre sont les mêmes pour tous les coordinateurs sélectionnés le choix se fera aléatoirement.

De cette section nous avons déduit que le MAC de IEEE 802.15.4 ne peut fonctionner correctement que si nous contraignons la construction de la topologie i.e si nous limitons le nombre de fils par coordinateur. Cette solution peut s'avérer difficile voire impossible dans certaines conditions i.e dans le cas où le réseau présente une grande densité de RFD par rapport aux FFD. Les chapitres suivants sont consacrés à l'étude du CSMA/CA du standard IEEE 802.15.4 et à la proposition d'une méthode pour l'amélioration de ses performances, les propositions ayant comme objectif de s'abstraire de la connaissance préalable du nombre de fils et de la quantité de trafic dans le réseau.

### 4.5.4 Choix du canal et du slot

Afin d'éviter les problèmes de surdité et de collisions il faut que:

- Les supertrames des coordinateurs et de leurs fils ne se superposent pas en temps

- Les coordinateurs étant dans un même voisinage soient sur des canaux ou slots temporels différents
- Les nœuds soient sur le bon canal afin de communiquer avec leurs parents

Une fois associés, tous les nœuds que ce soit des FFD ou des RFD doivent se synchroniser avec leur coordinateur. La synchronisation est effectuée par le biais des Beacons envoyés par les différents coordinateurs actifs du réseau.

Chaque nœud ayant été associé au réseau doit alors se réveiller au début du slot choisi par son parent comme outgoing superframe et sur le bon canal afin de recevoir le Beacon.

Les RFDs n'auront plus qu'à se réveiller périodiquement pour la réception du Beacon. Les FFDs lors de l'association au réseau devront en plus choisir un slot et un canal en vue de maintenir leurs futurs supertrames (outgoing superframe) afin de communiquer avec leurs propres fils.

La supertrame i.e. le temps entre deux Beacons successifs, comme dans MeshMAC, est découpée en intervalles de temps de la même taille ayant la longueur de la partie active de la supertrame (SD).

La partie active de chaque coordinateur devra ainsi être ordonnancée dans un des slots temporels. Chaque coordinateur devra décaler son outgoing supertrame par rapport à celle de son parent d'au moins un SD afin d'éviter toute superposition des incoming et outgoing superframe du coordinateur et de ses fils (StatTime=SD) .

Le nombre de slots temporels disponibles pour l'ordonnancement des différentes supertrames dans un voisinage donné sera alors $N = 2^{BO-SO}$ slots temporel. Il faut rajouter à cela les 15 canaux utilisés dans MCCT augmentant ainsi le nombre de slots à $N = 2^{BO-SO} * 15$ .

Afin de réduire le temps de remontée des données il est préférable que le futur coordinateur choisisse le slot pour sa propre supertrame (outgoing superframe) juste avant celle de son coordinateur (père) comme le montre la figure 4.3.

Les informations de canal et slot étant envoyées dans les Hellos les nœuds non associés choisir le temps d'envoi de leur propre supertrame sans interférer avec leurs parents et leur voisinage.

Le choix du slot et du canal dépendra des informations contenues dans les Hellos comprenant le canal et le slot choisis par les coordinateurs dans son voisinage à 2 sauts. Le nouveau coordinateur peut dès lors choisir son canal aisément en appliquant un algorithme glouton qui aura comme principe:

- Classement des canaux par ordre croissant d'utilisation
- Choix du canal le moins utilisé

Néanmoins, dû à une collision de Hello ou à une association simultanée de deux nœuds au réseau, il est possible que certains nœuds n'aient pas une vision complète de

| | |
|---|---|
| **Nombre de nœuds** | 60 (random) |
| **Nombre moyen de voisins** | 9 |
| **Type de trafic, rate** | CBR, 0.5 pkt/min |
| **Paramètres de 802.15.4** | SO = 1, BO $\in$ [2..14] |

**Table 4.2:** Paramètres de simulations

leur voisinage ou qu'ils choisissent le même canal et slot. Cette situation peut être évitée facilement si le coordinateur qui partage le même canal/slot dans son voisinage est un coordinateur actif. Le coordinateur passif en se réveillant sur son canal/slot au début de sa supertrame, afin d'écouter une éventuelle sollicitation d'un potentiel fils, entendra le Beacon de son voisin et pourra changer de slot en conséquence. Si les deux coordinateurs sont passifs ceci n'aura aucun impact et ce problème se résoudra de lui même quand ils se transformeront en coordinateurs actifs.

### 4.5.5 Maintenance du réseau

Durant son fonctionnement, le réseau a besoin d'un mécanisme permettant de détecter des coupures de liens dans le réseau. Dans MCCT on utilise le numéro de séquence du Beacon: Le PAN coordinateur incrémente et inclut ce numéro de séquence dans son Beacon. Chaque coordinateur récupère ce numéro de séquence et l'inclut dans son Beacon.

Les nœuds recevant les Beacons de leurs parents détectent une perte de connectivité dans l'arbre si ce numéro de séquence reste inchangé durant un certain nombre de BI (Fixé à trois dans nos simulations).

Dans ce cas ces coordinateurs arrêtent d'envoyer leur Beacon et recommencent toute la procédure de découverte de voisinage pour se reconnecter au réseau.

Le fait que la détection de la perte de connectivité prend plusieurs BI, le coordinateur pourra se rattacher au réseau avant que ses fils ne la détectent à leur tour, ce qui accélère la reconstruction puisqu'il n'est pas nécessaire que tous les nœuds du sous arbre recommencent la découverte de voisinage (destruction de tout le sous arbre).

## 4.6 Evaluation des performances

Nous avons comparé MCCT avec le standard IEEE 802.15.4 et le protocole MeshMAC [81], nous avons évalué ces trois approches en terme de délais, de PDR (packet delivery ratio) et de l'indice de Jain (Jain index) [82] qui représente l'équité en terme d'accès au canal pour les différents nœuds du réseau. Pour cela nous avons utilisé le simulateur WSNET [83]: un simulateur qui a été principalement développé pour les réseaux de capteurs [84]. Nous avons implémenté le mode Beacon-Enabled du standard IEEE 802.15.4. Notre proposition MCCT et une version idéale de MeshMAC où

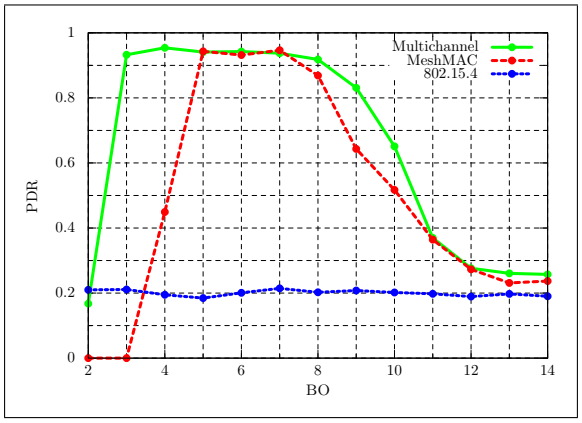


**Figure 4.5:** Packet Delivery Ratio dans une topologie aléatoire de 60 nœuds

tous les nœuds connaissent leur voisinage à deux sauts, ce qui annule toutes les erreurs d'ordonnancement quand cela est possible (nombre suffisant de slots pour ordonnancer l'ensemble des supertrame).

### 4.6.1 Topologie aléatoire

Pour l'évaluation de performances nous avons simulé dans un premier temps une topologie aléatoire de 60 nœuds déployés aléatoirement dans un cercle avec un degré moyen de voisins égal à 9. Pour ces simulations nous avons choisi un trafic de type convergecast où les nœuds envoient toutes leurs données vers le PAN coordinateur. Le tableau 4.2 présente les paramètres de simulation.

aazer La figure 4.5 montre les résultats en terme de PDR: on remarque que le standard IEEE 802.15.4 a des performances très médiocres et cela quelle que soient les valeurs de BO choisies, 80% des paquets envoyés sur le réseau n'arrivent pas à destination. Ces performances sont dues à l'absence d'ordonnancement des différentes supertrames, entrainant des collisions entre les Beacons des différents coordinateurs. Seuls les fils du PAN coordinateur parviendront à transmettre leurs paquets.

On constate également que les trois solutions présentent de mauvaises performances en terme de PDR pour les grandes valeurs de BO. La cause est que le nombre de paquets générés entre deux périodes d'activité est trop grand par rapport à la taille de la période active, dépassant la capacité offerte par le réseau. On peut tirer la conclusion suivante: dans des réseaux avec un trafic périodique et en utilisant un mécanisme de duty cycle comme le standard IEEE 802.15.4, il est préférable de favoriser des cycles d'apparition de Beacon fréquents avec des périodes d'activité courtes (petites valeurs de BO et SO). Comme montré précédemment, la couche MAC du standard IEEE 802.15.4 ne gère pas très bien la contention: la présence d'un grand nombre de paquets à transmettre en

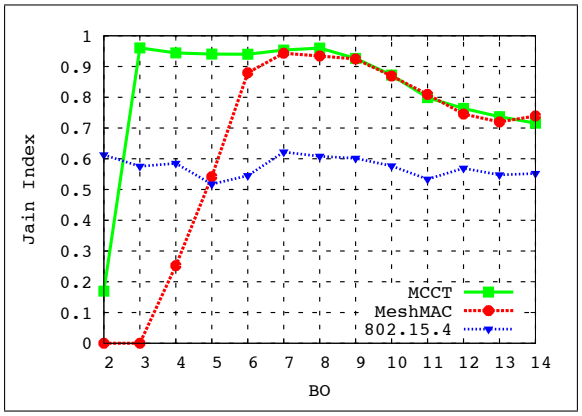


**Figure 4.6:** Jain Index dans une topologie aléatoire de 60 nœuds

début de supertrame a un impact très négatif sur les performances. Avoir des supertrames fréquentes évitera l'accumulation de paquets dans les buffers, et évitera la forte contention au début de la supertrame, difficile à gérer avec le CSMAC/CA tel qu'il est proposé dans la norme.

L'ordonnancement temporel de supertrames permet d'augmenter les performances comparées à la version native de IEEE 802.15.4 pour les valeurs au delà de $BO = 4$. Pour des valeurs plus petites de $BO$, MeshMAC atteindra au maximum 50% en terme de PDR. Ceci vient du fait qu'avec ce protocole il n'est possible d'ordonnancer que $2^{BO-SO} - 1$. MCCT permet d'atteindre des PDR de plus 90% pour de petites valeurs de BO (entre 2 *et* 5) grâce à la parallélisation des communications sur différents canaux qui augmentera le nombre de slots disponibles à ordonnancer (jusqu'à $2^{BO-SO} * 15$ slots). Ce qui est important de retenir est qu'avec le multi-canal on pourra utiliser de petites valeurs de BO et SO, ceci garantira une faible consommation énergétique (on pourra toujours utiliser de petits duty cycles), et répartir le trafic sur les différentes supertrames, évitant que les nœuds accumulent trop de paquets dans les buffers impliquant une réduction des retransmissions liée aux collisions de trames, et ainsi une réduction de la consommation en terme d'énergie.

La figure 4.6, comme pour le PDR, montre que l'approche multi-canal permet d'avoir une meilleure équité pour les petites valeurs de BO. Ce résultat correspond à ce qui a été présenté dans la figure 4.5 représentant le PDR. Le délai sera, lui aussi, réduit en utilisant le multi-canal: moins de collisions implique moins de retransmissions et un plus court délai (cf fig. 4.7 ).

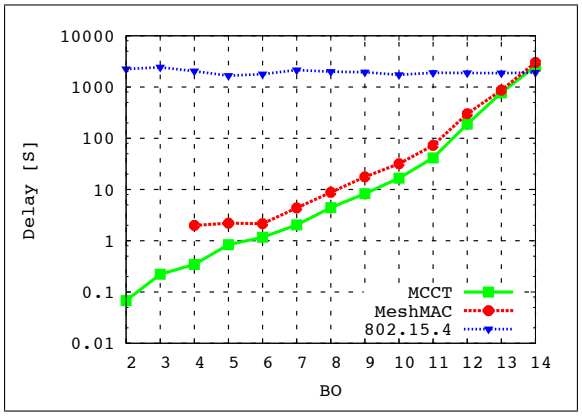


**Figure 4.7:** Délai dans une topologie aléatoire de 60 nœuds (échelle logarithmique)

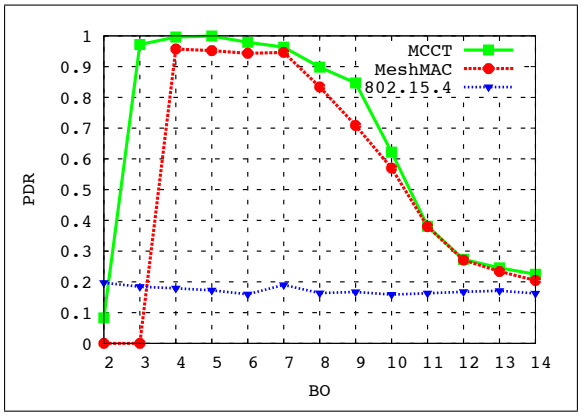


**Figure 4.8:** Packet Delivery Ratio pour une topologie en grille

### 4.6.2 Topologie en grille

Le deuxième jeu de simulations que nous proposons est fait sur une grille de nœuds de 49 nœuds (7 par 7) avec le coordinateur au centre montrent les mêmes résultats que ceux des topologies aléatoires. Les figures 4.8 et 4.9 représentent respectivement le Packet Delivery Ratio et le délais obtenu pour une la topologie en grille. Nous pouvons constater que malgré la faible densité que représente cette topologie, MCCT augmente la capacité du réseau et arrive à donner de meilleures performances que MeshMAC et IEEE 802.15.4.

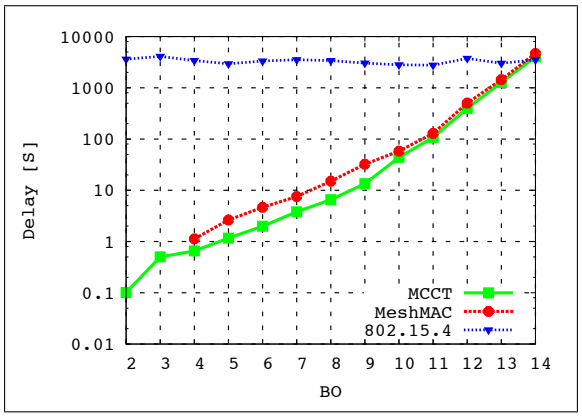


**Figure 4.9:** delai pour une topologie en grille (échelle logarithmique)

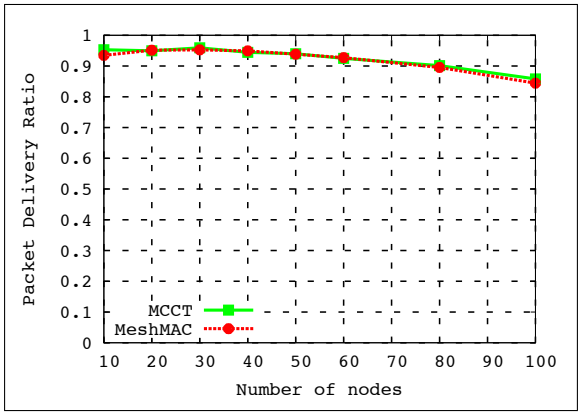


**Figure 4.10:** Scalabilité

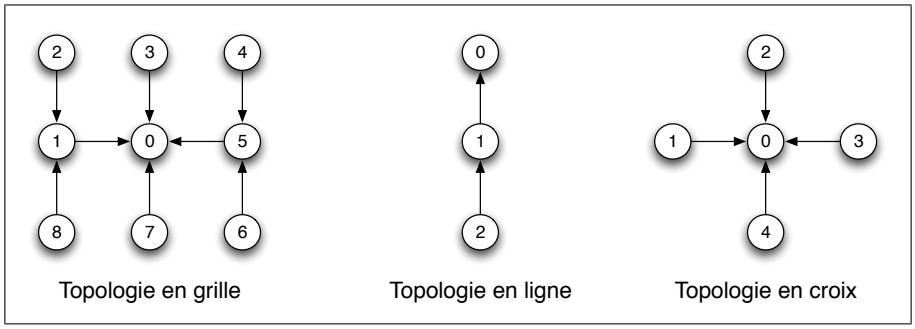


**Figure 4.11:** Exemple des topologies en grille, en ligne et en croix

### 4.6.3 Scalabilité

Nous avons voulu étudier l'impact de la taille du réseau sur les performances en terme de Packet Delivery Ratio sur les deux protocoles MCCT et MeshMac, pour cela nous avons simulé plusieurs topologies aléatoires de type cluster-tree où l'on a augmenté le nombre de nœuds de 10 à 100. Ces topologies ont été toutes générées aléatoirement mais avec de très faibles densités de voisinage i.e en moyenne 5 voisins par nœud. La figure 4.10 représente le Packet Delivery Ratio par rapport au nombre de nœuds. Nous pouvons constater que les deux protocoles réagissent très bien à l'augmentation du nombre de nœuds dans le réseau. Ceci revient au bas trafic et au grand nombre de slots disponibles pour l'ordonnancement des supertrames i.e 32 slots avec BO et SO respectivement à 7 et 2.

### 4.6.4 Impact de la densité sur les performances

Par définition, les réseaux de capteurs sont des réseaux à forte densité. Nous avons souhaité étudier l'impact de la densité de coordinateurs sur les performances en terme de Packet Delevery Ratio des deux protocoles MCCT et MeshMAC représentés sur la figure 4.12. Pour cela nous avons généré des topologies aléatoire où nous avons augmenté le degré moyen de voisins, nous avons fixé $BO = 7$ et $SO = 2$, ceci offrant un duty cycle de 3% et un nombre de 32 slots temporels ordonnançables.

La figure 4.12 montre les performances obtenues par MCCT et MeshMAC. Pour de faibles densités nous remarquons que les deux approches offrent un taux de PDR équivalent. Ceci provient du grand nombre de slots à la disposition de MeshMAC lui permettant un ordonnancement de supertrames sans collision. Cependant le PDR chutera plus rapidement avec l'augmentation de densité des coordinateurs dans le cas du protocole MeshMAC qui n'arrivera pas à ordonnancer les supertrames sans avoir des collisions de Beacons faute de slots disponibles. MCCT n'aura pas ce problème puisque chaque coordinateur aura à sa disposition 15 canaux pour chacun des 32 slots pour ordonnancer

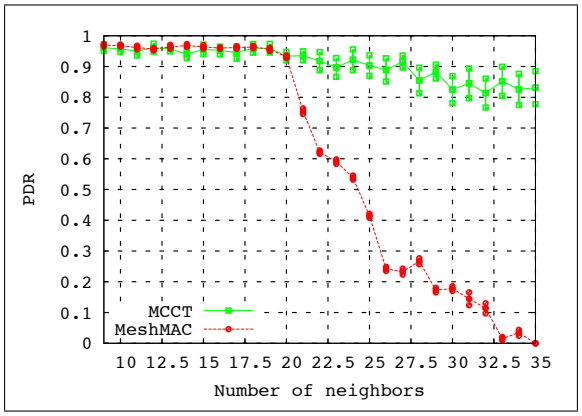


**Figure 4.12:** Impact de la densité de coordinateurs sur le Packet Delivery Ratio

sa supertrame. Avec MCCT le PDR restera entre 97% et 77% même si le nombre de coordinateurs voisins est de 35.

Nous avons aussi procédé à des simulations sur des topologies particulières: une topologie en ligne de vingt nœuds, une topologie en croix de 13 nœuds et une grille de 49 nœuds (7:7). Dans ces simulations nous avons gardé les topologies fixes, et nous avons joué sur la portée radio des nœuds afin de voir le comportement des deux protocoles MCCT et MeshMAC. Le but de ces simulations est de montrer l'intérêt d'avoir un protocole robuste face à l'augmentation de la densité.

Ces courbes tendent à confirmer que le protocole MCCT permet un déploiement de capteurs sans tenir compte de la densité des nœuds coordinateurs comme montré par la courbe 4.12, et ceci quelles que soient les topologies choisies (des topologies aléatoires ou des topologies suivant un plan de déploiement plus strict). L'intérêt de s'abstraire de la densité des nœuds dans un déploiement des réseaux de capteurs est de pouvoir choisir librement un duty cycle selon les besoins du réseau (réactivité, maximisation de la vie du réseau, etc.). Ceci n'est faisable qu'en ayant un protocole pouvant s'adapter et garantir de bonnes performances quelles que soient les valeurs de BO et SO choisies. MeshMAC est très sensible aux valeurs de ces deux paramètres: si le nombre de coordinateurs voisins dépasse le nombre de slots ordonnançables, les performances en pâtiront comme on peut le voir sur les figures 4.13, 4.14 et 4.15.

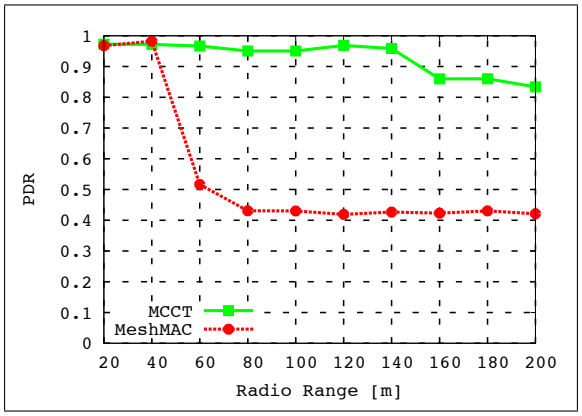


**Figure 4.13:** Impact de la densité de coordinateurs dans une topologie en croix

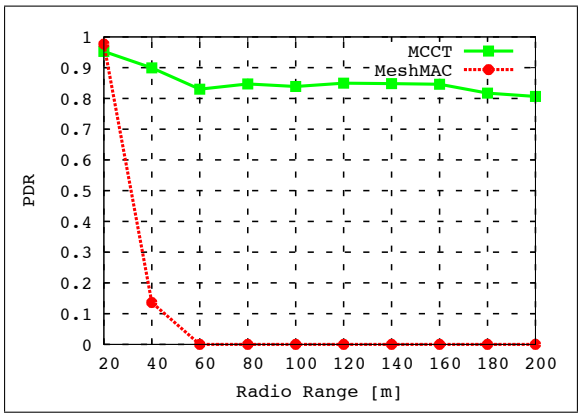


**Figure 4.14:** Impact de la densité de coordinateurs dans une topologie en grille

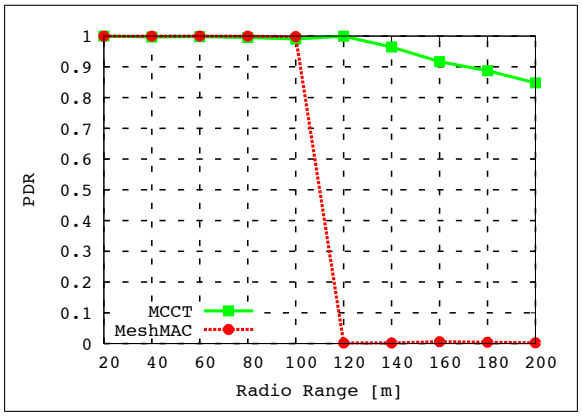


**Figure 4.15:** Impact de la densité de coordinateurs dans une topologie en ligne

## 4.7 Conclusion

Le standard IEEE 802.15.4 a été proposé afin de gérer les communications dans des réseaux contraints énergétiquement. La technique de duty cycle étant l'un des moyens les plus simples afin de garantir des économies d'énergie, elle a été retenue dans toutes les solutions MAC pour les réseaux de capteurs. Néanmoins cette technique peut mener à des collisions de Beacons dans IEEE 802.15.4 (interférences inter-cluster) et des interférences intra-cluster dues au temps réduit où les nœuds ont la possibilité de communiquer.

Dans ce chapitre nous avons proposé le protocole MCCT. Ce protocole a pour rôle de minimiser deux types d'interférences: d'une part les interférences inter-cluster en parallélisant les communications sur différents canaux fréquentiels, et d'autre part les interférences intra-cluster en imposant une politique d'association au réseau minimisant le nombre de fils par coordinateur et permettant ainsi une meilleure efficacité de la méthode d'accès.

Le protocole MCCT introduit une nouvelle trame appelée Hello, envoyée uniquement sur le canal de contrôle et servant essentiellement à la construction du réseau. L'envoi périodique de ce paquet peut être considéré de prime abord comme un overhead vu que tous les coordinateurs envoient cette trame à chaque supertrame. Le fait est que ce paquet envoyé uniquement sur le canal de contrôle permet d'éviter toute interférence avec les données utiles du réseau et de ce fait ne générera aucune réémission. D'un autre côté grâce à ce mécanisme de découverte de voisinage, nous avons pu réduire le nombre de coordinateurs envoyant des Beacons (Coordinateur actif), en introduisant la notion de coordinateurs passifs, ces derniers n'ayant pas à maintenir de supertrame tant qu'ils n'auront pas de fils. En effet, avec uniquement le tiers des coordinateurs, avec MCCT nous arrivons à maintenir un réseau et atteindre de meilleures performances en comparaison avec le standard IEEE 802.15.4 ou MeshMac. Dans les simulations des topologies aléatoires avec soixante nœuds nous n'avions que vingt coordinateurs actifs (coordinateur maintenant des supertrames) en moyenne alors que dans MeshMac et IEEE 802.15.4 tous les coordinateurs maintiennent une supertrame.

Nous pouvons aussi souligner le fait que le protocole MCCT gère plus efficacement les réseaux à forte densité, ce qui est intéressant dans un contexte de réseaux de capteurs souvent présentés comme des réseaux très denses.

Il serait intéressant de tester le protocole MCCT sur une plateforme réelle et ceci afin d'en observer le comportement dans un environnement avec la présence d'interférences extérieures.

Afin que notre protocole garde les mêmes propriétés dans ce genres de conditions, il faudrait rajouter un mécanisme de détection d'énergie pour le choix du canal qu'un coordinateur doit s'attribuer pour sa future supertrame.

Le scan à détection d'énergie sera ainsi effectué juste après la découverte de voisinage

faite par le nœud en s'associant au réseau. Ainsi si le nœud choisit un canal trop bruyant, il pourra changer de canal avant de commencer à l'annoncer sur le canal de contrôle (Hello).

Une autre adaptation pouvant être ajoutée à MCCT est le maintien de plusieurs parents par les nœuds, un parent favori et des parents secondaires, ceci pouvant être intéressant dans le cas où une couche routage est utilisée.

Nous pouvons rajouter alors dans ce cas un mécanisme utilisant la qualité du canal des différents parents comme métrique de routage.

Dans le chapitre suivant nous allons étudier la méthode d'accès du standard IEEE 802.15.4 et montrer l'impact négatif que peut avoir la concentration des communications sur de petites périodes de temps imposées par le duty cycle.

CHAPITRE 5

# ETUDE DU MÉCANISME D'ACCÈS AU CANAL PROPOSÉ DANS LE STANDARD IEEE 802.15.4

**Sommaire**


5.1 Introduction . . . . . **77**

5.2 Influence des paramètre du CSMAC/CA sur les performances de IEEE 802.15.4 . . . . . **78**

5.3 Etude de l'impact des différents paramètres utilisés dans le mode Slotted CSMA/CA sur les performances . . . . . **80**

5.3.1 macMaxCSMABackoffs . . . . . 82

5.3.2 macMaxFrameRetries . . . . . 84

5.3.3 BE (Backoff Exponent) . . . . . 84

5.4 Conclusion . . . . . **86**


## 5.1 Introduction

Comme présenté précédemment le standard IEEE 802.15.4 a été proposé pour les réseaux WPAN et fournit deux modes de fonctionnement (Beacon Enabled mode et non Beacon-Enabled mode). Nous nous concentrons sur le plus important des deux, Ce mode permet d'économiser de l'énergie efficacement grâce au mécanisme de duty cycle minimisant l'utilisation de l'interface radio. Cette dernière n'est utilisée que durant de petites portions de temps, sans pour autant pénaliser la connectivité du réseau.

Comme nous l'avons déjà exposé dans la section 3.5.6, le standard IEEE 802.15.4 utilise l'algorithme CSMA/CA comme méthode d'accès au canal: les nœuds, avant d'envoyer une trame, doivent patienter durant un temps aléatoire et vérifier si le canal est libre à la fin de ce temps d'attente afin de statuer sur l'envoi de la trame.

L'utilisation du tirage aléatoire des temps d'attente a pour but de désynchroniser les envois des différentes stations en concurrence pour le canal. Ce temps d'attente appelé temps de Backoff $B$ est toujours tiré d'une manière uniforme dans un intervalle borné conditionné par la valeur du Backoff-Exponent.

Dans ce chapitre nous proposons d'étudier les performances du standard IEEE 802.15.4 et plus exactement la méthode d'accès au canal utilisée (Slotted CSMA/CA).

Nous analyserons l'impact sur les performances des différents paramètres choisis pour la méthode d'accès au canal et les limitations introduites par ces derniers.

Comme nous allons le montrer dans ce chapitre le standard IEEE 802.15.4 n'offre de bonnes performances qu'en présence d'une faible densité de nœuds et d'un faible trafic. Cette description ne correspond pas tout à fait à l'idée que l'on se fait des réseaux de capteurs, qui sont généralement décrits comme des réseaux avec une grande densité de nœuds.

## 5.2 Influence des paramètre du CSMAC/CA sur les performances de IEEE 802.15.4

Les performances du standard IEEE 802.15.4 ont été étudiées dans un certain nombre d'articles. La majorité des articles sont des articles théoriques d'évaluation de performances. Ils se basent, pour la plupart, sur le modèle de chaine de Markov. Misic *et al.* proposent une modélisation du mode slotté de l'algorithme CSMA/CA du standard [85] dans un contexte de trafic montant (convergecast). Les auteurs s'intéressent à l'impact sur les performances du trafic, du nombre de stations, de la taille des buffers, de la taille des trames ainsi que de la longueur de BI.

Les auteurs concluent de cette étude que le choix des paramètres tels que la taille des trames, l'intensité du trafic ainsi que la taille du réseau est très important afin de maximiser le débit. En effet il ressort de leur étude qu'augmenter la taille des trames a comme résultat de baisser la probabilité d'accès au canal et d'augmenter la probabilité que le canal reste inutilisé. Ce phénomène s'accentue avec l'augmentation du nombre de nœuds dans le réseau.

Dans [50] les mêmes auteurs ont étudié cette fois-ci l'impact des conditions de trafic, de topologie, de la taille des buffers, de la taille des supertrames sur le mode Beacon du standard IEEE 802.15.4 mais avec cette fois-ci un trafic montant (uplink) et descendant (downlink).

Leur étude montre que le trafic descendant doit être restreint, un coordinateur ne pouvant gérer efficacement ce genre de trafic que s'il est d'une petite intensité. Après réception de la requête demandant les données en attente chez le coordinateur, ce dernier doit tirer un temps de backoff afin d'envoyer ses trames. Durant ce temps la radio étant éteinte toute trame venant des autres nœuds sera perdue. Ceci vient s'ajouter aux possibles collisions entre les trames de données et les requêtes envoyées au coordinateur,

ce qui causera une augmentation des délais.

Park *et al.* [86] proposent une approche similaire avec chaîne de Markov afin d'étudier les performances du standard IEEE 802.15.4 en terme de fiabilité, délais et consommation énergétique. Les résultats montrent que la fiabilité du réseau augmente avec l'augmentation des deux paramètres macMinBE et macMaxCSMABackoff ainsi que macMaxFrameRetries. Néanmoins les auteurs soulignent que même si l'augmentation de ce paramètre a un impact positif sur les performances ce n'est pas suffisant pour des applications denses en terme de trafic.

Les auteurs montrent que l'augmentation des valeurs des trois paramètres discutés ci-dessus réduit la consommation énergétique dans des applications à fort trafic. Effectivement les nœuds passeront la majorité du temps en Backoff (radio éteinte) avant l'envoi de la trame. Le fait d'avoir un grand macMinBE maximisant la probabilité que la trame arrive à destination au bout d'une tentative, des retransmissions et du gaspillage d'énergie pourront être évités. Néanmoins cette augmentation aura comme résultat logique l'augmentation du délai moyen.

D'autres travaux ont été effectués afin d'étudier par simulation les performances du standard IEEE 802.15.4: Koubaa *et al.* [87] ont étudié la topologie en étoile, dans le but de montrer l'impact de BO et SO sur le CSMA/CA slotted, le nombre de nœuds et la taille des paquets.

Les auteurs montrent que le choix de la valeur du paramètre macMinBE à un grand impact sur le délai et le débit. Les auteurs concluent que l'algorithme avec les paramètres par défaut n'est pas adapté pour de grands réseaux. Anastasi *et al.* [52] aussi grâce aux simulations qu'ils ont effectués, sont arrivés aux mêmes conclusions concernant les paramètres utilisés dans la méthode d'accès du standard IEEE 802.15.4 i.e le standard IEEE 802.15.4 offre des performances plutôt médiocres que ce soit dans des réseaux en étoile ou en multi-saut. Ceci est principalement dû au fait que les valeurs par défaut de BE (Backoff exponent) ne permettent pas de gérer efficacement l'accès au canal sauf dans le cas de réseaux de très petite taille.

De la même manière Koscielnik *et al.* montrent dans [88] que les stations cachées et exposées ont un impact négatif sur le débit dans IEEE 802.15.4, les auteurs préconisent l'ajout d'un mécanisme tel que RTS/CTS pour éviter ce genre de situation, sachant qu'aucun mécanisme n'a été proposé dans le standard afin d'y pallier.

D'autres auteurs ont préféré faire des expérimentations sur des capteurs réels: dans [89] Lee *et al.* proposent d'étudier l'impact sur les performances des modes de transfert de données (Direct et Indirect), de la taille des paquets, ainsi que du CSMA/CA slotted. Les auteurs montrent avec leurs expérimentations que la méthode d'accès est fortement influencée par le nombre de nœuds dans le réseau générant du trafic et la taille des trames: plus il y a de nœuds plus le PDR diminue, l'augmentation de la taille des trames ayant le même résultat à cause de l'augmentation de la probabilité de collision. Le délai augmentera aussi de par l'augmentation de la taille des fenêtres de contention

due aux multiples essais effectués par les différentes stations afin d'envoyer leurs trames.

Liu *et al.* [80] montrent grâce à leurs expérimentations que les performances du standard IEEE 802.15.4 en terme de probabilité de réception des trames par la destination sont très médiocres dans un contexte de réseau en étoile et multi-saut. La probabilité de délivrer une trame ne dépassera pas 15% dans un réseau avec un nombre de sauts dépassant 5. Ils démontrent aussi que le débit et le PDR se dégradent rapidement avec l'augmentation du nombre de nœuds et cela même dans des topologies en étoile.

En résumé toutes ses études convergent vers le même constat qui est que la méthode d'accès au canal proposée dans IEEE 802.15.4 est fortement dépendante:

- De la topologie (nombre de nœuds, densité de voisinage, etc.)
- De l'intensité du trafic
- De la taille des trames
- Du duty cycle

Une des solutions retenues par les différents auteurs pour améliorer les performances sera donc d'augmenter la valeur du paramètre macMinBE et macMaxBE.

## 5.3 Etude de l'impact des différents paramètres utilisés dans le mode Slotted CSMA/CA sur les performances

Comme expliqué dans la section 4.5.3 , le standard IEEE 802.15.4 ne peut gérer efficacement la contention qu'en présence d'un trafic faible et dans des cas où il n'y a pas trop de nœuds en contention. La figure 5.1 montre l'impact du nombre de paquets à écouler en début de supertrame sur les performances de la méthode d'accès en terme de Packet Delivery Ratio pour différents nombres de nœuds. Nous pouvons constater grâce à cette figure que tant que le nombre de fils ne dépasse pas 4 nœuds la couche MAC arrive à garantir un PDR autour de 90%, mais dès que ces 4 nœuds génèrent plus de trafic (plus d'un paquet au début de la supertrame) les performances se dégradent: avec un nombre de paquets de 7 on voit qu'il y a une baisse de l'ordre de 20% de PDR. Ceci se confirme avec l'augmentation du nombre de nœuds dans un cluster: avec 8 nœuds par exemple la couche MAC ne fera pas plus de 60% de PDR dans les meilleures conditions. Ceci revient à la concentration des trames en début de supertrame et à l'impossibilité de la méthode d'accès à gérer ce genre de situations.

Les paramètres de base choisis pour le fonctionnement de l'algorithme CSMA/CA ont un impact non négligeable sur les performances du standard IEEE 802.15.4 en terme de débit, PDR, délais et consommation énergétique vu qu'un nœud qui fait une retransmission de trame à cause des collisions aura tendance à consommer plus d'énergie qu'un nœud réussissant à transmettre sa trame au bout du premier essai.

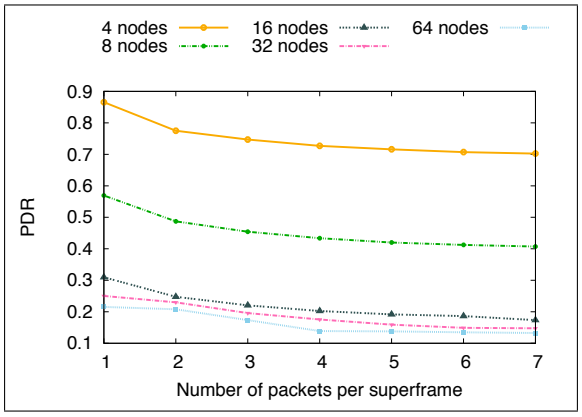


**Figure 5.1:** Impact du trafic sur le PDR

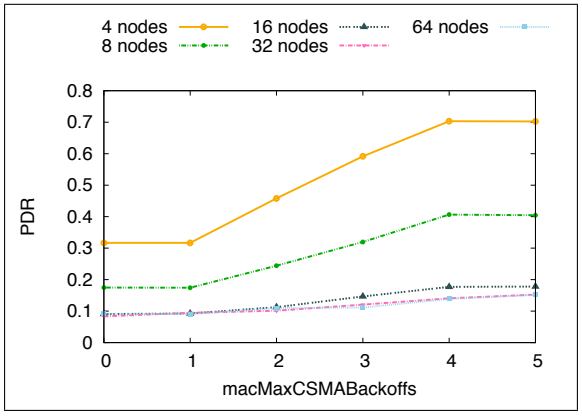


**Figure 5.2:** Impact du paramètre macMaxCSMABackoffs sur le PDR

Nous présentons dans ce qui suit les différents paramètres et leur impact sur les performances en terme de délais, PDR et débit.

Les paramètres utilisés dans le fonctionnement de l'algorithme CSMA/CA sont:

**`macMaxCSMABackoffs`** : Le nombre maximum de fois où le nœud testera si le canal est occupé avant d'abandonner l'émission de la trame en cours de transmission

**`macMaxFrameRetries`** : Le nombre maximum de retransmissions que le nœud effectuera en l'absence de réception d'acquittement avant d'abandonner l'envoi de la trame

**`BE (Backoff Exponent)`** : Détermine la longueur de l'intervalle (fenêtre de contention) dans lequel le nœud doit tirer le temps d'attente aléatoire qu'il devra attendre avant la transmission d'une trame (Backoff)

### 5.3.1 macMaxCSMABackoffs

Le paramètre macMaxCSMABackoffs représente le nombre de CCA (Clear Channel Assessment) qu'un nœud effectuera avant d'abandonner la transmission d'une trame pour cause de canal occupé.

Afin de montrer l'impact de ce paramètre sur la méthode d'accès utilisée dans le standard IEEE 802.15.4 nous avons simulé plusieurs topologies en étoile où nous avons augmenté la densité de nœuds autour du coordinateur. Chaque nœud ayant un paquet à envoyer en début de supertrame, nous avons fait varier la valeur de macMaxCSMABackoffs de 0 à 5. La figure 5.2 représente le PDR correspondant aux différentes valeurs de macMaxCSMABackoffs.

Augmenter la valeur macMaxCSMABackoffs augmente les performances en terme de PDR (cf. Fig. 5.2) et ceci quelles que soit le nombre de nœuds autour du coordinateur. Ceci s'explique par le fait qu'augmenter la valeur de macMaxCSMABackoffs revient à donner plus de chances aux nœuds de transmettre leurs trames.

Sachant que la probabilité que les nœuds génèrent des paquets durant la période active de la supertrame est très faible, et compte tenu du duty cycle (de l'ordre de 1%), la probabilité qu'un nœud sonde le canal et le trouve occupé en début de supertrame est très élevée.

Ceci implique que tous les nœuds participant à cette supertrame rentreront en compétition afin de transmettre leurs paquets. A cause de cette forte pression sur le canal en début de supertrame les nœuds trouveront le canal occupé la plupart du temps et plus la valeur du paramètre macMaxCSMABackoffs est petite plus ils abandonneront rapidement la transmission de la trame en cours, ce qui aura un impact négatif sur les performances, ce qui nous pousse à conclure que plus la valeur de ce paramètre est grande meilleures seront les performances.

La figure 5.3 est le résultat de la simulation d'un réseau en étoile où l'on a fait varier le trafic de un à sept paquets par supertrames.

La courbe représente le nombre d'abandons d'envois de paquets dus à un nombre d'essais dépassant les valeurs des paramètres macMaxFrameRetries et macMaxCSMABackoffs fixés à leur valeur par défaut par le standard soit respectivement 3 et 4. Dans un réseau de type IEEE 802.15.4 utilisant le mode slotted (utilisation de duty cycle) l'envoi des trames sera abandonné dans la majorité des cas parce que les stations trouvent le canal occupé.

On peut qualifier cette méthode d'accès comme très agressive puisqu'un nœud devra abandonner la transmission de ses paquets non pas en cas de multiples collisions ou en cas de non réception de l'acquittement mais dans la majorité des cas juste parce que le canal est occupé par l'envoi de paquets.

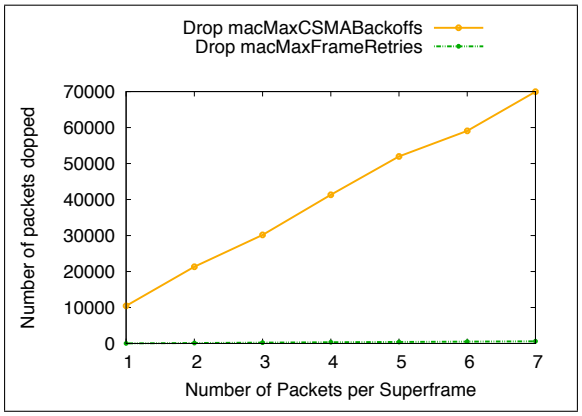


**Figure 5.3:** Nombre d'abandons de transmission dus aux paramètres macMaxCSMABackoffs et macMaxFrameRetries

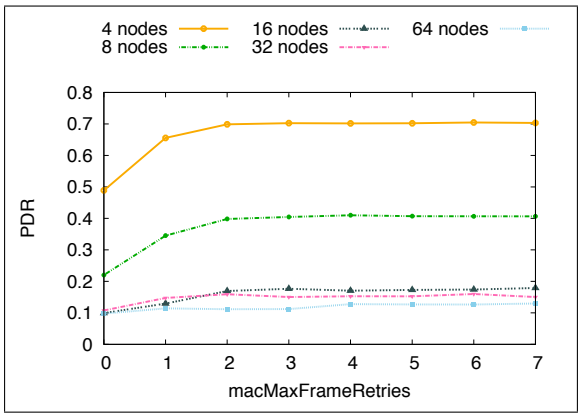


**Figure 5.4:** Impact du paramètre macMaxFrameRetries sur le PDR

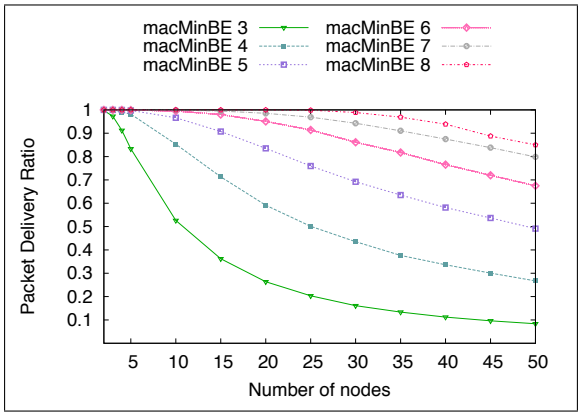


**Figure 5.5:** Impact du paramètre BE sur le Packet Delivery Ratio

### 5.3.2 macMaxFrameRetries

Les hypothèses de travail utilisées pour les simulations sur le paramètre macMaxCAMABackoff ont été reprises pour celles concernant le paramètre macMaxFramesRetries i.e un réseau en étoile avec augmentation de la densité de nœuds. Pour montrer l'impact de macMaxFrameRetries nous en avons fait varier la valeur dans l'intervalle [0,7].

La figure 5.4 représente le PDR atteint dans les différentes topologies en fonction de la valeur du paramètre macMaxFrameRetries.

Nous pouvons déduire grâce à cette figure que le paramètre macMaxFrameRetries n'a aucun impact sur les performances dès lors que la valeur choisie est supérieure ou égale à 2, quelles que soit le nombre de nœuds autour du coordinateur le PDR restant stable à partir de cette valeur. Ce qui implique que ce paramètre ne doit pas être considéré dès qu'il dépasse une valeur de 2.

### 5.3.3 BE (Backoff Exponent)

Afin de montrer l'impact du paramètre `BE (Backoff-Exponent)`, qui est le paramètre conditionnant la taille des fenêtres de contention dans l'algorithme CSMA/CA, nous avons procédé à deux jeux de simulations sur une topologie en étoile:

- Soit nous avons augmenté le nombre de nœuds autour du coordinateur, chaque nœud ayant un paquet à envoyer au début de chaque supertrame
- Soit nous avons simulé l'impact d'un trafic saturé où tous les nœuds ont toujours dans leur buffer un paquet à envoyer (trafic Backlog)

La Figure 5.5 représente l'impact du paramètre macMinBE sur le PDR dans un réseau en étoile avec un trafic non saturé, alors que les deux Figures 5.7 et 5.6 représentent l'impact de macMinBE sur le débit dans un contexte de trafic saturé.

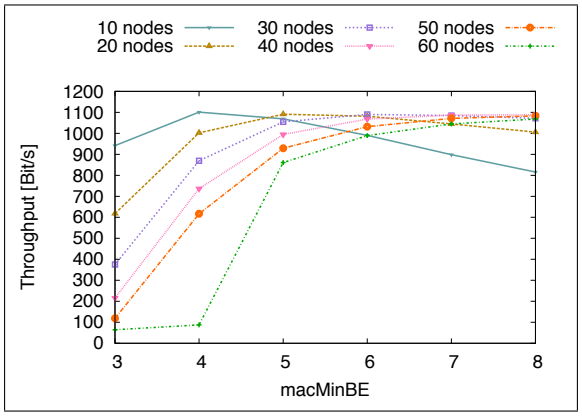


**Figure 5.6:** Impact du nombre de nœuds sur le débit

La Figure 5.5 montre que plus le nombre de nœuds augmente (avec n'importe quelle valeur de macMinBE) plus les performances se dégradent. Nous pourrions déduire qu'il est plus intéressant de choisir la valeur maximale de BE permise par le standard, à savoir 8, car cela permet de maximiser maximise le PDR.

Ceci n'est vrai que dans des applications n'ayant aucune contrainte temporelle, et où le trafic est assez bas pour ne pas remplir les tampons (buffers).

Le fait d'augmenter la valeur de BE à son maximum aura comme résultat d'augmenter la taille des fenêtres de contention, réduisant ainsi les probabilités de collisions mais en contrepartie augmentera les délais et donc baissera le débit.

Afin de maintenir des performances optimales en terme de PDR ainsi qu'en terme de délais il faudrait choisir de maximiser le débit au lieu du PDR. En effet maximiser le débit revient à maximiser l'utilisation de la bande passante permettant aux nœuds d'atteindre un PDR optimal tout en minimisant les délais.

La Figure 5.6 représente l'impact du nombre de nœuds sur le débit. Elle montre que la valeur optimale de macMinBE maximisant le débit change avec le nombre de nœuds.

En représentant le débit atteint dans une topologie en étoile en fonction du nombre de nœuds (cf. Fig 5.7) nous pouvons remarquer que la valeur optimale du débit se déplacera avec l'augmentation du nombre de nœuds émetteurs. Ce qui prouve que pour une quantité de nœuds donnée autour d'un coordinateur il existe une valeur du paramètre macMinBE maximisant le débit.

Si par exemple 5 nœuds sont en concurrence pour accéder au canal, maximiser le débit revient à choisir la valeur de macMinBE=3, alors que pour un nombre de 50 ou 60 nœuds la valeur maximisant le débit sera alors de macMinBE=8.

La courbe représentée en figure 5.7 montre l'importance du choix de la valeur de BE et son impact sur le débit. Le lecteur remarquera que le débit est ainsi divisé par 10 en passant d'un nombre de 10 nœuds à un nombre de 60 nœuds autour du coordinateur

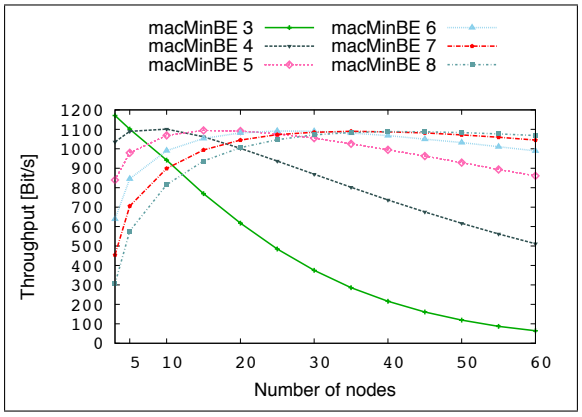


**Figure 5.7:** Impact du paramètre macMinBE sur le débit

(avec macMinBE=3 le débit passe de 1000 Bit/s à moins de 100 Bit/s).

## 5.4 Conclusion

Dans ce chapitre nous avons étudié la méthode d'accès utilisée dans le mode Beacon du standard IEEE 802.15.4. Les simulations que nous avons effectuées ainsi que les différentes études pratiques ou théoriques montrent que les paramètre utilisés dans l'algorithme CSMA/CA ont un grand impact sur les performances en terme de débit et de Packet Delivery Ratio.

Nous avons pu constater que les collisions ont un impact négatif sur le débit, impliquant un gaspillage de bande passante, d'énergie et augmentant les délais. Néanmoins dans le standard IEEE 802.15.4, la baisse des performances est due dans la majorité des cas non pas aux collisions mais à l'abandon des transmissions de la part des nœuds trouvant le canal toujours occupé comme le montre la Figure 5.3.

Nous pouvons déduire que le choix des paramètres par défaut définis par le standard soit: (macMinBE=3, macMaxCSMABackoffs=4 ) ne permettent pas d'atteindre des performances optimales dans toutes les conditions de topologie et de trafic.

En effet, il faudra lors du déploiement d'un réseau de capteurs tenir compte, dans un but de maximiser les performances, de l'impact des deux paramètres:

- macMinBE servant à augmenter la taille des fenêtres de contention, dans le but de minimiser la probabilité de trouver les canaux occupés et la probabilité de collisions
- macMaxCSMABackoffs permettant l'augmentation du nombre d'essais permis par la couche MAC avant d'abandonner l'émission de la trame

Ce choix définira les performances pouvant être atteintes par le réseau. En effet, une valeur constante et optimale des paramètres n'existant pas, nous allons étudier

dans le prochain chapitre des méthodes d'auto-adaptation dans le but d'optimiser les performances.

Il faut aussi tenir compte du mécanisme de duty cycle. En effet à cause de la contrainte énergétique, les capteurs essayent de maintenir le plus souvent possible la radio éteinte.

Dans IEEE 802.15.4 ceci est obtenu en jouant sur les valeurs des paramètres BO et SO. Ce mécanisme permet d'économiser l'énergie mais il a comme principal défaut de réduire les temps prévus pour les communications et ainsi la capacité du réseau. Les nœuds auront tendance à emmagasiner les paquets dans leur buffer durant la période où les radios sont éteintes. Ceci implique que toutes les stations ayant des paquets dans leur buffer rentrent en compétition dès le début de la partie active afin d'envoyer les paquets.

Le problème introduit par le duty cycle peut être géré en choisissant les bonnes valeurs pour les paramètres BO et SO en fonction du trafic. Comme expliqué précédemment, il vaut mieux avoir des BO/SO petits que le contraire ceci permettant d'avoir des supertrames plus fréquentes ce qui aura un double avantage:

- Distribuer le trafic des différentes stations sur différentes supertrames, afin d'éviter que les nœuds ne gardent les trames dans la mémoire tampon (Buffer)
- Offrir une plus grande réactivité du réseau en baissant la latence

# CHAPITRE 6

# AUTO-ADAPTATION DES PARAMÈTRES DU CSMA/CA

**Sommaire**


6.1 Introduction . . . . . **89**
6.2 Auto-adaptation des paramètres du CSMA/CA dans la littérature **92**
6.3 Première proposition . . . . . **94**
6.3.1 Evaluation de la proposition . . . . . 95
6.3.2 Comportement dynamique de l'auto configuration . . . . . 95
6.3.3 Limites de la proposition . . . . . 98
6.4 Deuxième proposition: SABE (Self Adapting BE) . . . . . **100**
6.4.1 Analyse . . . . . 100
6.4.2 Validation par simulations . . . . . 102
6.4.3 SABE (Self Adapting BE) . . . . . 103
6.4.4 Adaptation pour les rafales (bursts) de trafic . . . . . 105
6.4.5 Adaptation des CCAs . . . . . 106
6.4.6 Longest remaining time first (LRTF) vs Short remaining time first (SRTF) . . . . . 107
6.4.7 Evaluation de la proposition . . . . . 107
6.5 Conclusion . . . . . **111**


## 6.1 Introduction

Comme cela a déjà été expliqué dans [48] et l'étude que nous avons effectuée dans le chapitre précédent, l'utilisation de la valeur par défaut du paramètre macMinBE=3

préconisée par le standard aura un impact négatif sur les performances, cette valeur étant trop petite dans la majorité des cas.

Réduire au maximum l'utilisation du module radio reste la solution la plus efficace pour la réduction de l'énergie consommée par les nœuds. En contrepartie ceci réduit le temps où les nœuds peuvent transmettre leurs données, réduisant ainsi la capacité du réseau. Un autre point négatif est la forte pression sur le canal au début des supertrames.

Cette concentration de communication au début de la supertrame est due à la bufferisation (la mise en tampon) des trames par les nœuds durant la partie inactive de la supertrame. Dans cette situation la méthode d'accès aura du mal à gérer les communications qui se traduiront dans la majorité des cas par l'abandon de l'envoi des trames à cause de l'occupation du canal.

Nous pouvons donc conclure qu'il est indispensable de ne pas se restreindre à l'utilisation de valeurs fixes pour les paramètres macMinBE et macMaxCSMABackoffs, mais de jouer sur ces derniers afin d'optimiser les performances en terme de débit et PDR.

Il serait intéressant d'avoir un algorithme permettant l'adaptation des paramètres définis dans le standard afin d'offrir de meilleures performances.

Dans ce chapitre nous nous intéressons à la proposition de deux algorithmes dont le but est de permettre à la méthode d'accès proposée dans le CSMA/CA du standard IEEE 802.15.4 de s'auto adapter aux conditions du réseau pour offrir des performances qui tendent à être optimales et cela quelles que soient le nombre de nœuds, la topologie et la quantité de trafic.

Nous souhaitons maximiser le débit du réseau afin de maximiser l'utilisation de la bande passante, augmentant ainsi les performances et minimisant l'énergie gâchée dans la gestion de la contention et des retransmissions.

L'adaptation des paramètres du CSMA/CA comme nous le verrons dans la section suivante peut se faire de différentes façons i.e l'utilisation du PDR comme métrique dans [90, 91] , ou l'observation de l'état du canal dans [92] .

De notre côté nous proposons deux techniques qui permettent d'adapter la valeur du paramètre macMinBE, toutes deux orientées coordinateur (adaptation initiée par le coordinateur). La première technique se base sur l'analyse du débit reçu par le coordinateur afin d'adapter le paramètre macMinBE du cluster. Dans la deuxième technique nous proposons que l'adaptation se fasse en fonction de la quantité de trafic échangé dans la supertrame mais cette fois-ci ce ne sera pas le débit qui sera utilisé comme métrique mais la moyenne de slots libres durant la supertrame.

Nos deux techniques d'adaptation sont orientées coordinateur. A l'encontre des propositions présentées ci-dessus, dans les deux algorithmes que nous proposons la décision d'adapter la valeur du paramètre macMinBE sera prise par le coordinateur de cluster. Selon les conditions observées durant une supertrame, le coordinateur devra prendre une décision pour la future valeur de macMinBE et l'envoyer à ses fils dans le Beacon. De cette manière la valeur de macMinBE utilisée dans un cluster sera identique

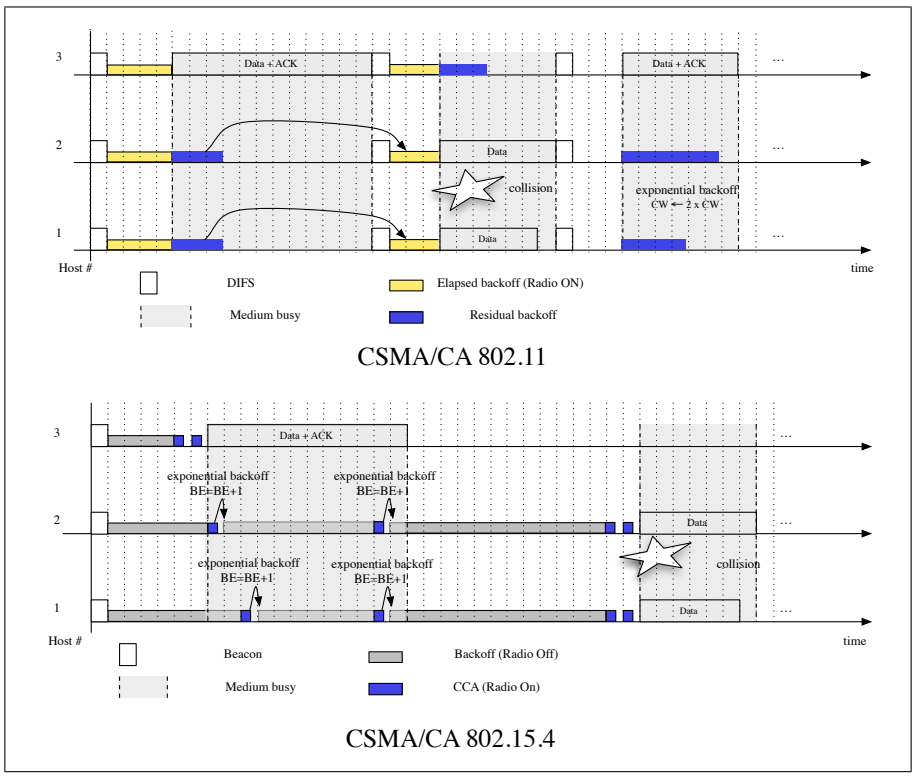


**Figure 6.1:** Fonctionnement du CSMA/CA dans IEEE 802.11 et IEEE 802.15.4

pour tous les nœuds le constituant, ce qui garantit une équité d'accès au canal pour tous les nœuds du cluster.

Par ailleurs, le coordinateur est le seul à rester éveillé durant toute la partie active de la supertrame, ceci justifie ainsi de centraliser la décision. En effet, seul le coordinateur possède une vision réelle de l'activité, les autres nœuds pouvant éteindre leur radio s'ils ne sont engagés dans aucune communication. Dans ces conditions le coordinateur est le seul nœud pouvant prendre une telle décision. Nous démontrerons dans ce chapitre que la méthode d'accès du standard IEEE 802.15.4 peut être modélisée de la même façon que la méthode d'accès utilisée dans 802.11 et cela malgré les différences de ces deux méthodes d'accès.

En effet durant le temps d'attente aléatoire dans IEEE 802.15.4 (Backoff), les stations n'ont aucun moyen de connaître l'état du canal, leur radio n'étant pas allumée. A l'arrivée d'une trame devant être envoyée, le nœud calcule son temps d'attente, garde sa radio éteinte durant tout ce temps, ne la rallumant qu'à la fin pour tester l'état du canal (CCA).

Au contraire dans le standard IEEE 802.11, les nœuds gardent toujours la radio allumée et ceci même pendant le Backoff. Si un nœud détecte une transmission il arrêtera

la décrémentation du temps de Backoff et ne reprendra qu'à la fin de cette transmission: ceci est appelé le Backoff résiduel et représente une des différences majeures entre les CSMA/CA du standard 802.11 et IEEE 802.15.4 (fig. 6.1 ). Dans la méthode d'accès au canal de IEEE 802.15.4 le fait de trouver le canal occupé par une transmission implique un doublement de la taille de la fenêtre de contention. Si le canal est trouvé occupé sur quatre essais successifs cela implique l'abandon de la transmission de la trame, à l'inverse du comportement des nœuds dans IEEE 802.11.

Une autre différence réside dans le fait qu'une station utilisant le CSMA/CA de IEEE 802.15.4 découvrant le canal occupé et cela que ce soit en CSMA/CA Slotted ou Unslotted, double la fenêtre de contention. Cela impliquent une iniquité à court terme, puisque la station qui n'a pas pu accéder au canal a moins de chance d'envoyer sa trame rapidement. A l'inverse, le CSMA/CA de 802.11, ne double sa fenêtre de contention qu'en cas de collision.

## 6.2 Auto-adaptation des paramètres du CSMA/CA dans la littérature

Un certain nombre d'auteurs ont étudié la faisabilité d'un algorithme auto adaptatif des paramètres de l'algorithme CSMA/CA du standard IEEE 802.15.4. Dans cette section nous présentons les principales propositions ayant été publiées.

Park *et al.* [92] proposent un modèle analytique afin de résoudre le problème d'optimisation du choix des valeurs des paramètres du CSMA/CA influant sur l'énergie consommée, la fiabilité et le délai i.e macMaxFrameRetries, macMaxCSMABackoffs et macMinBE.

L'algorithme proposé a pour but d'adapter les valeurs des paramètres du CSMA/CA, en estimant la probabilité que le canal soit occupé durant la période de CCA. Cet algorithme a pour rôle de maximiser la durée de vie du réseau tout en fournissant une bonne fiabilité au réseau et en minimisant les délais. Le désavantage de cette approche est que chaque nœud peut prendre une décision localement, ce qui peut mener à une iniquité en terme d'accès au canal.

Severino *et al.* [93] proposent afin d'offrir un support pour la qualité de service dans les réseaux de capteurs de faire de la différentiation de services. Les auteurs proposent de traiter différemment les paquets ayant une contrainte temporelle et les autres paquets (trafic temps réel et trafic best effort) dans des réseaux utilisant le standard IEEE 802.15.4.

Afin de respecter les contraintes temporelles liées à la priorité des paquets, les auteurs proposent d'ajuster les valeurs du paramètre CW (fixé à 2 aUnitBackoffPeriod dans le standard). Ainsi pour chaque type de trafic (haute priorité et faible priorité) ils appliquent des valeurs différentes à ce paramètre. Cette stratégie augmente le taux de trames haute priorité émises mais en contre partie diminue celle des trames moins prioritaires.

Rao *et al.* [94] proposent un autre algorithme pour l'adaptation des valeurs des paramètres macMinBE et macMaxBe (respectivement 3 et 5) dans le standard.
L'adaptation de ces valeurs est initiée par le coordinateur et l'algorithme passe par trois phases. La première phase est une phase d'analyse étalée sur trois supertrames où le coordinateur compte le nombre de paquets envoyés par chaque nœud du réseau afin de voir le taux de participation de chacun. La deuxième phase est la phase de décision, au cours de laquelle le coordinateur alloue les nouvelles valeurs de macMinBE grâce aux informations recueillies durant la phase d'analyse. Les nœuds sont divisés en deux groupes:

- Le premier groupe réunit les nœuds ayant été les moins actifs durant la supertrame, qui doivent décrémenter leur valeur de macMinBE

- Le deuxième groupe est formé des nœuds ayant le plus participé à la supertrame et qui doivent incrémenter leur macMinBE de 1

Ces informations (groupes) sont alors envoyées par le coordinateur dans le Beacon. La troisième phase commence à la réception du Beacon: les nœuds adaptent leur valeur de macMinBE selon les directives du coordinateur. Cependant ce protocole génère un trafic de contrôle important (overhead) imposé par l'envoi de toutes les adresses des nœuds du cluster afin de leur notifier la nouvelle valeur de macMinBE qu'ils doivent utiliser. Par ailleurs, du fait de la longueur de la phase d'analyse (trois supertrames) à la réception de la décision du coordinateur les conditions de trafic dans le réseau auront très probablement changé. Ce qui fait de cet algorithme un algorithme adapté uniquement aux réseaux avec un trafic qui ne varie pas dans le temps.

Anastasi *et al.* [90, 91] proposent une autre manière pour adapter les paramètres de l'algorithme CSMA/CA. Les auteurs proposent un algorithme cross-layer qui se base sur les besoins en terme de performances souhaitées par la couche application afin d'adapter les paramètres du CSMA/CA. Les auteurs estiment que le problème d'adaptation doit être pris en charge par la couche application qui fixe deux seuils à respecter en terme de PDR. Chaque station adaptera les valeurs des paramètres macMinBE et maxCSMABackoffs indépendamment des autres stations du réseau. La couche application fixe deux seuils, un seuil minimum et un seuil maximum au-delà duquel cette dernière demandera à la couche MAC de baisser la valeur du paramètre macMinBE ou maxCSMABackoffs et ainsi réduire les performances. Les auteurs jugent en effet qu'un Packet Delivery Ratio trop grand équivaut à un gaspillage de bande passante ; au contraire si la valeur du PDR baisse en dessous de la valeur minimum fixée par la couche application cette dernière demandera à la couche MAC d'augmenter la valeur de macMinBE. Ceci peut mener à un problème d'équité vu que les nœuds d'un même cluster n'auront pas les mêmes paramètres de CSMA/CA et ainsi pas la même chance d'accéder au canal.

## 6.3 Première proposition

Dans le mode Beacon du standard IEEE 802.15.4, avec l'utilisation de la supertrame, le temps où les nœuds peuvent communiquer est restreint à la période active, qui est dans la majorité des cas très courte. Comme cela a déjà été énoncé afin de maximiser les performances en terme de débit, délai et énergie il faut maximiser l'utilisation du canal, en évitant des temps de silence trop longs tout en minimisant les collisions.

Le premier algorithme que nous proposons se base sur l'exploitation de la valeur du débit reçu par le coordinateur. Maximiser ce débit revient à utiliser au maximum la bande passante. En surveillant l'évolution du débit reçu à la fin de chaque supertrame le coordinateur sera capable de changer la valeur du paramètre macMinBE afin de maximiser le débit de la future supertrame de son cluster.

Les nœuds dans le standard IEEE 802.15.4 initialisent les valeurs de macMinBE à 3 et macMaxBE à 5 et utilisent indéfiniment ces valeurs. Dans notre proposition nous procédons de la même façon pour macMinBE i.e 3 mais nous initialisons macMaxBE à 8.

Néanmoins à la fin de la supertrame le coordinateur au lieu de garder la valeur de macMinBE fixe, la changera en l'incrémentant de un. Le changement de cette valeur aura normalement un impact sur le débit de la nouvelle supertrame. De ce fait selon si à la fin de la supertrame:

- Le débit augmente: le coordinateur incrémente de un la valeur de macMinBE
- Le débit diminue: le coordinateur décrémente de un la valeur de MacMinBE
- Le débit est stable: le coordinateur garde la valeur de macMinBE en cours d'utilisation

En effet chaque coordinateur devra explorer différentes valeurs de macMinBE (incrémenter/décrémenter), selon si le débit augmente ou diminue. Le coordinateur continuera son exploration dans la direction choisie (augmentation ou diminution de la valeur de macMinBE) tant que le débit augmente. Dans le cas contraire le coordinateur inverse le sens de l'exploration.

Notre algorithme explore continuellement de nouvelles valeurs de macMinBE, ceci implique qu'il oscillera autour de la valeur optimale de macMinBE. La figure 5.7 montre que le fait d'osciller autour de la valeur optimale de macMinBE n'a qu'un impact minime.

Si le débit observé par le coordinateur est stable d'une supertrame à une autre (plus ou moins 1 trame par supertrame) et afin d'éviter des explorations non nécessaires pouvant mener à du gaspillage de bande passante ou à des collisions, le coordinateur pourra figer la valeur de macMinBE et cela au maximum pour trois supertrames consécutives avant d'entamer une nouvelle exploration. Cette dernière est faite aléatoirement en augmentant ou diminuant la valeur de macMinBE, ceci dans le but de voir si les conditions de trafic n'ont pas changé.

L'algorithme 1 représente le fonctionnement du mécanisme d'adaptation.

Les lignes 2 à 5 représentent les valeurs d'initialisation de l'algorithme.

Comme exposé précédemment les nœuds reçoivent le macMinBE à utiliser durant la supertrame en cours de la part du coordinateur qui rajoute cette information dans le Beacon (lignes 6-10).

Le coordinateur du cluster doit à chaque fois redéfinir la valeur du paramètre macMinBE à envoyer à ses fils:

- Si le débit décroit, le coordinateur change sa direction d'exploration (lignes 19-21)
- Si le débit reste stable (plus ou moins $\Delta$) ou se stabilise durant plusieurs supertrames (3 dans nos simulations), la valeur de macMinBE est changée en fonction de la dernière direction d'exploration (ligne 22)
- Si le débit reste stable par rapport à la supertrame précédente, le coordinateur continue à utiliser la valeur de macMinBE actuelle et l'envoie à ses fils
- Le coordinateur enregistre la valeur du débit obtenue dans la supertrame en cours (ligne 26) et envoie la valeur calculée de macMinBE dans son Beacon (ligne 25)

### 6.3.1 Evaluation de la proposition

Afin d'évaluer les performances de l'algorithme que nous proposons, nous l'avons comparé à la version originale du standard IEEE 802.15.4 pour différentes valeurs du paramètre macMinBE.

Nous avons mesuré le débit (trafic reçu par le puits) pour les simulations où le trafic est saturé, et nous avons rajouté le PDR pour les simulations avec un trafic non saturé.

Nous avons simulé deux types de topologies: une topologie en étoile où l'on a augmenté le nombre de nœuds de 5 à 60 autour du puits, avec un trafic saturé; et une topologie multisaut, composée d'une centaine de nœuds générant un trafic périodique de type CBR (Constant Bit Rate). Le tableau 6.1 résume les paramètres de ces simulations.

### 6.3.2 Comportement dynamique de l'auto configuration

La figure 6.2 montre que notre algorithme parvient à trouver la bonne valeur du paramètre macMinBE à utiliser et cela quelles que soit le nombre de nœuds présents autour du coordinateur. Notre algorithme commencera avec la valeur par défaut du paramètre macMinBE=3 comme cela est fait dans le standard IEEE 802.15.4, et l'augmentera à chaque supertrame jusqu'à la découverte du macMinBE maximisant le débit. Les conditions de trafic ne changeant pas au cours du temps à cause du trafic saturé, la valeur de macMinBE oscille toujours autour de la valeur optimale.

**Algorithm 1:** Updating the BE value

```
1  /* Initialization                                                    */
2  if isCoordinator then
3      myMacMinBE ←3 ;
4      direction ←+1 ;
5      oldTroughput ←0;

6  /* Children:  they read the macMinBE directly in beacons from their
       parent                                                            */
7  if ! isCoordinator then
8      receiveBeacon ;
9      macMinBE ←readValueFromBeacon ;
10     return ;

11 /* the coordinator must continuously update its BE                   */
12 if isCoordinator then
13     /* the throughput has not changed for a too long time (k
          superframedurations)                                           */
14     changeBE = BEUnchangedDuring (k * SD);
15     /* the throughput was changing:  the coordinator should continue
          changing its BE                                                */
16     changeBE |= (newThroughput < oldTroughput- Δ) ;
17     changeBE |= (newThroughput > oldTroughput + Δ) ;

18     if changeBE then
19         /* the throughput has decreased:  let change the direction   */
20         if newThroughput < oldTroughput- Δ then
21             direction ←( - direction);
22         myMacMinBE ←myMacMinBE + direction ;
23     /* the coordinator sends the BE in beacons                        */
24     /* NB: if the throughput was stable, it remains unchanged         */
25     sendBeacon(myMacMinBE);
26     oldTroughput ←newThroughput ;
```

| Probabilité | Description |
|---|---|
| Air de la simulation | 100m x 100m |
| Durée de simulation | 100000 s |
| Nombre de nœuds | 100 en multisaut |
| $k$ (Maximum des supertrame avec un BE fixe) | 3 |
| **(Trafic saturé)** | toujours un paquet à envoyer |
| **(Trafic non saturé)** | CBR |
| **BO/SO** | 13/6 |
| $\Delta$ (Seuil de débit) | $\frac{1}{SD}$ |

**Table 6.1:** Paramètres de la simulation

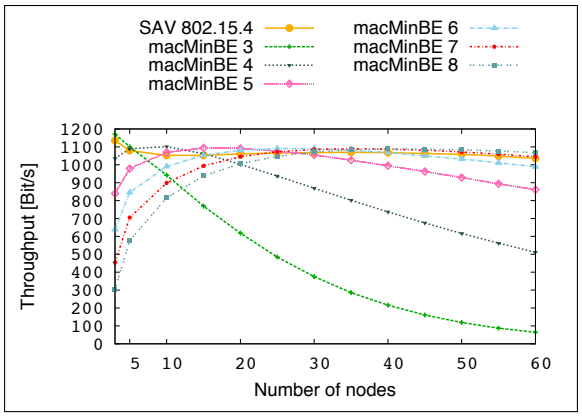


**Figure 6.2:** Débit dans un réseau en étoile en régime de trafic saturé

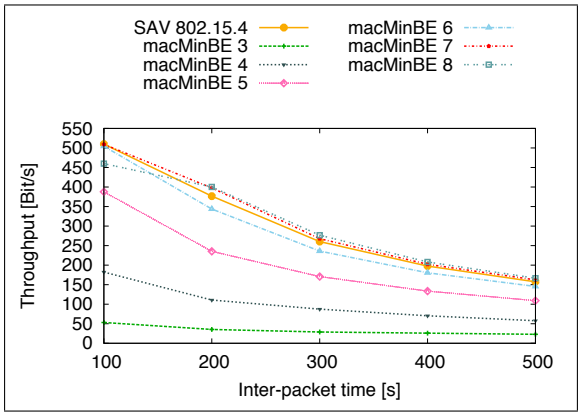


**Figure 6.3:** Debit dans un contexte de réseau multisaut avec un trafic CBR

Afin d'observer le comportement de notre proposition dans un contexte différent de celui étudié ci-dessus (trafic saturé), nous avons simulé un réseau en multisaut avec un trafic CBR (Constant Bit Rate) où chaque nœud générera un paquet par intervalle de temps. Les figures 6.3 et 6.4 montrent les résultats obtenus en terme respectivement de débit et de PDR.

L'analyse des deux figures 6.3 et 6.4 nous permet de conclure que notre algorithme arrive à adapter la valeur du paramètre macMinBE afin d'avoir des performances quasi optimales, et cela même dans un réseau multisaut et en présence d'un trafic hétérogène. L'hétérogénéité du trafic provenant du fait que dans un réseau cluster-tree, selon le sous arbre qu'il devra gérer, un coordinateur aura des intensités de trafic différentes des autres coordinateurs.

Le fait que notre algorithme adapte la valeur de macMinBE en se basant uniquement sur le débit observé peut être un handicap. Il se peut qu'il y ait une augmentation subite du trafic généré par les nœuds, ceci menant à des collisions sans pour autant réduire le débit, incitant le coordinateur à distribuer une valeur non optimale pour le paramètre macMinBE.

Afin d'éviter ce genre de situation notre algorithme a pour tâche d'enclencher une nouvelle exploration même si le débit est stable. De ce fait les résultats pouvant être obtenus tendent à être optimaux: en effet il est à noter que pour le PDR comme pour le débit les performances obtenues par notre algorithme sont en-dessous de l'optimal de 5%.

### 6.3.3 Limites de la proposition

Les résultats de simulations présentés dans ce chapitre prouvent que le mécanisme d'accès au canal du standard IEEE 802.15.4 n'est pas efficace et ne peut gérer les réseaux

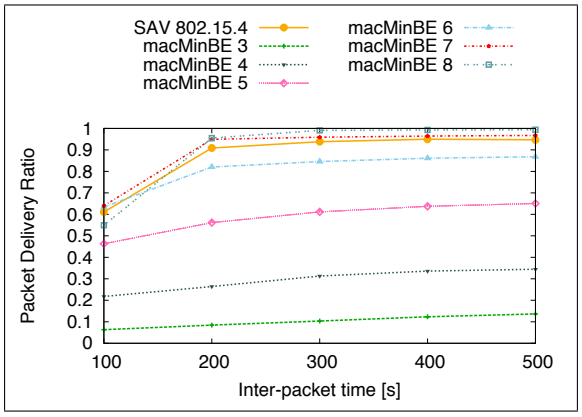


**Figure 6.4:** PDR dans un contexte de réseau multisaut avec un trafic CBR

présentant une forte densité en terme de nœuds ou de trafic.

L'algorithme proposé a pour but de montrer la faisabilité et l'intérêt d'adapter la valeur du paramètre macMinBE conditionnant la taille de la fenêtre de contention dans le mécanisme CSMA/CA proposé dans le standard IEEE 802.15.4. Cet algorithme même s'il est simple, arrive néanmoins à des taux de livraison (Packet Delivery Ratio) et débit inatteignables par le standard IEEE 802.15.4 avec les paramètres par défaut comme le montrent les figures 6.3 et 6.4 .

En effectuant plus d'expérimentations nous avons remarqué que la taille des supertrames utilisées (BO et SO) avaient une grande influence sur les résultats obtenus. Si le trafic est de type poissonnien, quelles que soit la taille des supertrames, les résultats seront identiques à ceux présentés précédemment, mais si la fréquence de génération des paquets fait qu'il y a des supertrames vides (si le temps de génération de paquets dépasse le temps où le coordinateur fixe la valeur de macMinBE avant de commencer une autre exploration) l'algorithme ne donne pas d'aussi bonne performances. L'utilisation du débit peut être une bonne métrique afin d'adapter la valeur de macMinBE, mais ce paramètre seul ne permet par d'avoir un protocole adaptatif robuste dans toutes les conditions de trafic.

Nous pouvons également souligner que non seulement l'algorithme présenté est trop dépendant de la périodicité du trafic, mais qu'en plus, il reste à améliorer le temps de convergence de l'algorithme qui est assez lent à s'adapter aux conditions de trafic, étant donné que cette adaptation se fait de manière séquentielle en incrémentant ou décrémentant la valeur de macMinBE après chaque supertrame.

## 6.4 Deuxième proposition: SABE (Self Adapting BE)

Comment s'adapter plus rapidement ou plus exactement comment un coordinateur peut-il trouver la valeur optimale de macMinBE sans pour autant tester les valeurs intermédiaires, comme dans le premier algorithme, représente l'objectif que nous voulions atteindre. La solution la plus naïve serait un protocole adaptatif ayant une capacité d'apprentissage: ce protocole pourrait apprendre les cycles d'apparition de trafic et adapter les valeurs de macMinBE en conséquence. Le désavantage de ce genre de solution vient du fait qu'elle n'est efficace que dans un contexte de trafic périodique et prédictible. Le trafic périodique n'étant pas l'unique trafic que l'on peut rencontrer dans un réseau de capteurs, cette solution n'est pas envisageable.

Une étude poussée sur le mécanisme d'accès utilisé dans WIFI a été effectuée par Heusse *et al.* [95, 96] . Ce travail a abouti sur l'élaboration du protocole appelé Idle Sense dont le rôle est d'adapter la taille de la fenêtre de contention dans la méthode d'accès utilisée dans le mode DCF du standard IEEE 802.11. Afin d'ajuster la taille de la fenêtre de contention ce protocole se base sur le nombre de slots libres entre deux accès au canal.

Dans le protocole Idle Sense les nœuds n'ayant aucune contrainte énergétique peuvent scruter le canal en continu ce qui leur permet de connaître l'état du canal en temps réel et de savoir s'il y a des communications ou non. Ceci permet ainsi de connaître le nombre de slots libres observés entre deux communications successives, et ainsi d'adapter la valeur de la fenêtre de contention qu'ils utiliseront pour leur futures communications minimisant les temps d'attente. Les auteurs se sont basés sur le calcul de $n_i$ (nombre de slots libres moyen) dans le but de s'abstraire de la connaissance préalable du nombre de nœuds sur le canal.

Ce protocole ayant prouvé son efficacité pour ce qui est de l'adaptation de la taille de la fenêtre de contention dans toutes les conditions en terme de densité de nœuds et de trafic nous nous somme fixé comme objectif de proposer un algorithme utilisant le même principe afin de déduire la taille des fenêtres de contention se basant sur l'estimation nombre de slots libres moyen pour le calcul de macMinBE.

### 6.4.1 Analyse

Afin de comprendre plus finement le mécanisme d'accès au canal proposé dans IEEE 802.15.4, nous avons cherché à modéliser mathématiquement le comportement de l'algorithme CSMA/CA proposé dans IEEE 802.15.4.

Pour ce faire, nous avons suivi la même démarche que les auteurs de Idle Sense qui se sont focalisés sur la méthode d'accès de IEEE 802.11.

Ces probabilités modélisent les interactions des nœuds et du canal lors des tentatives d'accès au canal des différents nœuds.

Ce calcul se décompose en quatre probabilités et montre leur interdépendance, le tableau 6.2 résume ces quatre probabilités.

Pour le calcul des équations : 6.1, , 6.2, 6.3 et 6.4 nous avons pris les hypothèses suivantes: le trafic est de type saturé (les nœuds ayant toujours un paquet à émettre), la topologie du réseau est une étoile sans stations cachées, avec un coordinateur central jouant le rôle de PAN coordinateur, et pour finir, afin d'éviter une trop grande complexité du modèle, le doublement de Backoff est interdit (fenêtre de contention fixe).

| Probabilité | Description |
|---|---|
| $P_e$ | Probabilité de tirage d'un slot donnée pour l'émission conditionnée par la taille de la fenêtre de contention |
| $P_c$ | Probabilité que plusieurs nœuds émettent dans le même slot menant à des collisions |
| $P_t$ | Probabilité qu'un nœud transmette durant un slot libre |
| $P_i$ | Probabilité qu'aucune station ne transmette durant un slot donnée |

**Table 6.2:** Récapitulatif des probabilités calculées

Le temps de backoff que l'on notera $B$ dans ce qui suit est choisi aléatoirement dans l'intervalle $[0, 2^{BE} - 1]$ où BE représente le Backoff Exponent. Ce temps de backoff rentre dans le calcul de Pe eq.6.1 qui est la probabilité qu'un nœud essaye de transmettre dans un slot choisi aléatoirement dans un intervalle conditionné par le choix de la valeur du BE.

$$P_e = \frac{2}{B+1} \tag{6.1}$$

La probabilité $P_t$ eq.6.2 représente la probabilité qu'une station arrive à transmettre dans un slot donné alors que les autres stations n'utilisent pas ce même slot, c'est à dire qu'une station transmette un paquet sans que cela se traduise par une collision, cette probabilité est conditionnée par la probabilité $P_e$.

$$P_t = N \times P_e \times (1 - P_e)^{N-1} \tag{6.2}$$

La probabilité $P_i$ eq.6.3 définit la probabilité qu'un slot donné soit libre c'est à dire qu'aucune station ne tente d'émettre pendant ce slot (aucun accès au canal).

$$P_i = (1 - P_e)^N \tag{6.3}$$

La probabilité de collisions eq.6.4 qui représente la probabilité que plusieurs stations choisissent le même slot pour effectuer leur transmission devient ainsi:

$$P_c = 1 - P_t - P_i \tag{6.4}$$

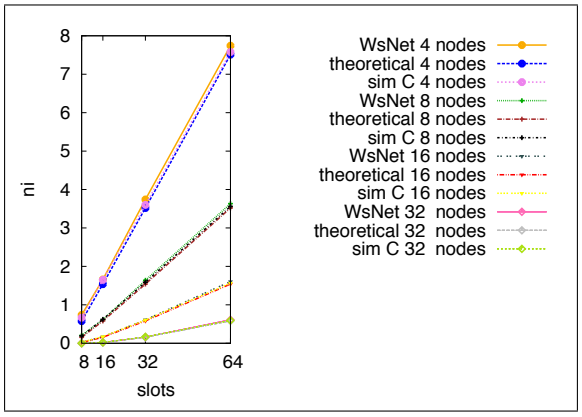


**Figure 6.5:** Nombre de slots libres en fonction de la taille de la fenêtre de contention

### 6.4.2 Validation par simulations

Afin de prouver que la modélisation mathématique que nous utilisons, la même que celle présentée dans [95] traitant de DCF dans IEEE 802.11, est applicable pour le mécanisme d'accès du standard IEEE 802.15.4, nous avons implémenté un simulateur en langage C, simulant le comportement du CSMA/CA du standard IEEE 802.15.4, nous avons aussi effectué des simulations sur le simulateur WSNET, afin de confirmer que les résultats obtenus théoriquement et ceux obtenus à l'aide des deux simulateurs concordaient.

Les simulations effectuées portaient sur un réseau en étoile où l'on a augmenté le nombre de nœuds de 4 à 32 autour du coordinateur, nous avons testé ceci pour différentes tailles de la fenêtre de contention ($2^{BE} - 1$, $BE \in [3, 4, 5, 6]$).

La figure 6.5 représente la comparaison entre les résultats obtenus en simulation (à partir du simulateur WSNET et du simulateur en C) et les résultats théoriques (équations présentées ci-dessus) en terme de nombre moyen de slots libres entre deux transmissions.

Nous pouvons constater que les trois courbes concordent parfaitement et cela indépendamment de la taille de la fenêtre de contention et quelles que soit le nombre de nœuds autour du coordinateur.

La figure 6.6 montre le taux de collisions par rapport à la taille de la fenêtre de contention. Cette figure a été obtenue en conservant les mêmes hypothèses que précédemment et montre que les résultats obtenus avec le simulateur WSNET, le simulateur en C et la modélisation mathématique respectivement les croix, ronds et les lignes continues sur la courbe donnent exactement les mêmes résultats en terme de taux de collisions.

Ces résultats de simulation valident notre analyse théorique. Le calcul des probabilités correspondent parfaitement avec la modélisation du CSMA/CA dans 802.11 déjà effectuée dans l'article Idle Sense.

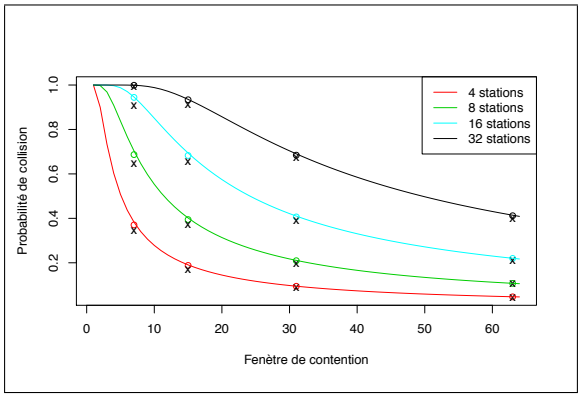


**Figure 6.6:** Probabilité de collision en fonction de la taille de la fenêtre de contention

### 6.4.3 SABE (Self Adapting BE)

Comme nous l'avons vu précédemment, le fait de fixer la valeur de macMinBE à 8 (maximum autorisé dans le standard IEEE 802.15.4) permet de maximiser le PDR. Néanmoins maximiser le PDR ne signifie pas pour autant que le canal soit utilisé de manière optimale.

Au contraire utiliser de trop grands temps de Backoff alors que le trafic ne l'exige pas revient à gaspiller de la bande passante, car une attente excessive de la part des nœuds avant chaque transmission laisse un nombre important de slots inutilisés (sans accès au canal).

Le but que nous cherchons à atteindre est de maximiser le débit. Ceci est atteint en trouvant la valeur de BE optimale minimisant les collisions, le nombre de slots libres entre les transmissions et maximisant les temps de transmissions réussies.

Le débit peut être exprimé en fonction de $P_e$ eq.6.5 où $T_t$ représente le temps moyen des transmissions, $T_c$ le temps moyen des collisions, $T_{slot}$ le temps d'un slot (Backoff slot) et $S_d$ la quantité d'informations envoyées sur le canal.

$$X(P_e) = \frac{P_t \times s_d}{P_t \times T_t + P_c \times T_c + P_i \times T_{slot}} \tag{6.5}$$

Maximiser le débit revient alors à minimiser la fonction de coût représentée par l'équation 6.6:

$$cost(P_e) = \frac{\frac{T_c}{T_{slot}} \times P_c + P_i}{P_t} \tag{6.6}$$

Le calcul des probabilités montrées ci-dessus est le même que celui obtenu par Heusse *et al.* dans Idle Sense, ce qui intuitivement n'est pas facile à déduire, vues les différences de fonctionnement entre le CSMA/CA slotted utilisé dans IEEE 802.15.4 et celui utilisé dans DCF en 802.11 fig.6.1 ( cf.5.3 ).

En effet la probabilité d'essai dans un slot donné $P_e$ dépend de la taille de la fenêtre de contention ($2^{BE}$ dans IEEE 802.15.4 et $CW$ dans IEEE 802.11). Les autres probabilités ($P_c$, $P_i$ et $P_t$) sont dépendantes de cette dernière.

La probabilité $P_e$ étant indépendante de ce qui se passe dans les slots voisins, elle ne change pas si une partie des slots ne sont pas considérés, soit les slots occupés par une transmission. Les slots étant indépendants l'arrêt du backoff durant une transmission comme dans IEEE 802.11 ou en aveugle comme dans IEEE 802.15.4 revient au même.

Nous avons repris le même procédé afin de calculer la valeur optimale du nombre de slots à attendre avant chaque transmission $N_i$ *opt* afin de maximiser le débit dans IEEE 802.15.4, $N_i$ *opt* sera alors égal à 2.66 slots.

Dans IEEE 802.11 les stations étant allumées tout le temps, les nœuds du réseaux auront comme objectif de régler leurs valeurs de fenêtre de contention (augmenter ou diminuer la valeur de cette dernière) selon les conditions observées sur le canal à chaque transmission.

Dans le standard IEEE 802.15.4 les nœuds doivent éteindre leur radio le plus possible. Ainsi un nœud ne pourra savoir l'état du canal qu'à la fin de son temps backoff et non pas pendant toute la durée de ce dernier comme dans 802.11.

La radio des nœuds étant éteinte durant toute la période d'attente aléatoire, ces derniers n'ont qu'une information partielle et ainsi ne peuvent prendre aucune décision à la volée concernant la réduction ou l'augmentation de la fenêtre de contention comme cela est fait dans Idle Sense. De ce fait nous ne pouvons utiliser le protocole Idle Sense tel qu'il est proposé pour la méthode d'accès 802.11.

Au lieu de changer le fonctionnement du CSMA/CA du standard IEEE 802.15.4 en permettant aux nœuds de scruter le canal en continu, ce qui aura comme conséquence une augmentation de la consommation énergétique, nous proposons de reporter cet effort sur le coordinateur.

En effet le coordinateur ne subit aucun surcoût énergétique puisqu'il doit écouter le canal durant toute la durée de la supertrame qu'il maintient pour réceptionner les trames de ses fils. Ceci lui permettra d'estimer le nombre de slots libres durant la supertrame ainsi que le nombre d'accès au canal (transmissions, collisions) permettant le calcul du nombre de slots libres moyen durant la supertrame $n_i$ ( 6.7).

Le nombre moyen de slots libres dépendant de la probabilité $P_i$ qu'un slot soit libre, nous pouvons l'exprimer comme suit (cf. 6.7):

$$n_i = \frac{P_i}{1 - P_i} \tag{6.7}$$

A partir de l'équation 6.7, nous pouvons calculer la valeur de $P_i$ observée par le coordinateur (6.8 ) qui nous permet d'estimer le nombre de nœuds $N$ sur le réseau à la fin de la supertrame.

$$P_i = \frac{n_i}{n_i + 1} \tag{6.8}$$

Ayant calculé le nombre de nœuds sur le canal pendant une supertrame, le coordinateur utilise la valeur trouvée afin de calculer la valeur optimale de la taille de la fenêtre de contention en utilisant la valeur optimale du nombre de slots que les nœuds doivent attendre avant chaque transmission $N_i$ *opt*.

$$P_i opt = \frac{n_{iopt}}{n_{iopt} + 1} = \frac{2,66}{3,66} \tag{6.9}$$

Nous voulons maintenir le nombre de slots libres entre deux transmissions le plus proche possible de la valeur de $N_i$ *opt*. Pour cela le coordinateur calcule la valeur de $CW_{opt}$ de la façon suivante à partir de $P_e$, $P_{iopt}$ et l'estimation de $N$ comme suit:

$$P_i = (1 - P_e)^N \Rightarrow (1 - P_e)^N = P_{iopt} \Rightarrow CW_{opt} = \frac{2}{1 - exp(\frac{P_{iopt}}{N})} \tag{6.10}$$

En résumé la méthode que nous proposons afin d'adapter la taille de la fenêtre de contention repose sur le fait que le coordinateur, grâce au nombre moyen de slots libres constatés durant la supertrame, pourra calculer le nombre de nœuds ayant participé a cette dernière et ainsi en déduire la valeur optimale de macMinBE i.e maximisant le débit et le PDR.

### 6.4.4 Adaptation pour les rafales (bursts) de trafic

Implémenter de cette façon l'algorithme proposé ne pourra gérer efficacement des trafics de type burst périodique (en dent de scie) en effet le trafic par supertrames n'est pas constant et certaines supertrames peuvent être vides.

Le coordinateur estime la valeur qu'il distribuera dans le prochain Beacon à partir des conditions observées durant la supertrame en cours: si la supertrame est vide le coordinateur décide de réduire la valeur de macMinBE à son minimum (macMinBE=3), ce qui provoquera un grand taux de collisions si les nœuds durant la supertrame d'après ont beaucoup de trames à envoyer. Dans ces conditions le coordinateur ne sera pas en mesure de corriger son estimation erronée sachant que l'envoi de la majorité des paquets sera abandonné par les nœuds trouvant le canal toujours occupé.

Afin de gérer les cas pathologiques comme présenté ci-dessus, un coordinateur en présence d'une supertrame vide en terme de trafic enverra à ses fils dans son Beacon la valeur maximum de macMinBE (macMinBE=8). Ceci permettra en cas de reprise forte du trafic d'éviter les collisions ainsi que l'abandon d'envoi des trames du fait de l'occupation du canal.

Nous considérons les supertrames où il y a une reprise des communications comme des supertrames tampon pour les coordinateurs, les grandes valeurs de macMinBE retardant

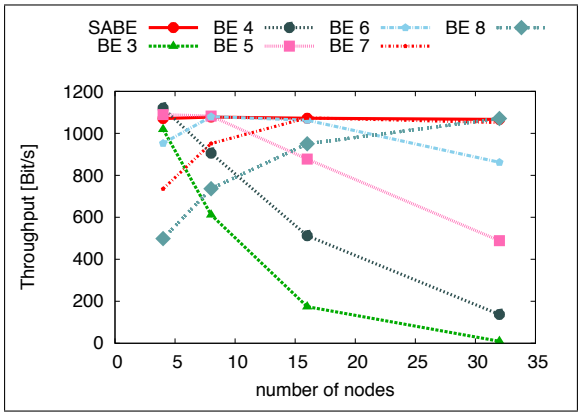


**Figure 6.7:** Débit en réception dans un contexte de trafic saturé

l'envoi des trames et permettant au coordinateur de s'adapter à ce brusque changement avec le calcul de la valeur optimale de macMinBE pour la supertrame suivante.

### 6.4.5 Adaptation des CCAs

Si le taux de génération de paquets par nœud est bas (une trame par nœud par exemple), mais que les nœuds tentent tous d'envoyer leurs trames durant la même supertrame, les premiers nœuds accédant au canal pourront transmettre leurs trames tandis que les autres jetteront leurs trames pour cause d'occupation du canal réduisant ainsi les performances. Le coordinateur ne pouvant savoir qu'un phénomène du genre est en train de se passer calculera une mauvaise valeur de macMinBE, vu que les trames sont jetées sans être envoyées. Dans ce cas les nœuds vont faire au minimum 4 CCA avant de jeter la trame, ce qui revient à une dépense énergétique inutile.

Nous proposons de prendre en compte non pas la valeur de macMaxCSMABackoffs comme c'est fait dans le standard IEEE 802.15.4 qui la fixe à 4, mais de compter le nombre de CCA qu'un nœud aura le droit de faire. En effet la norme permet de faire un maximum de 8 CCA (macMaxCSMABackoffs = 4). Dans notre algorithme le coordinateur détectant une intensité de trafic faible durant une supertrame (temps d'émission des trames + le temps moyen des fenêtres de contention < la durée d'activité d'une supertrame ) permet à ses fils de faire jusqu'à 8 CCA pour l'envoi d'une trame dans la prochaine supertrame.

Cette sécurité étant surtout utile dans le contexte où les nœuds ont peu de paquets à émettre, dans un contexte de trafic beaucoup plus soutenu ces derniers n'auront pas besoin de faire autant de CCA, sachant qu'il y aura assez d'accès au canal permettant au coordinateur de trouver la valeur optimale de macMinBE.

### 6.4.6 Longest remaining time first (LRTF) vs Short remaining time first (SRTF)

Dans des topologies de type cluster-tree, les différents coordinateurs n'auront pas la même quantité de trames à envoyer vers le puits du sous arbre qu'ils doivent gérer. Nous avons pensé à rajouter un mécanisme permettant un partage équitable de la bande passante entre les différents nœuds formant un cluster. Afin d'atteindre cet objectif nous avons tenté deux approches:

- La première a comme but de donner le plus de priorité aux nœuds ayant la plus grande charge (Longest remaining time first): le principe est de permettre à ces derniers de réduire la taille de leur fenêtre de contention en fonction du nombre de trames présentes dans leurs buffers;
- La deuxième propose de faire le contraire, c'est à dire donner plus de priorité aux nœuds ayant le moins de trafic à écouler (Short remaining time first) afin qu'ils finissent leurs transmissions rapidement au début de la supertrame laissant le reste du temps aux autres nœuds.

De prime abord cette idée nous a paru intéressante, mais en faisant des simulations nous nous sommes aperçus, qu'implémenter de la façon présentée ci-dessus, nous nous retrouvons avec les mêmes performances offertes par la version native de la méthode d'accès au canal du standard IEEE 802.15.4.

En effet, la valeur de macMinBe envoyée par le coordinateur dans notre protocole est une valeur optimale ayant comme objectif de répartir au maximum les transmissions tout en minimisant les délais. Le fait de manipuler cette valeur en fonction de la charge des buffers implique la concentration des trames des nœuds les moins chargés sur une portion de la supertrame et les nœuds ayant une plus grande charge sur une autre portion, ce qui est contradictoire avec le but recherché avec le protocole SABE.

### 6.4.7 Evaluation de la proposition

Afin dévaluer notre proposition nous avons d'abord commencé par simuler un réseau en étoile où l'on a augmenté le nombre de nœuds de 4 à 32 autour du coordinateur. Nous avons dans tout ce qui suit comparé notre approche au CSMA/CA du standard IEEE 802.15.4 avec les différentes valeurs de macMinBE allant de 3 à 8.

La figure 6.7 représente le débit obtenu dans ce réseau en étoile avec un trafic saturé où les nœuds ont toujours un paquet à envoyer. Cette figure montre que notre proposition appelée SABE (Self Adaptation BE) offre quelles que soit le nombre de nœuds autour du coordinateur, un débit optimal. En effet quelles que soit le nombre de nœuds autour du coordinateur nous pouvons voir que SABE offre toujours le même débit, le coordinateur dans notre algorithme étant toujours à même de trouver la valeur de macMinBE optimisant le débit. Si nous comparons notre approche par rapport au CSMA/CA avec

les valeurs du macMinBE=3 et macMaxBE=5 imposées comme valeurs par défaut dans le standard IEEE 802.15.4, on voit que l'adaptation de la valeur de macMinBE permet un gain d'à peu près 500 à 1000 bit/s, le CSMA/CA du standard n'offrant de bonnes performances que dans le cas où le nombre de nœuds est inférieur ou égal à 5.

Nous avons souhaité tester notre approche dans d'autres conditions de trafic. Nous avons conservé la même topologie en étoile pour laquelle nous avons augmenté le nombre de noeuds et avons observé l'impact de trois différents types de trafic comparé avec le standard IEEE 802.15.4:

- Un trafic périodique de type CBR uniforme où à chaque top d'horloge (en abscisse dans les courbes) les nœuds génèrent un paquet,
- Un trafic périodique non uniforme où à chaque top d'horloge les nœuds génèrent un nombre aléatoire de paquets. Ce cas est plus réaliste puisqu'il représente dans un certain sens le comportement des coordinateurs dans une topologie cluster-tree i.e les coordinateurs auront un trafic proportionnel au nombre de nœuds dans la sous branche de l'arbre qu'ils gèrent,
- Des trafic poissonnien de différentes intensités.

Les figures 6.8, 6.9 et 6.10 représentent le débit en réception obtenu au niveau du coordinateur dans un contexte de trafic non uniforme et où il y a des supertrames entièrement vides. Sur ces figures nous constatons que l'algorithme que nous proposons (SABE) maximise le débit même quand le trafic est sous forme de burst et ceci avec n'importe quel nombre de nœuds autour du coordinateur. Ce dernier arrivant toujours à adapter macMinBE aux conditions de trafic afin d'offrir des performances optimales. Les résultats obtenus en simulant les mêmes topologies avec un trafic uniforme (CBR) donnent exactement les mêmes résultats que pour le trafic non uniforme. Pour ces raisons nous ne les présenterons pas.

Ce qui est intéressant à noter à travers la figure 6.11 qui représente le délai obtenu dans les différentes topologies en étoiles avec le trafic non uniforme, est que notre algorithme offre en plus d'un débit optimal un délai plus bas comparé au standard IEEE 802.15.4 avec la valeur de macMinBE fixée à 8. Nous nous comparons à cette valeur parce qu'elle permet d'atteindre les meilleures performances en terme de débit pour le standard IEEE 802.15.4. Nous pouvons voir que le gain en délai peut atteindre la seconde comparé à macMinBE = 8 et ceci uniquement sur un saut. Ce résultat prouve que notre approche n'utilise de grandes valeurs de macMinBE (8 par exemple) que lorsque les conditions l'obligent, et c'est ce qui nous permet d'avoir de meilleurs délais.

La figure 6.12 représente le débit en fonction du taux de génération de paquets dans un réseau en étoile composé de 20 nœuds autour du coordinateur. Le trafic généré par les nœuds est un trafic de type poissonien. Nous pouvons voir sur cette figure que notre proposition offre toujours un débit optimal que ce soit dans des conditions de bas trafic

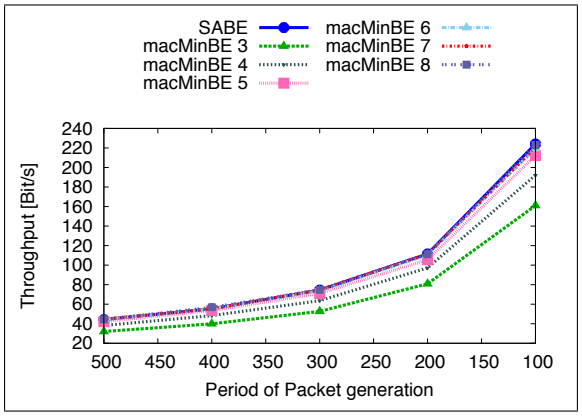


**Figure 6.8:** Débit dans un cluster de 4 nœuds autour du coordinateur un trafic non uniforme

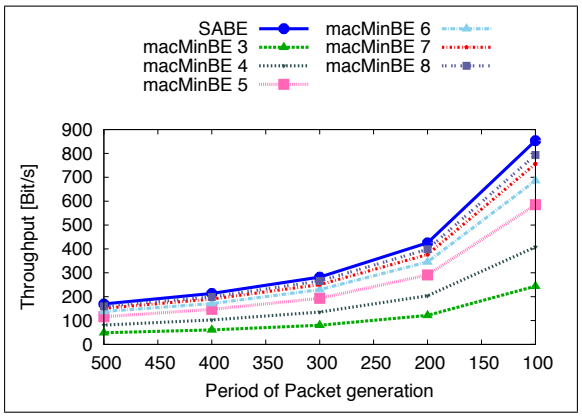


**Figure 6.9:** Débit dans un cluster de 16 nœuds autour du coordinateur un trafic non uniforme

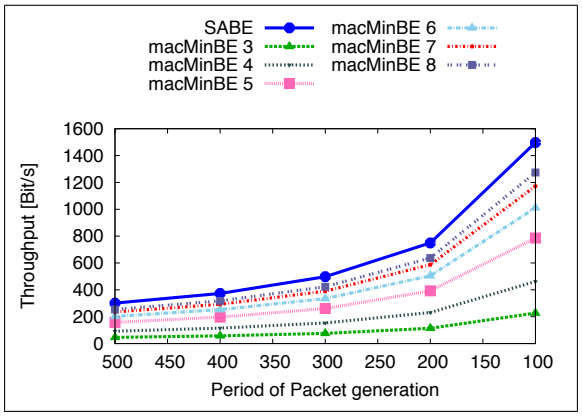


**Figure 6.10:** Débit dans un cluster de 32 nœuds autour du coordinateur un trafic non uniforme

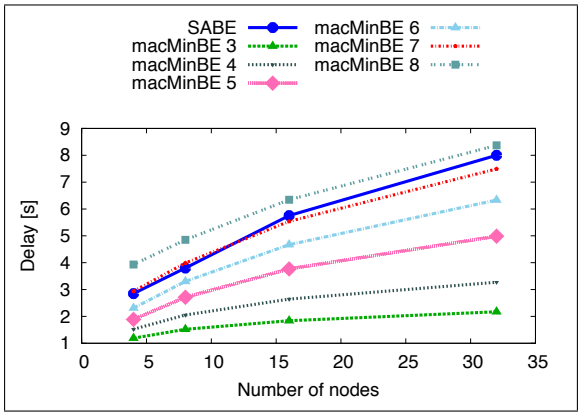


**Figure 6.11:** Délai avec un trafic non uniforme un trafic non uniforme

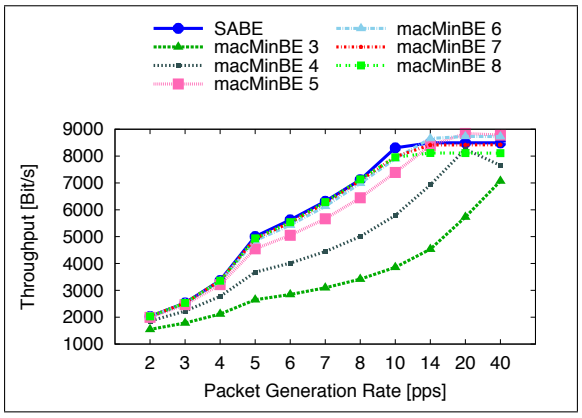


**Figure 6.12:** Délai dans un cluster de 20 nœuds autour du coordinateur un trafic poissonien

par exemple 2 paquets par seconde injectés dans le réseau, et dans des conditions de trafic beaucoup plus soutenu (20 à 40 trames/seconde), avec ce trafic nous remarquons qu'en fixant les valeurs de macMinBE à 5,6,7 où 8 donne le même trafic que notre proposition, ceci est dû au doublement de la taille de la fenêtre de contention. Comme pour le trafic uniforme et non uniforme, SABE offre d'aussi bonnes performances en terme de délais comme le montre la figure 6.13.

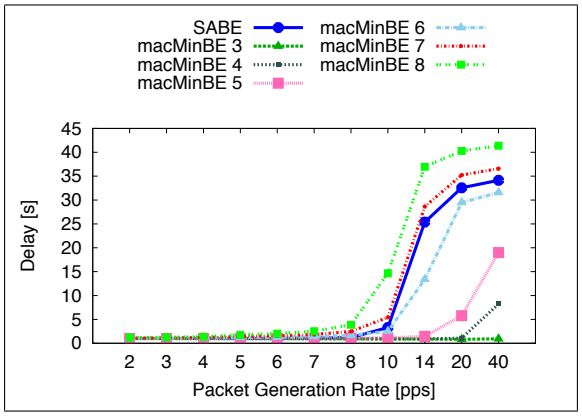


**Figure 6.13:** Délai dans un cluster de 20 nœuds autour du coordinateur un trafic poissonien

## 6.5 Conclusion

La méthode d'accès au canal a un rôle primordial dans les réseaux, elle doit pouvoir gérer les communications des différents nœuds d'un réseau quelque soit la topologie du réseau, les conditions en terme de densité et de l'intensité du trafic généré.

Dans ce chapitre nous avons analysé en profondeur la méthode d'accès au canal proposée dans le standard IEEE 802.15.4 dans les différentes conditions exposées ci-dessus. Nous avons pu constater que cette méthode d'accès souffrait de problèmes de conception, ne garantissant de bonnes performances que si elle est confrontée à des conditions particulières.

Le CSMA/CA proposé dans le standard IEEE 802.15.4 à cause de la contrainte énergétique est doté d'un mécanisme permettant aux nœuds de réduire la consommation énergétique utilisée pour l'envoi d'une trame donnée: les nœuds doivent abandonner l'envoi de la trame si le canal est trouvé occupé au bout de quatre essais. Les trames étant présentes dans les buffers dans la majorité des cas au début de la supertrame (trames reçues de la part des fils, ou générées pendant que la radio est éteinte) les nœuds auront tendance à tous vouloir communiquer au début de cette dernière, densifiant virtuellement le trafic. Dans ces conditions un nœud voulant envoyer sa trame aura de fortes chances de trouver le canal occupé, les valeurs conditionnant la taille de la fenêtre de contention étant trop petites. Cette situation mène à un fort taux de collision et d'abandon d'envoi des trames, dégradant les performances attendues.

Les deux propositions faites dans ce chapitre ont toutes deux vocation à permettre à la méthode d'accès au canal de gérer efficacement les communications sans connaissance préalable de la quantité de trafic et de la densité des nœuds. Ceci est obtenu en adaptant les tailles des fenêtres de contention en fonction des conditions observées localement dans le but de maximiser l'utilisation de la bande passante tout en garantissant un débit

optimal.

Les mécanismes proposés dans ce chapitre ne nécessitent pas un changement dans le comportement des nœuds fils (comme par exemple la nécessité de rester éveillés durant toute la supertrame), la décision d'adapter les valeurs de macMinBe revenant aux coordinateurs de cluster, qui ont la capacité d'observer les conditions sur le canal durant toute la durée de la supertrame (informations locales) et peuvent définir la valeur optimale de la fenêtre de contention afin de la distribuer à ses fils dans le Beacon.

Il serait intéressant de continuer le travail sur la méthode d'accès de IEEE 802.15.4. Cette méthode d'accès est équitable à long terme puisque le tirage du backoff est réarmé au début de chaque supertrame, néanmoins à court terme cette méthode d'accès n'est pas équitable puisque le nœud ayant accès au canal (en cours d'émission) aura plus de chance d'accéder une nouvelle fois au canal par rapport aux autres stations qui ont dû doubler leur fenêtre de contention pour cause d'occupation de canal.

# Partie III

# Conclusion et perpectives

# CHAPITRE 7

# CONCLUSION ET PERSPECTIVES

## 7.1 Conclusions

Dans ce travail de thèse nous nous sommes concentrés sur une des contributions pour les réseaux de capteurs que nous jugeons la plus pertinente le standard IEEE 802.15.4. Le but de notre travail a été d'étudier en profondeur les mécanismes proposés dans la couche MAC de ce standard afin d'en identifier les principales problématiques. Nous nous sommes rendus compte que le standard IEEE 802.15.4 souffrait d'un certain nombre d'anomalies de conception.

Dans le chapitre 2 nous avons présenté les principaux protocoles traitant de la couche MAC des réseaux de capteurs ayant été proposés dans la littérature. Dans ce chapitre nous avons essayé de faire un tour d'horizon des principales familles de couches MAC en présentant les points forts de ces propositions ainsi que leurs faiblesses.

Dans le chapitre 3 nous avons présenté le standard IEEE 802.15.4, son fonctionnement et les principales problématiques liées à la couche MAC de ce dernier. Le standard propose entre autres le mode Beacon permettant de faire de l'économie d'énergie en permettant aux nœuds de ne garder la radio allumée que durant le temps nécessaire à leurs communications. Ce mode introduit néanmoins son lot de problèmes. En effet nous avons pu constater dans ce chapitre que la couche MAC du standard IEEE 802.15.4 ne propose aucun mécanisme pour se prémunir des interférences entre les différents coordinateurs appartenant à un même réseau, problème très contraignant sachant que tout le fonctionnement du réseau repose sur la bonne réception des trames Beacons qu'envoient les coordinateurs. Nous avons également montré dans ce chapitre que le standard laisse ouverte la construction de topologies et ne résout pas les problèmes liés à la méthode d'accès.

Dans le chapitre 4 nous proposons une adaptation du standard IEEE 802.15.4: le

protocole MCCT qui est un protocole de construction de topologie dans IEEE 802.15.4 qui a comme principe l'utilisation de l'ensemble des canaux fournis par le standard afin de construire une topologie cluster-tree dans le but de minimiser les interférences inter et intra cluster. L'évaluation de notre proposition montre l'intérêt de l'utilisation du multi-canal, cette technique permettant d'ordonnancer les supertrames des coordinateurs sur différents canaux afin de permettre au standard de supporter une plus forte densité en terme de nœuds et cela indépendamment du duty cycle comme présenté dans d'autres propositions.

Dans le chapitre 5 nous nous sommes proposés d'étudier la méthode d'accès au canal du CSMA/CA Slotted proposée dans le mode Beacon du standard IEEE 802.15.4. Nous avons pu constater grâce aux simulations effectuées que cette méthode d'accès est fortement influencée par la valeurs des paramètres conditionnant la taille des supertrames, de la fenêtre de contention et le nombre d'essais que fait un nœud afin d'envoyer une trame. Nous avons conclu à partir des résultats présentés dans ce chapitre que face aux différentes applications que doivent gérer les réseaux de capteurs il faut avoir un mécanisme d'auto adaptation gérant les accès au canal.

Au vu des performances qu'offre la méthode d'accès du standard IEEE 802.15.4 nous proposons dans le chapitre 6 deux algorithmes dont le but est de permettre l'auto-adaptation de la taille de la fenêtre de contention dans la méthode d'accès au canal afin d'offrir des performances optimales en terme d'utilisation de la bande passante. Ces deux méthodes ont comme particularité d'être orientées coordinateur, c'est à dire que c'est le coordinateur selon les conditions observées dans son cluster qui adapte la valeur du paramètre conditionnant la taille de la fenêtre de contention en début de supertrame. Ceci évite d'avoir une mauvaise équité dans le cas où les nœuds sont libres de choisir indépendamment. Les résultats que nous avons proposé dans ce chapitre montrent l'intérêt de ce genre de technique dans les réseaux de capteurs où les charges de trafic peuvent être très variables selon l'application souhaitée. Un autre point non négligeable est que nos deux algorithmes adaptent leur taille de fenêtre de contention non pas en fonction du nombre de nœuds présents dans le réseau mais selon le nombre de stations émettrices. Les résultats présentés montrent qu'avec ces deux propositions nous améliorons significativement les performances du standard IEEE 802.15.4 et ceci quelles que soient les valeurs des fenêtres de contention choisies.

## 7.2 Perspectives

### 7.2.1 Proportional fair sharing

Dans les réseaux de capteurs utilisant des topologies de type collection tree il est fort probable que certains nœuds auront une charge plus importante à écouler par rapport aux nœuds partageant le canal dans leur voisinage. Une évolution dans la méthode d'accès que nous jugeons intéressante est de pourvoir les nœuds dans un voisinage donné

d'un accès équitable aux ressources radio i.e que le partage de la bande passante prenne en compte la charge de trafic devant être écoulé par chaque nœud (proportional fair sharing).

Cette propriété peut être obtenue de différentes façons:

- En diminuant la taille des fenêtres de contention en fonction du trafic dans les buffers comme nous l'avons vu dans le chapitre 6

- En intégrant la possibilité aux coordinateurs ayant une grande charge à écouler de le faire sur différents coordinateurs évitant ainsi la saturation d'une seule super-trame avec le trafic venant d'une seule branche du cluster-tree

Nous avons vu dans le chapitre 6 que la proposition qui consiste à jouer uniquement sur la taille des fenêtres de contention en fonction du trafic à écouler ne peut être retenue. Néanmoins nous pensons que la deuxième solution que nous proposons et qui a pour principe que les nœuds auront un coordinateur principal qui se chargera du trafic descendant et montant, mais aussi des coordinateurs secondaires n'offrant qu'un support pour le trafic montant. Ceci dans le but d'éviter que les nœuds ne soient dans l'obligation de suivre les Beacons de tous leurs coordinateurs (principal et secondaire).

### 7.2.2 Routage et crosslayer

Dans la majorité des réseaux de capteurs déployés en topologie de type cluster-tree les données envoyées par les différents nœuds seront adressées au coordinateur. Ces réseaux ne nécessitent qu'une fonction de routage réduite comme par exemple le routage hiérarchique.

Dès que le déploiement du réseau de capteurs n'a pas pour objectif final d'envoyer les données mesurées au puits mais vers des nœuds intermédiaires dans le réseau la fonction de routage devient très importante. On peut retrouver ce genre de communications dans des applications de type réseau maillé par exemple dans des applications domotique, ou de déclenchement de système anti-incendie.

Dans ce genre de réseaux nous retrouverons les mêmes contraintes inhérentes aux réseaux de capteurs mais avec des contraintes temporelles plus strictes. De ce fait il faut tout mettre en œuvre afin que le protocole de routage fasse le bon choix de route afin que les paquets soient délivrés à bon port dans les plus courts délais en minimisant les retransmissions. Dans ces conditions le design du protocole de routage doit prendre avantage de toutes les propriétés présentes dans les couches inférieures telles que: la qualité du lien radio, la charge des communications dans le voisinage ainsi que les tailles de fenêtres de contention utilisées par les différents parents.

### 7.2.3 Energie sporadique

Ces dernières années nous avons pu constater l'émergence de capteurs utilisant des modules de récupération d'énergie. Cette technologie règle dans un certain sens le problème lié à la durée de vie du capteur et ainsi du réseau. D'un autre côté elle peut invalider un grand nombre de protocoles proposés pour les réseaux de capteurs en imposant de nouvelles contraintes. Nous pouvons citer comme exemple la forte dynamicité du réseau, les nœuds n'ayant qu'une faible quantité d'énergie auront des réveils non périodiques et non prédictibles par leur voisinage. Ceci est due au taux de récupération d'énergie qui ne sera pas le même pour tous les nœuds du réseau (par exemple si l'énergie solaire est utilisée le taux de récupération sera tributaire de la luminosité que recevra le nœud), ce qui impose une nouvelle manière de penser les protocoles MAC et de routage pour ces réseaux, la création et le maintient de la topologie dans ce genre de contexte nous semble une problématique intéressante.

La majorité des propositions faites pour les réseaux de capteurs partent du principe que les capteurs ont comme source d'énergie la batterie et le but de ces protocoles est d'économiser cette ressource afin de maximiser la durée de vie du réseau.

Avec les capteurs utilisant un module de récupération d'énergie, la quantité d'énergie récupéré est limitée et non disponible tout le temps (sporadique), ainsi le challenge devient non pas la maximisation de la durée de vie du réseau, mais de garantir le fonctionnement de ce dernier avec le peu d'énergie disponible. Plusieurs pistes peuvent être étudiées à cet effet:

- Couche MAC:
  - Minimisation des messages de contrôles par exemple la non utilisation des acquittements sous certaines conditions
  - Contrôle de puissance
- Couche réseau: introduction de nouvelles métrique prenant en compte les conditions environnementales (liées à la technique de récupération d'énergie) pour le calcul des routes

# Partie IV

# Bibliographie

# BIBLIOGRAPHY


[1] I.F. Akyildiz et M.C. Vuran. *Wireless Sensor Networks*. Ian F. Akyildiz Series in Communications and Networking. Wiley, 2010.

[2] V. A. Kottapalli, A. S. Kiremidjian, J. P. Lynch, E. Carryer, T. W. Kenny, K. H. Law et Y. Lei. Two-tiered wireless sensor network architecture for structural health monitoring. *In* S.-C. Liu, éditeur. *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, volume 5057 de *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, pages 8–19, août 2003.

[3] Robert Szewczyk, Alan M. Mainwaring, Joseph Polastre, John Anderson et David E. Culler. An analysis of a large scale habitat monitoring application. *In SenSys*, pages 214–226, 2004.

[4] L. Girod D. Estrin K. Yao H. Wang, J. Elson et L. Vanderberge. Target Classification and Localization in Habitat Monitoring. *In IEEE Proceedings of the International Conference on Speech and Signal Processing*, pages II–597–II–600, April 2003.

[5] Md. Asdaque Hussain, Pervez Khan et Kwak Kyung Sup. WSN research activities for military application. *In Proceedings of the 11th international conference on Advanced Communication Technology - Volume 1*, ICACT'09, pages 271–274, Piscataway, NJ, USA, 2009. IEEE Press.

[6] Alberto Cerpa, Jeremy Elson, Michael Hamilton, Jerry Zhao, Deborah Estrin et Lewis Girod. Habitat monitoring: application driver for wireless communications technology. *In Workshop on Data communication in Latin America and the Caribbean*, SIGCOMM LA '01, pages 20–41, New York, NY, USA, 2001. ACM.

[7] Mani Srivastava, Richard Muntz et Miodrag Potkonjak. Smart kindergarten: sensor-based wireless networks for smart developmental problem-solving environ-

ments. *In Proceedings of the 7th annual international conference on Mobile computing and networking*, MobiCom '01, pages 132–138, New York, NY, USA, 2001. ACM.

[8] Tia Gao, Dan Greenspan et Matt Welsh. Improving Patient Monitoring and Tracking in Emergency Response. *In International Conference on Information Communication Technologies in Health*, July 2005.

[9] David Malan, Thaddeus Fulford-Jones, Matt Welsh et Steve Moulton. CodeBlue: An Ad Hoc Sensor Network Infrastructure for Emergency Medical Care. *In MobiSys 2004 Workshop on Applications of Mobile Embedded Systems (WAMES 2004)*, June 2004.

[10] Xingfa Shen, Zhi Wang et Youxian Sun. Wireless sensor networks for industrial applications. *In Intelligent Control and Automation, 2004. WCICA 2004. Fifth World Congress on*, volume 4, pages 3636 – 3640 Vol.4, june 2004.

[11] V.C. Gungor et G.P. Hancke. Industrial Wireless Sensor Networks: Challenges, Design Principles, and Technical Approaches. *Industrial Electronics, IEEE Transactions on*, 56(10):4258 –4265, oct. 2009.

[12] Jennifer Yick, Biswanath Mukherjee et Dipak Ghosal. Wireless sensor network survey. *Computer Networks*, 52(12):2292 – 2330, 2008.

[13] K. Langendoen et G. Halkes. Energy-Efficient Medium Access Control. pages 34–1, 2005.

[14] K. Langendoen. Energy-Efficient Medium Access Control. *Book chapter in Medium Access Control in Wireless Networks, H. Wu and Y. Pan (editors), Nova Science Publishers*, 2008.

[15] Bhaskaran Raman et Kameswari Chebrolu. Censor networks: a critique of "sensor networks" from a systems perspective. *SIGCOMM Comput. Commun. Rev.*, 38(3): 75–78, juillet 2008.

[16] Abdelmalik Bachir, Mischa Dohler, Thomas Watteyne et Kin K. Leung. MAC Essentials for Wireless Sensor Networks. *IEEE Communications Surveys and Tutorials*, 12(2):222–248, 2010.

[17] V. Raghunathan, C. Schurgers, Sung Park et M.B. Srivastava. Energy-aware wireless microsensor networks. *Signal Processing Magazine, IEEE*, 19(2):40 –50, mar 2002.

[18] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam et E. Cayirci. Wireless sensor networks: a survey. *Computer Networks*, 38(4):393 – 422, 2002.

[19] Linchao Zhang, Renato Ferrero, Erwing Ricardo Sanchez et Maurizio Rebaudengo. Performance analysis of reliable flooding in duty-cycle wireless sensor networks. *Transactions on Emerging Telecommunications Technologies*, 2012.

[20] Wei Ye, J. Heidemann et D. Estrin. An energy-efficient MAC protocol for wireless sensor networks. *In INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, volume 3, pages 1567 – 1576 vol.3, 2002.

[21] IEEE Standard for Information Technology - Telecommunications and Information Exchange Between Systems - Local and Metropolitan Area Networks - Specific Requirement. Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. Amendment 2: Higher-Speed Physical Layer (PHY) Extension in the 2.4 GHz Band - Corrigendum 1. *IEEE Std 802.11b-1999/Cor 1-2001*, pages 0–1, 2001.

[22] P. Lin, C. Qiao et X. Wang. Medium access control with a dynamic duty cycle for sensor networks. *In Wireless Communications and Networking Conference, 2004. WCNC. 2004 IEEE*, volume 3, pages 1534 – 1539 Vol.3, march 2004.

[23] H. Pham et S. Jha. An adaptive mobility-aware MAC protocol for sensor networks (MS-MAC). *In Mobile Ad-hoc and Sensor Systems, 2004 IEEE International Conference on*, pages 558 – 560, oct. 2004.

[24] Tijs van Dam et Koen Langendoen. An adaptive energy-efficient MAC protocol for wireless sensor networks. *In Proceedings of the 1st international conference on Embedded networked sensor systems*, SenSys '03, pages 171–180, New York, NY, USA, 2003. ACM.

[25] G. Lu, B. Krishnamachari et C.S. Raghavendra. An adaptive energy-efficient and low-latency MAC for data gathering in wireless sensor networks. *In Parallel and Distributed Processing Symposium, 2004. Proceedings. 18th International*, page 224, april 2004.

[26] Joseph Polastre, Jason Hill et David Culler. Versatile low power media access for wireless sensor networks. *In Proceedings of the 2nd international conference on Embedded networked sensor systems*, SenSys '04, pages 95–107, New York, NY, USA, 2004. ACM.

[27] A. El-Hoiydi et J.-D. Decotignie. WiseMAC: an ultra low power MAC protocol for the downlink of infrastructure wireless sensor networks. *In Proceedings of the Ninth International Symposium on Computers and Communications 2004 Volume 2 (ISCC"04) - Volume 02*, ISCC '04, pages 244–251, Washington, DC, USA, 2004. IEEE Computer Society.

[28] Adam Dunkels. The ContikiMAC Radio Duty Cycling Protocol. Rapport technique T2011:13, Swedish Institute of Computer Science, décembre 2011.

[29] David Moss et Philip Levis. BoX-MACs: Exploiting Physical and Link Layer Boundaries in Low-Power Networking. Rapport technique, Stanford, 2008.

[30] L. Bernardo, R. Oliveira, M. Pereira, M. Macedo et P. Pinto. A Wireless Sensor MAC Protocol for Bursty Data Traffic. *In Personal, Indoor and Mobile Radio Communications, 2007. PIMRC 2007. IEEE 18th International Symposium on*, pages 1 –5, sept. 2007.

[31] Michael Buettner, Gary V. Yee, Eric Anderson et Richard Han. X-MAC: a short preamble MAC protocol for duty-cycled wireless sensor networks. *In Proceedings of the 4th international conference on Embedded networked sensor systems*, SenSys '06, pages 307–320, New York, NY, USA, 2006. ACM.

[32] Wei Ye, Fabio Silva et John Heidemann. Ultra-low duty cycle MAC with scheduled channel polling. *In Proceedings of the 4th international conference on Embedded networked sensor systems*, SenSys '06, pages 321–334, New York, NY, USA, 2006. ACM.

[33] Beakcheol Jang, Jun Bum Lim et M.L. Sichitiu. AS-MAC: An asynchronous scheduled MAC protocol for wireless sensor networks. *In Mobile Ad Hoc and Sensor Systems, 2008. MASS 2008. 5th IEEE International Conference on*, pages 434 –441, 29 2008-oct. 2 2008.

[34] Ozlem Durmaz Incel. A survey on multi-channel communication in wireless sensor networks. *Computer Networks*, 55(13):3081 – 3099, 2011.

[35] R. Soua et P. Minet. A survey on multichannel assignment protocols in Wireless Sensor Networks. *In Wireless Days (WD), 2011 IFIP*, pages 1 –3, oct. 2011.

[36] G. Zhou, C. Huang, T. Yan, T. He, J. A. Stankovic et T. F. Abdelzaher. MMSN: Multi-Frequency Media Access Control for Wireless Sensor Networks. *In INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings*, pages 1 –13, april 2006.

[37] Lichun Bao et J.J. Garcia-Luna-Aceves. A New Approach to Channel Access Scheduling for Ad Hoc Networks. *In In Proc. ACM Seventh Annual International Conference on Mobile Computing and networking*, pages 210–221, 2001.

[38] M. Salajegheh, H. Soroush et A. Kalis. HYMAC: Hybrid TDMA/FDMA Medium Access Control Protocol for Wireless Sensor Networks. *In Personal, Indoor and Mobile Radio Communications, 2007. PIMRC 2007. IEEE 18th International Symposium on*, pages 1 –5, sept. 2007.

[39] Yafeng Wu, John A. Stankovic, Tian He et Shan Lin. Realistic and Efficient Multi-Channel Communications in Wireless Sensor Networks. *In INFOCOM*, pages 1193–1201. IEEE, 2008.

[40] Ozlem Durmaz Incel, Lodewijk van Hoesel, Pierre Jansen et Paul Havinga. MC-LMAC: A multi-channel MAC protocol for wireless sensor networks. *Ad Hoc Netw.*, 9(1):73–94, janvier 2011.

[41] L.F.W. Hoesel van et P.J.M. Havinga. A Lightweight Medium Access Protocol (LMAC) for Wireless Sensor Networks: Reducing Preamble Transmissions and Transceiver State Switches. *In 1st International Workshop on Networked Sensing Systems, INSS 2004*, pages 205–208, Tokio, Japan, 2004. Society of Instrument and Control Engineers (SICE).

[42] K.R. Chowdhury, N. Nandiraju, D. Cavalcanti et D.P. Agrawal. CMAC - A multi-channel energy efficient MAC for wireless sensor networks. *In Wireless Communications and Networking Conference, 2006. WCNC 2006. IEEE*, volume 2, pages 1172 –1177, april 2006.

[43] Xun Chen, Peng Han, Qiu-Sheng He, Shi liang Tu et Zhang long Chen. A Multi-Channel MAC Protocol for Wireless Sensor Networks. *In CIT*, page 224. IEEE, 2006.

[44] Joris Borms, Kris Steenhaut et Bart Lemmens. Low-Overhead Dynamic Multi-channel MAC for Wireless Sensor Networks. *In* Jorge Silva, Bhaskar Krishnamachari et Fernando Boavida, éditeurs. *Wireless Sensor Networks*, volume 5970 de *Lecture Notes in Computer Science*, pages 81–96. Springer Berlin / Heidelberg, 2010. 10.1007/978-3-642-11917-0-6.

[45] Youngmin Kim, Hyojeong Shin et Hojung Cha. Y-MAC: An Energy-Efficient Multi-channel MAC Protocol for Dense Wireless Sensor Networks. *In IPSN*, pages 53–63. IEEE/ACM, 2008.

[46] F. Cuomo, S. Della Luna, E. Cipollone, P. Todorova et T. Suihko. Topology Formation in IEEE 802.15.4: Cluster-Tree Characterization. *In Pervasive Computing and Communications, 2008. PerCom 2008. Sixth Annual IEEE International Conference on*, pages 276 –281, march 2008.

[47] Francesca Cuomo, Emanuele Cipollone et Anna Abbagnale. Performance analysis of IEEE 802.15.4 wireless sensor networks: An insight into the topology formation process. *Comput. Netw.*, 53(18):3057–3075, décembre 2009.

[48] Jianliang Zheng et Myung J Lee. A comprehensive performance study of IEEE 802.15. 4. *Sensor network operations*, 4:1–14, 2006.

[49] Ieee 802.15.4-2006 standard, `http://www.ieee802.org/15/pub/TG4.html`.

[50] J. Misic, S. Shafi et V.B. Misic. Performance of a beacon enabled IEEE 802.15.4 cluster with downlink and uplink traffic. *Parallel and Distributed Systems, IEEE Transactions on*, 17(4):361 – 376, april 2006.

[51] J. Misic, S. Shafi et V.B. Misic. Maintaining reliability through activity management in an 802.15.4 sensor cluster. *Vehicular Technology, IEEE Transactions on*, 55(3):779 –788, may 2006.

[52] G. Anastasi, M. Conti et M. Di Francesco. The MAC Unreliability Problem in IEEE 802.15.4 Wireless Sensor Networks. *In Proceedings of MSWiM*, 2009.

[53] Gang Lu, B. Krishnamachari et C.S. Raghavendra. Performance evaluation of the IEEE 802.15.4 MAC for low-rate low-power wireless networks. *In Performance, Computing, and Communications, 2004 IEEE International Conference on*, pages 701 – 706, 2004.

[54] A. Koubaa, M. Alves et E. Tovar. GTS allocation analysis in IEEE 802.15.4 for real-time wireless sensor networks. *In Parallel and Distributed Processing Symposium, 2006. IPDPS 2006. 20th International*, page 8 pp., april 2006.

[55] Anis Koubâa, Mário Alves, Eduardo Tovar et André Cunha. An implicit GTS allocation mechanism in IEEE 802.15.4 for time-sensitive wireless sensor networks: theory and practice. *Real-Time Syst.*, 39(1-3):169–204, août 2008.

[56] Fanliang Meng et Yigang Han. A new association scheme of IEEE 802.15.4 for real-time applications. *In Proceedings of the 5th International Conference on Wireless communications, networking and mobile computing*, WiCOM'09, pages 3432–3436, Piscataway, NJ, USA, 2009. IEEE Press.

[57] Ieee 802.15.4-2003 standard, `http://www.ieee802.org/15/pub/TG4.html`.

[58] Andreas Willig. Recent and Emerging Topics in Wireless Industrial Communications: A Selection. *IEEE Trans. Industrial Informatics*, pages 102–124, 2008.

[59] A. Koubâa, A. Cunha, M. Alves et E. Tovar. TDBS: A Time Division Beacon Scheduling Mechanism for ZigBee Cluster-Tree Wireless Sensor Networks. *Real-Time Systems*, 40(3):321–354, 2008.

[60] R. Burda et C. Wietfeld. A Distributed and Autonomous Beacon Scheduling Algorithm for IEEE 802.15.4/ZigBee Networks. *In Mobile Adhoc and Sensor Systems, 2007. MASS 2007. IEEE Internatonal Conference on*, pages 1 –6, oct. 2007.

[61] Panneer Selvan Muthukumaran, Rodolfo de Paz, Rostislav Špinar et Dirk Pesch. MeshMAC: Enabling Mesh Networking over Ieee 802.15.4 through Distributed Beacon Scheduling. *In Proceedings of AdHocNets*, Niagara Falls, Ontario, Canada, September 2009.

[62] H. Jeon et Y. Kim. BOP (Beacon-Only Period) and Beacon Scheduling for MEU (Mesh-Enabled USN) Devices. *In Proceedings of ICACT*, February 2007.

[63] Berta Carballido Villaverde, Rodolfo De Paz Alberola, Susan Rea et Dirk Pesch. Experimental Evaluation of Beacon Scheduling Mechanisms for Multihop Ieee 802.15.4 Wireless Sensor Networks. *In Proceedings of SENSORCOMM*, 2010.

[64] Eui-Jik Kim et Hyo-Hyun Choi. EBBS: Energy-efficient BOP-based Beacon transmission Scheduling for WSNs. *In Personal, Indoor and Mobile Radio Communications, 2008. PIMRC 2008. IEEE 19th International Symposium on*, pages 1 –6, sept. 2008.

[65] Ed Callaway. Cluster Tree Network : MAC Proposal for the Low Rate 802.15.4 Standard, april 2001.

[66] C.E. Perkins et E.M. Royer. Ad-hoc on-demand distance vector routing. *In Mobile Computing Systems and Applications, 1999. Proceedings. WMCSA '99. Second IEEE Workshop on*, pages 90 –100, feb 1999.

[67] C. Chaudet, D. Dhoutaut et I.G. Lassous. Performance issues with IEEE 802.11 in ad hoc networking. *Communications Magazine, IEEE*, 43(7):110 – 116, july 2005.

[68] Aravind Iyer et Catherine Rosenberg. Understanding the key performance issues with MAC protocols for multi-hop wireless networks: Research articles. *Wirel. Commun. Mob. Comput.*, 6(6):745–760, septembre 2006.

[69] Jorge Crichigno, Min-You Wu et Wei Shu. Protocols and architectures for channel assignment in wireless mesh networks. *Ad Hoc Networks*, 6(7):1051 – 1077, 2008.

[70] Jeonghoon Mo, H.-S.W. So et J. Walrand. Comparison of Multichannel MAC Protocols. *Mobile Computing, IEEE Transactions on*, 7(1):50 –65, jan. 2008.

[71] N. Jain, S.R. Das et A. Nasipuri. *In Computer Communications and Networks, 2001. Proceedings. Tenth International Conference on, title=A multichannel CSMA MAC protocol with receiver-based channel selection for multihop wireless networks*, pages 432 –439, 2001.

[72] Jiandong Li, Z.J. Haas, Min Sheng et Yanhui Chen. Performance evaluation of modified IEEE 802.11 MAC for multi-channel multi-hop ad hoc network. *In*

*Advanced Information Networking and Applications, 2003. AINA 2003. 17th International Conference on*, pages 312 – 317, march 2003.

[73] Shih-Lin Wu, Chih-Yu Lin, Yu-Chee Tseng et Jang-Laing Sheu. A new multi-channel MAC protocol with on-demand channel assignment for multi-hop mobile ad hoc networks. *In Parallel Architectures, Algorithms and Networks, 2000. I-SPAN 2000. Proceedings. International Symposium on*, pages 232 –237, 2000.

[74] Paramvir Bahl, Ranveer Chandra et John Dunagan. SSCH: slotted seeded channel hopping for capacity improvement in IEEE 802.11 ad-hoc wireless networks. *In Proceedings of the 10th annual international conference on Mobile computing and networking*, MobiCom '04, pages 216–230, New York, NY, USA, 2004. ACM.

[75] Hoi sheung Wilson So et Jean Walrand. McMAC: A Multi-Channel MAC Proposal for Ad-Hoc Wireless Networks. Rapport technique, IN PROC. OF IEEE WCNC 2007, HONGKONG, 2005.

[76] Jungmin So et Nitin H. Vaidya. Multi-channel mac for ad hoc networks: handling multi-channel hidden terminals using a single transceiver. *In Proceedings of the 5th ACM international symposium on Mobile ad hoc networking and computing*, MobiHoc '04, pages 222–233, New York, NY, USA, 2004. ACM.

[77] R. Maheshwari, H. Gupta et S.R. Das. Multichannel MAC Protocols for Wireless Networks. *In Sensor and Ad Hoc Communications and Networks, 2006. SECON '06. 2006 3rd Annual IEEE Communications Society on*, volume 2, pages 393 –401, sept. 2006.

[78] Jingbin Zhang, Gang Zhou, Chengdu Huang, S.H. Son et J.A. Stankovic. TMMAC: An Energy Efficient Multi-Channel MAC Protocol for Ad Hoc Networks. *In Communications, 2007. ICC '07. IEEE International Conference on*, pages 3554 –3561, june 2007.

[79] N. Shacham et P. King. Architectures and Performance of Multichannel Multihop Packet Radio Networks. *Selected Areas in Communications, IEEE Journal on*, 5(6):1013 – 1025, jul 1987.

[80] X. Liu, C. Leckie et S.K. Saleem. Performance evaluation of a converge-cast protocol for IEEE 802.15.4 tree-based networks. *In International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP)*, pages 73 –78. IEEE, dec. 2010.

[81] P.S. Muthukumaran, R. de Paz, R. Špinar et D. Pesch. MeshMAC: Enabling Mesh Networking over IEEE 802.15.4 through Distributed Beacon Scheduling. *In Proceedings of AdHocNets*, 2009.

[82] R. Jain, D. Chiu et W. Hawe. A Quantitative Measure Of Fairness And Discrimination For Resource Allocation In Shared Computer Systems. *eprint arXiv:cs/9809099*, septembre 1998.

[83] A. Fraboulet, G. Chelius et E. Fleury. Worldsens: Development and Prototyping Tools for Application Specific Wireless Sensors Networks. *In Proceedings of IPSN, SPOTS track*, 2007.

[84] Elyes Ben Hamida, Guillaume Chelius et Jean-Marie Gorce. On the Complexity of an Accurate and Precise Performance Evaluation of Wireless Networks using Simulations. *In MSWiM*, Vancouver, British Columbia, Canada, October 2008. ACM.

[85] Jelena Misic, Shairmina Shafi et Vojislav B. Misic. The impact of MAC parameters on the performance of 802.15.4 PAN. *Ad Hoc Networks*, 3(5):509 – 528, 2005. <ce:title>Data Communication and Topology Control in Ad Hoc Networks</ce:title>.

[86] Pangun Park, P. Di Marco, P. Soldati, C. Fischione et K.H. Johansson. A generalized Markov chain model for effective analysis of slotted IEEE 802.15.4. *In Mobile Adhoc and Sensor Systems, 2009. MASS '09. IEEE 6th International Conference on*, page 130 139, oct. 2009.

[87] A. Koubaa, M. Alves et E. Tovar. A comprehensive simulation study of slotted CSMA/CA for IEEE 802.15.4 wireless sensor networks. *In Factory Communication Systems, 2006 IEEE International Workshop on*, pages 183 –192, 0-0 2006.

[88] D. Koscielnik. Influence of the hidden stations and the exposed station for the throughput of the LR-WPAN. *In Industrial Electronics, 2009. ISIE 2009. IEEE International Symposium on*, pages 1190 –1195, july 2009.

[89] Jin-Shyan Lee. An experiment on performance study of IEEE 802.15.4 wireless networks. *In Emerging Technologies and Factory Automation, 2005. ETFA 2005. 10th IEEE Conference on*, volume 2, pages 8 pp. –458, sept. 2005.

[90] Mario Di Francesco, Giuseppe Anastasi, Marco Conti, Sajal K. Das et Vincenzo Neri. Reliability and Energy-Efficiency inIEEE 802.15.4/ZigBee Sensor Networks: An Adaptive and Cross-Layer Approach. *IEEE Journal on Selected Areas in Communications*, 29(8):1508–1524, 2011.

[91] Mario Di Francesco, Giuseppe Anastasi, Marco Conti, Sajal K. Das et Vincenzo Neri. An adaptive algorithm for dynamic tuning of MAC parameters in IEEE 802.15.4/ZigBee sensor networks. *In PerCom Workshops*, pages 400–405, 2010.

[92] Pangun Park, Carlo Fischione et Karl Henrik Johansson. Adaptive IEEE 802.15.4 protocol for energy efficient, reliable and timely communications. *In International Conference on Information Processing in Sensor Networks (IPSN)*, pages 327–338, Stockholm, Sweden, 2010. ACM/IEEE.

[93] Ricardo Severino, Manish Batsa, Mario Alves et Anis Koubaa. A Traffic Differentiation Add-On to the IEEE 802.15.4 Protocol: Implementation and Experimental Validation over a Real-Time Operating system. *In Proceedings of the 2010 13th Euromicro Conference on Digital System Design: Architectures, Methods and Tools*, DSD '10, pages 501–508, Washington, DC, USA, 2010. IEEE Computer Society.

[94] Vaddina Rao et Dimitri Marandin. Adaptive Backoff Exponent Algorithm for Zigbee (IEEE 802.15.4). *In* Yevgeni Koucheryavy, Jarmo Harju et Villy Iversen, éditeurs. *Next Generation Teletraffic and Wired/Wireless Advanced Networking*, volume 4003 de *Lecture Notes in Computer Science*, pages 501–516. Springer Berlin / Heidelberg, 2006. 10.1007/11759355-46.

[95] Martin Heusse, Franck Rousseau, Romaric Guillier et Andrzej Duda. Idle sense: an optimal access method for high throughput and fairness in rate diverse wireless LANs. *In Proceedings of the 2005 conference on Applications, technologies, architectures, and protocols for computer communications*, SIGCOMM '05, pages 121–132, New York, NY, USA, 2005. ACM.

[96] Yan Grunenberger, Martin Heusse, Franck Rousseau et Andrzej Duda. Experience with an Implementation of the Idle Sense Wireless Access Method. *In Proceedings of the 2007 ACM CoNEXT conference (CoNEXT'07)*, pages 1–12, Columbia University, New York, NY, décembre 10–13 2007. ACM.

# Partie V

# Liste des publications scientifiques et brevets

# LISTE DES PUBLICATIONS

## Conférence

- Nazim Abdeddaim, Fabrice Theoleyre, Franck Rousseau, Andrzej Duda. Multi-Channel Cluster Tree for 802.15.4 Wireless Sensor Networks. In Proceedings of the 23rd IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC'12). Sydney, Australia, September 9-12, 2012.

## Chapitre de livre

- Nazim Abdeddaim , Darties Benoît and Theoleyre Fabrice, "Bandwidth and Energy Consumption Tradeoff for IEEE 802.15.4 in Multihop Topologies" in the book "Wireless Sensor Networks - Technology and Applications" edited by Mohammad Matin, ISBN 978-953-51-0676-0, InTech, July 7, 2012

## Brevet

- Nazim Abddedaim, Fabrice Theoleyre and Benoit Ponsard, "Method for Constructing a Cluster Tree Topology in a Personal Area Network" European Patent - 12 305 800.0 - July, 3rd. 2012

**Analyse des performances d'un réseau de capteurs exploitant le standard IEEE 802.15.4**

Résumé

Les réseaux de capteurs suscitent un engouement croissant du fait du grand nombre d'applications mais également des défis inhérents à ce genre de réseaux. Le standard IEEE 802.15.4 a été proposé afin de standardiser les couches physique et MAC.

Dans ce travail nous avons dans un premier temps proposé une variante multi-canal pour le standard 802.15.4 permettant de résoudre le problème de collisions de supertrames. Pour cela nous proposons de construire un réseau en arbre avec la particularité que chaque cellule du réseau utilise un canal différent permettant ainsi une réduction conséquente des interférences et augmente la capacité du réseau. Nous avons également introduit un nouveau mécanisme de construction de topologie, d'allocation de canal et d'ordonnancement de supertrames nécessaire au bon fonctionnement d'une telle solution.

Dans un deuxième temps nous avons analysé l'impact des différents paramètres de la méthode d'accès du standard. Nous avons mis en exergue les faiblesses de la méthode d'accès dont les performances baissent drastiquement pour des réseaux trop importants.

Partant de ce constat, nous avons proposé des mécanismes d'auto-adaptation pour la méthode d'accès du standard. Ces derniers permettent d'adapter dynamiquement la taille des fenêtres de contention en fonction des conditions de trafic observées sur le canal. Le calcul des valeurs optimales est exécuté par chaque coordinateur pour résoudre le problème de surdité. Ces mécanismes sont distribués et convergent rapidement même en cas de trafic en rafales.

**Mots clès :** réseaux sans fil, réseaux de capteurs, IEEE 802.15.4, IEEE 802.11, méthodes d'accès, multi-canal.

**Performance Analysis of Wirless Sensor Networks Exploiting the Standard IEEE 802.15.4**

Abstract

An increasing interest has been observed in Wirless Sensor Network that can be explained by wide range of WSN applications as well as by the challenges involving the constraints of this type of networks. The IEEE 802.15.4 standard has been proposed with the objective of standardizing the physical and MAC layers.

In this work, we have firstly proposed an alternative multichannel scheme for the IEEE 802.15.4 standard. It is able to solve the problem of superframe collisions based on a cluster-tree topology approach in which each cluster uses a different channel allowing the reduction of the interference and increased network capacity. We have also difined a novel mechanism for topology construction, channel allocation, and superframe scheduling.

Secondly, we have analyzed the impact of different parameters on the medium access control. We have shown the weakness of the medium access method proposed in the standard. For instance, we have observed a decrease in performance when the standard is applied in large networks.

Motivated by the analysis and its results we have then proposed auto-adaptive mechanisms for the medium access control. They allow to dynamically adapt the size of the contention window according to the observed traffic conditions. Each coordinator computes the optimal values to avoid deafness. The proposed mechanisms rapidly converge even in the case of bursty traffic.

**Keywords:** wireless networks, wireless sensor networks, IEEE 802.15.4, IEEE 802.11, Medium access control, Multi-channel.

**Discipline :** Informatique
**Laboratoire :** LIG – quipe Drakkar
BP 72, 38402 Saint Martin d'Hres CEDEX, France