



**HAL**  
open science

# Ordonnancement et routage pour l'augmentation de la durée de vie dans les réseaux de capteurs sans fil

Lilia Lassouaoui

► **To cite this version:**

Lilia Lassouaoui. Ordonnancement et routage pour l'augmentation de la durée de vie dans les réseaux de capteurs sans fil. Sciences de l'information et de la communication. Conservatoire national des arts et métiers - CNAM, 2018. Français. NNT : 2018CNAM1187 . tel-01933268

**HAL Id: tel-01933268**

**<https://theses.hal.science/tel-01933268>**

Submitted on 23 Nov 2018

**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.

École Doctorale Informatique, Télécommunications et Électronique

Laboratoire Centre d'étude et de recherche en informatique et communications

## THÈSE DE DOCTORAT

*présentée par* : **Lilia LASSOUAOUI**

*soutenue le* : **06 JUILLET 2018**

*pour obtenir le grade de* : **Docteur du Conservatoire National des Arts et Métiers**

*Discipline* : **Sciences de l'information et de la communication**

*Spécialité* : **Informatique**

### Ordonnancement et routage pour l'augmentation de la durée de vie dans les réseaux de capteurs sans fil

#### THÈSE DIRIGÉE PAR

Mr. GRESSIER-SOUDAN Éric

*Professeur, CNAM*

Mme. WEI Anne

*Professeuse, CNAM*

Mr. ROVEDAKIS Stéphane

*Maître de conférences, CNAM*

#### RAPPORTEURS

Mr. BOURGEOIS Julien

*Professeur, Université de Franche-Comté*

Mr. VAL Thierry

*Professeur, Université de Toulouse Jean Jaurès*

#### EXAMINATRICES

Mme. POTOP-BUTUCARU Maria

*Président du jury, Professeur, Sorbonne Université*

Mme. CHABRIDON Sophie

*Directrice d'études (HDR), Télécom Paris-Sud*

Mme. MINET Pascale

*Chercheur (HDR), Inria de Paris*

Mme. QUADRI Dominique

*Maître de conférences (HDR), Université Paris-Sud*

*"Certains rêvent du succès pendant que d'autres se lèvent tôt et travaillent dur pour cela."  
Winston Churchill*

# Remerciements

C'est avec la plus profonde gratitude que je réserve ces lignes en signe de reconnaissance à tous ceux qui ont contribué de près ou de loin à l'élaboration de ce travail.

Mes premiers remerciements vont tout d'abord à ma directrice de thèse qui nous a quitté l'an dernier, Madame Anne Wei, Professeur au Conservatoire National des arts et Métiers. Je salue son courage, sa patience, sa rigueur ainsi que ses grandes valeurs humaines.

Mes profonds remerciements s'adressent aussi à mon encadrant Monsieur Stéphane Rovedakis, Maître de conférences au Conservatoire National des arts et Métiers, pour sa disponibilité, ses conseils lucides et son précieux encadrement qui ont permis de mener à bien ce travail.

Je remercie sincèrement Monsieur Eric Gressier-Soudan, Professeur au Conservatoire National des arts et Métiers, pour l'intérêt qu'il a bien voulu porter à ce travail en acceptant de reprendre la direction de ma thèse suite au décès de ma directrice.

Je remercie également mes rapporteurs et examinatrices d'avoir acceptés de rapporter ce travail et de faire partie de mon jury de thèse.

Je voudrais aussi remercier Madame Françoise Sailhan, Maître de conférences au Conservatoire National des arts et Métiers, et Viviane Gal, doctorante au Conservatoire National des arts et Métiers, pour leur sympathie, leur soutien, leur écoute et pour leur précieux conseils.

Je souhaite également remercier tous les membres du laboratoire CEDRIC, chercheurs, doctorants, stagiaires, pour leur sympathie et pour l'ambiance amicale qui règne au sein du laboratoire CEDRIC.

Enfin, je dédie ce travail à toutes les personnes chères à mes yeux, en particulier à ma mère Nouara, à mon père Belkacem, pour tout ce qu'ils ont fait pour m'apporter le bonheur. Qu'ils sachent à travers ces quelques mots combien je leur suis reconnaissante et combien je sais tout ce que je leur dois.

---

# Résumé

Les réseaux de capteurs sans fil (RCSF) sont très largement utilisés dans divers domaines d'applications civiles et militaires, comme la surveillance de champs de bataille, la détection de phénomène environnementaux, ou encore les bâtiments intelligents. Cependant, les réseaux de capteurs sans fil sont caractérisés par de fortes limitations au niveau de l'énergie disponible et des communications radio. Les travaux réalisés dans cette thèse visent à proposer des solutions garantissant une certaine qualité de service dans le contexte des réseaux de capteurs sans fil. La première partie concerne la couche liaison de données avec l'objectif d'augmenter la durée de vie du réseau. L'accès au médium sans fil est analysé et modélisé sous la forme d'un problème d'ordonnancement des liens de communication, tenant compte des collisions. Nous étudions alors la complexité de ce problème. Une approche distribuée et tolérante aux défaillances avec garantie de performance est proposée (SS-DD2EC) pour résoudre ce problème. La seconde partie de la thèse concerne le routage des messages à l'aide du protocole IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL). Tout d'abord, un comparatif entre les différentes métriques de routage existantes pour l'optimisation de l'énergie consommée a été mené. En plus de la durée de vie, les critères de fiabilité et de latence de bout-en-bout sont considérés pour évaluer ces métriques. Enfin, deux nouvelles métriques ( $R\_MinMax$  et  $R\_Delai$ ) ont été proposées pour RPL permettant d'atteindre des gains significatifs par rapport à l'état de l'art. La première ne tient compte que des critères de consommation d'énergie et de fiabilité, alors que la seconde intègre en plus la latence de bout-en-bout.

**Mots clés :** Réseaux de capteurs sans fil, Accès au médium, Coloration d'arêtes à distance-2, Routage, métrique RPL.



# Abstract

Wireless Sensor Networks (WSNs) is a technology that has a wide range of civil or military applications, including battlefield monitoring, environmental monitoring or smart city. However, WSN are characterised by high limitations in terms of energy (battery-operated nodes) and wireless links (low power and lossy links). The work done in this PhD thesis aims to provide solutions that guarantee a certain quality of service in the context of wireless sensor networks. The first part of this work concerns the medium access control layer with the aim of increasing the lifetime of the network. The access to the wireless medium is analyzed and modeled as a link scheduling problem, taking into account collisions. First, a study of the complexity of this problem is carried out, then a distributed and fault-tolerant approach with guaranteed performance is proposed (SS-DD2EC) to solve this problem. The second part is about message routing with the IPv6 Routing Protocol for Low Power and Lossy Networks (RPL). First of all, a comparison between the various existing routing metrics for the optimization of the energy consumed has been carried out. In addition of lifetime, the reliability and end-to-end latency criteria are considered for evaluating these metrics. Then, two new RPL metrics ( $R\_MinMax$  and  $R\_Delay$ ) were proposed, achieving significant gains over the state of the art. The first one only considers the energy consumption and reliability, while the second one takes also into account the end-to-end latency.

**Keywords :** Wireless Sensor Networks, Medium Access Control, Distance-2 edge coloring, Message routing, RPL metrics.



# Publications

Les contributions de cette thèse ont mené a des publications dans des conférences internationales et conférence nationale, enfin un article journal est en soumission.

## Journal International avec comité de lecture :

- **L. Lassouaoui, S. Rovedakis, A. Wei "Performance Evaluation of RPL Energy Aware Routing Metrics"**, en soumission dans Computer Networks.

## Conférences Internationales avec comité de lecture :

- **L. Lassouaoui, S. Rovedakis, A. Wei, L. Gui "Deterministic Allocation by Oriented Edge Coloring for Wireless Sensor Networks"**, IEEE 83rd Vehicular Technology Conference, January 2016, pp.1-5, Nanjing, China.
- **L. Lassouaoui, S. Rovedakis, F. Sailhan, A. Wei "Comparison of RPL Routing Metrics on Grids"**, 8th EAI International Conference on Ad Hoc Networks, September 2016, pp.1-12, Ottawa, Canada.
- **L. Lassouaoui, S. Rovedakis, F. Sailhan, A. Wei "Evaluation of energy aware routing metrics for RPL"**, The 12th IEEE International Conference on Wireless and Mobile Computing, Networking and Communications., October 2016, pp.1-8, New York, USA.

## Conférence Nationale avec comité de lecture :

- **L. Lassouaoui, S. Rovedakis, A. Wei "Routage dynamique adapté à l'environnement interactif sans fil"**, GRES2014, December 2014, pp.1-2,

# Table des matières

<b>1</b>	<b>Introduction</b>	<b>18</b>
	Introduction	18
<b>2</b>	<b>Introduction aux réseaux de capteurs sans fil</b>	<b>21</b>
2.1	Introduction . . . . .	21
2.2	Définition et caractéristiques des RCSF . . . . .	21
2.2.1	Définition . . . . .	21
2.2.2	Caractéristiques des RCSF . . . . .	23
2.2.3	Catégories des RCSF . . . . .	25
2.3	Domaines et défis d'applications des RCSF . . . . .	26
2.3.1	Domaines d'application des RCSFs . . . . .	26
2.3.2	Défis des applications RCSF . . . . .	28
2.4	Architecture et pile protocolaire . . . . .	31
2.4.1	Les standards et normes . . . . .	31
2.4.2	Architectures . . . . .	34
2.4.3	Pile protocolaire . . . . .	36
2.5	Conclusion . . . . .	39
<b>3</b>	<b>Méthodes d'accès au médium et routage pour les réseaux de capteurs sans fil</b>	<b>40</b>

## TABLE DES MATIÈRES

---

3.1	Introduction . . . . .	40
3.2	Les principales sources de surconsommation d'énergie . . . . .	41
3.3	Méthodes d'accès au médium radio . . . . .	43
3.3.1	La couche MAC de la norme IEEE 802.15.4 . . . . .	46
3.3.2	Accès avec contention . . . . .	49
3.3.3	Accès sans contention (par planification) . . . . .	53
3.3.4	Accès hybrides . . . . .	55
3.4	Routage dans les RCSF . . . . .	57
3.4.1	Protocoles de routage proactifs . . . . .	59
3.4.2	Protocoles de routage réactifs . . . . .	61
3.5	Conclusion . . . . .	64
<b>4</b>	<b>Ordonnancement sans collision des transmissions</b>	<b>65</b>
4.1	Introduction . . . . .	65
4.2	Modélisation de l'accès au médium . . . . .	66
4.2.1	Caractéristiques des communications sans fil . . . . .	67
4.2.2	Problèmes d'ordonnancement considérés dans la littérature . . . . .	69
4.2.3	Problème à résoudre et complexité . . . . .	74
4.3	Algorithme distribué DD2EC . . . . .	79
4.3.1	État de l'art . . . . .	80
4.3.2	Modélisation d'un système distribué . . . . .	85
4.3.3	Présentation de l'algorithme DD2EC . . . . .	86
4.4	Algorithme auto-stabilisant SS-DD2EC . . . . .	100
4.4.1	Modélisation d'un système auto-stabilisant . . . . .	100
4.4.2	État de l'art . . . . .	101
4.4.3	Présentation de l'algorithme SS-DD2EC . . . . .	103

## TABLE DES MATIÈRES

---

4.5	Conclusion . . . . .	106
<b>5</b>	<b>Évaluation de métriques de routage pour RPL</b>	<b>108</b>
5.1	Introduction . . . . .	108
5.2	Présentation du protocole de routage RPL . . . . .	109
5.2.1	Messages de contrôle RPL . . . . .	110
5.2.2	Construction du DODAG . . . . .	112
5.2.3	Algorithme Trickle . . . . .	113
5.2.4	Mécanisme de détection de boucle . . . . .	114
5.3	Fonction Objectif . . . . .	114
5.4	Métriques de routage pour RPL . . . . .	116
5.4.1	Métriques de routage liées à l'énergie proposées pour RPL . . . . .	118
5.5	Évaluation des métriques . . . . .	125
5.5.1	Configuration des simulations . . . . .	125
5.5.2	Critères d'évaluation . . . . .	126
5.5.3	Simulations et Analyse . . . . .	128
5.6	Conclusion . . . . .	135
<b>6</b>	<b>Nouvelles métriques de routage pour RPL</b>	<b>138</b>
6.1	Introduction . . . . .	138
6.2	Première proposition de métrique . . . . .	139
6.2.1	Contexte et motivation . . . . .	139
6.2.2	Modélisation . . . . .	140
6.2.3	Analyse des résultats de simulation . . . . .	143
6.3	Deuxième proposition de métrique . . . . .	150
6.3.1	Contexte et Motivation . . . . .	150
6.3.2	Modélisation . . . . .	152

## TABLE DES MATIÈRES

---

6.3.3	Analyse des résultats de simulation . . . . .	159
6.4	Évaluation des métriques dans une topologie hétérogène . . . . .	166
6.4.1	Configuration des simulations . . . . .	166
6.4.2	Simulations et Analyse . . . . .	167
6.5	Conclusion . . . . .	170
<b>7</b>	<b>Conclusion et perspectives</b>	<b>174</b>
	<b>Conclusion et perspectives</b>	<b>174</b>
7.1	Conclusion . . . . .	174
7.2	Perspectives . . . . .	176
	<b>Bibliographie</b>	<b>177</b>
	<b>Annexes</b>	<b>193</b>
	<b>Glossaire</b>	<b>204</b>

# Liste des tableaux

4.1	Plan d'ordonnement des slots . . . . .	70
4.2	Tableau de comparaison . . . . .	85
5.1	Métriques étudiées . . . . .	125
5.2	Paramètres de Simulation . . . . .	127
5.3	Distribution des nœuds suivant la distance Euclidienne à la racine . . . . .	127
5.4	Durée de vie du réseau -1- . . . . .	132
6.1	Durée de vie du réseau -2- . . . . .	146
6.2	Durée de vie du réseau -3- . . . . .	164

# Table des figures

2.1	Architecture RCSF [Akyildiz and Vuran 2010] . . . . .	22
2.2	Composants d'un nœud capteur . . . . .	23
2.3	Différentes caractéristiques des capteurs . . . . .	24
2.4	Domaines d'applications des RCSF . . . . .	27
2.5	Pile Protocolaire des réseaux de capteurs sans fil . . . . .	32
2.6	Principe d'architecture LoRa/Sigfox . . . . .	34
2.7	Les topologies des réseaux de capteurs sans fil . . . . .	35
3.1	Consommation énergétique liée aux tâches exécutées sur un capteur. . . . .	41
3.2	Nœud caché et nœud exposé . . . . .	42
3.3	Structure d'une superframe dans la norme IEEE 802.15.4. . . . .	46
3.4	Modes de fonctionnement de la sous-couche MAC de la norme IEEE 802.15.4. . . . .	47
3.5	Phase de transmission dans ContikiMAC [Dunkels 2011] . . . . .	52
3.6	Phase de transmission dans ContikiMAC [Dunkels 2011] . . . . .	53
3.7	Protocole Z-MAC [Rhee et al. 2008] . . . . .	56
3.8	Protocole OLSR . . . . .	60

TABLE DES FIGURES

---

4.1	Illustration des situations de conflits primaires et secondaires, où une flèche indique l'orientation d'une transmission de message et un trait pointillé représente une communication possible entre deux nœuds. Le schéma de gauche présente le Cas 1, les nœuds $b_1$ et $c_1$ envoient un message à destination du même nœud $a_1$ . Le schéma au centre concernant le Cas 2, le nœud $a_2$ envoie un message au nœud $b_2$ pendant qu'il reçoit un message envoyé par le nœud $c_2$ . Le schéma de droite illustre le Cas 3, le nœud $c_3$ reçoit simultanément un message qui lui est destiné envoyé par le nœud $d_3$ et un message envoyé par $b_3$ uniquement à destination de $a_3$ . . . . .	68
4.2	Coloration d'un graphe pour l'allocation des slots. . . . .	70
4.3	Coloration de sommets et coloration d'arêtes. . . . .	71
4.4	Illustration du problème de coloration à un saut. . . . .	72
4.5	Coloration non-orientée et orientée d'arêtes. . . . .	73
4.6	Illustration de la réduction du Problème 1 au problème de coloration de sommets, (a) instance pour le Problème 1 où les flèches représentent l'arbre orienté $T$ , les arêtes en pointillés les collisions secondaires, et le nombre sur chaque arête correspond à l'intervalle de temps affecté, et (b) instance pour le problème de coloration de sommets où le nombre à côté de chaque sommet correspond à sa couleur. . . . .	76
4.7	Illustration de la réduction du Problème 1 au problème de coloration de sommets dans le cas où le problème est polynomial à résoudre de façon optimale, (a) instance pour le Problème 1 où les flèches représentent l'arbre orienté $T$ et le nombre sur chaque arête correspond à l'intervalle de temps affecté, et (b) instance pour le problème de coloration de sommets où le nombre à côté de chaque sommet correspond à sa couleur. . . . .	79



## TABLE DES FIGURES

---

4.8	Illustration de la première heuristique proposée dans [Ghosh et al. 2011] et extraite de l'article. (a) Le graphe initial modélisant le réseau avec en grisé les nœuds récepteurs, en noir les arêtes de l'arbre et en pointillées les arêtes d'interférence, (b) l'affectation des canaux en écoute pour les nœuds récepteurs dans le graphe de conflit construit composé d'arêtes supplémentaires par rapport à l'arbre initial pour modéliser les conflits primaires et secondaires entre nœuds récepteurs, et (c) l'affectation des slots indiquée sur chaque arête de l'arbre. . . . .	81
4.9	Illustration des priorités en mode réception. (a) situation où $d_T(q) < d_T(p)$ , (b) situation où $d_T(q) = d_T(p)$ et l'identifiant du père de $q$ est inférieur à celui du père de $p$ , (c) situation où $d_T(q) = d_T(p) + 1$ et l'identifiant du père de $q$ est inférieur à celui du père de $p$ . . . . .	88
4.10	Illustration des priorités en mode émission. (d) situation où $d_T(q) < d_T(p) - 1$ , (e) situation où $d_T(q) = d_T(p) - 1$ et l'identifiant de $q$ est inférieur à celui du père de $p$ dans $T$ . . . . .	88
4.11	Exemple d'exécution de l'algorithme DD2EC. . . . .	93
5.1	Exemple de DAG et DODAG RPL . . . . .	110
5.2	Format des messages de contrôle RPL [Gaddour and Koubaa 2012] . . . . .	111
5.3	Format des messages de type DIO [Vasseur et al. March 2012] . . . . .	111
5.4	Format des messages de type DIS [Vasseur et al. March 2012] . . . . .	112
5.5	Format des messages de type DAO [Vasseur et al. March 2012] . . . . .	112
5.6	Construction du DODAG avec la métrique ETX. . . . .	119
5.7	Construction du DODAG avec la métrique ENG_MinMax. . . . .	122
5.8	Construction du DODAG avec la métrique ELT. . . . .	123
5.9	Les topologies de simulation . . . . .	127
5.10	Consommation énergétique. . . . .	131
5.11	Taux de livraison des paquets. . . . .	133

TABLE DES FIGURES

---

5.12	Le délai de bout en bout. . . . .	134
5.13	Les Messages de contrôle. . . . .	136
6.1	Construction du DODAG RPL avec plusieurs métriques à partir du nœud F : (a) exemple de réseau avec le DODAG construit avant l'ajout de F, (b) choix du parent préféré avec la métrique $ENG\_MinMax$ , (c) choix du parent préfére avec la métrique $R$ , (d) choix du parent préféré avec la métrique $R\_MinMax$ proposée. . . . .	142
6.2	Consommation énergétique. . . . .	146
6.3	Taux de livraison des paquets. . . . .	148
6.4	Le délai de bout en bout. . . . .	150
6.5	Les Messages de contrôle. . . . .	151
6.6	Métrique isotone [Yang and Wang 2008]. . . . .	155
6.7	Construction du DODAG RPL avec la métrique composite $R\_Delai$ en prenant en compte la paire $\langle ETX, ENG\_MinMax \rangle$ lors du choix du parent préféré. . . . .	159
6.8	Le délai de bout en bout (approche lexicographique). . . . .	160
6.9	Le délai de bout en bout (approche additive). . . . .	162
6.10	Consommation énergétique. . . . .	163
6.11	Taux de livraison des paquets. . . . .	165
6.12	Messages de contrôle. . . . .	166
6.13	Topologie de simulation . . . . .	167
6.14	Consommation énergétique. . . . .	168
6.15	Taux de livraison des paquets. . . . .	170
6.16	Le délai de bout en bout. . . . .	171
6.17	Les Messages de contrôle. . . . .	172

# Chapitre 1

## Introduction

Depuis une dizaine d'années, le contexte des réseaux de capteurs sans fil a conduit à beaucoup de travaux de recherche aussi bien théoriques qu'appliqués en lien avec les nouveaux domaines d'applications connexes qui ont émergé récemment comme par exemple les villes intelligentes [Zanella et al. 2014]. Grâce aux avancées technologiques dans le domaine sans fil, dans la conception des capteurs et leur miniaturisation, l'embarquement des capteurs dans divers objets de la vie quotidienne est favorisé. De plus, le faible coût de fabrication et la mise en œuvre facilitées grâce aux communications sans fil conduisent à une très large adoption de ce type de réseaux. On les retrouve donc dans divers domaines applicatifs, tels que le domaine environnemental pour la surveillance des catastrophes naturelles (séisme, inondation, éruption volcanique...) [Akyildiz and Vuran 2010], le domaine militaire pour la détection d'ennemis ou la télésurveillance des champs de bataille, le domaine de la santé pour le suivi des patients, ou encore le domaine civil avec les villes intelligentes [Rawat et al. 2014b].

Un réseau de capteurs sans fil peut être constitué de quelques nœuds à plusieurs centaines de nœuds déployés sur une zone à surveiller, capables de s'auto-organiser pour capturer les données environnementales puis les acheminer vers un ou plusieurs nœuds destinataires d'une manière collaborative. Les caractéristiques spécifiques des nœuds capteurs (petite taille des nœuds, ressources matérielles limitées, alimentation par des batteries et communication sans fil à faible débit et de mauvaise qualité) impliquent des contraintes et limitations en terme de fonctionnement. Ainsi, il est primordial de prendre en compte

notamment la durée de vie du réseau et les pannes dans les protocoles dédiés à ces réseaux. En effet, les nœuds d'un réseau de capteurs sont en général alimentés par une batterie intégrée non dissociable et non rechargeable. De plus, pour certains contextes applicatifs, les nœuds sont déployés dans des environnements hostiles et difficiles d'accès pour une intervention humaine. C'est pourquoi le premier point à prendre en compte est de minimiser la consommation en énergie des nœuds, afin d'augmenter la durée de vie du réseau. La communication radio est souvent la partie la plus gourmande en terme de consommation énergétique sur les nœuds capteurs. Cette consommation dépend à la fois de la quantité des données, de la distance de transmission, et des caractéristiques physiques de l'antenne radio. Pour augmenter la durée de vie du réseau, il faut agir sur les protocoles traitant les communications réalisées entre les différents nœuds dans le réseau afin de tenir compte de l'énergie disponible sur les nœuds et/ou réduire leur impact sur la consommation en énergie.

Cette thèse vise à apporter des solutions maximisant la durée de vie du réseau tout en prenant en compte les contraintes liées aux réseaux de capteurs. Pour atteindre cet objectif, nous nous sommes focalisés d'une part sur le partage de l'accès au médium radio au niveau de la couche MAC (*Medium Access Control*) et d'autre part sur le routage des paquets de données au niveau de la couche réseau. La couche MAC a un impact considérable sur la consommation énergétique globale d'un réseau de capteurs, notamment via la mise en œuvre des stratégies d'allocation du canal de communication et de mise en veille du module radio. La couche réseau a également un rôle important puisqu'elle est en charge de sélectionner un chemin efficace du point de vue de l'énergie consommée pour acheminer les données jusqu'au nœud destination. Ces deux couches sont interdépendantes car certains paramètres gérés par une couche peuvent avoir des répercussions sur l'autre du point de vue de la consommation énergétique.

Le présent manuscrit de thèse est organisé de la façon suivante :

- Le Chapitre 2 décrit le contexte des réseaux de capteurs sans fil en présentant les caractéristiques et défis de ce type de réseaux. De plus, le fonctionnement et l'architecture générale de ce type de réseaux est présenté.
- Le Chapitre 3 propose un état de l'art des principaux protocoles utilisés dans les

réseaux de capteurs pour le contrôle de l'accès au médium radio et le routage, en présentant le principe de fonctionnement pour chacun des protocoles. Nous mettons en avant les principales approches mises en œuvre pour réduire la consommation en énergie.

- Le Chapitre 4 est dédié à la modélisation du médium radio et la définition d'un problème spécifique d'ordonnancement des transmissions pour éviter les collisions lors de la collecte des données provenant des nœuds capteurs. Après l'étude de la complexité du problème, nous proposons une approche distribuée et tolérante aux fautes transitoires calculant un ordonnancement des transmissions dont l'objectif est de maximiser le nombre de transmissions simultanées sans collision.
- Le Chapitre 5 se focalise sur le routage des paquets avec le protocole standardisé RPL (*Routing Protocol for Low-power and lossy networks*) dans le contexte des réseaux de capteurs. Nous présentons le fonctionnement de RPL puis nous effectuons une évaluation comparative des performances atteintes par plusieurs métriques de routage proposées dans la littérature pour ce protocole suivant plusieurs critères : l'énergie consommée, le taux de transmission de paquets, le délai de transmission de bout en bout et le nombre de messages de contrôle échangés dans le réseau.
- Le Chapitre 5 est consacré à la présentation de deux nouvelles propositions de métriques de routage RPL que nous faisons pour améliorer les performances du protocole en terme de durée de vie du réseau, de taux de perte de paquets, et de délai de transmission des données. La première proposition est une métrique bicritères tenant compte de l'énergie résiduelle des nœuds et de la qualité des liens de communication du réseau, alors que la seconde proposition est une métrique composite combinant deux métriques RPL existantes dans la littérature pour prendre également en compte le délai d'acheminement des paquets de données. Les gains en performance atteints par ces deux nouvelles métriques sont comparés à celles obtenues par les métriques considérées au Chapitre 5.
- Finalement, nous donnons une conclusion générale portant sur l'ensemble des travaux de cette thèse, ainsi que des perspectives pouvant faire suite à nos travaux à court, moyen et long termes.

## Chapitre 2

# Introduction aux réseaux de capteurs sans fil

### 2.1 Introduction

Depuis une dizaine d'années, les réseaux de capteurs sans fil (RCSF) se sont répandus pour répondre à de nouveaux besoins introduisant de nouvelles contraintes à prendre en compte, par exemple le délai d'acheminement, les quantités croissantes de données à traiter ou encore la mobilité. Dans ce contexte, l'objectif de ce premier chapitre est de présenter le périmètre des réseaux de capteurs sans fil en dressant une synthèse des caractéristiques de ce type de réseaux, de leur déploiement et de leur domaine d'application.

Ce chapitre est composé de trois sections, nous commençons par donner une définition d'un réseau de capteurs sans fil et ses caractéristiques. Ensuite nous décrivons quelques domaines et défis d'application des RCSF. Enfin nous présentons la norme IEEE 802.15.4 au niveau de la couche MAC (Media Access Control) ainsi que la norme ZigBee au niveau de la couche réseau, suivies par une partie architecture et pile protocolaire dans les réseaux de capteurs sans fil.

### 2.2 Définition et caractéristiques des RCSF

#### 2.2.1 Définition

**Un réseau de capteurs sans fil (RCSF)** est un ensemble de dispositifs électroniques appelés capteurs ou nœuds capteurs autonomes en mesure de relever et de renvoyer

## 2.2. DÉFINITION ET CARACTÉRISTIQUES DES RCSF

---

une grandeur physique à partir de son environnement d'application (température, taux de pollution dans l'air, niveau de glucose dans le sang, etc...). En raison de l'absence d'une infrastructure physique formant le réseau cœur (backbone) pour le routage des messages, les nœuds capteurs s'auto-organisent pour construire et former ce réseau cœur multi-sauts communicant avec des liaisons sans fil, capable de transférer l'information d'un capteur vers un autre jusqu'à une station de base spécifique appelée puits ou racine (Sink). Ce réseau est formé grâce au routage multi-sauts vers le nœud puits, qui est considéré comme la station de collecte des données envoyées par les nœuds capteurs. Doté d'une source énergétique non épuisable, et d'une grande capacité de calcul et de stockage, le nœud puits sert de passerelle entre le réseau de capteurs et le centre de traitement des données recueillies (voir Figure 2.1). La distribution des nœuds capteurs n'est pas nécessairement prédéterminée, ils peuvent être déployés aléatoirement ou positionnés de façon précise, tout dépend de l'environnement d'application et des objectifs attendus.

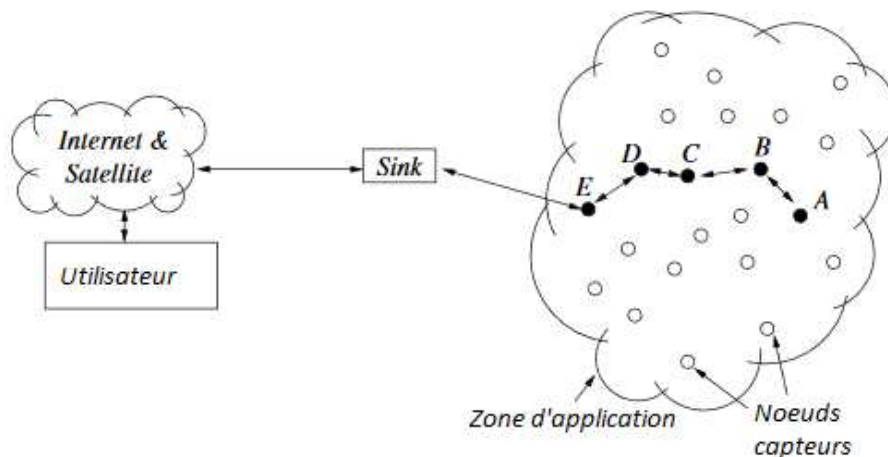


FIGURE 2.1 – Architecture RCSF [Akyildiz and Vuran 2010]

Un **nœud capteur** est un appareil électronique (Figure 2.2) composé principalement de 4 unités :

- **Une unité d'acquisition** : c'est l'unité principale du nœud capteur. Elle est constituée d'un ou plusieurs capteurs (pour diverses mesures physiques) et d'un convertisseur analogique numérique. Cette unité est caractérisée par un rayon d'acquisition qui définit la distance maximale d'acquisition d'une mesure physique. Les capteurs

## 2.2. DÉFINITION ET CARACTÉRISTIQUES DES RCSF

---

ont pour mission de relever des informations sur leur environnement, puis celles-ci seront ensuite converties et transmises à l'unité de traitement.

- **Une unité de traitement** : cette unité est constituée d'un micro-processeur et d'une mémoire incluant un système d'exploitation spécifique (ContikiOs, TinyOs, FreeRTOS, ...) adapté aux limitations matérielles des nœuds capteurs. Elle fait l'interface entre l'unité d'acquisition et l'unité de communication, et se charge d'assurer un contrôle intelligent du nœud capteur. Une fois que les informations sont récupérées de l'unité d'acquisition elle se charge de les analyser, les traiter puis les envoyer à l'unité de communication.
- **Une unité de communication** : elle est constituée d'un module radio permettant de réaliser des transmissions sans fil. Cette unité gère toute communication entrante ou sortante du nœud capteur (émission ou réception) sur le support radio fréquence (RF) pour les réseaux de capteurs sans fil.
- **Une unité d'alimentation** : elle représente l'unique source d'énergie du nœud capteur, typiquement une batterie ou une pile. Cette unité doit être en mesure d'assurer les besoins énergétiques de chaque unité citée précédemment.

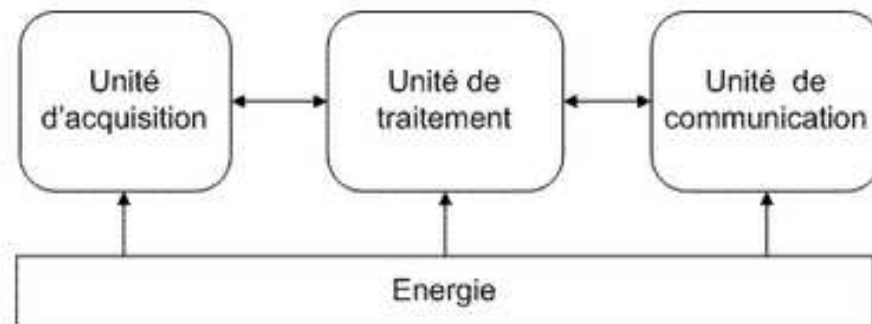


FIGURE 2.2 – Composants d'un nœud capteur

### 2.2.2 Caractéristiques des RCSF

Doté d'un espace de stockage limité, d'un faible débit pour les communications, d'une faible puissance de calcul, et généralement de peu d'autonomie énergétique, les nœuds capteurs sont très contraints en ressources matérielles. Ceci est dû au faible coût de fabrication mais aussi à la petite taille de ces composants. La force d'un réseau de capteurs provient



## 2.2. DÉFINITION ET CARACTÉRISTIQUES DES RCSF

Motes	MicaZ	Telos	Imote2	WSN430	Wavenis (Wavefront)
Processeur	Atmel AT-Mega 128L	TI MSP430	Intel PXA271 XScale	TI MSP430	TI MSP430
Vitesse du processeur	16 MHz	8 MHz	13 - 416 MHz	8 Mhz	8 MHz
RAM	4 Ko	2 Ko / 10 Ko	256 Ko	10 Ko	2 Ko
Espace programme	128 Ko	60 Ko / 48 Ko	32 Mo	48 Ko	128 octets
Flash	512 Ko	256 Ko	32 Mo	1 Mo	Néant
Communication série	UART	DIO, SPI, I2C, UART	UART, GPIO	DS2411	SPI, E2P
Batterie	2xAA	2/3A	3x AAA	PoLiFlex	2xAA
Voltage	2.7 V	1.8 - 3.6 V	3.2 - 4.5 V	2.2 V	2.4 - 6 V
Radio	TI CC2420 802.15.4	TI CC2420 802.15.4	TI CC2420	TI CC1100	ASIC RF Wavenis
Fréquence (MHz)	2400-2483	2400-2483	2400-2483	315/433/868/915	433/868/915
Débit de données (Kb/s)	250	250	500	250	4,8 - 153
Dimension (mm)	58x32x7	13x26x5	36x48x9	65x40x8	26x20x4.5

FIGURE 2.3 – Différentes caractéristiques des capteurs

de la collaboration de l'ensemble des nœuds capteurs pour remplir les tâches à accomplir. Par ailleurs suivant l'environnement d'applications des réseaux de capteurs sans fils, les nœuds capteurs disposent de caractéristiques matérielles différentes concernant la puissance du processeur (capacité de calcul), la taille de la mémoire (capacité de stockage), l'autonomie de la batterie, la taille des capteurs. Le tableau 2.3 présente ces caractéristiques pour certains types de capteurs, comme par exemple le MicaZ, Telos, Imote2, WSN430 ou Wavenis.

Par conséquent, en plus des caractéristiques matérielles, le choix d'utilisation d'un type de nœuds pour une application donnée repose sur les contraintes de l'application qui peuvent être influencées par plusieurs paramètres listés ci-dessous :

1. **L'environnement d'exploitation** : l'accessibilité de la zone d'application ou encore la présence d'éléments perturbateurs (pollution, mur...).
2. **La taille du réseau** : le nombre de capteurs peut varier d'une dizaine de capteurs à une centaine. La topologie et l'architecture obtenues peuvent impacter les performances d'un type de nœud capteur à un autre.

3. **La gestion d'énergie** : c'est le critère principal des réseaux de capteurs sans fil. Ainsi, la différence entre la technique utilisée d'un type de nœud capteur à un autre va influencer la durée de vie du réseau.
4. **La portée de communication** : la grandeur de la zone de couverture ou rayon de communication peut favoriser un type de capteur précis. Elle est bornée entre une valeur minimale et une valeur maximale.
5. **Le coût** : le prix d'un capteur varie de quelques euros à plusieurs dizaines d'euros.

### 2.2.3 Catégories des RCSF

En se basant sur le type d'environnement d'application des réseaux de capteurs sans fil, et en se référant à la littérature [Rawat et al. 2014a], [Yick et al. 2008], on peut distinguer 5 catégories d'application des réseaux de capteurs sans fil qui sont :

1. **Réseaux de capteurs terrestres sans fil** : les nœuds capteurs sont déployés sur terre dans une zone géographique précise. Ils se chargent de renvoyer une mesure physique vers le nœud racine ou sink. Le nœud racine a pour rôle de traiter cette mesure reçue pour conclure l'état de cet environnement ou déclencher une action de régulation par exemple. Il existe un nombre important d'applications dans cette catégorie, telles que la surveillance de la variation du climat avec les capteurs météorologiques ou la prévention de catastrophes naturelles (tremblement de terre, feu de forêt, tornade) ...
2. **Réseaux de capteurs souterrains sans fil** : les nœuds capteurs sont déployés sous terre dans des mines, tunnels de métro ou sous sol. Cette catégorie de réseaux a permis de révolutionner champs d'applications souterraines vu la simplicité de mise en œuvre, la densité de couverture et la facilité de dissimulation. De nombreuses applications ont vu le jour telles que la surveillance des champs agricoles en terme de concentration de sels minéraux, ou de présence de bactérie néfaste pour la production, ou encore la détection de fuites d'eau dans les tunnels de métro ...
3. **Réseaux de capteurs sous-marins sans fil** : les nœuds capteurs sont déployés sous l'eau dans des zones aquatiques ou océaniques. Les nœuds capteurs sous-marins sont extrêmement coûteux et sont déployés en petit nombre. La principale fonction de

cette catégorie est la collecte d'informations pour une navigation assistée des véhicules sous-marins ou encore la surveillance de la pollution des océans ou autres catastrophes naturelles. La communication entre les nœuds capteurs sans fil sous-marins se fait par transmission d'ondes acoustiques.

4. **Réseaux de capteurs corporels sans fil** : les nœuds capteurs sont déployés à l'intérieur du corps, sur le corps, ou près du corps à des endroits précis et en nombre limité. L'évolution de la nano-électronique et de la science ont conduit à une variété importante de traitements et d'applications de prise en charge et de suivi des patients à distance . Plusieurs projets collaboratifs ont été menés par des industriels et des universitaires, tel le projet Soli de Google [Google 2015] qui a pour objectif de développer des capteurs de détection de mouvements de la main et des doigts pour interagir avec les objets connectés mis sur le corps telle qu'une montre connectée. On peut citer aussi différents exemples d'applications comme les capsules endoscopiques (gastro-intestinale), ou encore les capteurs de stimulation cérébrale.
5. **Réseaux de capteurs multimédia sans fil** : grâce à l'évolution des technologies CMOS (Complementary Metal Oxide Semi-conductor), les nœuds capteurs sont équipés de vidéo, audio et image. Par conséquent, les réseaux de capteurs multimédia sans fil sont aujourd'hui utilisés pour assurer la vidéo surveillance dans divers domaines d'applications en se basant sur la collecte et l'analyse des images reçues. Les défis dans ce type d'applications sont le délai de transmission du flux multimédia, la qualité des images et la gestion de la consommation énergétique.

## 2.3 Domaines et défis d'applications des RCSF

### 2.3.1 Domaines d'application des RCSFs

L'utilisation des réseaux de capteurs sans fil a pris beaucoup d'ampleur depuis les dix dernières années. Ce type de réseaux a révolutionné divers domaines d'application, tant le domaine militaire, que le médical ou le domaine industriel. En s'appuyant sur certaines références telles que [Akyildiz and Vuran 2010] [Rawat et al. 2014b], nous décrivons ci-dessous les principaux domaines d'utilisation des réseaux de capteurs sans fil :

### 2.3. DOMAINES ET DÉFIS D'APPLICATIONS DES RCSF

---

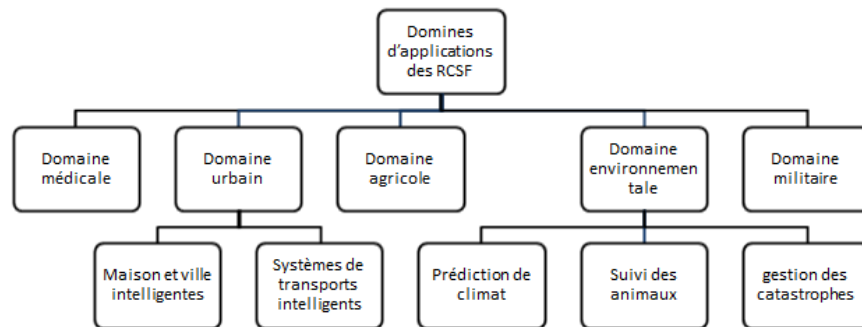


FIGURE 2.4 – Domaines d'applications des RCSF

- **Domaine militaire** : c'est l'un des premiers domaines d'application des réseaux de capteurs sans fil. Divers projets de recherches et d'expérimentations ont été menés dont l'objectif est de surveiller des zones difficiles d'accès, protéger les villes contre des attaques, aider les unités militaires dans un champ de bataille, suivre en temps réel le déplacement des armées, surveiller une zone militaire ... Dans ce type d'applications, les capteurs sont déployés parfois dans des environnements hostiles difficiles d'accès de façon aléatoire ou au contraire les capteurs sont positionnés sur des points stratégiques dans l'intérêt de mieux analyser les données rapportées au centre de traitement.
- **Domaine environnemental** : les réseaux de capteurs dans ce domaine d'application forment un système de détection et de prévention. De nos jours, il devient possible d'alerter les secours avant le déclenchement d'un feu en milieu forestier grâce à des relevés périodiques (température, humidité...). Dans le même principe, on peut utiliser un réseau de capteurs pour évaluer l'indice de pollution et analyser la qualité de l'air, cela concerne principalement les environnements urbains. On utilise aussi les réseaux de capteurs pour le comptage d'espèces animales sauvages en voie de disparition. Il devient ainsi possible d'observer le processus de développement de l'environnement, pour une meilleure analyse des facteurs problématiques dans l'environnement. Ces applications nous permettent non seulement d'améliorer ce processus de développement, mais de proposer des solutions plus efficaces et des moyens de lutte contre certaines catastrophes.

- **Domaine médical** : grâce à la taille réduite des nœuds capteurs, ces derniers peuvent être avalés ou implantés sur le patient pour effectuer des mesures physiologiques et assurer une surveillance permanente des organes vitaux (taux de cholestérol, tension artérielle, battements du cœur, température...), suivre l'impact du traitement, ou transmettre des images depuis l'intérieur du corps. Ces capteurs peuvent aussi être utilisés pour prévenir des comportements anormaux tels que la chute du patient, le cri subit ou un certain nombre d'informations précieuses qui pourront apporter un gain de temps et une réactivité plus adaptée...
- **Domaine industriel** : les applications des réseaux de capteurs dans le domaine industriel sont aussi nombreuses. L'objectif principal de ces applications est la surveillance du processus de fabrication pour assurer la bonne qualité du produit et le contrôle de la chaîne de production. D'autre part, ces capteurs peuvent aussi être utilisés pour le processus de livraison grâce aux capteurs de localisation ou pour le suivi de la consommation énergétique dans les immeubles. Ces applications nous permettent non seulement de maîtriser le processus de production, mais aussi de développer des moyens plus efficaces de lutte contre les corruptions industrielles.
- **Domaine domotique** : dans ce domaine, un réseau de capteurs est déployé dans l'habitation dans l'objectif d'assurer le confort, la sécurité des occupants et de concevoir des maisons intelligentes qui répondent aux besoins des habitants. La mesure et la régulation de la température, de la luminosité, de la fumée et de l'humidité sont les applications cibles de la domotique. Dans chacune de ces applications le comportement des capteurs est le même. À titre d'exemple on peut citer le thermostat intelligent Nest de Google qui permet d'économiser l'énergie au sein du foyer. Grâce à ses capteurs de détection de présence et de température le thermostat Nest s'adapte à la météo du jour, et aux habitudes des occupants pour utiliser le minimum d'énergie.

### 2.3.2 Défis des applications RCSF

Malgré l'émergence des réseaux de capteurs sans fil dans divers domaines d'applications, plusieurs contraintes et défis doivent être pris en compte. Nous donnons ci-dessous les

principaux défis pour ce type de réseaux :

- **Consommation énergétique** : en contrepartie de la taille réduite des nœuds capteurs, ces derniers se retrouvent alimentés par des batteries à petite charge difficile ou presque impossible à recharger. Cependant la première contrainte à considérer lors de la conception d'une application basée sur les réseaux de capteurs est la consommation énergétique car moins les nœuds capteurs consomment, plus la durée de vie du réseau se prolonge dans le temps (voir la définition de durée de vie dans la section suivante). L'optimisation de la consommation énergétique d'un nœud capteur revient principalement à minimiser l'énergie dissipée lors des communications entre les nœuds capteurs du réseau. En effet plusieurs études ont montré que la phase de transmission et réception peut atteindre jusqu'à 50% de la consommation énergétique globale [Baronti et al. 2007]. Par conséquent, pour prolonger la durée de vie d'un réseau de capteurs sans fil, on doit non seulement limiter le nombre de transmissions et réceptions de données au niveau de chaque nœuds capteurs, mais aussi mettre le nœud capteur en mode sommeil le plus souvent possible pour minimiser sa période d'activité et garantir un faible duty cycle tout en assurant le bon fonctionnement du RCSF. Le duty cycle représente la proportion de la période active sur la durée totale d'un cycle (période active + période sommeil), donc plus faible est le duty cycle mieux est la conservation d'énergie dans le RCSF.  
éviter la réception des messages qui ne lui sont pas destinés.
- **Qualité de service (QoS)** : dans la plupart des applications des réseaux de capteurs sans fil, garantir la qualité de service revient à assurer un certain niveau de débit, de latence, et de précision dans les données collectées. A cause de la nature des canaux de communication (liens radio), des pertes et des erreurs de réception ou transmission peuvent surgir dans le réseau à cause des collisions ou des interférences avec d'autres nœuds voisins ou autres perturbations sur les canaux. Pour assurer la QoS, différents mécanismes protocolaires sont mis en place pour définir l'accès au canal sans fil.
- **Passage à l'échelle** : les applications des réseaux de capteurs sans fil doivent être capables de s'adapter au changement de taille du réseau (d'une centaine de nœuds

capteurs à plusieurs milliers). Le passage à l'échelle garantit une facilité d'intégration de nouveaux nœuds capteurs malgré le changement de densité et de topologie de communication de l'application réseau considérée.

- **L'auto-configuration** : les réseaux de capteurs sans fil multi-sauts nécessitent une topologie de routage dynamique afin de supporter le changement de comportement d'un nœud capteur, l'ajout ou la suppression d'un nœud capteur. Cette capacité d'auto-configuration assure le maintien du service rendu par l'application des réseaux de capteurs sans fil malgré la défaillance de certains nœuds.
- **Mobilité** : en plus des déplacements des nœuds capteurs dans le réseau, la mobilité de l'environnement d'application est également à prendre en compte. Lors de déplacements des nœuds capteurs, la topologie du réseau change partiellement ou parfois totalement ce qui induit souvent un changement sur le routage des données.
- **Tolérance aux pannes** : au-delà des contraintes matérielles qu'un nœud capteur peut avoir, les applications basées sur les réseaux de capteurs doivent gérer les défaillances et les pannes qui peuvent empêcher ou limiter le fonctionnement global du réseau. Le degré de tolérance aux pannes dépend du type et du domaine d'application du réseau en question.
- **Sécurité** : la communication radio entre les nœuds capteurs facilite les intrusions et les attaques des applications. Par conséquent, plusieurs techniques ont été développées pour augmenter le degré de sécurité et de confidentialité des données échangées, mais la faible capacité de stockage et de calcul font que l'implémentation de ces techniques est difficile à mettre en œuvre pour ces réseaux.
- **Hétérogénéité** : différents types de nœuds capteurs peuvent être déployés dans un même réseau pour une même application. De ce fait un réseau hétérogène est composé de deux types (ou plus) de nœuds qui n'ont forcément pas les mêmes caractéristiques matérielles et/ou logicielles. Cependant une communication sans fil et un échange de données entre ces différents types de capteurs doit être possible et conforme aux exigences des applications visées. Dans le cas contraire, le réseau sera dit *homogène* car il est composé d'un ensemble de nœuds capteurs de même type et avec les mêmes caractéristiques matérielles et logicielles.

**Définition de durée de vie dans les réseaux de capteurs sans fil :** il existe plusieurs définitions pour la durée de vie d'un réseau de capteurs, en général la durée de vie d'un RCSF définit la période de temps de fonctionnement cohérent des nœuds capteurs. Un fonctionnement cohérent peut correspondre à la période de temps nécessaire pour :

- couvrir la totalité de la zone d'application du RCSF,
- maintenir assez de connectivité,
- assurer un taux de perte de données inférieur à un certain seuil.

Dans [Yoo et al. 2010] et [Lee et al. 2014], la durée de vie est définie comme étant le temps écoulé entre l'instant de déploiement et le moment où la batterie du premier nœud capteurs du réseau est totalement épuisée. Quant à Alfieri et al. [Alfieri et al. 2007] ou Zhao et al. [Zhao and Gurusamy 2008], la durée de vie d'un RCSF représente le temps qui s'est écoulé entre le moment où le réseau commence à fonctionner correctement et le moment où le réseau ne peut plus garantir un certain niveau d'exigence de couverture ou de connectivité.

Dans nos travaux de recherches et les simulations faites dans les chapitres suivants, nous définissons la durée de vie du réseau comme le temps qui s'est écoulé entre l'instant de déploiement et l'instant où le premier nœud capteur a sa batterie totalement épuisée.

## 2.4 Architecture et pile protocolaire

### 2.4.1 Les standards et normes

Afin de gérer la compatibilité et d'assurer l'interconnexion des nœuds capteurs de différents constructeurs, un certain nombre de standards ont été proposés. Tout d'abord, la norme IEEE 802.15.4 définit au niveau des couches physiques et MAC l'utilisation du médium radio, d'autre part le standard ZigBee établit le fonctionnement à suivre au niveau du routage des messages et le développement d'applications pour les RCSFs. Ce système de communication est assuré par les différentes couches protocolaires illustrées par la figure 2.5. Les deux couches basses (Physique et MAC) de la pile protocolaire des RCSFs sont définies par la norme IEEE 802.15.4 et les autres couches (Réseau, et Application) sont définies par la norme ZigBee [Gutierrez et al. 2001] [Val et al. 2008].



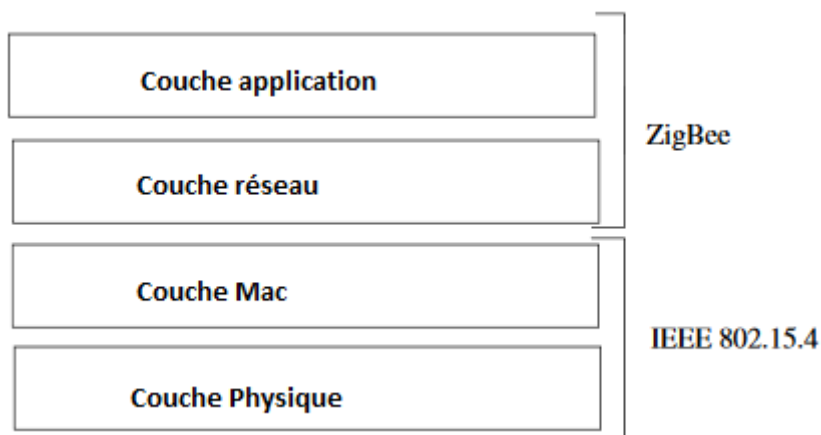


FIGURE 2.5 – Pile Protocolaire des réseaux de capteurs sans fil

Nous allons donner ci-dessous quelques précisions générales sur ces standards (le standard IEEE 802.15.4 sera détaillé dans le chapitre suivant) :

- **Le standard IEEE 802.15.4** : est le standard de développement considéré par la majorité des solutions pour réseaux de capteurs sans fil. La première version a été publiée en 2003 [Howitt and Gutierrez 2003], elle est conçue initialement pour des communications sans fil à courte portée WPAN (Wireless Personal Area Networks). Aujourd’hui, elle est utilisée dans les réseaux à faible débit appelés aussi LRWPAN (Low Rate Wireless Personal Area Network). La norme IEEE 802.15.4 définit les deux premières couches de la pile protocolaire des réseaux de capteurs sans fil, tenant compte des contraintes énergétiques et de la faible qualité des liens radio. Une nouvelle version nommée IEEE 802.15.4e apparaît en 2012 [De Guglielmo et al. 2016] afin de mieux répondre aux exigences des applications industrielles en terme de couverture, de latence et de fiabilité de transmission. Pour cela, la norme IEEE 802.15.4e introduit de nouveaux mécanismes d’accès au canal, tel que l’accès à intervalles de temps et la communication multi-canaux. Dans cette évolution du standard, cinq nouveaux protocoles MAC sont définis, notamment le protocole TSCH (Time Slotted Channel Hopping).
- **Le standard ZigBee** : ce standard est développé par la ZigBee Alliance [Al-

liance 2009] qui est un regroupement d'industriels leaders de la fabrication des semi-conducteurs et des technologies sans fil. Il est initialement conçu pour promouvoir le standard IEEE 802.15.4 en mai 2003. En s'appuyant sur les couches physique et MAC compatibles avec la norme IEEE 802.15.4, le standard ZigBee définit le fonctionnement des couches réseau et application de la pile protocolaire pour les réseaux de capteurs sans fil. Il a pour objectif de développer des protocoles à basse consommation énergétique adaptés aux réseaux de capteurs sans fil LP-WPAN (Low Power-Wireless Personal Area Network), ainsi que de répondre aux besoins du marché en terme de sécurité, de surveillance et de contrôle des communications sans fil.

- **Le standard 6LowPAN** : ce standard a été introduit pour interconnecter les nœuds capteurs avec tout type d'équipement connecté au réseau IPv6. Le groupe de travail IETF a adapté le protocole IPv6 au standard IEEE 802.15.4, cette adaptation a été dénommée 6LowPAN [Montenegro et al. 2007]. Jusqu'à présent, la majorité des déploiements des réseaux de capteurs sans fil est basée sur les standards ZigBee et 6LoWPAN. Les principales adaptations de 6LoWPAN sont :
  - La compression de l'entête du protocole IPv6 de 40 octets à 6 octets d'entête 6LoWPAN pour éviter le gaspillage d'énergie.
  - L'adaptation du mécanisme de découverte du voisinage d'IPv6 [Narten et al. 2007].
  - La fragmentation et la reconstitution des trames IPv6 qui utilisent initialement des trames de tailles minimales de 1280 octets, mais les réseaux de capteurs sont limités à 127 octets.

Jusque-là ces standards permettent des communications sans fil à courte portée avec des contraintes de latence, de quantité de données à transmettre, et de consommation énergétique. C'est notamment le cas d'applications pour les services de la ville intelligente qui nécessitent des communications sans fil de longue portée et de faible consommation énergétique. C'est ainsi que des solutions propriétaires sont apparues depuis les années 2000 de type LPWAN (Low-Power Wide-Area Networks), remplissant les critères de longue portée et de faible consommation énergétique. Les deux solutions propriétaires les plus connues

## 2.4. ARCHITECTURE ET PILE PROTOCOLAIRE

aujourd'hui sont Sigfox et LoRa [Lauridsen et al. 2017a,b]. Ces solutions révolutionnent le monde de communication machine à machine (M2M), elles permettent des transmissions de quelques centaines de mètres à plusieurs dizaines de kilomètres à une fréquence porteuse de 868 MHz avec la particularité d'être peu sensibles aux interférences. Dans ce type de technologie, le réseau est composé de plusieurs équipements sans fil qui communiquent directement avec une station de base ( passerelles) qui transmet à son tour ces messages au serveur de stockage et de traitement applicatif via une connexion IP. La topologie réseau est dite en étoile d'étoiles voir la figure 2.6, car les stations de base sont à la réception de messages provenant de plusieurs équipements sans fils, et chaque serveur de stockage est connecté à une multitude de station de base.

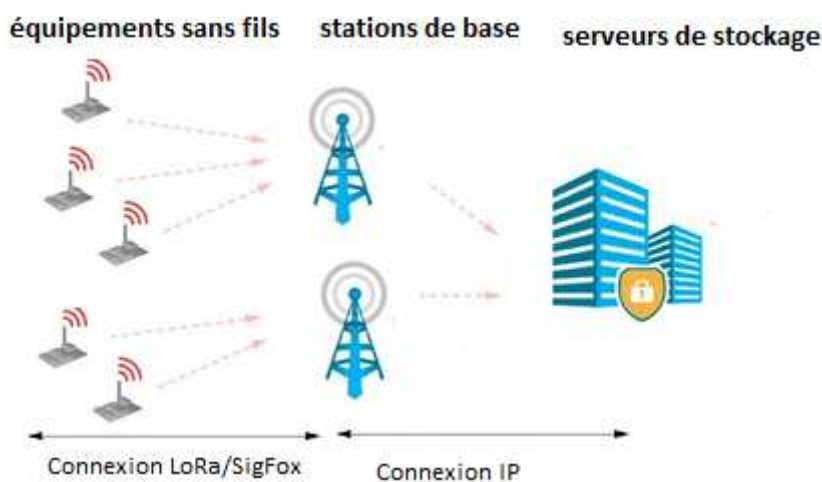


FIGURE 2.6 – Principe d'architecture LoRa/Sigfox

### 2.4.2 Architectures

Le déploiement des réseaux de capteurs sans fil en mode sans infrastructure nécessite une architecture de communication logique appelée aussi topologie de communication. Le standard IEEE 802.15.4 définit deux rôles. Dans un réseau de capteurs sans fil, un nœud capteur ne peut avoir que l'un des deux rôles ci-dessous au niveau de la couche MAC :

- **Le rôle RFD (Reduced Function Device)** : le nœud capteur est limité à des fonctionnalités restreintes et simples comme capter une mesure de son environne-

ment d'application. Dans ce rôle, le nœud est considéré comme un dispositif d'extrémité (ou feuille), il ne peut uniquement communiquer qu'avec un nœud de rôle FFD dans le réseau. Dans ce rôle, un nœud capteur ne prend pas en charge la fonction de routeur pour transférer les paquets d'un autre nœud du réseau.

- **Le rôle FFD (Full Function Device) :** le nœud capteur prend en charge toutes les fonctionnalités possibles dans le réseau. Ainsi en plus de capter une mesure, un nœud ayant le rôle FFD peut avoir la fonction de routeur ou de coordinateur pour relayer les paquets d'autres nœuds du réseau.

Dans le standard ZigBee, on trouve trois types de topologies : la topologie en étoile, la topologie en arbre et la topologie maillée (voir Figure 2.7). Dans chacune des topologies un nœud capteur peut avoir le rôle RFD (feuille) ou FFD (routeur ou coordinateur).

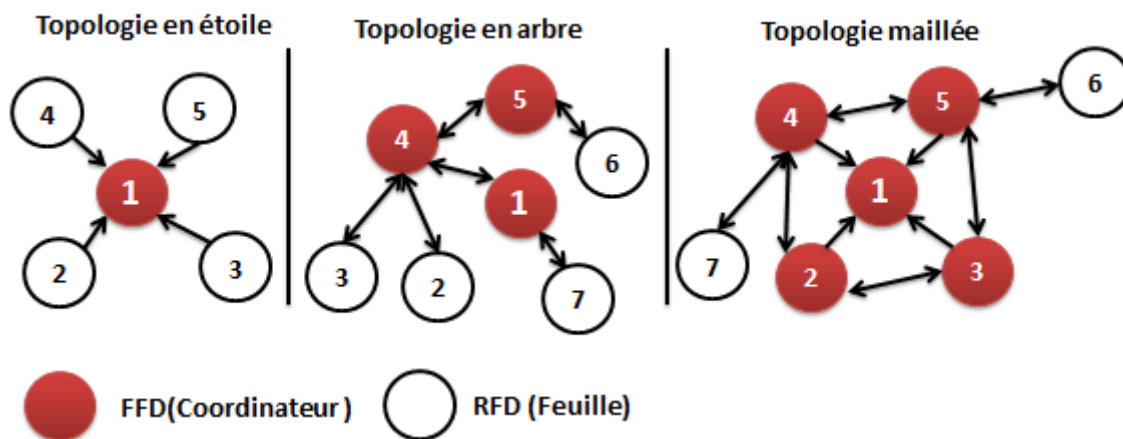


FIGURE 2.7 – Les topologies des réseaux de capteurs sans fil

Nous présentons ci-dessous chacune de ces topologies :

- **Topologie en étoile :** dans une topologie en étoile, aucun nœud capteur ne peut communiquer avec un autre nœud capteur à part le nœud coordinateur.
- **Topologie maillée (point-à-point) :** dans cette topologie les nœuds feuilles peuvent communiquer uniquement avec leur nœud coordinateur, tandis que les nœuds non feuilles peuvent communiquer avec n'importe quel autre nœud capteur dans leur voisinage.
- **Topologie en arbre :** dans cette topologie, on trouve un nœud principal qui a le

rôle de coordinateur et d'initiateur auquel sont rattachés des nœuds de rôle routeurs ou feuilles.

### 2.4.3 Pile protocolaire

Comme mentionné précédemment, un ensemble de protocoles sont mis en place sur chaque nœud capteur afin de permettre le fonctionnement du RCSF. Les caractéristiques des différentes couches liées aux RCSFs sont présentées ci-dessous :

1. **Couche physique** : elle est principalement conçue pour indiquer les caractéristiques du support physique de communication sans fil entre les nœuds capteurs du réseau, telles que le canal de transmission, la puissance du signal de transmission et le choix de la modulation. Nous définissons plus précisément ces trois caractéristiques :

- (a) **Canal de transmission** : le choix de la fréquence de communication des nœuds capteurs correspond au rôle de la couche physique. Cette couche prend en charge 27 canaux de transmission dans la bande de fréquences ISM (Industrielle, Scientifique et Médicale) répartis en 3 plages :

- 16 canaux dans la plage de fréquences de 2.4GHz à 2.4835GHz opérationnelle partout dans le monde avec un débit de 250kbit/s, la norme IEEE 802.15.4 favorise l'utilisation des canaux 15, 20 et 26.
- 10 canaux dans la plage de fréquence de 902 à 928MHz opérationnelle en Amérique du Nord avec un débit de 40kbit/s.
- Un canal dans la plage de fréquence de 868 à 870MHz opérationnelle en Europe avec un débit de 20kbit/s.

A ce jour, la majorité des déploiements utilisent la bande de fréquence 2.4GHz vue l'étendu de sa couverture, le nombre de canaux disponibles, ainsi que le débit maximal atteint.

- (b) **Puissance du signal** : la norme IEEE 802.15.4 ne définit pas la puissance maximale de transmission, celle-ci est à la charge du constructeur, en revanche la puissance recommandée est de 1mW équivalent à 0dBm et la sensibilité du nœud récepteur doit être inférieure à -85dBm. La puissance de transmission de 1mW dans la bande de fréquence 2.4GHz permet d'atteindre une portée théo-

rique d'environ 300m en extérieur, et moins de 100m en intérieur [Baronti et al. 2007]. La portée de communication dans les réseaux de capteurs est directement liée à la puissance d'émission. Cependant, une transmission de puissance faible cause des pertes de paquets transmis qui nécessitent ensuite une retransmission, induisant une consommation d'énergie supplémentaire. D'autre part, une transmission de puissance élevée aboutie à une transmission de données à faible taux de perte, mais à une consommation énergétique très importante.

(c) **Modulation** : la modulation définit le codage des bits à transmettre dans la forme du signal transmis. La couche physique spécifiée par la norme IEEE 802.15.4 définit différentes techniques de modulation, qui peuvent varier selon la bande de fréquences utilisée, par exemple la technique O-QPSK (Offset Quadrature Phase-Shift Keying) est utilisée comme modulation sur la bande 2.4GHz et la modulation Binary Phase-Shift Keying (BPSK) est utilisée sur la bande 868/915 MHz.

2. **Couche MAC** : cette couche permet de définir le protocole d'accès au canal de transmission pour gérer la concurrence d'accès au canal radio, ainsi que le format de la trame. La fonctionnalité majeure de cette couche est d'assurer une efficacité énergétique et d'éviter les collisions. Dans les réseaux de capteurs, la transmission radio est la plus grande source de dépense énergétique et l'activité radio est principalement gérée par la couche MAC. Afin d'économiser cette dépense énergétique la couche MAC doit limiter l'utilisation de la radio en la gardant éteinte le maximum de temps possible.

(a) **Protocole d'accès au canal** : le mécanisme d'accès au support radio est défini à ce niveau pour la transmission des données. Les protocoles MAC définissent les différentes phases d'établissement d'une transmission entre deux nœuds capteurs à portée de communication. Les protocoles MAC permettent de minimiser les pertes qui sont caractérisées par les collisions, le délai de transfert de données sur le canal radio et l'écoute active du canal de transmission. La norme IEEE 802.15.4 définit deux modes d'accès au canal radio : un mode non coordonné et un mode coordonné. Dans le mode non coordonné, le nœud capteur envoie

sa donnée dès que le canal radio est libre (CSMA/CA), alors que dans le mode coordonné la transmission des données est synchronisée par le nœud coordinateur (TDMA). Dans le chapitre suivant nous présenterons en détail ces différents modes d'accès au canal.

(b) **Format de la trame** : dans la norme IEEE 802.15.4, il existe quatre types de trames au niveau de la couche MAC qui assurent la communication entre deux nœuds capteurs à portée de communication :

- **Les trames Beacon** : elles sont envoyées uniquement par les nœuds coordinateurs pour synchroniser les transmissions de données.
- **Les trames de données** : ces trames sont utilisées pour le transfert des données entre les nœuds capteurs.
- **Les trames d'acquiescement** : elles sont utilisées pour confirmer la bonne réception des trames de données.
- **Les trames de contrôle** : elles sont utilisées pour effectuer des commandes spécifiques.

3. **Couche Réseau** : elle définit les règles de construction du réseau et des étapes d'association des nœuds dans un protocole de routage. Le protocole de routage définit les routes d'acheminement des données d'un nœud capteur jusqu'au nœud puits (racine) dans un réseau multi-sauts. Ceci est réalisé en considérant le ou les critères de sélection des liens. Le protocole de routage vise à sélectionner un chemin peu coûteux en énergie, en délai, et qui garantit la stabilité de la topologie. Suivant la technique de création et de maintien des routes lors de l'acheminement des données, les protocoles de routage peuvent être catégorisés en deux types : les protocoles proactifs et les protocoles réactifs. Dans le cas des protocoles proactifs, les routes d'acheminement sont établies à l'avance en se basant sur un échange périodique des tables de routage, contrairement aux protocoles réactifs où les routes sont établies à la demande. Nous les distinguons en détail dans le chapitre suivant.

4. **Couche Application** : cette couche permet d'assurer l'interfaçage avec les applications. Elle définit la manière dont vont être utilisées toutes les couches inférieures pour une application donnée. Cette couche garantit la compatibilité avec d'autres

capteurs proposés par des fabricants différents.

## 2.5 Conclusion

Dans ce premier chapitre nous avons présenté les caractéristiques principales des réseaux de capteurs sans fils sur lesquelles nous allons nous baser dans les chapitres suivants. Dans la première partie de ce chapitre, nous avons défini ce qu'est un réseau de capteurs sans fil ainsi que l'architecture matérielle d'un capteur, ensuite nous avons donné un aperçu de leurs défis et domaines d'applications. Enfin, nous avons abordé dans la troisième partie les différentes couches de la pile protocolaire du standards IEEE 802.15.4.

Dans le chapitre suivant, nous allons poursuivre l'état de l'art en nous intéressant plus particulièrement à l'étude des protocoles MAC et au routage conçu pour les réseaux de capteurs sans fils, en considérant la minimisation de la consommation énergétique et en garantissant les performances globales du réseau.



## Chapitre 3

# Méthodes d'accès au médium et routage pour les réseaux de capteurs sans fil

### 3.1 Introduction

Dans le chapitre précédent, nous avons présenté le contexte des réseaux de capteurs sans fil ainsi que leurs différents domaines d'application. Nous avons également cité les différents défis et exigences de ce type de réseaux. Pour faire face à ces exigences, plusieurs approches protocolaires ont été proposées dans la littérature. Dans ce chapitre, nous nous intéressons plus particulièrement au fonctionnement des protocoles liés au contrôle de l'accès au médium radio, ainsi qu'au routage dans le contexte des réseaux de capteurs. En effet, les protocoles mis œuvre dans ces deux couches influent majoritairement sur les performances atteintes dans ce type de réseaux.

Ce chapitre est composé de trois sections, dans la première section nous listons les principales causes de surconsommation d'énergie dans un réseau de capteurs. Nous décrivons ensuite les approches et protocoles proposés pour le contrôle d'accès au médium radio, puis nous finissons en présentant les principaux protocoles de routage utilisés dans les réseaux de capteurs sans fil.

### 3.2 Les principales sources de surconsommation d'énergie

La consommation énergétique est le critère principal à optimiser afin d'allonger la durée de fonctionnement du réseau. Cela est dû au fait que les nœuds de ce type de réseaux sont majoritairement alimentés par des batteries, car en général il est impossible voire trop coûteux de les alimenter via une source d'alimentation extérieure ou de recharger leur batterie. Il est donc crucial de mettre en place des stratégies économes en énergie.

L'énergie consommée sur chaque nœud capteur est répartie entre les différentes tâches exécutées sur celui-ci. Cependant, parmi ces tâches, l'émission et la réception de messages sont les plus consommatrices en énergie [Bal et al. 2009]. La Figure 3.1 illustre la consommation énergétique engendrée par les principales tâches exécutées sur les nœuds capteurs. Les émissions et réceptions de messages sont surtout liées aux couches de contrôle d'accès au médium radio et réseau. Il est donc essentiel de prendre en compte un certain nombre d'aspects associés à ces deux couches surtout au niveau MAC afin de limiter la consommation énergétique engendrée.

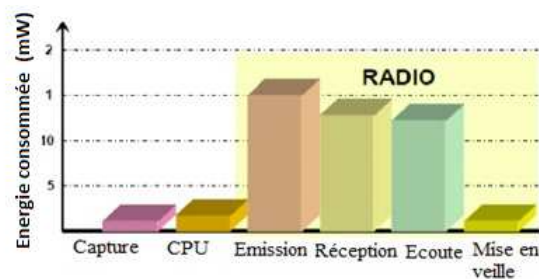


FIGURE 3.1 – Consommation énergétique liée aux tâches exécutées sur un capteur.

Nous allons décrire ci-dessous les causes majeures de surconsommation énergétique au niveau de la couche d'accès au médium :

1. **Les collisions de messages** : Ce phénomène se produit lorsqu'un nœud reçoit plusieurs messages simultanément. Les messages entrent en collision et il est difficile pour un nœud de décoder les messages envoyés sans mécanisme ou matériel particulier. Ces messages sont donc rejetés par le nœud destinataire et ils doivent être retransmis par les nœuds sources. Cela induit donc une surconsommation énergétique dans le réseau. Ce phénomène peut être illustré par le *problème du nœud caché* (voir la

## 3.2. LES PRINCIPALES SOURCES DE SURCONSOMMATION D'ÉNERGIE

---

Figure 3.2(a)).

Dans la Figure 3.2(a), le nœud B reçoit simultanément deux messages envoyés par les nœuds A et C, qui sont dans l'impossibilité de détecter une collision survenue au niveau de B. En effet, A et C ne se trouvent pas mutuellement dans le rayon de communication l'un de l'autre. Un autre phénomène, appelé *problème du nœud exposé* (illustré à la Figure 3.2 (b)), peut également survenir lorsque plusieurs nœuds trop proches les uns des autres désirent communiquer. Dans la situation illustrée par la Figure 3.2 (b), les nœuds A et C ont un message à transmettre respectivement au nœud B et D. Avant de transmettre un message, chaque nœud écoute le canal pour détecter s'il est libre. Le nœud A initie une communication vers B avant le nœud C, qui attend avant de transmettre son message pour éviter tout risque de collision. Cependant, la transmission du message par C ne crée pas de collision au niveau des nœuds D et B. Cette seconde situation est donc surtout gênante pour les performances dans le réseau.

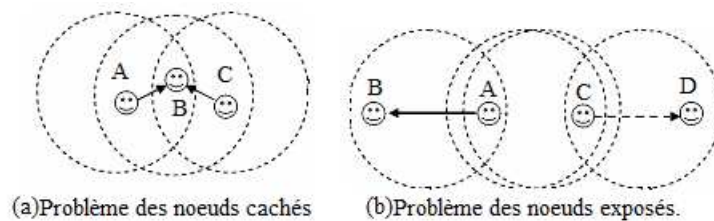


FIGURE 3.2 – Nœud caché et nœud exposé

2. **L'écoute active (overhearing)** : Cela se présente quand un nœud se met en écoute sur le canal de transmission pour recevoir des paquets qui ne lui sont pas destinés. De ce fait le nœud consommera de l'énergie en écoutant inutilement le canal. Cette situation peut être coûteuse en énergie dans le cas de réseaux denses et avec une charge de trafic importante.
3. **La surcharge (overhead)** : Cette situation est liée à un événement temporaire causant l'envoi d'un nombre de messages de contrôle anormalement élevés qui sont échangés pour la bonne transmission des données. Cela entraîne une surconsommation en énergie et une baisse du débit utile (ou effectif).

### 3.3. MÉTHODES D'ACCÈS AU MÉDIUM RADIO

---

4. **L'envoi sourd** : Ce phénomène se présente quand un nœud dans l'état actif envoie des données à un autre nœud dans l'état sommeil. Ce dernier n'est pas prêt pour recevoir les données qui lui sont transmises. Cela entraînera une surconsommation en énergie car le nœud émetteur devra renvoyer son message.

Après avoir mis en avant les principales sources de consommation d'énergie sur les nœuds d'un réseau de capteurs, nous allons nous intéresser dans la section suivante aux principales approches et protocoles mis en œuvre pour réduire la consommation énergétique au niveau de l'accès au médium radio.

### 3.3 Méthodes d'accès au médium radio

Comme nous l'avons souligné dans la Section 3.2, les protocoles au niveau de la couche d'accès au médium influencent significativement l'énergie consommée, car ils contrôlent l'activité radio des différents nœuds capteurs du réseau. Les protocoles au niveau de cette couche permettent d'assurer les services suivants :

1. **Partage de l'accès au médium** : la couche MAC contrôle l'accès au médium radio de sorte à partager efficacement le ou les canaux disponibles entre les nœuds du réseau.
2. **Gestion de la mise en veille des capteurs** : afin de réduire l'énergie consommée par les nœuds capteurs et ainsi prolonger la durée de vie du réseau, la couche MAC définit les périodes d'activité et de mise en veille de l'ensemble des nœuds capteurs du réseau.
3. **Structuration des trames** : la couche MAC définit le format des trames échangés à travers le médium radio.
4. **Détection d'erreur** : une analyse basée sur une somme de contrôle est effectuée afin de détecter d'éventuelles erreurs introduites par le canal radio lors de la retransmission des trames. En cas de détection d'erreur, la trame erronée sera retransmise.

Plusieurs approches ont été conçues pour partager au mieux un même médium de communication entre plusieurs utilisateurs. Nous décrivons ci-dessous le fonctionnement des principales méthodes d'accès que nous pouvons retrouver dans de nombreux protocoles :

### 3.3. MÉTHODES D'ACCÈS AU MÉDIUM RADIO

---

- **Méthode d'accès CSMA (Carrier Sense Multiple Access)** : Cette première méthode est une approche optimiste proposée pour la première fois en 1975 pour éviter les collisions. En effet, avec CSMA [Karl and Willig 2007] chaque utilisateur désirent communiquer doit attendre un temps aléatoire, appelé période de « *backoff* », avant de tenter de transmettre ses données. À la fin de cette période d'attente, l'utilisateur écoute le médium de communication pour déterminer si celui-ci est occupé, c'est-à-dire si des données sont en cours de transmission par un autre utilisateur. Plus spécifiquement, cette opération dite *Clear Channel Assessment* (CCA) évalue si la puissance du signal reçu est supérieure au seuil du canal bruité. Si c'est le cas on considère que le canal est occupé, sinon on considère que le canal est libre. Dans le cas où le canal est occupé, l'utilisateur devra attendre durant une nouvelle période aléatoire avant de tenter ultérieurement d'envoyer ses données, sinon le médium est disponible et il tente de transmettre ses données. Dans le contexte des réseaux de capteurs, l'utilisateur pourra passer en mode sommeil pour économiser sa batterie. Cette phase d'endormissement est appelée *Collision Avoidance* (CA), elle permet de réduire la probabilité d'apparition d'une collision lors de la transmission de données sur le médium partagé. Cette procédure d'évitement ne peut garantir des transmissions sans collisions car il est possible que deux utilisateurs ayant la même période d'attente émettent leurs données respectives au même moment sur le médium.
- **Méthode d'accès TDMA (Time Division Multiple Access)** : La méthode TDMA [Karl and Willig 2007] partage le médium de communication en répartissant son utilisation par les utilisateurs dans le temps. Le temps est divisé en intervalles réguliers nommés « slots » regroupés par bloc de N slots de façon cyclique. Dans chaque bloc, un slot est affecté à un utilisateur lui permettant ainsi périodiquement d'accéder au médium et d'utiliser toute la bande passante allouée pour envoyer ou recevoir des données. L'avantage principal de la méthode d'accès TDMA est sa faible consommation en énergie dans un réseau de capteurs, en effet il est possible de mettre en sommeil les capteurs durant les slots qui ne leur sont pas attribués [Cionca et al. 2008].
- **Méthode d'accès FDMA (Frequency Division Multiple Access)** : Contrai-

### 3.3. MÉTHODES D'ACCÈS AU MÉDIUM RADIO

---

rement à TDMA qui alloue toute la bande passante à chaque utilisateur pendant un temps limité, la méthode FDMA s'attache à diviser la bande passante en plusieurs canaux. Chaque nœud peut communiquer en utilisant un canal distinct par rapport aux autres utilisateurs. Ceci permet aux utilisateurs de communiquer simultanément sans perturber les transmissions effectuées par les autres utilisateurs. Contrairement à l'approche TDMA avec une planification dans le temps, l'approche FDMA minimise donc les problèmes de collision via une planification de l'utilisation des canaux radio par chaque utilisateur. Pour mettre en œuvre FDMA, les utilisateurs doivent être équipés d'un système radio capable de recevoir des signaux provenant de plusieurs canaux.

- **Méthode d'accès CDMA (Code-Division Multiple Access) :** Dans cette dernière méthode, introduite par Ageev Ageev [1935] en 1935, chaque utilisateur dispose d'un code différent lui permettant d'encoder (moduler) le signal représentant les données à transmettre. Cet encodage est généralement réalisé en effectuant une opération XOR entre les signaux de données et de code. Le récepteur du signal encodé pourra extraire le signal de données en utilisant le code dont il dispose localement en appliquant la même opération que pour le codage. Si le code correspond il pourra extraire les données, sinon un signal proche de zéro sera obtenu. Cette méthode atteint de bonnes performances uniquement si les différents codes utilisés par les utilisateurs sont orthogonaux, sans quoi les signaux de données extraits seront faussés ou incomplets. Le point crucial est donc la génération et la sélection des codes à utiliser.

Après avoir décrit le principe de fonctionnement des méthodes d'accès les plus utilisées, nous allons présenter dans le reste de cette section les principaux protocoles proposés dans le contexte des réseaux de capteurs. Pour cela, nous commençons par décrire le fonctionnement général du protocole MAC défini par le standard IEEE 802.15.4, majoritairement mis en œuvre par les constructeurs pour la communication dans les réseaux de capteurs sans fil.

#### 3.3.1 La couche MAC de la norme IEEE 802.15.4

La première version du protocole 802.15.4 a été standardisée par l'organisme IEEE en 2003 Gutierrez et al. [2003]. Ce standard définit les spécifications des couches physiques et de la sous-couche MAC pour les réseaux de capteurs sans fil. Dans la suite, nous nous concentrerons sur la partie concernant la sous-couche MAC. Deux modes de fonctionnement sont spécifiés pour l'accès au médium : le mode *CSMA non « sloté »* ou *CSMA « sloté »*. Dans le premier cas, le protocole défini par le standard fonctionne sans l'envoi de « Beacon ». Un Beacon est une trame courte envoyée par les coordinateurs du réseau afin de synchroniser les nœuds. À l'inverse, dans le mode CSMA sloté les coordinateurs envoient périodiquement des Beacons afin de synchroniser les nœuds du réseau et leur annoncer le début d'une « superframe » utilisée pour organiser les échanges de données. Dans le mode de fonctionnement CSMA non sloté, l'accès au médium suit la méthode CSMA décrite précédemment. Dans la suite, nous nous focaliserons plus particulièrement sur le fonctionnement du mode CSMA sloté car une partie reprend le fonctionnement du premier mode.

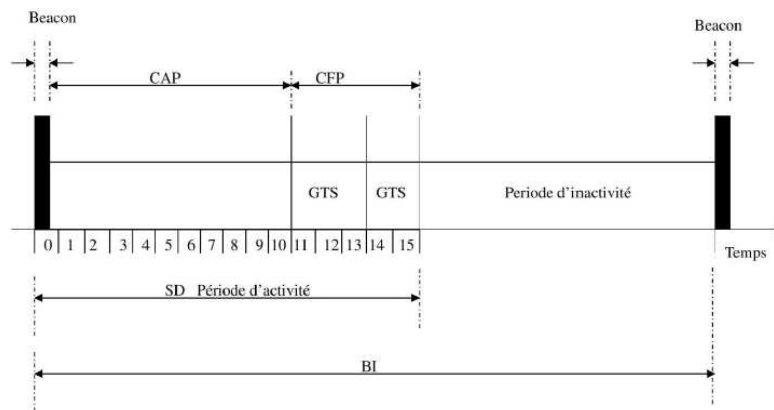


FIGURE 3.3 – Structure d'une superframe dans la norme IEEE 802.15.4.

Dans le mode CSMA sloté, la durée de la superframe est définie par le paramètre *SD* (*Superframe Duration*), ainsi que par le paramètre *BI* (*Beacon Interval*) indiquant le temps séparant deux Beacons successifs. La durée SD correspondant à la longueur de la superframe définit la période d'activité des nœuds, c'est durant cette période que les trames seront transmises. Le temps compris entre la fin d'une superframe et le prochain

### 3.3. MÉTHODES D'ACCÈS AU MÉDIUM RADIO

---

Beacon correspond à une période d'inactivité, où les nœuds du réseau pourront être mis en sommeil pour économiser leur batterie. Cela est illustré par la Figure 3.3. La période active est découpée en 16 slots de durée fixe. Tout d'abord, le slot n°0 est réservé pour l'envoi du Beacon par le coordinateur. Durant les 15 slots suivants, deux méthodes d'accès seront utilisées par les nœuds. Une méthode d'accès CSMA sera utilisée durant les dix premiers slots. Durant cette période, appelée *CAP (Contention Access Period)*, les nœuds seront en compétition pour l'accès au médium sans fil afin d'envoyer des trames de contrôle au coordinateur. Tandis que durant les cinq slots restants, l'accès au médium sans fil est effectué suivant une approche TDMA pour envoyer ou recevoir des trames de données avec le coordinateur. Pendant cette dernière période, appelée *CFP (Contention Free Period)*, chaque coordinateur attribue un slot à un nœud désirant envoyer une trame de données. La Figure 3.4 indique les types d'accès dans chaque mode de fonctionnement que nous venons de présenter rapidement.

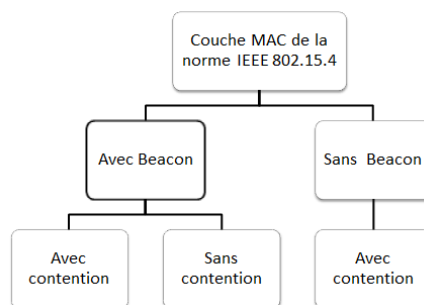


FIGURE 3.4 – Modes de fonctionnement de la sous-couche MAC de la norme IEEE 802.15.4.

Nous allons expliquer plus en détail le fonctionnement suivi durant la période d'activité avec contention (*CAP*) et sans contention (*CFP*). Pendant la période *CAP*, l'accès au médium durant chaque slot suit une approche CSMA. Ainsi, la norme spécifie que chaque nœud utilise un algorithme de tirage d'un temps d'attente aléatoire, appelé *Binary Exponential Backoff (BEB)*, pour réduire la probabilité de collisions. Ainsi, avant toute transmission chaque nœud initialise les trois paramètres suivants :

- Number of Backoff (NB) : ce paramètre indique le nombre de fois qu'un nœud a attendu durant une période aléatoire, initialisé à 0.
- Contention Window (CW) : cela représente le nombre de périodes d'attentes pour



### 3.3. MÉTHODES D'ACCÈS AU MÉDIUM RADIO

---

scruter si le médium est libre, initialisé à 2.

- Backoff Exponent (BE) : cette valeur définit l'exposant pour le tirage du temps d'attente aléatoire, initialisé à 3 par défaut.

Après avoir initialisé ces paramètres, un nœud tire un temps d'attente aléatoire compris dans l'intervalle  $[0, 2^{BE} - 1]$ . À la suite de ce temps d'attente, le nœud effectue deux tests (appelé *Clear Channel Assessments (CCA)*) successifs pour déterminer l'état du médium sans fil. Le nombre de tests est déterminé par le paramètre CW, si après une scrutation du médium celui-ci est occupé le paramètre CW est réinitialisé à 2, l'exposant BE est incrémenté de 1 (jusqu'à une valeur maximale définie par *macMaxBE*) ainsi que le paramètre NB. Puis, le nœud attend durant un nouveau temps tiré aléatoirement dans l'intervalle  $[0, 2^{BE} - 1]$ . Ainsi, tant que CW n'est pas égal à 0 le nœud continuera à scruter le médium. Un nœud ne pourra donc commencer à transmettre sa trame que si le médium est libre durant deux scrutations successives. Lorsqu'un nœud peut transmettre une trame, le paramètre BE est réinitialisé à 3, et le nœud attend un acquittement de la part du destinataire de sa trame. Si aucun acquittement n'est reçu, le nœud tente de retransmettre sa trame à nouveau en suivant la procédure décrite juste avant, un nombre maximum de fois défini par le paramètre *macMaxFrameRetries*. Si ce nombre maximum de tentatives est atteint, la trame est rejetée.

Enfin, une fois la période CAP écoulee la période CFP démarre. Pendant cette période, les nœuds autorisés par le coordinateur peuvent transmettre leur trame de données durant le slot qui leur a été attribué. Pendant ces slots, une approche TDMA est suivie et chaque nœud utilise toute la bande passante.

Suite à la présentation du fonctionnement général de la norme IEEE 802.15.4, nous allons discuter des avantages et inconvénients de celle-ci. Le protocole spécifié par l'IEEE est adapté au contexte des réseaux de capteurs sans fil. Tout d'abord, les nœuds peuvent se mettre en sommeil en attendant l'opportunité de pouvoir transmettre. De plus, la mise en sommeil des nœuds permet de réduire du même coup la probabilité de collisions. Cela est renforcé par les deux tentatives successives que doit réaliser chaque nœud avant de pouvoir transmettre sa trame. En effet, cela permet par exemple de transmettre son acquittement durant cet intervalle de temps d'attente et éviter des collisions. Cependant, ce protocole

### 3.3. MÉTHODES D'ACCÈS AU MÉDIUM RADIO

---

possède quelques défauts. L'un des défauts majeurs concerne l'algorithme BEB qui est sans mémoire entre deux exécutions. En effet, cela vient du choix de l'exposant BE qui est aléatoire et ne tient pas compte du contexte (nombre de nœuds voisins, le nombre de trames à transmettre ...). De plus, cet exposant est incrémenté à chaque fois de la même manière par un nœud qui a une trame à transmettre, malgré le nombre de tentatives d'accès au médium, le trafic au niveau du médium ou la priorité de la trame à transmettre. Ainsi un certain nombre de modifications ont été proposées à cette norme pour améliorer les performances Khanafer et al. [2014]. D'autre part, assez récemment une nouvelle version de cette norme dénommée 802.15.4-TSCH a été publiée IEEE [2015].

Nous venons de décrire le fonctionnement général de la couche MAC proposée dans le standard IEEE 802.15.4. Ce standard est mis en œuvre par la majorité des fabricants de matériels pour réseaux de capteurs. Cela permet une bonne interopérabilité entre les divers types de cartes et capteurs. En parallèle, un certain nombre de protocoles d'accès au médium radio ont spécifiquement été proposés pour les réseaux de capteurs. Ces protocoles peuvent être classés suivant trois catégories : les méthodes d'accès avec contention, les méthodes d'accès sans contention, et les méthodes d'accès hybrides. Nous allons présenter dans la suite de cette section quelques protocoles proposés dans la littérature et représentatifs de chacune de ces catégories.

#### 3.3.2 Accès avec contention

Les protocoles d'accès au médium par contention sont fondés principalement sur l'approche CSMA/CA. Plusieurs protocoles alternatifs basés sur cette approche existent, nous décrivons ci-après deux exemples de protocoles :

- **S-MAC (Sensor-MAC)** : ce protocole, proposé par Ye *et al.* [Ye et al. 2002], est considéré comme l'une des références parmi les protocoles MAC synchronisés pour les réseaux de capteurs sans fil. L'objectif de ce protocole est d'assurer la longévité du réseau en évitant les collisions, l'écoute active du canal de transmission, et la surcharge du réseau par un nombre trop important de messages de contrôle. Le principe de fonctionnement de S-MAC est de diviser le temps en deux périodes :

### 3.3. MÉTHODES D'ACCÈS AU MÉDIUM RADIO

---

*active et endormie.* En période active, un nœud peut émettre ou recevoir des données tandis qu'en période endormi aucune transmission ou réception n'est possible. Afin d'éviter le problème d'envoi sourd, S-MAC synchronise les périodes actives de chaque nœud du réseau avec celles de son voisinage. La période active est divisée en trois parties : la synchronisation du réveil des nœuds, l'avertissement de données à envoyer, et l'acquittement suivi de l'envoi des données. Durant la synchronisation des réveils, chaque nœud diffuse dans son voisinage un paquet de type SYNC contenant le temps de son prochain réveil. De plus, chaque nœud maintient une table stockant le temps du prochain réveil de chacun de ses voisins. Cette table est mise à jour lors de la réception d'un paquet SYNC d'un voisin. Cela permet à chaque nœud de connaître les intervalles de temps où il est possible de transmettre des données à destination d'un voisin. Lorsqu'un nœud désire envoyer des données à destination d'un de ses voisins également en période active, il doit l'informer via l'envoi d'un paquet de type RTS (Request To Send) durant la seconde partie de la période active en le désignant comme destinataire. L'envoi des paquets RTS est effectué en utilisant l'approche CSMA/CA. Enfin durant la dernière partie de la période active, un nœud destinataire acquitte un paquet RTS particulier reçu en envoyant en retour un paquet de type CTS (Clear To Send). À partir de cet instant, les deux nœuds voisins peuvent échanger une série de fragments de données durant le reste de la période active.

Le principal avantage de ce protocole est l'augmentation de la durée de vie du réseau grâce à une faible consommation de la batterie au niveau de chaque nœud, obtenue par l'utilisation de longues périodes de mise en veille des nœuds. De plus, S-MAC s'adapte aux changements de la topologie en cas d'ajout ou de suppression de nœuds. Enfin, c'est un protocole simple à implémenter et à mettre en œuvre. Cependant, S-MAC possède aussi deux inconvénients majeurs qui sont d'une part une consommation inutile des batteries des nœuds à cause d'une organisation des périodes d'activité et de sommeil qui ne tient pas compte du trafic à transmettre par chaque nœud. Ainsi, les phénomènes d'envoi sourd ou d'écoute active peuvent être fréquents. D'autre part, une latence de bout-en-bout importante pour la transmis-

sion des trames est souvent observée en partie due aux longues périodes de sommeil des nœuds.

- **B-MAC (Berkeley-MAC)** : B-MAC, proposé par Polastre et al. [Polastre et al. 2004], est un protocole d'accès au médium assez populaire à cause de sa simplicité de conception et de mise en œuvre. B-MAC utilise des préambules d'échantillonnage de faible taille (8 octets pour B-MAC contre 18 octets pour S-MAC), pour synchroniser émetteur et récepteur lors de la transmission d'une trame. Le but du préambule est d'alerter un nœud destinataire d'une trame, et de l'informer qu'il a des données à recevoir afin de rester actif lors de son réveil. De cette façon, chaque nœud émetteur précède sa transmission par la diffusion d'un préambule d'une longueur de temps nécessaire (supérieure à la période d'échantillonnage) pour que le (ou les) récepteur(s) soit (ou soient) réveillé(s). Dans le protocole B-MAC les nœuds se réveillent périodiquement pour écouter et échantillonner le canal à la recherche d'un préambule en utilisant la technique d'écoute à faible énergie (Low Power Listening - LPL). Dans cette technique, l'état du nœud alterne entre période active et période inactive, la période active est d'une durée courte permettant uniquement à un nœud de tester l'état du canal.

L'avantage du protocole B-MAC est la faible consommation d'énergie lors des communications grâce à la technique LPL, et l'utilisation de préambules courts permettant aux nœuds du réseau de rester endormis durant la majeure partie de leur cycle et de se réveiller périodiquement durant un très court intervalle de temps afin d'écouter le canal et ainsi prendre la décision de rester éveillé ou de se rendormir. D'après les comparatifs menés par Polastre et al. [Polastre et al. 2004], la consommation énergétique de S-MAC croît linéairement par rapport à l'augmentation de la quantité de données à transmettre dans le réseau. Cela est dû principalement à la synchronisation entre les nœuds. À l'inverse, la consommation énergétique de B-MAC croît de façon sous-linéaire. Ainsi, à partir de 45 bits/s B-MAC atteint une consommation moyenne inférieure à celle atteinte par S-MAC. Les inconvénients de ce protocole sont d'une part la longueur du préambule qui peut impacter la latence de bout en bout dans un réseau multi-sauts, et d'autre part une perte d'énergie due

### 3.3. MÉTHODES D'ACCÈS AU MÉDIUM RADIO

---

à l'écoute active du canal par les nœuds détectant l'envoi d'un préambule mais qui ne leur est pas destiné.

- **ContikiMAC** : Comme B-MAC [Polastre et al. 2004], le protocole ContikiMAC proposé par Dunkels [Dunkels 2011] utilise la technique du préambule et du duty-cycle pour réduire la consommation énergétique des nœuds d'un RCSF. Cela en alternant l'état du module radio de chaque nœud périodiquement (les nœuds se réveillent à des moments potentiellement différents) pour envoyer et recevoir des données. A chaque réveil, le nœud scrute rapidement l'activité du canal de transmission en utilisant deux CCA successifs pour déterminer la présence de trafic. Si le nœud n'a pas de données à transmettre et ne détecte aucune activité sur le canal, alors il se met en sommeil de manière anticipée. Ce mécanisme appelé « fast-sleep » permet d'optimiser la durée de la période de réveil et limiter l'impact de l'évaluation du CCA. ContikiMAC est le protocole MAC utilisé par défaut dans ContikiOS. D'après son auteur [Dunkels 2011], ContikiMAC permet d'atteindre une réduction de la consommation énergétique allant de 10% à 80% sur l'ensemble du réseau en fonction de la fréquence de réveil des nœuds. Dans ContikiMAC, les nœuds envoient continuellement les données pendant l'intervalle de réveil pour s'assurer que tous les voisins les ont reçu et cela jusqu'à réception de l'acquittement du ou des nœuds destinataires (voir Figure 3.5). Dans chaque acquittement les nœuds récep-

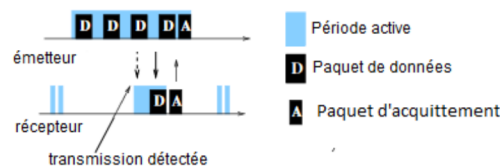


FIGURE 3.5 – Phase de transmission dans ContikiMAC [Dunkels 2011]

teurs indiquent le temps restant jusqu'à leur prochain réveil. Ainsi les nœuds se synchronisent localement en fonction du temps de réveil de leurs voisins directs. Grâce à cette synchronisation des périodes de réveil entre voisins les nœuds peuvent envoyer en continu les paquets de données durant une plus petite période (temps où le voisin est réveillé) pour économiser la batterie et limiter l'utilisation du canal

(voir Figure 3.6).

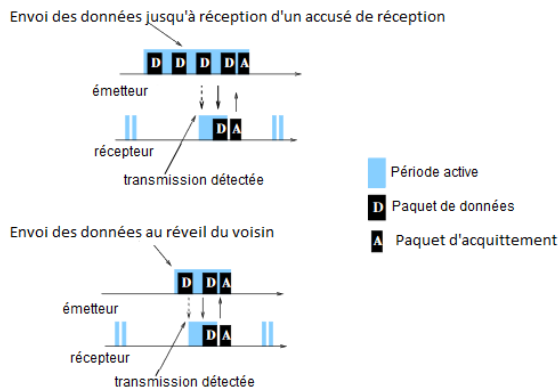


FIGURE 3.6 – Phase de transmission dans ContikiMAC [Dunkels 2011]

#### 3.3.3 Accès sans contention (par planification)

Les protocoles d'accès au médium sans contention se basent principalement sur la méthode TDMA ou sur la méthode FDMA. Parmi les protocoles proposés dans la littérature, nous présentons tout d'abord un protocole d'accès basé sur TDMA puis un second protocole associant les approches TDMA et FDMA :

- **TRAMA (TRaffic-Adaptive Medium Access protocol)** : c'est un protocole qui a été conçu par Rajendran et al. [Rajendran et al. 2003, 2006]. TRAMA est l'un des premiers protocoles basé sur TDMA proposé pour les réseaux de capteurs sans fil dont l'objectif est d'assurer des transmissions sans collision en utilisant un seul canal. L'idée principale de ce protocole est de pouvoir attribuer dynamiquement les intervalles de temps aux nœuds selon le taux de trafic qu'ils ont à transmettre et ainsi de réaffecter les slots non utilisés. Le protocole TRAMA se décompose en trois parties : un protocole de découverte de voisinage NP (*Neighbor Protocol*), un protocole d'échange des ordonnancements de réveil SEP (*Schedule Exchange Protocol*) et un protocole d'élection AEA (*Adaptive Election Algorithm*). Le protocole TRAMA considère l'utilisation d'un seul canal radio partagé dans le temps. Le temps est découpé en deux périodes qui se répètent : une période d'accès aléatoire au médium pour la signalisation et une période d'accès planifiée (divisée en slots de temps réguliers) pour la transmission des données. Le protocole NP permet à chaque nœud de

découvrir son voisinage à deux sauts dans le réseau. Cela est réalisé par l'envoi de la liste de voisinage via des trames de signalisation durant les périodes d'accès aléatoires. Le protocole SEP assure l'échange des informations sur le volume de trafic de données à transmettre ainsi que l'ordonnancement des nœuds dans le voisinage à distance deux. Enfin l'algorithme d'élection AEA détermine le nœud émetteur et récepteur(s) à activer pour chaque slot de la période d'accès planifiée. La sélection des nœuds est effectuée en utilisant les informations obtenues avec les protocoles NP et SEP.

L'avantage de TRAMA est le partage efficace du canal (sans collision) pour la transmission des données entre les nœuds du réseau. TRAMA permet de mettre en sommeil les nœuds du réseau plus longuement en fonction du trafic. De plus, le taux de trafic de données des nœuds du réseau est pris en compte. Les nœuds sont mis en sommeil jusqu'à 10% de plus par rapport à S-MAC dès un taux d'envoi de 6 paquets par seconde [Rajendran et al. 2006]. En revanche, la taille des trames dans TRAMA est relativement importante ce qui peut induire une latence de bout-en-bout élevée (de l'ordre de 100 à 1000 fois supérieure en comparaison de S-MAC [Rajendran et al. 2006]) en plus des collisions lors de la période d'accès aléatoire.

- **MFLAMA (Multi-Channel Flow-Aware Medium Access)** : MFLAMA [Decker et al. 2008] est un protocole MAC basé sur TDMA qui considère plusieurs canaux pour la transmission des données. Il est proposé comme une amélioration de TRAMA. MFLAMA se réfère à un algorithme distribué pour établir le plan d'ordonnancement des transmissions à travers l'ensemble des canaux. MFLAMA assure une transmission sans collision où chaque nœud doit connaître ses voisins à deux sauts ainsi que le volume d'informations à envoyer afin d'établir l'ordonnancement des transmissions. L'accès aux canaux dans MFLAMA est organisé en deux parties : la période d'accès aléatoire et la période d'accès déterministe. Durant la période d'accès aléatoire tous les nœuds accèdent à un canal de signalisation dédié afin de s'échanger : le temps de synchronisation local, les informations sur le trafic et les informations sur leurs voisins à deux sauts. Durant la période d'accès déterministe

### 3.3. MÉTHODES D'ACCÈS AU MÉDIUM RADIO

---

le temps est divisé en slots de taille fixe. Le nombre de slots durant cette partie peut varier d'une période d'accès déterministe à l'autre. Un algorithme distribué est mis en place pour affecter les slots aux nœuds émetteurs et récepteurs afin d'assurer une transmission des données sans collisions. Tous les nœuds exécutent cet algorithme afin de déterminer leur état (transmission, réception, veille) ainsi que le canal de transmission choisi pour chaque slot de la prochaine période d'accès déterministe. Cet algorithme assure qu'il n'y a qu'un seul nœud émetteur sur un canal dans un voisinage à deux sauts pour éliminer le problème du nœud caché. Il assure aussi que l'émetteur et le récepteur se mettent sur le même canal durant la transmission de données.

La méthode d'accès MFLAMA permet la transmission sans collision d'un plus grand nombre de paquets de données en exploitant les divers canaux disponibles. Cela permet de diviser par au moins quatre le délai de transmission de bout en bout en fonction du taux d'envoi de paquets. De plus, MFLAMA permet un gain jusqu'à 20% sur la consommation en énergie par rapport à TRAMA pour des taux d'envoi de paquets inférieurs à 6 paquets par seconde. Le principal inconvénient de ce protocole est la surcharge rapide du réseau par la transmission d'un grand nombre de messages de synchronisation suivant la taille et la densité du réseau.

#### 3.3.4 Accès hybrides

Les protocoles d'accès hybrides sont ceux qui combinent les deux techniques, à savoir l'accès avec contention (basé sur CSMA/CA) et l'accès sans contention (basé sur TDMA). L'objectif de ce type de protocole est de proposer des solutions ayant les avantages à la fois de TDMA pour éviter la collision lors de la transmission des paquets et de CSMA/CA pour réduire le temps d'attente avant l'envoi des paquets à chaque saut.

Nous présentons ci-dessous le protocole Z-MAC qui est le plus représentatif de la classe de protocole d'accès hybride :

- **Z-MAC (Zebra MAC)** : le protocole Z-MAC [Rhee et al. 2008] alterne des périodes d'accès avec contention et sans contention. L'utilisation d'une méthode ou l'autre pour l'accès au médium est guidée par la densité du trafic à chaque niveau



### 3.3. MÉTHODES D'ACCÈS AU MÉDIUM RADIO

---

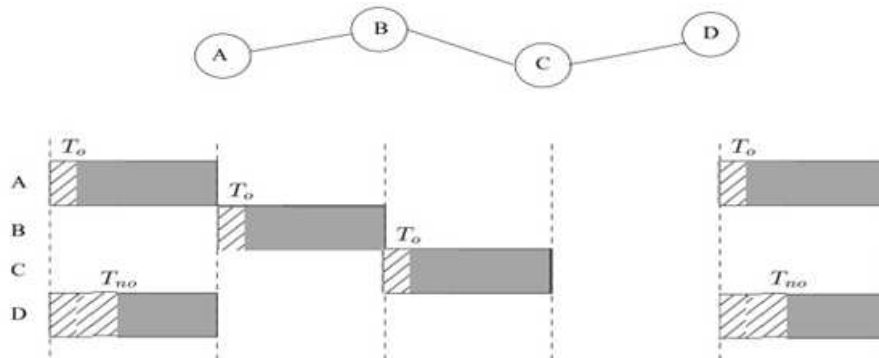


FIGURE 3.7 – Protocole Z-MAC [Rhee et al. 2008]

dans le réseau. La méthode d'accès CSMA/CA est utilisée pour une faible densité du réseau, alors que la méthode d'accès TDMA est privilégiée dans le cas d'une contention élevée. Dans Z-MAC, le temps est divisé en slots. Les nœuds du réseau commencent par exécuter une phase de découverte de leur voisinage à deux sauts suivie par une phase d'attribution des slots en utilisant un algorithme distribué. Comme dans TDMA, un slot est attribué périodiquement à chaque nœud du réseau, mais dans Z-MAC un nœud peut également transmettre dans d'autres slots qui sont assignés à d'autres nœuds, mais non utilisés par ceux-ci. Un nœud est prioritaire pour transmettre lors du slot qui lui correspond. Pour accéder au canal, si le nœud est propriétaire du slot courant alors il attend un temps aléatoire plus petit que  $T_0$  puis teste l'état du canal (voir le nœud A dans la Figure 3.7). Si ce dernier est libre alors le nœud émet sinon il recommence cette même étape. Sinon si le nœud n'est pas propriétaire du slot alors il attend un temps aléatoire supérieur à  $T_0$  et inférieur à  $T_{no}$  avant d'émettre (voir le nœud D dans la Figure 3.7). Mais si le nœud n'est pas propriétaire du slot et le slot appartient à un de ses voisins à deux sauts et il a reçu une indication de forte contention d'un de ses voisins alors il n'a pas le droit de l'utiliser.

L'avantage de Z-MAC est sa robustesse lors du changement de topologie et des échecs de synchronisation contrairement à TDMA. Z-MAC améliore aussi les performances de CSMA/CA en réduisant les collisions puisque les nœuds propriétaires de leur slot ont la chance de transmettre avant les autres nœuds et si un slot n'est

pas utilisé par son propriétaire alors les autres nœuds peuvent l'utiliser. Ainsi, cela permet d'assurer également une meilleure latence de bout-en-bout par rapport à l'approche TDMA. En revanche, Z-MAC n'a aucune garantie sur l'économie d'énergie et dans le pire des cas, sa performance est équivalente à un protocole suivant l'approche CSMA/CA.

Nous venons de voir à travers la présentation de protocoles proposés au niveau de la couche MAC différentes approches mises en place afin de réduire la consommation d'énergie au niveau de chaque du nœud du réseau. Pour de plus amples détails, on peut se référer à la taxonomie des approches proposées pour réduire la consommation d'énergie dans les RCSF, notamment au niveau de la couche MAC, présentée par Anastasi *et al.* [Anastasi et al. 2009]. D'autre part, une évolution des diverses stratégies mises en œuvre au niveau de la couche MAC pour les RCSF, ainsi que le fonctionnement de ces divers protocoles est décrite par Huang *et al.* [Huang et al. 2013].

Cependant, une surconsommation en énergie peut également être due à la façon dont les données sont acheminées par la couche réseau. En effet, la politique de routage mise en œuvre doit prendre en compte les caractéristiques et les modes d'échange de données propres aux réseaux de capteurs, sans quoi cela peut engendrer une surcharge du réseau induisant une surconsommation des batteries. Dans la section suivante, nous allons présenter une classification des protocoles de routage ainsi que les principaux protocoles de routage proposés dans cadre des réseaux de capteurs.

## 3.4 Routage dans les RCSF

Le routage intervient au niveau de la couche réseau, cette couche a pour principal objectif d'assurer l'acheminement des trames dans le réseau de capteurs sans fil, en réduisant la consommation énergétique.

En fonction des exigences applicatives, trois schémas de routage sont utilisés dans les réseaux de capteurs : la *collecte de données*, la *diffusion de données* et des *communications entre paire de capteurs*. Dans le premier schéma, l'objectif est d'acheminer au niveau du

nœud puits toutes les données relevées par les nœuds capteurs du réseau. Dans la diffusion de données, c'est l'inverse il faut acheminer une donnée envoyée par le puits à tous les nœuds du réseau. Enfin le dernier schéma, deux nœuds capteurs s'échangent des données sans que cela passe nécessairement par le puits. Parmi ces trois schémas, la collecte de données est le schéma de routage le plus couramment rencontré. La topologie de routage la plus utilisée dans ce cas est une topologie arborescente dont le puits est la racine.

Les protocoles de routage peuvent être classés soit selon la topologie de routage construite (routage à plat ou hiérarchique), soit selon le mode de construction des routes (construction proactive ou réactive).

- **Routage à plat (ou non-hiérarchique)** : dans ce type de routage, tous les nœuds du réseau ont le même rôle pour le routage des données, excepté le nœud puits chargé de la collecte des données du réseau. Chaque nœud du réseau est capable d'acheminer les données de proche en proche jusqu'au nœud puits. Le routage à plat est plus efficace et plus simple dans des réseaux de taille moyenne.
- **Routage hiérarchique** : ce type de routage est généralement appliqué dans des réseaux de grande taille. Dans ce cas, les nœuds capteurs du réseau sont regroupés en plusieurs parties (appelées « *cluster* ») dans le but de faciliter l'acheminement des données et réduire la consommation d'énergie. Chaque cluster est composé d'un ensemble de nœuds capteurs qui communiquent directement avec un nœud particulier, appelé *nœud chef* (« *Cluster Head* »). Deux types de nœuds existent : les nœuds chefs et les nœuds membres d'un cluster. Les nœuds chefs ont une quantité d'énergie plus importante que les nœuds membre d'un cluster, et ils ont le rôle de coordonner leur cluster et de router les messages de données entre le puits et les membres du cluster dont ils ont la charge. Les nœuds membres d'un cluster ont un rôle qui se restreint à la capture de l'information et l'envoi au nœud chef de leur cluster.
- **Routage proactif** : dans cette famille de protocoles de routage, les nœuds capteurs maintiennent une table contenant les informations de routage vers tous les autres nœuds du réseau. Cette table est mise à jour à l'aide d'échanges d'informations périodiques entre les nœuds ou suite à un changement topologique. Le changement

fréquent de la topologie du réseau peut donc engendrer un échange important d'information pour la mise à jour des tables de routage, entraînant une surconsommation d'énergie. Par conséquent, ces protocoles atteignent rapidement leurs limites pour des réseaux dynamiques et de grande taille.

- **ROUTAGE réactif** : dans cette famille de protocoles, les routes vers chaque destination ne sont pas connues à l'avance contrairement aux protocoles proactifs. Les nœuds capteurs doivent donc déterminer le chemin de transmission des données sur demande. Les chemins sont construits dynamiquement lorsqu'un nœud doit transmettre des données vers une destination. Au moment de transmettre un message le nœud lance un processus de découverte et de sélection de chemin pour déterminer une route vers le destinataire des données. À l'inverse des protocoles proactifs, ils sont plus efficaces à faible charge de trafic dans le réseau.

Dans la suite, nous présentons plusieurs protocoles de routage dédiés aux réseaux de capteurs, classés en fonction du mode suivi pour construire les routes dans le réseau.

#### 3.4.1 Protocoles de routage proactifs

Dans cette partie, nous présentons les principaux protocoles de routage proactifs proposés dans la littérature.

##### 3.4.1.1 Le protocole DSDV (Destination Sequence Distance Vector)

Le protocole DSDV [Perkins and Bhagwat 1994] est un protocole de routage proactif à vecteur de distance basé sur l'algorithme de Bellman-Ford pour la construction des routes. Dans ce protocole, chaque nœud du réseau maintient des informations sur tous les autres nœuds du réseau dans sa table de routage. Cette table de routage contient les informations suivantes : le nœud destinataire, le nombre de sauts pour l'atteindre et un numéro de séquence associé. Un numéro de séquence est associé à chaque nœud destinataire et permet de distinguer une nouvelle route d'une ancienne présente dans la table pour atteindre un destinataire. Ce numéro est initialisé par le nœud émetteur et incrémenté à chaque nouvelle route.

### 3.4. ROUTAGE DANS LES RCSF

---

La mise à jour des tables de routage se fait de manière périodique de proche en proche dans le réseau. Pour cela, chaque nœud transmet sa table de routage à tous ses voisins directs. Pour éviter de surcharger le réseau, DSDV s'appuie sur des mises à jour incrémentales où seules les entrées modifiées ou nouvelles sont envoyées.

#### 3.4.1.2 Le protocole OLSR (Optimized Link State Routing)

Le protocole OLSR [Clausen and Jacquet 2003] est un protocole de routage proactif à état de liens optimisé ; il permet de sélectionner les plus courts chemins pour acheminer les données. Contrairement à un protocole à état de liens traditionnel où chaque nœud du réseau diffuse un message contenant tous ses liens directs avec ses voisins dans le réseau, dans le protocole OLSR chaque nœud sélectionne un sous-ensemble de ses nœuds voisins comme nœuds relais (intermédiaires), appelés MPR (*MultiPoint Relaying*). Ces nœuds relais sont utilisés dans le cas où un nœud du réseau veut transmettre des données à un autre nœud destinataire se trouvant hors de sa portée de transmission. Ils sont utilisés pour diminuer le nombre de messages de contrôle échangés lors du changement de la topologie. Ils sont aussi utilisés pour calculer les plus courts chemins vers une destination quelconque dans le réseau. La Figure 3.8 illustre la sélection des nœuds relais effectuée par OLSR en comparaison des protocoles à état de liens classique.

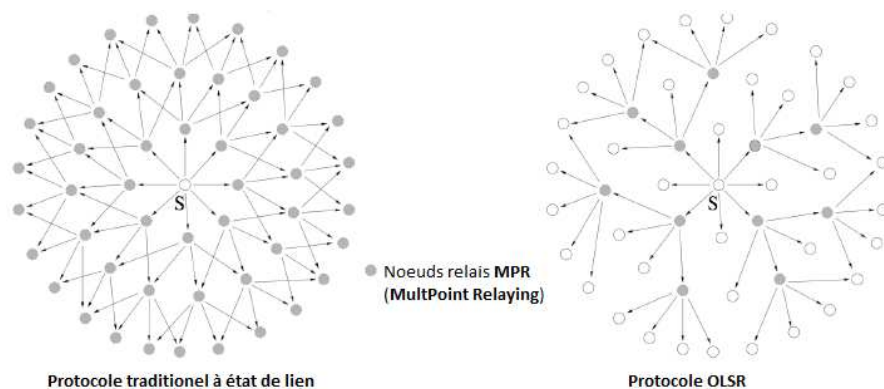


FIGURE 3.8 – Protocole OLSR

Tous les nœuds du réseau maintiennent une table de routage ayant une entrée par nœud relais voisin. Pour maintenir cette table de routage à jour, les nœuds OLSR échangent périodiquement des messages de type Hello contenant la liste des voisins. Un autre type

### 3.4. ROUTAGE DANS LES RCSF

---

de message, appelé message TC (Topology Control), est utilisé pour maintenir à jour les informations nécessaires au choix des nœuds relais.

#### 3.4.1.3 Le protocole RPL (IPv6 Routing Protocol for Low-Power and Lossy Networks)

Le protocole RPL [Winter et al. 2012] est un protocole de routage proactif à vecteur de distance standardisé par l'IETF. Il est basé sur IPv6 et fonctionne au-dessus de 6LoWPAN (*IPv6 Low poWer Personal Area Networks*) pour la construction et le maintien des routes dans le réseau. RPL essaie de construire les routes pour acheminer les données avant qu'elles ne soient nécessaires aux nœuds du réseau. Les routes conçues forment des topologies appelées graphes acycliques orientés dirigés vers une destination (Destination Oriented Acyclic Graphs - DODAG). Lors de la construction des routes du DODAG, RPL utilise un ensemble de messages de contrôle ICMPv6 de type DIO (DODAG Information Object), DIS (DODAG Information Solicitation) et DAO (Destination Advertisement Object). Le choix des routes dans RPL est défini par une fonction objectif, nommée OF (Objective Function). Nous ne détaillons pas davantage ici ce protocole car il sera étudié plus en détail dans le Chapitre 5.

#### 3.4.2 Protocoles de routage réactifs

Dans cette partie, nous présentons les principaux protocoles de routage réactifs proposés dans la littérature.

##### 3.4.2.1 Le protocole DSR (Dynamic Source Routing)

Le protocole DSR [Johnson et al. 2007] est un protocole de routage réactif, basé sur le principe de routage à la source, c'est-à-dire le nœud émetteur source détermine la séquence des nœuds intermédiaires à travers lesquels les données seront envoyées jusqu'au nœud destinataire.

Lorsqu'un nœud  $S$  souhaite émettre des données vers une destination  $D$ , il diffuse un message de type "Route Request". Une fois la découverte du chemin vers le nœud destinataire  $D$  réalisée le nœud émetteur  $S$  reçoit un message de type "Route Response"

contenant l'ensemble des nœuds intermédiaires de  $S$  jusqu'à  $D$ . Cette technique de routage permet aux nœuds du réseau d'économiser l'énergie d'envoi d'informations de mise à jour de la topologie.

Dans le protocole DSR, le mécanisme de découverte de chemin d'un nœud émetteur (nœud source  $S$ ) jusqu'au nœud destinataire  $D$  est lancé uniquement si il n'existe pas de chemin vers  $D$  dans le cache de  $S$ . Pour maintenir ces routes, un mécanisme d'envoi de message d'erreur est mis en place pour signaler le changement de topologie. Si une rupture d'un lien se produit entre deux nœuds intermédiaires sur un chemin dans le cache du nœud  $S$ , alors un message d'erreur est envoyé au nœud  $S$  pour utiliser un chemin alternatif.

#### 3.4.2.2 Le protocole AODV (Ad Hoc On-demand Distance Vector)

Comme le protocole DSR, le protocole AODV [Perkins et al. 2003] est un protocole de routage réactif basé sur le principe de routage à la source. Il est conçu essentiellement pour améliorer le protocole DSR en minimisant le nombre de messages diffusés pour la construction des routes. Le protocole AODV se base sur deux mécanismes :

- Création d'un chemin pour la transmission de données entre un nœud source  $S$  et un nœud destinataire  $D$ .
- Mise à jour des chemins au niveau de chaque nœud.

Le protocole AODV construit les chemins de routage d'un nœud source jusqu'à un nœud destinataire en se basant sur les messages Route Request et Route Reply. Lorsqu'un nœud source souhaite émettre des données vers un nœud destinataire pour lequel il ne possède pas de chemin dans sa table, il diffuse un message de type RREQ (Route Request) dans le réseau. Le nœud récepteur du message RREQ renvoie un message de type RREP (Route Reply) si c'est le nœud destinataire ou si c'est un nœud intermédiaire possédant une route vers le nœud destinataire avec un numéro de séquence supérieur ou égal à celui reçu dans le message RREQ. Le numéro de séquence permet d'identifier les chemins les plus récents. Contrairement au protocole DSR qui maintient la totalité des chemins, dans le protocole AODV les chemins de routage entre les nœuds sont maintenus tant qu'ils sont actifs. Un chemin est considéré comme *actif* tant que des paquets de données circulent périodiquement à travers celui-ci. Quand un nœud source arrête de transmettre des données sur un chemin,

ce dernier expire au bout d'un temps fixé et sera effacé de la table de routage.

#### 3.4.2.3 Le protocole LOADng (Lightweight On-demand Ad hoc Distance-vector Routing Protocol - Next Generation)

Le protocole LOADng [Clausen et al. 2013] est un protocole de routage réactif. Afin de construire une route vers une destination, un nœud émetteur inonde le réseau avec des demandes de route (message de type RREQ) pour découvrir un chemin de routage jusqu'au nœud destination. LOADng est une variante simplifiée du protocole AODV destiné aux réseaux 6LoWPAN (IPv6 Low poWer Personal Area Networks). Ce protocole nécessite moins de mémoire de stockage et d'énergie pour fonctionner et est donc plus approprié au contexte des réseaux de capteurs. Lors de la construction des routes, LOADng s'appuie sur les messages de contrôle tels que RREQ (Route Request), RREP (Route Reply), RREP-ACK et RERR (Route Error). Contrairement à AODV, dans le protocole LOADng seul le nœud destinataire a l'autorisation de répondre au message RREQ lors de la découverte des routes. Cela permet de minimiser le nombre de messages de contrôle échangés dans le réseau, ainsi que l'énergie consommée lors de ces échanges. Le maintien de chaque chemin de routage se fait de façon indépendante dans le protocole LOADng, contrairement à d'autres protocoles de routage, comme RPL, qui nécessitent un recalcul de topologie globale.

Un certain nombre de travaux ont été menés pour étudier les performances atteintes par les protocoles de routage mentionnés dans cette section. Nous pouvons notamment citer le travail de Yang *et al.* [Yang et al. 2007] qui démontrent à l'aide de simulations une meilleure stabilité du taux de réception des paquets au niveau du puits pour les protocoles AODV et DSR en comparaison de DSDV lorsque le déploiement des capteurs n'est pas régulier. D'autre part, Kulkarni *et al.* [Kulkarni et al. 2011] ont comparé les protocoles AODV, DSDV et DSR suivant divers scénarii de déploiement aléatoire d'un RCSF. Si les nœuds sont statiques, le protocole DSR atteint la plus faible consommation totale d'énergie suivi de AODV et DSDV. Yang *et al.* [Yang et al. 2009] ont comparé la durée de vie atteinte par les protocoles OLSR et AODV. Les résultats de simulation obtenus démontrent que le protocole OLSR a une durée de vie supérieure à celle de AODV. Enfin, Vucinic *et*



*al.* [Vucinic et al. 2013] ont étudié les performances atteintes par les protocoles RPL et LOADng dans le contexte de l’habitat intelligent. Les résultats de simulation tendent à montrer que RPL atteint de meilleurs délais de bout-en-bout, un nombre plus faible de messages de contrôle échangés et une plus faible quantité de mémoire sur les nœuds capteurs.

## 3.5 Conclusion

La maîtrise de la consommation énergétique dans les réseaux de capteurs pour maximiser la durée de vie du réseau reste le principal objectif. Dans la première partie de ce chapitre nous avons présenté les principales causes de surconsommation d’énergie, mettant en avant l’importance de la gestion des communications radio qui sont les plus consommatrices en énergie. Pour cela, il est nécessaire d’étudier les mécanismes mis en œuvre par les couches d’accès au médium radio et de routage pour la transmission des données.

Dans la deuxième partie de ce chapitre, nous avons présenté le fonctionnement des principaux protocoles d’accès au médium radio proposés dans la littérature pour les réseaux de capteurs sans fil, ayant comme objectif de minimiser la consommation énergétique en évitant les collisions des transmissions dans le réseau. Enfin, dans la dernière partie de ce chapitre nous avons présenté les principaux protocoles de routage suivant deux catégories : proactif ou réactif. Le principal inconvénient des protocoles de routage proactifs est la surcharge avec des messages de contrôle pour la maintenance des tables de routage. A l’inverse les protocoles de routage réactifs, construisent les chemins de routage à la demande mais cela induit un temps de transmission plus élevé.

Suite à notre étude des protocoles d’accès au médium radio, nous exposerons dans le chapitre suivant la solution que nous proposons pour minimiser les interférences et les collisions. Par la suite, dans le Chapitre 5 nous nous intéressons à l’étude des performances du protocole de routage RPL pour économiser l’énergie consommée et maximiser la durée de vie du réseau.

## Chapitre 4

# Ordonnancement sans collision des transmissions

### 4.1 Introduction

Dans le chapitre précédent, nous avons présenté les caractéristiques et contraintes liées aux communications dans le contexte des réseaux de capteurs sans fil. Nous avons également identifié les principales sources pouvant engendrer une surconsommation en énergie sur les nœuds d'un réseau de capteurs sans fil, qui sont principalement liées aux méthodes d'accès au médium radio et au routage des paquets dans le réseau. Nous avons ensuite présenté les principales approches et protocoles proposés dans la littérature associés aux problèmes liés à ces deux couches protocolaires. Dans ce chapitre, nous allons nous focaliser plus particulièrement sur l'accès et le partage du médium radio de sorte à réduire l'impact des interférences, liées aux communications radio ou à l'environnement autour du réseau de capteurs, sur la transmission des données. Plus spécifiquement, nous cherchons à minimiser les collisions engendrées par les interférences pour collecter des données transmises par les nœuds capteurs à destination du puits.

Dans ce chapitre, nous considérons un réseau de capteurs sans fil multi-sauts destiné à collecter au niveau d'un unique puits les données issues de l'ensemble des nœuds capteurs. Nous faisons l'hypothèse qu'une topologie arborescente connectant tous les nœuds et enracinée au puits est déjà construite et utilisée pour la collecte des données. La construction et le maintien de la topologie de routage est l'objet des chapitres suivants de ce manuscrit. De

plus, chaque nœud dispose d'un module radio pour communiquer qui utilise un seul canal radio identique pour tous les nœuds. Ainsi, nous devons prendre en compte dans le partage du médium radio les problèmes liés aux communications sans fil (problèmes du nœud caché et du nœud exposé) décrit au chapitre précédent. Enfin, l'accès au médium radio est découpé en intervalles de temps unitaires (« *slots* ») et nous adopterons donc une approche basée sur TDMA pour le partage du médium radio. En effet, cette approche permet de réduire la consommation énergétique en limitant les interférences grâce à la désactivation du module radio des nœuds durant les slots de temps où ceux-ci ne communiquent pas. Dans la suite de ce chapitre, nous cherchons d'une part à maximiser le nombre de transmissions simultanées réalisables, c'est-à-dire en tenant compte des caractéristiques du réseau que nous venons de décrire pour éviter les collisions, et d'autre part à minimiser le temps total (nombre d'intervalles de temps ou slots) nécessaires pour collecter chaque donnée émise par l'ensemble des nœuds capteurs.

Ce chapitre est composé de trois sections, nous commençons d'abord par présenter dans la Section 4.2 une modélisation du partage et de l'accès au médium radio, puis nous définissons le problème que nous cherchons à résoudre ainsi que sa complexité de résolution. Ensuite, nous présentons un premier algorithme distribué pour résoudre le problème formulé et terminons ce chapitre en proposant une évolution de cet algorithme fournissant des garanties de tolérance aux fautes transitoires.

### 4.2 Modélisation de l'accès au médium

Dans la suite, nous considérerons des topologies statiques multi-sauts dans lesquelles les nœuds possèdent des rayons de communication bornés mais non forcément uniformes entre eux. Chaque nœud est équipé d'un seul module radio half-duplex, c'est-à-dire que pour communiquer un nœud pourra être soit en mode *transmission* pour envoyer des données à un voisin ou en mode *réception* pour recevoir des données envoyées par un voisin mais pas dans les deux modes simultanément. De plus, l'accès au médium radio est découpé en intervalles de temps unitaires (« *slots* »). Nous nous focaliserons sur les stratégies déterministes basées sur l'approche TDMA pour le partage du médium radio entre les nœuds du réseau. Comme décrit au Chapitre 3, cela permet de réduire la consommation énergétique

en mettant en sommeil les nœuds pendant les slots qui ne leur sont pas réservés et minimise les interférences radio. De plus, il est possible de borner le nombre de slots nécessaires pour la collecte des données jusqu'au nœud puits.

Dans la suite, nous considérerons uniquement les transmissions de messages contenant des données relevées par les capteurs et qui doivent être acheminées au puits. Nous allons dans un premier temps définir les situations conduisant à la perte de messages de données lors de leurs transmissions suite aux interférences pouvant se produire dans le réseau. Nous présenterons ensuite différents problèmes étudiés dans la littérature pour minimiser les pertes de messages. Enfin, nous définirons formellement le problème que nous cherchons à résoudre avec une étude de complexité.

### 4.2.1 Caractéristiques des communications sans fil

Les communications radio dans un réseau de capteurs sans fil multi-sauts sont soumises à différentes contraintes qui peuvent être matérielles ou environnementales, telles que les obstacles physiques ou des perturbations électromagnétiques. Tout ceci induit des *interférences* au niveau de la transmission des messages, qui se traduisent par des pertes de messages. Pour minimiser ces pertes dues aux interférences engendrées entre les communications, nous devons analyser l'impact de la transmission de chaque message entre deux nœuds voisins sur les autres transmissions dans le réseau. Plus précisément, nous cherchons à éviter les interférences qui conduisent à obtenir des *collisions* lors de la réception de plusieurs messages, interférant les uns sur les autres lors de leurs transmissions simultanées. Plusieurs travaux [Ghosh et al. 2011] ont défini les différentes situations à prendre en compte générant des collisions. Nous listons ci-dessous ces différentes situations illustrées à la Figure 4.1 :

**Cas 1 :** Plusieurs nœuds transmettent simultanément un message différent à destination du même nœud récepteur.

**Cas 2 :** Un nœud est à la fois émetteur d'un message et récepteur d'un message émis par un autre nœud.

**Cas 3 :** Un nœud est récepteur au même instant d'au moins deux messages, l'un qui lui est destiné et au moins un autre dont il n'est pas le destinataire.

## 4.2. MODÉLISATION DE L'ACCÈS AU MÉDIUM

---

Les Cas 1 et 2 indiqués ci-dessus sont couramment appelés *conflits primaires* (ou *conflit de type 1*), tandis que le Cas 3 est appelé *conflit secondaire* (ou *conflit de type 2*) [Ghosh et al. 2011].

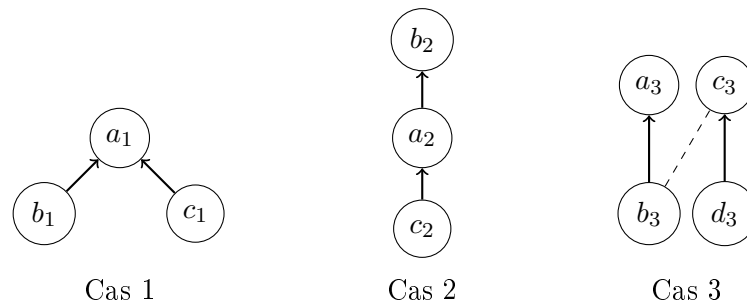


FIGURE 4.1 – Illustration des situations de conflits primaires et secondaires, où une flèche indique l'orientation d'une transmission de message et un trait pointillé représente une communication possible entre deux nœuds. Le schéma de gauche présente le Cas 1, les nœuds  $b_1$  et  $c_1$  envoient un message à destination du même nœud  $a_1$ . Le schéma au centre concernant le Cas 2, le nœud  $a_2$  envoie un message au nœud  $b_2$  pendant qu'il reçoit un message envoyé par le nœud  $c_2$ . Le schéma de droite illustre le Cas 3, le nœud  $c_3$  reçoit simultanément un message qui lui est destiné envoyé par le nœud  $d_3$  et un message envoyé par  $b_3$  uniquement à destination de  $a_3$ .

Dans le contexte des réseaux de capteurs multi-sauts, plusieurs travaux [Gandham et al. 2008b; Incel and Krishnamachari 2008; Rajendran et al. 2006] ont montré qu'une approche basée sur TDMA pour la collecte de données permet d'atteindre de bonnes performances grâce à l'augmentation du nombre de transmissions de données simultanées sans conflit. Cela est dû à la répartition dans le temps (ou *ordonnancement*) des transmissions en conflit, permettant ainsi d'éliminer les conflits primaires et secondaires. Cependant, la taille de l'ordonnancement peut être importante dans le cas des réseaux à grande échelle si la méthode d'ordonnancement utilisée ne maximise pas correctement le nombre de transmissions parallèles sans conflit à chaque slot de temps. Dans la suite, nous allons présenter plusieurs problèmes d'optimisation considérés dans la littérature pour réduire la taille de l'ordonnancement des transmissions, c'est-à-dire réduire le délai de bout-en-bout, avant de présenter formellement le problème qui nous intéresse.

### 4.2.2 Problèmes d'ordonnement considérés dans la littérature

Dans le contexte d'un réseau de capteurs multi-sauts et à grande échelle, nous nous intéressons à définir une approche permettant de calculer un ordonnancement des transmissions de données minimisant le nombre de slots nécessaires pour collecter une information mesurée par chaque capteur, et minimisant l'énergie consommée (c'est-à-dire n'introduisant aucun conflit lors des transmissions des données). Avant de présenter différents problèmes d'optimisation considérés pour l'ordonnement des transmissions sans fil, nous allons tout d'abord introduire quelques notations qui seront utiles pour la suite.

#### 4.2.2.1 Notations

Un réseau de capteurs sans fil peut être modélisé par un graphe orienté  $G = (V, E)$ , où  $V$  représente l'ensemble des nœuds capteurs et  $E$  représente l'ensemble des liens de communication bidirectionnels. On note  $n = |V|$  le nombre de nœuds (ou sommets) et  $m = |E|$  le nombre de liens dans le réseau (ou arêtes). Chaque nœud  $v \in V$  possède un identifiant unique. Deux nœuds  $i, j \in V$  sont adjacents (voisins) si et seulement si le lien  $(i, j) \in E$ . L'ensemble des voisins d'un nœud  $i \in V$  est désigné par  $N(i)$ . On note  $\delta_i = |N(i)|$  le degré du nœud  $i$  et  $\Delta = \max_{j \in V} \delta_j$  le degré maximum d'un nœud dans  $G$ . Un nœud particulier du graphe  $s \in V$  est associé au nœud puits. Dans certains cas, on pourra considérer deux types de liens dans le réseau : des arêtes orientées formant un arbre couvrant  $T = (V, E_T)$  de  $G$  enraciné en  $s$  et utilisées pour les transmissions de données, avec  $E_T \subseteq E$ , et des arêtes bidirectionnelles définies par  $E - E_T$  représentant les interférences pouvant se produire lors de la transmission des données. Un nœud  $i \in V$  avec au moins deux nœuds voisins dans  $T$  sera appelé un nœud *interne*, et dans le cas contraire  $i$  est un nœud *feuille*. Enfin, on notera  $d_T(i)$  la distance du nœud  $i \in V$  au puits  $s$  dans l'arbre  $T$ .

#### 4.2.2.2 Ordonnement des transmissions et coloration de graphe

Plusieurs travaux [Gandham et al. 2008a; Grable and Panconesi 1997] ont montré une relation entre les problèmes d'allocation de ressources, notamment l'ordonnement des transmissions sans fil, et les problèmes de coloration de graphe. En effet, l'allocation des slots peut être modélisée comme un problème de coloration de graphe, où une cou-

## 4.2. MODÉLISATION DE L'ACCÈS AU MÉDIUM

leur est associée à chaque slot. Ainsi, la minimisation du nombre de slots utilisés dans le plan d'ordonnement correspond à l'optimisation du nombre de couleurs dans un graphe représentant les contraintes sur les transmissions. Deux types de coloration de graphe existent, la coloration des sommets du graphe ou la coloration des arêtes du graphe. Le tableau 4.1 illustre le cas d'un plan d'ordonnement correspondant à la coloration du graphe de la Figure 4.2. Ce plan d'ordonnement est répété sur deux super-frame successives composées de trois slots de temps chacune. Dans cet exemple, la couleur bleu est associée au slot 1, la couleur rouge au slot 2 et la couleur verte au slot 3. Dans le cas d'une coloration des sommets, les couleurs indiquent le slot utilisé par ceux-ci en émission, tandis que dans le cas d'une coloration d'arêtes la couleur définie le slot durant lequel les deux extrémités d'une arêtes sont activées en émission ou en réception. On peut noter que dans le tableau 4.1, de l'agrégation de données est utilisée. Ainsi dans la Super-frame 2, au slot 1 le puits  $s$  reçoit dans la même trame les données issues de  $a_1$  et  $a_4$ , de même au slot 2 le puits  $s$  reçoit les données issues de  $a_2$ ,  $a_5$  et  $a_6$ .



FIGURE 4.2 – Coloration d'un graphe pour l'allocation des slots.

TABLE 4.1 – Plan d'ordonnement des slots

Récepteur	Super-frame 1			Super-frame 2		
	slot 1	slot 2	slot 3	slot 1	slot 2	slot 3
$s$	1	2	3	(1,4)	(2,5,6)	3
$a_1$	-	4	-	-	4	-
$a_2$	5	-	6	5	-	6

Dans la suite, nous présentons divers problèmes de coloration de graphe utilisés pour modéliser différents problèmes d'allocation de ressources pour les communications sans fil. Pour de plus amples détails concernant la coloration de graphe, il est possible de se référer aux deux livres Balasundaram and Butenko [2006]; Chartrand and Zhang [2008].

### 4.2.2.3 Coloration de sommets

Une coloration des sommets d'un graphe  $G$  est une affectation de couleurs aux sommets telle que toute paire de sommets voisins dans  $G$  aient des couleurs différentes. Le nombre minimum de couleurs utilisées pour colorer un graphe est appelé le nombre chromatique, noté  $\chi(G)$  [Jensen and Toft 2011]. Il est NP-difficile [Holyer 1981] de résoudre de façon optimale le problème de coloration minimum des sommets pour un graphe quelconque. Vizing [Melnikov and Vizing 1999] montre que pour tout graphe de degré maximum  $\Delta$ , le nombre de couleurs nécessaires pour colorer tous les sommets est au plus de  $\Delta + 1$  couleurs. De plus, il est possible de calculer pour tout graphe une coloration utilisant  $\Delta + 1$  couleurs en temps polynomial.

Le problème de coloration de sommets ne permet pas toujours de modéliser précisément l'allocation des slots. En effet comme le montre la Figure 4.3, dans le cas d'une coloration de sommets on affecte les slots de temps aux sommets (voir Figure 4.3(a)). Si A transmet un message au même moment que B (ayant la même couleur) à destination de C, alors une collision se produit au niveau de C malgré une coloration correcte des sommets dans le graphe. En comparaison, le problème de coloration d'arêtes permettra de mieux modéliser les contraintes du problème initial. L'affectation de couleurs différentes aux arêtes voisines permettra de prendre en compte les conflits primaires. Comme on peut le voir sur la Figure 4.3(b), C recevra dans un premier temps le message provenant de A et dans un second temps le message provenant de B car une couleur différente sera affectée à chaque arête adjacente à C.

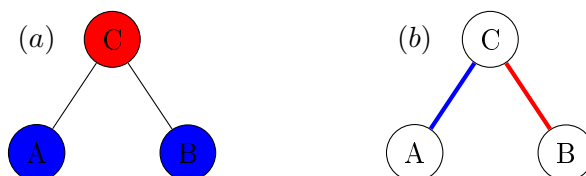


FIGURE 4.3 – Coloration de sommets et coloration d'arêtes.



#### 4.2.2.4 Coloration d'arêtes

Une coloration des arêtes d'un graphe  $G$  est une affectation de couleurs aux arêtes telle que les arêtes ayant une extrémité en commun possèdent des couleurs différentes [Jensen and Toft 2011]. Ce problème est aussi NP-difficile à résoudre, car il est possible de se ramener au problème précédent en appliquant un nombre polynomial de changements sur le graphe d'entrée (c'est-à-dire en construisant le line-graphe du graphe d'entrée). Par conséquent, le théorème de Vizing reste valable dans le cas de la coloration d'arêtes. Ainsi, les arêtes de tout graphe de degré maximum  $\Delta$  peuvent être colorées avec au plus  $\Delta + 1$  couleurs.

Cependant, la coloration d'arêtes ne permet pas de modéliser finement le partage du médium. En effet, les conflits secondaires ne sont pas pris en compte par ce problème. La Figure 4.4(a) illustre une coloration d'arêtes allouant la même couleur pour les deux arêtes  $e_1$  et  $e_2$  non adjacentes, mais qui sont toutes deux adjacentes d'une troisième arête  $e_3$ . Si le sommet  $x$  de l'extrémité commune à  $e_1$  et  $e_3$  transmet un message à destination du sommet  $y$  et le sommet  $v$  de l'extrémité commune à  $e_2$  et  $e_3$  reçoit au même moment un message envoyé par  $u$ , alors une collision se produit au niveau de  $v$  qui recevra deux messages dont celui envoyé par  $x$  dont il n'est pas le destinataire (conflit secondaire). Afin de tenir compte des conflits secondaires, il est nécessaire d'utiliser une variante du problème de coloration d'arêtes, appelé problème de *coloration d'arêtes à distance 2*. La Figure 4.4(b) illustre une solution retournée par ce problème permettant d'allouer des couleurs distinctes pour les arêtes  $e_1$ ,  $e_2$  et  $e_3$ .

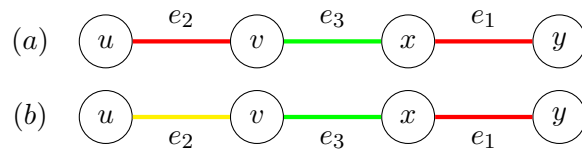


FIGURE 4.4 – Illustration du problème de coloration à un saut.

#### 4.2.2.5 Coloration d'arêtes à distance 2

Une coloration d'arêtes à distance 2 affecte des couleurs aux arêtes d'un graphe  $G$  de telle sorte que pour tout chemin de longueur trois de  $G$  chaque arête du chemin se voit

affecter une couleur distincte. Le défaut des approches de coloration d'arêtes à distance 2 est de ne pas tenir compte de la direction de transmission des données, conduisant parfois à un plan d'ordonnement de taille plus importante que nécessaire. La Figure 4.5(a) illustre un graphe sans information sur l'orientation des transmissions. Dans cette situation, il est nécessaire d'affecter au moins quatre couleurs et donc d'utiliser quatre slots, tandis que sur la Figure 4.5(b) une orientation de certaines arêtes est indiquée pour la transmission de deux messages non interférant entre les sommets 1 et 4 d'une part, et entre les sommets 2 et 3 d'autre part. Dans cette nouvelle situation, seules deux couleurs sont nécessaires. Ainsi, la prise en compte de l'orientation des transmissions peut conduire à réduire de façon significative le nombre de couleurs et donc de slots nécessaires. Pour prendre en compte les orientations des transmissions, une variante du problème de coloration à distance 2 a été introduite par Herman et al. [Herman et al. 2006].

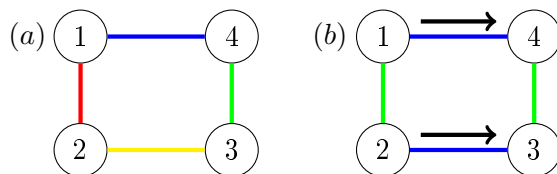


FIGURE 4.5 – Coloration non-orientée et orientée d'arêtes.

#### 4.2.2.6 Coloration d'arêtes orientées

On appelle *orientation* d'un graphe  $G$  non-orienté une assignation d'une direction de communication pour chaque arête de  $G$ . Si  $O$  est une orientation du graphe  $G$ , alors pour chaque arête  $e = \{u, v\} \in E(G)$ , on a  $O(e) \in \{(u, v), (v, u)\}$ . Étant donné une orientation  $O$  d'un graphe  $G = (V, E)$ , pour toute arête  $e \in E$  l'orientation  $O(e) = (u, v)$  indique que le sommet  $u$  est la queue de l'arête  $e$  et que  $v$  est la tête. Une coloration d'arêtes orientées d'un graphe  $G$  est une paire  $(C, O)$ , où  $C$  correspond à une coloration des arêtes et  $O$  une orientation de  $G$ , telles que pour toutes paires d'arêtes  $e_1 = \{u_1, v_1\}$  et  $e_2 = \{u_2, v_2\}$  adjacentes ayant la même couleur, l'orientation  $O$  satisfait la condition suivante : Pour toute paire de sommets  $x \in e_1$  et  $y \in e_2$  si  $x$  et  $y$  sont voisins alors soit  $x$  et  $y$  sont des queues de  $e_1$  et  $e_2$  respectivement, ou  $x$  et  $y$  sont des têtes de  $e_1$  et  $e_2$  respectivement [Herman et al. 2006].

Comme expliqué précédemment et illustré à la Figure 4.5, le problème de coloration d'arêtes orientées peut permettre de réduire le nombre de slots nécessaires pour collecter les données issues des nœuds capteurs d'un réseau. Cependant, ce problème ne permet pas de prendre en compte tous les conflits secondaires qui peuvent se produire durant les transmissions de données dans le réseau. D'autre part, il n'est pas forcément nécessaire d'utiliser tous les liens de communications du réseau pour la transmission des données, c'est-à-dire colorer toutes les arêtes du graphe. En effet, bien souvent une topologie est construite sélectionnant une partie des liens de communications à utiliser pour les transmissions de données. Ainsi, nous introduisons dans la Section 4.2.3 un problème de coloration d'arêtes spécifiquement dédié à la prise en compte de ces caractéristiques. Il est à noter que nous avons présenté plusieurs problèmes de coloration de graphe assez standards, le lecteur intéressé par des problèmes de coloration à distance 2 plus spécifiques peut consulter le vaste état de l'art établi par Calamoneri Calamoneri [2011]. De plus, Drira et al. [2013] ont étudié le problème de coloration à distance  $l$  pour différentes classes de graphes (les hypercubes, les grilles carrées à deux dimensions ou encore les graphes de puissance issus des arbres et des cycles) et ont établi l'exact nombre ou une borne supérieure sur le nombre de couleurs nécessaire.

### 4.2.3 Problème à résoudre et complexité

Après avoir présenté le lien entre l'allocation des slots et la coloration d'un graphe, nous allons introduire le problème qui nous intéresse et que nous chercherons ensuite à résoudre. Nous étudierons également sa complexité de résolution dans plusieurs situations.

#### 4.2.3.1 Définition du problème

Nous nous intéressons à un problème d'allocation de slots aux nœuds du réseau de capteurs de sorte à partager l'accès au médium sans fil sans collision pour collecter des données issues des nœuds capteurs. Plus précisément, nous désirons minimiser la durée de la collecte de données tout en éliminant les collisions durant la transmission des données. Ce problème de réservation de ressources peut être formulé comme un problème de coloration à distance 2 des arêtes dans un graphe, avec des contraintes supplémentaires associées aux

interférences entre les communications.

Pour cela, nous considérons un réseau de capteurs composé d'un puits (nœud de collecte), noté  $s$ , et un ensemble de nœuds capteurs ayant les mêmes caractéristiques. Notamment, les nœuds de ce réseau communiquent à l'aide de communications sans fil via une même et unique fréquence radio et chaque nœud possède le même rayon de communication. Ce réseau est modélisé dans la suite par un graphe  $G = (V, E)$  comme présenté en Section 4.2, et nous supposons qu'un arbre couvrant orienté  $T = (V, E_T)$  de  $G$  enraciné en  $s$  est utilisé pour la collecte des données. Ainsi, nous distinguons deux types d'arêtes celles appartenant à  $E_T$  et les autres arêtes  $E - E_T$  en modélisant une partie des interférences (conflits secondaires) engendrées durant les communications lors de la collecte des données.

**Problème 1 (Ordonnancement sans collision des transmissions)**

*Étant donné un graphe quelconque  $G = (V, E)$ , un arbre orienté  $T = (V, E_T)$  de  $G$  et un entier  $q > 0$ , existe-t-il une affectation d'intervalles de temps (ou slots) sur les arêtes de  $E_T$  telle qu'il n'y ait aucun conflit primaire ou secondaire et que la durée totale de l'ordonnancement (makespan) soit inférieure ou égale à  $q$  ?*

**4.2.3.2 Preuves de complexité**

Dans cette section, nous allons étudier la complexité du problème d'ordonnancement sans collision des transmissions (Problème 1). Une généralisation du Problème 1 a été étudiée dans [Ghosh et al. 2011, 2009]. Ghosh et al. ont considéré une variante du Problème 1 dans lequel les capteurs disposent de plusieurs fréquences pour communiquer. Les auteurs ont montré que cette variante du problème est NP-difficile à résoudre et ont proposé des heuristiques avec garantie de performance dans le cas où le nombre de fréquences est borné ou non.

Bien que l'un des résultats de complexité démontré dans [Ghosh et al. 2011, 2009] puisse être appliqué au Problème 1, nous proposons dans cette section une analyse simplifiée et nous montrons également dans quel cas le Problème 1 est facile à résoudre.

**Théorème 1**

*Étant donné un graphe  $G = (V, E)$  modélisant le réseau et un arbre couvrant orienté  $T = (V, E_T)$  enraciné en  $s$  tel que  $E - E_T \neq \emptyset$ . Le Problème 1 est NP-complet.*

**Preuve.** Il est facile de vérifier que le Problème 1 est dans NP. Étant donné une solution quelconque pour ce problème, on peut vérifier en temps polynomial que (i) au plus  $q$  intervalles de temps sont affectés aux arêtes de  $E_T$ , (ii) un intervalle de temps différent est assigné à toute paire d'arêtes  $e_1, e_2 \in E_T$  à distance inférieure ou égale à 2, et (iii) aucune collision n'est générée par l'affectation des  $q$  intervalles de temps.

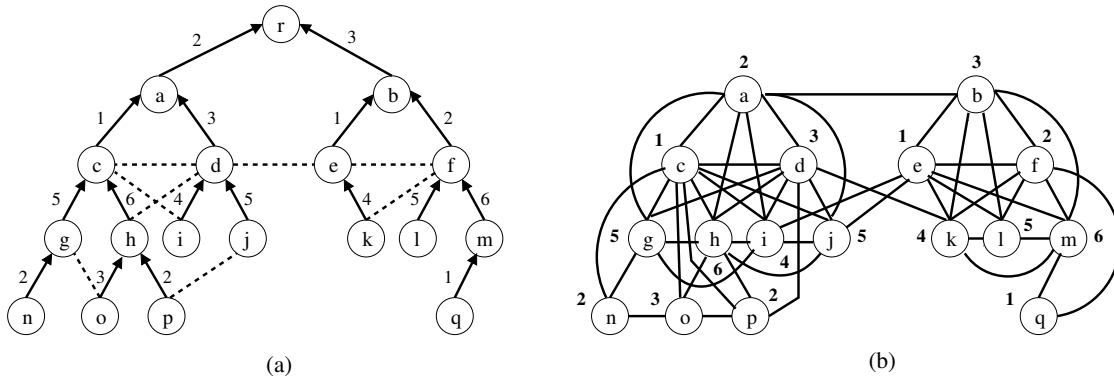


FIGURE 4.6 – Illustration de la réduction du Problème 1 au problème de coloration de sommets, (a) instance pour le Problème 1 où les flèches représentent l'arbre orienté  $T$ , les arêtes en pointillés les collisions secondaires, et le nombre sur chaque arête correspond à l'intervalle de temps affecté, et (b) instance pour le problème de coloration de sommets où le nombre à côté de chaque sommet correspond à sa couleur.

Pour montrer la NP-difficulté, nous allons réduire toute instance  $G' = (V', E')$  pour le problème de coloration de sommets en une instance  $G = (V, E)$  pour le Problème 1. Cette réduction est illustrée par la Figure 4.6. Considérons tout d'abord la transformation de toute instance  $G$  du Problème 1 en une instance  $G'$  pour le problème de coloration de sommets. Pour construire  $G' = (V', E')$ , nous allons créer un sommet  $v' \in V'$  pour chaque sommet  $v \in V, v \neq s$ . Ensuite, pour chaque arête  $e = (v_1, v_2) \in E_T$  telles que  $v_1, v_2 \neq s$ , il faut créer une arête  $e' = \{v'_1, v'_2\} \in E'$ . De plus, chaque conflit primaire sera modélisé par une arête dans  $G'$ . Ainsi, pour toute paire d'arêtes  $e = (v_1, v_2), f = (u_1, u_2) \in E_T$  telles que  $v_2 = u_2$ , l'arête  $\{v'_1, u'_1\}$  est ajoutée à  $E'$ . De même, pour toute paire d'arêtes  $e = (v_1, v_2), f = (u_1, u_2) \in E_T$  telles que  $v_2, u_2 \neq s$  et  $v_2 = u_1$  (ou resp.  $u_2 = v_1$ ) alors l'arête  $\{v'_1, u'_2\}$  (ou resp.  $\{v'_2, u'_1\}$ ) est ajoutée à  $E'$  respectivement. Enfin, il faut prendre en compte les interférences issues des arêtes hors de l'arbre de collecte de données  $T$ . Pour chaque arête  $e = (v_1, v_2) \in E \setminus E_T$  nous noterons  $Ch_{v_1}$  (resp.  $Ch_{v_2}$ ) l'ensemble des sommets

fils de  $v_1$  (resp.  $v_2$ ) dans  $T$ . Nous ajoutons à  $E'$  une arête  $\{v_1, u\}$  (resp.  $\{u, v_2\}$ ) pour chaque sommet  $u \in Ch_{v_2}$  (resp.  $u \in Ch_{v_1}$ ). Cette transformation est réalisée en temps polynomial fonction du nombre de sommets et d'arêtes dans  $G$ .

Nous allons montrer qu'il existe une solution pour le problème de coloration des sommets de  $G'$  avec au plus  $q$  couleurs différentes si et seulement s'il existe un ordonnancement sans collision dans  $G$  utilisant au plus  $q$  intervalles de temps. Supposons tout d'abord qu'il existe une coloration des sommets de  $G'$  ne nécessitant pas plus de  $q$  couleurs différentes. Soient  $c(v')$  la couleur du sommet  $v' \in V'$ ,  $v \in V$  le sommet associé à  $v'$  dans  $G$ , et  $itv(e)$  l'intervalle associé à l'arc  $e \in E_T$ . Pour chaque arc  $e = (v, u) \in E_T, v \neq u$ , on affecte l'intervalle  $itv(e) = c(v')$ . La coloration des arcs de  $T$  ainsi obtenue constitue bien une solution au Problème 1. Tout d'abord, on peut observer que chaque arc  $e = (v, u) \in E_T$  possède un intervalle  $itv(e)$  différent des arcs voisins dans  $E_T$ , c'est-à-dire  $itv(e) \neq itv(f)$ ,  $\forall f \in E_T$  tel que  $f = (u, w), w \neq v$ , ou  $f = (w, v), w \neq u$ . De plus, des intervalles différents sont affectés aux arcs  $itv(v, w)$  et  $itv(u, x)$  tels que  $v, u \in V$  sont à distance 2 dans  $T$ , car il existe une arête  $e' = (v', u')$  dans  $G'$  avec  $v', u' \in V'$  les sommets correspondant respectivement à  $v, u \in V$ . Par conséquent, les intervalles affectés aux arcs de  $T$  prennent en compte les conflits primaires le long des chemins dans  $T$ . D'autre part, toute paire d'arcs  $e_1 = (w, v), e_2 = (x, v) \in E_T$  vérifie  $itv(e_1) \neq itv(e_2)$  car il existe une arête  $\{w', x'\} \in E'$  dans  $G'$  entre les sommets  $w', x' \in V'$  correspondant respectivement à  $w, x \in V$ . Ainsi, les conflits primaires entre les sommets fils de chaque sommet de  $T$  sont considérés. Enfin, toute paire de sommets  $v_1, v_2 \in V$  telle que  $\{v_1, v_2\} \in E - E_T$  (conflits secondaires dans  $G$ ) vérifie que  $itv(v_1, u) \neq itv(w, v_2)$  pour chaque sommet fils  $w \in Ch_{v_2}$  dans  $T$ . Cela est dû à l'existence d'une arête  $\{v'_1, w'\}$  dans  $G'$ , avec  $v'_1, w' \in V'$  les sommets associés respectivement à  $v_1, w \in V$ . Ainsi,  $q$  intervalles différents ont été affectés aux arcs de  $T$  définissant un ordonnancement sans collision.

Considérons maintenant le sens inverse. Supposons qu'il existe un ordonnancement sans collision de  $G$  ne nécessitant pas plus de  $q$  intervalles de temps. Soit l'intervalle de temps  $itv(e)$  associé à chaque arc  $e = (v, u) \in E_T$  dans  $G$ . Pour chaque sommet  $v' \in V'$  associé au sommet  $v \in V, v \neq u$ , on affecte la couleur  $c(v') = itv(e)$ . Nous pouvons remarquer que toute solution pour le Problème 1 constitue une solution pour le problème de coloration

de sommets à distance 2, car des intervalles de temps différents sont affectés pour tout ensemble de trois arcs consécutifs dans  $T$ . Il en va de même pour tous les arcs partageant le même sommet extrémité. Mis à part les conflits secondaires, comme une coloration des sommets à distance 2 constitue également une coloration propre des sommets, une solution pour le Problème 1 de  $T$  définit donc une coloration des sommets de  $G'$ . Considérons les conflits secondaires. Pour chaque paire d'arcs  $e_1 = (v_1, u_1), e_2 = (v_2, u_2) \in E_T$  tels que  $\{v_1, u_2\} \in E - E_T$ , comme  $itv(v_1, u_1) \neq itv(v_2, u_2)$  dans  $T$  on a donc  $c(v'_1) \neq c(v'_2)$  dans  $G'$  avec  $v'_1, v'_2 \in V'$  correspondant à  $v_1, v_2 \in V$ , car par construction de  $G'$  on a  $\{v_1, v_2\} \in E'$ . Ainsi, on obtient une coloration des sommets de  $G'$  utilisant au plus  $q$  couleurs différentes.  $\square$

Nous allons maintenant nous intéresser à un cas pour lequel une solution optimale pour le problème d'ordonnancement sans collision (Problème 1) peut être calculée en temps polynomial.

**Lemme 1**

*Étant donné un graphe  $G = (V, E)$  modélisant le réseau et un arbre couvrant orienté  $T = (V, E_T)$  enraciné en  $s$  tel que  $E - E_T = \emptyset$ . Le Problème 1 est polynomial.*

**Preuve.** Nous considérons ici que  $G$  constitue un arbre couvrant, c'est-à-dire que  $G = T$ ,  $G$  appartient donc à la classe des graphes cordaux. Un graphe cordal est un graphe dont tout cycle de taille supérieure ou égale à 4 possède une arête reliant deux de ses sommets non-adjacents. Comme il n'y a pas de conflits secondaires, une solution pour le Problème 1 revient à résoudre le problème de coloration de sommets à distance 2. De plus, comme indiqué par Mahdian [Mahdian 2002] le problème de coloration de sommets à distance 2 est équivalent au problème de coloration de sommets dans le graphe de puissance 2 du line graph (ou *graphe dual*) associé à  $G$ . Cameron [Cameron 1989] a démontré que le graphe de puissance 2 du line graph d'un graphe cordal reste cordal. De plus, Gavril [Gavril 1972] a prouvé que le problème de coloration de sommets est polynomial à résoudre dans les graphes cordaux. Par conséquent, on en déduit que le Problème 1 est polynomial à résoudre si  $G$  ne contient pas de conflits secondaires.

Nous pouvons aussi arriver au même résultat en reprenant la construction du graphe

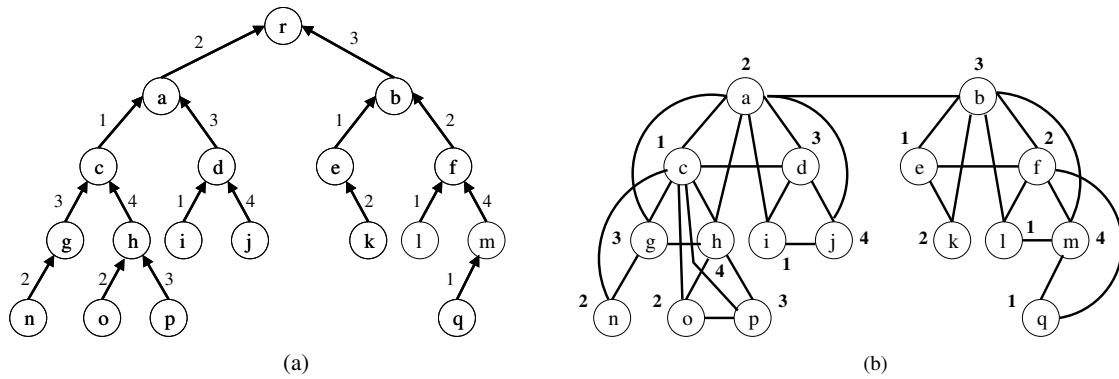


FIGURE 4.7 – Illustration de la réduction du Problème 1 au problème de coloration de sommets dans le cas où le problème est polynomial à résoudre de façon optimale, (a) instance pour le Problème 1 où les flèches représentent l'arbre orienté  $T$  et le nombre sur chaque arête correspond à l'intervalle de temps affecté, et (b) instance pour le problème de coloration de sommets où le nombre à côté de chaque sommet correspond à sa couleur.

$G'$  pour le problème de coloration de sommets décrit dans la preuve du Théorème 1. Contrairement au cas considéré dans le Théorème 1, le graphe  $G'$  construit contient ici moins d'arêtes car aucune arête associée aux conflits secondaires n'est ajoutée à  $E'$ . La Figure 4.7 illustre le graphe  $G'$  obtenu associé au graphe  $G = T$ . Nous pouvons remarquer que le graphe  $G'$  est un graphe cordal et pouvons donc utiliser le résultat de Gavril [Gavril 1972] pour conclure.  $\square$

Après avoir défini le problème que nous cherchons à résoudre, nous allons présenter dans la suite de ce chapitre deux algorithmes distribués que nous proposons pour résoudre le problème d'ordonnement sans collision (Problème 1).

### 4.3 Algorithme distribué DD2EC

Avant de présenter le premier algorithme distribué DD2EC (Distributed Distance-2 Edge Coloring) que nous proposons, nous allons décrire dans un premier temps plusieurs travaux de la littérature assez proches et associés au problème d'ordonnement sans collision que nous cherchons à résoudre.



### 4.3.1 État de l'art

L'affectation de slots aux liens de communication d'un réseau sans fil correspond à un problème de planification de l'ordonnancement des liens de communication (appelé « link scheduling problem »). Plusieurs algorithmes ont été proposés ayant pour but de maximiser le nombre de transmissions simultanées activables sans produire de collision.

[Ghosh et al. 2011] proposent deux heuristiques centralisées, ayant une connaissance globale de la topologie du réseau, basés sur une approche TDMA pour affecter des slots aux liens de communication. Dans ce travail, les auteurs considèrent que plusieurs canaux sont à la disposition des nœuds pour communiquer. De plus, une arborescence couvrante  $T$  enracinée au nœud puits  $s$  est considérée et l'objectif est de réduire le nombre de slots utilisés pour collecter les données en tenant compte des collisions dues aux conflits primaires et secondaires. La première heuristique concerne la situation où le nombre de canaux n'est pas contraint alors que la seconde heuristique prendra en compte un nombre de canaux fixé en entrée. La première heuristique construit un graphe des conflits  $G_C$  en partant de l'arbre couvrant pour éliminer les conflits. Comme illustré à la Figure 4.8, ce graphe possède un sommet par nœud récepteur dans l'arborescence, et l'ensemble des arêtes est composé d'une part des arêtes de l'arbre couvrant et d'autre part d'arêtes ajoutées pour modéliser les conflits primaires et secondaires. Ainsi, ce graphe des conflits contient deux types d'arêtes : les arêtes appartenant à l'arbre couvrant (auxquelles il faudra affecter des slots) et les arêtes de conflits à prendre en considération mais sans affectation de slot.

Dans une première phase, l'heuristique affecte des canaux aux nœuds récepteurs dans l'arbre de sorte à supprimer tous les conflits secondaires. Pour cela, la plus petite fréquence est allouée aux nœuds récepteurs différents des nœuds voisins et en suivant un ordre décroissant des degrés. Enfin, dans une seconde phase le plus petit slot disponible est affecté à chaque arête de l'arbre en suivant un parcours en largeur de l'arbre et de sorte à ce que deux arêtes adjacentes aient des slots différents. Cette heuristique nécessite au plus  $\Delta(G_C) + 1$  canaux et au plus  $\Delta(T)$  slots, avec  $\Delta(G_C) + 1$  le degré maximum dans le graphe des conflits et  $\Delta(T)$  le degré maximum dans l'arbre couvrant  $T$ . La seconde heuristique proposée dans [Ghosh et al. 2011] est inspirée de la première heuristique à la différence

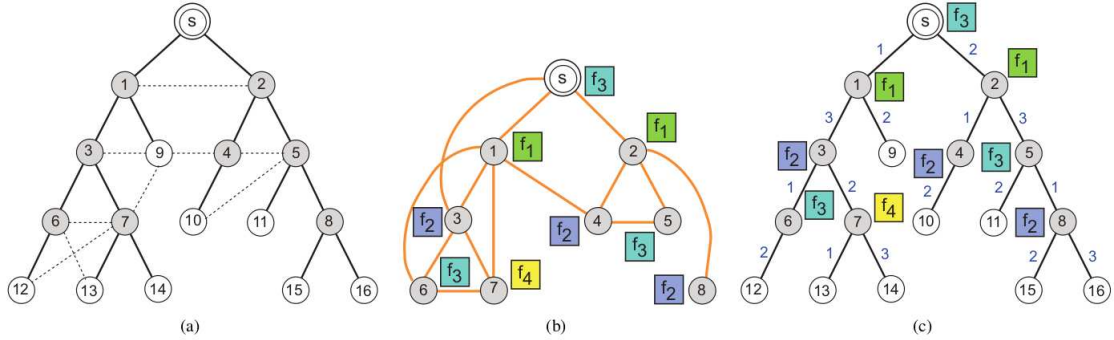


FIGURE 4.8 – Illustration de la première heuristique proposée dans [Ghosh et al. 2011] et extraite de l'article. (a) Le graphe initial modélisant le réseau avec en gris les nœuds récepteurs, en noir les arêtes de l'arbre et en pointillées les arêtes d'interférence, (b) l'affectation des canaux en écoute pour les nœuds récepteurs dans le graphe de conflit construit composé d'arêtes supplémentaires par rapport à l'arbre initial pour modéliser les conflits primaires et secondaires entre nœuds récepteurs, et (c) l'affectation des slots indiquée sur chaque arête de l'arbre.

qu'un découpage de la zone 2D contenant tous les nœuds est effectué suivant une grille carrée et l'allocation des canaux est effectuée indépendamment dans chaque cellule de la grille.

Contrairement au travail de [Ghosh et al. 2011], [Gandham et al. 2008a] proposent un algorithme distribué. Cet article calcule dans un premier temps une coloration des arêtes en utilisant au plus  $\Delta + 1$  couleurs, puis dans une seconde phase chaque couleur est associée à un slot unique et les arêtes du réseau sont orientées en tenant compte de l'affectation des slots réalisée suite à la première phase de sorte à supprimer les problèmes de terminaux cachés et exposés. Pour calculer l'orientation des arêtes, un algorithme distribué de parcours en profondeur du graphe est utilisé. Cependant, cette approche fonctionne uniquement s'il y a un nombre pair d'arêtes de même couleur dans chaque cycle du graphe fourni en entrée. La complexité en temps et en messages de cet algorithme est de  $O(n\Delta^2 + n^2m)$ , avec  $n$  la taille du graphe,  $m$  le nombre d'arêtes, et  $\Delta$  le degré maximum dans le graphe. [Gandham et al. 2008a] montrent que chaque arbre couvrant  $T$  admet une coloration d'arêtes orientées de  $\Delta(T)$  couleurs avec  $\Delta(T)$  le degré maximum dans  $T$ , tandis que certains arbres utilisent  $2\Delta(T) - 1$  couleurs pour toute coloration à distance 2.

[Ma et al. 2009] proposent un algorithme centralisé et un algorithme distribué pour une

variante du problème de coloration d'arêtes à distance 2 pour des topologies arborescentes. Les auteurs cherchent à construire des colorations telles que des couleurs consécutives soient affectées aux arêtes adjacentes de chaque sommet. L'intérêt pour ce type de coloration est motivé par les gains en énergie dû à un nombre plus faible de changement d'état. En effet, le changement d'état (actif/passif) d'un nœud induit un coût en énergie. Les auteurs démontrent que ce problème est NP-difficile et proposent un algorithme distribué avec un facteur d'approximation constant par rapport à l'optimal pour le nombre de couleurs utilisé. Ils suivent une approche aléatoire pour affecter les couleurs, les nœuds sont en compétition pour l'accès au médium radio. Si un nœud obtient le médium alors il attribue à ses arêtes adjacentes dans l'arbre le plus petit ensemble de couleurs non utilisé dans son voisinage et cette allocation à tous ses voisins, sinon il attend un temps aléatoire avant de réitérer cette procédure.

[Mahfoudh et al. 2010] ont proposé un algorithme distribué, dénommé SERENA, qui planifie l'activité des nœuds dans le réseau en construisant une coloration des sommets à distance 2 ou 3. La distance de coloration dépend du mode de fonctionnement considéré, une coloration à distance 3 est utilisée si on désire transmettre des accusés de réception immédiatement après la réception d'un paquet de données, sinon une coloration à distance 2 sera utilisée. Suite au mode de fonctionnement choisi, une priorité de coloration est attribuée à chaque nœud en fonction du nombre de nœuds dans son voisinage à distance 2 ou 3. Il est à noter que chaque nœud envoie périodiquement un message à ses voisins à 1 saut pour permettre l'échange des listes de voisins, des priorités des nœuds et des couleurs déjà utilisées. Grâce à ces informations, une coloration est effectuée par ordre décroissante des priorités en attribuant la plus petite couleur non affectée dans le voisinage à distance 2 ou 3. Dans le cas de structures hiérarchiques et acycliques, chaque nœud choisit la plus petite couleur non utilisée à distance 2 ou 3 et supérieure à la couleur de son parent. Les auteurs ont évalué le nombre de couleurs utilisées par SERENA en faisant varier la densité dans le réseau. Le nombre de couleurs varie de 0,4 et 1 fois le nombre de nœuds suivant la taille du réseau. [Amdouni et al. 2011] s'intéressent au problème de coloration des sommets à distance  $h \geq 1$  pour l'ordonnancement des transmissions suivant divers rayons de transmissions dans les grilles carrées. Dans ce travail, le problème est démontré comme

étant NP-complet, puis le nombre exact de couleurs pour plusieurs rayons de transmission est prouvé dans les grilles. Cela définit des patrons qui peuvent être répétés pour colorer tous les nœuds de la grille. Enfin, les auteurs comparent ces résultats théoriques avec les solutions obtenues par l'heuristique distribuée SERENA [Mahfoudh et al. 2010] pour différentes tailles de grille et de rayons de transmission.

[Grable and Panconesi 1997] ont proposé un algorithme distribué de coloration d'arêtes avec une faible complexité en temps. Cet algorithme calcule de façon non déterministe une coloration utilisant au plus  $(1 + \epsilon)\Delta$  couleurs en temps  $O(\text{polylog}(n))$ . [Herman et al. 2006] se sont intéressés aux topologies arborescentes et ont étendu l'approche de [Grable and Panconesi 1997] pour prendre en compte les problèmes de terminaux cachés et exposés. Pour cela, ils ont proposé un algorithme distribué probabiliste pour résoudre le problème de coloration d'arêtes orientées dans les arbres. Leur approche exécute dans une première phase l'algorithme de [Grable and Panconesi 1997] pour obtenir une coloration d'arêtes, puis dans une deuxième phase chaque arête de l'arbre est choisie avec probabilité  $p = 0,5$  pour être re-colorée. Enfin, dans une dernière phase une orientation est attribuée à chaque arête en tenant compte des couleurs affectées aux arêtes de l'arbre lors des deux premières phases. Cet algorithme établit avec forte probabilité une coloration d'arêtes orientées pour tout arbre en utilisant au plus  $(1 + \epsilon)\Delta$  couleurs et en temps  $O(\text{polylog}(n))$ . Il est à noter que cela ne concerne que des topologies acycliques. [Cheng and Yin 2007] ont proposé un algorithme centralisé pour le problème de coloration d'arêtes orientées pour une topologie quelconque. Cet algorithme fonctionne de façon itérative pour affecter les diverses couleurs à un graphe quelconque non orienté. À chaque itération, un arbre orienté est construit à partir d'un arc sélectionné aléatoirement (non encore coloré). L'algorithme colore un maximum d'arcs sans conflit de la couleur correspondant à l'itération courante. Les auteurs évaluent les performances atteintes par leur algorithme à l'aide de simulations pour diverses classes de graphes et avec des tailles allant de 100 à 1500 sommets. Les résultats de simulations tendent à montrer que le nombre de couleurs reste proche du degré maximum  $\Delta$  du graphe.

Récemment, [Soua et al. 2015] ont proposé un algorithme distribué, dénommé DiSCA, dont l'objectif est de minimiser le temps de collecte des données dans un réseau de capteurs en limitant les collisions. Ce travail considère un contexte identique à celui considéré

par [Ghosh et al. 2011]. Tout d’abord, la topologie du réseau est quelconque et un arbre couvrant  $T$  existant enraciné au puits  $s$  est utilisé pour la collecte des données. De plus, plusieurs canaux sont utilisés conjointement à une affectation de slots aux transmissions pour éliminer les conflits primaires et secondaires lors de la transmission des données. Cependant, contrairement à [Ghosh et al. 2011] DiSCA prend en compte le trafic des nœuds dans la planification de leur activité. Ainsi, chaque nœud possède un niveau de priorité statique égal à la quantité de paquets de données à transmettre, associée aux données qu’il génère et à celles issues de ses descendants dans son sous-arbre. De plus, chaque nœud a connaissance d’un ensemble de nœuds en conflits dans  $G$  avec lesquels des conflits au niveau des transmissions peuvent avoir lieu. Pour chaque paquet de données à transmettre, chaque nœud exécute une itération de l’algorithme DiSCA qui affecte la plus petite paire de slot et canal disponible, c’est-à-dire non déjà utilisés par un nœud dans son voisinage à 1 saut ou voisinage en conflits. Cette affectation est effectuée par ordre croissant des priorités, en cas d’égalité entre une paire de nœuds la profondeur dans l’arbre  $T$  et les identifiants sont utilisés pour les ordonner. Les simulations menées par les auteurs pour évaluer les performances atteintes par DiSCA indiquent que le nombre de slots affectés est proche de l’optimal (entre 10% et 20% supérieur).

Le Tableau 4.2 résume les points importants associés aux algorithmes de coloration cités ci-dessus. Tous ces algorithmes affectent une couleur à tous les liens du réseau même si ceux-ci ne sont pas utilisés pour l’échange d’information entre voisins, excepté les approches proposées dans [Ghosh et al. 2011; Mahfoudh et al. 2010; Soua et al. 2015]. L’algorithme proposé par [Mahfoudh et al. 2010] établit une coloration de sommets à distance 2 ou 3 sans distinction de liens dédiés à la collecte de données. Contrairement aux autres travaux cités ci-dessus, [Ghosh et al. 2011] considèrent deux types de liens dans le graphe : les liens de communication et les liens de conflits dû aux interférences. Le problème considéré [Ghosh et al. 2011] et [Soua et al. 2015] est proche du problème d’ordonnancement sans collision (Problème 1) qui nous intéresse, ce qui les différencie est le nombre de canaux disponibles. En effet, dans notre problème (Problème 1) les nœuds ne disposent que d’un seul canal qui doit être partagé par tous les nœuds du réseau.

### 4.3. ALGORITHME DISTRIBUÉ DD2EC

---

Références	Problème de coloration	Nbre de couleurs	Topologie	Type d'algorithme
[Grable and Panconesi 1997]	Coloration d'arêtes	$(1 + \epsilon)\Delta$	quelconque	distribué
[Ghosh et al. 2011]	Coloration d'arêtes à distance 2	$\Delta + 1$	arbre avec conflits	centralisé
[Gandham et al. 2008a]	Coloration d'arêtes à distance 2	$2\Delta - 1$	quelconque	distribué
[Ma et al. 2009]	Coloration d'arêtes à distance 2	-	arbre	centralisé + distribué
[Mahfoudh et al. 2010]	Coloration de sommets à distance 2 ou 3	-	quelconque	distribué
[Herman et al. 2006]	Coloration orientée d'arêtes	$(1 + \epsilon)\Delta$	arbre	distribué
[Cheng and Yin 2007]	Coloration orientée d'arêtes	-	quelconque	centralisé
[Soua et al. 2015]	Coloration de sommets à distance 2	-	quelconque	distribué

TABLE 4.2 – Tableau de comparaison

Dans la suite, nous allons définir formellement un système distribué puis nous présenterons l'algorithme distribué DD2EC que nous proposons pour résoudre le problème d'ordonnancement sans collision (Problème 1).

#### 4.3.2 Modélisation d'un système distribué

Un système distribué est composé d'un ensemble de nœuds pouvant communiquer entre eux. Les nœuds du système exécutent un algorithme local, en général identique pour tous les nœuds du système, ayant accès à des informations partielles du système (bornées à un rayon donné par rapport à chaque nœud). L'*état* d'un nœud est défini par les valeurs des variables utilisées par son algorithme local. Une *configuration* du système est le produit des états des nœuds du système. Un algorithme local est un système de transitions  $S = (\mathcal{C}, \rightarrow, \mathcal{I})$ , où  $\mathcal{C}$  est l'ensemble des configurations du système,  $\rightarrow$  est une transition binaire (ou action) sur  $\mathcal{C}$  et  $\mathcal{I}$  est une configuration initiale du système ( $\mathcal{I} \subseteq \mathcal{C}$ ). Deux modèles sont communément utilisés pour les communications : (i) le *modèle par passage de messages* dans lequel les nœuds s'échangent des informations à l'aide de messages de taille bornée qui peuvent être

envoyés à un ou plusieurs voisins dans le réseau, et (ii) le *modèle à états* dans lequel les nœuds communiquent avec leurs voisins dans le réseau à travers des registres partagés en lecture. Un algorithme *distribué* est la collection des algorithmes locaux exécutés par les nœuds du système.

Une exécution distribuée est décomposée en pas de calcul, composées de trois phases exécutées de façon atomique : (i) un nœud évalue ses actions, (ii) un ordonnanceur (ou *démon*) choisit les nœuds à activer, (iii) chaque nœud choisit et exécute l'action la plus prioritaire de son algorithme local. Lorsque ces trois phases sont terminées un nouveau pas de calcul commence. Un démon (ou *ordonnanceur*) est dit *distribué* s'il choisit un sous-ensemble des nœuds du système à chaque pas de calcul. Une exécution est dite *maximale* si la séquence d'actions exécutées est finie (c'est-à-dire une configuration terminale est atteinte où aucune action n'est exécutable) ou alors infinie. Un système est dit *synchrone* si l'envoi et la réception des messages entre émetteurs et récepteurs sont coordonnées, c'est-à-dire forment une seule transition du système, dans le cas contraire le système est dit *asynchrone*.

La complexité en temps d'un algorithme distribué est défini à l'aide de la notion de *ronde de calcul*. Étant donnée une exécution maximale  $e$ , la première ronde  $e'$  de  $e$  est le plus petit préfixe de  $e$  contenant l'exécution d'une action de chaque nœud du système à partir de la configuration initiale. Soit  $e''$  le suffixe de  $e$  tel que  $e = e'e''$ . La seconde ronde de  $e$  correspond à la première ronde de  $e''$ . La complexité en messages d'un algorithme distribué correspond au nombre total de messages échangés durant une exécution maximale  $e$ , tandis que la complexité en mémoire est égale à la quantité maximale de mémoire utilisée par un algorithme pour tout nœud du système.

#### 4.3.3 Présentation de l'algorithme DD2EC

Dans cette section, nous présentons notre algorithme distribué DD2EC (Distributed Distance-2 Edge Coloring) pour résoudre le problème d'ordonnancement sans collision (Problème 1) défini précédemment. Ce problème sera modélisé comme un problème de coloration des arêtes d'un graphe  $G = (V, E)$  modélisant le réseau de capteurs. Plus précisément, on considère qu'une arborescence couvrante  $T = (V, E_T)$  orientée et enracinée

au nœud puits  $s$  est construite et nous cherchons une coloration orientée des arêtes dans  $E_T$  tenant compte des contraintes de conflits primaires et secondaires. Chaque nœud du graphe possède en entrée un ensemble statique de couleurs qui est utilisé pour la coloration des arêtes (ensemble identique pour tous les nœuds), noté  $\mathcal{C}$ .

L'algorithme DD2EC que nous proposons est exécuté sur chaque nœud du réseau et utilise le modèle par passage de messages pour la communication entre nœuds voisins. Une description formelle est donnée dans l'Algorithme 1. L'algorithme DD2EC doit affecter une couleur à chaque arête  $e \in E_T$  correspondant au slot à utiliser par les deux extrémités de  $e$  pour la transmission d'un paquet de données à travers  $e$ , en mode émission pour l'une et en mode réception pour l'autre suivant l'orientation associée à  $e$ . Ainsi, après exécution de l'algorithme chaque nœud aura la couleur (slot) à utiliser pour envoyer un paquet de données vers son parent dans  $T$  (en mode émission), et pour chacun de ses fils dans  $T$  la couleur (slot) à utiliser en réception pour recevoir un paquet de données envoyé par son fils qui utilisera la même couleur en mode émission.

La coloration des arêtes de l'arbre  $T$  est réalisée en partant du puits  $s$  et en suivant un ordre (ou *priorité*) de coloration. Cet ordre est basé sur les distances dans l'arbre  $T$  afin d'établir une priorité entre les arêtes à colorer. Deux arêtes au même niveau dans l'arbre qui ne sont pas en conflit, c'est-à-dire qui ne partagent pas une même extrémité, et ne sont pas voisines via une arête de conflit, peuvent être colorées en parallèle et sans ordre particulier. L'ordre de priorité suivi pour la coloration des arêtes est présenté ci-après. Nous allons définir l'ordre de coloration des arêtes de l'arbre  $T$  basé sur les distances dans  $T$  qui sera suivi par l'algorithme DD2EC. Cet ordre de coloration sera suivi à chaque niveau dans l'arbre  $T$  en partant du puits  $s$  jusqu'aux feuilles. L'ordre défini ci-dessous considère les conflits associés à chaque arête  $e \in E_T$  à colorer.

**Ordre de coloration :** Tout d'abord, on note  $d_T(u)$  la distance de  $u$  au puits  $s$  dans l'arbre  $T$ . En mode réception, un voisin  $q$  en conflit avec  $p$  est plus prioritaire soit si  $d_T(q) < d_T(p)$ , soit si  $d_T(q) = d_T(p)$  ou  $d_T(q) = d_T(p) + 1$  et le père de  $q$  possède un identifiant inférieur à celui du père de  $p$ , comme illustré dans la Figure 4.9. En mode émission, un voisin  $q$  en conflit avec  $p$  est plus prioritaire soit si  $d_T(q) < d_T(p) - 1$ , soit



si  $d_T(q) = d_T(p) - 1$  et l'identifiant de  $q$  est inférieur à celui du père de  $p$ , comme illustré dans la Figure 4.10.

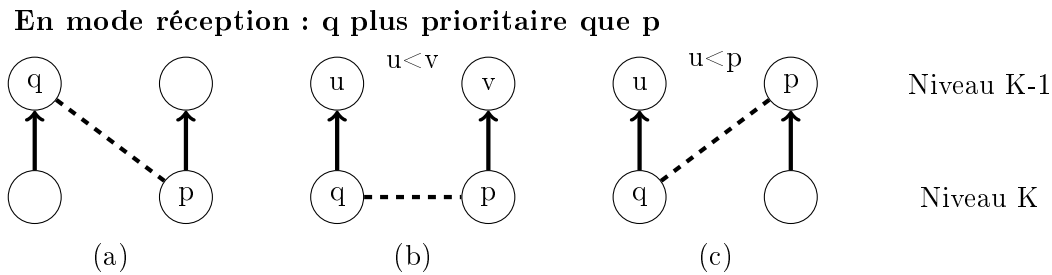


FIGURE 4.9 – Illustration des priorités en mode réception. (a) situation où  $d_T(q) < d_T(p)$ , (b) situation où  $d_T(q) = d_T(p)$  et l'identifiant du père de  $q$  est inférieur à celui du père de  $p$ , (c) situation où  $d_T(q) = d_T(p) + 1$  et l'identifiant du père de  $q$  est inférieur à celui du père de  $p$ .

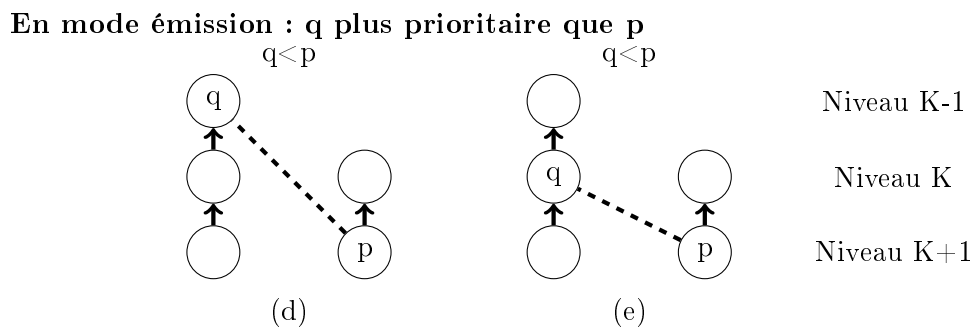


FIGURE 4.10 – Illustration des priorités en mode émission. (d) situation où  $d_T(q) < d_T(p) - 1$ , (e) situation où  $d_T(q) = d_T(p) - 1$  et l'identifiant de  $q$  est inférieur à celui du père de  $p$  dans  $T$ .

L'algorithme DD2EC utilisera l'ordre donné ci-dessus pour définir la priorité de coloration de chaque arc  $e \in E_T$  par rapport aux autres arcs  $e' \in E_T, e \neq e'$ , en fonction soit de leurs adjacences directes à  $e$  dans  $T$  soit à travers une arête de conflit dans  $E - E_T$ . Si une paire d'arcs  $e, e' \in E_T$  sont incomparables suivant l'ordre ci-dessus, alors ceux-ci peuvent être colorés indépendamment l'un de l'autre. Afin que l'ordre de coloration puisse prendre en compte les conflits entre arcs, les deux extrémités de chaque arc  $e$  de  $T$  vont tout d'abord indépendamment établir leur liste respectivement d'arcs dans  $T$  avec lesquels il y a un conflit (primaire ou secondaire). Ensuite, chaque nœud extrémité calculera les *couleurs autorisées* en retirant de la palette de couleurs initiales  $\mathcal{C}$  les couleurs déjà utilisées par les

arcs *en conflit* (c'est-à-dire voisins via une arête dans  $E - E_T$ ) plus prioritaires par rapport à  $e$ , et cette liste sera envoyée au nœud en tête de l'arc  $e$  par le nœud en queue de l'arc  $e$ . Enfin, lorsqu'un nœud extrémité en tête d'un arc a reçu la liste des couleurs autorisées pour chacun de ses fils dans  $T$  le nœud extrémité affecte une couleur en émission à chacun de ses fils en tenant compte de leurs couleurs autorisées respectives. Enfin, lorsqu'un nœud reçoit la couleur en émission attribuée par son père celui-ci la propage à tous ses voisins afin de les informer (en particulier ses voisins moins prioritaires) de la couleur qui lui a été affectée en émission.

Nous allons décrire plus en détail ci-après le fonctionnement de l'algorithme DD2EC dont une description formelle est donnée dans l'Algorithme 1.

#### 4.3.3.1 Description détaillée

Chaque nœud  $p \in V$  possède des informations locales en entrée de l'algorithme telles que :

- son voisinage dans le réseau, noté  $Neig_p$ ,
- son parent dans l'arbre, noté  $par_p$ ,
- l'ensemble de ses fils dans l'arbre, noté  $Child_p$ ,
- sa distance au nœud puits  $s$  dans l'arbre, notée  $dist_p$ .

De plus,  $p$  dispose des informations suivantes pour chacun de ses voisins  $q$  :

- du parent de  $q$ , noté  $par_q$ ,
- des fils de  $q$ , noté  $Child_q$ ,
- de la distance de  $q$  au puits  $s$ , noté  $dist_q$ .

Nous noterons  $NI(p)$  l'ensemble des voisins en conflit d'un nœud  $p \in V$  dans  $G$ , plus précisément  $NI(p) = Neig_p - (Child_p \cup \{par_p\})$ . De plus, l'algorithme DD2EC est exécuté sur chaque nœud  $p$  et utilise plusieurs variables : l'ensemble des couleurs autorisées en émission et réception sont stockées respectivement dans les variables  $ace_p$  et  $acr_p$ , le résultat de l'algorithme est stocké dans les variables  $ce_p$  et  $cr_p$  contenant respectivement la couleur d'émission vers le parent et les couleurs de réception pour chacun des fils dans

l'arbre, les variables  $rep\_diff_p$  et  $rep\_prop_p$  sont utilisées pour stocker les identifiants des voisins ayant envoyés des messages attendus par  $p$  pendant l'exécution de l'algorithme, enfin les variables  $prop_p$  et  $cu_p$  sont des variables temporaires utilisées pour le calcul local des couleurs.

Enfin, l'algorithme DD2EC utilise trois types de messages :

$\langle \mathbf{PropCE}, q, ece \rangle$  : est utilisé par un nœud  $q$  pour envoyer à son parent l'ensemble de ses couleurs  $ece$  autorisées en émission.

$\langle \mathbf{Diff}, q, ecr \rangle$  : est utilisé par chaque nœud  $q$  pour envoyer à tous ses fils l'ensemble  $ecr$  contenant les couleurs en réception qu'il a affectées au lien dans l'arbre avec chacun de ses fils.

$\langle \mathbf{DiffAck}, q, ce \rangle$  : est utilisé par un nœud  $q$  pour informer ses voisins que la couleur  $ce$  lui a été réservée en émission avec son parent.

Chaque nœud  $p \in V$  détermine dans chaque mode l'ensemble de ses voisins avec lesquels il est en conflit et qui sont plus prioritaires que lui dans l'ordre de coloration. La fonction  $PrioR(p)$  (resp.  $PrioE(p)$ ) retourne l'ensemble de ces nœuds à  $p$  dans le mode réception (resp. émission).

Initialement, pour tout nœud  $p$  aucune couleur n'est affectée en émission et en réception (i.e.,  $ce_p = cr_p = \emptyset$ ) et toutes les couleurs sont autorisées en émission et réception (i.e.,  $ace_p = acr_p = \mathcal{C}$ ). Ces variables seront mises à jour sur chaque nœud au fur et à mesure de l'exécution de l'algorithme D2EC.

À chaque niveau dans l'arbre, pour le mode réception un nœud  $p$  attend un message  $\langle PropCE \rangle$  envoyé par chacun de ses fils avec l'ensemble des couleurs autorisées en émission. Chaque fils  $q$  peut envoyer son message  $\langle PropCE \rangle$  dès qu'il a reçu un message  $\langle Diff \rangle$  de tous ses voisins plus prioritaires indiqués par la fonction  $PrioE(q)$ . Lorsque c'est le cas,  $q$  met à jour son ensemble de couleurs autorisées  $ace_q$  en éliminant la couleur contenue dans le message  $\langle Diff \rangle$  (voir bloc associé à l'événement de réception d'un message  $\langle Diff \rangle$  dans l'Algorithme 1). Au démarrage de l'algorithme, seuls les nœuds  $q$  avec  $PrioE(q) = \emptyset$  (hormis le puits  $s$ ) peuvent envoyer un message  $\langle PropCE \rangle$  à leur père. C'est le cas de tous les fils du nœud puits  $s$ .

**Algorithm 1**    Algorithme DD2EC

<p><b>Entrées :</b></p> <ul style="list-style-type: none"> <li>- <math>C</math> : palette de couleurs possibles;</li> <li>- <math>Neig_p</math> : ensemble ordonné des voisins de <math>p</math>;</li> <li>- <math>par_p</math> : parent de <math>p</math> dans l'arbre couvrant <math>T</math>;</li> <li>- <math>Child_p</math> : ensemble de fils de <math>p</math> dans l'arbre couvrant <math>T</math>;</li> <li>- <math>dist_p</math> : distance entre <math>p</math> et la racine <math>s</math> de l'arbre couvrant <math>T</math>;</li> <li>- <math>par_q</math> : parent dans l'arbre couvrant <math>T</math> du voisin <math>q</math> de <math>p</math>;</li> <li>- <math>Child_q</math> : ensemble de fils dans l'arbre couvrant <math>T</math> du voisin <math>q</math> de <math>p</math>;</li> <li>- <math>dist_q</math> : distance entre tout voisin <math>q</math> de <math>p</math> et la racine <math>s</math> de l'arbre couvrant <math>T</math>;</li> </ul> <p><b>Variables :</b></p> <ul style="list-style-type: none"> <li>- <math>ce_p</math> : variable stockant la couleur pour <math>p</math> en mode émission;</li> <li>- <math>cr_p</math> : variable stockant un ensemble de paires <math>(q, col)</math>, avec <math>col</math> la couleur de <math>p</math> en mode réception avec son fils <math>q</math>;</li> <li>- <math>ace_p</math> : variable stockant l'ensemble des couleurs autorisées pour <math>p</math> en mode émission;</li> <li>- <math>acr_p</math> : variable stockant l'ensemble des couleurs autorisées pour <math>p</math> en mode réception;</li> <li>- <math>rep\_diff_p</math> : variable stockant l'ensemble des voisins ayant envoyés un message <math>\langle Diff \rangle</math>;</li> <li>- <math>rep\_prop_p</math> : variable stockant l'ensemble des voisins ayant envoyés un message <math>\langle PropCE \rangle</math> or <math>\langle DiffAck \rangle</math>;</li> <li>- <math>prop_p</math> : variable stockant un ensemble de paires <math>(q, sc)</math>, avec <math>sc</math> la couleur en mode émission proposée par le fils <math>q</math> de <math>p</math>;</li> <li>- <math>cu_p</math> : variable stockant l'ensemble des couleurs utilisées par les fils de <math>p</math>;</li> </ul> <p>.....</p> <p><b>Macros :</b></p> <ul style="list-style-type: none"> <li>- <math>NI(p) = Neig_p - Child_p - \{par_p\}</math></li> <li>- <math>NCh(p) = \{q \in NI(p) :  Child_q  = 0\}</math></li> <li>- <math>P(p) = NI(p) - NCh(p)</math></li> <li>- <math>PrioR(p) = \{q \in NI(p) : (dist_q &lt; dist_p) \vee (dist_q = dist_p \wedge par_q \neq par_p \wedge par_q &lt; par_p) \vee (dist_q = dist_p + 1 \wedge par_q &lt; p)\}</math></li> <li>- <math>PrioE(p) = \{q \in P(p) : (dist_q &lt; dist_p - 1) \vee (dist_q = dist_p - 1 \wedge q &lt; par_p)\}</math></li> </ul> <p>.....</p> <p><b>Procédure :</b></p> <p>Children_Colors()</p> <p><b>if</b> <math>(ce_p \neq \emptyset) \vee (dist_p = 0)</math> <b>then</b></p> <p>  <b>for all</b> <math>e = (e_1, e_2) \in prop_p</math> par ordre croissant des valeurs de <math> e_2 </math> puis de <math>e_1</math> <b>do</b></p> <p>    <math>val := \min((acr_p \setminus cu_p) \cap e_2)</math>;</p> <p>    <math>cr_p := cr_p \cup \{(e_1, val)\}</math>;</p> <p>    <math>cu_p := cu_p \cup \{val\}</math>;</p> <p>  <b>end for</b></p> <p>  Envoyer le message <math>\langle Diff, p, cr_p \rangle</math> aux voisins de <math>p</math></p> <p><b>end if</b></p>	<p><b>Initialisation :</b></p> <p><math>ce_p := cr_p := prop_p := \emptyset</math>;</p> <p><math>ace_p := acr_p := C</math>;</p> <p><math>rep\_diff_p := rep\_prop_p := \emptyset</math>;</p> <p><b>if</b> <math>(PrioE(p) = \emptyset \wedge dist_p &gt; 0)</math> <b>then</b></p> <p>  Envoyer le message <math>\langle PropCE, p, ace_p \rangle</math> à <math>par_p</math></p> <p><b>end if</b></p> <p><b>À la réception</b> <math>\langle PropCE, q, ece \rangle</math> <b>du fils</b> <math>q</math> <b>de</b> <math>p</math> :</p> <p><math>prop_p := prop_p \cup \{(q, ece)\}</math>;</p> <p><math>rep\_prop_p := rep\_prop_p \cup \{q\}</math>;</p> <p><b>if</b> <math>rep\_prop_p = (Child_p \cup PrioR(p))</math> <b>then</b></p> <p>  <math>Children\_colors()</math>;</p> <p><b>end if</b></p> <p><b>À la réception</b> <math>\langle Diff, q, ecr \rangle</math> <b>du voisin</b> <math>q</math> <b>de</b> <math>p</math> :</p> <p><b>if</b> <math>q = par_p \vee q \in PrioE(p)</math> <b>then</b></p> <p>  <b>if</b> <math>q \neq par_p</math> <b>then</b></p> <p>    <b>for all</b> <math>e = (x, c) \in ecr</math> <b>do</b></p> <p>      <math>ace_p := ace_p - \{c\}</math>;</p> <p>    <b>end for</b></p> <p>    <math>rep\_diff_p := rep\_diff_p \cup \{q\}</math>;</p> <p>    <b>if</b> <math>rep\_diff_p = PrioE(p)</math> <b>then</b></p> <p>      Envoyer le message <math>\langle PropCE, p, ace_p \rangle</math> à <math>par_p</math></p> <p>    <b>end if</b></p> <p>  <b>else</b></p> <p>    <b>for all</b> <math>e = (x, c) \in ecr</math> <b>do</b></p> <p>      <b>if</b> <math>x = p</math> <b>then</b></p> <p>        <math>ce_p := c</math>;</p> <p>        <math>acr_p := acr_p - \{c\}</math>;</p> <p>      <b>else</b></p> <p>        <b>if</b> <math>x \in NI(p)</math> <b>then</b></p> <p>          <math>acr_p := acr_p - \{c\}</math>;</p> <p>        <b>end if</b></p> <p>    <b>end for</b></p> <p>    Envoyer le message <math>\langle DiffAck, p, ce_p \rangle</math> aux voisins de <math>p</math></p> <p>  <b>end if</b></p> <p>  <b>if</b></p> <p>    <math>Child_p \neq \emptyset \wedge rep\_prop_p = (Child_p \cup PrioR(p))</math></p> <p>  <b>then</b></p> <p>    <math>Children\_Colors()</math>;</p> <p>  <b>end if</b></p> <p><b>end if</b></p> <p><b>À la réception</b> <math>\langle DiffAck, q, ce \rangle</math> <b>du voisin</b> <math>q</math> <b>de</b> <math>p</math> :</p> <p><b>if</b> <math>q \in PrioR(p) \vee q = par_p</math> <b>then</b></p> <p>  <math>acr_p := acr_p - \{ce\}</math>;</p> <p>  <b>if</b> <math>Child_p \neq \emptyset \wedge q \neq par_p</math> <b>then</b></p> <p>    <math>rep\_prop_p := rep\_prop_p \cup \{q\}</math>;</p> <p>    <b>if</b> <math>rep\_prop_p = (Child_p \cup PrioR(p))</math> <b>then</b></p> <p>      <math>Children\_Colors()</math>;</p> <p>    <b>end if</b></p> <p>  <b>end if</b></p> <p><b>end if</b></p>
---	---

Pendant ce temps, le nœud parent  $p$  attend la réception du message  $\langle DiffAck \rangle$  de la part de tous ses voisins plus prioritaires que lui en mode réception indiqué par la fonction  $PrioR(p)$ , afin de mettre à jour son ensemble de couleurs autorisées en réception  $acr_p$ . Lorsqu'un nœud  $p$  a reçu tous les messages  $\langle PropCE \rangle$  de la part de tous ses fils et tous les messages  $\langle DiffAck \rangle$  de la part de tous ses voisins plus prioritaires, à ce moment  $p$  peut attribuer des couleurs à ses fils en utilisant la procédure  $Children\_Colors()$ . Cette procédure affecte de façon gloutonne une couleur à chacun de ses fils  $q$  en les considérant dans l'ordre croissant du nombre de couleurs autorisées  $ace_q$ . Une fois que l'opération d'affectation de couleurs est terminée, le nœud  $p$  envoie un message  $\langle Diff \rangle$  à ses voisins, en particulier à ses fils, contenant l'ensemble des couleurs attribuées. Lorsqu'un fils  $q$  de  $p$  reçoit le message  $\langle Diff \rangle$ , il sauvegarde la couleur que lui a affecté en émission son parent dans sa variable  $ce_q$ . Enfin, le fils  $q$  de  $p$  diffuse la couleur contenue dans sa variable  $ce_q$  à tous ses voisins à l'aide d'un message  $\langle DiffAck \rangle$ . Il est à noter que les messages  $\langle Diff \rangle$  et  $\langle DiffAck \rangle$  sont échangés pour assurer la bonne mise à jour des ensembles de couleurs autorisées en mode émission et réception dans le voisinage de chaque nœud. Ce mécanisme est nécessaire afin d'obtenir une coloration d'arêtes à distance-2 dans l'arbre  $T$ .

#### 4.3.3.2 Exemple d'exécution

Nous décrivons dans cette partie un exemple d'exécution de l'algorithme DD2EC illustré à la Figure 4.11. Nous considérons le graphe  $G = (V, E)$  et l'arbre  $T = (V, E_T)$  associé représentés sur la gauche de la Figure 4.11, dont le degré maximum  $\Delta$  d'un sommet dans le graphe  $G$  est de 6. La palette de couleurs  $\mathcal{C}$  à disposition de chaque nœud est composé de couleurs allant de 1 à  $(\Delta + 2)\Delta + 1$ . Le nombre de maximum de couleurs nécessaires à l'algorithme DD2EC étant donné tout graphe  $G = (V, E)$  et un arbre couvrant associé  $T = (V, E_T)$  est analysé ci-après dans le Théorème 2.

Comme présenté précédemment, la coloration est effectuée en partant du puits représenté par le sommet  $A$  dans l'exemple de la Figure 4.11 jusqu'aux feuilles de  $T$ . Nous déroulons ci-dessous les diverses étapes suivies par l'algorithme DD2EC pour colorer toutes les arêtes de l'arbre  $T$  de la Figure 4.11 :

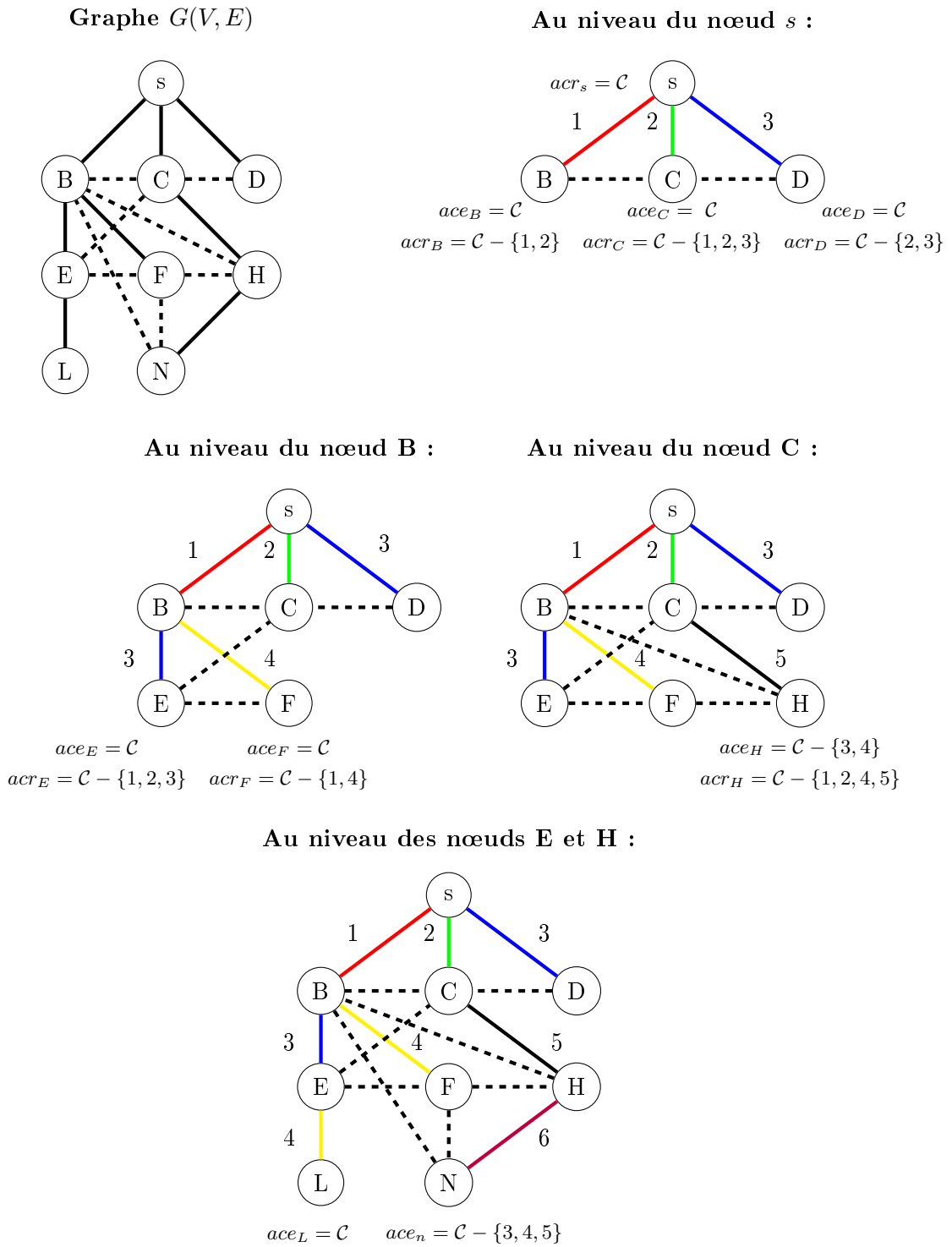


FIGURE 4.11 – Exemple d'exécution de l'algorithme DD2EC.

- **Au niveau du nœud  $A$**  : pour les arêtes entre le puits  $A$  et ses fils  $B, C, D$  toutes les couleurs de la palette  $\mathcal{C}$  sont autorisées en réception et en émission car aucune autre arête n'est plus prioritaire. Ainsi à l'instant  $t = 0$ , les fils  $q$  du puits n'ont pas de voisins en conflits et plus prioritaires (donc  $PrioE(q) = \emptyset$ ). Chaque fils  $q$  du puits envoie donc un message  $\langle PropCE, q, \mathcal{C} \rangle$ . Lorsque  $A$  reçoit les messages  $\langle PropCE \rangle$  de tous ses fils  $q$  avec leurs listes  $ace_q$  de couleurs autorisées en émission, il ordonne ses fils par ordre croissant de la taille des ensembles  $ace_q$  et attribue la première couleur non utilisée appartenant à la fois dans son ensemble de couleurs autorisées en réception et dans l'ensemble  $ace_q$  pour chaque fils  $q$ . Une fois les allocations de couleurs faites pour ses fils,  $A$  envoie un message  $\langle Diff \rangle$  à tous ses voisins avec l'ensemble de paires  $(q, cr_q)$ , où  $cr_q$  correspond à la couleur affectée par  $A$  à l'arête vers son fils  $q$ .

Lorsque les nœuds  $B, C$ , et  $D$  reçoivent ce message  $\langle Diff \rangle$  de  $A$ , chacun d'eux récupère la couleur qui leur a été attribuée en émission par  $A$  et la stocke dans leur variable  $ce$  puis retire cette couleur de sa liste de couleurs autorisées en réception  $acr$ . Enfin un message  $\langle DiffAck \rangle$  est envoyé aux voisins dans le graphe. La réception du message  $\langle DiffAck \rangle$  par les fils de  $B, C$  et  $D$  leur permettra de mettre à jour leur liste de couleurs autorisées en réception. La Figure 4.11 montre l'affectation des couleurs entre  $A$  et ses fils.

- **Au niveau du nœud  $B$**  : une couleur en émission lui a été attribuée par son parent ( $ce_B \neq \emptyset$ ) et on a  $PrioR(B) = \emptyset$ , donc  $B$  peut commencer à attribuer une couleur à chacun de ses fils une fois qu'il aura reçu un message  $\langle PropCE \rangle$  de chacun d'eux. D'après l'ordre décrit précédemment,  $E$  n'a pas de voisin plus prioritaire en émission, bien qu'il ait les voisins en conflits  $C$  et  $F$ . En effet,  $C$  est moins prioritaire car le parent de  $E$  possède un identifiant inférieur à  $C$ , et  $F$  possède le même parent que  $E$ . Donc, nous avons  $PrioE(E) = \emptyset$ . De même pour le fils  $F$ ,  $PrioE(F) = \emptyset$  car  $F$  possède une distance inférieure à  $N$  dans l'arbre, et  $H$  est au même niveau que  $F$  avec un parent d'identifiant inférieur à  $B$ . Donc, les nœuds  $E$  et  $F$  envoient leur message  $\langle PropCE \rangle$  à  $B$  qui leur attribuera une couleur différente en exécutant la procédure  $ChildrenColors()$ . Ensuite,  $B$  envoie un message  $\langle Diff \rangle$  avec les

couleurs qu'il a attribué en émission à ses fils  $E$  et  $F$ . Enfin, ceux-ci mettront à jour leur variable  $ce$  et enverront un message  $\langle DiffAck \rangle$  dans leur voisinage avec leur couleur en émission.

- **Au niveau du nœud  $C$**  : comme décrit précédemment pour le fils  $E$  de  $B$ , on a  $PrioR(C) = E$  donc  $C$  ne pourra pas colorer ses fils tant que  $E$  n'aura pas envoyé sa couleur d'émission via un message  $\langle DiffAck \rangle$ . De plus, il doit attendre la réception du message  $\langle PropCE \rangle$  de son fils  $H$ .
- **Au niveau du nœud  $H$**  : on a  $PrioE(H) = B$ , donc  $H$  est moins prioritaire que  $B$  car à une distance supérieure à  $B$  et l'identifiant du père de  $H$  est supérieur à  $B$ . A l'inverse,  $H$  est plus prioritaire que  $D$ . Donc  $H$  pourra envoyer son message  $\langle PropCE \rangle$  une fois qu'il aura reçu le message  $\langle Diff \rangle$  de la part de  $B$ . Ainsi, son parent  $C$  lui attribuera une couleur dès que celui-ci aura reçu un message  $\langle DiffAck \rangle$  de la part de  $E$ .
- **Au niveau du nœud  $L$**  : on a  $PrioE(L) = \emptyset$ , donc  $L$  peut envoyer son  $\langle PropCE \rangle$  à son parent  $E$  dès le début de l'exécution de l'algorithme, car il n'a aucun voisins en conflits plus prioritaire. Ainsi,  $E$  pourra lui attribuer une couleur dès que son père lui aura également attribué une couleur en émission.
- **Au niveau du nœud  $N$**  : on a  $PrioE(N) = B, F$ ,  $N$  compte  $B$  et  $F$  comme voisins en conflit plus prioritaires car tous deux ont une distance inférieure à  $N$  et  $F$  est d'identifiant inférieur au père de  $N$ . Cependant, il faut noter que  $F$  n'est pas prioritaire car c'est une feuille de l'arbre et ne passera donc jamais en mode réception.  $N$  attendra uniquement que  $B$  termine sa phase de coloration afin de mettre à jour son ensemble de couleurs autorisées avec la réception du message  $\langle Diff \rangle$ . Enfin, il enverra son message  $\langle PropCE \rangle$  à son père qui pourra lui attribuer une couleur en émission une fois que lui-même aura obtenu sa couleur en émission.

#### 4.3.3.3 Analyse de l'algorithme DD2EC

Dans cette section, nous allons analyser l'algorithme DD2EC présenté précédemment pour établir le nombre maximum de couleurs nécessaires, ainsi que la complexité en temps



et en messages pour construire une solution au problème d'ordonnement sans collision (Problème 1).

**Lemme 2**

Étant donné un graphe  $G = (V, E)$  et un arbre couvrant  $T = (V, E_T)$  de  $G$ , l'algorithme DD2EC construit une coloration orientée d'arêtes de  $T$  pour résoudre le Problème 1.

**Preuve.** L'algorithme DD2EC affecte une couleur à chaque arête  $e \in E_T$  en partant de la racine jusqu'aux feuilles en suivant l'ordre défini précédemment basé sur les distances dans  $T$  de chaque nœud au puits  $s$ .

Chaque nœud  $p$  maintient deux variables  $ace_p$  et  $acr_p$  avec la liste des couleurs utilisables par  $p$  en émission et en réception respectivement. D'après l'Algorithme 1, ces deux variables sont initialisées avec l'ensemble de la palette de couleurs  $\mathcal{C}$ . Ensuite, chaque nœud  $p$  modifie sa variable  $ace_p$  lors de la réception d'un message  $\langle Diff, q, ecr \rangle$  de la part d'un voisin  $q$  en conflit plus prioritaire lorsque  $p$  est en émission (c'est-à-dire  $q \in PrioE(p)$ ). Dans ce cas, pour chaque paire  $(x, c) \in ecr$  le nœud  $p$  retire la couleur  $c$  de  $ace_p$  car cette couleur est affectée à  $q$  en réception. D'autre part, chaque nœud  $p$  modifie sa variable  $acr_p$  lors de la réception d'un message  $\langle Diff, q, ecr \rangle$  de son père  $q$  dans  $T$ . Dans ce cas, pour chaque paire  $(x, c) \in ecr$  le nœud  $p$  retire la couleur  $c$  de  $acr_p$  soit si  $c$  est la couleur en émission affectée par son père  $q$  (c'est-à-dire  $x = p$ ), soit si  $p$  est en conflit avec son frère  $x$  (c'est-à-dire  $x \in NI(p)$ ) car l'arc  $(x, q)$  dans  $T$  est plus prioritaire que les arcs  $(y, p)$  avec les fils  $y$  de  $p$  dans  $T$ . Enfin, chaque nœud  $p$  modifie également sa variable  $acr_p$  lors de la réception d'un message  $\langle DiffAck, q, ce \rangle$  d'un voisin  $q$ .  $p$  retire la couleur  $ce$  de  $acr_p$  car la couleur  $ce$  correspond à la couleur affectée en émission soit au père  $q$  de  $p$  dans  $T$ , soit à un voisin  $q$  en conflit et plus prioritaire (c'est-à-dire  $q \in PrioR(p)$ ).

On peut donc déduire les propriétés suivantes de la mise à jour des variables  $ace_p$  et  $acr_p$  pour chaque nœud  $p$  :

1. chaque nœud  $p$  ne peut utiliser une couleur en réception depuis l'un de ses fils dans  $T$  qui est égale à la couleur affectée à  $p$  en émission vers son père dans  $T$ ,
2. chaque nœud  $p$  ne peut utiliser une couleur en réception égale à la couleur affectée au père de  $p$  dans  $T$ ,

3. chaque nœud  $p$  ne peut utiliser une couleur en émission affectée en réception aux voisins  $q$  en conflit dans  $G$  et plus prioritaires,
4. chaque nœud  $p$  ne peut utiliser une couleur en réception affectée en émission aux voisins  $q$  en conflit dans  $G$  et plus prioritaires.

Ensuite, pour chaque arc  $(q, p)$  dans  $T$  le parent  $p$  attend la liste des couleurs autorisées en émission  $ace_q$  de ses fils  $q$ . Cette liste est envoyée dans un message  $\langle PropCE \rangle$  par chaque fils  $q$  dès que celui-ci a reçu un message  $\langle Diff \rangle$  de la part de tous ses voisins en conflit et plus prioritaires ou s'il n'a pas de tel voisin. Lorsque  $p$  dispose des ensembles de couleurs autorisées en émission  $ace_q$  de tous ses fils  $q$  et sa liste complète de couleurs autorisées en réception  $ace_p$ ,  $p$  peut affecter des couleurs différentes aux arcs  $(q, p)$ . D'après l'Algorithme 1, pour chaque nœud  $p \in V$  la procédure Children\_Colors() affecte une couleur différente pour chaque arête avec un fils  $q$  de  $p$  dans  $T$ . Cette affectation tient compte des couleurs autorisées en émission  $ace_q$  pour  $q$  et en réception  $acr_p$  pour  $p$ . Une fois l'affectation établie, un message  $\langle Diff \rangle$  est envoyé contenant la liste de ces affectations. Il est à noter que cette fonction est appelée à trois endroits dans l'Algorithme 1, (i) soit  $p$  connaît déjà sa couleur en émission ainsi que celles de ses voisins en conflits et plus prioritaires lorsqu'il reçoit le dernier message  $\langle PropCE \rangle$  de ses fils  $q$  (bloc réception message  $\langle PropCE \rangle$ ), (ii) soit  $p$  a reçu tous les messages  $\langle PropCE \rangle$  de ses fils ainsi que les couleurs en émission de ses voisins en conflits et plus prioritaires avant que sa couleur en émission lui soit affectée (bloc réception message  $\langle Diff \rangle$ ), (iii) soit  $p$  connaît déjà sa couleur en émission et a reçu tous les messages  $\langle PropCE \rangle$  de ses fils avant de recevoir toutes les couleurs en émission de ses voisins en conflits et plus prioritaires (bloc réception messages  $\langle DiffAck \rangle$ ). □

**Lemme 3**

Étant donné un graphe  $G = (V, E)$  et un arbre couvrant  $T = (V, E_T)$  de  $G$ , l'algorithme DD2EC construit une coloration orientée d'arêtes de  $T$  utilisant  $\min(n - 1, (\Delta + 2)\Delta + 1)$  couleurs pour résoudre le Problème 1, avec  $n$  le nombre de sommets et  $\Delta$  le degré maximum du graphe  $G$ .

**Preuve.** Tout d'abord, nous dénotons par  $\Delta_T$  le degré maximum d'un sommet dans

l'arbre  $T$ . On peut observer que chaque arc  $e = (q, p) \in E_T$  possède au plus  $2(\Delta_T - 1)$  arcs incidents dans  $T$  avec lesquelles  $e$  doit avoir une couleur différente. De plus, lorsque  $p$  est en réception il peut avoir au plus  $\Delta - \Delta_T$  voisins en conflits et plus prioritaires. De même, lorsque  $q$  est en émission il peut avoir au plus  $\Delta - \Delta_T$  voisins en conflits et plus prioritaires. Ces  $\Delta - \Delta_T$  voisins en conflits sont en réception d'au plus  $\Delta_T - 1$  fils chacun. Ainsi,  $q$  doit avoir une couleur en émission différente des  $(\Delta - \Delta_T) \times (\Delta_T - 1)$  arcs de  $T$ , dont les nœuds en réception sont en conflits avec  $q$ .

Par conséquent, un arc  $(q, p) \in E_T$  peut être en conflit avec au plus :

$$2(\Delta_T - 1) + (\Delta - \Delta_T) + (\Delta - \Delta_T) \times (\Delta_T - 1) \leq (\Delta + 2)\Delta \text{ autres arcs de } T.$$

Donc,  $(\Delta + 2)\Delta + 1$  couleurs sont suffisantes pour une coloration orientée d'arêtes de  $T$ . D'autre part, tout arbre couvrant  $T$  est composé de  $n - 1$  arêtes que l'on cherche à colorier, dans tous les cas le nombre de couleurs ne peut dépasser le nombre d'arêtes de  $T$ . En conclusion, la coloration retournée par l'algorithme DD2EC nécessite au plus  $\min(n - 1, (\Delta + 2)\Delta + 1)$  couleurs.  $\square$

### **Théorème 2**

Étant donné un graphe  $G = (V, E)$  et un arbre couvrant  $T = (V, E_T)$  de  $G$ , l'algorithme DD2EC construit une coloration orientée d'arêtes de  $T$  utilisant  $\min(n - 1, (\Delta + 2)\Delta + 1)$  couleurs pour résoudre le Problème 1 en temps  $O(\Delta^2 n)$  et en échangeant  $O(m)$  messages de taille  $O(\Delta \log(n))$  bits, avec  $n$  et  $m$  le nombre de sommets et d'arêtes de  $G$ , et  $\Delta$  le degré maximum du graphe  $G$ .

**Preuve.** D'après les Lemmes 2 et 3, l'algorithme DD2EC construit une coloration orientée d'arêtes de  $T$  avec  $\min(n - 1, (\Delta + 2)\Delta + 1)$  couleurs.

Pour la complexité en messages, d'après l'Algorithme 1 l'algorithme DD2EC utilise 3 types de messages :

- un message  $\langle PropCE \rangle$  est envoyé par chaque fils  $q$  à son père  $p$  dans  $T$  pour lui communiquer sa liste  $ace_q$  de couleurs autorisées en émission,
- chaque nœud  $p$  envoie un message  $\langle Diff \rangle$  à tous ses voisins (en particulier ses fils  $q$ ) pour les informer de la couleur affectée à chaque arc  $(q, p)$ ,
- enfin un message  $\langle DiffAck \rangle$  est envoyé par chaque fils  $q$  à tous ses voisins pour

relayer l'information au cas où le voisinage de  $p$  et  $q$  n'est pas identique.

Ainsi, chaque arête de  $G$  est traversée par un nombre constant de messages. De plus, chaque type de message transporte au plus  $(\Delta - 1)$  identifiants et/ou couleurs chacun codé sur  $\log(n)$  bits. Donc,  $O(m)$  messages de  $O(\Delta \log(n))$  bits sont échangés par l'algorithme DD2EC.

Pour la complexité en temps, nous considérons que chaque message est transmis au bout d'une unité de temps. De plus, si des messages sont envoyés simultanément ils sont transmis au bout d'une seule unité de temps. La coloration des arcs  $(q, p)$  ayant une même extrémité  $p$  est réalisée en trois phases :

1. La première phase permet à  $p$  et  $q$  de construire leur liste de couleurs autorisées  $acr_p$  en réception et  $ace_q$  en émission. Pour cela,  $p$  attend la réception d'au plus  $\Delta - \Delta_T \leq \Delta$  messages de type  $\langle Diff \rangle$  ou  $\langle DiffAck \rangle$ , et  $q$  attend la réception d'au plus  $\Delta - \Delta_T \leq \Delta$  messages de type  $\langle Diff \rangle$ . La réception de l'ensemble de ces messages pour tous les fils de  $p$  est effectuée au plus au bout de  $\Delta^2 + \Delta$  unités de temps.
2. La seconde phase permet à  $p$  de collecter la liste  $ace_q$  pour chacun de ses fils  $q$ . Pour cela,  $p$  attend la réception d'au plus  $\Delta_T \leq \Delta$  messages de type  $\langle PropCE \rangle$ . La réception de l'ensemble de ces messages est effectuée au plus au bout de  $\Delta$  unités de temps.
3. La dernière phase permet à  $p$  de communiquer la couleur affectée à chacun des arcs le reliant à ses fils. Pour cela,  $p$  envoie au plus  $\Delta$  messages de type  $\langle Diff \rangle$  nécessitant au plus 1 unités de temps.

Donc, l'ensemble de ces arcs sont colorés au bout de  $\Delta^2 + 2\Delta + 1 \leq O(\Delta^2)$  unités de temps.

De plus, dans tout arbre  $T$  il y a au maximum  $n - 1$  groupes d'arcs ne pouvant être colorés simultanément. Comme chaque groupe d'arcs nécessite au plus  $O(\Delta^2)$  unités de temps pour être coloré, cela implique qu'il faut au plus  $O(\Delta^2 n)$  unités de temps pour construire une coloration orientée d'arêtes pour tout arbre couvrant  $T$ .  $\square$

## 4.4 Algorithme auto-stabilisant SS-DD2EC

L'algorithme distribué DD2EC décrit dans la section précédente ne permet pas de tolérer les fautes transitoires pouvant survenir au niveau des mémoires ou informations échangées. Ces défaillances ont une forte probabilité d'apparaître plus la taille du réseau est importante. Afin de tolérer ces défaillances, nous décrivons dans cette section une extension de l'algorithme DD2EC en utilisant le paradigme de l'auto-stabilisation. Comme présenté ci-après, ce paradigme garantit au système de retrouver un comportement correct en temps fini quel que soit l'état initial des nœuds et l'état initial des liens de communication.

### 4.4.1 Modélisation d'un système auto-stabilisant

Le paradigme d'*auto-stabilisation* est une approche générique introduite par Dijkstra [Dijkstra 1974] pour tolérer les fautes transitoires dans un système distribué. Une faute transitoire est une faute survenant de façon inopinée dans le système et qui altère le comportement du système (processus ou canaux de communication). Un algorithme distribué est dit *auto-stabilisant* si à partir d'une configuration arbitraire (éventuellement générée par des fautes ou attaques) le système est capable d'atteindre à nouveau une configuration légitime (configuration vérifiant certaines propriétés désirées) en un temps fini sans intervention extérieure (par exemple humaine). Comme ce paradigme ne fait aucune hypothèse sur la nature ni l'étendue des fautes, celui-ci peut être utilisé pour traiter les changements dynamiques survenant sur la topologie du réseau.

**Définition 1 (Auto-stabilisation)** *Étant donné un prédicat (non vide)  $\mathcal{L}_A$ , un algorithme  $A$  est dit auto-stabilisant s'il satisfait les conditions suivantes : (i) Toute exécution de  $A$  partant d'une configuration vérifiant  $\mathcal{L}_A$  préserve  $\mathcal{L}_A$  (clôture). (ii) Toute exécution de  $A$  partant d'une configuration arbitraire contient une configuration qui satisfait  $\mathcal{L}_A$  (convergence).*

Nous utiliserons le *modèle à états* dans lequel les nœuds communiquent avec leurs voisins dans le réseau à travers des registres partagés en lecture. Les algorithmes auto-stabilisants sont souvent décrits sous la forme d'un ensemble d'actions. Chaque action est de la forme suivante :  $(nom) : (garde) (instruction)$ . Le nom d'une action est un identifiant

utilisé uniquement dans les raisonnements. La garde d'une action est un prédicat sur les variables du nœud et de ses voisins. L'instruction d'une action du nœud met à jour une ou plusieurs variables du nœud. Une action ne peut être exécutée que si elle est *activable*, c'est-à-dire si sa garde est vraie. Un nœud est dit *activable* si au moins une de ses actions est activable. L'évaluation des gardes et l'exécution éventuelle des actions sont supposées atomiques.

Nous rappelons qu'une exécution distribuée est décomposée en pas de calcul, composées de trois phases exécutées de façon atomique : (i) un nœud évalue ses actions, (ii) un ordonnanceur (ou *démon*) choisit les nœuds à activer, (iii) chaque nœud choisit et exécute l'action la plus prioritaire de son algorithme local. Lorsque ces trois phases sont terminées un nouveau pas de calcul commence. Un démon (ou *ordonnanceur*) est dit *distribué* s'il choisit un sous-ensemble des nœuds du système à chaque pas de calcul, dans le cas contraire il est dit *centralisé*. D'autre part, le démon peut activer l'exécution de l'algorithme sur les nœuds de façon plus ou moins équitable. Dans la suite, nous considérerons la notion d'*équité faible* définie comme suit : si un nœud  $p$  est activable en continu alors  $p$  sera activé à terme par le démon pour exécuter une action.

#### 4.4.2 État de l'art

Les problèmes de coloration d'un graphe sont fondamentaux dans les systèmes distribués et sont beaucoup utilisés pour l'allocation de ressources. Ainsi, un certain nombre d'algorithmes auto-stabilisants ont été proposés pour ces problèmes. Gradinariu et Tixeuil [Gradinariu and Tixeuil 2000] ont proposé le premier algorithme auto-stabilisant distribué et déterministe pour colorer avec un nombre minimum les sommets d'un graphe arbitraire. Cet algorithme utilise au plus  $\Delta + 1$  couleurs et converge vers une configuration légitime en temps  $O(\Delta n)$  pas de calcul en considérant un ordonnanceur distribué et des identifiants uniques sur les sommets, avec  $\Delta$  le degré maximum d'un sommet et  $n$  le nombre de sommets du graphe. [Masuzawa and Tixeuil 2007] ont considéré le problème de coloration des arêtes d'un graphe arbitraire et anonyme (sans identifiants sur les sommets ou numérotation des identiques du voisinage des sommets). Ils ont proposé un algorithme distribué tolérant les fautes transitoires ainsi que les fautes byzantines. Cet algorithme colore les arêtes du

graphe avec au plus  $2\Delta - 1$  couleurs et se stabilise au plus en  $2\Delta + 2$  rondes. [Drira et al. 2009] ont proposé un algorithme auto-stabilisant centralisé pour la coloration d'arêtes d'un graphe arbitraire nécessitant au plus  $\Delta + 1$  couleurs. [Tzeng et al. 2007] ont étudié le problème respectivement pour les graphes planaires et les graphes bipartis et ont proposé des algorithmes auto-stabilisants centralisés ayant besoin respectivement d'au plus  $\Delta + 4$  et  $\Delta$  couleurs. [Bernard et al. 2010] se sont intéressés à l'impact d'une approche probabiliste sur le temps de convergence dans les graphes anonymes et ont proposé un algorithme auto-stabilisant probabiliste pour la coloration de sommets. Cet algorithme converge vers une configuration légitime en  $O(\Delta n)$  pas de calcul.

Concernant la coloration à distance 2 dans un graphe, [Herman and Tixeuil 2004] ont considéré le contexte des réseaux de capteurs utilisant une approche TDMA pour l'accès au médium. Les auteurs ont proposé un algorithme auto-stabilisant distribué et probabiliste pour attribuer une couleur différente à chaque sommet par rapport à son voisinage à distance 2 en temps  $O(1)$ . Cet algorithme s'appuie sur la construction d'un nommage unique des sommets probabiliste et d'un ensemble indépendant maximal probabiliste. [Blair and Manne 2012] ont proposé un algorithme distribué auto-stabilisant nécessitant au plus  $\Delta^2 + 1$  couleurs et convergeant en au plus  $O(\Delta^2 m)$  pas de calcul. Cet algorithme n'utilise aucune information à distance deux dans le graphe, celui-ci est basé sur un mécanisme de permissions dans chaque voisinage. Un sommet ayant obtenu la permission de tous ses voisins peut proposer une couleur qui doit être ensuite acceptée par ceux-ci avant de se l'affecter. Récemment, [Lee and Liu 2014] ont proposé un algorithme distribué auto-stabilisant pour le problème de coloration d'arêtes à distance 2 dans un graphe quelconque. Au plus  $2\Delta(\Delta - 1) + 1$  couleurs sont utilisées et une configuration légitime est atteinte en au plus  $O(\Delta^2 m + n)$  rondes. Cet algorithme se fonde sur la proposition d'un mécanisme de synchronisation des arêtes à distance 2 dans le graphe pour empêcher leur coloration simultanée.

D'autres problèmes de coloration ont été étudiés, on peut citer par exemple le travail de [Chaudhuri and Thompson 2011] qui ont proposé un algorithme auto-stabilisant pour construire un  $L(2, 1)$ -étiquetage des sommets pour la classe des graphes arborescents. Un  $L(2, 1)$ -étiquetage des sommets est une fonction attribuant un entier (ou couleur) à chaque

sommet du graphe tel que (1) toutes paires de sommets voisins reçoivent un étiquetage dont les valeurs diffèrent d'au moins 2, et (2) toutes paires de sommets à distance 2 reçoivent une étiquette dont les valeurs diffèrent d'au moins 1. L'algorithme proposé fonctionne avec un ordonnanceur distribué et construit un  $L(2,1)$ -étiquetage en  $O(h)$  rondes, avec  $h$  la hauteur de l'arbre.

Les travaux cités ci-dessus pour le problème de coloration à distance 2 d'un graphe ne prennent pas en compte les deux types d'arêtes, à savoir les arêtes de l'arbre de collecte à colorer et les arêtes indiquant des collisions possibles. Dans la suite de ce chapitre, nous adaptons l'approche suivie dans l'algorithme DD2EC présenté précédemment pour concevoir un algorithme auto-stabilisant pour résoudre le Problème 1 que nous considérons.

#### 4.4.3 Présentation de l'algorithme SS-DD2EC

Dans cette section, nous allons présenter l'algorithme SS-DD2EC qui est une adaptation de l'algorithme DD2EC afin de pouvoir tolérer les fautes transitoires sur la mémoire des nœuds du réseau. L'algorithme SS-DD2EC décrit dans cette section utilise le modèle à états pour les communications entre nœuds voisins. Chaque nœud exécute le même algorithme (excepté le puits exécutant une version simplifiée). Une description formelle de l'algorithme est donnée par l'Algorithme 2 pour le puits  $s$  et par l'Algorithme 3 pour tous les autres nœuds du réseau.

##### 4.4.3.1 Description de l'algorithme

Nous considérons comme précédemment qu'une arborescence  $T$  enracinée au puits  $s$  est construite. L'algorithme auto-stabilisant SS-DD2EC décrit ici suit le même ordre de coloration des arêtes de  $T$  que l'algorithme distribué DD2EC présenté dans la section précédente. Comme pour l'Algorithme DD2EC, chaque nœud possède deux modes (émission et réception) et les fonctions  $PrioR(p)$  et  $PrioE(p)$  sont réutilisées pour définir l'ordre de coloration des arêtes de l'arbre  $T$ . Les arêtes sont colorées à partir du puits  $s$  de haut en bas en suivant les distances dans l'arbre.

Chaque nœud  $p \in V$  dispose des mêmes informations en entrée que dans l'algorithme DD2EC. De plus, l'état de chaque nœud  $p \in V$  est défini par les variables locales ci-



dessous :

- les variables  $ce_p$  et  $cr_p$  contiennent respectivement la couleur d'émission vers le père de  $p$  et de réception avec tous les fils de  $p$  dans l'arbre,
- l'ensemble des couleurs autorisées en mode émission et réception sont stockées respectivement dans les variables  $ace_p$  et  $acr_p$ ,
- enfin les variables temporaires  $color\_temp_p$  et  $cr\_temp_p$  sont utilisées pour le calcul des couleurs dans la fonction  $ChildColor()$ .

En plus de la procédure  $ChildColor()$  qui a la même fonctionnalité que  $Children\_Colors()$  ; la procédure  $ColorCE()$  permet de calculer la couleur en émission du nœud qui l'exécute, et la procédure  $ColorR(cr_q)$  sauvegarde dans une variable temporaire les couleurs du nœud en mode réception.

---

**Algorithm 2** Algorithme SS-DD2EC pour le nœud puits  $s$ 


---

**Entrées :**

- $C$  : la palette des couleurs possibles ;
- $Neigs$  : ensemble des voisins de  $s$  ordonné localement ;
- $Child_s$  : ensemble des nœuds fils de  $s$  dans l'arbre couvrant  $T$  ;

**Variables :**

- $cr_s$  : variable stockant la couleur  $col$  utilisée par  $s$  en réception associée à son fils  $q$ , ensemble de paires  $(q, col)$  ;

**Fonctions :**

$ChildColor()$

```

 $cr\_temp_s := \emptyset$ ;
 $color\_temp_s := \emptyset$ ;
for all  $q \in Child_s$  ordonnés par ordre croissant de  $|ace_q|$  do
   $cr\_temp_s := cr\_temp_s \cup \{(q, \min(acr_s \cap ace_q) \setminus color\_temp_s)\}$ ;
   $color\_temp_s := color\_temp_s \cup \{\min((acr_s \cap ace_q) \setminus color\_temp_s)\}$ ;
end for
retourner  $cr\_temp_s$ 

```

**Règles :**

R0 ::  $cr_s \neq ChildColor() \rightarrow cr_s := ChildColor()$ ;

---

L'algorithme SS-DD2EC s'appuie sur quatre règles qui sont exécutées de façon asynchrone par tous les nœuds du système. La convergence de l'état du système est réalisée de niveau en niveau dans l'arbre à partir de la racine. Ainsi lorsqu'un niveau de l'arbre est coloré, l'état des nœuds correspondant ne change plus car plus aucune action de l'algorithme n'est activable (sauf si une faute survient sur l'état d'un nœud). Pour simplifier l'explication du fonctionnement de l'algorithme, nous décrivons l'exécution des actions en considérant la coloration effectuée à un niveau quelconque dans l'arbre par un nœud  $p$  avec ses fils  $q$ .

---

**Algorithm 3** Algorithme SS-DD2EC pour tout noeud  $p \neq s$

---

**Entrées :**

- $\mathcal{C}$  : palette des couleurs possibles ;
- $Neig_p$  : ensemble des voisins de  $p$  ordonné localement ;
- $par_p$  : parent de  $p$  dans l'arbre couvrant  $T$  ;
- $Child_p$  : ensemble des fils de  $p$  dans l'arbre couvrant  $T$  ;
- $dist_p$  : distance dans  $T$  entre  $p$  et le puits  $s$  ;
- $par_q$  : parent dans l'arbre couvrant  $T$  pour chaque voisin  $q$  de  $p$  ;
- $Child_q$  : ensemble des fils dans l'arbre couvrant  $T$  pour chaque voisin  $q$  de  $p$  ;
- $dist_q$  : distance dans  $T$  entre un voisin  $q$  de  $p$  et le puits  $s$  ;

**Variables :**

- $ce_p$  : variable stockant la couleur de  $p$  en émission ;
- $cr_p$  : variable stockant la couleur  $col$  de  $p$  en réception pour chaque fils  $q$ , ensemble de paires  $(q, col)$  ;
- $ace_p$  : variable stockant l'ensemble des couleurs autorisées en émission pour  $p$  ;
- $acr_p$  : variable stockant l'ensemble des couleurs autorisées en réception pour  $p$  ;
- $color\_temp_p$  : variable temporaire interne à la fonction ChildColor() stockant l'ensemble des couleurs en réception pour  $p$  ;
- $cr\_temp_p$  : variable temporaire interne à la fonction ChildColor() stockant la couleur  $col$  de  $p$  en émission pour chacun de ses fils  $q$ , ensemble de paires  $(q, col)$  ;

**Macros :**

- $NI(p) = Neig_p - Child_p - \{par_p\}$
- $NCh(p) = \{q \in NI(p) : |Child_q| = 0\}$
- $P(p) = NI(p) - NCh(p)$
- $PrioR(p) = \{q \in NI(p) : (dist_q < dist_p) \vee (dist_q = dist_p \wedge par_q < par_p) \vee (dist_q = dist_p + 1 \wedge par_q < p)\}$
- $PrioE(p) = \{q \in P(p) : (dist_q < dist_p - 1) \vee (dist_q = dist_p - 1 \wedge q < par_p)\}$

**Fonctions :**

ChildColor()

```

 $cr\_temp_p := \emptyset;$ 
 $color\_temp_p := \emptyset;$ 
for all  $q \in Child_p$  ordonnés par ordre croissant de  $|ace_q|$  do
   $cr\_temp_p := cr\_temp_p \cup \{(q, \min(acr_p \cap ace_q) \setminus color\_temp_p)\};$ 
   $color\_temp_p := color\_temp_p \cup \{\min((acr_p \cap ace_q) \setminus color\_temp_p)\};$ 
end for
retourner  $cr\_temp_p$ 

```

ColorCE()

```

for all  $e = (e_1, e_2) \in cr_{par_p}$  do
  if  $(e_1 == p)$  then
    retourner  $e_2$ 
  end if
end for

```

Color( $cr_q$ )

```

 $color\_temp_p := \emptyset;$ 
for all  $e = (e_1, e_2) \in cr_q$  do
   $color\_temp_p := color\_temp_p \cup \{e_2\};$ 
end for
retourner  $color\_temp_p$ 

```

**Prédicats :**

- $Good\_ACE(p) \equiv ace_p = \mathcal{C} \setminus (\bigcup_{q \in PrioE(p)} Color(cr_q))$
- $Good\_CE(p) \equiv ce_p = ColorCE()$
- $Good\_ACR(p) \equiv acr_p = \mathcal{C} \setminus (ce_{q \in (PrioR(p) \cup \{p, par_p\})})$
- $Good\_CR(p) \equiv cr_p = ChildColor()$

**Règles :**

- |    |    |   |   |   |
|----|----|---|---|---|
| R0 | :: | $\neg Good\_ACE(p)$   | → | $ace_p := \mathcal{C} \setminus (\bigcup_{q \in PrioE(p)} Color(cr_q));$    |
| R1 | :: | $Good\_ACE(p) \wedge \neg Good\_CE(p)$  | → | $ce_p := ColorCE();$  |
| R2 | :: | $Good\_ACE(p) \wedge Good\_CE(p) \wedge \neg Good\_ACR(p)$                    | → | $acr_p := \mathcal{C} \setminus (ce_{q \in (PrioR(p) \cup \{p, par_p\})});$ |
| R3 | :: | $Good\_ACE(p) \wedge Good\_CE(p) \wedge Good\_ACR(p) \wedge \neg Good\_CR(p)$ | → | $cr_p := ChildColor();$   |
-

Tout nœud fils  $q$  met à jour si nécessaire (suivant la valeur du prédicat  $\neg Good\_ACE(q)$ ) son ensemble de couleurs autorisées en émission stocké dans sa variable  $ace_q$  à l'aide de la règle R0, en tenant compte des couleurs utilisées par ses voisins en conflit de plus forte priorité. De son côté, le nœud père  $p$  met également à jour si nécessaire (suivant la valeur du prédicat  $\neg Good\_ACR(p)$ ) son ensemble de couleurs autorisées en réception en exécutant la règle R2. Dans ce cas,  $p$  modifie l'état de sa variable  $acr_p$  en tenant également compte de l'état de ses voisins en conflit et plus prioritaires. Au niveau du parent  $p$ , si une erreur est détectée sur sa variable  $cr_p$  concernant les couleurs attribuées aux liens avec ses fils (suivant la valeur du prédicat  $\neg Good\_CR(p)$ ), alors  $p$  exécute la règle R3 pour mettre à jour sa variable  $cr_p$  contenant l'ensemble des couleurs attribuées à ses fils. Enfin, lorsqu'un fils  $q$  de  $p$  détecte une incohérence sur la couleur d'émission stockée dans sa variable  $ce_q$  avec celle indiquée par son père  $p$  dans sa variable  $cr_p$ , celui-ci exécute la règle R1 pour corriger sa couleur d'émission stockée dans sa variable  $ce_q$ .

L'étude de complexité et de correction de l'algorithme SS-DD2EC présentée en annexe 7.2 a pour conséquence le théorème suivant.

**Théorème 1**

*L'algorithme SS-DD2EC est auto-stabilisant pour la Spécification 1 sous un ordonnanceur faiblement équitable.*

## 4.5 Conclusion

Dans ce chapitre, nous avons présenté les principales caractéristiques du médium radio à prendre en compte pour limiter les collisions lors des transmissions pour la collecte des données dans un réseau de capteurs sans fil. Nous nous sommes orientés vers une approche TDMA pour le partage du médium sans fil et nous avons présenté une modélisation de l'accès au médium. Pour cela, nous avons défini un problème spécifique de coloration des arêtes, coloration orientées des arêtes, d'un graphe dont l'objectif est d'activer le plus grand nombre de liens de communication simultanément sans générer de collision. Nous avons également étudié la complexité de ce problème, qui est NP-complet à résoudre dans le cas général excepté certains cas particuliers où le problème devient polynomial. Nous

## 4.5. CONCLUSION

---

avons ensuite proposé deux algorithmes distribués pour ce problème, le premier algorithme, nommé DD2EC, fonctionne dans un cas parfait où il n'y a pas de fautes dans le réseau et le second, nommé SS-DD2EC, est un algorithme auto-stabilisant permettant de tolérer les fautes transitoires survenant sur la mémoire des nœuds.

Il y a plusieurs perspectives aux travaux proposés dans ce chapitre. Nous avons considéré le cas d'un réseau de capteurs dans lequel tous les nœuds utilisent la même fréquence pour communiquer. Comme premier axe, il serait intéressant d'aller plus loin en considérant l'utilisation de plusieurs fréquences et également des informations plus fines sur la position ou la densité des nœuds. Des travaux se sont déjà intéressés à cette variante, comme le travail de Ghosh *et al.* [Ghosh et al. 2011]. Dans un premier temps, l'objectif serait d'intégrer ces éléments et de modifier le problème de coloration considéré dans ce chapitre, et ensuite de proposer des approches distribuées. Comme second axe, il serait également utile de considérer des problèmes d'étiquetage pour tenir compte d'autres contraintes sur l'affectation des fréquences ou des intervalles de temps (slots) dans le voisinage à distance 2 et 3.

## Chapitre 5

# Évaluation de métriques de routage pour RPL

### 5.1 Introduction

L'objectif des protocoles de communication pour les réseaux de capteurs est la maîtrise de la consommation énergétique, et la prolongation de la durée de vie du réseau. Pour cela dans le chapitre précédent, nous avons proposé une nouvelle technique de communication au niveau de la couche MAC basée sur l'approche TDMA pour minimiser l'énergie consommée lors des échanges entre les nœuds capteurs. Dans ce chapitre nous nous intéressons à l'étude de l'un des premiers protocoles de routage IPv6 normalisé par le groupe de travail IETF (Internet Engineering Task Force) dédié aux réseaux à faible consommation énergétique nommé RPL (Routing Protocol for Low-power and lossy networks).

RPL est aujourd'hui le protocole de routage au niveau de la couche réseau le plus utilisé dans les applications utilisant la couche d'adaptation 6LoWPAN (IPv6 Low power Wireless Personal Area Networks). Il construit une topologie de routage logique de type arbre orienté vers le nœud racine de destination appelée DODAG (Destination Oriented Directed Acyclic Graph) en utilisant une fonction objectif. La fonction objectif se base sur une ou plusieurs contraintes appelées métriques pour définir les meilleurs chemins de routage vers le nœud racine. En revanche, la formation du DODAG et les performances du réseau construit en terme de durée de vie, de délai de transmission ou du taux de transmission des paquets dépendent du choix de la métrique utilisée.

Par conséquent, dans ce qui suit, nous étudions le fonctionnement de RPL et les performances de plusieurs métriques de routage RPL proposées pour économiser l'énergie consommée et maximiser la durée de vie du réseau.

Ce chapitre est composé de trois sections, nous commençons par introduire le protocole de routage RPL. Ensuite nous décrivons quelques métriques de routage appliquées au protocole RPL. Enfin nous présentons une évaluation de ces métriques.

### 5.2 Présentation du protocole de routage RPL

Le protocole de routage RPL (Routing Protocol for Low-power and lossy networks) est conçu pour les réseaux de capteurs caractérisés par de faibles capacités (batteries, calcul, mémoire ...) et un taux de perte de paquets important lors des transmissions [Winter et al. 2012] [Gaddour and Koubaa 2012]. Ce protocole a été proposé pour la première fois en 2012 par le groupe de travail IETF Routing Over Low Power and Lossy (ROLL). RPL est un protocole de routage à vecteur de distance à destination des réseaux IPv6. Conformément à l'architecture IPv6, il construit un graphe orienté acyclique (DAG) afin d'établir des chemins de communication bidirectionnels entre les capteurs. RPL est conçu principalement pour envoyer des données depuis un ensemble de capteurs du réseau vers un nœud particulier, appelé nœud puits ou nœud racine. Un réseau de capteurs peut être utilisé pour différentes applications, par conséquent plusieurs nœuds puits peuvent coexister, c'est-à-dire, il peut y avoir potentiellement un nœud racine par application. Pour cela un DAG orienté vers la destination (DODAG) est construit pour chaque application selon une fonction spécifique (appelée fonction objectif) qui optimise un critère spécifié pour le routage des données, par exemple minimiser la distance au nœud racine dans le DODAG, maximiser le débit de transmission des données vers le nœud puits ou encore minimiser la consommation énergétique totale. Chaque DODAG est enraciné au nœud puits (DODAG racine) de l'application correspondante. À cette fin, le concept d'instance de DODAG a été introduit dans RPL. Une instance RPL rassemble un sous-ensemble de DODAGs dans un réseau de capteurs qui suivent la même fonction objectif. Plusieurs instances RPL peuvent s'exécuter simultanément, mais un nœud appartient au plus à un seul DODAG par instance RPL.

## 5.2. PRÉSENTATION DU PROTOCOLE DE ROUTAGE RPL

---

Lors de la construction du DODAG, RPL supporte deux modes d'opération, le mode avec

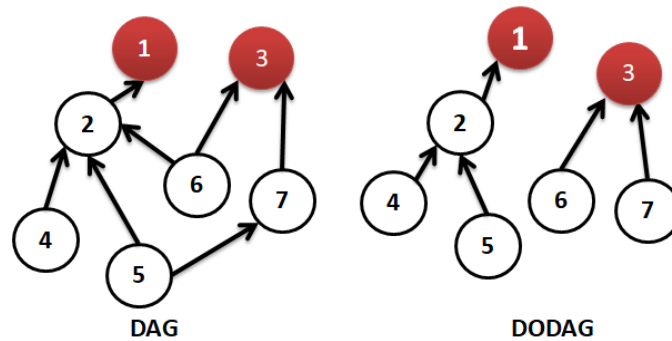


FIGURE 5.1 – Exemple de DAG et DODAG RPL

stockage et le mode sans stockage, mais un seul mode peut être choisi à la fois.

- **Mode avec stockage** : les capteurs sont configurés en tant que routeurs et maintiennent une table de routage. Ces tables de routage sont utilisées pour maintenir les routes vers les autres capteurs intermédiaires du DODAG. À chaque saut, le nœud examine sa table de routage et décide à quel voisin envoyer ensuite son paquet de données.
- **Mode sans stockage** : aucun capteur ne stocke les routes vers les autres capteurs intermédiaires du DODAG ; le seul endroit de stockage de la table de routage est la racine qui agit comme un routeur, c'est-à-dire, la racine indique dans le paquet de données le chemin complet vers la destination.

### 5.2.1 Messages de contrôle RPL

Pour créer et maintenir le DODAG, RPL spécifie un ensemble de messages de contrôle ICMPv6, dont la structure est donnée dans la Figure 5.2. Ces messages peuvent être de type DODAG Information Object (DIO), de type DODAG Information Solicitation (DIS) ou encore de type Destination Advertisement Object (DAO). Nous présentons ci-dessous chacun de ces types de messages :

- **DODAG Information Object (DIO)** : ce message DIO contient plusieurs paramètres de configuration RPL, dont un champ (i) RPLInstanceID qui indique l'affiliation du DODAG ainsi que d'autres paramètres tels que la fonction objectif associée

## 5.2. PRÉSENTATION DU PROTOCOLE DE ROUTAGE RPL

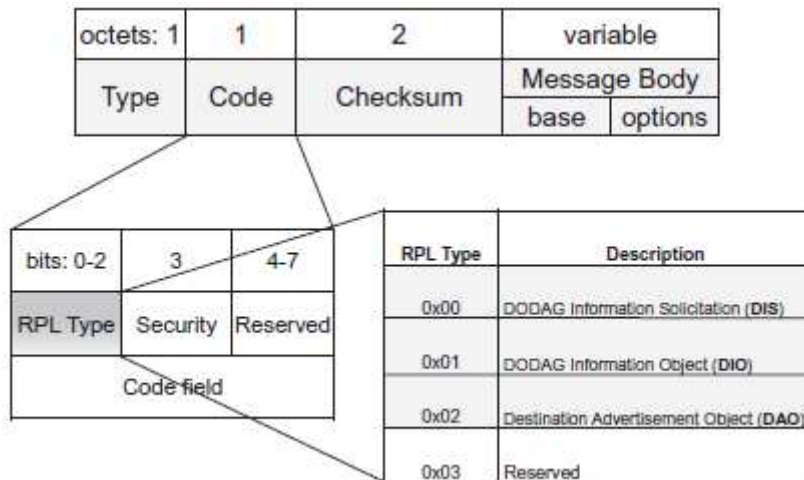


FIGURE 5.2 – Format des messages de contrôle RPL [Gaddour and Koubaa 2012]

pour le calcul des routes, (ii) le numéro de version qui indique la version du DODAG (ce numéro de version s'incrémente lors du changement de la topologie du DODAG). (iii) le rang (Rank) représente la position du nœud envoyant le message DIO dans le DODAG, (iv) DTSN (Destination Advertisement Trigger Sequence Number) indicateur de message DIO utilisé pour maintenir les routes montantes dans le DODAG, (v) Grounded (G) est un champ indiquant si le DODAG actuel satisfait l'objectif défini par l'application, (vi) mode d'opération (MOP) indique le mode de stockage, (vii) DODAGID indique l'identifiant du DODAG, et (viii) option DIO, où on trouve le champ *DAG metric container* qui définit la métrique de sélection des chemins et le poids qui lui est associé. Le format du message DIO est illustré dans la Figure 5.3.

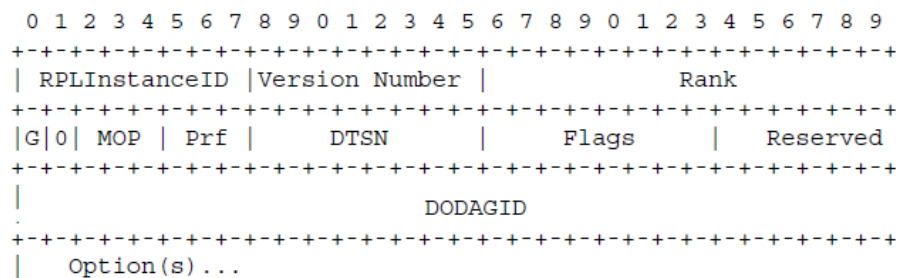


FIGURE 5.3 – Format des messages de type DIO [Vasseur et al. March 2012]



## 5.2. PRÉSENTATION DU PROTOCOLE DE ROUTAGE RPL

---

- **DODAG Information Solicitation (DIS)** : les messages de type DIS sont diffusés dans le voisinage de l'émetteur, l'objectif est de solliciter l'envoi d'informations par les nœuds dans ce voisinage déjà rattachés au DODAG. Ce type de message est aussi utilisé pour mettre à jour la liste des nœuds voisins. En réponse, le nœud émetteur recevra un message de type DIO.

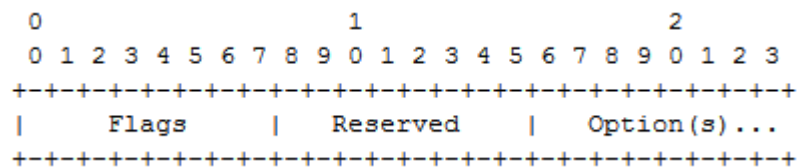


FIGURE 5.4 – Format des messages de type DIS [Vasseur et al. March 2012]

- **Destination Advertisement Object (DAO)** : RPL établit des itinéraires descendants en utilisant des messages DAO (Destination Advertisement Object). Les messages de type DAO sont envoyés en mode unicast, à partir des nœuds feuilles jusqu'au nœud racine (du bas vers le haut). Chaque nœud envoie un message DAO à son parent contenant le préfixe des nœuds de son sous-DODAG qui sont agrégés jusqu'à la racine. Le message DAO est une fonctionnalité facultative, nécessaire uniquement pour la prise en charge de trafic de type capteur à capteur (P2P). Comme le montre la Figure 5.5, les champs principaux du message DAO sont : (i) RPLInstanceID et (ii) DAO séquence.

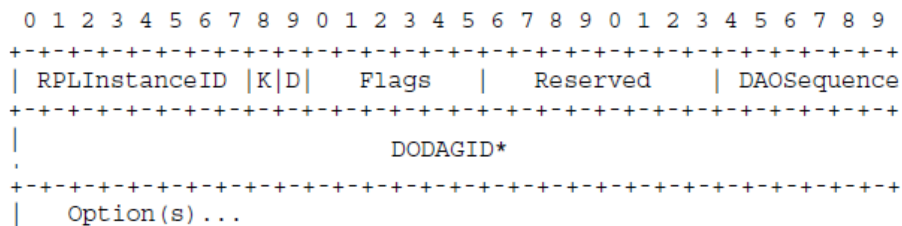


FIGURE 5.5 – Format des messages de type DAO [Vasseur et al. March 2012]

### 5.2.2 Construction du DODAG

La construction du DODAG commence au niveau du nœud racine, qui représente le nœud de collecte de données. Pour créer les chemins de transmission des données de la

racine vers les nœuds, le nœud racine commence par diffuser un message de type DIO pour annoncer les paramètres de configuration du DODAG tel que : le DODAGID, le rang et la Fonction Objectif considérée. Ce message sera reçu par les nœuds qui se trouvent dans le même rayon de communication que le nœud racine. Ces nœuds vont traiter le message DIO, puis décident de se joindre ou non au DODAG selon la fonction objectif, le rang des nœuds voisins et le coût du chemin annoncé. Dans le cas où un nœud se joint au DODAG, il sélectionne grâce à la fonction objectif son parent dans le DODAG puis met à jour sa position dans le DODAG, le coût de son chemin vers la racine et enfin diffuse localement son propre message DIO.

Plutôt que d'attendre le message DIO, un nœud du réseau peut également envoyer un message de type DIS demandant à son voisinage des informations de configuration afin de se rattacher à un DODAG.

### 5.2.3 Algorithme Trickle

Afin de maintenir à jour la topologie du DODAG, des messages DIO sont échangés périodiquement, sans quoi il est difficile de maintenir à jour les routes dans chaque DODAG. Ainsi, plus la topologie devient stable, moins les messages DIO sont envoyés. Cependant pour augmenter la durée de vie du réseau, RPL utilise un algorithme de planification adaptatif de diffusion des messages DIO, appelé Trickle [Philip and Heide 2011]. Cet algorithme normalisé par L'IETF optimise la fréquence de diffusion des messages dans le réseau. Cela est réalisé en ajustant un compteur de temps  $I$  entre l'intervalle de temps  $[I_{min}, I_{max}]$ . Un nœud capteur exécutant l'algorithme Trickle, envoie un message DIO toutes les  $I$  ms pendant la construction du DODAG et lorsque la construction a convergé (c'est-à-dire qu'il n'y a plus de changements) cet intervalle est doublé jusqu'à atteindre un intervalle maximum correspondant à  $I_{max}$  ms. Lors d'une re-configuration du DODAG due à l'ajout d'un nouveau nœud ou à la détection d'une incohérence (par exemple sur les rangs dans le DODAG), RPL réinitialise le compteur de temps  $I$  à  $I_{min}$ .

On peut résumer le déroulement de l'algorithme Trickle comme suit :

1. initialiser une variable  $I$  à une valeur entre  $[I_{min}, I_{max}]$ , généralement  $I = I_{min}$
2. initialiser un compteur  $c = 0$

3. à la réception d'un message DIO pour lequel le nœud est dans le même état que son précédent message DIO (le même rank, le même parent, le même identifiant DAG, ...), incrémenter le compteur  $c$  d'une valeur,  $c = c + 1$ .
4. au timeout de  $I$ , si  $c \geq k$  ( $k$  est la constante de redondance), l'algorithme Trickle supprime l'envoi du message DIO planifié. Sinon, le message est transmis. (La condition  $c \geq k$  signifie que le nœud possède suffisamment de voisins dans le même état que leurs précédents messages DIO, dans le cas contraire  $c < k$ , le nœud possède beaucoup trop de voisins dont l'état a changé).
5. si Trickle a détecté un message DIO d'un voisin dont l'état a changé, Trickle réinitialise  $I$  à  $I_{min}$ , puis ré-exécute l'algorithme à partir de l'étape 2.

#### 5.2.4 Mécanisme de détection de boucle

RPL comprend également un mécanisme pour détecter et supprimer les boucles de routage dans le DODAG. Ce mécanisme est basé sur les rangs des nœuds dans le DODAG. Les boucles sont évitées en assurant que les rangs des nœuds capteurs augmentent d'une manière strictement monotone le long des chemins d'un DODAG, de la racine jusqu'aux nœuds feuilles. Plus précisément, les nœuds de chaque niveau  $i$  dans le DODAG ont un rang inférieur aux nœuds du niveau  $i+1$ . Pour renforcer cette garantie le standard RPL impose un incrément minimum, appelé `rank_increase`. Par conséquent, un nœud capteur ne peut pas se rattacher en tant que fils dans le DODAG à un nœud voisin ayant un rang supérieur. Donc, pour détecter une incohérence, matérialisée par la réception d'un paquet de données ayant un rang plus élevé, chaque nœud compare son rang à celui de ses voisins. Lorsque le nœud détecte une boucle, il déclenche un mécanisme d'empoisonnement (c'est-à-dire qu'il se détache du DODAG et diffuse un rang infini) de manière à déclencher une reconstruction de son sous-DODAG.

### 5.3 Fonction Objectif

Une **fonction objectif (FO)** permet à chaque nœud du réseau de sélectionner la meilleure route à considérer pour la construction et le maintien d'un DODAG. Cette fonc-

tion prend en entrée un ou plusieurs paramètres (associés aux contraintes applicatives) et retourne en sortie une valeur numérique définissant le coût du chemin dans le DODAG suivant une ou plusieurs métriques. Le coût d'un chemin est calculé par la fonction objectif pour chaque nœud voisin appartenant au DODAG et permet ainsi de sélectionner comme parent (préférée) pour un nœud désirant se rattacher à un DODAG celui de meilleur coût. Chaque nœud calcule ensuite son rang (sa position) par rapport au nœud racine dans le DODAG en ajoutant un incrément minimum par rapport au rang de son parent.

La définition de la fonction objectif est séparée du reste du protocole RPL, ce qui lui permet en plus de trouver la meilleure route à destination de la racine, de se focaliser uniquement sur les définitions des métriques sans se soucier de gérer l'échange des messages de contrôle. De ce fait, elle permet à RPL de répondre à différents critères d'optimisation pour une grande variété d'applications. Pour un aperçu détaillé de la FO, le lecteur intéressé peut se référer à [Gaddour and Koubaa 2012].

Le groupe de travail ROLL a spécifié deux types de FO : Objectif Fonction zéro (OF0) et Minimum Rank with Hysteresis Objective Function (MRHOF). OF0 est la fonction objectif par défaut qui utilise le nombre de sauts comme métrique de routage. Quant à la fonction objectif MRHOF, elle minimise le nombre de retransmissions d'un paquet sur toute la route d'acheminement de données du capteur jusqu'à la racine. Qasem *et al.* [Qasem et al. 2015] ont comparé la performance des deux fonctions objectifs OF0 et MRHOF avec différents taux de réception de paquets. Leurs conclusions montrent que MRHOF est meilleure que OF0 pour la plupart des scénarii considérés.

Pour construire et mettre à jour le DODAG, chaque nœud non-racine doit sélectionner un parent *préférée*. Cette sélection est effectuée en calculant le coût du chemin pour chaque nœud pouvant être parent (voisin avec un rang inférieur). Le *coût du chemin* est une valeur numérique qui représente une propriété du chemin vers le nœud récepteur. Elle est calculée en additionnant la métrique nœud/liaison sélectionnée au coût du chemin annoncé (dans le cas d'une métrique additive par exemple). Le meilleur coût renvoyé par la FO en utilisant la métrique spécifiée est utilisé pour sélectionner le meilleur parent donc le chemin de transmission des données de ce nœud vers le nœud racine. Le coût du chemin est calculé à nouveau si la métrique nœud/liens est mise à jour ou si une nouvelle métrique est annoncée.

Lorsque MRHOF est utilisée, selon le mécanisme d'hystérésis, le parent courant est changé si la différence entre le coût courant et le nouveau coût du chemin est au moins égal à un seuil spécifié. Après avoir sélectionné son parent préféré  $P_i$ , un nœud non-racine  $i$  calcule son rang  $R(i)$  comme suit :  $R(i) = R(P_i) + \text{rank\_increase}$ , avec  $R(P_i)$  définissant le rang annoncé par  $P_i$  et  $\text{rank\_increase}$  l'incrément minimal sur le rang. Il est à noter que la racine d'un DODAG annonce un rang égal à  $\text{rank\_increase}$ . Le rang et le coût du chemin calculé par chaque nœud sont disséminés dans le message DIO.

### 5.4 Métriques de routage pour RPL

L'objectif principal de tout protocole de routage est d'avoir un mécanisme de sélection et de maintien des chemins respectant les critères et contraintes pour la transmission des données. Pour répondre à ce besoin, la conception du protocole de routage RPL se réfère à une métrique basée sur les caractéristiques du réseau et les applications cibles. Cependant plusieurs métriques de routage ont été proposées dans la littérature pour les réseaux de capteurs sans fil, dont le but est de prolonger la durée de vie du réseau, maximiser la fiabilité ou encore minimiser le délai de transmission de bout en bout (latence). En revanche assurer les exigences applicatives tout en respectant les caractéristiques et les limites dans le contexte des réseaux de capteurs rend difficile le choix et la conception d'une métrique.

**Une métrique de routage** est une valeur quantitative utilisée pour évaluer le coût du chemin de transmission de données [Winter et al. March 2012]. Elle peut être appliquée sur un lien de communication ou un nœud capteur. La métrique appliquée sur un nœud représente l'état du nœud comme l'énergie du nœud, ou sa position. La métrique appliquée sur un lien de communication reflète la qualité de ce dernier en terme de latence, débit ou nombre de retransmissions estimées (ETX). Pour une meilleure optimisation des chemins de routage, la métrique est utilisée en association avec la fonction objectif. Une métrique peut être définie localement ou globalement (agrégation des liens tout au long du chemin) [Vasseur et al. March 2012]. Dans le cas d'une *métrique locale*, le choix du parent préféré ne dépend que d'informations locales dans le voisinage du nœud, aucune visibilité sur l'état du reste du chemin vers la racine. En revanche dans le cas d'une *métrique globale* ou *agrégée*, le choix du parent préféré dépend non seulement des informations locales dans

le voisinage du nœud mais aussi de l'état de tout le reste des nœuds/liens sur les mêmes chemins construits jusqu'à la racine.

Dans le cas de métrique agrégée, la valeur de la métrique est ajustée durant la propagation du message DIO dans le DODAG. Elle peut être de type additive, multiplicative, minimum et maximum. Si  $w(i, j)$  représente la valeur de la métrique entre les nœuds  $i$  et  $j$  alors pour tout chemin  $p(i, j, k, \dots, q, r)$ , la métrique est de type :

- Additive si :  $w(p) = w(i, j) + w(j, k) + \dots + w(q, r)$
- Multiplicative si :  $w(p) = w(i, j) * w(j, k) * \dots * w(q, r)$
- Concave si :  $w(p) = \min[w(i, j), w(j, k), \dots, w(q, r)]$   
ou  $w(p) = \max[w(i, j), w(j, k), \dots, w(q, r)]$

**Exemple de métrique additive :** La métrique latence est de type additive, elle est définie comme étant une somme des latences de transmission de chaque lien de communication le long du chemin vers la racine. Chaque nœud sélectionne son parent préféré le nœud annonçant le chemin ayant la latence minimale.

**Exemple de métrique concave :** La métrique débit est de type concave, elle définit la valeur de débit minimale le long du chemin vers la racine. Chaque nœud sélectionne son parent préféré le nœud annonçant le chemin ayant le débit maximal.

Si la fonction objectif considère une seule métrique agrégée lors de la construction du DODAG, alors la relation d'ordre pour la sélection du meilleur chemin est dérivée du type de la métrique. Pour la métrique nombre de sauts (Hop count) par exemple, le meilleur chemin est celui ayant le moins de sauts possibles. Mais pour la métrique énergie, le meilleur chemin est celui ayant l'énergie consommée minimale. Dans ce cas on dit que la fonction objectif se base sur une métrique primaire. **Une métrique primaire** est une métrique qui ne considère qu'un seul critère à optimiser dans le calcul du meilleur chemin.

Les applications et contraintes associées aux réseaux de capteurs nécessitent bien souvent de prendre en compte plusieurs critères dans le choix des routes vers le puits pour acheminer les informations. Par exemple, on désire emprunter des chemins qui d'une part sont composés de nœuds avec un niveau de batterie important et d'autre part ayant un nombre de sauts

faible. Il est donc crucial de définir une fonction objectif pour le routage qui considère non pas une mais plusieurs métriques agrégées. Cela est pris en compte par la notion de *métrique composite*. **Une métrique composite** est une métrique qui combine plusieurs métriques primaires suivant un processus défini dans la référence [Karkazis et al. 2012]. Concevoir une telle métrique peut se révéler plus complexe si les différentes métriques n'ont pas toutes la même relation d'ordre pour sélectionner le meilleur chemin. En effet, la définition de la fonction objectif associée est délicate.

### 5.4.1 Métriques de routage liées à l'énergie proposées pour RPL

Assurer les exigences de l'application tout en respectant les caractéristiques et les limites des WSNs ne facilite pas le choix d'une métrique à considérer. Cependant, afin de supporter toutes ces exigences, un ou plusieurs paramètres doivent être pris en compte dans le routage.

L'énergie est un critère clé dans les performances des réseaux de capteurs sans fil. Par conséquent, dans ce chapitre nous nous intéresserons à la prise en compte et l'optimisation de la consommation énergétique dans le routage réalisé par RPL. Chacune de ces métriques cherche à sélectionner un chemin préservant le plus haut niveau d'énergie de chaque nœud capteur au puits. Ce qui les différencie c'est la manière dont l'énergie est modélisée, ainsi que la méthode d'agrégation des informations le long des chemins. Nous présenterons différentes métriques de routage liées à l'optimisation de l'énergie proposées dans la littérature, et analyserons les forces et faiblesses de chacune ainsi que les performances atteintes.

#### 5.4.1.1 Métriques additives

- **Métrique ETX (Expected Transmission count)** : cette métrique agrégée additive permet d'estimer le nombre de retransmissions sur un lien de communication avant la réception d'un acquittement (ACK). Plus petite est cette valeur, meilleure est la qualité du lien de communication. Cette métrique permet d'optimiser indirectement l'énergie consommée par chaque capteur. En effet, à chaque saut dans le DODAG moins un message contenant des données à transmettre sera retransmis, plus faible sera l'énergie consommée dans le DODAG pour collecter les données jusqu'au puits. Cette métrique est utilisée par défaut avec la fonction objectif MRHOF.

#### 5.4. MÉTRIQUES DE ROUTAGE POUR RPL

Elle peut être calculée comme suit :

$$ETX = \frac{1}{PDR_{s \rightarrow d} \times PDR_{d \rightarrow s}}, \quad (5.1)$$

avec  $PDR_{u \rightarrow v}$  définissant l'estimation du taux de réception des paquets envoyés de  $u$  à  $v$ . Parmi les voisins  $N_i$  d'un nœud  $i$ , en utilisant MRHOF  $i$  choisit comme parent préféré, le voisin dont le nombre total de retransmissions le long de son chemin jusqu'au puits est le plus faible, c'est-à-dire le voisin ayant le plus faible coût défini par la formule suivante :  $\min_{j \in N_i} ETX_j$ . Il est à noter que seuls les voisins déjà rattachés au DODAG ne seront considérés par  $i$ , les autres nœuds sont identifiés car ils ont un rang infini. Le rang  $R(i)$  du nœud  $i$  via le parent sélectionné, appelé aussi *parent préféré*,  $P_i$  est donné par :  $R(i) = R(P_i) + rank\_increase$ . Ce rang est diffusé ensuite par le nœud  $i$  dans le message DIO, et il est utilisé à la fois pour éviter les boucles de routage et pour indiquer le coût du chemin. La Figure 5.6 illustre la sélection des chemins dans le DODAG en se basant sur la métrique ETX. Le nœud 6 peut choisir le nœud 2 ou 3 comme son parent préféré. Le nœud 3 est celui qui a été sélectionné puisqu'il diffuse le plus faible ETX cumulatif vers la racine (le plus faible nombre de retransmissions du nœud 6 jusqu'à la racine).

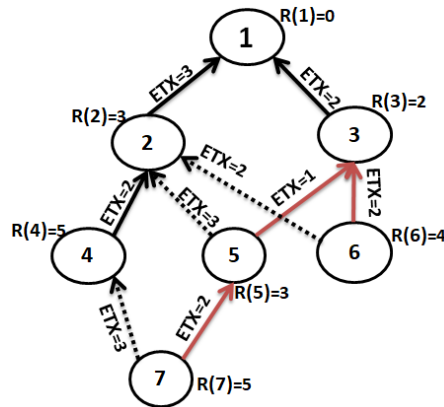


FIGURE 5.6 – Construction du DODAG avec la métrique ETX.

Les métriques présentées ci-après sont spécialement conçues pour optimiser la durée de vie du réseau.

- **Métrique ENG-TOT** : cette première métrique tenant compte de l'état des bat-



teries des nœuds capteurs se base sur *l'énergie résiduelle*, notée  $ResEng_i$  pour le nœud  $i$ , pour modéliser l'énergie. L'énergie résiduelle est calculée comme la différence entre le niveau maximum de la batterie  $MaxEng_i$  et l'énergie consommée  $EngCons_i$  sur le nœud  $i$ , c'est-à-dire  $ResEng_i = MaxEng_i - EngCons_i$ . L'énergie consommée par un nœud capteur est essentiellement due à la communication radio (c'est-à-dire la transmission et l'écoute). Nous considérerons qu'il n'y a pas ou peu de traitements réalisés sur les nœuds capteurs, et ceux-ci consomment une quantité d'énergie négligeable par rapport aux communications.

Demicheli [Demicheli 2014] a proposé la première métrique RPL qui considère l'énergie totale consommée par les nœuds capteurs le long d'un chemin. Comme dans la métrique ETX, la proposition de Demicheli [Demicheli 2014] est une métrique additive qui agrège par une somme les énergies consommées sur chaque chemin dans le DODAG. Chaque nœud  $i$  envoie dans les messages DIO son rang, obtenu en ajoutant un incrément (fixé à 16 par l'auteur) au Rang  $R(P_i)$  de son parent préféré  $P_i$ , ainsi que l'énergie consommée sur son chemin, notée  $PathEngCons(i)$ , dans le champ Metric Container. Cette valeur est calculée comme suit par chaque nœud  $i$  :

$$PathEngCons(i) = PathEngCons(P_i) + EngCons_i, \quad (5.2)$$

avec  $PathEngCons(P_i)$  qui représente l'énergie consommée par les nœuds capteurs appartenant au chemin du parent préféré  $P_i$  de  $i$  jusqu'au puits du DODAG, et  $EngCons_i$  est l'énergie consommée par le nœud  $i$ . Le parent préféré est le voisin dans le DODAG ayant la plus faible énergie consommée sur son chemin, cela est défini par la formule :  $\min_{j \in N_i} PathEngCons(j)$ . L'inconvénient principal de cette métrique est le fait qu'un chemin vers la racine peut contenir un nœud avec une énergie résiduelle très faible (que l'on peut appeler *nœud bloquant* en reprenant la terminologie introduite par O. Iova dans son manuscrit de thèse [Iova 2014]). En effet, un faible niveau de batterie d'un nœud peut être masqué sur un chemin par d'autres niveaux de batterie plus élevés un peu plus bas sur le chemin, ceci est due à la somme réalisée sur les niveaux de batterie des nœuds le long des chemins dans le DODAG.

- Xu et al. [Xu and Lu 2013] ont proposé une métrique additive proche de celle décrite par Demicheli [Demicheli 2014]. Xu et al. ont modélisé l'énergie de chaque nœud capteur  $i$  à l'aide de l'énergie résiduelle  $ResEng_i$  et non l'énergie consommée. Tout comme la métrique ETX, le choix du parent préféré se fait sur le rang, le champ Metric Container du message DIO n'est donc pas utilisé. Le rang  $R(i)$  de chaque nœud  $i$  est obtenu en ajoutant  $ResEng_i$  au rang  $R(P_i)$  de son parent préféré  $P_i$ , c'est-à-dire  $R(i) = R(P_i) + ResEng_i$ . Ce rang est ensuite diffusé par le nœud  $i$  dans le message DIO. Le parent préféré est le voisin ayant la plus grande énergie résiduelle sur son chemin, à savoir celui atteignant la valeur retournée par la formule suivante :  $\max_{j \in N_i} R(j)$ . Il est à noter que chaque nœud  $i$  ne considère que ses voisins dans le DODAG (c'est-à-dire avec un rang différent de l'infini).

Pour cela, le rang  $R(i)$  de chaque nœud  $i$  est obtenu en ajoutant  $ResEng_i$  au rang  $R(P_i)$  de son parent préféré  $P_i$ , c'est-à-dire  $R(i) = R(P_i) + ResEng_i$ . Ce rang est ensuite diffusé par le nœud  $i$  dans le message DIO. Cependant, cette métrique possède les mêmes inconvénients que celle proposée par Demicheli [Demicheli 2014].

### 5.4.1.2 Métriques concave :

- **Métrique ENG\_MinMax** : Kamgueu et al. [Kamgueu et al. 2013] ont proposé une métrique de routage pour RPL afin d'optimiser la consommation énergétique dans le réseau de capteurs. Cette métrique est basée sur l'énergie résiduelle pour modéliser l'énergie d'un nœud. Contrairement aux métriques présentées précédemment, l'agrégation le long des chemins est réalisée à l'aide d'une fonction de type minimum. La plus faible énergie résiduelle est propagée le long de chaque chemin. Cela permet d'avoir une estimation de la plus faible énergie restante sur chaque chemin, et permettre à un nœud de choisir comme parent le voisin ayant un chemin jusqu'au puits dont la plus faible énergie restante est la plus grande. Nous noterons  $PW_i$  le coût du chemin de  $i$  au puits dans le DODAG, représentant la plus faible énergie résiduelle sur ce chemin. Plus précisément, chaque nœud sélectionne comme parent préféré celui qui annonce un coût de chemin le plus élevé (c'est-à-dire parmi ses voisins  $j$  celui ayant la plus grande valeur  $PW_j$ ), puis  $i$  utilise le champ Metric

Container dans le message DIO pour diffuser le coût de son chemin comme étant le minimum entre son énergie résiduelle  $ResEng_i$  et le coût du chemin envoyé par son parent préféré. La Figure 5.7 illustre la construction d'un DODAG en se basant sur la métrique  $ENG\_MinMax$ . Le nœud 6 peut choisir entre le nœud 2 ou 3 son parent préféré. Le nœud 2 est le meilleur parent préféré puisque l'énergie résiduelle propagée par le nœud 2 ( $PW_2=200$ ) est plus grande que l'énergie résiduelle propagée par le nœud 3 ( $PW_3=155$ ). Par conséquent, en se basant sur la métrique  $ENG\_MinMax$  on peut constater que le choix du parent préféré est construit sur la base de l'énergie résiduelle même si la qualité du lien de communication correspondante est mauvaise. Cela peut provoquer l'épuisement rapide de la batterie de certain nœud du chemin construit. L'équation 5.3 décrit le calcul de la valeur  $PW_i$  associée à un nœud  $i$  dans le DODAG (avec  $N_i$  les voisins de  $i$ ) :

$$PW_i = \min(\max_{j \in N_i}\{PW_j\}, ResEng_i), \quad (5.3)$$

Le rang  $R(i)$  du nœud  $i$  est égal au rang du parent préféré  $R(P_i)$  auquel on ajoute l'énergie résiduelle  $ResEng_i$  de  $i$ , c'est-à-dire,  $R(i) = R(P_i) + ResEng_i$ .

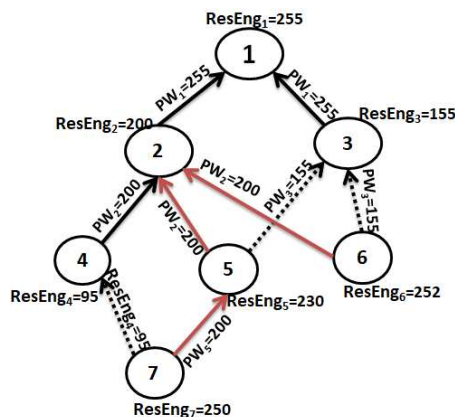


FIGURE 5.7 – Construction du DODAG avec la métrique  $ENG\_MinMax$ .

- **Métrique ELT** : récemment, Iova et al. [Iova et al. 2014] ont proposé une autre métrique de routage de type concave, appelée Expected LifeTime (ELT), permettant une estimation plus fine de la durée de vie des chemins dans le DODAG. Pour établir cette estimation, plusieurs paramètres sont pris en compte : le nombre de

## 5.4. MÉTRIQUES DE ROUTAGE POUR RPL

retransmissions, l'énergie résiduelle et le trafic. Chaque nœud  $i$  calcule une estimation de la durée de vie  $ELT_i$  du chemin vers le puits dans le DODAG en utilisant l'Équation 5.4.

$$ELT_i = \frac{ResEng_i}{T_i \times \frac{ETX(i, P_i)}{Data\_Rate} \times PowTX_i}, \quad (5.4)$$

où  $T_i$  est le trafic de  $i$  (en bits/s),  $ETX(i, P_i)$  est la valeur ETX du lien vers le parent préféré  $P_i$  de  $i$ ,  $Data\_Rate$  est le débit avec lequel les données sont envoyées (en bits/s) et  $PowTX_i$  est la puissance consommée par une transmission radio sur  $i$ . La durée de vie estimée est propagée par chaque nœud le long des chemins dans le DODAG de haut en bas à l'aide du champ Metric container des messages DIO. Chaque nœud  $i$  choisit comme parent préféré le voisin qui offre un chemin avec une estimation la plus élevée, c'est-à-dire en suivant la formule suivante :  $\max_{j \in N_i} ELT_j$ . Le rang associé à un nœud  $i$  dans le DODAG est calculé de la même façon que pour la métrique ETX. La Figure 5.8 illustre la construction d'un DODAG en se basant sur la métrique ELT. Le nœud 6 peut choisir le nœud 3 comme son parent préféré au lieu du nœud 2 puisque cela permettra d'équilibrer la consommation énergétique des nœuds les plus restreints (le nœud 2 dans le cas de la métrique ENG\_MinMax ou le nœud 3 dans le cas de la métrique ETX).

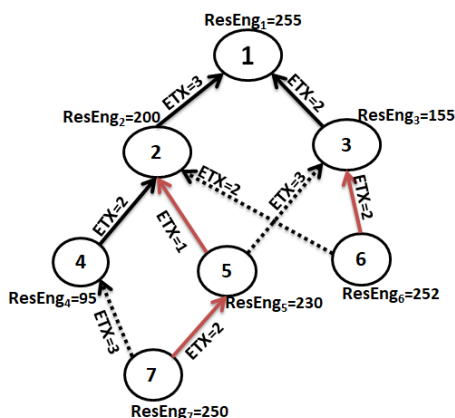


FIGURE 5.8 – Construction du DODAG avec la métrique ELT.

### 5.4.1.3 Métrique locale multi-paramètres

Les métriques présentées précédemment permettaient d'agrèger des informations sur l'état des batteries des nœuds ou le nombre de retransmissions le long des chemins dans le DODAG. Nous présentons ici une métrique qui cherche à optimiser la durée du réseau en faisant des *choix locaux*, c'est-à-dire en ne se basant que sur des informations propre à leur voisinage à un saut. Il n'y a donc pas d'agrégation d'informations sur l'état des nœuds ou des liens le long des chemins dans le DODAG.

- **Métrique R** : Chang et al. [Chang et al. 2013] ont défini une métrique locale basée sur plusieurs paramètres locaux pour l'optimisation de la durée de vie dans le réseau. Cette métrique combine pour chaque nœud  $i$  le nombre de retransmissions ( $ETX_j$ ) sur le lien de communication avec un voisin  $j$  et l'énergie résiduelle  $ResEng_i$  de  $i$ . Comme l'échelle de valeurs est différentes pour ces deux paramètres ( $ETX_i$  varie de 1 à 10 et  $ResEng_i$  de 1 à 255), donc ils seront donc exprimés sur la même échelle entre 0 et 1 dans la fonction objectif. L'équation 5.5 définit cette fonction objectif :

$$R_i = \alpha \frac{ETX_j}{Max\_ETX} + (1 - \alpha) \times \left(1 - \frac{ResEng_j}{MaxEng}\right), \quad (5.5)$$

avec  $j$  un voisin de  $i$ ,  $\alpha$  le facteur de pondération entre les deux paramètres,  $Max\_ETX$  et  $MaxEng$  sont respectivement la valeur maximale ETX d'un lien de communication et le niveau de batterie maximum d'un nœud dans le réseau. Dans la suite, nous considérerons que la qualité du lien de communication et l'énergie résiduelle ont autant de poids dans le choix des chemins, donc on choisira  $\alpha = 0,5$  comme dans [Chang et al. 2013]. Chaque nœud  $i$  choisit comme parent préféré le voisin  $j$  retournant la plus petite valeur  $R_i$  définie par l'équation 5.5. Enfin, le rang de chaque nœud  $i$  est calculé de la même manière que pour la métrique ETX (décrite précédemment).

Le tableau 5.1 présente un résumé des métriques étudiées précédemment avec le nom de chaque métrique, le type d'agrégation, et les paramètres utilisés.

TABLE 5.1 – Métriques étudiées

Métriques	Type d'agrégation	Paramètres utilisés
ETX [Vasseur et al. March 2012]	additive	nombre de retransmissions
ENG-ToT [Demicheli 2014]	additive	énergie consommée
ENG_MinMax [Kamgueu et al. 2013]	concave	énergie résiduelle
ELT [Iova et al. 2014]	concave	nombre de retransmissions, énergie résiduelle et le trafic
R [Chang et al. 2013]	locale	nombre de retransmissions et énergie résiduelle

## 5.5 Évaluation des métriques

Dans cette section, nous présentons l'évaluation que nous avons menée avec les métriques de routage RPL décrites dans la section précédente.

### 5.5.1 Configuration des simulations

Avant de présenter les résultats de simulation, nous décrivons tout d'abord le cadre utilisé pour les simulations. Nous utilisons le simulateur Cooja [Osterlind et al. 2006] avec ContikiOS 3.0, un simulateur flexible basé sur Java qui prend en charge le langage de programmation C en utilisant l'interface Java Native Interface. Nous simulons un réseau de capteurs sans fil LLN constitué de 56 nœuds de type Tmote sky mote [Polastre et al. 2005] (une plate-forme de capteurs largement utilisée) avec un transceiver de type CC2420 de 2,4 GHz à la couche physique compatible avec la norme IEEE 802.15.4. La portée de communication de chaque capteur a été fixée à 45m alors que le rayon d'interférence a été fixé à 70m. Nous utilisons le protocole ContikiMac [Dunkels 2011] au niveau de la couche MAC, car ce protocole fournit un moyen efficace d'accès au canal de transmission avec une faible consommation d'énergie (le transceiver est éteint 99% du temps de fonctionnement des capteurs). Nous nous appuyons également sur le protocole RPL au niveau de la couche réseau pour le routage et nous simulons un trafic de type multi-points à point, où chaque nœud envoie périodiquement à la racine des paquets de données, avec un taux de 6 paquets par minute. On considère que chaque nœud commence à envoyer son premier paquet de données 65s après le début d'une simulation. Les principaux paramètres utilisés

## 5.5. ÉVALUATION DES MÉTRIQUES

---

pour la simulation sont résumés dans le Tableau 5.2. Les valeurs TX et RX définissent respectivement les taux de réussite lors de l'émission et de réception des données. Nous considérons trois valeurs différentes pour RX afin de modéliser les pertes sur le canal : un cas parfait sans perte de paquets (RX=100%) et un autre plus réaliste avec 40% de perte de paquets (RX=60%). Dans le cas de 20% de perte de paquets (RX=80%), les résultats étaient proches de ceux avec RX=100%.

Les 56 nœuds sont déployés sur une surface carrée de 300 x 300 m. Dans nos simulations nous considérons deux topologies : une topologie en grille et une autre topologie arbitraire. Dans la topologie grille, la racine est située dans le coin inférieur droit (Figure 5.9(a)). Cet emplacement de la racine représente l'un des pires scénarii (comparé à une racine située au centre de la grille). En effet, une congestion plus élevée est observée autour de la racine, car très peu de capteurs sont dans son voisinage. En outre, dans cette topologie, nous avons une densité moyenne de 6.5 nœuds, la moitié des nœuds ont 8 voisins tandis que les autres ont 5 voisins dans le réseau (sauf les quatre nœuds sur les coins de la topologie qui ont 3 voisins). Dans la topologie arbitraire, les nœuds sont placés au hasard dans une même surface carrée de 300 x 300 m (Figure 5.9(b)) avec la racine (nœud vert) située à gauche. La densité dans cette deuxième topologie est plus petite que dans la topologie grille, les nœuds ont trois voisins en moyenne. Le Tableau 5.3 indique la distance séparant les nœuds de la racine dans les deux types de topologie.

Dans nos simulations, nous concéderons que tous les nœuds ont la même charge de batterie initiale de 853 mAh. Pour une meilleure utilisation, nous avons représenté cette charge dans Contiki sur une échelle de 255 paliers (comme suggéré dans la norme RPL [Winter et al. 2012]) et chaque palier représente 3,345mAh sur la capacité de la batterie.

### 5.5.2 Critères d'évaluation

Dans nos travaux, nous évaluons les performances de chaque métrique selon quatre critères définis ci-dessous :

- **Énergie consommée** : afin de calculer la consommation d'énergie, nous nous appuyons sur le mécanisme Power-trace [Dunkels et al. 2011] fourni par Contiki. Au

## 5.5. ÉVALUATION DES MÉTRIQUES

TABLE 5.2 – Paramètres de Simulation

Paramètres	Valeur
FO	MRHOF
RPL MOP	NO_DOWNWARD_ROUTE
Start Delay	65s
Imin	$2^{12}$ ms
Imax	$2^{20}$ ms
Intervalle d'envoi de data	6 pkt/min
RX - TX	100% ou 80% - 60%
Distance TX	45m
Surface d'interférence	70m

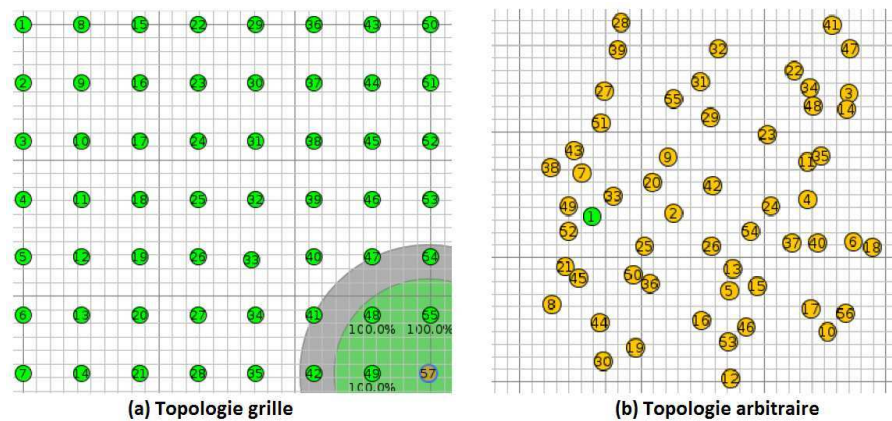


FIGURE 5.9 – Les topologies de simulation

TABLE 5.3 – Distribution des nœuds suivant la distance Euclidienne à la racine

Topologie grille		Topologie arbitraire	
Distance à la racine	Nombre de nœuds	Distance à la racine	Nombre de nœuds
70m	3	0-45m	4
140m	5	45m-90m	14
210m	7	90m-135m	9
280m	9	135m-185m	11
350m	11	185m-230m	6
420m	13	230m-275m	8
490m	7	275m	3

cours de nos expériences, nous nous concentrons sur la période pendant laquelle la radio est allumée. Nous calculons ensuite la consommation d'énergie  $E_i$  pour chaque



nœud  $i$  (en mJ) comme suit :

$$E_i = \frac{(T_{CPU} * I_{CPU} + T_{RX} * I_{RX} + T_{TX} * I_{TX}) * V}{R_{timer}} \quad (5.6)$$

où  $V$  correspond à la tension de la batterie ( $=3.6V$ ),  $I_{CPU}$  ( $=1.8mAh$ ),  $I_{RX}$  ( $=20mAh$ ) et  $I_{TX}$  ( $=17.7mAh$ ) représentent la charge de la batterie qui a été consommée respectivement pendant le temps d'utilisation du processeur  $T_{CPU}$ , le temps où la radio du nœud  $i$  est en écoute  $T_{RX}$  et le temps où la radio du nœud  $i$  est en transmission  $T_{TX}$  (tous exprimés en ticks), et  $R_{timer}$  représente le nombre de ticks par seconde ( $=32768$  ticks/s sur la plateforme de simulation).

- **Taux de paquets transmis (PDR)** : cela représente le nombre de paquets qui sont reçus avec succès par le puits divisé par le nombre de paquets envoyés par tous les nœuds au puits.
- **Délai de bout en bout** : cette valeur définit pour chaque nœud source le temps (dans le simulateur) qui s'est écoulé entre la génération du paquet par la couche application sur le nœud source et la réception de celui-ci par le puits au niveau de la couche application également.
- **Messages de contrôle** : nous comptabilisons le nombre de messages de contrôle RPL (c'est-à-dire les messages DIO et DIS) échangés dans le réseau durant la durée d'une simulation.

### 5.5.3 Simulations et Analyse

Dans ce qui suit, nous présentons nos résultats de simulation. En particulier, nous considérons les cinq métriques RPL présentées précédemment conçues pour augmenter la durée de vie du réseau :

- **ETX** : il s'agit de la métrique par défaut de RPL, qui tient compte du nombre de retransmissions des paquets pour chaque lien de communication le long des chemins du DODAG.
- **Énergie totale (ENG-ToT)** : nous considérons la métrique proposée par Demicheli [Demicheli 2014] dans laquelle le coût du chemin de transmission des données est représenté par la somme des énergies consommées de chaque nœud source jusqu'au puits.

- **Energie résiduelle minimale (ENG\_MinMax)** : nous considérons la métrique proposée par Kamgueu et al. [Kamgueu et al. 2013] dans laquelle le coût de chaque chemin est donné par l'énergie résiduelle minimale le long de celui-ci.
- **Energie résiduelle + ETX (R)** : nous avons sélectionné la métrique proposée par Chang et al. [Chang et al. 2013] dans laquelle le coût du chemin est égal à une fonction pondérée intégrant les paramètres ETX et l'énergie résiduelle. Les deux paramètres ont le même poids dans notre simulation, c'est-à-dire que nous définissons  $\alpha = 0,5$ .
- **Expected lifetime (ELT)** : nous choisissons la métrique proposée par Iova et al. [Iova et al. 2014] car elle modélise assez finement la durée de vie estimée d'un nœud dans le réseau. Cette métrique prend en compte à la fois l'énergie résiduelle, le nombre de retransmissions sur le lien de communication, le taux de trafic du nœud, et la puissance de transmission. D'une part, l'énergie résiduelle et le nombre de retransmissions sont utilisés pour estimer le nombre maximum de paquets pouvant être transmis via un chemin, et d'autre part le taux de trafic indiquera une estimation de la durée maximale pendant laquelle le chemin sera présent dans le DODAG. Cependant, dans notre modélisation de cette métrique sous contiki nous ne mettons pas en œuvre le trafic attendu associé à chaque nœud, car il nécessite d'échanger des messages de contrôle supplémentaires pour estimer le trafic dans le sous-DODAG de chaque nœud.

Nous présentons dans la suite les résultats de simulation obtenus pour les quatre critères décrits précédemment. Les résultats représentent la moyenne des valeurs obtenus sur 10 simulations, chacune d'une durée de 5 heures.

**Consommation d'énergie** : nous considérons d'abord l'énergie consommée par chaque métrique de routage (Figure 5.10). Dans la Figure 5.10, le pourcentage de temps pendant lequel la radio est allumée reflète principalement l'énergie consommée par le protocole RPL. Dans une topologie grille avec RX=100% (Figure 5.10(a)), la consommation d'énergie augmente et diminue ensuite en fonction de la distance au nœud puits (racine) pour

## 5.5. ÉVALUATION DES MÉTRIQUES

---

les cinq métriques. Cette augmentation est due au fait que la racine représente un goulot d'étranglement. ETX est la métrique qui a la plus forte consommation d'énergie autour de la racine (au moins 2 sauts de la racine), suivie par les autres métriques énergétiques ENG\_MinMax et ELT, puis ENG-TOT et R. Cependant, pour les nœuds les plus éloignés, l'énergie consommée par la métrique ETX est inférieure comparée aux métriques ENG\_MinMax, ELT et ENG-TOT en raison des faibles taux de retransmission. La métrique R atteint la consommation d'énergie la plus faible par rapport aux autres métriques, la radio reste allumée au plus 1% du temps de simulation. Dans la topologie grille avec RX=60%, les cinq métriques se comportent de la même manière que pour le cas RX=100%. Nous observons une légère augmentation de la consommation d'énergie due à l'augmentation du nombre de retransmissions. De plus, la métrique R affiche des résultats similaires à ceux obtenus avec d'autres métriques énergétiques, contrairement au cas RX=100%. La métrique R donne les meilleurs résultats dans la première moitié du réseau et ensuite nous obtenons des résultats similaires pour toutes les métriques dans la seconde moitié du réseau. Dans le cas de la topologie arbitraire (Figures 5.10(c) et 5.10(d)), on observe le même comportement entre RX=100% et RX=60%. La consommation d'énergie est un peu plus élevée dans le cas où RX=60% à cause des pertes qui induisent des retransmissions de paquets. La consommation d'énergie diminue et augmente alors en fonction de la distance à la racine pour les cinq métriques. La consommation d'énergie est plus élevée autour du puits, ceci est dû à un goulot d'étranglement, puis elle diminue jusqu'à une certaine distance de la racine (135m), puis augmente encore une autre fois en raison de la perte de paquets cette fois. Pour les deux topologies, nous observons une faible consommation d'énergie pour la métrique ENG-TOT comparée aux autres métriques, mais c'est un effet de bord dû au faible nombre de paquets transmis comme l'indique le faible PDR (comme expliqué ci-dessous pour le critère PDR).

Le Tableau 5.4 présente pour chaque métrique le pourcentage d'énergie consommée après 5 heures de simulation et une extrapolation de la durée de vie du réseau exprimée en jours. Il est à noter que la courte durée de vie du réseau est liée à la faible charge initiale de la batterie (de 853 mAh au lieu de 2000 mAh dans la plate-forme sky mote [Nataf and Festor 2013]). Nous observons que la métrique R surpasse toutes les autres métriques dans

## 5.5. ÉVALUATION DES MÉTRIQUES

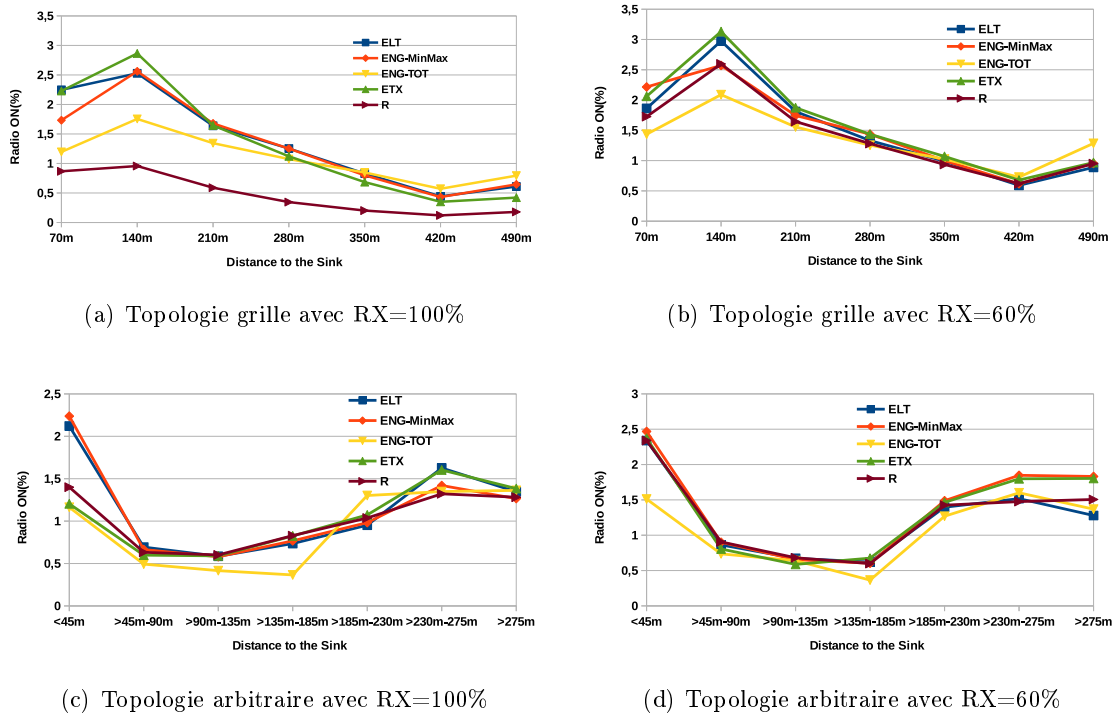


FIGURE 5.10 – Consommation énergétique.

le cas de la topologie grille avec RX=100%. Cette métrique atteint une durée de vie de 133 jours, environ 5 à 7 fois mieux que les autres métriques cherchant à optimiser l'énergie. Toutes les autres métriques ont une durée de vie du réseau similaire (excepté ENG-TOT). Dans le cas de la topologie grille avec RX=60%, nous observons que la métrique R obtient une durée de vie équivalente aux autres métriques. Ceci est dû aux retransmissions de paquets à cause du taux de perte élevé. Sur une topologie arbitraire avec RX=100% ou RX=60%, toutes les métriques (excepté ENG-TOT) donnent des durées de vie de réseau similaires (de 10 à 13 jours). Elles représentent des valeurs deux fois inférieures par rapport à la topologie grille. Ceci est une conséquence des retransmissions de paquets en raison d'un faible PDR induit par la faible densité dans le réseau qui est en moyenne égale à 3 (voir analyse du PDR ci-dessous). Comme déjà remarqué, la métrique ENG-TOT a une bonne durée de vie du réseau en général, mais c'est un effet dû à la faible charge du réseau causée par un faible PDR.

TABLE 5.4 – Durée de vie du réseau -1-

<b>Topologie Grille</b>					
	ENG_MinMax	ENG-TOT	ETX	ELT	R
RX= 100 %	1.12 %	0.38 %	1.15 %	0.97 %	0.157 %
	124698 mj	42212 mj	166098 mj	107822 mj	17359 mj
	19 jours	55 jours	18 jours	22 jours	133 jours
RX= 60 %	0.81 %	0.40 %	1.37 %	0.82 %	0.93 %
	88630 mj	43803 mj	151250 mj	91583 mj	103099 mj
	26 jours	52 jours	15 jours	25 jours	23 jours
<b>Topologie arbitraire</b>					
	ENG_MinMax	ENG-TOT	ETX	ELT	R
RX= 100 %	1.77 %	0.71 %	1.71 %	1.71 %	1.78 %
	196499 mj	78502 mj	188999 mj	189759 mj	196700 mj
	11 jours	29 jours	13 jours	13 jours	12 jours
RX= 60 %	1.96 %	0.84 %	1.98 %	1.95 %	1.91 %
	219994 mj	93744 mj	220104 mj	215300 mj	211666 mj
	10 jours	24 jours	10 jours	11 jours	11 jours

**Taux de livraison des paquets (PDR) :** dans la Figure 5.11, le taux de livraison des paquets (PDR) diminue en fonction de la distance à la racine dans les deux types de topologie. Globalement, la métrique ETX perd moins de paquets que les autres métriques, parce qu'elle prend en compte la qualité des liens de communication lors du choix du meilleur parent, tandis que la métrique ENG-TOT qui ignore la qualité de la liaison présente le pire PDR. Il est à noter que dans une topologie grille avec RX=100% (Figure 5.11(a)), les meilleurs résultats sont donnés par la métrique R avec un PDR proche de 100% pour tous les nœuds, car elle tient compte à la fois de la qualité des liens de communication et de l'énergie résiduelle du parent, contrairement à la métrique ETX pour laquelle une quantité importante de paquets est perdue en raison de la batterie épuisée sur les chemins dans le DODAG. Les métriques ENG\_MinMax et ELT atteignent un PDR un peu inférieur à 100% près de la racine, et jusqu'à 30% pour les nœuds les plus éloignés, ENG-TOT donne le pire PDR.

Pour RX=60% dans la topologie grille, on observe la même tendance pour toutes les métriques, mais avec des valeurs PDR plus faibles que pour RX=100%. Nous pouvons remarquer un renversement entre les métriques ETX et R. La métrique R atteint un PDR inférieur à celui de la métrique ETX, mais donne des résultats légèrement meilleurs que

## 5.5. ÉVALUATION DES MÉTRIQUES

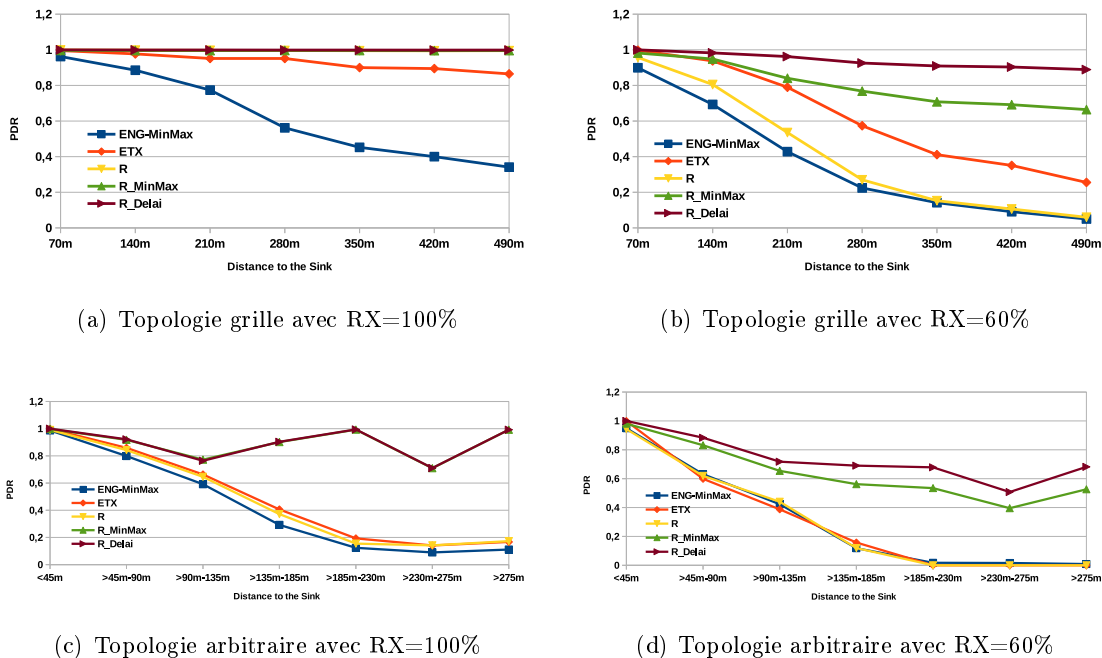


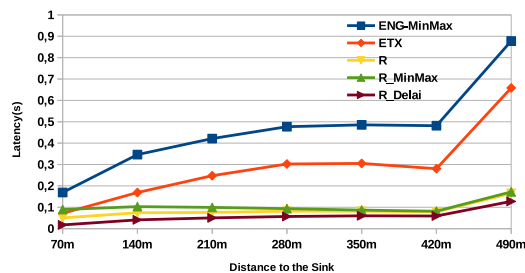
FIGURE 5.11 – Taux de livraison des paquets.

les autres métriques (excepté ENG-TOT). La métrique ETX a le PDR le plus élevé de 100% près de la racine et de 25% pour les nœuds les plus éloignés. Dans le cas de la topologie arbitraire (Figures 5.11(c) et 5.11(d) ), le taux de livraison des paquets diminue plus rapidement à cause d'une densité moyenne faible. Toutes les métriques atteignent pratiquement les mêmes valeurs de PDR. Pour RX=100%, les valeurs s'étalent de 100% près de la racine à 15% pour les nœuds les plus éloignés, alors que pour RX=60%, les valeurs de PDR sont plus nettes avec des valeurs entre 100% et 15% pour la première moitié du réseau (jusqu'à 135 m) et aucun paquet de données envoyé par un capteur dans la seconde moitié du réseau (distance supérieure à 135 m) n'a été reçu par le puits. Nous pouvons remarquer que la métrique ENG-TOT réalise le plus faible PDR dans tous les cas.

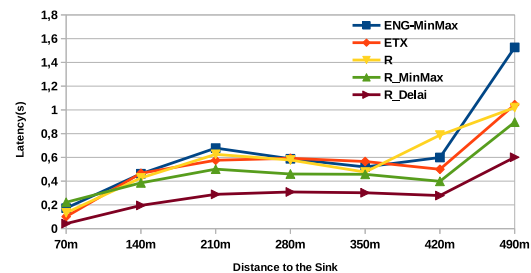
**Délai de bout en bout :** le délai de bout en bout (latence) augmente naturellement en fonction de la distance vers le puits (Figure 5.12). Dans la topologie grille avec RX=100% (Figure 5.12(a)), le meilleur délai de bout en bout sur toutes les métriques est de nouveau obtenu par la métrique R avec un délai inférieur à 200ms pour les nœuds les plus éloignés

## 5.5. ÉVALUATION DES MÉTRIQUES

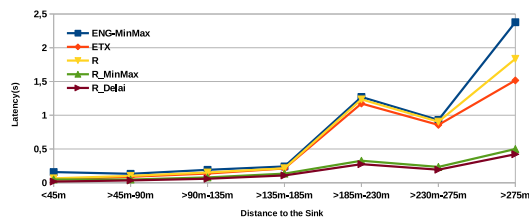
et 100ms pour les autres. Les résultats obtenus sont en relation avec les valeurs de PDR décrites précédemment. Il est jusqu'à 5 fois meilleur que le pire délai atteint par la métrique ENG\_MinMax. La métrique R est ensuite suivie par la métrique ETX qui prend uniquement en compte la qualité du lien de communication. La métrique ETX a un délai 3 fois supérieur à la métrique R. Pour les métriques cherchant à optimiser l'énergie, la métrique ENG\_MinMax donne les résultats les plus médiocres avec un délai entre 300 et 500 ms pour la plupart des nœuds sauf pour les nœuds les plus éloignés dont les paquets sont transmis avec un délai de 900ms. La métrique ELT est située entre les métriques ENG-TOT et ENG\_MinMax. Dans la même topologie avec RX=60% (Figure 5.12(b)), la métrique ENG-TOT atteint le délai le plus faible en raison de la faible quantité de paquets à acheminer contrairement aux autres métriques. La métrique R perd son efficacité par rapport au cas où RX=100%, puisque le même poids est fourni pour la qualité de la liaison et l'énergie résiduelle. Pourtant, elle reste proche de la métrique ETX.



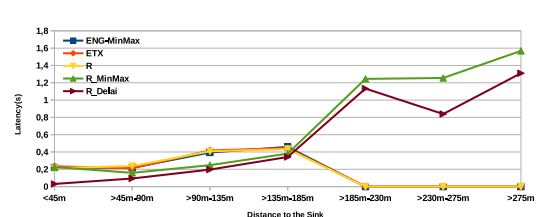
(a) Topologie grille avec RX=100%



(b) Topologie grille avec RX=60%



(c) Topologie arbitraire avec RX=100%



(d) Topologie arbitraire avec RX=60%

FIGURE 5.12 – Le délai de bout en bout.

Pour la topologie arbitraire avec RX=100% (Figure 5.12(c)), on peut voir qu'une légère augmentation est suivie d'une forte augmentation corrélée au nombre élevé de retrans-

missions dus au faible PDR comme indiqué dans la Figure 5.11(c). Toutes les métriques (excepté ENG-TOT) donnent des résultats similaires. Avec un RX inférieure de 60%, la latence augmente dans la première moitié du réseau pour les cinq métriques jusqu'à atteindre un maximum de 350ms pour ELT et 450ms pour d'autres métriques (excepté ENG-TOT) pour les nœuds à 135m. Ensuite, le délai est nul dans la seconde moitié du réseau parce que tous les paquets sont perdus, à ce stade la fiabilité de RPL est compromise. On peut noter que, ENG-TOT donne le plus faible délai dans tous les cas en raison du taux de perte de paquets important. Enfin, ELT atteint dans ce cas la latence la plus faible parmi toutes les métriques dans la première moitié du réseau.

**Messages de contrôle :** la surcharge du réseau est exprimée comme la quantité de messages de contrôle envoyés par RPL. On peut noter qu'elle augmente lentement en fonction de la distance vers le puits (Figure 5.13). Dans tous les cas, nous observons que le réseau est surchargé avec la métrique ENG-TOT. La quantité élevée des messages de contrôle pour la métrique ENG-TOT est liée au faible PDR obtenu par cette métrique. Cette grande quantité de messages de contrôle échangés résulte d'instabilités des routes dans le DODAG. La surcharge du réseau causée par les métriques ELT et ENG\_MinMax est relativement stable.

Nous notons que, outre la métrique ENG-TOT, le nombre de messages de contrôle est négligeable pour toutes les autres métriques dans les différents scénarii de simulation par rapport au trafic de données, et plus le DODAG se stabilise plus le nombre de messages de contrôle diminue significativement.

## 5.6 Conclusion

Minimiser la consommation énergétique, garantir des communications avec une faible latence et fournir un taux de livraison élevé à la fois est particulièrement difficile avec les contraintes associées aux réseaux de capteurs. Cela nécessite de concevoir des stratégies de routage tenant compte de plusieurs critères. En conséquence, RPL a été spécifié par le groupe de travail IETF ROLL comme un protocole de routage à vecteur de distance pour les LLNs. Un graphe orienté acyclique (DODAG) est construit en optimisant une fonction



## 5.6. CONCLUSION

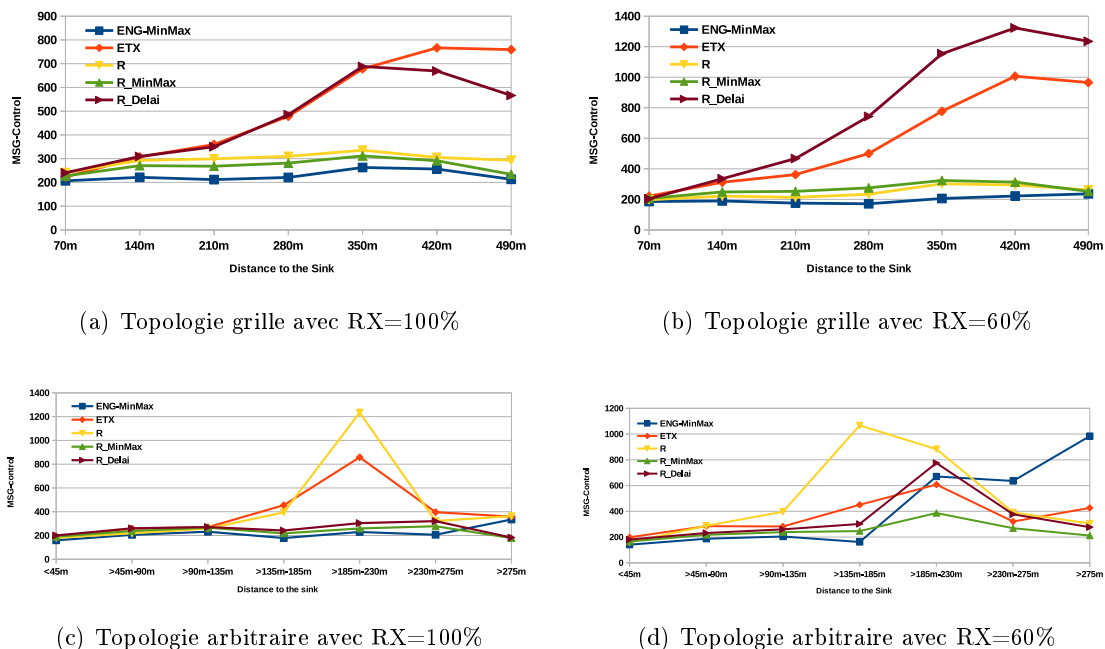


FIGURE 5.13 – Les Messages de contrôle.

objectif qui prend en compte des métriques et des contraintes pour la sélection des routes vers le puits.

Dans ce chapitre, nous avons présenté une étude comparative de plusieurs métriques représentatives des métriques proposées pour améliorer RPL en vue de prolonger la durée de vie du réseau. La métrique par défaut ETX tient compte du nombre de retransmissions et permet de garantir une bonne fiabilité des communications. Cependant, sans surprise ETX atteint une faible durée de vie du réseau car celle-ci ne tient pas compte des niveaux de batterie des nœuds. Certaines métriques RPL atteignent une meilleure durée de vie du réseau, malgré des délais de bout en bout vers le puits pouvant être importants. Les métriques tenant compte de l'énergie, comme la métrique ENG-TOT, présentent des délais de bout en bout qui peuvent être faibles mais cela vient du faible PDR atteint dans le DODAG. En outre, il semble que les métriques bicritères telles que la métrique R montrent de meilleures performances en termes de durée de vie du réseau et de délais de bout en bout dans le cas RX=100%. Cela s'explique par le fait que les paramètres optimisés par cette métrique ne sont pas orthogonaux. Nos résultats montrent qu'il est nécessaire de

## 5.6. CONCLUSION

---

concevoir des métriques multi-critères qui tiennent compte à la fois de la qualité des liens de communication et de l'énergie résiduelle des nœuds pour améliorer la durée de vie dans les réseaux de capteurs. Cela fait l'objet du chapitre suivant dans lequel nous nous intéressons à l'optimisation de plusieurs métriques primaires pour répondre aux besoins applicatifs associés aux réseaux de capteurs.

## Chapitre 6

# Nouvelles métriques de routage pour RPL

### 6.1 Introduction

Les contraintes et limitations liées aux réseaux de capteurs sans fil, comme de faibles débits sur les liens de communication radio ou nœuds capteurs alimentés sur batterie à faible capacité, nous oblige à concevoir des protocoles qui soient à la fois économes en énergie et puissent assurer une certaine qualité de service (délai de transmission, taux de livraison de données ou stabilité du réseau) nécessaire au bon fonctionnement des applications utilisant ce type de réseaux.

Pour faire face à ces exigences, dans ce chapitre, nous faisons une proposition de nouvelles métriques de routage pour améliorer les performances du protocole de routage RPL. Pour cela nous nous focalisons sur trois critères pour estimer ces performances dans RPL :

1. Durée de vie du réseau,
2. Taux de perte de paquets,
3. Délai de bout en bout pour la transmission des données.

Dans ce chapitre nous proposons deux nouvelles métriques pour RPL, la première métrique vise à maximiser la durée de vie du réseau tout en limitant le taux de perte de paquets. La deuxième métrique vient compléter les performances atteintes par notre première proposition en réduisant également le délai de transmission de bout en bout des paquets de chaque nœud source jusqu'au puits dans le DODAG.

Nous avons mené un nombre important et varié de simulations afin d'évaluer les performances atteintes par les métriques proposées. Nous avons comparé ces résultats à ceux obtenus pour les métriques présentées au chapitre précédent. Cette comparaison démontre des améliorations significatives en termes de consommation d'énergie, de taux de livraison de paquets (PDR), et de délai de transmission de bout en bout atteintes par les métriques proposées en regard des métriques existantes.

Ce chapitre est composé de deux sections, tout d'abord dans la première section nous présentons notre première proposition de métrique puis la seconde section décrit une évolution de cette première métrique pour prendre en compte le délai de bout en bout. Pour chaque section, nous commençons par le contexte général de la métrique proposée, puis nous présentons la modélisation de la métrique de routage appliquée au protocole RPL et enfin nous établissons un comparatif suivant deux modèles de transmission, aucune perte de paquet (avec  $RX=100\%$ ), et 40% de perte de paquets (avec  $RX=60\%$ ).

## 6.2 Première proposition de métrique

### 6.2.1 Contexte et motivation

À ce niveau, il faut retenir qu'une fonction objectif dans RPL permet à chaque nœud du réseau de sélectionner le meilleur chemin vers la racine appartenant au DODAG construit. Cette fonction objectif définit le coût du meilleur chemin pour chaque nœud dans le DODAG en se basant sur une métrique indiquée en entrée. La métrique considérée dans la fonction objectif prend en charge un ou plusieurs paramètres en lien avec les contraintes à prendre en compte lors du choix du parent préféré pour la sélection du meilleur chemin.

En premier lieu, nous proposons une métrique de routage pour RPL, appelée  $R\_MinMax$ , qui prend en compte pour la sélection du chemin de meilleur coût vers la racine du DODAG :

1. le niveau d'énergie des nœuds du réseau et,
2. la qualité des liens de communication.

L'objectif de notre métrique  $R\_MinMax$  est d'améliorer la durée de vie des réseaux

de capteurs et d'assurer une meilleure qualité d'acheminement des données (réduction du taux de perte) lors des communications sans fil. Par conséquent, nous proposons une métrique qui s'appuie sur deux paramètres : tout d'abord sur la valeur ETX des liens et l'énergie résiduelle minimale des nœuds capteurs sur les chemins du DODAG. ETX permet d'indiquer la qualité du lien de communication en fournissant une estimation du nombre de retransmissions nécessaires d'un paquet de données afin qu'il soit reçu à l'autre extrémité du lien. La prise en compte de ce paramètre ETX dans le choix du parent, permet d'une part de sélectionner des liens nécessitant moins de retransmissions d'un paquet de données mais aussi de réduire l'énergie consommée par chaque nœud grâce à un nombre de transmissions plus faible. De plus, nous considérons l'énergie résiduelle minimale sur chaque chemin afin de tenir compte de la plus faible énergie résiduelle sur le chemin (également nœud bloquant) lors de la sélection du parent préféré d'un nœud. En effet, un nœud bloquant pour un chemin est le nœud ayant le plus faible niveau d'énergie résiduelle. Il peut être la cause d'instabilité de routage et de pertes de paquets, conduisant à une hausse de la consommation des batteries suite aux retransmissions, s'il n'a plus suffisamment de batterie. Pour cela, la plus faible énergie résiduelle est propagée le long de chaque chemin par les nœuds ce qui permet lors du choix du parent préféré d'en avoir connaissance. Sachant que ce type de réseau est très contraint en ressources énergétiques, la métrique  $R\_MinMax$  que nous proposons cherche à augmenter la durée de vie du réseau en évitant de choisir pour chaque nœud :

1. un chemin vers la racine dont le nœud bloquant possède peu d'énergie résiduelle,
2. un lien de communication vers le parent préféré avec un nombre important de retransmissions.

De là est venu l'intérêt d'étudier la combinaison des deux paramètres : l'énergie résiduelle minimale des nœuds sur les chemins du DODAG et la qualité des liens de communication pas à pas représentée par le paramètre ETX.

### 6.2.2 Modélisation

Comme le montre le chapitre précédent, la métrique  $R$  proposée par Chang et al. [Chang et al. 2013] pour RPL prend déjà en compte l'énergie résiduelle et la qualité des liens

de communication dans le choix du parent. En revanche lors de la prise en compte de l'énergie résiduelle pour la sélection du meilleur chemin le nœud capteur ne considère que l'énergie résiduelle de son parent à un saut, il en va de même pour la qualité des liens de communication. Il n'y a donc aucune visibilité sur le niveau de l'énergie résiduelle ou le nombre de retransmissions nécessaires des nœuds se trouvant le long des chemins dans le DODAG en direction de la racine. Par conséquent, la métrique  $R$  atteint ses limites dès lors qu'il y a un grand nombre de pertes de paquets à différents niveaux sur un chemin vers la racine dans le DODAG. Comme montré dans la Figure 5.10, 5.11 du chapitre précédent dans le cas de la topologie grille avec  $RX=60\%$ , ou nous observons la dégradation des performances de la métrique  $R$  en terme de consommation d'énergie et de taux de livraison des paquets. Cela s'explique par le fait que les paramètres optimisés par cette métrique sont restreints aux informations locales uniquement pour la sélection du chemin.

Si on considère l'exemple de réseau donné à la figure 6.2.2, le nœud F possède plusieurs parents candidats et fera un choix différent pour son parent préféré en fonction de la métrique considérée dans RPL parmi les nœuds C, D ou E. La Figure 6.2.2(c) illustre la construction du DODAG en fonction de la métrique  $R$ . Le nœud F choisit C comme parent préféré puisque c'est le voisin ayant l'énergie résiduelle la plus importante à son niveau et également la valeur la plus faible pour le paramètre ETX pour l'envoi des données de F vers C. Cependant, on peut remarquer que le nœud A, plus haut sur ce chemin, possède un faible niveau d'énergie résiduelle. Le choix de C comme parent de F conduira à un épuisement rapide de la batterie de A, et à des instabilités sur le routage des paquets pour C et F puisque le chemin ne sera plus disponible.

Pour faire face à ce problème, la métrique  $R\_MinMax$  que nous proposons reprend la définition de la fonction objectif introduite par la métrique  $R$  [Chang et al. 2013]. Cependant, nous adaptons la fonction objectif afin de tenir compte de l'énergie résiduelle minimale associée à chaque chemin vers la racine du DODAG, de la même manière que la métrique  $ENG\_MinMax$  [Kamgueu et al. 2013]. La Figure 6.2.2(b) illustre la construction du DODAG en fonction de la métrique  $ENG\_MinMax$ . Le nœud F choisit E comme parent préféré (indiqué par une flèche rouge) puisqu'il diffuse la plus grande énergie résiduelle minimale, à savoir une énergie de 200 pour E en comparaison d'une valeur de 95 pour C et

## 6.2. PREMIÈRE PROPOSITION DE MÉTRIQUE

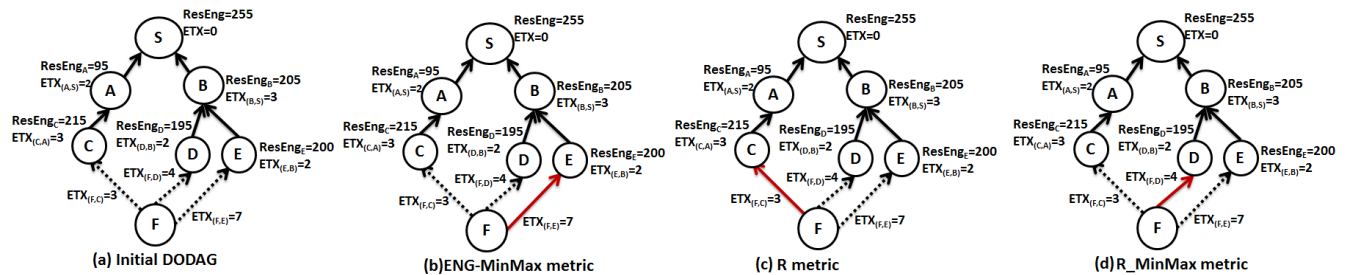


FIGURE 6.1 – Construction du DODAG RPL avec plusieurs métriques à partir du nœud F : (a) exemple de réseau avec le DODAG construit avant l’ajout de F, (b) choix du parent préféré avec la métrique ENG\_MinMax, (c) choix du parent préféré avec la métrique  $R$ , (d) choix du parent préféré avec la métrique  $R\_MinMax$  proposée.

195 pour D. Toute fois, ce choix aura un impact sur le coût en énergie consommée par le nœud F lors des transmissions des données car le lien vers F possède une valeur élevée pour le paramètre ETX. Et les performances de cette même métrique se trouvent rapidement dégradée en présence de pertes sur les liens de communication, car ce paramètre n’est pas pris en compte par cette métrique pour la sélection du parent préféré d’un nœud.

Pour synthétiser la discussion ci-dessus, notre contribution majeure avec la métrique  $R\_MinMax$  que nous proposons est d’une part la prise en compte de l’énergie résiduelle des nœuds bloquants de chaque chemin (c’est-à-dire à plusieurs sauts), grâce à la propagation de l’énergie minimale sur le chemin, en comparaison avec la métrique  $R$  proposée par Chang et al. [Chang et al. 2013]. Et d’autre part, la considération de la qualité des liens de communication par rapport à la métrique ENG\_MinMax proposée par Kamgueu et al. [Kamgueu et al. 2013]. La Figure 6.2.2(d) illustre la construction du DODAG réalisée avec la métrique  $R\_MinMax$ . D possède le meilleur équilibre entre qualité du lien de communication vers le parent et l’énergie résiduelle minimale sur le chemin, soit une valeur de 4 pour ETX et 195 pour l’énergie résiduelle minimale comparée aux valeurs de 3 et 95 pour C et 7 et 200 pour E respectivement. En mettant au même plan la fiabilité sur les liens de communication et la minimisation du niveau de vie du réseau (pondération avec  $\alpha = 0,5$ ), le nœud F choisit donc D comme parent préféré car il permet d’atteindre le meilleur résultat en combinant les paramètres sur la qualité du lien de communication vers le parent et l’énergie résiduelle minimale sur le chemin du parent.

Par conséquent, nous donnons dans l'équation 6.1 ci-dessous la définition de la fonction objectif associée à la métrique  $R\_MinMax$  que nous proposons pour chaque nœud  $i$  du réseau :

$$R\_MinMax_i = \min_{j \in N_i} \left\{ \alpha \frac{ETX_{(i,j)}}{Max\_ETX} + (1 - \alpha) \times \left( 1 - \frac{PW_j}{MaxEng} \right) \right\}, \quad (6.1)$$

Où  $N_i$  est l'ensemble des voisins de  $i$  et  $PW_j$  est égal à l'énergie résiduelle minimale sur le chemin vers la racine du DODAG à partir du parent candidat  $j \in N_i$ . Le parent préféré de chaque nœud  $i$  est le parent candidat qui possède le plus petit coût retourné par la fonction  $R\_MinMax_i$ , c'est-à-dire le parent ayant le meilleur taux entre l'énergie résiduelle minimale sur son chemin et le taux du nombre de retransmissions vers ce même parent.

### 6.2.3 Analyse des résultats de simulation

En se basant sur le même cadre de simulation (paramètres de configuration et topologies identiques) qu'au chapitre 5, nous présentons dans ce qui suit les résultats de simulation obtenus pour la métrique  $R\_MinMax$ . Afin de comparer plus aisément les performances atteintes, nous remettons les résultats présentés au chapitre précédent associés aux métriques  $ETX$ ,  $ENG\_MinMax$  et  $R$ . À noter que nous avons choisi de retirer les courbes des métriques  $ENG\_TOT$  et  $ELT$  pour ne pas alourdir la lecture. En effet, comme présenté dans le comparatif réalisé au chapitre précédent les performances atteintes par ces deux métriques sont inférieures aux performances obtenues avec les métriques  $ENG\_MinMax$  et  $R$ .

**Consommation énergétique :** Nous considérons d'abord l'énergie consommée par chaque métrique de routage (Figure 6.2). Dans notre simulation, les nœuds ont la même charge de batterie initiale de 853 mAh et le paramètre de pondération  $\alpha$  est égal à 0,5 pour la métrique  $R$  et  $R\_MinMax$ . Dans la figure 6.2, le pourcentage de temps pendant lequel la radio est activée reflète principalement l'énergie consommée par un nœud. Dans la topologie grille avec  $RX=100\%$  (Figure 6.2(a)), la consommation énergétique augmente puis décroît en fonction de la distance à la racine pour les quatre métriques. Cette augmentation est



due au fait que la racine du DODAG représente un goulot d'étranglement. ETX est la métrique qui a la plus forte consommation en énergie autour de la racine, suivie par les autres métriques  $ENG\_MinMax$ ,  $R$  et  $R\_MinMax$ . Cependant, pour les nœuds les plus éloignés, l'énergie consommée par la métrique ETX est inférieure à la métrique  $ENG\_MinMax$  en raison du faible taux de retransmissions (comme le montre la figure 6.3(a)). Les métriques  $R$  et  $R\_MinMax$  montrent des résultats proches entre elles avec une consommation en énergie jusqu'à 2 à 3 fois inférieure par rapport aux métriques ETX et  $ENG\_MinMax$ . La radio est activée au plus 1% du temps de simulation. Par conséquent nous pouvons conclure que dans ce cas de simulation la métrique  $R$  et  $R\_MinMax$  équilibrent mieux l'engorgement du trafic des différents nœuds se trouvant autour de la racine.

Dans la même topologie avec  $RX=60\%$  (Figure 6.2(b)), les métriques se comportent de la même manière que pour le cas  $RX=100\%$ , excepté pour la métrique  $R$ . Nous observons une légère augmentation de la consommation d'énergie due à l'augmentation du nombre de retransmissions (40% de pertes). On observe que la métrique  $R$  perd de son efficacité, et obtient des résultats similaires aux métriques ETX et  $ENG\_MinMax$ . Le module radio des nœuds est actif plus souvent également avec la métrique  $R\_MinMax$ , mais cela est bien inférieur par rapport aux autres métriques. En effet, le module radio actif au plus 1.9% du temps de simulation contre 2.5% pour  $ENG\_MinMax$  et  $R$ , et 3% pour ETX. La métrique  $R\_MinMax$  permet une réduction de la consommation énergétique pour un nœud d'au plus 24% par rapport aux métriques  $ENG\_MinMax$  et  $R$ , et 36% par rapport à la métrique ETX. Contrairement à la métrique  $R$ , on peut observer que la métrique  $R\_MinMax$  que nous proposons parvient à maintenir son efficacité énergétique malgré un taux de pertes de paquet important dans le réseau.

Avec une topologie arbitraire (Figure 6.2(c), 6.2(d)) nous observons le même comportement lorsque  $RX=100\%$  ou  $RX=60\%$ . La consommation énergétique est un peu plus élevée quand  $RX=60\%$  à cause des retransmissions de paquets. Elle diminue puis augmente en fonction de la distance à la racine. La consommation d'énergie est plus élevée autour de la racine à cause d'un phénomène de goulot d'étranglement. La consommation diminue ensuite pour les nœuds jusqu'à une distance de 135m de la racine, puis elle augmente en raison des retransmissions dues aux pertes de paquets (comme indiqué par le PDR dans les

## 6.2. PREMIÈRE PROPOSITION DE MÉTRIQUE

---

figures 6.3(c), 6.3(d) présentées ci-après). Là encore nous pouvons constater que la métrique  $R\_MinMax$  est plus efficace par rapport aux restes des métriques puisqu'elle parvient à réduire la consommation énergétique sur l'ensemble des nœuds du réseau en choisissant comme parent le voisin qui revoit la plus importante énergie minimale sur son chemin vers la racine et avec qui le lien de communication possède un nombre de retransmissions faible.

Pour les deux topologies, la métrique proposée  $R\_MinMax$  réussie à garantir un routage équilibré en énergie, et de meilleurs résultats de simulation par rapport aux autres métriques, puisqu'elle tient compte à la fois de la qualité du lien de communication et de l'énergie résiduelle minimale jusqu'à la racine dans la décision du choix du parent.

Tab. 6.1 présente pour chaque métrique le pourcentage d'énergie maximale consommée par un nœud du réseau après 5 heures de simulation et une extrapolation de la consommation énergétique dans le réseau exprimée en jours jusqu'à la panne du premier nœud. Il est à noter que la courte durée de vie du réseau est liée à la faible charge initiale de la batterie (de 853 mAh au lieu de 2000 mAh dans la plate-forme Tmote sky mote).

Nous observons que les métriques  $R$  et  $R\_MinMax$  surpassent toutes les autres métriques sur la topologie grille avec  $RX=100\%$ . Elles atteignent une durée de vie de 133 jours, environ 5 à 7 fois mieux que le reste des métriques  $ETX$  et  $ENG\_MinMax$ . Avec  $RX=60\%$  sur la topologie grille, nous observons que la métrique  $R\_MinMax$  montre une fois de plus sa robustesse par rapport à  $R$ ,  $ENG\_MinMax$  et  $ETX$ . Elle atteint une durée de vie de 74 jours, environ 3 à 5 fois mieux que les métriques  $R$ ,  $ENG\_MinMax$  et  $ETX$ . Sur la topologie arbitraire avec  $RX=100\%$  ou  $RX=60\%$ , les métriques  $R$ ,  $ENG\_MinMax$  et  $ETX$  donnent des durées de vie similaires de 10 à 13 jours, à l'exception de notre métrique  $R\_MinMax$ . En effet, celle-ci atteint une durée de vie de 82 jours avec  $RX=100\%$  et 52 jours avec  $RX=60\%$ . L'amélioration est importante par rapport aux autres métriques, soit une durée de vie de 6 à 7 fois supérieure lorsque  $RX=100\%$  et environ 5 fois pour  $RX=60\%$  par rapport aux autres métriques.

**Taux de livraison des paquets :** La figure 6.3 présente les taux de livraison de paquets associés aux quatre métriques considérées dans l'évaluation. Comme attendu, le taux de livraison de paquets (PDR) diminue en fonction de la distance à la racine à la fois dans les

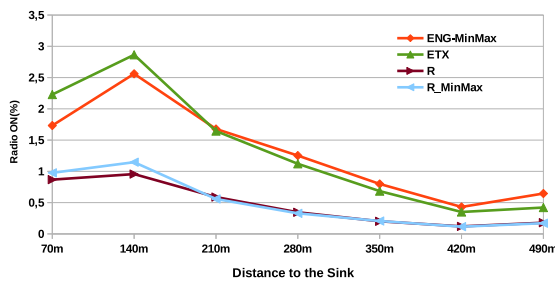
## 6.2. PREMIÈRE PROPOSITION DE MÉTRIQUE

TABLE 6.1 – Durée de vie du réseau -2-

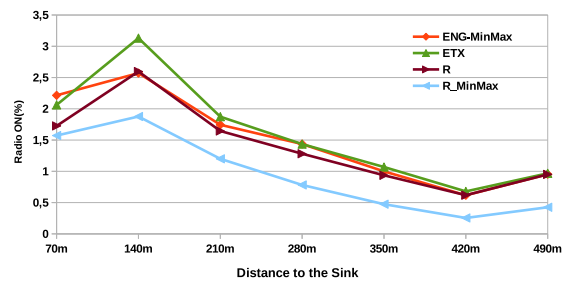
Topologie grille				
	ENG_MinMax	ETX	R	R_MinMax
RX=	1.12 %	1.15 %	0.157 %	0.158 %
100 %	124698 mj	166098 mj	17359 mj	17535 mj
	19 jours	18 jours	133 jours	132 jours
RX=	0.81 %	1.37 %	0.93 %	0.28 %
60 %	88630 mj	151250 mj	103099 mj	30996 mj
	26 jours	15 jours	23 jours	74 jours

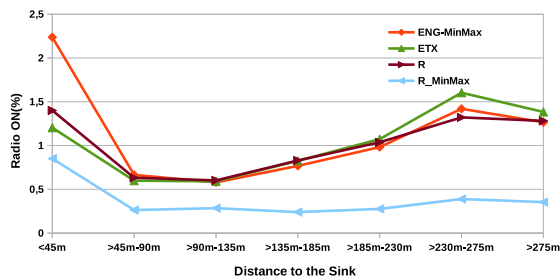
Topologie arbitraire				
	ENG_MinMax	ETX	R	R_MinMax
RX=	1.77 %	1.71 %	1.78 %	0.25 %
100 %	196499 mj	188999 mj	196700 mj	28001 mj
	11 jours	13 jours	12 jours	82 jours
RX=	1.96 %	1.98 %	1.91 %	0.41 %
60 %	219994 mj	220104 mj	211666 mj	44372 mj
	10 jours	10 jours	11 jours	52 jours



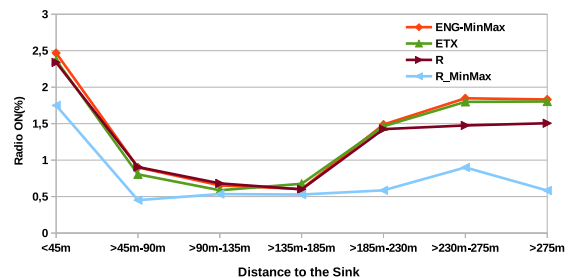
(a) Topologie grille avec RX=100%



(b) Topologie grille avec RX=60%



(c) Topologie arbitraire avec RX=100%



(d) Topologie arbitraire avec RX=60%

FIGURE 6.2 – Consommation énergétique.

topologies grille et arbitraire. Globalement la métrique ETX atteint de bonnes valeurs pour le PDR car elle prend en compte la qualité des liens de communication dans la sélection du meilleur parent. Malgré cela, il est important de noter que dans une topologie grille avec RX=100% (Figure 6.3(a)), les meilleurs résultats sont donnés par les métriques  $R$  et  $R\_MinMax$  avec un PDR proche de 100% pour chaque nœud. Ces meilleurs résultats s'expliquent par le fait que les métriques  $R$  et  $R\_MinMax$  tiennent compte à la fois de la qualité du lien de communication (à un saut) mais aussi de l'énergie résiduelle minimale jusqu'à la racine pour la sélection du parent, contrairement à la métrique ETX qui ne tient pas compte de l'état des batteries. Une quantité non négligeable de paquets est donc perdue durant la reconfiguration du DODAG suite à l'épuisement des batteries des nœuds bloquants sur les chemins du DODAG. A l'inverse, la métrique ENG\_MinMax obtient un PDR inférieur à 100% près de la racine et un PDR à 30% pour les nœuds les plus éloignés, car celle-ci ne prend pas en compte la qualité des liens de communication pour la construction et le maintien du DODAG.

Pour RX=60% sur la topologie de grille (figure 6.3(b)), nous observons la même tendance pour toutes les métriques, mais avec des valeurs PDR bien plus faibles que pour le cas où RX=100%. Nous pouvons remarquer une inversion entre les métriques ETX et  $R$ . La métrique  $R$  atteint un PDR inférieur à la métrique ETX, mais elle atteint de meilleures performances que la métrique ENG\_MinMax. La métrique  $R\_MinMax$  que nous proposons obtient des valeurs de PDR les plus élevées, de 100% près de la racine jusqu'à 65% pour les nœuds les plus éloignés. Avec une topologie arbitraire (Figures 6.3(c) et 6.3(d)), le PDR diminue rapidement pour toutes les métriques sauf pour la métrique  $R\_MinMax$ . Les métriques  $R$ , ENG\_MinMax et ETX atteignent pratiquement les mêmes valeurs pour le PDR, de 100% près de la racine jusqu'à 15% pour les nœuds les plus éloignés dans le cas où RX=100%. Lorsque RX=60%, les valeurs pour le PDR associées aux métriques  $R$ , ENG\_MinMax et ETX sont comprises entre 100 et 15% pour la première moitié du réseau (jusqu'à 135m), puis plus aucun paquet de données n'est reçu par le puits dans la seconde moitié du réseau (distance supérieure à 135 m). En conséquence, toute une partie des données captées par les nœuds de la seconde moitié du réseau ne sont pas acheminées au puits ce qui peut être critique d'un point de vu applicatif. Il est important d'observer

## 6.2. PREMIÈRE PROPOSITION DE MÉTRIQUE

que contrairement aux autres métriques, notre métrique  $R\_MinMax$  arrive à acheminer les données de l'ensemble des nœuds du réseau. Des valeurs de PDR allant de 100% près de la racine à 60% pour la première moitié du réseau sont obtenues. Pour les nœuds situés dans la seconde moitié du réseau, 60% à 40% des paquets de données sont acheminés là où aucun paquet n'est acheminé en utilisant les autres métriques au delà de 185m de la racine. La métrique  $R\_MinMax$  atteint les meilleurs résultats pour les deux topologies en comparaison des autres métriques quelque soit le taux de réception RX considéré.

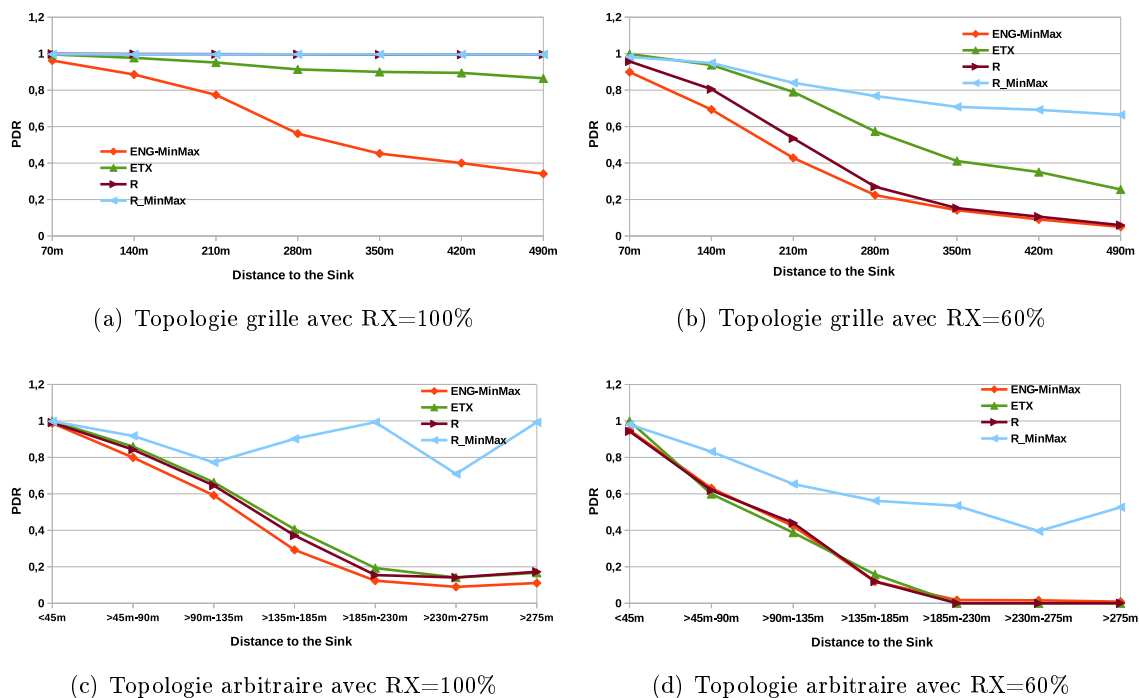


FIGURE 6.3 – Taux de livraison des paquets.

**Délai de bout en bout :** Le délai de bout en bout augmente en fonction de la distance à la racine (Figure 6.4). Dans la topologie grille avec RX=100% (Figure 6.4(a)), sur l'ensemble des métriques le meilleur délai de bout en bout est de nouveau obtenu par la métrique  $R\_MinMax$ , puis par la métrique  $R$  avec des délais assez similaires inférieur à 200ms pour les nœuds les plus éloignés (>420m) et proche de 100ms pour les autres nœuds. Les résultats obtenus sont en accord avec les valeurs de PDR décrites précédemment. Ce résultat est jusqu'à 5 fois meilleur que le pire délai atteint par la métrique  $ENG\_MinMax$ ,

## 6.2. PREMIÈRE PROPOSITION DE MÉTRIQUE

---

avec un délai entre 300ms et 500ms pour la plupart des nœuds, sauf pour les nœuds les plus éloignés dont les paquets sont transmis avec un délai de 900ms. La métrique ETX donne de meilleurs résultats que ENG\_MinMax, mais le délai de bout en bout est 3 fois plus important pour la majorité des nœuds du réseau. Dans la même topologie avec RX=60% (Figure 6.4(b)), *R\_MinMax* obtient de meilleurs résultats que toutes les autres métriques. Ce qui est à nouveau en accord avec les valeurs de PDR. En effet, plus le risque de perte de paquets diminue, plus le paquet a de meilleures chances d'être relayé dans les meilleurs délais. On peut cependant remarquer que les délais obtenus par les diverses métriques sont assez proches, sauf lorsque l'on considère les nœuds situés à une distance supérieure à 350m de la racine.

Pour la topologie arbitraire avec RX=100% (Figure 6.4(c)), on peut observer une forte augmentation du délai pour les nœuds situés à une distance entre 135m et 185m. Cela est corrélé au nombre élevé de retransmissions due à un PDR faible représenté dans la figure 6.3(c). Toutes les métriques donnent des résultats similaires sauf la métrique *R\_MinMax* qui atteint de meilleurs résultats dans la deuxième moitié du réseau (distance supérieure à 135m). En effet, le délai de bout en bout obtenu est 3 fois inférieur comparé à la métrique ETX pour les nœuds au-delà de 185m de la racine. Avec RX=60 %, la latence augmente dans la première moitié du réseau pour toutes les métriques jusqu'à atteindre un maximum de 450ms pour les nœuds à 135m de la racine. Le délai nul dans la seconde moitié du réseau représente les paquets qui ne sont jamais reçus par la racine ; à ce stade, la fiabilité de RPL est mise en cause, excepté dans le cas de la métrique *R\_MinMax*. En effet, dans la seconde moitié du réseau *R\_MinMax* est la seule métrique assurant une certaine fiabilité, les paquets de tous les nœuds du réseau sont transmis mais avec un délai de bout en bout important.

**Messages de contrôle :** La surcharge du réseau est exprimée par la quantité de messages de contrôle envoyés par le protocole RPL. Cette charge augmente lentement en fonction de la distance vers le nœud racine (comme illustré par la Figure 6.5). Dans tous les cas, nous observons que la métrique *R\_MinMax* est relativement stable. C'est l'une des seules métriques avec ENG\_MinMax à envoyer le plus faible nombre de messages de contrôle.

## 6.3. DEUXIÈME PROPOSITION DE MÉTRIQUE

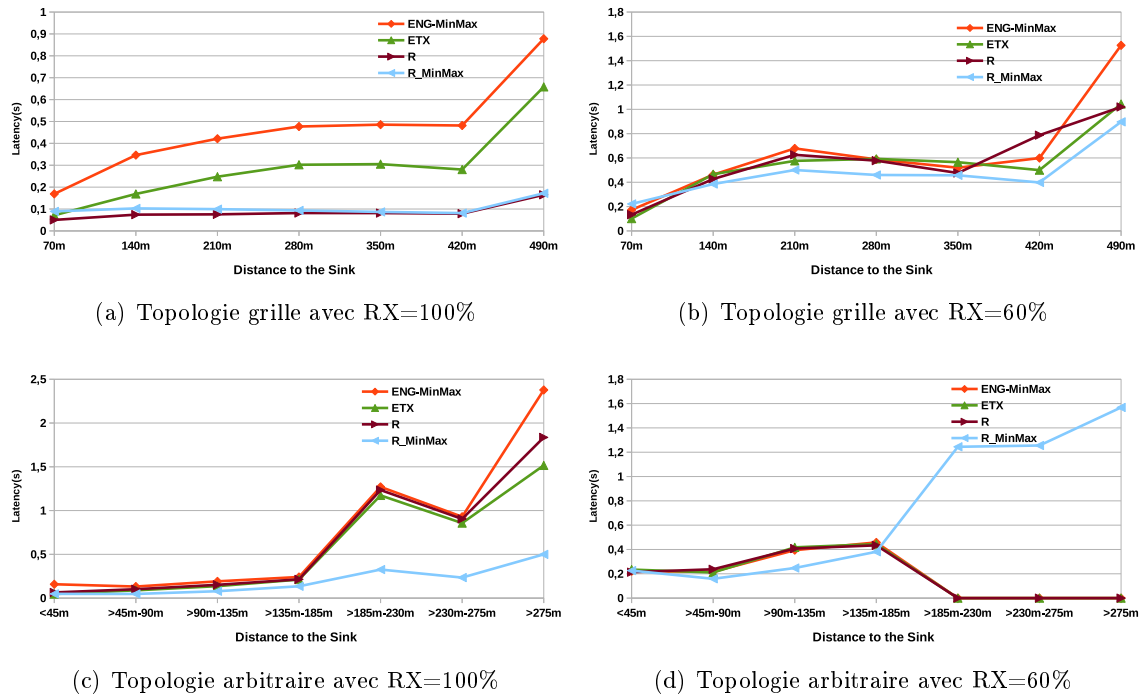


FIGURE 6.4 – Le délai de bout en bout.

On peut observer que dans le cas de la topologie arbitraire avec RX=60% un nombre important de messages de contrôle est envoyé quelque soit la métrique considéré, excepté pour la métrique  $R\_MinMax$  qui limite au double l'augmentation du nombre de messages échangés pour l'ensemble des nœuds du réseau. Ceci est en lien avec les valeurs obtenues pour le PDR par la métrique  $R\_MinMax$ .

## 6.3 Deuxième proposition de métrique

### 6.3.1 Contexte et Motivation

La diversité des domaines d'application des réseaux de capteurs sans fil voient émerger de nouveaux critères de performance en plus de la durée de vie du réseau, comme la fiabilité des communications ou encore le délai de transmission de bout en bout. Pour atteindre ces exigences de QoS des applications WSN, beaucoup de travaux ont été consacrés ces dernières années à la conception de métriques de routage multi-critères pour RPL. Dans la première partie de ce chapitre, nous avons étudié l'optimisation de la durée de vie du

### 6.3. DEUXIÈME PROPOSITION DE MÉTRIQUE

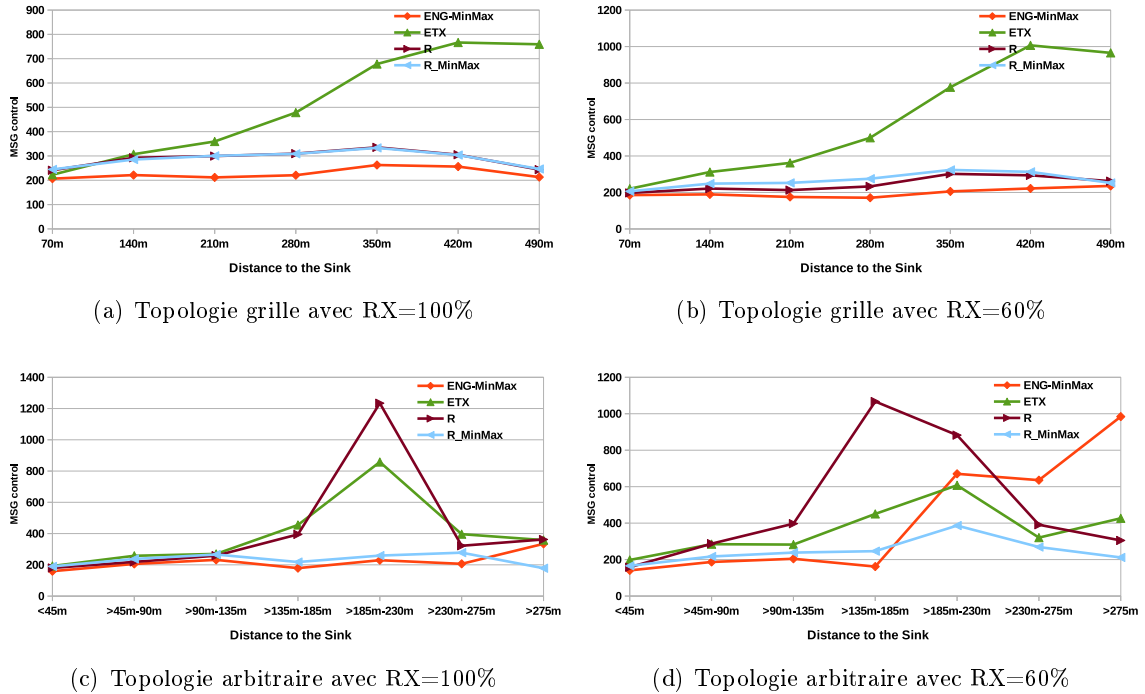


FIGURE 6.5 – Les Messages de contrôle.

réseau et la fiabilité de transmission des données. Les résultats de simulation obtenus pour la métrique  $R\_MinMax$  montrent un gain significatif pour ces deux critères, par rapport à certaines métriques de routage RPL existantes. Cependant, en terme de délai de transmission de bout en bout les résultats de simulation de  $R\_MinMax$  restent similaire aux autres métriques avec lesquelles nous l'avons comparée, même si de bonnes performances sont atteintes pour la topologie arbitraire.

Pour cela, dans cette deuxième partie du chapitre, nous nous sommes intéressés à la minimisation du délai de transmission des données sans dégrader les performances atteintes par notre première métrique  $R\_MinMax$  pour les autres critères. Dans notre cas le délai de transmission est la différence entre le moment où un paquet de donnée a été envoyé par la couche application sur un nœud capteur du réseau et le moment où ce même paquet de donnée a été reçu par la couche application sur le nœud racine du DODAG. L'objectif est de concevoir une nouvelle métrique, notée  $R\_Delai$ , permettant à la fois de :

1. prolonger la durée de vie du réseau,



2. réduire le taux de pertes des données,
3. réduire le délai de transmission des données de bout en bout à partir de chaque nœud capteur.

### 6.3.2 Modélisation

Pour minimiser le délai de transmission de bout en bout tout en sélectionnant un chemin avec des nœuds bloquants avec une énergie résiduelle élevée, nous avons combiné la métrique ETX avec la métrique ENG\_MinMax proposée par Kamgoue et al. [Kamgoue et al. 2013] pour composer notre métrique  $R\_Delai$ . Dans ce qui suit, nous considérons un réseau homogène où tous les liens de communications ont les mêmes caractéristiques physique (débit, latence, taux d'erreur ...).

Le paramètre ETX a été repris pour la définition de nouvelles métrique de routage, car c'est une représentation simple et assez fidèle de la qualité d'un lien de communication sans fil, mais pas uniquement. En effet, nous avons également utilisé ce paramètre dans la définition de notre seconde métrique pour représenter d'autres informations utiles. Tout d'abord, la métrique ETX garantie la sélection d'un chemin optimal à la racine en terme de nombre de retransmissions de données [Thomson et al. 2016]. Par ailleurs, on peut observer que ce même chemin permettra aussi d'assurer une collecte rapide des données jusqu'à la racine, car plus le nombre de retransmissions est faible, meilleur est le temps de transmission d'un paquet de données sur chaque lien. Ainsi, ce paramètre donne des informations utile pour réduire le taux de perte sur les chemins mais aussi améliorer les délais de transmission dans le DODAG. Il y a une dernière remarque importante à faire sur ETX en lien avec la consommation énergétique sur un nœud capteur, en considérant que la communication est l'activité la plus consommatrice d'énergie sur les nœuds capteurs. Le paramètre ETX peut aussi être utilisé pour représenter l'énergie consommée par chaque nœud pour transmettre chaque paquet de données. Ainsi, plus la valeur de ETX est faible sur un lien de communications, plus la quantité d'énergie dépensée par un nœud capteur pour envoyer un paquet sera faible. Même si cela constitue des informations utiles à prendre en compte pour l'optimisation de l'énergie consommée par les nœuds capteurs, cela n'est pas suffisant. En effet, utiliser uniquement le paramètre ETX ne peut empêcher la sélection d'un chemin

dans le DODAG contenant un nœud bloquant avec une énergie résiduelle faible. Comme expliqué au chapitre précédent, le paramètre ETX et la métrique associée ne fournissent aucune information sur le niveau d'énergie des nœuds du réseau. Cette constatation nous a donc amené à intégrer les informations fournies par la métrique ENG\_MinMax pour tenir compte de l'énergie résiduelle des nœuds bloquants dans la sélection des chemins.

Comme expliqué dans le paragraphe précédent, il est nécessaire d'utiliser plusieurs paramètres dans la définition de cette seconde métrique. Pour cela, nous devons regarder de plus près la façon de combiner ces paramètres, chacun constituant une métrique primaire. Contrairement à la métrique à la première métrique  $R\_MinMax$ , qui combine une information locale liée à la qualité des liens de communication (exprimée par ETX) et la métrique primaire ENG\_MinMax, la métrique  $R\_Delai$  combine deux métriques primaires : une métrique additive basée sur le paramètre ETX et la métrique primaire concave ENG\_MinMax.

#### 6.3.2.1 Conception d'une métrique composite

Pour rappel, une métrique primaire est une métrique qui ne considère qu'une seule contrainte dans le choix des chemins du DODAG. Elle peut être représentée mathématiquement par un triplet  $(S, \oplus, \prec)$ , où  $S$  est le domaine de définition de la métrique,  $\oplus$  est la fonction d'agrégation des coûts le long des chemins, et  $\prec$  est la relation d'ordre considérée pour la sélection du parent [Karkazis et al. 2012]

Comme détaillé dans [Zahariadis and Trakadas 2012], la combinaison de multiples métriques primaires en une métrique composite peut se faire suivant deux approches distinctes : de manière lexicographique ou additive.

- **Approche lexicographique :** la combinaison de plusieurs métriques primaires en une métrique de routage composite lexicographique, signifie qu'un ordre de priorité est suivi entre les métriques primaires pour la prise en compte des coûts fournis par chacune d'elle. Étant donné deux vecteurs de coûts  $\langle a_1, \dots, a_k \rangle$  et  $\langle b_1, \dots, b_k \rangle$  associés à deux chemins distincts  $A$  et  $B$  fournis par les  $k$  métriques primaires considérées, un nœud  $i$  choisira le chemin  $A$  pour se rattacher au DODAG ssi  $\exists l \leq k, \forall 1 \leq j \leq l, a_j \leq b_j$ , sinon le nœud  $i$  choisira le chemin  $B$ .

- **Approche additive** : la combinaison de plusieurs métriques primaires en une métrique de routage composite additive, signifie que l'on optimise plusieurs critères via une fonction objectif exprimée comme une somme pondérée des métriques primaires. L'Equation 6.2 en est un exemple :

$$W_i = \alpha_1 w_1(u) + \alpha_2 w_2(u) + \dots + \alpha_k w_k(u) \quad (6.2)$$

où  $W_i$  représente le coût du chemin jusqu'à la racine pour le nœud  $i$  en passant par le voisin  $u$ . Le coût  $W_i$  est calculé en fonction des coûts  $w_j(u)$ ,  $1 \leq j \leq k$ , associés à chacune des  $k$  métriques primaires pour le chemin via le voisin  $u$ , et  $\alpha_j$ ,  $1 \leq j \leq k$ , représentent le poids associé à chaque métrique primaire sur le coût final de la métrique composite. Le nœud  $i$  choisit comme parent préféré le voisin fournissant le meilleur coût  $W_i$ .

Une métrique composite doit vérifier les propriétés suivantes [Zahariadis and Trakadas 2012] :

- **Être bien définies** : chaque métrique primaire composée doit avoir un domaine de définition précis. La combinaison des métriques primaires donne place à un nouveau domaine de définition commun.
- **Refléter les caractéristiques de base des LLN** : lors de la composition de plusieurs métriques primaires, on doit s'assurer de la conservation des caractéristiques fondamentales des LLN.
- **Être orthogonales et non antagonistes** : l'orthogonalité signifie que les deux métriques primaires expriment des informations différentes et non redondantes. Cependant, les informations représentées par les diverses métriques primaires composées ne doivent pas être contradictoires.
- **Être continue** : la métrique composite doit refléter les variations mineures sur le coût associé à chaque métrique primaire. Il faut prendre garde à ne pas introduire d'instabilité et d'incohérence sur le coût renvoyé par la métrique composite.
- **Suivre les mêmes propriétés et règles** : les métriques de routage primaires ayant différentes propriétés doivent être transformées pour suivre les mêmes propriétés avant de les combiner en une métrique composite. La divergence des métriques peut

### 6.3. DEUXIÈME PROPOSITION DE MÉTRIQUE

---

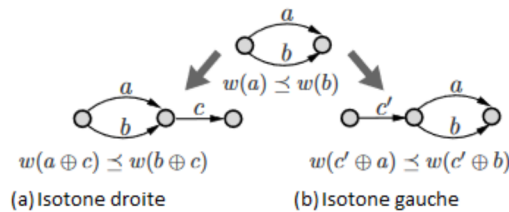
conduire à des instabilités de routage et à la sélection de chemins non optimaux. Cela implique que les métriques primaires soient redéfinies si nécessaire pour utiliser le même domaine de valeurs et la même relation d'ordre.

- **Garantir les propriétés d'isotonicité et de monotonicité** : l'isotonie et la monotonie sont deux propriétés qu'une métrique de routage doit respecter pour garantir des routes cohérentes, c'est-à-dire sans boucle et optimales. Yang et al. [Yang and Wang 2008] démontrent qu'il suffit qu'une métrique soit monotone et isotone pour que le protocole de routage soit sans boucles et optimal. Étant donné une métrique représentée par le triplet  $(S, \oplus, \preceq)$ , nous définissons ci-dessous les notions de monotonie et d'isotonicité :

- **La monotonie** : caractérise le fait que le coût du chemin retourné par une métrique ne peut décroître le long d'un chemin. Plus précisément, une métrique est dite *monotone* si et seulement si  $w(a) \preceq w(b)$  implique à la fois  $w(a) \preceq w(a \oplus b)$  et  $w(a) \preceq w(c \oplus a)$  où  $w(a), w(b)$  et  $w(c)$  représente le coût des chemins a, b et c respectivement. Par conséquent, une métrique monotone ne peut inclure de boucles dans les chemins sélectionnés pour le routage.

- **L'isotonie** : caractérise le fait que le coût total d'un chemin retourné par une métrique est le résultat de l'agrégation du coût des sous chemins qui le compose. Cela implique que si chaque sous chemin est optimal alors le chemin total l'est aussi. Plus formellement, une métrique est isotone si et seulement si étant donné trois sous chemins a, b et c, si  $w(a) \preceq w(b)$  alors  $w(a \oplus c) \preceq w(b \oplus c)$  et  $w(c \oplus a) \preceq w(c \oplus b)$ . Cela est illustré par la Figure 6.6. Cette propriété garantie donc un routage optimal.

FIGURE 6.6 – Métrique isotone [Yang and Wang 2008].



Cependant, lors de la combinaison de ces deux métriques primaires (ETX et ENG\_MinMax),

nous devons respecter certaines propriétés pour assurer la compatibilité avec les spécifications du protocole RPL (convergence, optimalité, absence de boucle de routage).

### 6.3.2.2 Définition de la métrique composite $R\_Delai$

Dans cette section, nous allons présenter deux variantes de la métrique composite que nous dénommerons  $R\_Delai$ . Suite aux éléments présentés ci-dessus, nous allons tout d'abord définir formellement les métriques ETX et ENG\_MinMax (présentées au chapitre précédent), en précisant le domaine de valeurs, la fonction d'agrégation et la relation d'ordre pour ETX, en vue de leur utilisation pour constituer la métrique composite  $R\_Delai$ .

- **ETX(Expected Transmission Count)** : ETX est une métrique de type additive, le coût d'un chemin vers la racine définit le nombre maximal de retransmissions nécessaires pour acheminer un paquet de données sur ce chemin. En suivant cette métrique, chaque nœud sélectionne le voisin annonçant le chemin avec le plus petit nombre de retransmissions jusqu'à la racine. Formellement, le coût d'un chemin avec ETX est représenté par entier avec un domaine de valeurs allant de 1 à 255, la fonction d'agrégation est l'opérateur '+' (somme des coûts le long de chaque chemin) et la relation d'ordre est l'opérateur '<' (choix du voisin avec le coût minimum comme parent). Ainsi, la métrique ETX peut être représentée par le triplet :  $([1, 255], '+', '<')$ .
- **ENG\_MinMax** : ENG\_MinMax est une métrique de type concave, le coût d'un chemin vers la racine représenté par l'énergie résiduelle minimale le long du chemin. Avec cette métrique, chaque nœud sélectionne comme parent préféré le voisin annonçant le chemin ayant la plus grande énergie résiduelle sur son chemin vers la racine. Plus formellement, le coût d'un chemin pour la métrique ENG\_MinMax est représenté par un réel dont le domaine de valeurs est compris entre 0 et 1. Ce réel est donné par le ratio entre l'énergie résiduelle du nœud bloquant du chemin et l'énergie maximale d'un nœud dans le réseau, c'est-à-dire par le ratio  $\frac{Energieresiduelle}{Energimaximale}$ . A chaque niveau, l'agrégation des coûts est réalisée avec la fonction objectif. Selon le draft [Yang and Wang 2008], elle peut être représentée comme un triplet  $([0, 1], 'min', 'max')$ .

### 6.3. DEUXIÈME PROPOSITION DE MÉTRIQUE

---

Nous avons étudié les deux approches décrites précédemment pour associer les métriques primaires ETX et ENG\_MinMax : approche lexicographique et additive. Nous définissons ci-dessous les deux variantes de la métriques composites  $R\_Delai$ .

**Métrique composite lexicographique  $R\_Delai$  :** dans cette première approche les métriques ETX et ENG\_MinMax sont combinées de façon lexicographique. Ainsi, chaque voisin  $j \in N_i$  d'un nœud  $i$  diffuse un vecteur  $\langle ETX_j, PW_j \rangle$  et le nœud  $i$  sélectionne comme parent le voisin  $p \in N_i$  tel que  $\forall j \in N_i, (ETX_p + ETX_{(i,p)}) < (ETX_j + ETX_{(i,j)})$  ou  $((ETX_p + ETX_{(i,p)}) = (ETX_j + ETX_{(i,j)}))$  et  $PW_p \geq PW_j$ .

**Métrique composite additive  $R\_Delai$  :** dans cette approche, on veut combiner la métrique ETX définie par le triplet  $([1, 255], '+', '<')$  avec la métrique ENG\_MinMax définie par le triplet  $([0, 1], 'min', 'max')$ . La combinaison de ces deux métriques nécessite quelques transformations car sinon la métrique composite obtenue ne vérifie pas les propriétés énoncées dans [Zahariadis and Trakadas 2012], et par Yang et Wang [Yang and Wang 2008] présentées précédemment. On explicite ci-dessous les transformations appliquées aux deux métriques primaires ETX et ENG\_MinMax pour vérifier ces propriétés :

- **Domaine de définition :** afin d'avoir un domaine de valeurs commun aux deux métriques primaires, nous avons dérivé la métrique ETX de sorte à exprimer non pas le nombre de retransmissions le long d'un chemin à partir d'un nœud  $i$  mais plutôt le taux de retransmissions nécessaire exprimé suivant l'Equation 6.3 :

$$\frac{\sum_{j \in N_i}^s ETX_j + ETX_{(i,j)}}{Max\_Depth \times Max\_ETX} \quad (6.3)$$

où  $Max\_Depth$  correspond à la profondeur maximale que peut avoir un DODAG dans RPL. Ce taux de retransmissions est un réel sur l'intervalle  $[0, 1]$ , plus ce taux est faible meilleur est la fiabilité et la latence sur le chemin.

- **Relation d'ordre :** par simplicité, nous avons transformé la métrique ENG\_MinMax pour utiliser l'opérateur d'ordre ' $<$ ' en considérant le réel  $(1 - \frac{ENG_{res}}{ENG_{Max}})$  au lieu du ratio  $\frac{ENG_{res}}{ENG_{Max}}$ .

Grâce à ces modifications, nous définissons une métrique composite additive définie dans

### 6.3. DEUXIÈME PROPOSITION DE MÉTRIQUE

---

le domaine de valeurs  $[0, 1]$  et avec la fonction d'ordre 'min'. Chaque nœud  $i$  choisit comme parent préféré celui ayant le plus petit coût de chemin, suivant l'équation 6.4 ci-dessous :

$$R\_Delai_i = \alpha \frac{ETX_j + ETX_{(i,j)}}{Max\_Depth \times Max\_ETX} + (1 - \alpha) \times \left(1 - \frac{PW_j}{MaxEng}\right) \quad (6.4)$$

où  $ETX_j + ETX_{(i,j)}$  représente la somme des ETX du nœud  $i$  jusqu'à la racine en passant par le parent candidat  $j \in N_i$  et  $PW_j$  est égal à l'énergie résiduelle minimale sur le chemin entre la racine et le parent candidat  $j \in N_i$ .

Avant de présenter les résultats obtenus avec les deux variantes de la métrique  $R\_Delai$  que nous proposons, nous devons nous assurer que cette nouvelle métrique composite définit bien des routes cohérentes, sans boucle et optimales, grâce aux propriétés de monotonie et isotonicité.

- **Monotonie** : comme montré dans l'article [Karkazis et al. 2013] la métrique primaire ETX vérifie la propriété de monotonie, car la fonction objectif est une somme des coûts des sous chemins. De plus on peut constater que cette propriété est aussi vérifiée par la métrique ENG\_MinMax. En effet la fonction objectif de cette métrique est de type concave ce qui a pour conséquence d'assurer une continuité et une monotonie des coûts le long des chemins. On peut donc utiliser le Théorème 3 démontré dans [Karkazis et al. 2013] pour conclure que la combinaison de ces deux métriques primaires produit une métrique composite satisfaisant la propriété de monotonie.
- **Isotonie** : comme démontré dans [Karkazis et al. 2013], les métriques ETX et ENG\_MinMax sont isotones. Cependant, la composition de deux métriques primaires isotones ne donnent pas forcément une métrique composite isotone. En effet, lors de la construction d'un chemin l'ajout d'un nouveau lien peut remettre en question l'optimalité d'un sous chemin par rapport à l'optimalité du chemin (c'est-à-dire composé du lien ajouté et du sous chemin). Par exemple sur la Figure 6.3.2.2, en calculant les coûts des chemins (1,2,4) et (1,3,4), il est juste que le nœud 4 sélectionne le nœud 2 comme parent préféré, puisque  $w(1,2,4) < w(1,3,4)$ . Ayant comme seul parent potentiel le nœud 4, le nœud 5 le sélectionnera comme parent préféré bien

### 6.3. DEUXIÈME PROPOSITION DE MÉTRIQUE

que son chemin optimal est (1, 3, 4, 5) puisque  $w(1, 2, 4, 5) > w(1, 3, 4, 5)$ , violant donc la propriété d'isotonicité sur la métrique composite.

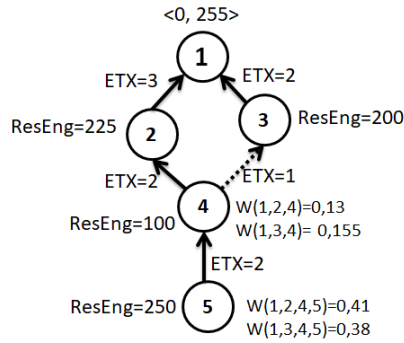


FIGURE 6.7 – Construction du DODAG RPL avec la métrique composite  $R\_Delai$  en prenant en compte la paire  $\langle ETX, ENG\_MinMax \rangle$  lors du choix du parent préféré.

#### 6.3.3 Analyse des résultats de simulation

En se basant sur le même scénario de simulation (paramètres de configuration et topologies) qu'au chapitre 5. Nous présentons dans la suite de ce chapitre les résultats de simulation obtenus pour la métrique  $R\_Delai$  en considérant l'approche additive. Nous ne détaillons pas les résultats associés à l'approche lexicographique car ceux-ci ne sont pas significatif par rapport à l'approche additive (voir Figure 6.8).

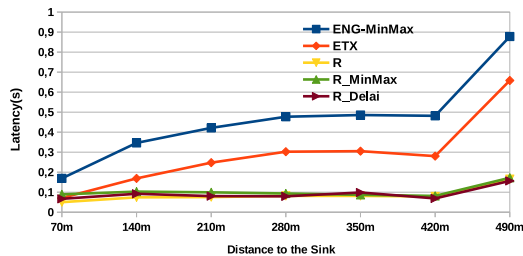
Comme dans la précédente section associée à la métrique  $R\_MinMax$ , pour des raisons de clarté et de pertinence nous ne comparons que les résultats liés aux métriques ETX,  $ENG\_MinMax$ ,  $R$ ,  $R\_MinMax$  avec ceux de la métrique  $R\_Delai$ .

Nous commençons d'abord par évaluer les performances obtenues pour le délai de bout en bout afin de mettre en valeur le gain apporté par notre proposition.

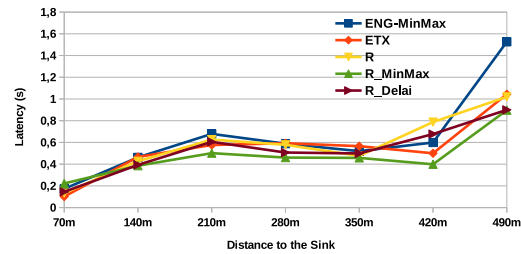
**Délai de bout en bout :** Les résultats de simulation concernant le délai de bout en bout pour les différentes métriques sont donnés par la Figure 6.9. Globalement, on observe que la métrique  $R\_Delai$  atteint des gains compris entre 100 et 200ms par rapport à la métrique  $R\_MinMax$ , proposée à la section précédente, dans le cas où il y a un taux de perte de paquets important (RX=60%) et dans les deux types de topologie. Les gains



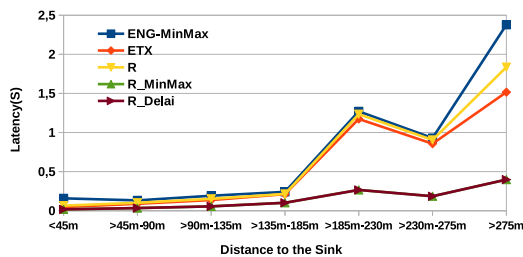
### 6.3. DEUXIÈME PROPOSITION DE MÉTRIQUE



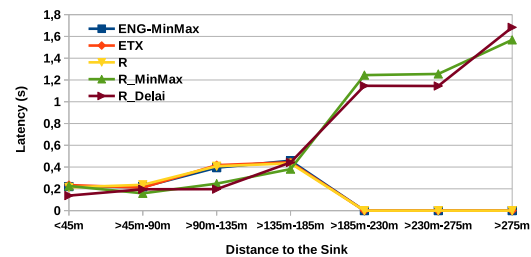
(a) Topologie grille avec RX=100%



(b) Topologie grille avec RX=60%



(c) Topologie arbitraire avec RX=100%



(d) Topologie arbitraire avec RX=60%

FIGURE 6.8 – Le délai de bout en bout (approche lexicographique).

peuvent aller jusqu'à être 3 fois meilleurs par rapport à la métrique ETX dans le cas parfait où le taux de perte de paquets est nul (RX=100%). Plus précisément, dans la topologie grille avec RX=100% (Figure 6.9(a)) le meilleur délai de bout en bout est atteint par les métriques  $R\_Delai$  avec un délai inférieur à 150ms pour les nœuds les plus éloignés, suivie de  $R\_MinMax$  et  $R$ . Les métriques ETX puis ENG\_MinMax atteignent respectivement des performances 3 fois et 4 fois inférieures au-delà de 250m du puits. Dans la même topologie avec RX=60% (Figure 6.9(b)),  $R\_Delai$  obtient de meilleurs résultats en comparaison des autres métriques, avec un délai inférieur à 100ms vis-à-vis de la métrique  $R\_MinMax$  (quelque soit la distance au puits) et 2 fois inférieur par rapport à toutes les autres métriques avec une latence autour de 300ms au lieu de 600ms pour les autres métriques (quelque soit la distance au puits). Ces gains en performance sont essentiellement dû à la considération combinée du nombre de retransmissions et l'énergie résiduelle minimale le long des chemins du DODAG.

Pour la topologie arbitraire, avec RX=100% (Figure 6.9(c)), le délai atteint par la mé-

### 6.3. DEUXIÈME PROPOSITION DE MÉTRIQUE

---

trique  $R\_Delai$  est très légèrement meilleur que celui atteint par la métrique  $R\_MinMax$ , mais reste jusqu'au moins 3 fois meilleur que les métriques ETX et ENG\_MinMax. Le délai atteint par les diverses métriques est proche jusqu'à 100m par rapport au puits, puis deux groupes de métriques se distinguent ( $R\_MinMax$  et  $R\_Delai$  d'une part et ETX, ENG\_MinMax et  $R$  d'autre part) avec un écart sur le délai entre 50ms et 1s pour la majorité des nœuds du réseau à une distance comprise entre 135m et 250m. On peut noter une différence de délai importante par rapport à la métrique ENG\_MinMax, ne tenant pas compte de la qualité des liens, différence allant jusqu'à 2s pour les nœuds les plus éloignés. Dans le cas où RX=60%, le délai augmente dans la première moitié du réseau pour toutes les métriques jusqu'à atteindre un maximum entre 350 et 450ms suivant les métriques pour les nœuds à 185m du puits. Ensuite, dans la seconde moitié du réseau (de 185 à 300m), nous retrouvons les deux mêmes groupes de métriques qui atteignent des performances radicalement orthogonales. Le second groupe de métriques, composé de ETX, ENG\_MinMax et  $R$ , atteint un délai de bout en bout nul indiquant que plus aucun paquet de données n'a pu être acheminé par RPL jusqu'au puits, contrairement au premier groupe de métriques, composé de  $R\_MinMax$  et  $R\_Delai$  que nous proposons, qui assure l'acheminement des paquets de données malgré des délais élevés. Nous pouvons noter que se sont les seules métriques qui garantissent une certaine fiabilité parmi les métriques évaluées. On peut observer que  $R\_Delai$  atteint de meilleurs délais par rapport à  $R\_MinMax$ , de 100 à 400ms inférieur. En effet,  $R\_Delai$  achemine les paquets avec un délai entre 800ms et 1100ms pour les nœuds entre 185 et 250m de la racine contre 1250ms pour les nœuds les plus éloignés (au-delà de 275m), alors que  $R\_MinMax$  atteint des délais supérieurs à 1200ms pour toute cette seconde moitié du réseau.

**Consommation énergétique :** La Figure 6.10 illustre les résultats de simulations concernant la consommation énergétique engendrée par les différentes métriques. De façon générale, les nœuds proches du puits (jusqu'à une distance de 40m dans la topologie grille et environ 90m pour la topologie arbitraire) consomment plus d'énergie que ceux qui sont plus éloignés à cause du routage des paquets de données dans le réseau. Une augmentation de la consommation énergétique pour certaines métriques (ETX, ENG\_MinMax et  $R$ ) a

### 6.3. DEUXIÈME PROPOSITION DE MÉTRIQUE

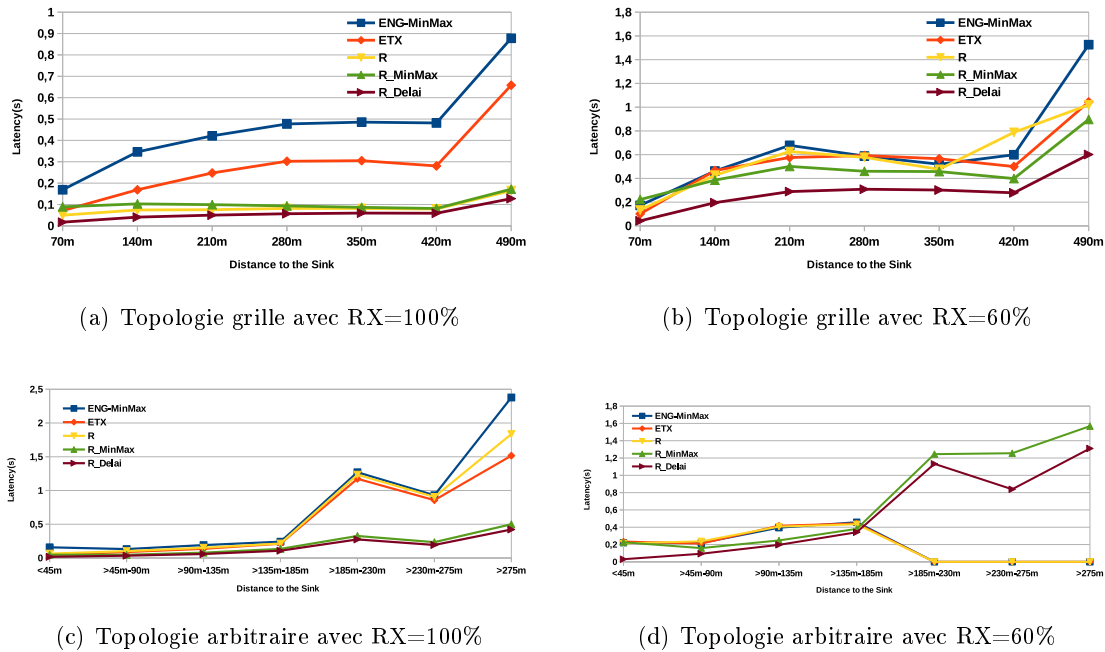


FIGURE 6.9 – Le délai de bout en bout (approche additive).

lieu dans la seconde moitié du réseau dans la topologie arbitraire, cela est dû à des re-transmissions des paquets en conséquence d’instabilités dans le DODAG (comme indiqué par la Figure 6.11 illustrant l’évolution du PDR). De plus, les métriques  $R$ ,  $R\_MinMax$  et  $R\_Delai$ , qui tiennent compte de plusieurs paramètres (énergie résiduelle des nœuds et qualité des liens), consomment beaucoup moins d’énergie que les métriques primaires ETX et ENG\_MinMax. Ainsi, le module radio est éteint de 2 à 3 fois plus longtemps par rapport aux métriques primaires, malgré que la métrique ENG\_MinMax soit conçue pour augmenter la durée de vie du réseau. Cette observation se vérifie pour les deux topologies et les deux taux de pertes de paquets dans le réseau (RX=100% et 60%). Plus précisément, avec la métrique  $R\_Delai$  le module radio est allumé au plus 0.7% du temps dans le cas où RX=100% et au plus 1.7% du temps avec RX=60% (toutes topologies confondues), contre au plus 1.2% du temps si RX=100% et 1.8% du temps si RX=60% avec la métrique  $R\_MinMax$  (toutes topologies confondues).

Comme dans les précédentes sections d’analyse, le Tab. 6.2 présente pour chaque métrique le pourcentage d’énergie maximale consommée par un nœud du réseau après 5 heures

### 6.3. DEUXIÈME PROPOSITION DE MÉTRIQUE

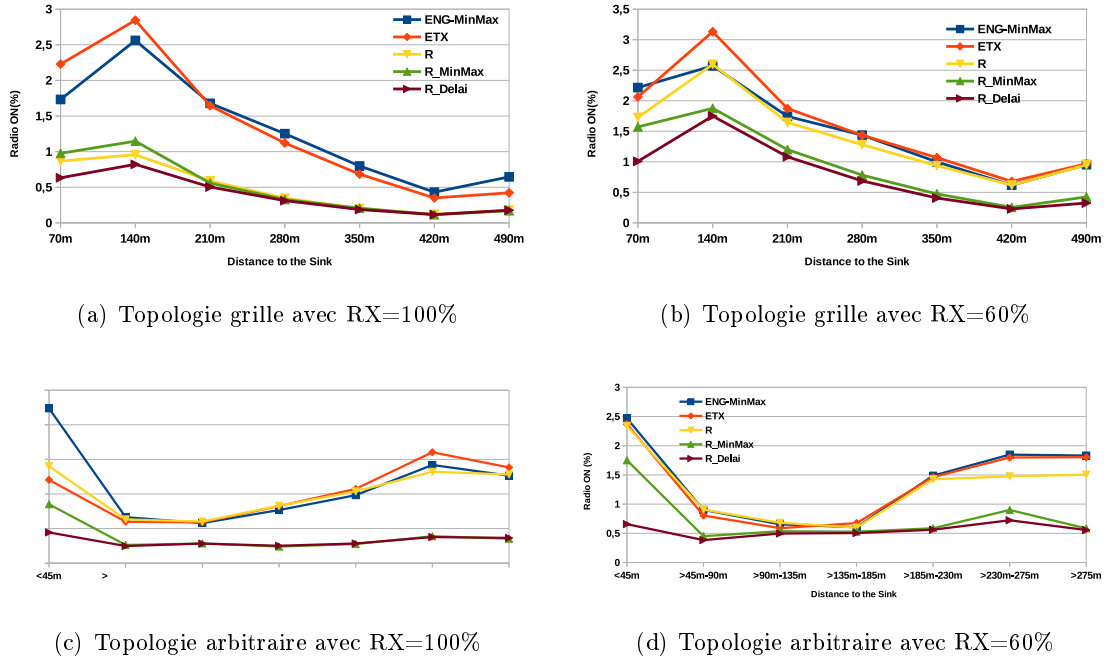


FIGURE 6.10 – Consommation énergétique.

de simulation et une extrapolation de la consommation énergétique dans le réseau exprimée en jours jusqu'à la panne du premier nœud. Ce tableau reprend les valeurs du tableau 6.1 donné à la section précédente, dans lequel est ajouté en dernière colonne les résultats obtenus pour la métrique  $R\_Delai$ . Nous observons que les métriques  $R\_MinMax$  et  $R\_Delai$  donnent des durées de vie similaires quelque soit le type de topologie ou le taux de réception. Elles atteignent les meilleures durée de vie avec 132 jours en RX=100% dans la topologie grille et 74 jours avec RX=60% dans le même type de topologie. Dans la topologie arbitraire avec RX=100% ou RX=60%, les métriques  $R\_MinMax$  et  $R\_Delai$  atteignent des durées de vie de 82 jours et 54 jours par rapport à 15 jours ou 10 jours donnés par les métriques ENG\_MinMax et ETX. Cela nous permet de conclure que notre deuxième métrique proposé  $R\_Delai$  est aussi performante que la métriques  $R\_MinMax$  pour la consommation d'énergie.

**Taux de livraison des paquets :** La Figure 6.11 présente les courbes associées aux taux de livraison de paquets (PDR) pour chaque métrique. Les résultats montrent encore

TABLE 6.2 – Durée de vie du réseau -3-

<b>Topologie grille</b>					
	ENG_MinMax	ETX	R	R_MinMax	R_Delai
RX=	1.12 %	1.15 %	0.157 %	0.158 %	0.150 %
100 %	124698 mj	166098 mj	17359 mj	17535 mj	17010 mj
	19 jours	18 jours	133 jours	132 jours	132 jours
RX=	0.81 %	1.37 %	0.93 %	0.28 %	0.27 %
60 %	88630 mj	151250 mj	103099 mj	30996 mj	30704 mj
	26 jours	15 jours	23 jours	74 jours	74 jours
<b>Topologie arbitraire</b>					
RX=	1.77 %	1.71 %	1.78 %	0.25 %	0.24 %
100 %	196499 mj	188999 mj	196700 mj	28001 mj	26989 mj
	11 jours	13 jours	12 jours	82 jours	82 jours
RX=	1.96 %	1.98 %	1.91 %	0.41 %	0.39 %
60 %	219994 mj	220104 mj	211666 mj	44372 mj	42208 mj
	10 jours	10 jours	11 jours	52 jours	54 jours

une fois que la métrique  $R\_Delai$  surpasse toutes les autres métriques dans les deux topologies et taux de pertes (RX=100% et 60%), même la métrique ETX qui est pourtant parmi les métriques de référence lorsqu'on cherche un routage tenant compte de la qualité des liens. Cela s'explique par le fait qu'elle prend en compte la qualité des liens de communication et l'énergie résiduelle minimale le long de chaque chemin lors de la sélection d'un parent pour chaque nœud dans le DODAG. Cela permet de réduire considérablement les instabilités dans le DODAG dues aux nœuds ayant épuisés leur batterie. Plus particulièrement, la métrique  $R\_Delai$  atteint un taux de livraison des paquets identique à la métrique  $R\_MinMax$  lorsque RX=100%, quelque soit la topologie considérée, en revanche les résultats diffèrent lorsque RX=60%. En effet dans la topologie grille avec RX=60%, la métrique  $R\_Delai$  atteint un PDR d'au moins 90% sur l'ensemble du réseau, contre un taux de livraison allant de 100% à 65% en fonction de la distance au puits avec la métrique  $R\_MinMax$ . Les taux de livraison avec les autres métriques évoluent aussi en fonction de la distance au puits mais les taux sont bien inférieurs, atteignant des taux de 25% pour ETX et 5% pour les métriques ENG\_MinMax et R pour les nœuds les plus éloignés du puits. Les résultats sont encore plus marqués lorsque l'on considère la topologie arbitraire avec RX=60%. Les métriques ETX, ENG\_MinMax et R donnent des résultats similaires,

### 6.3. DEUXIÈME PROPOSITION DE MÉTRIQUE

le taux de livraison diminue de façon plus drastique en fonction de la distance au puits jusqu'à 185m du puits ensuite le PDR est nul car plus aucun paquet de données n'est délivré au puits. Les métriques  $R\_MinMax$  et  $R\_Delai$  atteignent respectivement des taux de livraison jusqu'à 40% et 60% supérieurs aux trois autres métriques pour les nœuds à une distance jusqu'à 185m du puits, au-delà de 185m seuls les métriques  $R\_MinMax$  et  $R\_Delai$  permettent de livrer les paquets de données envoyés par les nœuds situés dans la seconde partie du réseau. En effet, les trois autres métriques atteignent un taux de livraison nul dans cette partie du réseau.

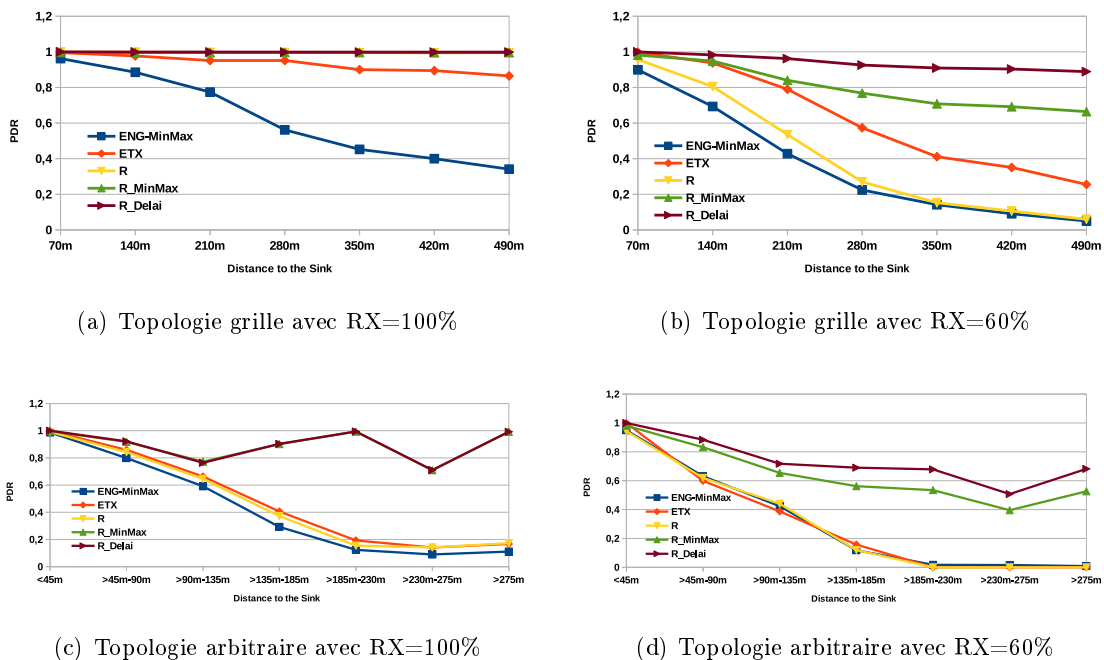


FIGURE 6.11 – Taux de livraison des paquets.

**Messages de contrôle :** La surcharge du réseau induite par les messages de contrôle échangés par le protocole RPL augmente en fonction de la distance au puits, comme illustré par la Figure 6.12. Dans le cas de la topologie grille, nous observons que la métrique  $R\_Delai$  échange un nombre plus important de messages de contrôle afin d'adapter le DODAG et assurer une meilleure fiabilité, sans pour autant trop déstabiliser les routes dans le DODAG comme le montre la Figure 6.11. On observe le même comportement pour

## 6.4. ÉVALUATION DES MÉTRIQUES DANS UNE TOPOLOGIE HÉTÉROGÈNE

la métrique ETX. Alors qu'avec les trois autres métriques, un nombre assez bas et stable de messages de contrôle sont échangés (la contre partie est un PDR faible comme présenté par la Figure 6.11). Dans la topologie arbitraire, le nombre de messages de contrôle échangés est faible et stable pour RX=100% et un peu moins stable pour RX=60%, mais on observe des quantités de messages assez proche des autres métriques.

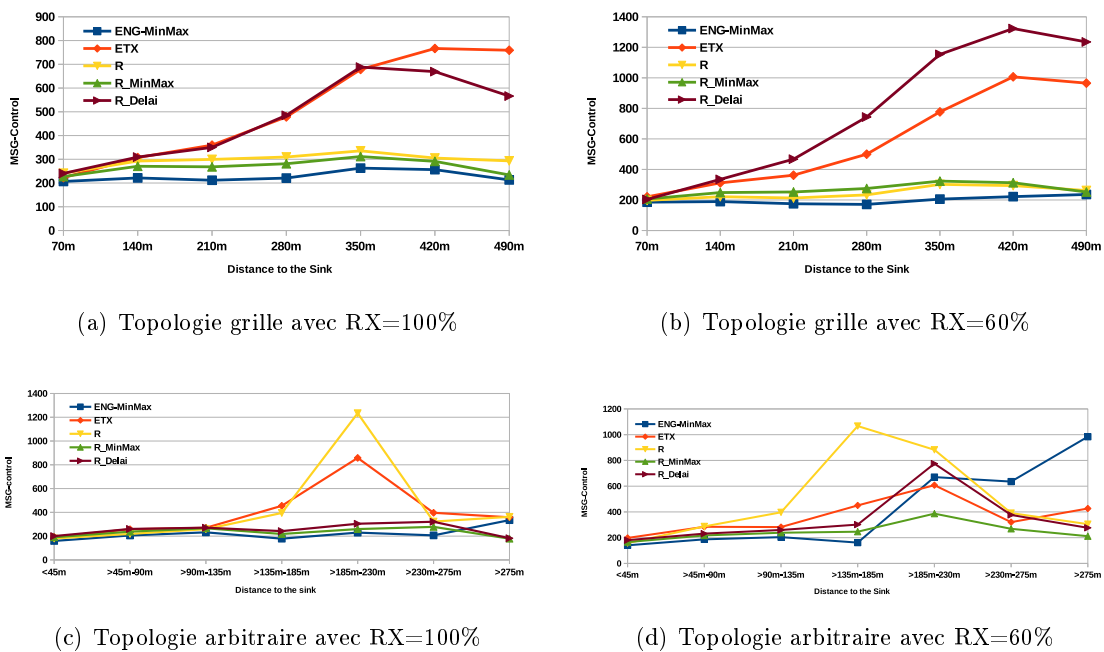


FIGURE 6.12 – Messages de contrôle.

## 6.4 Évaluation des métriques dans une topologie hétérogène

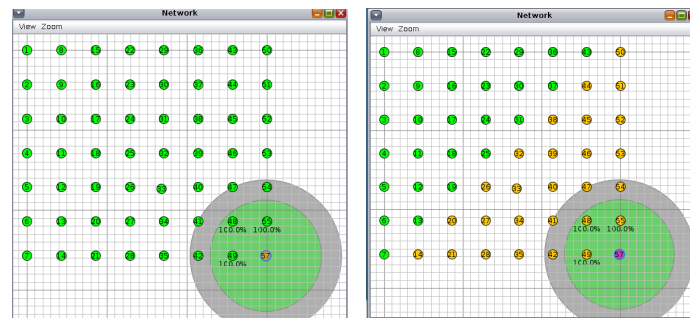
Dans cette section, nous présentons l'évaluation des métriques de routage  $R\_Delai$ ,  $R\_MinMax$ ,  $R$ ,  $ENG\_MinMax$  et  $ETX$  dans un environnement hétérogène. L'objectif est de mettre en avance les performances de nos métriques proposées.

### 6.4.1 Configuration des simulations

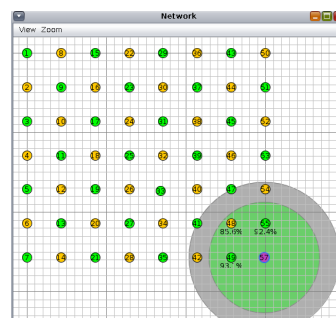
En se basant sur les mêmes paramètres de configuration des simulations précédentes. Nous présentons dans cette partie les résultats de simulation obtenus en considérant que

## 6.4. ÉVALUATION DES MÉTRIQUES DANS UNE TOPOLOGIE HÉTÉROGÈNE

l'ensemble des nœuds formant le réseau n'ont pas forcément la même charge de batterie initiale (c'est-à-dire 100% pour certain nœuds capteurs et 2% pour d'autres). Dans nos simulations, les 56 nœuds sont déployés en topologie grille, la première moitié des nœuds ont la charge maximale (les nœuds jaunes sur la figure 6.13(b) et 6.13(c) et les nœuds verts dans le cas de la topologie homogène 6.13(a)) et la deuxième moitié à 2% de la charge initiale qui est de 853 mAh.



(a) topologie grille homogène (b) topologie grille hétérogène 1



(c) topologie grille hétérogène 2

FIGURE 6.13 – Topologie de simulation

### 6.4.2 Simulations et Analyse

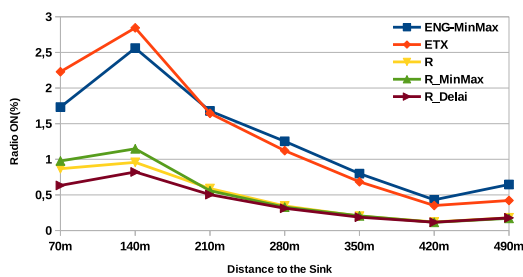
Comme dans les simulation précédentes, nous évaluons les performances de chaque métrique selon quatre critères définis dans le chapitre 5.

**Consommation d'énergie :** La Figure 6.14 représente le pourcentage de temps pendant lequel la radio est allumée, ce qui reflète principalement l'énergie consommée par chaque métrique de routage dans le protocole RPL. Nous observons que la consommation

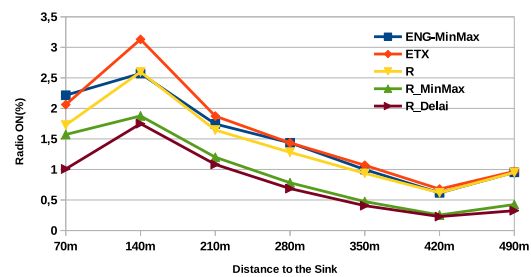


## 6.4. ÉVALUATION DES MÉTRIQUES DANS UNE TOPOLOGIE HÉTÉROGÈNE

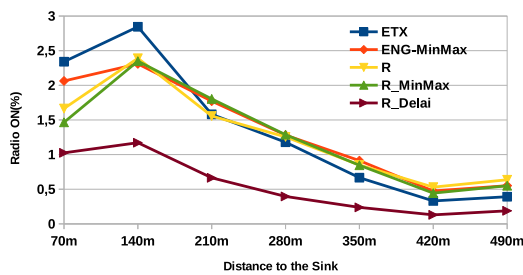
d'énergie augmente puis diminue ensuite en fonction de la distance du nœud puits (racine) pour toutes les métriques quelque soit le taux de pertes de paquets ou la disposition des nœuds hétérogène. Cependant, on peut constater que la métrique  $R\_Delai$  atteint la plus faible consommation énergétique par rapport aux restes des métriques, et le cas hétérogène permet de mettre encore plus en avant cette deuxième métrique proposée par rapport à la métrique  $R\_MinMax$ .



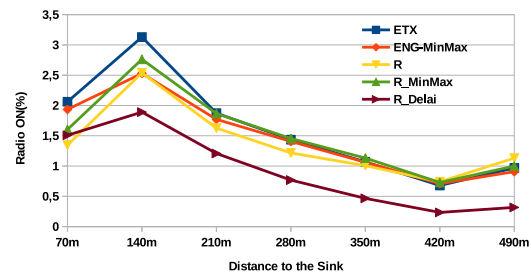
(a) topologie grille homogène avec RX=100%



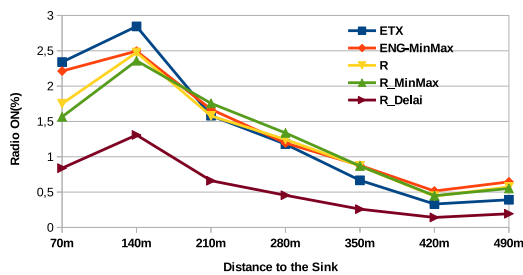
(b) topologie grille homogène avec RX=60%



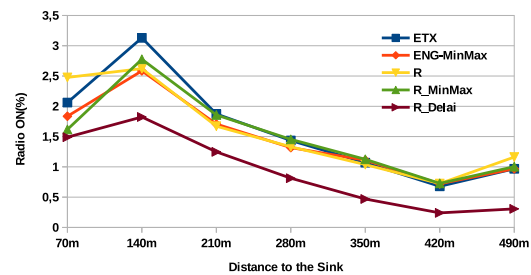
(c) topologie grille hétérogène 1 avec RX=100%



(d) topologie grille hétérogène 1 avec RX=60%



(e) topologie grille hétérogène 2 avec RX=100%



(f) topologie grille hétérogène 2 avec RX=60%

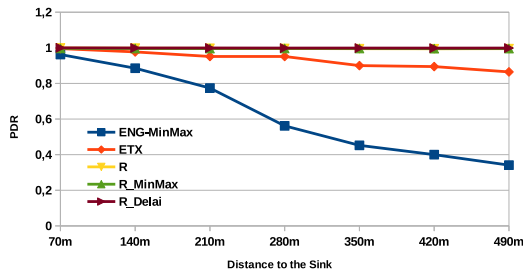
FIGURE 6.14 – Consommation énergétique.

**Taux de livraison des paquets (PDR) :** Dans la Figure 6.15, nous observons le taux de livraison des paquets (PDR). Là encore la métrique  $R\_Delai$  atteint le meilleur taux de livraison des paquets par rapport aux restes des métriques quelque soit le taux de pertes de paquets ou la disposition des nœuds hétérogène. Dans le cas  $RX=100\%$ , dans la topologie homogène les métriques  $R\_Delai$ ,  $R\_MinMax$  et  $R$  atteignent globalement le même taux de livraison, tandis que dans la topologie hétérogène les  $R\_MinMax$  et  $R$  atteignent un PDR inférieur au PDR atteint par la métrique  $R\_Delai$ , ceci est dû à la prise en compte locale de la qualité des liens de communication lors de la sélection du parent préféré. La présence des pertes ( $RX=60\%$ ) ne fait que mettre en avant les performances de la métrique  $R\_Delai$ .

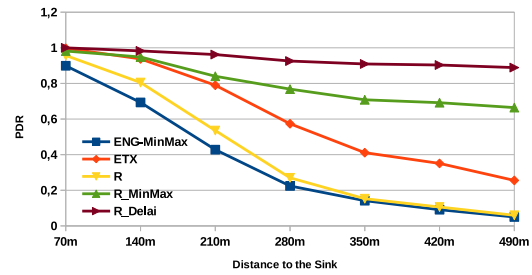
**Délai de bout en bout :** Dans la Figure 6.16, nous observons le délai de bout en bout des métriques de routage  $R\_Delai$ ,  $R\_MinMax$ ,  $R$ ,  $ENG\_MinMax$  et  $ETX$ . Il est à noter que les meilleurs gains en délai de bout en bout de la métrique  $R\_Delai$  sont obtenus dans le cas de topologie hétérogène avec  $RX=100\%$ , car ceci est en relation avec les valeurs de PDR décrites précédemment. Toujours dans le cas hétérogène avec  $RX=100\%$ , près de la racine (jusqu'à 135 m), les métriques  $R\_MinMax$  et  $ETX$  atteignent le délai le plus faible en raison de la stabilité de leur topologie contrairement à la métrique  $R\_Delai$  (voir Figure 6.17). Dans le cas  $RX=60\%$ , globalement la métrique de routage  $R\_Delai$  atteint le délai le plus faible puisqu'elle prend en compte à la fois la qualité de la liaison de communication de bout en bout mais aussi l'énergie minimal au long du chemin.

**Messages de contrôle :** La stabilité de la topologie de routage et la surcharge du réseau est exprimée par la quantité de messages de contrôle RPL échangés par nœuds. Dans la Figure 6.17, nous observons que le nombre de messages de contrôle avec la métrique  $R\_Delai$  est important par rapport aux restes des métriques. Cette importante quantité est liée à la stabilité de cette métrique dû au changement de la sélection du parent préféré.

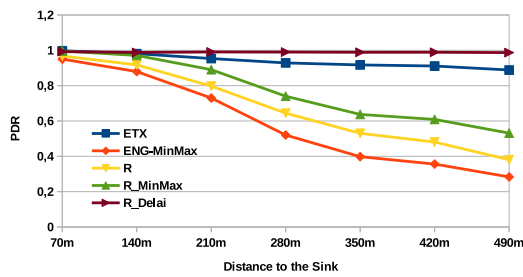
## 6.5. CONCLUSION



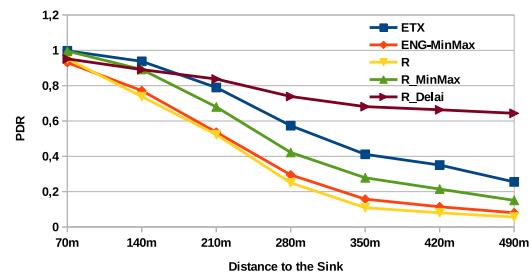
(a) topologie grille homogène avec RX=100%



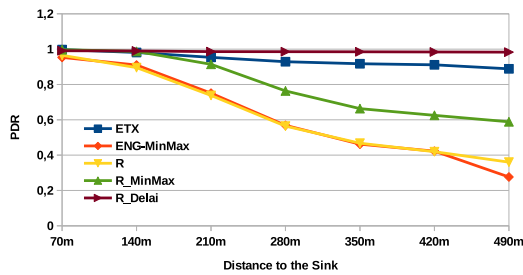
(b) topologie grille homogène avec RX=60%



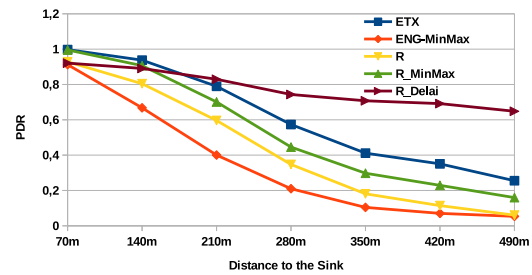
(c) topologie grille hétérogène 1 avec RX=100%



(d) topologie grille hétérogène 1 avec RX=60%



(e) topologie grille hétérogène 2 avec RX=100%



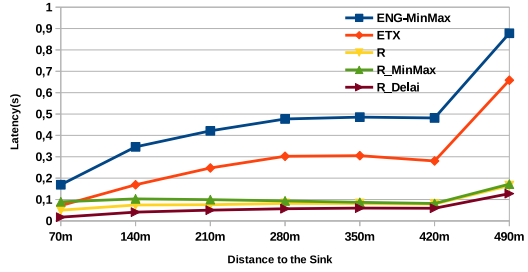
(f) topologie grille hétérogène 2 avec RX=60%

FIGURE 6.15 – Taux de livraison des paquets.

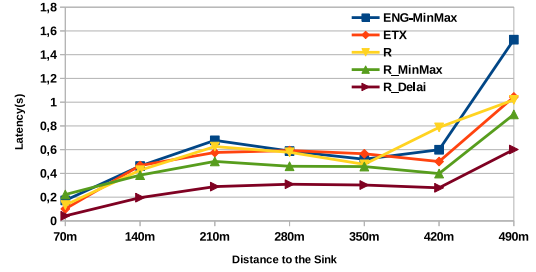
## 6.5 Conclusion

Ce chapitre a été consacré à la présentation et l'évaluation de deux nouvelles métriques de routage proposées pour le protocole RPL, dénommées  $R\_MinMax$  et  $R\_Delai$ , en vue d'améliorer la durée de vie du réseau et avec de bonnes performances sur le taux de livraison et le délai de bout en bout pour acheminer les données au puits. Nous avons évalué les performances de ces métriques par rapport à trois autres métriques RPL de l'état de l'art

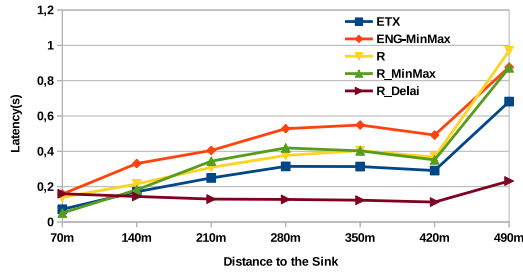
## 6.5. CONCLUSION



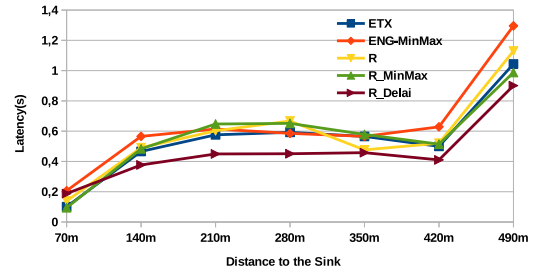
(a) topologie grille homogène avec RX=100%



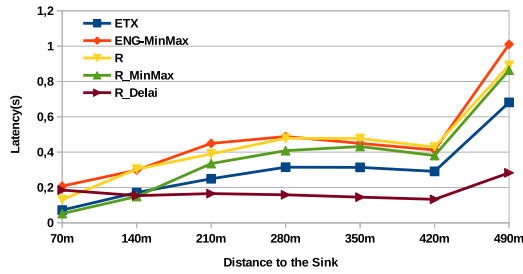
(b) topologie grille homogène avec RX=60%



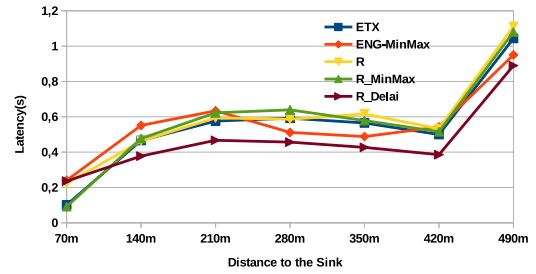
(c) topologie grille hétérogène 1 avec RX=100%



(d) topologie grille hétérogène 1 avec RX=60%



(e) topologie grille hétérogène 2 avec RX=100%



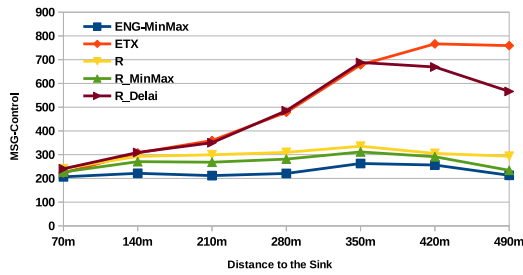
(f) topologie grille hétérogène 2 avec RX=60%

FIGURE 6.16 – Le délai de bout en bout.

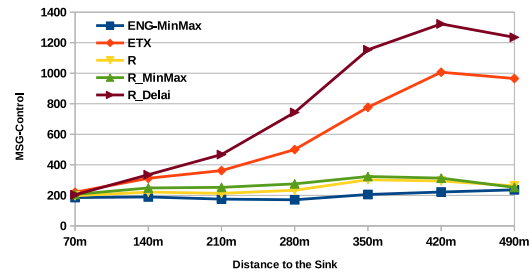
et avons montré les gains apportés grâce à la composition dans une métrique de plusieurs paramètres importants, qui sont la qualité des liens de communication et l'énergie résiduelle des nœuds bloquant le long des chemins.

Dans la première partie de ce chapitre, nous avons présenté la métrique  $R\_MinMax$  basée sur ETX et la métrique primaire  $R$  proposé dans [Chang et al. 2013], qui tient compte de l'énergie résiduelle minimale le long des chemins vers la racine dans le DODAG.

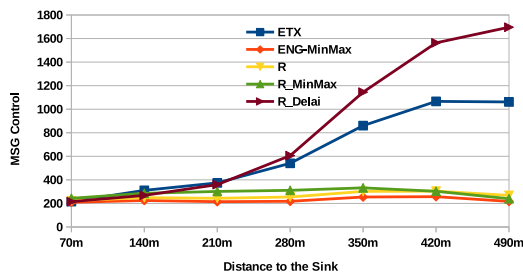
## 6.5. CONCLUSION



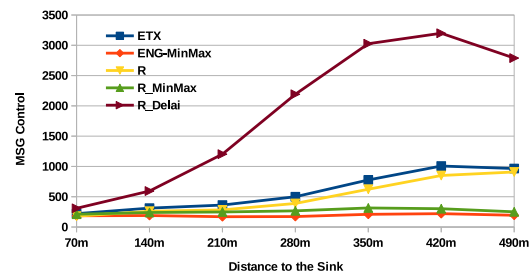
(a) topologie grille homogène avec RX=100%



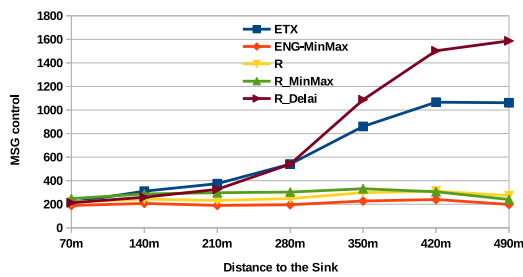
(b) topologie grille homogène avec RX=60%



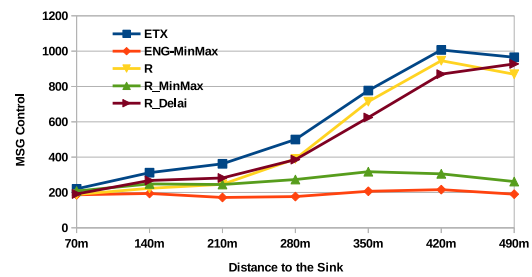
(c) topologie grille hétérogène 1 avec RX=100%



(d) topologie grille hétérogène 1 avec RX=60%



(e) topologie grille hétérogène 2 avec RX=100%



(f) topologie grille hétérogène 2 avec RX=60%

FIGURE 6.17 – Les Messages de contrôle.

L'objectif principal de cette proposition est principalement de prolonger la durée de vie du réseau. Les résultats de simulation obtenus montre que cette première métrique atteint dans la topologie grille une durée de vie d'environ 5 à 7 fois plus importante que les autres métriques lorsque RX=100%, et environ 3 à 5 fois meilleure pour RX=60%. Pour la topologie arbitraire, la métrique  $R\_MinMax$  atteint une amélioration plus importante par rapport aux autres métriques, avec une durée de vie 6 à 7 fois supérieure lorsque RX=100%

et environ 5 fois plus pour RX=60%.

Dans la deuxième partie de ce chapitre, nous décrivons une seconde métrique, appelée *R\_Delai*, qui tient compte à la fois de l'énergie résiduelle minimale mais aussi de la qualité des liens de communication le long des chemins dans le DODAG. L'objectif principal de cette autre proposition est de minimiser le délai de transmission de bout en bout des paquets de données. Pour cela, nous avons conçu une métrique composite additive combinant la métrique ETX avec la métrique ENG\_MinMax proposée par Kamgoue et al. [Kamgoue et al. 2013]. Les résultats de simulation obtenus montrent que la métrique *R\_Delai* atteint un délai jusqu'à 2 à 3 fois inférieur par rapport aux autres métriques pour les deux topologies et taux de pertes dans le réseau. Il est à noter que pour la topologie arbitraire la métrique *R\_Delai* permet d'acheminer les paquets pour la moitié des nœuds les plus éloignés du puits là où toutes les autres métriques (excepté *R\_MinMax*) sont mis en défaut.

Nos résultats montrent qu'il est nécessaire de concevoir des métriques multi-critères qui tiennent compte à la fois de la qualité des liens de communication et de l'énergie résiduelle des nœuds pour améliorer la qualité de service dans les WSN. Une perspective majeure envisagée est d'implémenter ces deux propositions de métriques RPL sur des déploiements réels de réseaux de capteurs afin de valider nos résultats de simulation. D'autre part, il serait intéressant de développer d'autres métriques composites permettant de garantir d'autres formes de qualité de service.

# Chapitre 7

## Conclusion et perspectives

### 7.1 Conclusion

Cette dernière décennie a vu l'émergence de nouveaux domaines d'applications en lien avec l'internet des objets (*IoT - Internet of Things*), comme par exemple la maison intelligente ou encore la ville intelligente. Cela concerne également les réseaux de capteurs sans fil qui exigent un besoin de plus en plus fort en termes de qualité de service d'une part et de prolongation de la durée de vie du réseau d'autre part. Beaucoup de travaux de recherche et développement ont été menés durant ces dix dernières années pour améliorer les algorithmes et protocoles au niveau du partage des ressources et du routage des données dans les réseaux de capteurs sans fil.

Dans cette thèse, notre objectif a été d'une part d'augmenter la durée de vie du réseau de capteurs, et d'autre part de fournir des garanties sur le temps de collecte des données. Nous avons tout d'abord décrit au Chapitre 2 les caractéristiques et limitations liées à ce type de réseaux, puis nous avons présenté au Chapitre 3 les principales approches et protocoles pour l'accès au médium sans fil et le routage des données, en mettant en avant les défis à relever sur ces deux parties du point de vue de la consommation énergétique et du temps de collecte des données. En effet, ces deux couches protocolaires ont un impact considérable sur la durée de vie du réseau et le temps d'acheminement des données.

Dans le Chapitre 4, nous nous sommes intéressés plus spécifiquement au partage de l'accès au médium radio afin d'ordonner les transmissions sans fil pour des applications de collecte de données dans les réseaux de capteurs. Nous avons adopté une approche

partageant le médium dans le temps entre les nœuds du réseau (approche TDMA) et avons défini un problème spécifique de coloration des arêtes dans un graphe pour modéliser l'allocation d'intervalles de temps (« slots ») aux liens de communication sans générer de collisions sur les transmissions. L'objectif est donc d'ordonnancer les transmissions de données à travers un arbre de collecte, de sorte à maximiser le nombre de liens activés simultanément en tenant compte des conflits possibles entre les transmissions. Nous avons ensuite étudié la complexité de ce problème de coloration et avons proposé une approche distribuée DD2EC et tolérante aux fautes transitoires SS-DD2EC pour résoudre celui-ci.

Par la suite dans le Chapitre 5, nous avons étudié le fonctionnement du protocole de routage RPL (*Routing Protocol for Low-power and lossy networks*), qui est à ce jour le seul protocole de routage standardisé pour les réseaux de capteurs multi-sauts basés sur IPv6. Nous avons ensuite présenté une étude comparative de plusieurs métriques de routage proposées dans la littérature pour améliorer RPL en vue de prolonger la durée de vie du réseau. Cette évaluation a été menée sur une topologie grille et une topologie arbitraire, en considérant deux modèles de transmissions (avec ou sans pertes de paquets). L'analyse du comportement de différentes métriques a été réalisée suivant plusieurs critères : l'énergie consommée, le taux de transmission de paquets, le délai de transmission de bout en bout et le nombre de messages de contrôle échangés dans le réseau. Les résultats de simulation montrent que les métriques bicritères présentent de meilleures performances en termes de durée de vie du réseau et de délais de bout en bout.

Suite aux résultats de l'évaluation du Chapitre 5, nous avons proposé au Chapitre 5 deux nouvelles métriques de routage multi-critères pour améliorer les performances atteintes avec le protocole RPL. La première métrique, dénommée  $R\_MinMax$ , est une métrique bicritères qui tient compte de l'énergie résiduelle des nœuds et de la qualité des liens de communication du réseau. Les résultats de simulation obtenus montrent que cette première métrique atteint des performances jusqu'à 5 à 7 fois meilleures pour la durée de vie du réseau en conservant un taux de transmission de paquets équivalent en comparaison des métriques considérées au Chapitre 5 dans le modèle sans perte de données, et jusqu'à 3 à 5 fois meilleures dans le modèle avec perte de données. Dans ce même chapitre, une seconde métrique est également proposée, appelée  $R\_Delai$ . Cette seconde proposition est



une métrique composite qui combine deux métriques RPL existantes dans la littérature (les métriques ETX et *ENG\_MinMax*). L'objectif de cette proposition est de minimiser le délai de transmission de bout en bout des données jusqu'au puits, en plus de l'augmentation de la durée de vie du réseau. Les résultats de simulation obtenus montrent encore une nouvelle fois que notre proposition *R\_Delai* atteint un délai jusqu'à 2 à 3 fois inférieur par rapport aux métriques considérées au Chapitre 5, sans dégrader les performances atteintes sur la durée de vie du réseau.

## 7.2 Perspectives

Les problématiques du domaine des réseaux de capteurs étant très vastes, nous avons choisi dans cette thèse de nous concentrer sur la consommation énergétique des réseaux de capteurs. Pour prolonger les résultats présentés dans ce manuscrit, il reste de nombreuses pistes intéressantes à explorer pour de futurs travaux de recherche dans le but de continuer à améliorer la durée de vie des réseaux de capteurs.

Dans cette thèse, nous avons étudié et analysé le gain sur la consommation énergétique, le délai de transmission et le taux de perte en se basant sur l'évaluation des solutions proposées par des preuves théoriques ou des simulations. Il serait donc intéressant à court terme d'évaluer les propositions de nos travaux et de les tester via des expérimentations. Cela permettrait d'évaluer les gains sur la consommation énergétique, le délai de transmission et le taux de pertes en situation réelle afin de déterminer l'intérêt d'une mise en application de nos propositions.

Tout d'abord, il serait intéressant de mettre en place sur un déploiement réel de réseau de capteurs l'approche proposée au Chapitre 3 pour l'ordonnancement des transmissions. Cela permettrait de mesurer les performances atteintes en comparaison d'autres protocoles d'accès au médium. Afin de compléter notre proposition au niveau de la couche d'accès au canal de transmission avec comme objectif d'optimiser encore plus le nombre de couleurs nécessaires pour la coloration du graphe, il serait intéressant d'étudier les nouvelles méthodes d'accès au médium introduites dans le standard IEEE 802.15.4e qui combine l'accès au médium par TDMA et la technique de multi-fréquence radio, pour présenter de nou-

veaux protocoles d'accès nettement plus performants et complexes que ceux du standard IEEE 802.15.4.

Un autre axe de perspective serait d'approfondir nos travaux au niveau du routage. Pour chacune de nos propositions nous avons décrit le contexte général des métriques proposées, la modélisation de ces métriques de routage appliquées au protocole RPL et enfin nous complétons ces propositions par une analyse comparative des résultats de nos simulations. Ces comparaisons montrent qu'il est nécessaire de concevoir des métriques multi-critères qui tiennent compte à la fois de la qualité des liens de communication et de l'énergie résiduelle des nœuds pour faire face aux exigences attendues. La validation de ces simulations seraient à réaliser, dans l'idéal, sur du matériel réel ou sur la plateforme FIT IoT-LAB<sup>1</sup> ou la plateforme OpenWiNo<sup>2</sup>. Afin d'améliorer le taux de réception des données tout en maximisant la durée de vie, il serait aussi intéressant de proposer une nouvelle métrique composite prenant en compte des critères supplémentaires, comme le trafic ou le débit, en utilisant la logique floue [Gaddour et al. 2014].

A plus long terme, nous envisageons d'orienter nos travaux de recherche dans certaines directions que nous considérons comme intéressantes. Dans un premier temps, il serait intéressant d'étendre notre étude d'hétérogénéité au niveau de l'énergie où certains nœuds ont moins d'énergie initiale que d'autres à une hétérogénéité au niveau des liens de communications en terme de délai de transmission ou encore au niveau des puissances de transmission. Puis dans un second temps il serait aussi intéressant d'étudier la stabilité et les performances de RPL dans le cas de la mobilité des nœuds capteurs. Des travaux ont déjà été menés pour introduire la mobilité dans RPL, mobilité des nœuds avec prédiction de la trajectoire [Soma et al. 2016], et mobilité des nœuds avec un puits statique [Lamaazi et al. 2017].

---

1. <https://www.iot-lab.info/>

2. <https://wino.cc/>

# Bibliographie

- D.V. Ageev. Bases of the theory of linear selection. code demultiplexing. In *Proceedings of the Leningrad Experimental Institute of Communication*, pages 3–35, 1935. 45
- Ian F Akyildiz and Mehmet Can Vuran. *Wireless sensor networks*, volume 4. John Wiley and Sons, 2010. 14, 18, 22, 26
- Arianna Alfieri, Andrea Bianco, Paolo Brandimarte, and Carla-Fabiana Chiasserini. Maximizing system lifetime in wireless sensor networks. *European journal of Operational research*, 181 :390–402, 2007. 31
- ZigBee Alliance. Ieee 802.15. 4, zigbee standard, 2009. 32
- Ichrak Amdouni, Pascale Minet, and Cedric Adjih. Node coloring in wireless networks : complexity results and grid coloring. In *Wireless and Mobile Networking Conference (WMNC), 2011 4th Joint IFIP*, pages 1–9. IEEE, 2011. 82
- Giuseppe Anastasi, Marco Conti, Mario Di Francesco, and Andrea Passarella. Energy conservation in wireless sensor networks : A survey. *Ad Hoc Networks*, 7(3) :537–568, 2009. doi : 10.1016/j.adhoc.2008.06.003. URL <https://doi.org/10.1016/j.adhoc.2008.06.003>. 57
- Mert Bal, Weiming Shen, and Hamada Ghenniwa. Collaborative signal and information processing in wireless sensor networks : a review. In *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics, San Antonio, TX, USA, 11-14 October 2009*, pages 3151–3156, 2009. doi : 10.1109/ICSMC.2009.5346152. URL <http://dx.doi.org/10.1109/ICSMC.2009.5346152>. 41

## BIBLIOGRAPHIE

---

- Balabhaskar Balasundaram and Sergiy Butenko. Graph domination, coloring and cliques in telecommunications. In *Handbook of Optimization in Telecommunications*, pages 865–890. Springer, 2006. ISBN 978-0-387-30662-9. doi : 10.1007/978-0-387-30165-5\_30. URL [https://doi.org/10.1007/978-0-387-30165-5\\_30](https://doi.org/10.1007/978-0-387-30165-5_30). 70
- Paolo Baronti, Prashant Pillai, Vince WC Chook, Stefano Chessa, Alberto Gotta, and Y Fun Hu. Wireless sensor networks : A survey on the state of the art and the 802.15.4 and zigbee standards. *Computer communications*, 30(7) :1655–1695, 2007. 29, 37
- Samuel Bernard, Stéphane Devismes, Katy Paroux, Maria Potop-Butucaru, and Sébastien Tixeuil. Probabilistic self-stabilizing vertex coloring in unidirectional anonymous networks. In *11th International Conference on Distributed Computing and Networking*, pages 167–177, 2010. doi : 10.1007/978-3-642-11322-2\_19. URL [https://doi.org/10.1007/978-3-642-11322-2\\_19](https://doi.org/10.1007/978-3-642-11322-2_19). 102
- Jean R. S. Blair and Fredrik Manne. An efficient self-stabilizing distance-2 coloring algorithm. *Theoretical Computer Science*, 444 :28–39, 2012. doi : 10.1016/j.tcs.2012.01.034. URL <https://doi.org/10.1016/j.tcs.2012.01.034>. 102
- Tiziana Calamoneri. The  $L(h, k)$ -labelling problem : An updated survey and annotated bibliography. *Comput. J.*, 54(8) :1344–1371, 2011. doi : 10.1093/comjnl/bxr037. URL <https://doi.org/10.1093/comjnl/bxr037>. 74
- Kathie Cameron. Induced matchings. *Discrete Applied Mathematics*, 24(1-3) :97–102, 1989. doi : 10.1016/0166-218X(92)90275-F. URL [https://doi.org/10.1016/0166-218X\(92\)90275-F](https://doi.org/10.1016/0166-218X(92)90275-F). 78
- Lin-Huang Chang, Tsung-Han Lee, Shu-Jan Chen, and Cheng-Yen Liao. Energy-efficient oriented routing algorithm in wireless sensor networks. In *IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 3813–3818, 2013. 124, 125, 129, 140, 141, 142, 171
- Gary Chartrand and Ping Zhang. *Chromatic graph theory*. CRC Press, US, 2008. ISBN 9781584888000. 70

## BIBLIOGRAPHIE

---

- Pranay Chaudhuri and Hussein Thompson. Improved self-stabilizing algorithms for  $l(2, 1)$ -labeling tree networks. *Mathematics in Computer Science*, 5(1) :27–39, 2011. doi : 10.1007/s11786-011-0081-6. URL <https://doi.org/10.1007/s11786-011-0081-6>. 102
- Maggie X. Cheng and Li Yin. Transmission scheduling in sensor networks via directed edge coloring. In *Proceedings of IEEE International Conference on Communications (ICC)*, pages 3710–3715, 2007. doi : 10.1109/ICC.2007.611. URL <https://doi.org/10.1109/ICC.2007.611>. 83, 85
- Victor Cionca, Thomas Newe, and Vasile Dadârlat. Tdma protocol requirements for wireless sensor networks. In *Sensor Technologies and Applications, 2008. SENSORCOMM'08. Second International Conference on*, pages 30–35. IEEE, 2008. 44
- Thomas Clausen and Philippe Jacquet. Optimized link state routing protocol (olsr). RFC 3626, Internet Engineering Task Force (IETF), 2003. URL <https://www.rfc-editor.org/rfc/rfc3626.txt>. 60
- Thomas Clausen, A Colin de Verdiere, Jiazi Yi, Afshin Niktash, and Yuichi Igarashi. The lightweight on-demand ad hoc distance-vector routing protocol-next generation (loadng). *draft-clausen-lln-loadng-09 (work in progress)*, 2013. 63
- Domenico De Guglielmo, Simone Brienza, and Giuseppe Anastasi. Ieee 802.15. 4e : A survey. *Computer Communications*, 88 :1–24, 2016. 32
- Eric B. Decker, Venkatesh Rajendran, Katia Obraczka, and J. J. Garcia-Luna-Aceves. The multi-channel flow-aware medium access control protocol for wireless sensor networks. In *Proceedings of the IEEE 19th International Symposium on Personal, Indoor and Mobile Radio Communications, PIMRC 2008, 15-18 September 2008, Cannes, French Riviera, France*, pages 1–5, 2008. doi : 10.1109/PIMRC.2008.4699744. URL <http://dx.doi.org/10.1109/PIMRC.2008.4699744>. 54
- Fabio Demicheli. *Design, implementation and Evaluation of an Energy RPL Routing Metric*. LAR lambert Academic Publishing, 2014. 120, 121, 125, 128
- E.W. Dijkstra. Self-stabilizing systems in spite of distributed control. *Commun. ACM*, 17 (11) :643–644, 1974. 100

- Kaouther Drira, Lyes Dekar, and Hamamache Kheddouci. A self-stabilizing  $(\delta+1)$ -edge-coloring algorithm of arbitrary graphs. In *International Conference on Parallel and Distributed Computing, Applications and Technologies (PDCAT)*, pages 312–317, 2009. doi : 10.1109/PDCAT.2009.71. URL <https://doi.org/10.1109/PDCAT.2009.71>. 102
- Kaouther Drira, Hamida Seba, Brice Effantin, and Hamamache Kheddouci. Distance edge coloring and collision-free communication in wireless sensor networks. *Networks*, 62(1) : 35–47, 2013. doi : 10.1002/net.21491. URL <https://doi.org/10.1002/net.21491>. 74
- Adam Dunkels. The contikimac radio duty cycling protocol. Technical Report T2011 :13, Swedish Institute of Computer Science, 2011. 14, 52, 53, 125
- Adam Dunkels, Joakim Eriksson, Niclas Finne, and Nicolas Tsiftes. Powertrace : Network-level power profiling for low-power wireless networks. Technical Report T2011 :05, Swedish Institute of Computer Science, 2011. 126
- Olfa Gaddour and Anis Koubaa. RPL in a nutshell : A survey. *Computer Networks*, 56(14) : 3163–3178, 2012. ISSN 1389-1286. doi : <http://dx.doi.org/10.1016/j.comnet.2012.06.016>. 16, 109, 111, 115
- Olfa Gaddour, Anis Koubaa, Nouha Baccour, and Mohamed Abid. OF-FL : qos-aware fuzzy logic objective function for the RPL routing protocol. In *12th International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks, WiOpt 2014, Hammamet, Tunisia, May 12-16, 2014*, pages 365–372, 2014. doi : 10.1109/WIOPT.2014.6850321. 177
- Shashidhar Gandham, Milind Dawande, and Ravi Prakash. Link scheduling in wireless sensor networks : Distributed edge-coloring revisited. *J. Parallel Distrib. Comput.*, 68(8) :1122–1134, 2008a. doi : 10.1016/j.jpdc.2007.12.006. URL <http://dx.doi.org/10.1016/j.jpdc.2007.12.006>. 69, 81, 85
- Shashidhar Gandham, Ying Zhang, and Qingfeng Huang. Distributed time-optimal scheduling for convergecast in wireless sensor networks. *Computer Networks*, 52(3) :610–629, 2008b. 68

## BIBLIOGRAPHIE

---

- Fanica Gavril. Algorithms for minimum coloring, maximum clique, minimum covering by cliques, and maximum independent set of a chordal graph. *SIAM Journal on Computing*, 1(2) :180–187, 1972. doi : 10.1137/0201013. URL <https://doi.org/10.1137/0201013>. 78, 79
- Amitabha Ghosh, Özlem Durmaz Incel, V. S. Anil Kumar, and Bhaskar Krishnamachari. Multichannel scheduling and spanning trees : Throughput-delay tradeoff for fast data collection in sensor networks. *IEEE/ACM Trans. Netw.*, 19(6) :1731–1744, 2011. doi : 10.1109/TNET.2011.2146273. URL <http://dx.doi.org/10.1109/TNET.2011.2146273>. 16, 67, 68, 75, 80, 81, 84, 85, 107
- Amitabha Amitava Ghosh, Özlem Durmaz Incel, Bhaskar Krishnamachari, and Anil Vullikanti. Multi-channel scheduling algorithms for fast aggregated convergecast in sensor networks. In *IEEE 6th International Conference on Mobile Adhoc and Sensor Systems, 12-15 October 2009, Macau (S.A.R.), China*, pages 363–372, 2009. doi : 10.1109/MOBHOC.2009.5336979. URL <https://doi.org/10.1109/MOBHOC.2009.5336979>. 75
- Google. About soli, 2015. URL <https://atap.google.com/soli/>. 26
- David A. Grable and Alessandro Panconesi. Nearly optimal distributed edge coloring in  $o(\log \log n)$  rounds. *Random Struct. Algorithms*, 10(3) :385–405, 1997. doi : 10.1002/(SICI)1098-2418(199705)10:3<385::AID-RSA6>3.0.CO;2-S. URL [http://dx.doi.org/10.1002/\(SICI\)1098-2418\(199705\)10:3<385::AID-RSA6>3.0.CO;2-S](http://dx.doi.org/10.1002/(SICI)1098-2418(199705)10:3<385::AID-RSA6>3.0.CO;2-S). 69, 83, 85
- Maria Gradinariu and Sébastien Tixeuil. Self-stabilizing vertex coloration and arbitrary graphs. In *4th International Conference on Principles of Distributed Systems*, pages 55–70, 2000. 101
- Jose A Gutierrez, Marco Naeve, Ed Callaway, Monique Bourgeois, Vinay Mitter, and Bob Heile. Ieee 802.15. 4 : a developing standard for low-power low-cost wireless personal area networks. *IEEE network*, 15(5) :12–19, 2001. 31
- Jose A. Gutierrez, Edgar H. Callaway, and Raymond Barrett. *IEEE 802.15.4 Low-Rate Wireless Personal Area Networks : Enabling Wireless Sensor Networks*. IEEE Standards Office, New York, NY, USA, 2003. ISBN 0738135577. 46

## BIBLIOGRAPHIE

---

- Ted Herman and Sébastien Tixeuil. A distributed TDMA slot assignment algorithm for wireless sensor networks. In *First International Workshop on Algorithmic Aspects of Wireless Sensor Networks (ALGOSENSORS)*, pages 45–58, 2004. doi : 10.1007/978-3-540-27820-7\_6. URL [https://doi.org/10.1007/978-3-540-27820-7\\_6](https://doi.org/10.1007/978-3-540-27820-7_6). 102
- Ted Herman, Imran Pirwani, and Sriram Pemmarraju. Oriented edge colorings and link scheduling in sensor networks. In *Proceedings of 1st International Workshop on Software for Sensor Networks (SENSORWARE 2006)*, 2006. 73, 83, 85
- Ian Holyer. The np-completeness of some edge-partition problems. *SIAM J. Comput.*, 10(4) :713–717, 1981. doi : 10.1137/0210054. URL <http://dx.doi.org/10.1137/0210054>. 71
- Ivan Howitt and Jore A Gutierrez. Ieee 802.15. 4 low rate-wireless personal area network coexistence issues. In *Wireless Communications and Networking, 2003. WCNC 2003. 2003 IEEE*, volume 3, pages 1481–1486. IEEE, 2003. 32
- Pei Huang, Li Xiao, Soroor Soltani, Matt W. Mutka, and Ning Xi. The evolution of MAC protocols in wireless sensor networks : A survey. *IEEE Communications Surveys and Tutorials*, 15(1) :101–120, 2013. doi : 10.1109/SURV.2012.040412.00105. URL <https://doi.org/10.1109/SURV.2012.040412.00105>. 57
- IEEE. *IEEE 802.15.4-2015 - IEEE Standard for Low-Rate Wireless Networks*. IEEE Standards Office, New York, NY, USA, 2015. 49
- Ozlem Durmaz Incel and Bhaskar Krishnamachari. Enhancing the data collection rate of tree-based aggregation in wireless sensor networks. In *Sensor, Mesh and Ad Hoc Communications and Networks, 2008. SECON'08. 5th Annual IEEE Communications Society Conference on*, pages 569–577. IEEE, 2008. 68
- Oana Iova, Fabrice Theoleyre, and Thomas Noel. Improving the network lifetime with energy-balancing routing : Application to rpl. In *7th IFIP Wireless and Mobile Networking Conference*, pages 1–8. IEEE, 2014. 122, 125, 129
- Oana-Teodora Iova. *Standards optimization and network lifetime maximization for wireless*



## BIBLIOGRAPHIE

---

- sensor networks in the Internet of things*. Theses, Université de Strasbourg, 2014. URL <https://tel.archives-ouvertes.fr/tel-01275066>. 120
- Tommy R. Jensen and Bjarne Toft. *Graph Coloring Problems*. Wiley, 2011. ISBN 9780471028659. 71, 72
- David Johnson, Yin-chun Hu, and David Maltz. The dynamic source routing protocol (dsr) for mobile ad hoc networks for ipv4. RFC, Internet Engineering Task Force (IETF), 2007. URL <https://www.rfc-editor.org/rfc/rfc4728.txt>. 61
- Patrick Olivier Kamgueu, Emmanuel Nataf, Thomas Ndié Djotio, and Olivier Festor. Energy-based metric for the routing protocol in low-power and lossy network. In *2nd International Conference on Sensor Networks*, pages 145–148, 2013. 121, 125, 129, 141, 142, 152, 173
- Panagiotis Karkazis, Helen-Catherine Leligou, Lambros Sarakis, Theodore Zahariadis, Panagiotis Trakadas, Terpsichori Helen Velivassaki, and Christos Capsalis. Design of primary and composite routing metrics for rpl-compliant wireless sensor networks. In *Telecommunications and Multimedia (TEMU), 2012 International Conference on*, pages 13–18. IEEE, 2012. 118, 153
- Panagiotis Karkazis, Panagiotis Trakadas, Helen-Catherine Leligou, Lambros Sarakis, Ioannis Papaefstathiou, and Theodore B. Zahariadis. Evaluating routing metric composition approaches for qos differentiation in low power and lossy networks. *Wireless Networks*, 19(6) :1269–1284, 2013. doi : 10.1007/s11276-012-0532-2. URL <http://dx.doi.org/10.1007/s11276-012-0532-2>. 158
- Holger Karl and Andreas Willig. *Protocols and architectures for wireless sensor networks*. John Wiley & Sons, 2007. 44
- Mounib Khanafer, Mouhcine Guennoun, and Hussein T. Mouftah. A survey of beacon-enabled IEEE 802.15.4 MAC protocols in wireless sensor networks. *IEEE Communications Surveys and Tutorials*, 16(2) :856–876, 2014. doi : 10.1109/SURV.2013.112613.00094. URL <https://doi.org/10.1109/SURV.2013.112613.00094>. 49

## BIBLIOGRAPHIE

---

- N. Kulkarni, R. Prasad, H. Cornean, and N. Gupta. Performance evaluation of aodv, dsdv and dsr for quasi random deployment of sensor nodes in wireless sensor networks. In *2011 International Conference on Devices and Communications (ICDeCom)*, pages 1–5, 2011. doi : 10.1109/ICDECOM.2011.5738571. 63
- Hanane Lamaazi, Nabil Benamar, and Antonio J. Jara. Rpl-based networks in static and mobile environment : A performance assessment analysis. *Journal of King Saud University - Computer and Information Sciences*, 2017. ISSN 1319-1578. doi : <https://doi.org/10.1016/j.jksuci.2017.04.001>. URL <http://www.sciencedirect.com/science/article/pii/S1319157816301574>. 177
- Mads Lauridsen, Huan Nguyen, Benny Vejlgaard, István Z. Kovács, Preben Mogensen, and Mads Sorensen. Coverage comparison of gprs, nb-iot, lora, and sigfox in a 7800 km<sup>2</sup> area. In *85th IEEE Vehicular Technology Conference, VTC Spring 2017, Sydney, Australia, June 4-7, 2017*, pages 1–5, 2017a. doi : 10.1109/VTCSpring.2017.8108182. URL <https://doi.org/10.1109/VTCSpring.2017.8108182>. 34
- Mads Lauridsen, Benny Vejlgaard, István Z. Kovács, Huan Nguyen, and Preben E. Mogensen. Interference measurements in the european 868 mhz ISM band with focus on lora and sigfox. In *2017 IEEE Wireless Communications and Networking Conference, WCNC 2017, San Francisco, CA, USA, March 19-22, 2017*, pages 1–6, 2017b. doi : 10.1109/WCNC.2017.7925650. URL <https://doi.org/10.1109/WCNC.2017.7925650>. 34
- Chia-Lin Lee and Tzong-Jye Liu. A self-stabilizing distance-2 edge coloring algorithm. *The Computer Journal*, 57(11) :1639–1648, 2014. doi : 10.1093/comjnl/bxt072. URL <https://doi.org/10.1093/comjnl/bxt072>. 102
- Keontaek Lee, Hak-Jin Kim, Sunju Park, and Seungjae Han. Satisfying the target network lifetime in wireless sensor networks. *Computer Networks*, 65 :41 – 55, 2014. URL <http://www.sciencedirect.com/science/article/pii/S1389128614000942>. 31
- Junchao Ma, Wei Lou, Yanwei Wu, Xiang-Yang Li, and Guihai Chen. Energy efficient TDMA sleep scheduling in wireless sensor networks. In *28th IEEE International Confe-*

## BIBLIOGRAPHIE

---

- rence on Computer Communications (INFOCOM)*, pages 630–638, 2009. doi : 10.1109/INFOCOM.2009.5061970. URL <https://doi.org/10.1109/INFOCOM.2009.5061970>. 81, 85
- Mohammad Mahdian. On the computational complexity of strong edge coloring. *Discrete Applied Mathematics*, 118(3) :239–248, 2002. doi : 10.1016/S0166-218X(01)00237-2. URL [http://dx.doi.org/10.1016/S0166-218X\(01\)00237-2](http://dx.doi.org/10.1016/S0166-218X(01)00237-2). 78
- Saucene Mahfoudh, Pascale Minet, and Ichrak Amdouni. Energy efficient routing and node activity scheduling in the OCARI wireless sensor network. *Future Internet*, 2(3) : 308–340, 2010. doi : 10.3390/fi2030308. URL <https://doi.org/10.3390/fi2030308>. 82, 83, 84, 85
- Toshimitsu Masuzawa and Sébastien Tixeuil. Stabilizing link-coloration of arbitrary networks with unbounded byzantine faults. *International Journal of Principles and Applications of Information Science and Technology*, 1(1) :1–13, 2007. 101
- Leonid S. Melnikov and Vadym G. Vizing. The edge chromatic number of a directed/mixed multigraph. *Journal of Graph Theory*, 31(4) :267–273, 1999. doi : 10.1002/(SICI)1097-0118(199908)31:4<267::AID-JGT1>3.0.CO;2-D. URL [http://dx.doi.org/10.1002/\(SICI\)1097-0118\(199908\)31:4<267::AID-JGT1>3.0.CO;2-D](http://dx.doi.org/10.1002/(SICI)1097-0118(199908)31:4<267::AID-JGT1>3.0.CO;2-D). 71
- Gabriel Montenegro, Nandakishore Kushalnagar, J Hui, and D Culler. Transmission of IPv6 packets over IEEE 802.15.4 networks. *Internet proposed standard RFC*, 4944, 2007. 33
- T Narten, E Nordmark, and W Simpson. Neighbor discovery for ip version 6 (ipv6). *RFC 4861*, 2007. 33
- Emmanuel Nataf and Olivier Festor. Accurate online estimation of battery lifetime for wireless sensors network. In *2nd International Conference on Sensor Networks*, pages 59–64, 2013. 130
- Fredrik Osterlind, Adam Dunkels, Joakim Eriksson, Niclas Finne, and Thiemo Voigt. Cross-level sensor network simulation with cooja. In *31st IEEE conference on Local computer networks*, pages 641–648, 2006. 125

## BIBLIOGRAPHIE

---

- Charles Perkins, Elizabeth Belding-Royer, and Samir Das. Ad hoc on-demand distance vector (aodv) routing. Technical report, Internet Engineering Task Force (IETF), 2003. URL <https://tools.ietf.org/html/draft-gnawali-roll-etxof-00>. 62
- Charles E Perkins and Pravin Bhagwat. Highly dynamic destination-sequenced distance-vector routing (dsdv) for mobile computers. In *ACM SIGCOMM computer communication review*, volume 24, pages 234–244. ACM, 1994. 59
- Levis Philip and Clausen Thomas Heide. The trickle algorithm. (*IETF*) *Internet Engineering Task Force*, 2011. 113
- Joseph Polastre, Jason L. Hill, and David E. Culler. Versatile low power media access for wireless sensor networks. In *Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems, SenSys 2004, Baltimore, MD, USA, November 3-5, 2004*, pages 95–107, 2004. doi : 10.1145/1031495.1031508. URL <http://doi.acm.org/10.1145/1031495.1031508>. 51, 52
- Joseph Polastre, Robert Szewczyk, and David E. Culler. Telos : enabling ultra-low power wireless research. In *Fourth International Symposium on Information Processing in Sensor Networks (IPSN)*, pages 364–369, 2005. doi : 10.1109/IPSN.2005.1440950. URL <http://dx.doi.org/10.1109/IPSN.2005.1440950>. 125
- Mamoun Qasem, Hussien Altawssi, Muneer Bani Yassien, and Ahmed Yassin AlDubai. Performance evaluation of RPL objective functions. In *15th IEEE Int. Conf. on Computer and Information Technology*, pages 1606–1613, 2015. 115
- Venkatesh Rajendran, Katia Obraczka, and J. J. Garcia-Luna-Aceves. Energy-efficient collision-free medium access control for wireless sensor networks. In *Proceedings of the 1st International Conference on Embedded Networked Sensor Systems, SenSys 2003*, pages 181–192, 2003. doi : 10.1145/958491.958513. URL <http://doi.acm.org/10.1145/958491.958513>. 53
- Venkatesh Rajendran, Katia Obraczka, and J. J. Garcia-Luna-Aceves. Energy-efficient, collision-free medium access control for wireless sensor networks. *Wireless Networks*, 12

## BIBLIOGRAPHIE

---

- (1) :63–78, 2006. doi : 10.1007/s11276-006-6151-z. URL <http://dx.doi.org/10.1007/s11276-006-6151-z>. 53, 54, 68
- Priyanka Rawat, Kamal Deep Singh, Hakima Chaouchi, and Jean-Marie Bonnin. Wireless sensor networks : a survey on recent developments and potential synergies. *The Journal of Supercomputing*, 68(1) :1–48, 2014a. doi : 10.1007/s11227-013-1021-9. URL <https://doi.org/10.1007/s11227-013-1021-9>. 25
- Priyanka Rawat, Kamal Deep Singh, Hakima Chaouchi, and Jean Marie Bonnin. Wireless sensor networks : a survey on recent developments and potential synergies. *The Journal of supercomputing*, 68(1) :1–48, 2014b. 18, 26
- Injong Rhee, Ajit Warriar, Mahesh Aia, Jeongki Min, and Mihail L. Sichitiu. Z-MAC : a hybrid MAC for wireless sensor networks. *IEEE/ACM Trans. Netw.*, 16(3) :511–524, 2008. doi : 10.1145/1399562.1399564. URL <http://doi.acm.org/10.1145/1399562.1399564>. 14, 55, 56
- Fatma Soma, Inès El Korbi, and Leïla Azouz Saïdane. Mobility support over RPL using sensor nodes speed classification. In *13th IEEE/ACS International Conference of Computer Systems and Applications, AICCSA 2016, Agadir, Morocco, November 29 - December 2, 2016*, pages 1–6, 2016. doi : 10.1109/AICCSA.2016.7945688. URL <https://doi.org/10.1109/AICCSA.2016.7945688>. 177
- Ridha Soua, Pascale Minet, and Erwan Livolant. Disca : A distributed scheduling for convergecast in multichannel wireless sensor networks. In *Integrated Network Management (IM), 2015 IFIP/IEEE International Symposium on*, pages 156–164. IEEE, 2015. 83, 84, 85
- Craig Thomson, Isam Wadhaj, Imed Romdhani, and Ahmed Al-Dubai. Performance evaluation of rpl metrics in environments with strained transmission ranges. *Computer Systems and Applications (AICCSA), 2016 IEEE/ACS 13th International Conference*, 2016. 152
- Chi-Hung Tzeng, Jehn-Ruey Jiang, and Shing-Tsaan Huang. A self-stabilizing (delta+4)-edge-coloring algorithm for planar graphs in anonymous uniform systems. *Information*

## BIBLIOGRAPHIE

---

- Processing Letters*, 101(4) :168–173, 2007. doi : 10.1016/j.ipl.2006.09.004. URL <https://doi.org/10.1016/j.ipl.2006.09.004>. 102
- Thierry Val, Eric Campo, and Adrien Van Den Bossche. Technologie zigbee / 802.15.4 : Protocoles, topologies et domaines d’application. *Techniques de l’Ingenieur*, TE 7508 : 8, 2008. 31
- JP. Vasseur, Cisco Systems, and M. Kim. Rpl : Ipv6 routing protocol for low-power and lossy networks. *Internet Engineering Task Force*, March 2012. 16, 111, 112, 116, 125
- Malisa Vucinic, Bernard Tourancheau, and Andrzej Duda. Performance comparison of the RPL and loadng routing protocols in a home automation scenario. In *2013 IEEE Wireless Communications and Networking Conference (WCNC), Shanghai, Shanghai, China, April 7-10, 2013*, pages 1974–1979, 2013. doi : 10.1109/WCNC.2013.6554867. URL <https://doi.org/10.1109/WCNC.2013.6554867>. 64
- T. Winter, P. Thubert, A. Brandt, J. Hui, and Kelsey. Rpl : Ipv6 routing protocol for low-power and lossy networks. RFC 6550, Internet Engineering Task Force (IETF), 2012. URL <https://tools.ietf.org/html/rfc6550>. 61, 109, 126
- T. Winter, P. Thubert, and Cisco Systems. Routing metrics used for path calculation in low power and lossy networks. *Internet Engineering Task Force*, March 2012. 116
- Gen Xu and Gang Lu. Multipath routing protocol for DAG-based WSNs with mobile sinks. *2nd International Conference on Computer Science and Electronics Engineering (ICCSEE 2013)*, 2013. 121
- Tao Yang, Makoto Ikeda, Giuseppe De Marco, and Leonard Barolli. Performance behavior of aodv, DSR and DSDV protocols for different radio models in ad-hoc sensor networks. In *2007 International Conference on Parallel Processing Workshops (ICPP Workshops 2007), 10-14 September 2007, Xi-An, China*, page 6, 2007. doi : 10.1109/ICPPW.2007.67. URL <https://doi.org/10.1109/ICPPW.2007.67>. 63
- Tao Yang, Leonard Barolli, Makoto Ikeda, Fatos Xhafa, and Arjan Duresi. Performance analysis of OLSR protocol for wireless sensor networks and comparison evaluation with

- AODV protocol. In *NBiS 2009, 12th International Conference on Network-Based Information Systems, Indianapolis, Indiana, USA, 19-21 August 2009*, pages 335–342, 2009. doi : 10.1109/NBiS.2009.35. URL <https://doi.org/10.1109/NBiS.2009.35>. 63
- Yaling Yang and Jun Wang. Design guidelines for routing metrics in multihop wireless networks. In *INFOCOM 2008. The 27th conference on computer communications. IEEE*, pages 1615–1623. IEEE, 2008. 17, 155, 156, 157
- Wei Ye, John S. Heidemann, and Deborah Estrin. An energy-efficient MAC protocol for wireless sensor networks. In *Proceedings IEEE INFOCOM 2002, The 21st Annual Joint Conference of the IEEE Computer and Communications Societies, New York, USA, 2002*. URL <http://www.ieee-infocom.org/2002/papers/947.pdf>. 49
- Jennifer Yick, Biswanath Mukherjee, and Dipak Ghosal. Wireless sensor network survey. *Computer networks*, 52(12) :2292–2330, 2008. 25
- Hongseok Yoo, Moonjoo Shim, Dongkyun Kim, and Kyu Hyung Kim. Global : A gradient-based routing protocol for load-balancing in large-scale wireless sensor networks with multiple sinks. In *Computers and Communications (ISCC), 2010 IEEE Symposium on*, pages 556–562. IEEE, 2010. 31
- Theodore Zahariadis and Panagiotis Trakadas. Design Guidelines for Routing Metrics Composition in LLN. Technical report, Internet Engineering Task Force, 2012. URL <https://tools.ietf.org/pdf/draft-zahariadis-roll-metrics-composition-04.pdf>. 153, 154, 157
- Andrea Zanella, Nicola Bui, Angelo Castellani, Lorenzo Vangelista, and Michele Zorzi. Internet of things for smart cities. *IEEE Internet of Things journal*, 1(1) :22–32, 2014. 18
- Qun Zhao and Mohan Gurusamy. Lifetime maximization for connected target coverage in wireless sensor networks. *IEEE/ACM Transactions on Networking (TON)*, 16 :1378–1391, 2008. 31

# Annexes





# Preuve de correction de l'algorithme SS-DD2EC

Dans cette partie, nous présentons quelques preuves de correction de l'algorithme SS-DD2EC décrit formellement dans les Algorithmes 2 et 3. Nous commençons par spécifier une coloration orientée d'arêtes auto-stabilisante.

## Spécification 1 (Coloration orientée d'arêtes auto-stabilisante)

Soit  $\mathcal{S}$  l'ensemble des configurations possibles du système. Un algorithme  $\mathcal{A}$  calculant une coloration orientée d'arêtes auto-stabilisantes satisfait les conditions suivantes :

1. L'algorithme  $\mathcal{A}$  atteint un ensemble de configurations terminales  $\mathcal{T} \subseteq \mathcal{S}$  en temps fini,
2. Toute configuration  $\gamma \in \mathcal{T}$  satisfait le Problème 1.

Nous introduisons quelques concepts pour définir une configuration légitime pour la Spécification 1. Dans la suite, on notera  $G = (V, E)$  le graphe modélisant le réseau et  $T = (V, E_T)$  un arbre couvrant de  $G$  enraciné au puits  $s$ . On notera  $NI(p)$  l'ensemble des voisins en conflits de  $p$ , c'est-à-dire que  $NI(p) = Neig_p \setminus (Child_p \cup \{par_p\})$ .

## Définition 1 (Priorité des arcs d'un arbre)

Soit  $T = (V, E_T)$  un arbre couvrant enraciné au puits  $s$  du graphe  $G = (V, E)$ . La priorité associée à un arc  $e = (p, q) \in E_T$  est définie par le tuple  $((d_T(q, s), ID_q), (d_T(p, s), ID_p))$ , avec  $d_T(x, s)$  la distance du nœud  $x$  au puits  $s$  dans  $T$ . Étant donné deux arcs  $e = (p, q), e' = (p', q') \in E_T$ , on dira que  $e$  possède une priorité plus importante que  $e'$ , noté  $prio(e) \prec prio(e')$ , ssi soit  $d_T(q, s) < d_T(q', s)$  soit  $d_T(q, s) = d_T(q', s)$  et  $ID_q < ID_{q'}$ .

**Définition 2 (Couleur correcte)**

Soit  $\mathcal{C}$  la palette de couleurs sur chaque nœud  $p \in V$ . Chaque arc  $(p, q) \in E_T$  possède une couleur correcte si les conditions suivantes sont satisfaites pour  $p$  et  $q$  :

1.  $acr_q \cap \{ce_x : (x, y) \in E_T \wedge x \in \{q, par_q\}\} = \emptyset$ ,
2.  $acr_q \cap \{ce_x : (x, y) \in E_T \wedge x \in NI(q) \wedge prio(x, y) \prec prio(p, q)\} = \emptyset$ ,
3.  $ace_p \cap \{ce_x : (x, y) \in E_T \wedge y \in NI(p) \wedge prio(x, y) \prec prio(p, q)\} = \emptyset$ ,
4.  $(\exists(i, c) \in cr_q : i = p \wedge c \in (acr_q \cap ace_p) \subseteq \mathcal{C} \wedge [\forall(j, c') \in cr_q : (j = p \wedge c' = c) \vee c' \neq c])$ ,
5.  $(p, ce_p) \in cr_q$ .

**Définition 3 (Configuration légitime)**

Soit  $\mathcal{S}$  l'ensemble des configurations du système. Une configuration  $\gamma \in \mathcal{S}$  est dite légitime pour l'algorithme SS-DD2EC si pour chaque arc  $(p, q) \in E_T$  la Définition 2 est satisfaite.

Dans la suite, nous considérons que les entrées fournies à l'algorithme SS-DD2EC sont correctes. Nous commençons par montrer dans le Théorème 2 que dans toute configuration illégitime du système il existe un nœud qui dispose d'une action de l'algorithme SS-DD2EC activable.

**Remarque 1**

Au niveau de chaque nœud  $p \in V - \{r\}$ , la fonction  $ChildColor()$  retourne un ensemble ordonné de paires satisfaisant les propriétés suivantes  $\forall q \in Child_p$  :

1.  $|\{(i, c) \in ChildColor() : i = q\}| = 1$ ,
2.  $(\exists(i, c) \in ChildColor() : i = q \wedge c \in (acr_p \cap ace_q) \subseteq \mathcal{C} \wedge [\forall(j, c') \in ChildColor() : (j = q \wedge c' = c) \vee c' \neq c])$ .

**Théorème 2**

Soit l'ensemble de configurations  $\mathcal{B} \subset \mathcal{S}$  tel que chaque configuration  $\gamma \in \mathcal{B}$  satisfait la Définition 3.  $\forall \gamma \in (\mathcal{S} - \mathcal{B})$ ,  $\exists p \in V$  tel que  $p$  est activable dans  $\gamma$ .

**Preuve.** Supposons, par contradiction, que  $\exists \gamma \in (\mathcal{S} - \mathcal{B})$  tel que  $\forall p \in V$  aucune action de l'Algorithme SS-DD2EC n'est activable dans  $\gamma$ . D'après la Définition 3, cela implique qu'il existe un arc  $(p, q) \in E_T$  pour lequel l'une des conditions de la Définition 2 n'est pas satisfaite.

Considérons un arc  $(p, q) \in E_T$  de plus forte priorité dans  $E_T$  tel que la Définition 2 n'est pas satisfaite pour  $(p, q)$  dans  $\gamma$ . Cela implique que nous supposons les Prédicats  $Good\_ACE(q)$  et  $Good\_CE(q)$  satisfait pour le nœud  $q$  dans  $\gamma$ .

Si la Condition 3 de la Définition 2 n'est pas satisfaite dans  $\gamma$ , cela implique que l'ensemble des couleurs autorisées en émission pour  $p$  contient une couleur déjà utilisée en réception par un voisin  $y$  en conflit avec  $p$  pour l'arc  $(x, y)$  de priorité plus importante que l'arc  $(p, q)$ , c.-à-d.,  $(\exists c \in ace_p : ce_x = c \wedge (x, y) \in E_T \wedge y \in R_I(p) \wedge prio(x, y) \prec prio(p, q))$ . D'après la description formelle de l'Algorithme SS-DD2EC, la Fonction  $ColorR(cr_w)$  renvoie l'ensemble des couleurs utilisées en réception par les voisins  $w \in R_I(p)$  de  $p$ . L'Algorithme SS-DD2EC suit la priorité des arcs de  $E_T$  pour l'affectation des couleurs, cela est pris en compte par la Macro  $PrioE(p)$  sur  $p$  pour l'arc  $(p, q)$ . Nous avons donc :  $ce_x \in (\bigcup_{z \in PrioE(p)} ColorR(cr_z))$  car  $prio((x, y)) \prec prio((p, q))$ . Il y a donc une contradiction car le Prédicat  $Good\_ACE(p)$  n'est pas satisfait et la règle  $R0$  est activable sur  $p$  puisque  $ace_p \neq \mathcal{C} \setminus (\bigcup_{z \in PrioE(p)} ColorR(cr_z))$ . Nous considérons donc dans la suite que  $Good\_ACE(p)$  est satisfait par  $p$  dans  $\gamma$ .

Si la Condition 1 ou 2 de la Définition 2 n'est pas satisfaite dans  $\gamma$ , cela implique que l'ensemble des couleurs autorisées en réception pour  $q$  contient une couleur utilisée en émission soit par  $q$ , soit par le père de  $q$ , soit par un voisin  $x$  en conflit avec  $q$  et dont l'arc  $(x, y)$  est de plus forte priorité, c.-à-d.,  $(\exists c \in acr_q : ce_x = c \wedge (x, y) \in E_T \wedge x \in (R_I(q) \cup \{q, par_q\}) \wedge prio(x, y) \prec prio(p, q))$ . L'Algorithme SS-DD2EC suit la priorité des arcs de  $E_T$  pour l'affectation des couleurs aux arcs, cela est pris en compte par la Macro  $PrioR(q)$  sur  $q$  pour l'arc  $(p, q)$ . Nous avons donc que :  $ce_x \in \{ce_z : z \in (PrioR(q) \cup \{q, par_q\})\}$  comme  $prio((x, y)) \prec prio((p, q))$ . Par conséquent, il y a une contradiction car le Prédicat  $Good\_ACR(q)$  n'est pas satisfait et la règle  $R2$  est activable par  $q$  comme  $acr_q \neq \mathcal{C} \setminus \{ce_z : z \in (PrioR(q) \cup \{q, par_q\})\}$ . Nous considérons donc dans la suite que  $Good\_ACR(q)$  est satisfait par  $q$  dans  $\gamma$ . On peut remarquer que les Conditions 1 et 2 de la Définition 2 ne concernent pas le cas particulier du puits (soit si  $q = s$ ) comme  $s$  ne possède pas de variable  $ace_s$ .

Si la Condition 4 de la Définition 2 n'est pas satisfaite dans  $\gamma$ , cela implique que soit  $cr_q$  ne contient aucun couple contenant une couleur pour  $p$ , soit la couleur attribuée à  $p$  par  $q$

## BIBLIOGRAPHIE

---

est déjà attribuée à un autre fils  $z$  de  $q$ , soit la couleur attribuée à  $p$  par  $q$  n'appartient ni à l'ensemble de couleurs possibles  $\mathcal{C}$  ni à  $(acr_q \cap ace_p)$  (c.-à-d., on a  $(\forall(i, c) \in cr_q : i \neq p \vee c \notin (acr_q \cap ace_p)) \vee c \notin \mathcal{C} \vee [\exists(z, c') \in cr_q : z \neq p \wedge c' = c]$ ). D'après la Remarque 1, la Fonction  $ChildColor()$  renvoie un ensemble de couples associant un couple  $(i, c)$  pour chaque fils  $p$  de  $q$  tel que  $(\exists(i, c) \in ChildColor() : i = p \wedge c \in (acr_a \cap ace_p) \subseteq \mathcal{C} \wedge [\forall(j, c') \in ChildColor() : j \neq p \Rightarrow c' \neq c])$ . Il y a donc une contradiction car  $cr_q \neq ChildColor()$  et la règle  $R3$  est activable par  $q$  comme  $Good\_CR(q)$  n'est pas satisfait. Nous considérons donc dans la suite que  $Good\_CR(q)$  est satisfait par  $q$ . On peut remarquer que les arguments ci-dessus s'appliquent également au cas où  $q$  est le puits (soit si  $q = s$ ), en considérant la règle  $R0$  au lieu de la règle  $R3$  et que la Fonction  $ChildColor()$  sur  $s$  retourne un ensemble de couples  $(i, c)$  pour chaque fils  $i$  de  $s$  tel que  $(\exists(i, c) \in ChildColor() : i = p \wedge c \in \mathcal{C} \wedge [\forall(j, c') \in ChildColor() : j \neq p \Rightarrow c' \neq c])$ .

Si la Condition 5 de la Définition 2 n'est pas satisfaite dans  $\gamma$ , cela implique que la variable  $ce_p$  contient une couleur en émission différente de la couleur en émission attribuée à  $p$  par  $q$  (c.-à-d., on a  $(\forall(i, c) \in cr_q : i = p \wedge c \neq ce_p)$ ). D'après la Remarque 1, la Fonction  $ChildColor()$  retourne un ensemble contenant un unique couple  $(p, c)$  pour  $p$ . De plus, d'après la description formelle de l'Algorithme SS-DD2EC la Fonction  $ColorCE()$  exécutée par  $q$  attribue une couleur en émission à  $p$ . Par conséquent, il y a une contradiction car nous avons  $ce_p \neq ColorCE()$  et la règle  $R1$  est activable par  $p$  comme  $Good\_CE(p)$  n'est pas satisfait.  $\square$

Dans la suite, nous noterons  $\mathcal{S}_1 \subseteq V - \{r\}$  l'ensemble des nœuds dans  $\gamma \in \mathcal{S}$  pour lesquels l'ensemble de couleurs autorisées en émission est correct, c.-à-d., on a  $\mathcal{S}_1 = \{p \in V - \{s\} : Good\_ACE(p)\}$ .

### Lemme 4

Lorsque la règle  $R0$  est activable par  $p \in V - \{s\}, p \notin \mathcal{S}_1$ , alors elle le reste jusqu'à ce que  $p$  active  $R0$  ou que  $p \in \mathcal{S}_1$ .

**Preuve.** Soit  $\gamma \mapsto \gamma'$  un pas de calcul. Supposons, par contradiction, que la règle  $R0$  est activable par  $p \in (V - \mathcal{S}_1)$  dans  $\gamma$  et non activable par  $p$  dans la configuration  $\gamma'$  (c.-à-d.,  $ace_p = \mathcal{C} \setminus (\bigcup_{w \in PrioE(p)} ColorR(cr_w))$  dans  $\gamma'$ ) alors que  $p$  n'a pas activé la règle

$R0$  durant le pas  $\gamma \mapsto \gamma'$ . D'après l'hypothèse du lemme, nous supposons que  $p \in (V - \mathcal{S}_1)$  dans  $\gamma'$ , donc on a  $\neg Good\_ACE(p)$  qui est satisfait dans  $\gamma'$ . Comme  $p$  n'a pas activé de règle durant  $\gamma \mapsto \gamma'$  et que sa variable  $ace_p$  peut uniquement être modifiée par  $p$  en activant la règle  $R0$ , cela conduit à une contradiction. La règle  $R0$  est donc toujours activable par  $p$  dans  $\gamma'$ .  $\square$

**Lemme 5**

Dans toute configuration  $\gamma \in (\mathcal{S} - \mathcal{B})$ ,  $p \in V - \{s\}$  met à jour son ensemble de couleurs autorisées en émission si la Condition 3 de la Définition 2 n'est pas satisfaite pour l'arc  $(p, q) \in E_T$ .

**Preuve.** D'après la description formelle de l'Algorithme SS-DD2EC,  $p \in V - \{s\}$  doit activer la règle  $R0$  pour mettre à jour son ensemble de couleurs autorisées (c.-à-d., sa variable  $ace_p$ ). Comme  $(p, q)$  ne satisfait pas la Condition 3 de la Définition 2, cela implique qu'il y a une couleur  $c \in ace_p$  telle que  $c \in \{ce_x : (x, y) \in E_T \wedge y \in R_I(p) \wedge prio(x, y) < prio(p, q)\}$ .

Supposons, par contradiction, que  $p$  ne met pas à jour son ensemble de couleurs autorisées en émission. Cela implique soit que la règle  $R0$  est non activable par  $p$  dans  $\gamma$ , mais en contradiction avec la preuve du Théorème 2 (Condition 3 de la Définition 2 non satisfaite), ou ce n'est pas la règle activable par  $p$  la plus prioritaire dans  $\gamma$ , mais en contradiction avec la description formelle de l'Algorithme SS-DD2EC où  $R0$  est la règle la plus prioritaire pour  $p$ . Enfin, d'après le Lemme 4 la règle  $R0$  activable par  $p \notin \mathcal{S}_1$  jusqu'à son activation.  $\square$

Dans la suite, on notera  $\mathcal{S}_2 \subseteq \mathcal{S}_1$  l'ensemble des nœuds  $p \in \mathcal{S}_1$  dans  $\gamma \in \mathcal{S}$  pour lesquels la couleur en émission attribuée à  $p$  est correcte, c.-à-d., on a  $\mathcal{S}_2 = \{p \in \mathcal{S}_1 : Good\_CE(p)\}$ .

**Lemme 6**

Lorsque la règle  $R1$  est activable par  $p \in \mathcal{S}_1, p \notin \mathcal{S}_2$ , alors elle le reste jusqu'à ce que  $p$  active  $R1$  ou que  $p \in \mathcal{S}_2$ .

**Preuve.** Soit  $\gamma \mapsto \gamma'$  un pas de calcul. Supposons, par contradiction, que la règle  $R1$  est activable par  $p \in (\mathcal{S}_1 - \mathcal{S}_2)$  dans  $\gamma$  et non activable dans  $\gamma'$  (c.-à-d.,  $ce_p = ColorCE()$ )

pour  $p$  dans  $\gamma'$ ) alors que  $p$  n'a pas activé la règle  $R1$  durant le pas  $\gamma \mapsto \gamma'$ . D'après l'hypothèse du lemme, on suppose que  $p \in (\mathcal{S}_1 - \mathcal{S}_2)$  dans  $\gamma'$ , donc on a  $\neg Good\_CE(p)$  dans  $\gamma'$ . Comme  $p$  n'a pas activé de règle durant  $\gamma \mapsto \gamma'$  et que la variable  $ce_p$  peut uniquement être modifiée localement par  $p$  en activant la règle  $R1$ , alors  $R1$  est activable par  $p$  dans  $\gamma'$ . Il y a donc contradiction et la règle  $R1$  est donc toujours activable par  $p$  dans  $\gamma'$ .  $\square$

**Lemme 7**

*Dans toute configuration  $\gamma \in (\mathcal{S} - \mathcal{B})$ ,  $p \in \mathcal{S}_1$  met à jour sa couleur en émission si la Condition 5 de la Définition 2 n'est pas satisfaite pour l'arc  $(p, q) \in E_T$ .*

**Preuve.** D'après la description formelle de l'Algorithme SS-DD2EC, un nœud  $p \in \mathcal{S}_1$  active la règle  $R1$  pour mettre à jour sa couleur en émission (c.-à-d., sa variable  $ce_p$ ). Comme l'arc  $(p, q)$  ne satisfait pas la Condition 5 de Définition 2, on a  $(\forall (i, c) \in cr_q : i = p \wedge c \neq ce_p)$ . Supposons, par contradiction, que  $p$  ne met pas à jour sa couleur en émission. Cela implique soit que la règle  $R1$  n'est pas activable par  $p$  in  $\gamma$ , mais en contradiction avec le Théorème 2 (Condition 5 de la Définition 2 non satisfaite), ou que ce n'est pas la règle activable par  $p$  de plus forte priorité dans  $\gamma$ , mais en contradiction avec le fait que  $R0$  est non activable par  $p$  dans  $\gamma$  car  $(p \in \mathcal{S}_1) \Rightarrow Good\_ACE(p)$ . Enfin, d'après le Lemme 6 la règle  $R1$  est activable par  $p \in (\mathcal{S}_1 - \mathcal{S}_2)$  jusqu'à son activation.  $\square$

Dans la suite, nous noterons  $\mathcal{S}_3 \subseteq \mathcal{S}_2$  l'ensemble des nœuds dans  $\gamma \in \mathcal{S}$  pour lesquels l'ensemble des couleurs autorisées en réception est correct, c.-à-d., on a  $\mathcal{S}_3 = \{p \in \mathcal{S}_2 : Good\_ACR(p)\}$ .

**Lemme 8**

*Lorsque la règle  $R2$  est activable par  $p \in \mathcal{S}_2, p \notin \mathcal{S}_3$ , alors elle le reste jusqu'à ce que  $p$  active  $R2$  ou que  $p \in \mathcal{S}_3$ .*

**Preuve.** Soit  $\gamma \mapsto \gamma'$  un pas de calcul. Supposons, par contradiction, que la règle  $R2$  est activable par  $p \in (\mathcal{S}_2 - \mathcal{S}_3)$  dans  $\gamma$  et non activable dans  $\gamma'$  (c.-à-d.,  $acr_p = \mathcal{C} \setminus \{ce_x : x \in (PrioR(p) \cup \{p, par_p\})\}$  dans  $\gamma'$ ) alors que  $p$  n'a pas activé la règle  $R2$  durant le pas  $\gamma \mapsto \gamma'$ . D'après l'hypothèse du lemme, on suppose que  $p \in (\mathcal{S}_2 - \mathcal{S}_3)$  dans  $\gamma'$ , donc on

a  $Good\_ACE(p) \wedge Good\_CE(p) \wedge \neg Good\_ACR(p)$  dans  $\gamma'$ . Comme  $p$  n'a pas activé de règle durant  $\gamma \mapsto \gamma'$  et que sa variable  $acr_p$  peut uniquement être modifiée localement par  $p$  en activant la règle  $R2$ , alors  $R2$  est activable par  $p$  dans  $\gamma'$ . Il y a donc contradiction et la règle  $R2$  est donc toujours activable par  $p$  dans  $\gamma'$ .  $\square$

**Lemme 9**

*Dans toute configuration  $\gamma \in (\mathcal{S}-\mathcal{B})$ ,  $q \in \mathcal{S}_2$  met à jour son ensemble de couleurs autorisées en réception si la Condition 1 ou 2 de la Définition 2 n'est pas satisfaite pour l'arc  $(p, q) \in E_T$ .*

**Preuve.** D'après la description formelle de l'Algorithme SS-DD2EC, un nœud  $q \in \mathcal{S}_2$  active la règle  $R2$  pour mettre à jour son ensemble de couleurs autorisées en réception (c.-à-d., sa variable  $acr_q$ ). Comme l'arc  $(p, q)$  ne satisfait pas la Condition 1 ou 2 de la Définition 2, il y a une couleur  $c \in acr_q$  telle que  $c \in \{ce_x : (x, y) \in E_T \wedge x \in R_I(q) \cup \{q, par_q\} \wedge prio(x, y) \prec prio(p, q)\}$ .

Supposons, par contradiction, que  $q$  ne met pas à jour son ensemble de couleurs autorisées en réception. Cela implique soit que la règle  $R2$  est non activable par  $q$  dans  $\gamma$ , mais en contradiction avec le Théorème 2 (Condition 1 ou 2 de la Définition 2 non satisfaite), ou que ce n'est pas la règle activable par  $p$  de plus forte priorité dans  $\gamma$ , mais en contradiction avec le fait que les règles  $R0$  et  $R1$  sont non activables par  $q$  dans  $\gamma$  car  $(q \in \mathcal{S}_2) \Rightarrow (Good\_ACE(q) \wedge Good\_CE(q))$ . Enfin, d'après le Lemme 8 la règle  $R2$  est toujours activable par  $q \in (\mathcal{S}_2 - \mathcal{S}_3)$  jusqu'à son activation.  $\square$

Dans la suite, nous noterons  $\mathcal{S}_4 \subseteq \mathcal{S}_3$  l'ensemble des nœuds dans  $\gamma \in \mathcal{S}$  pour lesquels l'ensemble des couleurs en réception pour chaque fils dans  $T$  est correcte, c.-à-d., on a  $\mathcal{S}_4 = \{p \in \mathcal{S}_3 : Good\_CR(p)\}$ .

**Lemme 10**

*Lorsque la règle  $R3$  est activable par  $p \in \mathcal{S}_3, p \notin \mathcal{S}_4$ , alors elle le reste jusqu'à ce que  $p$  active  $R3$  ou que  $p \in \mathcal{S}_4$ .*

**Preuve.** Soit  $\gamma \mapsto \gamma'$  un pas de calcul. Supposons, par contradiction, que la règle  $R3$  est activable par  $p \in (\mathcal{S}_3 - \mathcal{S}_4)$  dans  $\gamma$  et non activable dans  $\gamma'$  (c.-à-d.,  $cr_p =$



$ChildColor()$  pour  $p$  dans  $\gamma'$ ) alors que  $p$  n'a pas activé la règle  $R3$  durant le pas  $\gamma \mapsto \gamma'$ . D'après l'hypothèse du lemme, nous supposons que  $p \in (\mathcal{S}_3 - \mathcal{S}_4)$  dans  $\gamma'$ , donc on a  $Good\_ACE(p) \wedge Good\_CE(p) \wedge \neg Good\_ACR(p)$  dans  $\gamma'$ . Comme  $p$  n'a pas activé de règle durant le pas  $\gamma \mapsto \gamma'$  et que sa variable  $cr_p$  ne peut uniquement être modifiée localement par  $p$  en activant la règle  $R3$ , alors  $R3$  est activable par  $p$  dans  $\gamma'$ . Il y a donc contradiction et la règle  $R3$  est donc toujours activable par  $p$  dans  $\gamma'$ .  $\square$

**Lemme 11**

*Dans toute configuration  $\gamma \in (\mathcal{S} - \mathcal{B})$ ,  $q \in \mathcal{S}_3$  met à jour son ensemble de couleurs en réception si la Condition 4 de la Définition 2 n'est pas satisfaite pour l'arc  $(p, q) \in E_T$ .*

**Preuve.** D'après la description formelle de l'Algorithme SS-DD2EC, un nœud  $q \in \mathcal{S}_3$  doit activer la règle  $R3$  pour mettre à jour son ensemble de couleurs en réception pour chacun de ses fils dans  $T$  (c.-à-d., sa variable  $cr_q$ ). Comme l'arc  $(p, q)$  ne satisfait pas la Condition 4 de la Définition 2, on a  $(\forall(i, c) \in cr_q : i \neq p \vee c \notin (acr_q \cap ace_p) \vee c \notin \mathcal{C} \vee [\exists(z, c') \in cr_q : z \neq p \wedge c' = c])$ .

Supposons, par contradiction, que  $q$  ne met pas à jour son ensemble de couleurs en réception pour chacun de ses fils dans  $T$ . Cela implique soit que la règle  $R3$  est non activable par  $q$  dans  $\gamma$ , mais en contradiction avec le Théorème 2 (Condition 4 de la Définition 2 non satisfaite), ou que la règle  $R3$  n'est pas la règle activable par  $q$  de plus forte priorité dans  $\gamma$ , mais en contradiction avec le fait que les règles  $R0$ ,  $R1$  et  $R2$  sont non activables par  $q$  dans  $\gamma$  car  $(q \in \mathcal{S}_3) \Rightarrow (Good\_ACE(q) \wedge Good\_CE(q) \wedge Good\_ACR(q))$ . Enfin, d'après le Lemme 10 la règle  $R3$  est toujours activable par  $q \in (\mathcal{S}_3 - \mathcal{S}_4)$  jusqu'à son activation.  $\square$

**Lemme 12**

*À partir d'une configuration quelconque, l'arc  $e = (p, q) \in E_T$  de plus forte priorité dans  $T$  obtient une couleur correcte en  $O(1)$  rondes (asynchrones) si sa couleur n'est pas correcte (c.-à-d., ne satisfait pas la Définition 2).*

**Preuve.** Soit un arc  $e = (p, q) \in E_T$  de plus forte priorité ne satisfaisant pas la Définition 2 dans la configuration  $\gamma \in \mathcal{S}$ .

Si la Condition 1 ou 2 de la Définition 2 n'est pas satisfaite pour  $q$  dans  $\gamma$ , alors on a  $\neg Good\_ACR(q)$  pour  $q$  et  $q \in \mathcal{S}_2$ . D'après l'hypothèse du lemme, on a les prédicats

$Good\_ACE(q)$  et  $Good\_CE(q)$  qui sont satisfaits pour  $q$  car tous les autres arcs de plus haute priorité par rapport à l'arc  $e$  satisfont la Définition 2. Donc, comme l'ordonnanceur est faiblement équitable, d'après le Lemme 9 dans la première configuration de la ronde 2 le prédicat  $Good\_ACR(q)$  est satisfait pour  $q$ . En parallèle, si la Condition 3 de la Définition 2 n'est pas satisfaite par  $p$  dans  $\gamma$ , alors on a  $\neg Good\_ACE(p)$  pour  $p$ . Comme l'ordonnanceur est faiblement équitable, d'après le Lemme 5 dans la première configuration de la ronde 2 le prédicat  $Good\_ACE(p)$  est satisfait pour  $p$ . Par conséquent, dans la première configuration de la ronde 2 on a  $q \in \mathcal{S}_3$  et  $p \in \mathcal{S}_1$ . Si la Condition 4 de la Définition 2 n'est pas satisfaite pour  $q$  dans la première configuration de la ronde 2, alors on a  $\neg Good\_CR(q)$  pour  $q$ . Comme l'ordonnanceur est faiblement équitable, d'après le Lemme 11 dans la première configuration de la ronde 3 on a  $Good\_CR(q)$  pour  $q$ . Si la Condition 5 de la Définition 2 n'est pas satisfaite pour  $p$  dans la première configuration de la ronde 3, alors on a  $\neg Good\_CE(p)$  pour  $p \in \mathcal{S}_1$ . Comme l'ordonnanceur est faiblement équitable, d'après le Lemme 7 dans la première configuration de la ronde 4 on a  $Good\_CE(p)$  pour  $p$ . Donc, la Définition 2 est satisfaite pour l'arc  $e = (p, q) \in E_T$  en  $O(1)$  rondes.  $\square$

**Lemme 13**

À partir d'une configuration quelconque, l'Algorithme *SS-DD2EC* atteint une configuration légitime (Définition 3) au plus en  $O(\Delta n)$  rondes (asynchrones), avec  $\Delta$  le degré maximum d'un nœud dans  $G$  et  $n$  le nombre de nœuds. De plus,  $O(\min(n, \Delta^2)(\log(n) + \log(\min(n, \Delta^2))))$  bits de mémoire sont nécessaires sur chaque nœud.

**Preuve.** On définit par  $E_T^k \subseteq E_T$ ,  $1 \leq k \leq H(T)$ , l'ensemble des arcs  $e = (p, q) \in E_T^k$  tels que l'extrémité  $p$  est à distance  $k$  du puits  $s$  dans  $T$ . On notera  $H(T)$  la hauteur de  $T$  et  $n_k$  le nombre d'arcs dans  $E_T^k \subseteq E_T$ ,  $1 \leq k \leq H(T)$ .

Nous montrons d'abord par induction sur les distances dans  $T$  la proposition suivante : au plus en  $O(\Delta \times k \times n_k)$  rondes la Définition 2 est satisfaite pour chaque arc  $e = (p, q) \in E_T^k$ ,  $1 \leq k \leq H(T)$ .

Cas de base  $k = 1$  : on doit vérifier la proposition entre  $s$  ses fils dans  $T$ . D'après le Lemme 12, en  $O(1)$  rondes l'arc de plus haute priorité satisfait la Définition 2. On peut observer que pour chaque arc  $e = (p, s) \in E_T^1$  la Macro *PrioE*( $p$ ) retourne un ensemble

vide car il n'y a pas d'autres arcs plus prioritaires. Comme l'ordonnanceur est faiblement équitable, en  $O(1)$  rondes supplémentaires tous les arcs dans  $E_T^1$  satisfont la Définition 2, vérifiant la proposition.

Cas inductif : On suppose que la proposition est vraie pour chaque arc de  $E_T^{k-1}$ . Nous devons montrer que la proposition est vraie pour chaque arc de  $E_T^k$ . Chaque extrémité d'un arc  $e = (p, q) \in T$  est en conflit avec au plus  $\Delta - 1$  voisins, donc chaque arc  $e$  est en conflit avec au plus  $2\Delta$  autres nœuds. Cela implique que  $e$  devient prioritaire lorsque les  $2\Delta$  nœuds en conflit avec  $e$  et plus prioritaires ont terminé leur affectation de couleurs. Étant donné un arc  $(x, y)$  plus prioritaire associé à un nœud  $y$  en conflit avec  $p$  ou  $q$ , il faut considérer deux situations : (1) soit  $(x, y) \in E_T^i$  avec  $1 \leq i \leq k - 1$ , (2) soit  $(x, y) \in E_T^k$ . Dans la première situation le nœud  $y$  a terminé son affectation de couleurs, ce qui n'est pas le cas dans la seconde situation. Ainsi, dans ce dernier cas il faudra au plus  $O(1)$  rondes d'après le Lemme 12 pour que l'arc  $(x, y) \in E_T^k$  de plus forte priorité vérifie la Définition 2. Donc, après  $O(\Delta)$  rondes l'arc  $e$  a la plus forte priorité permettant d'appliquer le Lemme 12 pour vérifier la Définition 2. Comme il y a au plus  $n_k$  arcs dans  $E_T^k$ , au bout de  $O(\Delta n_k)$  rondes supplémentaires tous les arcs de  $E_T^k$  satisfont la Définition 2, ce qui vérifie la proposition. Par conséquent, pour chaque niveau  $1 \leq i \leq H(T)$  dans l'arbre  $T$  les arcs de  $E_T^i$  satisfont la Définition 2 en  $O(\Delta n_i)$ , donc au bout de  $O(\Delta \sum_{i=1}^{H(T)} n_i) \leq O(\Delta n)$  rondes une configuration légitime est atteinte.

Concernant la complexité en mémoire, chaque nœud  $p \in V$  maintient des listes de couleurs dans les variables  $ace_p$ ,  $acr_p$  et  $color\_temp_p$ , ainsi que deux listes de couples identifiants et couleurs en réception par chaque fils  $cr_p$  et  $cr\_temp_p$ . D'après le Lemme 3, l'Algorithme SS-DD2EC nécessite au plus  $\min(n - 1, (\Delta + 2)\Delta + 1)$  couleurs. Ainsi, les variables  $ace_p$ ,  $acr_p$  et  $color\_temp_p$  nécessitent  $O(\min(n, \Delta^2) \log(\min(n, \Delta^2)))$  bits. Les variables  $cr_p$  et  $cr\_temp_p$  nécessitent  $O(\Delta(\log(n) + \log(\min(n, \Delta^2))))$  bits. Enfin la variable  $ce_p$  stocke une seule couleur, la couleur en émission, et nécessite donc  $O(\log(\min(n, \Delta^2)))$  bits. Par conséquent, la complexité en mémoire est de  $O(\min(n, \Delta^2)(\log(n) + \log(\min(n, \Delta^2))))$  bits.  $\square$

**Lemme 14**

*Dans toute configuration  $\gamma \in \mathcal{B}$ , aucune règle de l'Algorithme SS-DD2EC n'est activable.*

**Preuve.** Supposons, par contradiction, que  $\exists \gamma \in \mathcal{B}$  tel qu'il existe un nœud ayant une règle activable pour l'Algorithme SS-DD2EC. Considérons l'arc  $e = (p, q) \in E_T$  avec la plus forte priorité tel que  $p$  ou  $q$  est activable. Si la règle  $R0$  est activable, alors la liste de couleurs autorisées en émission  $ace_p$  pour  $p$  n'est pas à jour (prédicat  $Good\_ACE(p)$  non satisfait). Ce voudrait dire qu'un voisin a changé de couleur, en contradiction avec le fait que  $\gamma \in \mathcal{B}$ . Le même argument s'applique pour  $q$  si la règle  $R2$  est activable concernant la liste de couleurs autorisées en réception (on considère que les prédicats  $Good\_ACE(q)$  et  $Good\_CE(q)$  sont satisfaits car par hypothèse  $q$  est uniquement activable à cause de l'arc  $e$ ). Enfin, si les règles  $R1$  et  $R3$  (ou  $R0$  pour le puits  $s$ ) sont activables respectivement pour  $p$  et  $q$ , cela implique que la couleur en émission  $ce_p$  est différente de celle attribuée en réception dans  $cr_q$  pour  $q$  par rapport à  $p$ , ce qui est en contradiction avec l'hypothèse que  $\gamma \in \mathcal{B}$ . □

Le Théorème 2 et les Lemmes 13 et 14 induisent le théorème suivant.

**Théorème 3**

*L'algorithme SS-DD2EC est auto-stabilisant pour la Spécification 1 sous un ordonnanceur faiblement équitable.*

# Glossaire

- *CSMA* : Carrier Sense Multiple Access
- *DD2EC* : Distributed Distance-2 Edge Coloring)
- *DODAG* : Destination Oriented Acyclic Graphs
- *DIO* : DODAG Information Object
- *DIS* : DODAG Information Solicitation
- *ETX* : Expected Transmission count
- *FO* : Fonction Objectif
- *IPV6* : Internet Protocol version 6
- *LPWAN* : Low-Power Wide-Area Networks
- *LRWPAN* : Low Rate Wireless Personal Area Network
- *6LowPAN* : IPV6 Low Power Area Network
- *MAC* : Medium Access Control
- *PDR* : Taux de livraison des paquets
- *RCSF* : Réseaux de Capteurs Sans Fil
- *RPL* : Routing Protocol for Low-power and Lossy networks
- *RX* : Taux de réception des données
- *SS – DD2EC* : Self-Stabilizing Distributed Distance-2 Edge Coloring
- *TDMA* : Time Division Multiple Access
- *TX* : Taux de transmission des données



**Résumé :**

Les réseaux de capteurs sans fil (RCSF) sont très largement utilisés dans divers domaines d'applications civiles et militaires, comme la surveillance de champs de bataille, la détection de phénomène environnementaux, ou encore les bâtiments intelligents. Cependant, les réseaux de capteurs sans fil sont caractérisés par de fortes limitations au niveau de l'énergie disponible et des communications radio. Les travaux réalisés dans cette thèse visent à proposer des solutions garantissant une certaine qualité de service dans le contexte des réseaux de capteurs sans fil. La première partie concerne la couche liaison de données avec l'objectif d'augmenter la durée de vie du réseau. L'accès au médium sans fil est analysé et modélisé sous la forme d'un problème d'ordonnancement des liens de communication, tenant compte des collisions. Nous étudions alors la complexité de ce problème. Une approche distribuée et tolérante aux défaillances avec garantie de performance est proposée (SS-DD2EC) pour résoudre ce problème. La seconde partie de la thèse concerne le routage des messages à l'aide du protocole IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL). Tout d'abord, un comparatif entre les différentes métriques de routage existantes pour l'optimisation de l'énergie consommée a été mené. En plus de la durée de vie, les critères de fiabilité et de latence de bout-en-bout sont considérés pour évaluer ces métriques. Enfin, deux nouvelles métriques ( $R\_MinMax$  et  $R\_Delai$ ) ont été proposées pour RPL permettant d'atteindre des gains significatifs par rapport à l'état de l'art. La première ne tient compte que des critères de consommation d'énergie et de fiabilité, alors que la seconde intègre en plus la latence de bout-en-bout.

**Mots clés :**

Réseaux de capteurs sans fil, Accès au médium, Coloration d'arêtes à distance-2, Routage, métrique RPL.

**Abstract :**

Wireless Sensor Networks (WSNs) is a technology that has a wide range of civil or military applications, including battlefield monitoring, environmental monitoring or smart city. However, WSN are characterised by high limitations in terms of energy (battery-operated nodes) and wireless links (low power and lossy links). The work done in this PhD thesis aims to provide solutions that guarantee a certain quality of service in the context of wireless sensor networks. The first part of this work concerns the medium access control layer with the aim of increasing the lifetime of the network. The access to the wireless medium is analyzed and modeled as a link scheduling problem, taking into account collisions. First, a study of the complexity of this problem is carried out, then a distributed and fault-tolerant approach with guaranteed performance is proposed (SS-DD2EC) to solve this problem. The second part is about message routing with the IPv6 Routing Protocol for Low Power and Lossy Networks (RPL). First of all, a comparison between the various existing routing metrics for the optimization of the energy consumed has been carried out. In addition of lifetime, the reliability and end-to-end latency criteria are considered for evaluating these metrics. Then, two new RPL metrics ( $R\_MinMax$  and  $R\_Delai$ ) were proposed, achieving significant gains over the state of the art. The first one only considers the energy consumption and reliability, while the second one takes also into account the end-to-end latency.

**Keywords :**

Wireless Sensor Networks, Medium Access Control, Distance-2 edge coloring, Message routing, RPL metrics.