HAL
open science

Collecte et remontée multi-sauts de données issues de lecteurs RFID pour la surveillance d'infrastructures urbaines

Abdoul Aziz Mbacké

To cite this version:

Abdoul Aziz Mbacké. Collecte et remontée multi-sauts de données issues de lecteurs RFID pour la surveillance d'infrastructures urbaines. Informatique [cs]. Université de Lille, 2018. Français. NNT : . tel-01901740

HAL Id: tel-01901740

https://inria.hal.science/tel-01901740v1

Submitted on 23 Oct 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.

Université de Lille

Université de Lille

École doctorale **ED Régionale SPI 72**
Unité de recherche **Inria Lille - Nord Europe**

Thèse présentée par **Abdoul Aziz MBACKE**
Soutenue le **18 octobre 2018**
En vue de l'obtention du grade de docteur de l'Université de Lille

Discipline **Informatique et Applications**
Spécialité **Réseaux**

Titre de la thèse

Collecte et remontée multi-sauts de données issues de lecteurs RFID pour la surveillance d'infrastructures urbaines

Thèse dirigée par	Nathalie MITTON	directrice
	Hervé RIVANO	co-directeur

Composition du jury

Rapporteurs	Hakima CHAOUCHI	professeure à l'Institut Télécom SudParis
	André-Luc BEYLOT	professeur à l'INP/ENSEEIHT Toulouse
Examinateurs	Mamadou MBOUP	professeur à l'Université de Reims
	Cheikh SARR	professeur à l'Université de Thiès
Directeurs de thèse	Nathalie MITTON	directrice de recherche à l'Inria
	Hervé RIVANO	professeur à l'Université de Lyon Inria INSA Lyon CITI

Université de Lille

Université de Lille

École doctorale **ED Régionale SPI 72**
Unité de recherche **Inria Lille - Nord Europe**

Thèse présentée par **Abdoul Aziz MBACKE**
Soutenue le **18 octobre 2018**
En vue de l'obtention du grade de docteur de l'Université de Lille

Discipline **Informatique et Applications**
Spécialité **Réseaux**

Titre de la thèse

Collecte et remontée multi-sauts de données issues de lecteurs RFID pour la surveillance d'infrastructures urbaines

Thèse dirigée par	Nathalie MITTON	directrice
	Hervé RIVANO	co-directeur

Composition du jury

Rapporteurs	Hakima CHAOUCHI	professeure à l'Institut Télécom SudParis
	André-Luc BEYLOT	professeur à l'INP/ENSEEIHT Toulouse
Examinateurs	Mamadou MBOUP	professeur à l'Université de Reims
	Cheikh SARR	professeur à l'Université de Thiès
Directeurs de thèse	Nathalie MITTON	directrice de recherche à l'Inria
	Hervé RIVANO	professeur à l'Université de Lyon Inria INSA Lyon CITI

Université de Lille

Université de Lille

Doctoral School **ED Régionale SPI 72**
University Department **Inria Lille - Nord Europe**

Thesis defended by **Abdoul Aziz MBACKE**

Defended on **18th October, 2018**

In order to become Doctor from Université de Lille

Academic Field **Computer Science and Applications**
Speciality **Networking**

Thesis Title

Collection and multi-hop forwarding of RFID data for the monitoring of urban infrastructures

Thesis supervised by	Nathalie MITTON	Supervisor
	Hervé RIVANO	Co-Supervisor

Committee members

Referees	Hakima CHAOUCHI	Professor at Institut Télécom SudParis
	André-Luc BEYLOT	Professor at INP/ENSEEIHT Toulouse
Examiners	Mamadou MBOUP	Professor at Université de Reims
	Cheikh SARR	Professor at Université de Thiès
Supervisors	Nathalie MITTON	Senior Researcher at Inria
	Hervé RIVANO	Professor at Université de Lyon Inria INSA Lyon CITI

Cette thèse a été préparée dans les laboratoires suivants.

Inria Lille - Nord Europe

Parc Scientifique de la Haute Borne
40, Avenue du Halley
Bât A - Park Plaza
59650 Villeneuve d'Ascq
France

☎ (+33) 03 59 57 78 00
(+33) 03 59 57 78 50
contact-lille@inria.fr
Site https://www.inria.fr/centre/lille

CITI Lab

Building Claude Chappe
6 avenue des Arts
F-69621 Villeurbanne
France

☎ (+33) 04 72 43 73 22
citi.direction@insa-lyon.fr
Site http://www.citi-lab.fr/

IoT is about observing the world in real time to make life better

---

Lo doonul talibeem, mënulo doon serignam

L'Opportunité est le Dieu Inventeur,
L'Expérience est la fille des fautes des fautes graves,
Le Génie est l'ami des paradoxes

Believe in something. Even if it means sacrificing everything

---

Colin Kaepernick

I am the master of my fate, I am the captain of my soul

William Ernest Henley

Collecte et remontée multi-sauts de données issues de lecteurs RFID pour la surveillance d'infrastructures urbaines

Résumé

La forte urbanisation dont le monde est témoin exige une meilleure gestion des villes. Cette gestion améliorée passe par la surveillance et la maintenance des équipements et infrastructures urbaines afin d'assurer plus de sécurité et bien-être aux habitants. Pour atteindre ces objectifs, un rôle clé a été confié aux technologies de l'information et de la communication (TIC) à travers les concepts de l'Internet des Objets (IoT) et des Villes Intelligentes. Nous avons donc observé un nombre grandissant de dispositifs de captation de données intégrés à différents éléments de la ville pour offrir des services et applications dédiés. Cependant, certaines contraintes matérielles, technologiques, économiques et environnementales relatives à la nature des dispositifs utilisés freinent encore les objectifs de couverture et de gestion totales des infrastructures urbaines. Cette thèse se positionne dans ce contexte et propose l'Identification par Radio Fréquence (RFID) en complément des techniques déjà utilisées afin de pallier ces manquements. Toutefois l'adoption de la RFID à grande échelle pour les centres urbains nécessite de résoudre deux principaux problèmes : les collisions de lecture et la collecte et remontée des données. A travers les travaux menés dans cette thèse, nous avons d'abord cherché à identifier les solutions déjà proposées dans la littérature pour réduire les collisions. Sur la base de cette étude, nous avons proposé deux algorithmes distribués d'anti-collision de lecture DEFAR et CORA. Ils permettent d'assurer un débit de lecture important tout en maintenant un taux de collisions et une latence de couverture faible comparés aux solutions de la littérature pour des déploiements denses et/ou mobiles de lecteurs RFID. Par la suite, nous avons proposé DACAR, un algorithme distribué de collecte des données issues des lecteurs RFID de manière multi-sauts. Ce protocole s'adapte en fonction du protocole d'anti-collision utilisé et de la position des lecteurs déployés pour fournir un ratio de délivrance des paquets fiable et un faible délai de bout-en-bout. Une version améliorée de DACAR est ultérieurement proposée qui prend en compte la priorisation des données pour offrir des chemins différents plus appropriés à l'aide d'une combinaison de différents paramètres grâce à la logique floue.

Mots clés : rfid, internet des objets, réseaux de capteurs sans-fil, villes intelligentes

Inria Lille - Nord Europe
Parc Scientifique de la Haute Borne – 40, Avenue du Halley – Bât A - Park Plaza – 59650 Villeneuve d'Ascq – France

Collection and multi-hop forwarding of RFID data for the monitoring of urban infrastructures

Abstract

The strong urbanization witnessed by the world requires better management of cities. This improved management involves the monitoring and maintenance of urban infrastructures and equipment to ensure greater safety and well-being for residents. To achieve these objectives, a key role has been given to Information and Communication Technologies (ICT) through the concepts of Internet of Things (IoT) and Smart Cities. As a result, we have seen a growing number of data capture devices integrated into different parts of the city to deliver dedicated services and applications. However, given material, technological, economic and environmental constraints related to the nature of the devices in use, the objectives of total coverage and management of urban infrastructures is still hindered. This thesis is positioned in this context and proposes Radio Frequency Identification (RFID) in addition to the techniques already in use to overcome these shortcomings. However, the adoption of large-scale RFID for urban centers requires solving two main issues: reading collisions and data collection and forwarding. Through the work carried out in this thesis, we first sought to identify the solutions already proposed in the literature to reduce collisions. Based on this study, we proposed two distributed anti-collision algorithms named DEFAR and CORA. They provide high throughput while maintaining low collision and latency rates compared to literature solutions for dense and/or mobile deployments of RFID readers. Subsequently, we proposed DACAR, a distributed algorithm for multi-hop data collection using RFID readers. This protocol is adaptable and according to the anti-collision protocol used and the position of deployed readers, provides a reliable packet delivery ratio and low end-to-end delay. An improved version of DACAR is later proposed that takes into account the prioritization of data to provide different, improved and more suitable paths using a combination of different parameters through fuzzy logic.

Keywords: rfid, internet of things, wireless sensor networks, smart cities

Sommaire

Résumé **xi**

Sommaire **xiii**

Liste des tableaux **xvii**

Table des figures **xix**

1 Introduction générale **1**

2 Anticollision de lecteurs RFID - Problématique et Etat de l'art **11**
2.1 Présentation de la problématique des collisions 11
2.1.1 Collisions de tags RFID 12
2.1.2 Collisions de lecteurs RFID 14
2.2 Présentation des métriques d'évaluation des protocoles d'anti-collision RFID 16
2.2.1 Débit de lecture 16
2.2.2 Collisions 17
2.2.3 Efficacité 17
2.2.4 Indice d'équité de Jain (IEJ) 18
2.2.5 Délai de couverture 18
2.3 Etat de l'art sur les protocoles d'anti-collision de lecteurs RFID 19
2.3.1 Catégorisation des protocoles 19
TDMA ou CSMA 19
Centralisé ou Distribué 21
Monocal ou Multicanal 22
2.3.2 Protocoles TDMA 22
Approches Distribuées 22
Approches Centralisées 25
2.3.3 Protocoles CSMA 29
Approches Distribuées 29

Approches centralisées 32
2.3.4 Évaluation de Performances des protocoles proposés . . 33
2.4 Conclusion 38

3 Anticollision de lecteurs RFID - Nos contributions 39
3.1 Distributed Efficient & Fair Anti-collision for RFID (DEFAR) . 40
3.1.1 Positionnement 40
3.1.2 Fonctionnement 43
3.1.3 mobile-DEFAR (mDEFAR) 49
3.2 Coverage Oriented RFID Anti-collision (CORA) 49
3.2.1 Positionnement 50
3.2.2 Fonctionnement 52
3.3 Résultats obtenus 55
3.3.1 Déploiements statiques 57
3.3.2 Déploiements dynamiques 60
Entrepôt 60
Ville intelligente 61
Résultats 61
3.4 Conclusion 65

4 Remontée d'informations RFID 67
4.1 RFID en complément des réseaux de capteurs classiques pour la ville intelligente 68
4.1.1 Sensing tags 69
4.1.2 Exemples d'applications utilisant la RFID 71
4.2 Remontée de données RFID 72
4.3 Distributed & Adaptable Crosslayer Anti-collision and Routing (DACAR) 75
4.3.1 Positionnement 76
4.3.2 Fonctionnement 79
4.3.3 Résultats 84
4.3.4 Discussion 86
4.4 DACAR + Qualité de Service 86
4.4.1 Positionnement 87
Simplification des données ou Fuzzification 88
Définition des règles floues 89
Obtention de la variable de sortie ou Defuzzification . . 89
4.4.2 Fonctionnement 89
Envoi avec $parent_i$ 90
Envoi avec $parentprio_i$ 91
4.4.3 Résultats 92

4.5 Conclusion 94

5 Conclusion générale **97**
5.1 Conclusion sur les contributions de la thèse 98
5.2 Travaux futurs 99
5.2.1 Modèle de propagation 100
5.2.2 Redondance et Compression des données 100
5.2.3 Consommation d'énergie 102
5.3 Perspectives 102

Bibliographie **107**

Table des matières **115**

Liste des tableaux

1.1 Classification RFID 6

2.1 Protocoles d'anti-collision de tags 13
2.2 Récapitulatif des protocoles d'anti-collision de lecteurs RFID . 33

4.1 Définition des règles floues 89
4.2 Définition des seuils low_m et up_m 93

Table des figures

1.1 Variation populations françaises urbaine et rurale [20] 2
1.2 Vision de la ville intelligente 3
1.3 Rétrodiffusion ou Backscattering 4
1.4 Différents types de tags RFID 5
1.5 Différents types de lecteurs RFID 5

2.1 Collisions de tags RFID 13
2.2 Collisions de lecteurs RFID 15
2.3 Évaluation de performances 34
2.4 Performance des protocoles d'anti-collision en fonction des contraintes d'applications 36
2.5 Performances en fonction des contraintes de collisions et latence pour des déploiements denses et mobiles de lecteurs 37

3.1 Structure d'un *round* - DEFAR 43
3.2 Diagramme d'états 43
3.3 Illustration DEFAR 48
3.4 Structure d'un *round* - CORA 51
3.5 Résolution collisions CORA 52
3.6 Illustration CORA 55
3.7 Résultats de simulations sur déploiements statiques 58
3.8 Scénarios simulations dynamiques 60
3.9 Comparaison débit de lecture, collisions et efficacité dans les différents scénarios 62
3.10 Résultats délai de couverture 62

4.1 Comparaison solutions capteurs traditionnels et "sensing tag" RFID 72
4.2 Taxonomie des protocoles de routage [78] 73
4.3 Exemple d'un réseau multi-sauts RFID 74
4.4 Exemple d'un réseau multi-sauts RFID 77

4.5 Exemple d'organisation des lectures et transmission données avec DACAR 80
4.6 Organisation de la contention, des lectures et de l'envoi des données avec DACAR 81
4.7 Illustration DACAR 83
4.8 Résultats - DACAR 84
4.9 Etapes de la logique floue 90
4.10 Définition du noeud relais en fonction des tags lus 91
4.11 Illustration DACAR + QoS 92
4.12 Résultats - DACAR+QoS 94

5.1 Considération du modèle de propagation 101
5.2 Exemple d'un produit du commerce avec tag RFID 104
5.3 Identification des déchets et instruction 105
5.4 Réseau de bacs à déchets 105

Chapitre 1

Introduction générale

Les villes ou centres urbains ont toujours été des espaces où se développent les communautés sur les plans économiques, sociaux et culturels. Elles constituent une pièce maîtresse de l'organisation territoriale permettant de centraliser les services mis à disposition des citoyens et servant de levier pour le développement des zones rurales environnantes. La révolution industrielle créant de nouveaux métiers et le délaissement des activités agricoles primaires aidant, on assiste à une migration importante des populations rurales vers ces centres urbains. Cette migration s'est faite rapidement avec aujourd'hui un constat édifiant faisant état de déjà plus de la moitié de la population mondiale qui vivent dans ces centres urbains [20]. Ce chiffre est appelé à atteindre les 68% d'ici 2050 selon l'Organisation des Nations Unies [65]. Dans le cas particulier de la France, la population urbaine atteint aujourd'hui déjà environ 80% de la population totale avec un taux prévisionnel d'un peu moins de 90% d'ici 2050 (voir Figure 1.1).

Les infrastructures, équipements et ressources de ces centres urbains n'ont pas été conçus pour supporter cette croissance rapide et sont aujourd'hui en difficulté. Il est important de prendre les actions et mesures nécessaires afin d'accommoder les villes à cette démographie croissante. Pour faire en sorte que les bénéfices de l'urbanisation soient partagés et que personne ne soit laissé pour compte, les politiques de gestion de la croissance urbaine doivent garantir l'accès aux infrastructures et aux services pour tous. Et ceci doit être fait en mettant l'accent sur les besoins des citadins en termes de logement, d'éducation, de soins

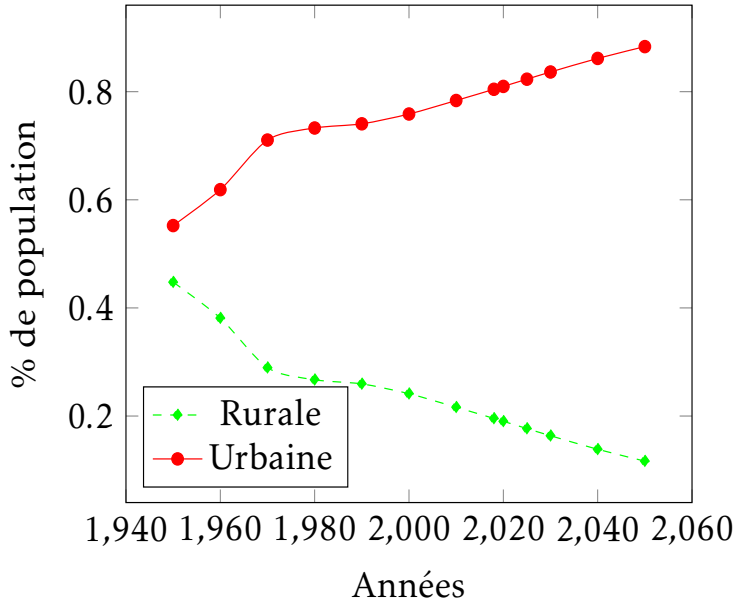


FIGURE 1.1 – Variation populations françaises urbaine et rurale [20]

de santé et surtout de sécurisation de leur environnement.

A cet effet, germa, au début des années 1990, l'idée de mettre en oeuvre les avancées technologiques numériques au profit de la ville [33] et naquît le concept de "Ville Intelligente". La vision commune est que l'ensemble de l'espace urbain, y compris les bâtiments, ponts, lignes électriques, conduites de gaz, routes, réseaux de transport et téléphones cellulaires, peut être mis en réseau et surveillé. Dans l'ensemble, la vision de la ville intelligente soulève la perspective que les villes vont devenir des environnements plus durables, avec pour objectif d'améliorer le « bien-être » des citadins. Cette perspective et ces objectifs permettront l'évolution vers de nouvelles façons de vivre grâce à la participation active des habitants par différentes voies pour une meilleure régulation, exploitation et gestion des villes et de leurs ressources (Voir Figure 1.2).

Ce manuscrit s'inscrit dans ce contexte et se concentre sur le "*monitoring*" des infrastructures urbaines (comme les ponts ou bâtiments). Ce type d'infrastructures nécessite d'être constamment surveillé pour ce qui concerne les vibrations, les agressions de toutes sortes (le gel, le vent, les voitures, pressions, etc.) grâce à différents capteurs permettant de renseigner ces données environnementales.

FIGURE 1.2 – Vision de la ville intelligente

Toutefois, ces capteurs ne peuvent pas tous être connectés physiquement à l'aide de câbles en raison du coût de déploiement d'une telle infrastructure câblée. Le sans-fil devient la solution. On assiste alors au déploiement à grande échelle de solutions de réseaux de capteurs sans-fil (Wireless Sensor Networks, WSN) auxquels sont rattachés des capteurs pour récupérer des données environnementales.

Tout d'abord, pour certaines mesures, les capteurs doivent être enterrés ou enfouis (dans le béton par exemple) mais cela entraînerait une dégradation des communications. Par ailleurs, ces capteurs nécessitent la présence d'une batterie afin de maintenir leur fonctionnement et les mesures requises. Un déploiement de plusieurs centaines, voire milliers de capteurs à travers la ville rend la maintenance de ces batteries particulièrement fastidieuse et engendre éventuellement un coût énergétique important. Ces inconvénients conduisent à de nouvelles questions relatives aux contraintes matérielles et à la mise en place de réseaux sans fil. Une solution technologique consiste à envisager l'usage de l'Identification par Radio Fréquence (Radio Frequency Identification, RFID).

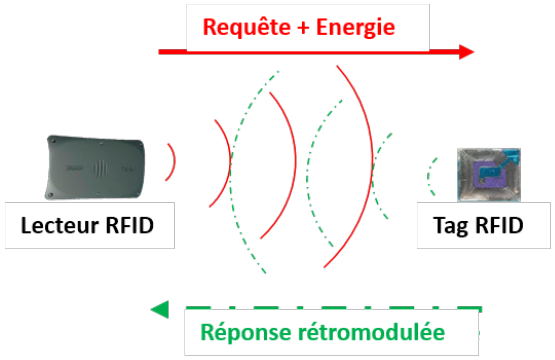


FIGURE 1.3 – Rétrodiffusion ou Backscattering

La RFID est une technologie issue du monde militaire qui pendant la deuxième guerre mondiale permettait d'identifier les aéronefs ennemis de ceux des alliés pour les troupes au sol. Aujourd'hui, et ce depuis quelques années déjà, face aux contraintes et limites des codes barres en termes de portée de lecture, de niveau de complexité, de quantité de données et de visibilité, la RFID est devenue une alternative performante permettant l'identification sans contact ni besoin de visibilité directe de biens ou de personnes. Elle fonctionne sur la base de deux acteurs principaux à savoir :

1. les tags : ce sont de petites puces organisées autour d'unités de traitement, de stockage et de communication. Ces puces stockent l'information désirée et la renvoie lors de la réception d'un signal d'interrogation défini. On distingue principalement 3 types de tags RFID (voir Figure 1.4) :
   (a) tags actifs : ils disposent d'une batterie qui alimente le système et leur permet de communiquer et d'émettre leurs informations de manière autonome (voir Figure 1.4a) ;
   (b) tags passifs (voir Figure 1.4b) : ils n'ont aucune source d'énergie interne et sont alimentés par le signal d'interrogation reçu en utilisant le phénomène d'induction et rétrodiffusion (backscattering) [28] (voir Figure 1.3) ;
   (c) tags semi-passifs : ils sont à mi-chemin entre les passifs et les actifs. Ils comportent également une batterie mais cette dernière n'est utilisée que pour alimenter des composants externes (mémoire pour stocker

des données à intervalles réguliers dans la traçabilité alimentaire par exemple), la communication se faisant toujours selon le principe du backscattering comme les tags passifs (voir Figure 1.4c).

2. les lecteurs : ce sont des dispositifs utilisés pour interroger les tags et ainsi récupérer les informations stockées. Ils transmettent l'énergie nécessaire aux tags passifs et semi-actifs pour leur alimentation et fonctionnement. Nous distinguerons deux types de lecteurs (voir Figure 1.5) :
   (a) lecteurs fixes ou statiques : ils sont rattachés à des points d'ancrages précis et identifient donc les tags qui se retrouvent dans leur portée (exemple : portiques dans les commerces) ;
   (b) lecteurs mobiles : ils sont généralement plus petits et prennent la forme de dispositifs portables (voir Figure 1.5b).

(a) Tag RFID actif (b) Tag RFID passif (c) Tag RFID semi passif avec sonde de température

FIGURE 1.4 – Différents types de tags RFID

(a) Portiques RFID (b) Lecteur RFID mobile

FIGURE 1.5 – Différents types de lecteurs RFID

RFID				
Bandes de Fréquence	*Basses Fréquences*	*Hautes Fréquences*	*Ultra Hautes Fréquences*	
	125- 134 kHZ	*13.56 MHz*	*Tags actifs*	*Tags passifs*
			433 & 856 - 960 MHz	*856 - 960 MHz*
Caractéristiques	Très bas coût Très courte portéé	Bas coût Courte portée	Coût très élevé Très longue portée	Bas coût Portée variable
Exemples d'applications	Surveillance animale Chronométrage sportif	Paiement Passeports biométriques	Surveillance Santé	Logistique Péage autoroute

Tableau 1.1 – Classification RFID

La technologie RFID fonctionne sur plusieurs bandes de fréquence qui déterminent la portée mais également le type d'applications dédiées (voir Tableau 1.1). Nous retiendrons principalement les Basses Fréquences (Low Frequency, LF) avec une portée d'une dizaine de centimètres au plus ; les Hautes Fréquences (High Frequency , HF) avec une portée plus grande montant jusqu'à 1m environ ; les Ultra Hautes Fréquences (Ultra High frequency, UHF). Les tags UHF sont généralement plus petits et permettent une interrogation sur une portée de plusieurs mètres. Dans la suite de ce manuscrit, nous nous concentrerons principalement sur les tags RFID passifs en UHF en respect du standard [1]. Ces derniers sont plus en concordance avec nos enjeux en terme de portée de communication pour couvrir un environnement urbain mais également en terme de consommation énergétique car ne nécessitant pas de batterie. La longue portée de lecture, la faible consommation d'énergie, elle-même fournie par le lecteur, et le développement de la technologie permettent aujourd'hui de coupler les tags à des dispositifs de captation de données environnementales [88, 15, 93, 7]. Cela conforte l'idée que la technologie RFID devrait aujourd'hui être considérée comme un complément important des réseaux de capteurs sans fils pour les applications de villes intelligentes.

Toutefois, comme la majorité des technologies sans fils, la RFID fait face à une problématique importante qui est celle du partage du canal de communication. Les lecteurs communiquent avec les tags sur un canal d'interrogation défini et il est nécessaire d'organiser l'accès à ce canal afin d'éviter des accès multiples compétitifs. En effet, lorsque plusieurs dispositifs communiquent simultanément sur le support, on a à faire à des **collisions** [9, 23, 25]. Ces dernières peuvent être de différents types en fonction des dispositifs (tags et/ou lecteurs) engagés et engendrent des erreurs de lectures se solvant par la non identification,

voire la non détection, des tags. Ces collisions affectent donc grandement les performances des solutions RFID envisagées et doivent être évitées par la mise en place d'algorithmes d'anti-collision permettant un accès organisé au canal entre les différents dispositifs. Ces algorithmes se devront d'être évolutifs au vu de l'activité des villes et de leur expansion continue. L'installation de nouveaux tags et lecteurs pouvant se faire continuellement, il faudra donc faire face à des déploiements denses de lecteurs et/ou tags tout en gardant une organisation d'accès local au canal d'interrogation. Cette forte densité de déploiement peut également être couplée à une mobilité des dispositifs déployés pour permettre une couverture plus large.

Cette problématique des collisions fera donc l'objet de la réponse à notre première question à savoir "**Comment améliorer les performances des solutions RFID grâce à des algorithmes d'anti-collision ?**". Pour répondre à cette question nous avons d'abord cherché à identifier les différentes solutions déjà proposées dans la littérature, comprendre les mécanismes mis en place et leurs performances pour ensuite les classifier en fonction de leurs aptitudes. Sur la base des constats faits de ces propositions, nous avons cherché à produire des algorithmes améliorés tenant compte des contraintes de déploiement sus-cités. Ainsi nous proposons d'abord "*Distributed Efficient & Fair Anticollision for RFID (DEFAR)*" et son dérivé "*mobile-DEFAR (mDEFAR)*". Ces deux algorithmes permettent de réduire de manière importante les collisions tout en maintenant une activité performante des lecteurs. Par la suite, nous avons proposé "*Coverage Oriented RFID Anti-collision (CORA)*" qui permet de réduire la durée nécessaire pour lire l'ensemble des tags déployés grâce à un constat sur la nature particulière des collisions en RFID.

Une fois les tags lus et les informations récupérées au niveau des différents lecteurs déployés, une deuxième problématique fait surface : celle de la **remontée des données** vers un centre de traitement et de décision. Tout comme les capteurs d'un réseau de capteurs sans-fil ne sauraient tous être câblés pour être reliés à une infrastructure centrale au vu non seulement du coût engendré mais aussi de la complexité logistique qui en découle, les lecteurs devront donc également pouvoir communiquer entre eux et avec un point de collecte de manière sans-fil. Une solution sans fil devra tenir compte des exigences des applications

en termes de latence, fiabilité et continuité du service, mais également de la stratégie de déploiement des lecteurs (densité et mobilité), de leurs positions et de leurs activités. Par ailleurs, pour des raisons d'efficacité énergétique, il faudra également penser à faire communiquer les lecteurs de pair à pair afin d'éviter des transmissions énergivores sur de longues distances. Cela passera par la mise en place de techniques de collecte de données multi-sauts entre les différents lecteurs déployés.

Cette deuxième problématique pourra être résumée par la question suivante : "**Comment intégrer la RFID au coeur de la ville intelligente ?**". La réponse à cette question s'est faite d'abord par l'étude des solutions RFID permettant l'identification et la captation de données de manière passive avec la présentation des principales propositions faites. Dans un second temps, nous avons proposé "*Distributed & Adaptable Crosslayer Anti-collision and Routing (DACAR)*" : cet algorithme s'adapte à différentes solutions d'anti-collision de la littérature pour collecter et acheminer les données des tags lus. Les résultats obtenus permettent d'avoir une collecte fiable des données dans le respect de la latence tolérée par les applications considérées. Ultérieurement, une version améliorée a également été proposée. Elle prend en compte la gestion des données critiques pour les acheminer de manière prioritaire mais utilise également de nouveaux critères d'états du réseau. Cette déclinaison propose une latence encore réduite et une fiabilité maximale de la transmission des données.

Cette thèse s'inscrit dans le cadre du projet CityLab@Inria. Ce projet étudie des solutions pour les villes intelligentes qui promeuvent la durabilité sociale et environnementale. En vue de cet objectif, il entreprend un programme de recherches multidisciplinaires à travers l'intégration d'études scientifiques et technologiques notables, afin de participer à l'établissement de villes intelligentes. Les thèmes de recherche couvrent : les protocoles éco-énergétiques de communication sans fil, le sensing social et physique à l'échelle urbaine, la gestion des données urbaines dans le cloud, etc. Par ailleurs, CityLab@Inria s'appuie sur des collaborations à l'échelle internationale et spécialement en partenariat avec le programme Inria@SiliconValley.

Notre sujet de thèse, intitulé "**Collecte et remontée multi-sauts de données issues de lecteurs RFID pour la surveillance d'infrastructures urbaines**",

pourra donc être traité par la réponse aux deux questions précédemment établies. Le manuscrit est organisé en cinq chapitres traitant successivement des réponses apportées aux questions présentées. L'organisation se fait comme suit :

— le chapitre 2 détaille la problématique des collisions et précise notre travail par rapport au type de collisions qui seront abordées. Les différentes métriques d'évaluation des algorithmes d'anti-collision sont présentées. Elle permettent par la suite de pouvoir mieux évaluer l'état de l'art sur l'anti-collision de lecteurs RFID proposé par la suite.

— le chapitre 3 présente nos contributions relatives à l'anti-collision de lecteurs suite au constat du chapitre précédent. Les résultats obtenus grâce à ces propositions sont comparés à ceux d'un algorithme de l'état de l'art.

— le chapitre 4 s'intéresse à la remontée des données avec d'abord la présentation des solutions existantes pour la captation de données en utilisant la RFID et ensuite nos contributions pour la remontée de données RFID multi-saut via les lecteurs.

Enfin, le chapitre 5 conclut ce manuscrit par la présentation des travaux futurs susceptibles d'améliorer les résultats obtenus mais également par l'exploration de perspectives générales de réutilisation de nos travaux et des interrogations qu'il faudrait résoudre pour y parvenir.

Chapitre 2

Anticollision de lecteurs RFID - Problématique et Etat de l'art

Avec le succès de la RFID, les applications reposant sur cette technologie sont devenues de plus en plus nombreuses. La technologie étant basée sur une communication sans fil entre deux principaux acteurs (tag et lecteur), elle est sujette à l'un des inconvénients majeurs de ce type de communication à savoir les collisions. Ces collisions sont néfastes et impactent négativement les performances des applications RFID en générant des erreurs de lecture de tags. A cet effet, plusieurs solutions ont été proposées dans la littérature pour réduire ces collisions à travers des méthodes d'ordonnancement et/ou de planification des activités des lecteurs.

Dans ce chapitre, nous cherchons à comprendre ce phénomène de collision, identifier les métriques de performances des solutions RFID et différentes propositions d'anti-collision de lecture RFID afin de mieux positionner des travaux de notre thèse.

2.1 Présentation de la problématique des collisions

La communication en RFID, comme présentée en Chapitre 1, repose sur l'échange d'informations entre les tags et les lecteurs. Les tags possèdent une information et nécessitent l'interrogation d'un lecteur afin de pouvoir la parta-

ger. En prenant l'exemple d'une entreprise agro-alimentaire, il est possible de retrouver la RFID à chaque étape de la chaîne de production jusqu'à la commercialisation. En effet, au sein de l'entrepôt où les produits sont traités, on peut rattacher des tags RFID en fonction de différents critères (nature des produits, date de fabrication, étape dans la chaîne de fabrication, etc) et ainsi retrouver un produit ou une classe de produits en interrogeant les tags à l'aide de lecteurs soit déployés à des points stratégiques de l'entrepôt, soit confiés aux ouvriers chargés du traitement. A la sortie d'entrepôt, pour la répartition des produits dans les différents points de vente, l'utilisation des tags RFID permet encore une fois d'identifier les denrées et de s'assurer de leur bonne distribution. Une fois dans les points de vente, ces mêmes tags permettent de retrouver les différentes unités de produits au sein de l'espace de vente et/ou de stockage. Aujourd'hui ces tags peuvent même être utilisées pour faciliter la vente des produits en facilitant le passage en caisse automatique des consommateurs.

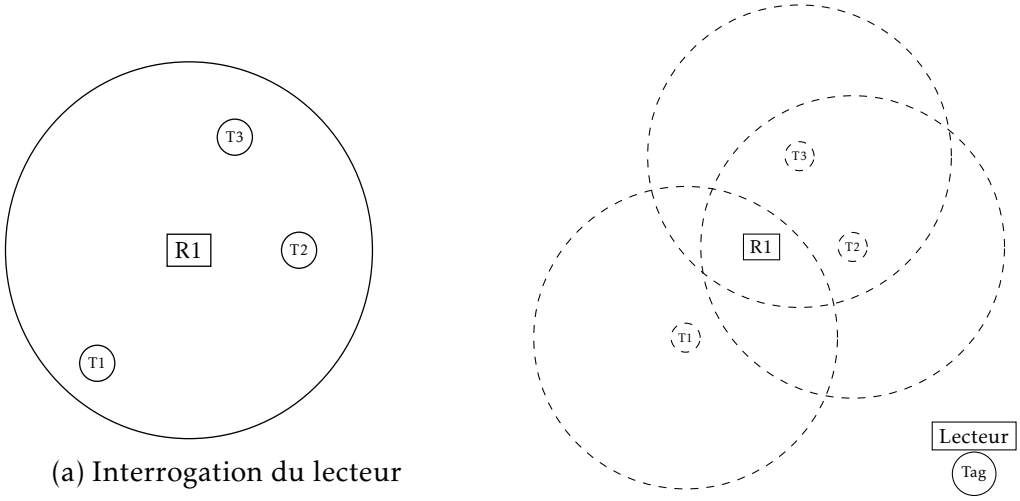
Avec le développement des applications basées sur la RFID, le déploiement de plus en plus de tags et de lecteurs a été nécessaire afin de s'assurer une large couverture mais également de pouvoir récupérer toujours plus d'informations. Cependant, ce large déploiement ne se fait pas sans entrave due à la nature sans fil des communications et provoque donc des **collisions**. En fonction de leur niveau d'occurrence et des acteurs impliqués (tags ou lecteurs), les collisions ont été classifiées en deux catégories : les collisions de tags et les collisions de lecteurs.

2.1.1 Collisions de tags RFID

Lorsque plusieurs tags sont à portée radio d'un lecteur, ils reçoivent tous le signal d'interrogation, sont excités et répondent simultanément au lecteur en question. Ces réponses concurrentes parviennent au lecteur qui ne peut les distinguer les unes des autres. On constate alors une erreur de lecture.

Aloha	Parcours d'arbre	Hybrides
Aloha pur [53]	Division d'arbres [55]	Arbre-slotté Aloha [84]
Aloha slotté [53]	Arbre de requête [39]	Arbre de requête hybride (HQT) [18]
Aloha fenêtré-slotté [87]	Recherche binaire [5]	Variantes HQT [40]
-	Arbitrage bit à bit [52]	Arbre de hash [94]

Tableau 2.1 – Protocoles d'anti-collision de tags



(a) Interrogation du lecteur

(b) Réponses des tags

Figure 2.1 – Collisions de tags RFID

La Figure 2.1 montre l'exemple d'un lecteur, R1, avec 3 tags {T1 ;T2 ;T3} à portée de lecture. Lorsque R1 diffuse son signal d'interrogation sur sa portée (voir Figure 2.1a, tous les tags répondent en même temps. On voit sur la Figure 2.1b que les différents signaux des réponses des 3 tags se superposent au niveau du lecteur. Ainsi les 3 tags ne seront pas reconnus et donc non identifiés. En ramenant cette collision à l'exemple d'un entrepôt avec plusieurs produits et les tags rattachés à ces derniers, cela reviendrait à manquer l'identification de produits qui pourraient être à des stades critiques de la chaîne. Afin de pallier ces collisions, une solution d'accès multiple doit être mise en place afin d'éviter que les réponses des différents tags n'interfèrent entre elles. Cet obstacle aux bonnes performances des solutions RFID a fait l'objet de plusieurs propositions d'algorithmes basés sur différentes techniques.

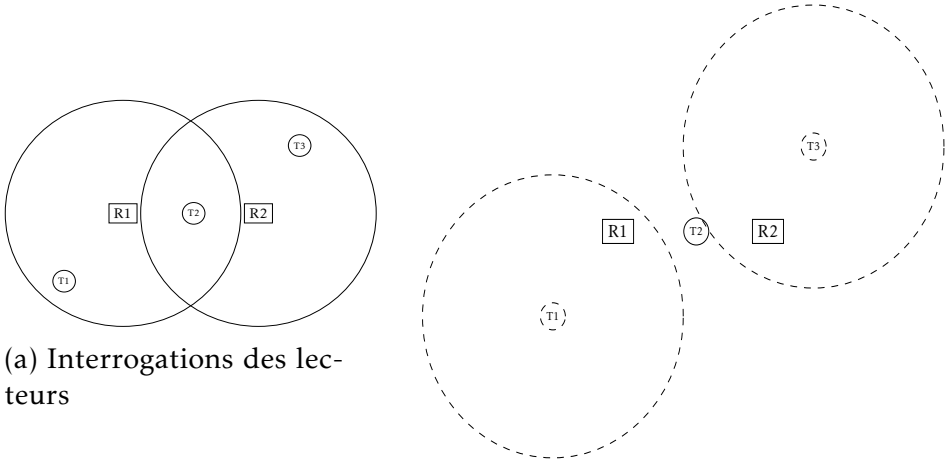
Le tableau 2.1 détaille les différentes catégories d'algorithmes proposés dans la littérature pour réduire les collisions de tags RFID. Dans [19], les auteurs font une revue des principaux algorithmes avec une évaluation permettant un choix guidé d'un algorithme d'anti-collision de tags RFID. On retiendra principalement 3 familles d'algorithmes à savoir :

— les algorithmes ALOHA fonctionnent sur la base d'un accès stochastique des tags au canal afin de transmettre leurs données au lecteur à portée. Ils fonctionnent bien avec un nombre réduit de tags en collisions mais sont plus laborieux dans des environnements denses ;
— les algorithmes de parcours d'arbres sont basés sur une interrogation à tour de rôle des différents tags. Les tags sont classifiés selon un arbre binaire à l'aide de leur identifiant, cet arbre sera par la suite parcouru par le lecteur afin d'éviter les collisions entre tags ;
— les algorithmes hybrides, comme le nom l'indique, font une combinaison des techniques ALOHA et de parcours d'arbre.

Dans cette thèse nous ne traitons pas de cette problématique mais plutôt des collisions de lecteurs.

2.1.2 Collisions de lecteurs RFID

Ces collisions sont le résultat de tentatives simultanées d'interrogations par deux ou plusieurs lecteurs de tags à portée radio. En effet, lorsque des signaux d'interrogations concurrents parviennent au niveau d'un tag RFID, ce dernier n'est pas en mesure d'interpréter les différentes requêtes. Conséquemment, le tag ne peut répondre et renvoyer son information. Cela entraîne donc encore une fois des erreurs de lecture qui peuvent être critiques.



(a) Interrogations des lecteurs

(b) Réponses des tags

Figure 2.2 – Collisions de lecteurs RFID

La Figure 2.2 illustre un exemple avec deux lecteurs R1 et R2 ainsi que 3 tags {T1 ;T2 ;T3}. D'abord sur la Figure 2.2a, on voit que les lecteurs R1 et R2 tentent simultanément d'interroger les tags à portée radio. Le signal d'interrogation de R1 est respectivement reçu par les tags T1 et T2 tandis que celui de R2 est reçu par T3 et également T2. Le tag T2 reçoit les deux signaux d'interrogations superposés et n'est en mesure de répondre à aucun des lecteurs. Ainsi, sur la Figure 2.2b, les tags T1 et T3 répondent respectivement aux signaux d'interrogation des lecteurs R1 et R2. Le tag T2 ne sera pas lu. En rapportant cet exemple à celui d'un point de vente avec des produits équipés de tags RFID, de telles collisions pourraient s'exprimer sous la forme de produits non validés en caisse ou perdus dans le stock, pouvant générer un réel manque à gagner.

Afin de remédier à ces collisions, plusieurs protocoles d'anti-collisions de lecteurs RFID ont été proposés. Néanmoins, celles-ci demeurent encore un inconvénient majeur des solutions RFID par opposition aux collisions de tags qui ont largement été étudiées avec plusieurs solutions qui sont maintenant intégrées aux solutions RFID du marché.

2.2 Présentation des métriques d'évaluation des protocoles d'anti-collision RFID

Afin d'appréhender les performances des différentes solutions proposées pour remédier aux collisions, il est nécessaire d'établir des métriques de mesures de performances. L'utilisation de ces métriques permet d'avoir une vision plus claire des protocoles proposés et de fournir une comparaison équitable de leurs performances. Dans [31], les auteurs proposent 10 métriques d'évaluations différentes, nous en retiendrons 5 dans la suite de nos travaux. Ce choix a été guidé par le fait que ces 5 métriques choisies sont non seulement les plus réutilisées dans la littérature, mais aussi parce-qu'elles sont suffisamment représentatives. Elles sont d'ailleurs pertinente pour évaluer d'autres technologies.

2.2.1 Débit de lecture

La plupart des solutions RFID sont mises en place pour effectuer des inventaires et donc arriver à lire un maximum de tags avec le moins de tentatives possibles. Le débit de lecture permet de comptabiliser le nombre d'accès fructueux au canal pour l'ensemble des lecteurs, sachant qu'un accès au canal réussi se traduit par une lecture des tags à portée. Ainsi en fonction du protocole d'anti-collision choisi, un **Accès Canal Réussi (ACR)** sera décompté à chaque fois qu'un lecteur remportera la contention pour avoir l'accès au canal et lire les tags à portée. La contention est ici définie comme la phase de compétition engagée par un lecteur RFID pour avoir l'accès au canal. La résolution de cette contention se déroulera en fonction de l'algorithme d'anti-collision défini. On pourrait supposer que plus le débit de lecture est important, plus les lecteurs ont accès au canal pour lire les tags. Cependant, cette métrique ne permet pas, à elle seule, d'inférer sur la distribution spatiale et temporelle des lectures au cours de l'activité des lecteurs. En effet, un débit de lecture élevé peut signifier que seul un sous ensemble des lecteurs accèdent régulièrement au canal tandis que les autres sont constamment en échec. Un autre exemple pourrait être qu'en fonction du déplacement des lecteurs, ils aient tous accès au canal à un instant donné et plus du tout à d'autres moments à cause d'une forte congestion. Il

est donc nécessaire de combiner le débit de lecture à d'autres métriques pour estimer la qualité de service d'un système RFID.

2.2.2 Collisions

Les collisions, comme exprimé en Section 2.1.2, se traduisent sur le plan applicatif par des erreurs de lecture et donc des tags non lus. Ces erreurs de lecture peuvent se révéler coûteuses dans le cadre d'applications commerciales (gestion de stock, point de vente, etc.) voire même critiques pour d'autres types d'applications (surveillance d'animaux, contrôle d'accès, etc.). Cette métrique permet donc d'identifier le nombre d'**Accès Canal Échoués (ACE)**. On comptabilisera donc un ACE à chaque fois qu'un lecteur échouera en contention pour l'accès au canal. Grâce à nos outils de simulation, nous pourrons comptabiliser les accès de prime abord réussis mais ayant conduit à des collisions dans les ACE. Un nombre important de collisions peut être interprété comme une perte d'énergie importante, ce qui va à l'encontre du principe d'utilisation de la RFID. En effet, lorsqu'un lecteur entre en contention pour l'accès au canal pour finalement échouer, il a perdu inutilement de l'énergie. En fonction de l'algorithme utilisé, il a même pu faire échouer d'autres lecteurs. Par ailleurs, cette contention se solvant par un échec impacte le délai de lecture des tags que nous verrons ultérieurement.

2.2.3 Efficacité

Cette métrique combine les deux précédentes afin de déterminer les dispositions des algorithmes à éviter les collisions et autoriser l'accès au canal aux lecteurs. L'efficacité se calcule comme suit :

$$Eff = \frac{ACR}{(ACR + ACE)}$$

La distribution temporelle des ACR pointée plus haut comme indéterminée avec le débit de lecture, peut ici être inférée. En effet, en combinant les ACR et ACE, on arrive à définir la distribution des accès au canal en fonction des contentions. Ainsi, avec un débit de lecture faible, exprimant un accès au canal difficile car

congestionné, les collisions sont nombreuses et donc l'efficacité reste faible. Par contre, si l'efficacité et le débit de lecture sont tous deux élevés, on peut en déduire que les collisions sont faibles et donc que l'accès au canal est plus libre.

2.2.4 Indice d'équité de Jain (IEJ)

Cette métrique, très partagée dans plusieurs domaines [44], permet d'apprécier l'équité dans l'accès au canal entre les différents lecteurs. L'IEJ se calcule comme suit :

$$IEJ = \frac{|\sum_{i=1}^{n} ACR_i|^2}{n \times \sum_{i=1}^{n} ACR_i^2}$$

L'IEJ permet d'avoir une idée de la distribution spatiale des lectures ce qui est impossible avec le débit de lecture. En effet, si les lecteurs ont tous les mêmes valeurs en terme d'ACR, l'IEJ est égale à 1. A l'inverse, dans le pire cas, un seule lecteur concentre tous les ACR et l'IEJ vaut $1/n$. Ainsi, plus les ACR seront équitablement répartis, plus la valeur de l'IEJ se rapprochera de 1. Sinon elle tendra vers 0. Dans le cadre d'applications de logistique, de bonnes performances en IEJ permettent d'affirmer que l'ensemble des produits sous surveillance sont vérifiés régulièrement et qu'aucun groupe de produits n'est délaissé. Dans l'ensemble, un indice d'équité performant renseigne sur une répartition équitable de la consommation énergétique des lecteurs. Dans le cas où ils seraient alimentés par une batterie, l'ensemble des batteries se déchargeraient de manière plus ou moins synchronisée en fonction de la durée de l'activité des lecteurs.

2.2.5 Délai de couverture

Cette métrique permet de connaître le temps nécessaire pour interroger l'ensemble des tags à portée de lecture. Quand bien même l'Indice d'Equité de Jain permet d'avoir une idée sur les performances de l'ensemble des lecteurs, il ne garantit pas une lecture rapide des tags déployés. Dans l'exemple de l'application logistique dans une entrepôt avec des produits alimentaire, si un produit

périmé est recherché dans l'ensemble, il est nécessaire de pouvoir le retrouver au plus vite. Ainsi utiliser un algorithme offrant un délai de couverture rapide est préférable. En fonction de l'application concernée, un temps de couverture élevé peut être considéré comme un inconvénient.

2.3 Etat de l'art sur les protocoles d'anti-collision de lecteurs RFID

Comme rapporté en Section 2.1.2, de nombreux algorithmes d'anti-collision de lecteurs ont été proposés et sont disponibles dans la littérature. Afin de faciliter la vue d'ensemble sur ces différents protocoles, nous proposerons ici une catégorisation en fonction de leur modus operandi. Par la suite, nous établirons un catalogage suivant différents paramètres permettant de guider le choix d'un algorithme ou d'un autre en fonction des contraintes de l'application à servir.

2.3.1 Catégorisation des protocoles

En fonction de leur procédure de contention, les protocoles d'anti-collision de lecteurs peuvent être classés en deux catégories : les **accès multiple à répartition dans le temps (Time Division Multiple Access ,TDMA)** ou l'**écoute d'un Support à Accès Multiple (Carrier Sense Multiple Access, CSMA)**. Ces deux catégories, peuvent elles-mêmes être sous découpées en fonction de l'organisation des lecteurs et le fait que les protocoles soient **centralisés** ou **distribués**. En plus de cette catégorisation, d'autres paramètres moins déterminants peuvent également être considérés comme l'utilisation d'un canal unique ou de plusieurs canaux.

TDMA ou CSMA

Afin d'accéder au canal et interroger les tags, les lecteurs peuvent soit suivre une méthode de réservation suite à un découpage temporel ou "écouter" le canal pour s'assurer de son inactivité au préalable.

Dans le premier cas, on parlera d'algorithmes TDMA. Dans cette configuration, le temps d'activité des lecteurs est divisé en unités de temps appelées *timeslots*. Ces intervalles de temps sont attribués ou choisis par les lecteurs, en fonction du protocole d'anti-collision en cours, pour accéder au canal et interroger les tags. Ce schéma garantit qu'un seul lecteur fonctionne dans son voisinage pendant son timeslot. Aucune collision n'est possible puisque les lecteurs voisins ont soit choisi des intervalles de temps différents, soit été désactivés pendant la procédure de contention. De tels systèmes permettent une meilleure gestion de l'énergie puisque les lecteurs peuvent *dormir* et rester inactifs jusqu'à ce que leur timeslot arrive, s'activer puis retourner au repos garantissant un bornage temporel de l'activité du système.

Dans le deuxième cas où les lecteurs vont *écouter* l'activité sur le canal pour s'assurer de son inactivité, on parlera d'algorithmes CSMA. Puisque le médium est partagé entre les lecteurs, ils peuvent sentir l'activité de leurs voisins. Si le canal est inactif, ils pourront interroger les tags sans collision. Dans le cas où le médium est occupé, les lecteurs attendent une période aléatoire ou prédéfinie avant de vérifier à nouveau l'activité du canal. Cela garantit une meilleure flexibilité vu qu'aucune répartition préalable n'est à effectuer. La mise à l'échelle est plus simple. Cependant, le principal défi de cette solution est que dans le cas d'un déploiement dense de lecteurs, le temps d'attente pour un lecteur peut être long avant qu'il n'ait la chance d'accéder au canal afin d'interroger les tags, ce qui en retour affecte les performances du système. Cela rend ces protocoles inadaptés aux applications impliquant l'utilisation de tags mobiles. Dans le cadre d'une application qui propose de rattacher des tags à des véhicules dans une ville pour avoir un sens de la traçabilité, si on observe de longs temps d'attente pour les lecteurs, avec des voitures en mouvement rapide, cela pourrait signifier que plusieurs voitures ne seraient pas identifiées.

Bien que nous ayons réparti les propositions selon ces deux schémas, il est important de noter que certaines approches reposent également sur des techniques de **Frequency Division Multiple Access (FDMA)** pour éviter les collisions. En effet, grâce aux différentes fréquences introduites par ETSI EN 302.208 [1], les lecteurs peuvent fonctionner sur différents canaux pour éviter les collisions de lecture. Néanmoins, ne dépendre que de ces différents canaux

pour éviter les collisions ne suffit pas compte tenu de l'éventuel déploiement dense des lecteurs RFID et de leur proximité. Certaines propositions ont pris en compte l'aspect multicanal, comme on le verra par la suite, mais toujours sur un schéma TDMA ou CSMA pour planifier les opérations.

Centralisé ou Distribué

Afin d'assurer la coordination nécessaire entre les lecteurs pour éviter les collisions, une forme de communication doit être établie entre les lecteurs eux-mêmes ou avec une entité supérieure chargée de leur synchronisation. Le choix de cette forme de communication définit non seulement la nature de l'algorithme mais affecte également ses performances.

Dans le cas centralisé, les lecteurs communiquent avec une entité supérieure (serveur central) responsable de la planification des opérations. Le serveur central est capable, après avoir recueilli toutes les informations concernant la topologie des lecteurs, de calculer un schéma de lecture optimal réduisant les collisions. Cependant, en général, l'utilisation d'un serveur central limite la mobilité des lecteurs au détriment d'un niveau de calcul et de latence plus élevé que dans le cas d'un algorithme distribué. Un lien de communication doit également être établi entre les lecteurs et cette entité supérieure. Ajouté à cela, avoir des lecteurs dépendant d'une entité supérieure pour toute opération rend les solutions moins réactives. Les solutions dépendant de l'utilisation d'un serveur central se trouvent généralement dans les schémas basés sur du TDMA.

Dans une configuration distribuée, les lecteurs communiquent directement les uns avec les autres et localement (dans le temps et l'espace) pour s'accorder de pair à pair sur leurs activités et réduire les collisions. Dans le cas sans-fil, les lecteurs sont capables d'échanger avec leurs pairs dans l'étendue de leur portée de communication définissant leur voisinage. Cela permet aux solutions basées sur ce paradigme d'être évolutives et de supporter des changements dynamiques en topologie comme ce serait le cas avec les lecteurs mobiles. Chaque décision prise par un lecteur donné est dictée par sa connaissance de l'état son voisinage à un moment donné. Des solutions distribuées sont trouvées dans les algorithmes TDMA et CSMA.

Monocal ou Multicanal

Aux débuts de la RFID, tous les lecteurs devaient identifier les tags en utilisant un seul canal commun. Ce média monofréquence est devenu une ressource rare avec des déploiements denses où plusieurs lecteurs sont à proximité et entraînent des collisions croissantes. Pour pallier cette déficience, le multicanal a été introduit dans la mise à jour de la norme apportée par le standard [1]. Les lecteurs sont maintenant en mesure d'interroger les tags sur quatre canaux différents, ce qui rend les lectures de tags moins compétitives et moins sujettes à des collisions. En effet, en affectant efficacement ces canaux aux lecteurs situés dans le même voisinage, la capacité peut être multipliée par quatre fois, améliorant ainsi l'efficacité des systèmes RFID. Cependant, avoir plus de fréquences n'empêche pas les signaux issus des lecteurs RFID d'entrer en collision. En effet, les tags sous lecteurs concurrents, même sur des canaux différents, ne peuvent toujours pas être lus et sont soumis aux interférences sur canaux adjacents [3].

2.3.2 Protocoles TDMA

Comme décrit plus-haut, les approches TDMA sont retrouvées aussi bien dans le cadre d'algorithmes centralisés que distribués. Dans cette section, nous passerons en revue les principaux protocoles TDMA de l'état de l'art.

Approches Distribuées

La plupart des approches d'anti-collision RFID TDMA distribuées dérivent d'un algorithme antérieur appelé **Distributed Color Selection** et utilisent un canal de communication dédié entre lecteurs pour organiser leur activité.

Distributed Color Selection (DCS) Avec DCS [82], les lecteurs réservent périodiquement des intervalles de temps (ici appelés *couleurs*) en choisissant au hasard parmi la gamme de couleurs disponibles. Ces intervalles de temps sont ensuite utilisés pour communiquer avec les tags. Si deux ou plusieurs lecteurs voisins choisissent les mêmes couleurs, leurs signaux entrent en collision et les tags couverts sont manqués. En cas de collision, les lecteurs concernés sélec-

tionnent de nouvelles couleurs parmi celles disponibles et envoient un message *kick* aux voisins pour réserver l'intervalle de temps pour la ronde d'interrogation suivante. Tous les lecteurs de la couleur correspondante au *kick* doivent passer à un intervalle de temps différent pour le tour suivant. Le nombre de couleurs disponibles est fixe et est donné au début. Les lecteurs disposent donc de deux interfaces de communication : la première pour interroger les tags et la deuxième pour la coordination locale entre voisins. L'inconvénient principal de cet algorithme est qu'en fonction du nombre maximum de couleurs disponibles, le système RFID est fortement affecté. En effet, si la valeur des couleurs max est trop faible, un grand nombre de lecteurs finissent par choisir les mêmes couleurs et entrent en collision, alors que si la valeur est trop élevée, certains intervalles de temps ne sont pas occupés et le débit et le délai de couverture sont impactés.

Colorwave Aussi connu sous Variable-maximum DCS [82], cet algorithme résout le problème principal de DCS. Comme son nom l'indique, il permet de modifier le nombre de couleurs maximum disponibles tout au long de la vie du système RFID. Afin de définir la valeur du nombre de couleurs maximal en fonction de l'état du réseau, deux variables seuils sont introduites *UpSafe* & *DnSafe*. Chaque lecteur surveille son nombre d'interrogations réussies. Selon qu'il atteigne la valeur de *UpSafe* ou *DnSafe*, les lecteurs augmentent ou diminuent respectivement leur valeur locale du maximum de couleurs disponibles et envoient un message *kick* contenant la nouvelle couleur choisie. Cependant dans un voisinage proche où plusieurs lecteurs entrent en collision, une fois qu'ils atteignent une valeur seuil, ils envoient tous des messages *kick* pour réserver leurs couleurs, générant une vague de mise à jour de couleurs d'où le nom Colorwave.

Probabilistic DCS (PDCS) PDCS [32] est une autre version améliorée et probabiliste de DCS et la première dérivée à proposer une solution multicanal. Un paramètre p est introduit comme la probabilité pour un lecteur de changer sa couleur après une collision. Trois cas sont donc possibles :

— cas 1 : les lecteurs impliqués dans la collision ne changent pas de couleur, ils envoient des messages *kick* qui inciteront les lecteurs voisins à changer de couleur ;

— cas 2 : un des lecteurs change de couleur et envoie un message *kick* pour réserver la nouvelle couleur, l'autre lecteur interroge les tags avec la couleur précédente sans changer ;
— cas 3 : les lecteurs changent de couleur, dans ce cas ils envoient des messages *kick* et réservent leurs nouvelles couleurs, c'est l'algorithme basique de DCS.

Cependant, comme pour DCS, le nombre maximum de couleurs est fixe, induisant les mêmes problèmes ainsi, une version basée sur Colorwave a également été proposée.

Distributed Color Non-cooperative Selection (DCNS) DCNS [30] est encore un autre algorithme dérivé de Colorwave. La première différence avec Colorwave est que les lecteurs n'envoient pas de messages *kick* mettant à jour leur valeur du nombre de couleurs maximal, ici nommée μ. Un autre paramètre introduit est η qui détermine la probabilité pour un lecteur d'interroger les tags une fois sur son créneau temporel. Les lecteurs sont classés en trois différents types :

— *killer* pour $\mu == 2$, avec une gamme de couleurs si faible, ces lecteurs interrogent fréquemment les tags, donc ils n'envoient pas de *kick* ni ne changent de canal pour éviter les collisions avec les autres *killer* voisins ;
— *normal* pour $2 < \mu < threshold$, ces lecteurs agissent comme des lecteurs standards suivant Colorwave ;
— *killed* pour $\mu > threshold$, ces lecteurs envoient constamment des *kick* et interrogent rarement les tags, ils augmentent leur valeur de η afin d'augmenter leurs chances d'interrogation.

Cette configuration fait que le système RFID n'atteint jamais vraiment d'état de convergence stable. Dans le cas d'utilisation de lecteurs mobiles, le délai de couverture peut être affecté si les tags éloignés sont couverts par des lecteurs en position *killed*.

MAximum LIkelihood COlorwave (MALICO) MALICO [11], apporte encore une autre amélioration à Colorwave par rapport à sa convergence. Au lieu de s'appuyer sur un ensemble de seuils et de déclencheurs entrés manuellement, les lecteurs mettent automatiquement à jour leur nombre de couleurs disponibles

pour réduire les collisions. La mise à jour est effectuée par chaque lecteur suite à l'observation des contentions réussies, en collision et au repos dans la ronde précédente pour estimer le nombre de voisins. Sur la base de cette estimation, un certain nombre de couleurs disponibles est défini pour optimiser le débit de lecture. Ce processus a l'avantage de rejeter les phases de *kick* présentes dans Colorwave pour augmenter les interrogations et le débit. Cependant, les lecteurs utilisant MALICO doivent disposer d'antennes bi-statiques afin d'écouter et d'enregistrer les collisions possibles pendant qu'ils accèdent au canal. En raison de l'écoute et du calcul de chaque lecteur, la performance est liée à la vitesse des lecteurs dans les déploiements mobiles. Si le calcul peut être fait pour un état donné de voisins lors d'un tour d'interrogation donné, cet état change malheureusement en raison de la mobilité lors de la ronde suivante. Toutefois, grâce à la simplicité du processus, le coût de calcul peut être maintenu plutôt bas et n'a pas d'impact sur les rondes d'interrogation.

Approches Centralisées

Ces approches dépendent de l'utilisation d'un serveur central chargé d'allouer les timeslots aux différents lecteurs soit aléatoirement en suivant une distribution donnée, soit sur la base d'informations recueillies sur le système telles que la position des lecteurs, leur voisinage ou leur activité sur les tours précédents.

Neighbor Friendly RFID Anticollision (NFRA) Dans NFRA [24], les interrogations des lecteurs sont organisées en rondes coordonnées par un serveur. Ce dernier arrange les différentes rondes grâce à la diffusion d'une *Commande d'arrangement (Arrangement Command)* annonçant le nombre maximum de timeslots disponibles. À sa réception, chaque lecteur sélectionne aléatoirement un intervalle de temps et attend la *Commande d'Ordonnancement (Ordering Command OC)* du serveur correspondant au timeslot choisi. A la réception de l'OC correspondant, le lecteur diffuse un message à ses voisins. Si aucune collision n'est observée pendant cette diffusion, le lecteur envoie un *Message de Priorité (Overriding Frame OF)* pour désactiver tous les voisins de la ronde en cours.

Quand un lecteur reçoit un OF, il attend le prochain AC du serveur pour être de nouveau en contention. Dans les déploiements très denses, cet algorithme a un nombre élevé de lecteurs désactivés dû aux OF. Par ailleurs, les lecteurs ayant choisi une valeur de timeslot faible sont privilégiés par rapport aux autres.

NFRA+ & NFRA ++ Ces propositions [27] sont des versions améliorées de NFRA comme leur nom l'indique. Le premier NFRA+, corrige l'inconvénient mentionné ci-dessus en ce qui concerne les valeurs élevées de timeslots. Cet algorithme tente d'améliorer l'équité en augmentant la priorité des lecteurs ayant eu un long temps d'attente sans interrogation de tags. Cette augmentation de priorité renforce la probabilité pour le lecteur de réduire la valeur de son timeslot lors des rondes suivantes et vice versa pour les lecteurs à faible priorité avec un faible temps de demande. Cela affecte l'équité de l'algorithme mais pas le nombre élevé de lecteurs handicapés à chaque ronde. NFRA++ tente de corriger ce problème en offrant une seconde chance aux lecteurs entrés en collision précédemment. Dans cet algorithme, après avoir envoyé l'OC correspondant à la ronde en cours, le serveur envoie alors un autre OC aux lecteurs qui étaient entrés en collision pour leur donner une deuxième chance. Les lecteurs déterminent alors une probabilité T de diffuser un message à ses voisins lors de cet OC. Cela donne une chance de collision avec une probabilité T. L'équité renforcée et le deuxième OC offrant une deuxième chance aux lecteurs en collisions permettent à cet algorithme d'avoir de bien meilleures performances (cf Section 2.3.4).

Geometric Distribution Reader Anti-collision (GDRA) Avec NFRA et ses variantes, les lecteurs ayant une faible valeur de timeslot ont plus de chances d'accéder au canal et moins de chances de recevoir un *OF* qui va les désactiver. GDRA [13] corrige cet inconvénient en utilisant une distribution géométrique appelée Sift [45] pour le choix des timeslots en lieu et place de la distribution uniforme classique. En utilisant cette distribution géométrique, un sous-ensemble de lecteurs sélectionnent une valeur de timeslot faible tandis que tous les autres sélectionnent des valeurs plus élevées. Cela permet de réduire considérablement le nombre de collisions qui vont désactiver les lecteurs pour le cycle en cours. Cet algorithme est très performant mais joue toujours sur le nombre élevé de

lecteurs désactivés par les OF dans les déploiements denses.

Fair Reader Collision Avoidance (FRCA) Dans [72], les auteurs proposent deux versions de leur algorithme. Ces deux versions sont basées sur des observations des lacunes de NFRA et de GDRA et tentent de les corriger. Dans FRCA1, les lecteurs suivent le même schéma que dans NFRA. Cependant, en cas de collision, lorsqu'ils ont choisi les mêmes timelsots, les lecteurs observent leur nombre de succès précédents. Le lecteur ayant le taux le plus bas a accès au canal et l'autre patiente pour la prochaine ronde. Cela permet au protocole d'être équitable en ce qui concerne l'accès au support partagé. Dans FRCA2, en plus de la comparaison de taux de réussite, les lecteurs estiment la distance qui les sépare en fonction de la puissance de signal reçue lors de l'échange de messages. Sur la base de cette distance entre eux, le lecteur en échec avec le taux de réussite plus élevé peut interroger les tags couverts par son voisin sur un canal différent. Les auteurs proposent également l'utilisation de la distribution Sift afin de diminuer le nombre de lecteurs en contention sur les valeurs de timeslots faibles.

Adaptive Color-based Reader Anticollision Scheduling (ACoRAS) Dans ACoRAS [36], au lieu que les lecteurs choisissent aveuglément un timeslot dans une plage disponible, ces derniers sont assignés directement par le serveur après la construction d'un *Minimum Independent Set*. En utilisant les informations sur la topologie des lecteurs (voisins en collision, position des lecteurs, tags à portée, ...), le serveur affecte un ensemble de couleurs (timeslots) à chaque lecteur pour s'assurer qu'il n'entrera pas en collision avec les autres dans son voisinage. Un algorithme d'optimisation est ensuite exécuté par le serveur pour réduire le nombre de couleurs attribuées et donc par là-même réduire le temps de latence et d'inactivité du système. Ceci est fait en tirant parti des tags couverts par plusieurs lecteurs à la fois. Ainsi, si un tag est couvert par plusieurs lecteurs en même temps, le serveur n'affecte qu'une couleur à l'un des lecteurs et désactive les autres pour l'emplacement actuel. Bien que cet algorithme puisse fournir très peu ou pas de collisions enregistrées, il repose sur un surcoût en calcul élevé (problème NP-difficile) afin de déterminer la distribution optimale des couleurs

et ne peut pas être pris en compte pour un déploiement mobile aléatoire.

Geometric Distribution-based Multiple Readers Scheduling Optimization Algorithm using Artificial Immune System (GD-MRSOA-AIS) Comme son nom l'indique, dans [21], les lecteurs utilisent une distribution géométrique, Sift [45], afin de réduire le nombre de voisins en conflit sur les timeslots inférieurs. En plus de cela, une optimisation mimant un système immunitaire artificiel est introduite afin d'améliorer le schéma d'ordonnancement. La plage d'interrogation est alors plus efficace en utilisant l'algorithme correspondant. Cependant, cet algorithme repose sur la connaissance de la topologie et les positions des lecteurs et ne gère pas les scénarios d'applications mobiles.

Centralized and Aligned Scheduler compatible with EPCGlobal (CASE) Dans CASE [10], les ressources sont allouées aux lecteurs par un serveur central qui a une vision globale de l'ensemble du déploiement. En plus de fonctionner à des instants différents, les lecteurs fonctionnent également sur les différentes fréquences autorisées par ETSI EN 302-208 [1]. En fonction de la position des lecteurs, supposée connue, le serveur alloue des slots et des fréquences aux lecteurs en fonction de leur priorité et de leurs distances relatives. Cette allocation est réalisée en résolvant un problème de MILP (Mixed Integer Linear Programming) qui vise à : *(i)* maximiser le nombre total de slots alloués et *(ii)* assurer une distribution équitable de la ressource disponible. En utilisant un facteur de pondération α pour maximiser le débit et/ou l'équité et un ensemble de contraintes, le serveur est capable d'optimiser l'ordonnancement des interrogations des lecteurs. La proposition a été testée avec différents niveaux de densité et les performances en termes de débit chutent rapidement avec un nombre croissant de lecteurs. Le coût de calcul est également une caractéristique qui a été examinée et les résultats montrent qu'il augmente exponentiellement avec le nombre de lecteurs déployés, la densité de déploiement, la mobilité ou les niveaux de priorité abordés.

2.3.3 Protocoles CSMA

Ces protocoles, comme annoncé plus-haut, sont basés sur une écoute du canal pour s'assurer de sa disponibilité avant activité et interrogation des tags à portée. Comme on le verra par la suite, certains algorithmes nécessitent d'implémenter 2 interfaces de communications sur les lecteurs afin d'interroger les tags et converser avec les autres lecteurs.

Approches Distribuées

Étant donné que la disponibilité du canal est une variable locale à chaque lecteur dans son voisinage direct, la plupart des approches CSMA sont distribuées. Ainsi les lecteurs interagissent en fonction de l'activité dans leur voisinage.

Listen Before Talk (LBT) C'est le protocole standard pour les collisions de lecteurs [25]. Dans cette proposition, les lecteurs "écoutent" le canal pendant une période définie avant d'essayer de "parler" (interroger les tags). Si jamais le média est occupé, le lecteur bascule sur un canal différent et effectue une nouvelle session d'"écoute". Cependant, dans le cas de déploiements très denses de lecteurs, ces derniers peuvent être bloqués dans une boucle d'écoute en essayant de trouver un canal de communication disponible. De plus, ce protocole peut présenter des performances très médiocres dans un scénario mobile, car un lecteur ayant écouté le canal dans une disposition donnée ne pourrait pas assurer un canal disponible tout au long de son trajet d'interrogation. Inversement, ce n'est pas parce-que le lecteur se trouve dans une zone radio occupée lors de son écoute que le canal ne sera pas diponible sur le reste de son trajet.

Pulse Ce protocole [8], tout comme LBT, permet aux lecteurs d'écouter le média avant d'interroger les tags. Cependant, dans ce cas, pour éviter les problèmes d'écoute, les lecteurs «pulsent» constamment un signal pour alerter leurs voisins pendant le fonctionnement. Lorsqu'un lecteur reçoit la "pulsation" d'un voisin, il se désactive et attend que le support soit disponible. Cela a l'avantage d'assurer qu'un seul lecteur interroge les tags dans un voisinage donné. Néanmoins, dans un environnement mobile dense, les lecteurs qui envoient une «impulsion»

pourraient finir par désactiver un grand nombre de leurs voisins inutilement, ce qui aurait un impact considérable sur le débit et l'efficacité du système.

Anti-collision Protocol for RFID (APR) Ce protocole [71] utilise des échanges de messages entre lecteurs pour estimer la distance entre eux et réduire les collisions sur les tags couverts. En effet, après une période de backoff, les lecteurs diffusent un message dans leur voisinage. La période de backoff est calculée à partir du niveau d'énergie résiduelle de chaque lecteur afin d'éviter les collisions de diffusion. Les messages diffusés informent les voisins sur les tags couverts. En fonction de la puissance du signal reçu, les lecteurs estiment la distance entre eux. Après avoir interrogé les tags à portée, les lecteurs s'échangent ensuite les informations entre voisins. Bien que cela puisse aider à réduire les collisions, l'estimation de la distance entre les lecteurs en fonction de la puissance de signal reçu est sujette à des erreurs. De plus, les tags qui sont couverts par un seul lecteur en collision avec des voisins en couverture sur d'autres tags sont désavantagés.

High Adapative MAC (HAMAC) Dans cet algorithme [3], les lecteurs attendent une période de backoff aléatoire dans une fenêtre de contention (Contention Window CW) prédéfinie avant d'écouter le canal. Si le support est occupé, le lecteur passe à un canal différent et vérifie s'il est disponible. Si tous les canaux sont occupés, le lecteur divise alors la CW par 2 et tire aléatoirement un nouveau backoff. La nouvelle taille de CW permet au lecteur d'avoir une période de backoff plus courte augmentant sa priorité. Le lecteur effectue ensuite à nouveau la vérification sur tous les canaux. Ce processus d'écoute des canaux et division de la fenêtre de contention est conservé jusqu'à une taille minimale de CW. Une fois cette valeur atteinte, le lecteur recommence avec la taille CW la plus élevée. Bien que cette solution tende à augmenter les chances d'accès au canal et d'interrogation des lecteurs, le fait d'avoir des tailles de fenêtres de contention CW différentes chez les lecteurs peut augmenter la latence pour certains d'entre eux, ce qui a un impact négatif sur l'équité.

Distributed Multi-Channel Collision Avoidance (DiMCA) Légèrement différent des algorithmes CSMA précédents, le protocole DiMCA [73] propose aux lecteurs d'échanger des messages sur deux canaux de contrôle différents fonctionnant sur des portées différentes. Le premier canal couvre la portée d'interrogation du lecteur où les messages contenant l'ID du lecteur sont envoyés et le deuxième canal couvre la plage d'interférence où les messages contenant à la fois l'ID du lecteur et le canal d'interrogation choisi sont envoyés. Avant d'interroger des tags, un lecteur attend une période aléatoire au cours de laquelle il peut recevoir des messages sur les canaux de contrôle. Ainsi, selon le type de message reçu, un lecteur conserve deux listes de voisins en collision : ceux avec lesquels il peut fonctionner en même temps mais sur une fréquence différente et ceux pour lesquels il doit impérativement fonctionner à des instants différents. Avant de lancer son opération d'interrogation, un lecteur vérifie sa file d'attente et, selon l'état, choisit un canal différent pour opérer et le diffuse à ses voisins ou attend un signal END de ses voisins pour fonctionner à un autre moment. Bien que cette solution améliore à la fois le débit et l'efficacité du système RFID, elle repose sur un surcoût créé par les messages échangés, ce qui peut avoir un impact sur le retard. Les auteurs n'abordent pas non plus la façon dont ils évitent les collisions concernant les messages échangés.

Enhanced Distributed Multi-Channel (EDMC) EDMC [90] propose une version améliorée de DiMCA en demandant aux lecteurs de vérifier s'ils ont reçu d'autres messages des voisins après avoir choisi leur canal et avant d'envoyer leur propre message. Cela réduit le risque de collisions de messages ou de messages mal reçus de la part des voisins juste avant l'interrogation des tags. Les auteurs prétendent améliorer légèrement le délai ainsi que réduire les collisions en utilisant cette technique par rapport à DiMCA.

Efficient Multichannel Reader Collision Avoidance (EMRCA) EMRCA [47] est une amélioration de [8] pour prendre en compte l'aspect multicanal introduit par la norme. Les auteurs identifient deux types de collisions en fonction des portées d'interrogation et d'interférence des lecteurs. Les lecteurs commencent par détecter le canal de contrôle commun utilisé par tous les nœuds pour communi-

quer. Si aucune activité n'est détectée pendant une période donnée, le lecteur commence la phase de contention. Sinon, en fonction de la source d'activité, soit il commence une nouvelle session d'écoute à la fin de l'activité en cours, soit il poursuit le décompte avant d'entrer en contention. Pendant la contention, les lecteurs attendent un backoff aléatoire. Si un lecteur reçoit un message pendant ce backoff, il revient à la détection du canal de contrôle. Sinon, si le backoff s'épuise sans aucune réception de message, le lecteur passe à l'interrogation des tags à portée. Il occupe alors le canal d'interrogation choisi et diffuse périodiquement un message d'annonce sur le canal de contrôle commun. Ce protocole améliore l'équité et l'efficacité globales de Pulse mais souffre encore de la mobilité et de la forte densité de déploiement des lecteurs.

Approches centralisées

Les protocoles CSMA reposent sur les lecteurs qui écoutent le canal d'interrogation comme mentionné. Ainsi, avoir un serveur central est superflu puisque les lecteurs peuvent être autonomes dans de telles configurations. Cependant, quelques solutions CSMA sont encore centralisées en utilisant un serveur pour allouer des ressources aux lecteurs sur la base d'une vue globale et de l'historique du système RFID. C'est le cas de l'algorithme HiQ. Dans HiQ [42], 3 entités sont définies : lecteurs, serveurs R et serveur Q. Le nombre et le type de collisions enregistrées sont communiqués par les lecteurs au serveur Q. Sur la base de ces informations et en utilisant un algorithme Q-learning, le serveur Q définit ensuite les slots et la distribution de fréquences optimale pour les lecteurs. Un ensemble de slots qui optimise le fonctionnement des lecteurs est ensuite donné aux serveurs R. Les lecteurs reçoivent ensuite des emplacements par leur serveur R répondant dans la plage disponible. Cet algorithme décrit simplement la construction de cette topologie à trois niveaux. La complexité de l'algorithme d'apprentissage pour trouver une solution optimale est corrélée à la densité de déploiement des lecteurs. Dans le cas d'un réseau dense, la complexité peut induire un surcoût important. De plus, cet algorithme ne conviendrait pas dans un scénario mobile avec des paramètres changeant de façon permanente.

	Centralized	**Distributed**
TDMA	ACoRAS[36] FRCA[72] GD-MRSOA-AIS[21] GDRA[13] NFRA NFRA++[27] CASE[10]	DCS[82] Colorwave[82] DCNS[30] MALICO[11] PDCS[32]
CSMA	HiQ[42]	APR[71] DiMCA[73] EDMC[90] HAMAC[3] LBT[1] Pulse[8] EMRCA[47]

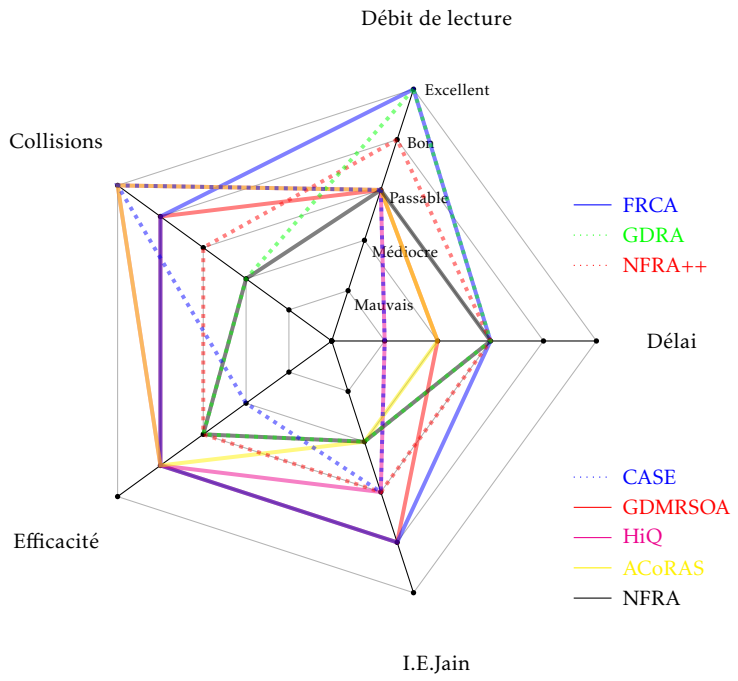
Tableau 2.2 – Récapitulatif des protocoles d'anti-collision de lecteurs RFID

2.3.4 Évaluation de Performances des protocoles proposés

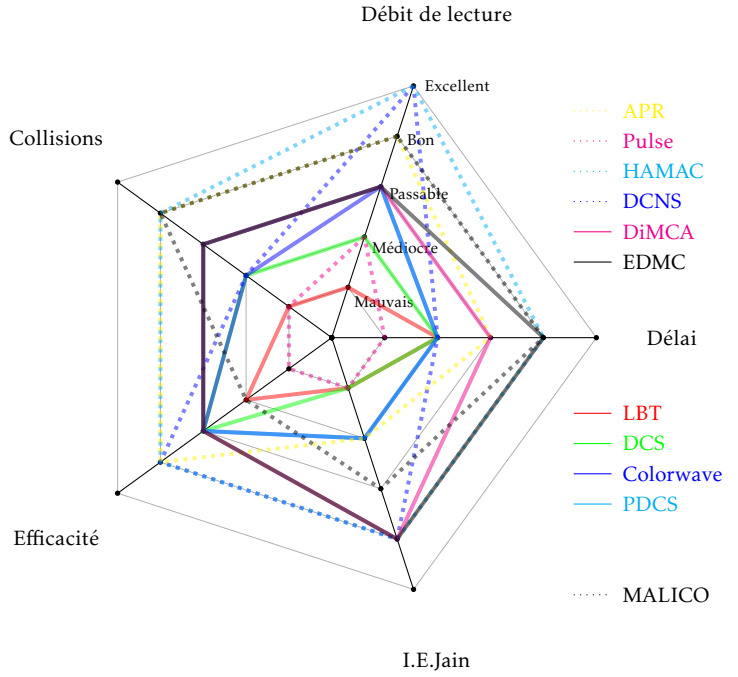
Au vu de la classification proposée des protocoles d'anti-collision, il est donc possible de récapituler l'ensemble des propositions présentées tel que dressé dans le Tableau 2.2.

Sur la base des métriques proposées en Section 2.2, des résultats présentés dans chacun des manuscrits étudiés et introduits dans la Section 2.3 mais également d'une analyse du fonctionnement et comportement des algorithmes introduits, nous proposons une évaluation des performances des protocoles d'anti-collision.

Les diagrammes de la Figure 2.3 permettent d'avoir une vue globale des performances de chaque protocole en fonction des métriques étudiées. Plus la courbe concernant un protocole se rapproche des bordures extérieures, plus les performances sont bonnes. Le diagramme de la Figure 2.3a illustre les protocoles centralisés. De prime abord, on remarque que les performances en terme de délai de couverture sont faibles pour l'ensemble des protocoles. Cela est dû au fait que les protocoles centralisés nécessitent généralement une phase de collecte d'information avant d'attribuer des timeslots de fonctionnement, ce qui a tendance à rajouter un sur-coût en terme de latence. Les protocoles de type NFRA



(a) Protocoles centralisés



(b) Protocoles distribués

FIGURE 2.3 – Évaluation de performances

ou GDRA sont également impactés par le fait qu'un grand nombre de collisions enregistrées désactive de nombreux lecteurs ce qui par conséquent augmente le délai de couverture des tags à portée et réduit considérablement l'index d'équité de Jain. Concernant les résultats des protocoles distribués, ils sont illustrés sur la Figure 2.3b. On peut y observer que les performances en termes de délai de couverture et d'équité sont généralement meilleures au détriment d'un plus grand nombre de collisions enregistrées. On remarquera que plus le débit de lecture et l'index d'équité sont profitables, moins le protocole sera apte à éviter les collisions. Un protocole tel que DCNS permet d'obtenir de bonnes performances en termes de débit de lecture, collisions et d'index d'équité. Il souffre cependant du délai de convergence nécessaire pour atteindre de bonnes performances, ce qui peut se voir sur son délai de couverture.

On peut donc en conclure qu'en fonction des contraintes en termes de débit de lecture, collision, délai de couverture d'une application, le choix d'un protocole ou d'un autre devra être adapté judicieusement. Une solution maximisant l'ensemble des métriques n'est malheureusement pas disponible au vu de l'état actuel des propositions faites dans la littérature. Afin de guider le choix d'un protocole ou d'un autre en fonction des contraintes de déploiement et de performances, nous proposons les Figures 2.4 et 2.5. Un protocole qui fonctionne bien en termes de délai de couverture et d'équité interroge les tags plus rapidement car plus de lecteurs sont activés avec succès et lisent les tags à portée. Le délai de couverture et l'équité donnent un avantage dans le cas de scénarios dynamiques avec lecteurs et/ou tags mobiles. Relativement, les performances en termes de débit de lecture, de collisions et d'efficacité donnent une idée des performances du protocole dans les environnements de déploiement de lecteurs denses. Dans de telles conditions, les lecteurs ne réussissent pas dans leurs procédures de contention et échouent à plusieurs reprises lorsqu'ils tentent d'accéder au support. Seul un sous-ensemble de lecteurs est actif et seuls les tags à portée sont interrogées. Les caractéristiques observées dans les Figures 2.4 et 2.5 sont :

— **densité** : pour savoir si le protocole peut fonctionner dans des scénarios de déploiements très denses où plusieurs lecteurs sont à proximité. Ces scénarios peuvent être trouvés dans les usines où les produits doivent être vérifiés le long d'une ligne de fabrication ;

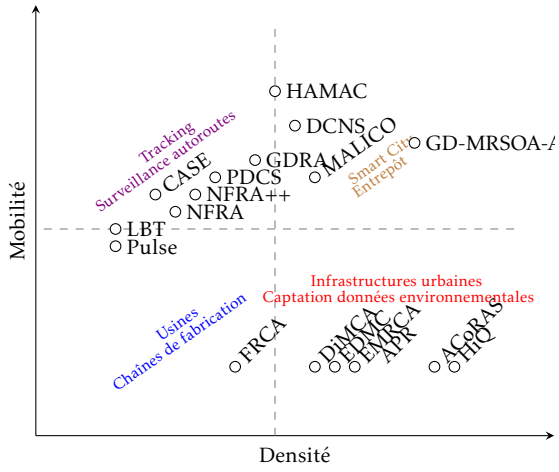
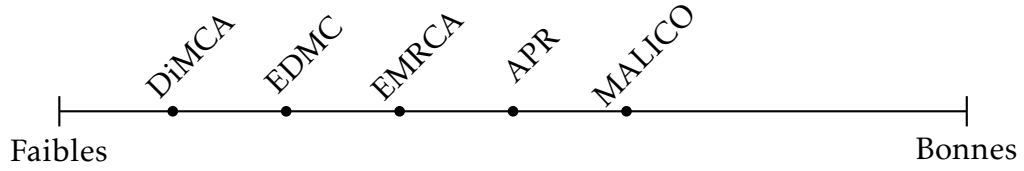


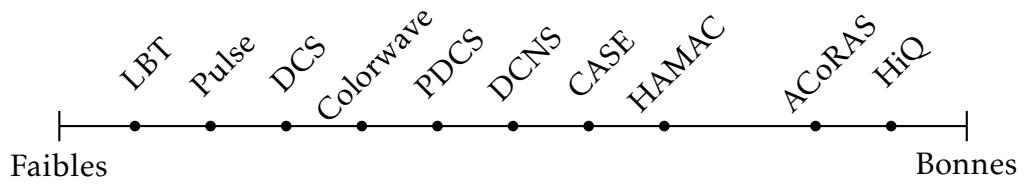
FIGURE 2.4 – Performance des protocoles d'anti-collision en fonction des contraintes d'applications

— **mobilité** : pour vérifier si le protocole est capable de gérer les scénarios mobiles. Cela pourrait être le cas dans un entrepôt avec des lecteurs montés sur des chariots élévateurs ou dans une ville intelligente où les lecteurs sont montés sur des véhicules de transport en commun pour capter des données environnementales ;
— **applications à faible latence** : afin de valider si le protocole doit être pris en compte lors de la construction d'une application sensible à la latence et que les informations des tags doivent être rapidement disponibles. Cela pourrait être le cas dans une application de détection où un problème critique peut se produire et toutes les données détectées sont cruciales ;
— **applications sensibles aux collisions** : ce sont les applications où le moins de collisions sont attendues afin d'avoir un système de travail optimal. Dans le cas d'un port où des conteneurs, identifiés grâce à des tags RFID, sont chargés ou déchargés des bateaux, les collisions et les tags manqués pourraient entraîner la perte de conteneurs ou de quantité importantes de biens.

L'analyse des Figure 2.4 et 2.5 permet d'avancer que pour surveiller plusieurs cartons dans un entrepôt avec des lecteurs RFID fixés aux murs, où les produits doivent être rapidement identifiés et traités pour éviter les pertes, une bonne



(a) Performances en terme de délai de couverture



(b) Performances en terme de collisions

FIGURE 2.5 – Performances en fonction des contraintes de collisions et latence pour des déploiements denses et mobiles de lecteurs

solution pourrait être d'utiliser DiMCA, EDMC ou APR qui peuvent fonctionner dans des conditions de déploiement denses de lecteurs avec sensibilité à la latence. Si jamais, certains lecteurs doivent être mobiles (montés sur des chariots élévateurs ou portatifs par des travailleurs) une meilleure solution serait d'utiliser GD-MRDOA-AIS ou MALICO qui montrent les meilleures performances dans les déploiements mobiles, même si ces dernières pourraient être améliorées pour un meilleur délai de couverture. Pour un port avec des lecteurs fixés au sol et des tags rattachés aux conteneurs, une solution comme ACoRAS pourrait offrir le meilleur compromis, mais si certains lecteurs sont mobiles, l'utilisation de solutions comme PDCS, DCNS ou HAMAC est plus fiable. Une application de surveillance RFID pour l'agriculture, où les lecteurs ne nécessitent pas un déploiement dense à proximité pour surveiller les niveaux d'humidité et de température, sans données critiques nécessitant une surveillance constante, Colorwave ou LBT pourraient suffire. S'il faut traiter avec des lecteurs rattachés aux agriculteurs ou leurs moissonneuse-batteuses mobiles, une solution comme Pulse pourrait être mise en œuvre.

2.4 Conclusion

Le problème des collisions en RFID demeure un souci important dans la mise en place de solutions basées sur la technologie. En effet, les erreurs de lectures peuvent se prouver critiques en fonction de l'application entraînant des dégâts matériels, financiers voire même humains. Il est donc nécessaire d'y remédier.

Les nombreuses solutions proposées dans l'état de l'art parviennent toutes à réduire les collisions tant bien que mal en utilisant des mécanismes différents. Toutefois, en fonction des contraintes de déploiements des dispositifs (lecteurs et/ou tags) et des exigences des applications, les performances des algorithmes varient grandement, pouvant guider vers le choix d'un algorithme ou d'un autre. Une solution utilisable quelles que soient les conditions aurait certes été une aubaine mais malheureusement, compte tenu de l'état de l'art actuel, cela est difficilement concevable.

Au vu de notre objectif d'offrir une solution permettant la collecte de données RFID pour la surveillance d'infrastructures urbaines, nous avons choisi de nous positionner dans l'optique d'offrir des solutions performantes pour les déploiements denses et mobiles. Nos solutions devront être en mesure d'offrir un débit de lecture important avec une équité convenable permettant d'assurer que l'ensemble des infrastructures concernées sont surveillées. Concernant le délai de couverture, il devra être maintenu faible pour des raisons de sécurité évidentes permettant d'être informé au plus vite des défaillances structurelles des infrastructures urbaines. Ces remarques en combinaison avec les observations faites concernant les différents algorithmes nous ont poussé à chercher des protocoles distribués TDMA.

Publication

Journal

Abdoul Aziz Mbacké, Nathalie Mitton et Herve Rivano. "A survey of RFID readers anticollision protocols" dans *IEEE Journal of Radio Frequency Identification, IEEE, 2018*

Chapitre 3

Anticollision de lecteurs RFID - Nos contributions

L'utilisation de solutions RFID pour la surveillance d'infrastructures urbaines nécessite de disposer de mécanismes d'anti-collision adéquates et en concordance avec les contraintes de déploiement dans ces environnements. En effet, afin de couvrir une ville et ses infrastructures (ponts, routes, tunnels, bâtiments, etc.), il est nécessaire de pouvoir, non seulement, déployer une importante quantité de tags RFID, pour identifier et récupérer le maximum de données environnementales, mais également de faire appel à un grand nombre de lecteurs disposés partout à travers la ville et ses équipements. A cette contrainte de densité s'ajoute celle de mobilité vu que le déploiement de lecteurs fixes seuls pour couvrir l'ensemble des tags est non seulement très coûteux mais également fastidieux et inefficace. Il est donc plus judicieux d'utiliser des lecteurs, par exemple, montés sur les véhicules de transport en commun ou des vélos en libre service, afin d'assurer une couverture maximale tout en conservant un nombre acceptable de lecteurs. Les mécanismes d'anti-collisions devront également être en mesure d'assurer une couverture optimale de l'ensemble des tags avec une latence faible.

Partant de ce constat et de l'absence de solutions permettant d'offrir les performances nécessaires avec les exigences requises, nous proposons dans ce chapitre deux solutions répondant favorablement à l'ensemble des contraintes

sus-citées.

3.1 Distributed Efficient & Fair Anti-collision for RFID (DEFAR)

DEFAR est un algorithme multicanal, distribué et TDMA d'anti-collision de lecteurs RFID. Il se base sur l'utilisation de deux interfaces de communications par les lecteurs. Une première interface d'interrogation des tags et une seconde pour la communication entre voisins. Un critère récurrent des solutions d'anti-collision de lecteurs RFID est l'inactivité systématique de l'ensemble des lecteurs en collision. En effet, comme nous avons pu le voir dans les algorithmes en Section 2.3, si lors de la phase de contention, une collision potentielle est détectée. L'ensemble des lecteurs concernés sont inactifs et manquent donc l'opportunité de lire les tags à portée. Un autre défaut remarqué parmi les solutions de l'état de l'art était le manque d'équité. En effet, les tags étant déployés à plusieurs endroits, donner l'accès régulièrement au même sous-ensemble de lecteurs constitue un risque important d'information ratée pour des applications de surveillance urbaines. Il est donc nécessaire d'avoir un accès équitable au canal pour les différents lecteurs.

Afin de remédier à cela, DEFAR propose la récupération d'au moins un des lecteurs en contention pour permettre la lecture des tags à sa portée. Ce mécanisme permet parallèlement d'augmenter le débit de lecture et de diminuer le délai de couverture du système RFID. Par ailleurs, pour corriger l'équité, DEFAR met en place un système de priorisation des lecteurs lors de la contention. Cette hiérarchisation est accordée en fonction des succès/échecs de contentions sur les tentatives d'accès au canal précédentes.

3.1.1 Positionnement

DEFAR a été conçu en vu d'améliorer les performances des solutions RFID dans les déploiements denses et mobiles et s'appuie sur l'observation des aspects négatifs des algorithmes de l'état de l'art. Ainsi avons nous choisi de baser

DEFAR sur une approche TDMA. L'utilisation d'une telle solution permet de s'affranchir des phases d'écoute de canal plus ou moins longues et donc de définir un "agenda" d'activité pour les lecteurs. Nous avons également choisi d'opter pour une solution distribuée avec les lecteurs qui déroulent localement le même algorithme. Se passer d'un serveur central, peut certes augmenter le nombre de collisions mais permet d'avoir des décisions locales des lecteurs en fonction de leur environnement direct et non pas d'un calcul global. Les décisions locales ne subissent pas la latence induite par la collecte d'informations globales d'un serveur, leur calcul et redistribution aux lecteurs. L'ensemble de ces choix a donc été effectué dans l'optique de maximiser les performances en termes de débit de lecture, efficacité, équité et réduction de latence afin d'obtenir un protocole adapté aux contraintes des environnements et déploiements étudiés.

Afin de mettre en œuvre un protocole TDMA distribué avec entente locale des lecteurs, il a été nécessaire d'utiliser deux interfaces de communication au niveau des lecteurs. Ces deux interfaces de communications, comme retrouvées dans les protocoles présentés en Sections 2.3.2 et 2.3.3, permettent pour l'une de lire les tags à portée et pour l'autre d'organiser et de coordonner les activités de lecture entre voisins. Deux portées de communication sont donc identifiées :

— la portée de lecture notée d_{CRT}. Elle désigne la distance maximale à laquelle les tags peuvent être lus et identifiés depuis un lecteur ;

— la portée de communication notée d_{Com}. C'est la distance maximale à laquelle les lecteurs sont en mesure d'échanger des messages pour leur coordination. Afin de réduire les collisions, cette distance a été fixée à $d_{Com} = 2 \times 3.3 \times d_{CRT}$ en adéquation avec l'analyse faite dans [3]. En effet, l'utilisation du multi-canal génère des interférences entre canaux adjacents. Ainsi, deux lecteurs sur deux canaux différents doivent au minimum être séparés par une distance d'au moins d_{Com} afin de pouvoir interroger les tags à portée d_{CRT} sans interférences. En fixant cette valeur de d_{Com} on s'assure donc que si un lecteur, lors de la contention, ne reçoit aucun message sur le canal de communication dédié, il n'est en collision avec aucun autre dans son voisinage.

L'utilisation d'un mécanisme TDMA permet d'organiser les activités de lecteurs sur le temps. A cet effet, l'accès au canal a été organisé en "*rounds*". Un

round définit une durée durant laquelle l'ensemble des lecteurs déployés entrent en contention pour l'accès au canal une et une seule fois. Ces *rounds* sont ensuite eux-mêmes subdivisés en max_slots timeslots de durée T_{Slot}. A chaque timeslot, un sous-ensemble du nombre total de lecteurs entre en contention pour l'accès au canal. Chaque timeslot est structuré en 2 phases comme suit (voir Figure 3.1) :

— Premièrement, une phase d'échange de messages entre lecteurs voisins sur leur interface de portée d_{Com}. Chaque lecteur découvre son voisinage et s'assure de pouvoir accéder au canal. Le voisinage désigne l'ensemble des lecteurs à portée de communication sur le canal dédié et donc susceptible d'être en collision de lecture sur le canal d'interrogation. Avant de diffuser un message durant cette phase, chaque lecteur observe d'abord un temps d'attente tiré aléatoirement afin de réduire les collisions de diffusion. La conception de ces temps d'attente a été faite de sorte à ce que chaque lecteur puisse diffuser son message à ses voisins avec un nombre négligeable de collisions. Cette phase de diffusion de messages est subdivisée en différents slots d'attente notés B_1, B_2, ..., B_n (voir Figure 3.1). Cette phase sera notée T_{Beacon} ;

— Deuxièmement, une phase d'interrogation des tags pendant une durée elle aussi déterminée notée T_{CRT}. Au vu de la phase d'échange de messages précédente, les lecteurs en collision sont désactivés et ne participent pas à la phase de lecture des tags. Aucune collision n'est donc possible durant la lecture des tags. En effet, seul au plus un lecteur par slot et par fréquence pourra être actif dans un voisinage donné.

Les lecteurs sont en mesure de suivre l'évolution des *rounds* et timeslots grâce à une approche TDMA basée sur l'utilisation d'horloges internes aux lecteurs. La durée d'un *round* sera donc définit comme :

$$T_{round} = max_slots \times T_{Slot} = max_slots \times (T_{Beacon} + T_{CRT})$$

Afin d'éviter la désactivation de l'ensemble des lecteurs lors d'une contention, nous avons introduit un système de comparaison d'ID permettant de conserver au moins un lecteur qui sera alors capable de lire les tags à portée. Par ailleurs, pour éviter que les mêmes lecteurs aient constamment accès au canal, nous défi-

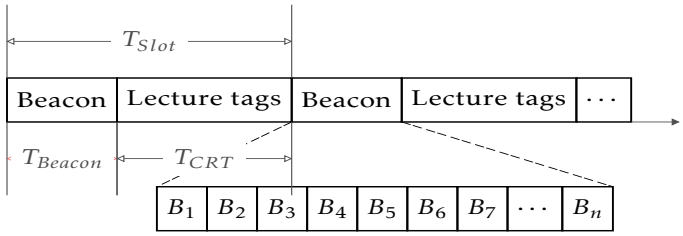


FIGURE 3.1 – Structure d'un *round* - DEFAR

nissons différents niveaux de priorité pour augmenter les chances des lecteurs en échec face aux lecteurs ayant réussi sur les *rounds* précédents. Nous mettons en place 3 niveaux de priorité définis comme suit (voir Figure 3.2) :

— **NEUTRAL** est le niveau de priorité de l'ensemble des lecteurs au départ ;
— **LAZY** est le niveau de priorité inférieur affecté aux lecteurs ayant réussi leur contention sur le *round* précédent ;
— **PUMPED UP** est le niveau de priorité supérieur accordé aux lecteurs en échec sur le *round* précédent et leur permettant d'augmenter leurs chances d'accès au canal sur le *round* en cours.

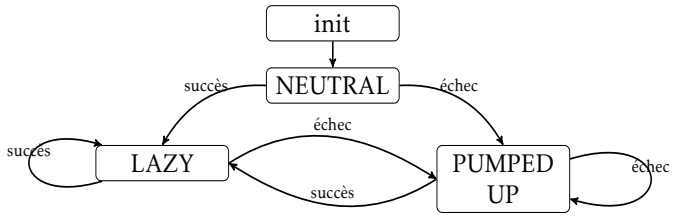


FIGURE 3.2 – Diagramme d'états

3.1.2 Fonctionnement

A l'initialisation, les lecteurs sont dans le premier *round*, avec le niveau de priorité NEUTRAL, et choisissent aléatoirement un slot sur la plage $[0; max_slots[$ et un canal parmi ceux accordés par ETSI [1]. Chaque lecteur R_i, connaissant la durée d'un slot, attend donc son slot choisi $slot_i$ pendant une durée $t = slot_i \times T_{Slot}$. Une fois au slot correspondant, R_i observe un temps aléatoire avant de diffuser une message à ses voisins contenant son ID et le canal d'interrogation f_i choisi.

Ce message est diffusé sur l'interface de communication dédiée et sur la portée d_{Com}. Deux scénarios peuvent alors se produire (voir Algorithme 1) :

1. **Scénario 1** : R_i ne reçoit aucun message durant T_{Beacon}. R_i considère alors qu'il est le seul lecteur sur ce slot dans son voisinage. Il peut alors accéder au canal f_i choisi et interroger les tags pendant T_{CRT} avant de passer en priorité LAZY (Algo 1 lignes 6-7) ;
2. **Scénario 2** : R_i reçoit au moins un message d'au moins un de ses voisins ayant choisi le même canal d'interrogation sur la durée T_{Beacon}. R_i compare alors les IDs contenus dans les messages reçus avec le sien (Algo 1 ligne 8) :
   — R_i a l'ID le plus petit parmi ceux en contention. R_i remporte alors la contention et lit les tags à portée pendant T_{CRT} avant de passer en état LAZY pour le *round* suivant (Algo 1 lignes 9-10) ;
   — R_i n'a pas le plus petit ID, un ou plusieurs ID parmi ceux reçus est plus faible que celui de R_i. Ce dernier perd alors la contention et passe en priorité PUMPED UP pour le tour suivant. R_i enregistre alors une collision (Algo 1 ligne 11).

Ce processus de contention est maintenu pendant le premier *round* où l'ensemble des lecteurs démarrent en priorité NEUTRAL.

Algorithm 1 DEFAR - Premier *round* pour un lecteur R_i

```
1:  slot_i ← (int)random[0; max_slots]
2:  f_i ← (int)random[1; f_max]
3:  prio_i ← NEUTRAL
4:  if slot en cours == slot_i then
5:      Diffusion message
6:      if Pas de message reçu then                          ▷ Pas de collision
7:          Lecture tags ; prio_i ← LAZY
8:      else                          ▷ Au moins un lecteur sur le même canal f_i
9:          if R_i a l'ID le plus petit then          ▷ R_i remporte la contention
10:             Lecture tags ; prio_i ← LAZY
11:         else prio_i ← PUMPEDUP
12:         end if
13:     end if
14: end if
```

Sur les *rounds* de contention suivants, on conserve la même technique de résolution de contention avec quelques modifications pour prendre en compte les nouvelles priorités introduites. En effet, suite au premier *round*, tous les lecteurs sont soit en LAZY, si ils ont réussi leur contention, soit en PUMPED UP si ils ont échoué. Pour le nouveau tour de contention, les lecteurs choisissent à nouveau aléatoirement un $slot_i$ et un canal f_i. Ils attendent le slot correspondant à leur choix pendant le temps $t = slot_i \times T_{Slot}$, observent le temps d'attente déterminé avant de diffuser leurs messages aux voisins sur le canal de communication dédié. Cependant, cette fois-ci le message diffusé contient en plus de l'ID et du canal choisi, l'état $prio_i$ du lecteur. Ainsi en fonction du niveau de priorité d'un lecteur R_i, on peut avoir les cas suivants :

1. Scénario 1 : R_i est priorité LAZY et a donc réussi sa contention sur le *round* précédent (Algo. 2 ligne 8) :
   - pas de collisions enregistrées lors de l'échange de messages : il n'y donc aucun lecteur concurrent dans le voisinage de R_i qui peut de nouveau accéder au canal et lire les tags à sa portée pendant T_{CRT} (Algo 2 lignes 5-6) ;
   - au moins un voisin diffuse un message avec le même canal f_i : dans ce cas, la résolution de la contention se fera en fonction du niveau de priorité des lecteurs :
     - les autres lecteurs en contention sont également en priorité LAZY : l'ensemble des lecteurs engagés ont déjà eu accès au canal sur le *round* précédent. Dans ce cas, la résolution se fera par rapport à l'ID comme lors du premier *round*. Si R_i a l'ID le plus faible, il accède au canal et interroge les tags, sinon, il passe en PUMPED UP pour le *round* suivant (Algo 2 lignes 11-16) ;
     - au moins un des lecteurs en contention est en priorité PUMPED UP : R_i perd la contention et passe en PUMPED UP pour le tour suivant (Algo 2 lignes 9-10).
2. Scénario 2 : R_i est en priorité PUMPED UP et a échoué lors de la contention sur le *round* précédent (Algo 2 ligne 17) :
   - pas de collisions enregistrées lors de l'échange de messages : il n'y donc aucun lecteur concurrent dans le voisinage de R_i qui peut enfin

accéder au canal et lire les tags à sa portée pendant T_{CRT}. R_i passera en priorité LAZY sur le *round* suivant (Algo 2 lignes 5-6) ;

— au moins un voisin diffuse un message avec le même canal f_i : dans ce cas, la résolution de la contention se fera en fonction du niveau de priorité des lecteurs :
  — au moins un des lecteurs en contention est en priorité PUMPED UP : les lecteurs sont dans le niveau de priorité supérieur car ayant échoué sur le *round* précédent. Dans ce cas, la résolution se fera par rapport à l'ID comme lors du premier *round*. Si R_i a l'ID le plus faible, il accède au canal et interroge les tags avant de passer en LAZY, sinon, il reste en PUMPED UP pour le *round* suivant (Algo 2 lignes 20-25) ;
  — les autres lecteurs en contention sont en priorité LAZY : R_i gagne la contention et lit les tags à sa portée pendant T_{CRT} avant de passer en priorité LAZY pour le tour suivant (Algo 2 lignes 18-19).

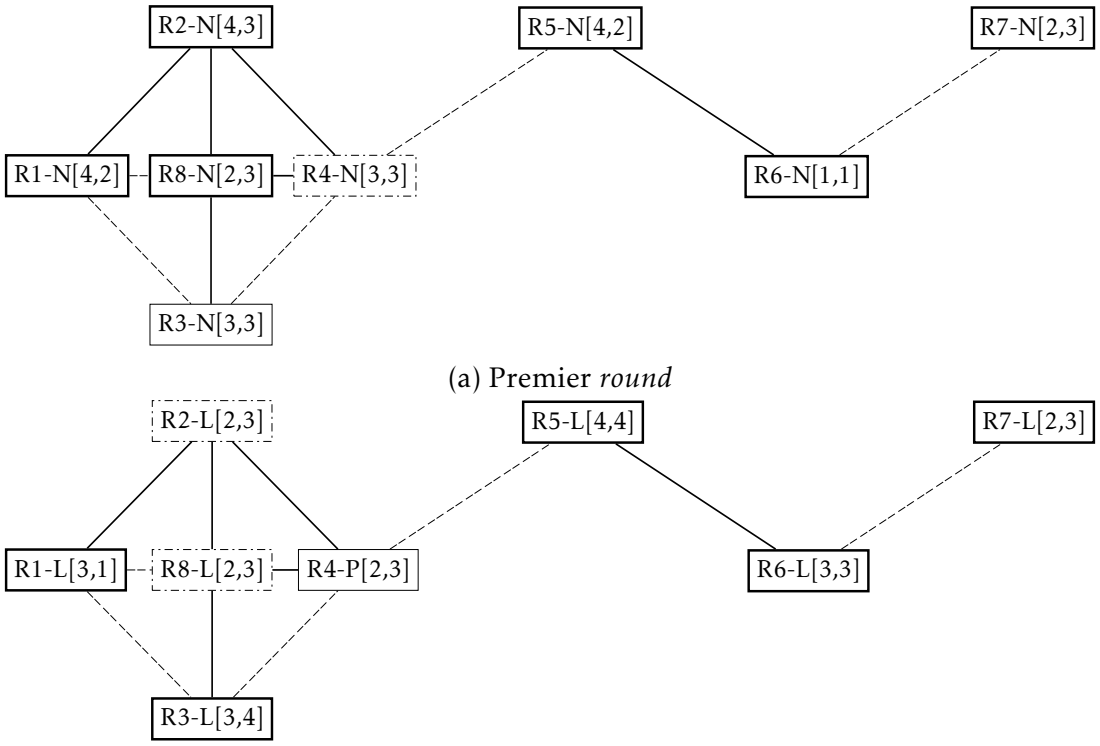
Dans le cas de l'arrivée d'un nouveau lecteur dans le système, il prend la priorité LAZY et grâce à son interface de communication dédiée écoute le canal en attente du prochain *round*. Il choisit alors un timeslot et un canal aléatoirement comme les autres lecteurs et attend le slot correspondant avant de diffuser son message. L'ordre de priorité définit est : PUMPED UP puis NEUTRAL et enfin LAZY. Le nouveau lecteur prendra donc la main face à des lecteurs LAZY en cas de collision avant de passer lui-même en LAZY pour le tour suivant (Algo 2 lignes 27-28). En cas de contention face à un lecteur PUMPED UP, il perdra la contention et passera lui-même en PUMPED UP (Algo 2 lignes 29-30). Sinon, face à un autre nouveau lecteur, en priorité NEUTRAL également, la contention sera résolue comme lors de l'initialisation (Algo 2 lignes 31-34).

Afin de faciliter la compréhension de notre algorithme, nous proposons le schéma de la Figure 3.3 avec 8 lecteurs déployés. Le système RFID est représenté sous la forme d'un graphe où les lecteurs sont les sommets et les distances entre lecteurs sont les arêtes. Si la distance entre deux lecteurs est inférieure ou égale à d_{CRT}, la ligne est entière, sinon si la distance est supérieure à d_{CRT} mais inférieure à la distance d'interférences sur canaux adjacents, le trait est tireté. Sur la Figure 3.3a, les "N" représentent les états NEUTRAL des lecteurs

Algorithm 2 DEFAR - *rounds* suivants pour un lecteur R_i

```
1:  slot_i ← (int)random[0; max_slots]
2:  f_i ← (int)random[1; f_max]
3:  if slot en cours == slot_i then
4:      Diffusion message
5:      if Pas de message reçu then                          ▷ Pas de collision
6:          Lecture tags; prio_i ← LAZY
7:      else                          ▷ Au moins un lecteur R_j avec le même canal
8:          if prio_i == LAZY then
9:              if prio_j == PUMPEDUP or prio_j == NEUTRAL then
10:                 prio_i ← PUMPEDUP                          ▷ R_i perd la contention
11:             else
12:                 if R_i a l'ID le plus petit then           ▷ R_i gagne la contention
13:                     Lecture tags; prio_i ← LAZY
14:                 else prio_i ← PUMPEDUP
15:                 end if
16:             end if
17:         else if prio_i == PUMPEDUP then
18:             if prio_j == LAZY or prio_j == NEUTRAL then
19:                 Lecture tags; prio_i ← LAZY
20:             else
21:                 if R_i a l'ID le plus petit then           ▷ R_i gagne la contention
22:                     Lecture tags; prio_i ← LAZY
23:                 else prio_i ← PUMPEDUP
24:                 end if
25:             end if
26:         elseprio_i == NEUTRAL
27:             if prio_j == LAZY then
28:                 Lecture tags; prio_i ← LAZY
29:             else if prio_j == PUMPEDUP then
30:                 prio_i ← PUMPEDUP
31:             else                                           ▷ prio_j == NEUTRAL
32:                 if R_i a l'ID le plus petit then           ▷ R_i gagne la contention
33:                     Lecture tags; prio_i ← LAZY
34:                 else prio_i ← PUMPEDUP
35:                 end if
36:             end if
37:         end if
38:     end if
39: end if
```

au départ et les valeurs entre crochets représentent respectivement le slot et le canal choisis $[slot_i; f_i]$. Lors du premier *round*, seuls les lecteurs R3 et R4 entrent en collision car ayant choisi le même canal et slot. R3 remporte la contention car ayant l'ID le plus petit et va donc lire les tags avant de passer en LAZY, représenté par un "L", tandis que, R4 échoue et passe en priorité PUMPED UP représentée par un "P". Tous les autres lecteurs passent la phase de contention car ayant choisi des slots et canaux différents. Ils vont donc lire les tags à portée et passer en priorité "L". Lors du *round* suivant (voir Figure 3.3b), les lecteurs R2, R4 et R8 choisissent les mêmes slots et canaux. Vu que R4 a échoué lors de la contention précédente, il est en priorité PUMPED UP alors que les autres sont en LAZY. R4 remporte la contention et accède au canal pour lire les tags avant de passer en LAZY. R2 et R8 se mettent en PUMPED UP pour le *round* suivant afin d'être avantagés.



(a) Premier *round*

(b) Deuxième *round*

Figure 3.3 – Illustration DEFAR

3.1.3 mobile-DEFAR (mDEFAR)

mDEFAR est une variation de DEFAR plus appropriée pour les déploiements encore plus denses et mobiles de lecteurs RFID. Tandis que dans DEFAR, la portée de communication des lecteurs devait au minimum être de $d_{Com} = 2 \times 3.3 \times d_{CRT}$ afin d'éviter les collisions sur canaux adjacents, avec mDEFAR nous proposons d'utiliser une solution monocanal. Les interférences sur canaux adjacents éliminées, il est donc possible de limiter la portée de communication d_{COM} à une valeur plus petite et donc réduire le voisinage de chaque lecteur et parallèlement le nombre de lecteurs désactivés lors de la contention au détriment de la couverture. La portée de l'interface dédiée à la communication entre lecteurs est donc fixée à $d_{Com} = 2 \times d_{CRT}$. En dehors de cette modification concernant la portée de communication, le reste de l'algorithme et la procédure de contention restent intacts et se déroulent donc comme dans DEFAR.

Cette nouvelle approche permet d'améliorer encore plus le débit de lecture et le délai de couverture. Cependant, l'utilisation d'une fréquence unique réduit grandement le nombre de choix disponibles pour les lecteurs qui ne sont alors différenciés que par leurs slots. Le nombre de collisions est donc plus important mais reste dans les limites d'un compromis favorable comme on pourra le voir dans les résultats.

3.2 Coverage Oriented RFID Anti-collision (CORA)

CORA est un protocole distribué, TDMA et monocanal d'anti-collision de lecteurs RFID. Tout comme DEFAR, il nécessite l'utilisation de deux interfaces de communication par lecteur afin d'interroger les tags à portée et de coordonner les activités. Dans la conception de CORA, nous sommes partis d'un constat supplémentaire concernant les différentes propositions d'anti-collision en RFID. La plupart des solution proposées envisagent la désactivation de l'ensemble des lecteurs en collisions sans égard au nombre de tags couverts par chacun d'entre eux. Avec CORA, nous proposons de prendre la densité de déploiement et la mobilité des lecteurs à notre avantage afin d'augmenter le débit de lecture et réduire le délai de couverture totale quitte à augmenter, dans des proportions

raisonnables, le nombre de collisions. On repose sur le fait que les collisions ne sont pas totales vu que certains tags peuvent quand même être lus car hors zone de collision. CORA tire donc profit des contraintes environnementales de déploiement pour améliorer ses performances.

3.2.1 Positionnement

Toujours dans l'optique d'améliorer les performances des solutions RFID dans les déploiements denses et mobiles de lecteurs/tags pour la surveillance d'infrastructures urbaines, nous proposons CORA. Notre algorithme est également basé sur un mode opératoire TDMA et une activité cadencée des lecteurs. Le choix d'une solution distribuée permet également de s'affranchir de l'utilisation d'un serveur central qui aurait certes pu améliorer les performances en termes de débit de lecture et collisions mais au prix d'une latence plus importante.

La conception de notre choix d'algorithme distribué s'est fait grâce à l'utilisation de deux interfaces de communication pour les lecteurs comme DEFAR, cependant avec quelques différences. Les deux interfaces considérées sont les suivantes :

— une interface d'interrogation des tags sur une portée d_{CRT} désignant la distance maximale de lecture d'un tag ;

— une interface de communication d_{Com} utilisée pour la coordination locale des lecteurs. Contrairement à DEFAR et similairement à mDEFAR, la portée de l'interface de communication a été fixée à $d_{Com} = 2 \times d_{CRT}$. En optant pour un fonctionnement monocanal de notre algorithme, nous faisons abstraction des collisions induites par les canaux adjacents. Grâce à cette valeur de d_{Com}, les lecteurs peuvent donc s'assurer qu'il ne sont en collision avec aucun voisin sur leur portée de lecture.

L'organisation de l'activité des lecteurs a, en fonction de notre schéma TDMA, été organisée en "*rounds*". Tout comme pour DEFAR, un *round* désigne une durée durant laquelle l'ensemble des lecteurs entre en contention pour l'accès au canal d'interrogation une et une seule fois. Ces mêmes *rounds* sont ensuite organisés en deux phases de tailles max_slots (voir Figure 3.4) :

1. une phase d'échange de messages entre lecteurs voisins sur l'interface de

communication de portée d_{Com}. Cette phase est subdivisée en max_slots timeslots d'échange de message de durée T_{Beacon}. Ainsi chaque lecteur en fonction d'un choix aléatoire définira son timeslot de diffusion ;

2. une phase d'interrogation des tags sur l'interface de communication de portée d_{CRT}. Cette phase est également subdivisée en max_slots timeslots de lecture de durée T_{CRT}. A chaque timeslot de diffusion de message correspondra un timeslot d'interrogation de tags.

La durée d'un *round* dans CORA sera donc définie comme suit :

$$T_{round} = max_slots \times T_{Beacon} + max_slots \times T_{CRT} = max_slots \times (T_{Beacon} + T_{CRT})$$

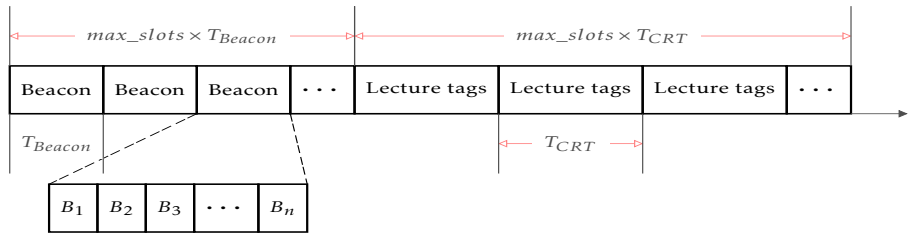
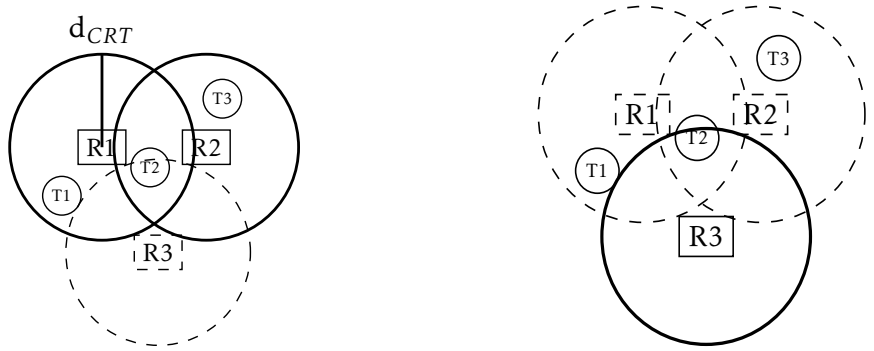


FIGURE 3.4 – Structure d'un *round* - CORA

Dans le but de profiter de la densité de déploiement des lecteurs et de leur mobilité, le processus de contention tiendra compte pour chaque lecteur donné, de son voisinage et de leur processus de contention. Ainsi un lecteur, en connaissance des choix de contention de ses voisins pourra accéder au canal au risque de provoquer des collisions. La Figure 3.5 montre un exemple avec 3 lecteurs R1, R2 et R3 en couverture sur les tags T1, T2 et T3. On y voit que les 3 lecteurs sont en collision sur le tag T2. Par défaut, les algorithmes d'anti-collision auraient désactivé l'ensemble des lecteurs en contention et ainsi aucun des trois tags ne serait lus quand bien même T1 et T3 ne sont pas en zones de superposition. Avec CORA, les lecteurs R1 et R2 prennent connaissance de l'existence de R3 et lisent simultanément et respectivement les tags T1 et T3. Le tag T2 couvert pas les deux lecteurs ne sera pas reconnu. Plus tard, lors de l'accès au canal de R3, le tag T2 pourra enfin être lu. Ainsi, au lieu d'avoir les 3 lecteurs activés à 3 instants différents, il est possible d'obtenir une couverture

complète des tags en juste 2 temps de lecture. Ces performances améliorées en débit de lecture et délai de couverture se font cependant comme on peut le voir au coût d'une collision enregistrée.



(a) Collision sur un tag (b) Lecture de R3

FIGURE 3.5 – Résolution collisions CORA

3.2.2 Fonctionnement

L'algorithme 3 représente le processus de contention de CORA. Au début de chaque *round*, les lecteurs choisissent aléatoirement un numéro de slot $slot_i$ dans l'intervalle $[0; max_slots[$ (Algo. 3 ligne 1). A chacun des slots disponibles correspondent un timeslot d'échange de messages entre lecteurs de durée T_{Beacon} et un timeslot d'interrogation de tags de durée T_{CRT}. Les lecteurs attendent donc une durée $t = slot_i \times T_{Beacon}$ avant de diffuser sur leur interface de communication un message précisant leurs slots choisis $slot_i$. A la réception d'un message (Algo. 3 ligne 7), chaque lecteur compare son slot choisi $slot_i$ avec celui reçu $slot_j$ et met à jour deux paramètres α et β :

— si $slot_i == slot_j$ (Algo. 3 lignes 9-10), le lecteur incrémente le paramètre α qui désigne le nombre de lecteurs concurrents car désirant lire sur le même slot ;

— si $slot_i \neq slot_j$ (Algo. 3 lignes 11-12), le lecteur incrémente le paramètre β qui désigne le nombre de lecteurs complémentaires car pouvant couvrir les tags en cas de collision.

Après une durée $T = max_slots \times T_{Beacon}$, l'ensemble des lecteurs du système ont diffusé leurs message dans leurs voisinages respectifs. Chaque lecteur procède à une opération de vérification consistant en l'opération suivante $M = \alpha - \beta$ (Algo. 3 ligne 15). En fonction du résultat de l'opération, deux situations peuvent se produire :

1. $M > 0$ (Algo. 3 lignes 16-17), le lecteur a trop de voisins concurrents sur le même slot et se désactive pour le *round* en cours. Le nombre potentiel de collisions engendrées par la lecture serait trop important et rendrait la lecture inefficace et inopportune ;

2. $M \leq 0$ (Algo. 3 lignes 18-20), le lecteur considère avoir assez de voisins complémentaires pour couvrir les tags en superposition. Le lecteur attend donc son slot correspondant pendant $t' = slot_i \times T_{CRT}$ avant de lire les tags à portée et se désactive en attendant le *round* suivant. Les tags potentiellement en zone de collision pourront être couverts par les autres lecteurs complémentaires durant le *round* en cours.

A chaque *round*, les paramètres α et β sont réinitialisés (Algo. 3 lignes 24-25), le lecteur choisi aléatoirement un nouveau $slot_i$ et le processus est répété.

Algorithm 3 CORA - Déroulement d'un *round*

```
1:  slot_i ← (int)random[0; max_slots[
2:  Diffusion message
3:  if Pas de message reçu then                          ▹ Pas de collisions
4:      if slot en cours == slot_i then
5:          Lecture tags
6:      end if
7:  else                                  ▹ R_i reçoit des message de ses voisins
8:      while R_i reçoit les messages do
9:          if slot_j == slot_i then              ▹ Voisin R_j sur même slot que R_i
10:             α++
11:         else                              ▹ Voisin R_j sur slot différent de R_i
12:             β++
13:         end if
14:     end while
15:     M = α − β
16:     if M > 0 then                                ▹ Trop de voisins concurrents
17:         Attend le round suivant
18:     else                                  ▹ Assez de voisins complémentaires
19:         if slot en cours == slot_i then
20:             Lecture tags
21:         end if
22:     end if
23: end if
24: α ← 0
25: β ← 0
```

La figure 3.6 illustre le déroulement de CORA au cours d'un *round* avec 8 lecteurs A ;B ;C ;D ;E ;F ;G ;H. Le système RFID est représenté sous la forme d'un graphe où les lecteurs sont les sommets et les arêtes représentent la distance entre les lecteurs. Deux lecteurs sont reliés si ils peuvent être en communication, ce qui signifie que la distance qui les sépare est inférieure ou égale à d_{Com}.

La valeur entre crochets correspond pour chaque lecteur au slot aléatoirement choisi. Après diffusion des messages et calcul des nombres de voisins concurrents et complémentaires, on obtient les valeurs de M suivantes : $M_A = -3$;$M_B = -1$;$M_C = -1$;$M_D = -3$;$M_E = 2$;$M_F = 0$;$M_G = -1$;$M_H = 0$. Ainsi, seul le lecteur E sera désactivé pour le *round* en cours car souffrant de collisions avec ses voisins D et F rendant son activation inefficace. Tous les autres lecteurs, grâce à leurs voisins complémentaires, pourront accéder au canal durant leurs slots et voir les tags en zone de collision être lus par les autres lecteurs sur des slots différents.

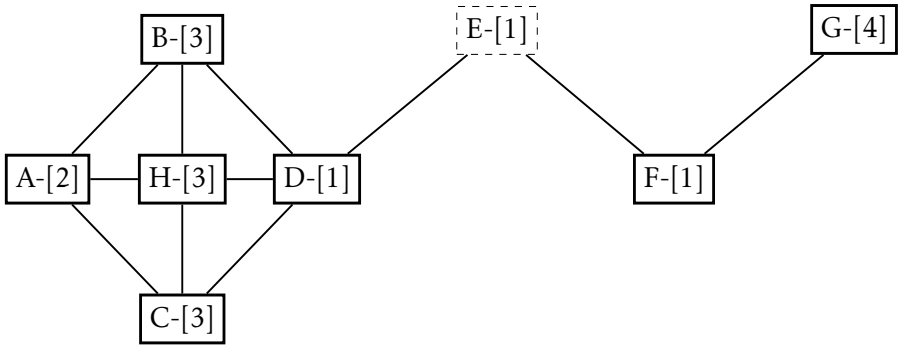


Figure 3.6 – Illustration CORA

3.3 Résultats obtenus

L'évaluation de performances des algorithmes proposés a été effectuée par simulation. L'ensemble des protocoles ont été implémentés sur WSNet [22] qui est un simulateur événementiel pour les réseaux de capteurs sans fil à grande échelle. WSNet est un simulateur modulaire qui permet de spécifier différents paramètres de simulation et paramètres d'équipements comme la mobilité, la source d'énergie, la couche application, les protocoles de routage, les protocoles mac, les applications, l'interface radioélectrique, les antennes, etc. Le choix de ce simulateur nous a permis d'évaluer nos propositions sur des déploiements denses et mobiles avec un très grand nombre de lecteurs et tags déployés.

Nous avons considéré deux configurations principales pour l'évaluation des protocoles :

— déploiements statiques : lors de ces simulations, les lecteurs et tags sont tous immobiles. Les résultats obtenus nous permettent de valider les performances sous la contrainte de densité de déploiement ;
— déploiement mobiles : lors de ces simulations, nous avons une combinaison de lecteurs et tags aussi bien statiques que mobiles se déplaçant au sein de la zone de déploiement. Les déplacements se font à différentes vitesses et configurations. Avoir des scénarios plus dynamiques qui restent denses en termes de déploiements de lecteurs et de tags permet d'observer les performances de nos algorithmes dans les applications prévues pour la collecte d'informations dans les villes intelligentes.

Pour chaque configuration, nous avons effectué 100 simulations. Les résultats sont présentés avec un intervalle de confiance de 95%. Concernant les paramètres des algorithmes, à savoir max_slots, T_{Beacon} et T_{CRT}, ils ont été fixés soit expérimentalement soit en accord avec le standard et les propositions de l'état de l'art. En effet, après plusieurs simulations avec des valeurs de $\{4;6;8;10;16;32\}$, nous avons décidé de fixer la valeur de $max_slots = 4$ qui offrait le meilleur compromis. L'utilisation de valeurs plus importantes permet de réduire le nombre de collisions vu la diversité de choix aléatoire des lecteurs mais se traduit par une baisse du débit de lecture et une augmentation de la latence car de nombreux slots sont soit vides, soit très peu occupés. Pour les valeurs de T_{Beacon} et T_{CRT}, elles ont été tirées du standard ETSI [1] et reprises par plusieurs propositions de l'état de l'art [24, 13, 3, 72, 27] et donc fixées à $T_{Beacon} = 5ms$ et $T_{CRT} = 460ms$.

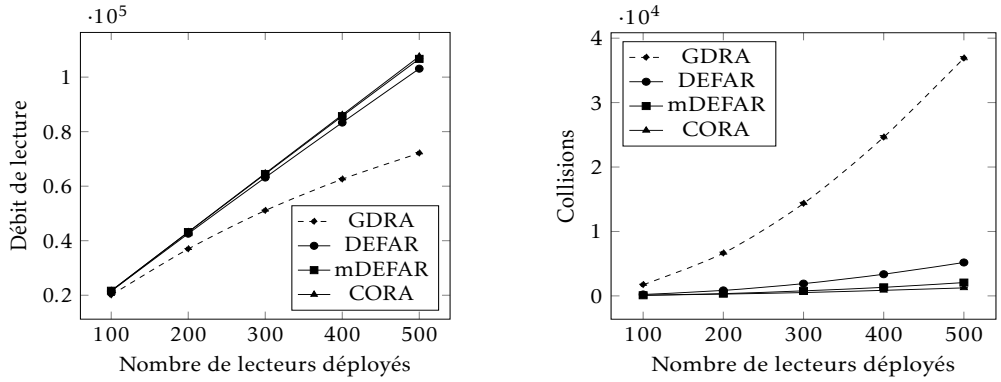
Les algorithmes ont été comparés à GDRA [13]. Ce choix s'est fait car lors de nos recherches, il s'est avéré que GDRA était le protocole le plus performant pour l'anti-collision en RFID. Aussi, GDRA est un algorithme multicanal, TDMA centralisé, ce qui nous permet de vérifier que le pari de miser sur une solution distribuée et monocanal (pour mDEFAR et CORA) offre un bon compromis en termes de performances. Nous avons mesuré pour chaque protocole (GDRA, DEFAR, mDEFAR et CORA) les performances suivant les métriques présentées en Section 2.2.

3.3.1 Déploiements statiques

On considère une surface de $1000 \times 1000m^2$ avec 2000 tags déployés aléatoirement. Un nombre croissant de lecteurs allant de 100 à 500 lecteurs par pas de 100 est également déployé aléatoirement sur toute la surface. Ces paramètres correspondent à une densité moyenne de 3 lecteurs concurrents et 4 tags couverts par lecteur.

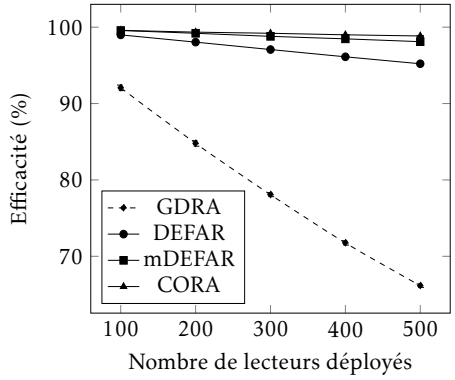
— **Débit de lecture** : Sur la Figure 3.7a, le débit de nos trois propositions est comparé à GDRA pour les différentes densités de réseau testées. Toutes les valeurs de débit augmentent avec le nombre de lecteurs déployés dans le système. Ceci est attendu car avec plus de lecteurs, nous avons plus d'Accès Canal Réussis (ACR) potentiels. Nous remarquons que toutes les valeurs de débit sont similaires au début avec une centaine de lecteurs déployés. Un écart croissant est ensuite constaté entre nos propositions et GDRA. Pour GDRA, cela s'explique par le fait qu'en cas de collision, tous les lecteurs d'un voisinage donné sont désactivés. Dans DEFAR, mDEFAR et CORA, il y a toujours au moins un lecteur activé pour identifier les tags. Les trois propositions montrent des valeurs assez similaires avec mDEFAR et CORA fonctionnant légèrement mieux puisque la portée d'interférence et de communication réduites conduisent à l'augmentation du nombre de lecteurs activés.

— **Collisions** : Deux types de collisions sont identifiées, *collision d'accès au canal* et *collision de lecture*. Les *collisions d'accès au canal* se produisent lorsque plusieurs lecteurs choisissent le même slot de diffusion de message ou lorsque des lecteurs différents, dans le même voisinage, ne choisissent pas différents timeslots et/ou canaux (pour DEFAR, mDEFAR et GDRA). Les *collisions de lecture* surviennent lorsque deux ou plusieurs lecteurs accèdent en même temps à des tags dans leur portée de lecture (dans le cas d'accès multiples autorisés par CORA). La Figure 3.7b montre le nombre de collisions de lecture et d'accès au canal en fonction de la densité de lecteurs déployés. GDRA enregistre plus de collisions que nos propositions en raison de son processus de résolution des conflits. En effet, alors que nous pouvons avoir jusqu'à $max_slots \times 4$ pour DEFAR ou

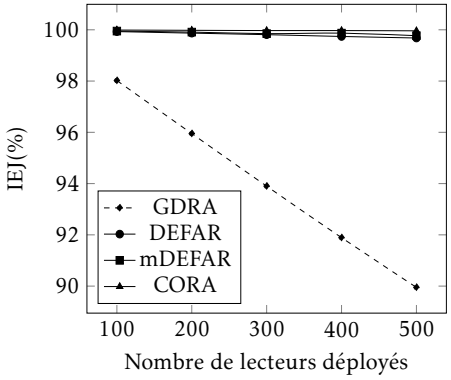


(a) Débit de lecture

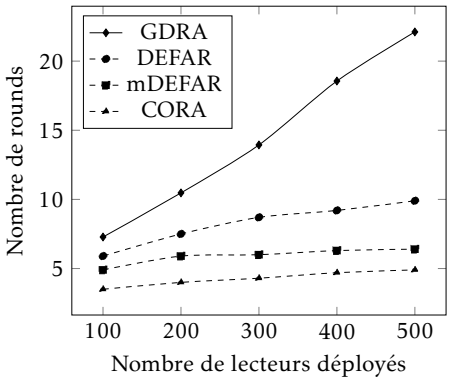
(b) Collisions



(c) Efficacité



(d) Indice d'équité de Jain



(e) Délai de couverture

FIGURE 3.7 – Résultats de simulations sur déploiements statiques

max_slots pour mDEFAR, seul un seul lecteur peut être activé dans le voisinage d'un lecteur avec GDRA. Ceci conduit à un plus grand nombre de lecteurs inactifs, d'où le nombre de collisions enregistrées.

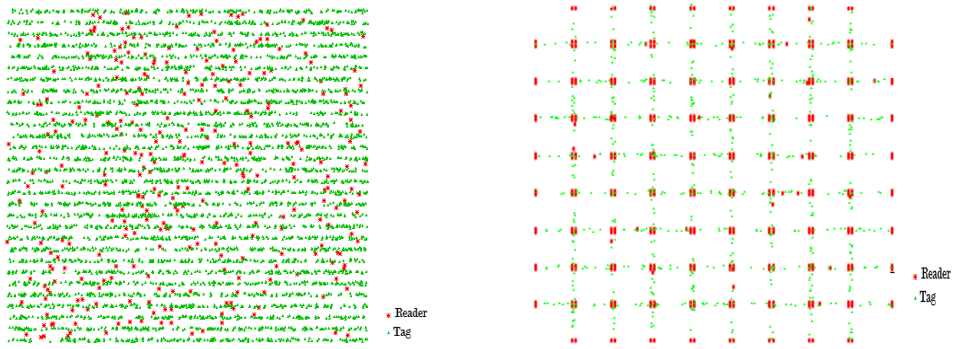
— **Efficacité** : Sur la Figure 3.7c, l'efficacité pour tous les protocoles comparés en fonction de la densité est montrée. Comme ce résultat est une combinaison des deux premières métriques, cela explique pourquoi GDRA a l'efficacité la plus faible (passant de 92% à 66%) alors que toutes nos propositions restent au-dessus de 95%. Cela prouve que nos propositions sont toutes bien adaptées aux déploiements denses.

— **Indice d'équité de Jain** : Puisque les simulations ont été effectuées avec des lecteurs statiques déployés de manière aléatoire, il est intéressant de comprendre à quel point chaque lecteur obtient un accès équitable au canal pour lire les tags qu'il couvre. Si l'accès au canal d'interrogation n'est pas équitablement distribué, certains tags peuvent ne pas être identifiés. La Figure 3.7d montre que tous les algorithmes testés fonctionnent très bien en termes d'équité puisque tous ont des valeurs supérieures à 90%. Cependant, DEFAR et mDEFAR obtiennent de meilleurs résultats grâce aux différents niveaux de priorité introduits qui donnent aux lecteurs défaillants une meilleure chance de réussir dans les *rounds* suivants.

— **Délai de couverture** : Comme les lecteurs sont ici statiques et que les tags et les lecteurs ont été déployés aléatoirement, tous les tags ne sont pas forcément couverts et à portée de lecture. Seuls les tags couverts par des lecteurs ont été pris en compte dans la Figure 3.7e. Elle montre que nos propositions fonctionnent plus rapidement que GDRA, ce qui est expliqué par les résultats précédents. Il est cependant intéressant de voir que le compromis concernant les collisions faites avec CORA permet une couverture plus rapide. En ce qui concerne la différence entre DEFAR et mDEFAR, l'utilisation d'un seul canal a permis de réduire la plage d'interférence. Comme moins de lecteurs entrent en collision, d'autres sont activés et les tags sont identifiés plus rapidement.

3.3.2 Déploiements dynamiques

Pour ces déploiements, nous avons considéré deux scénarios : un entrepôt avec des produits stockés à lire et une ville intelligente avec des infrastructures urbaines à surveiller ainsi que des véhicules à contrôler. Ces scénarios choisis permettent de valider aussi bien les déploiements denses que la mobilité des lecteurs et des tags.



(a) Capture entrepôt (b) Capture ville intelligente

FIGURE 3.8 – Scénarios simulations dynamiques

Entrepôt

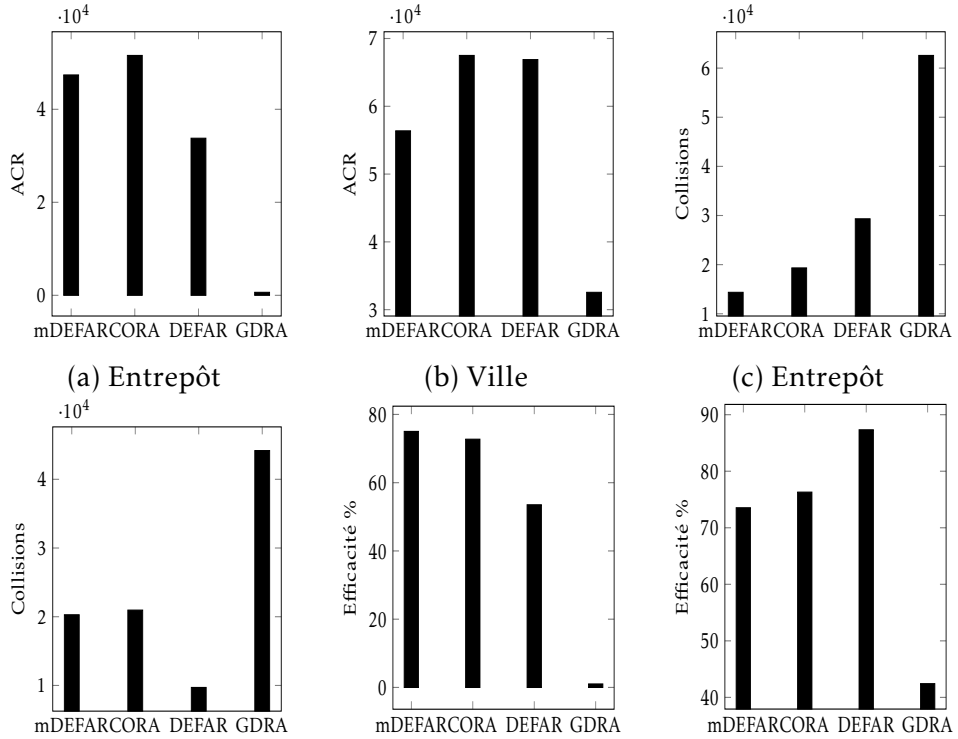
Nous considérons un large entrepôt d'une surface de $200 \times 200\text{m}^2$ avec 6000 tags déployés supposés être rattachés aux produits stockés dans les rayons. Les rayons sont alignés horizontalement et sont séparés de 7m. Cela correspond à un total de 30 rayons de 200 tags chacun. 10 lecteurs sont déployés entre chaque rayon pour un total de 290 lecteurs. Ces derniers sont supposés mobiles et se déplacent à une vitesse de 0.7m/s le long des rayons et font demi-tour après avoir atteint l'un des bouts de l'entrepôt. Cette vitesse a été déterminée pour représenter la marche d'un ouvrier détenant sur lui un lecteur RFID portable. Une capture instantanée de ce scénario est représentée sur la Figure 3.8a.

Ville intelligente

Un segment de ville intelligente est représenté sur une surface de $500 \times 500 m^2$ avec 800 tags fixes et mobiles déployés. Les tags sont utilisés pour superviser l'état des infrastructures urbaines et contrôler le trafic des véhicules. Les véhicules se déplacent avec une vitesse moyenne de 10m/s. Les rues ont une largeur de 5m et sont perpendiculaires se croisant tous les 50m. Deux types de lecteurs sont identifiés, des fixes et des mobiles pour un total de 352 lecteurs. Les fixes sont placés à chaque coin de rue. Les mobiles quant à eux sont montés sur des vélos et véhicules de transport en commun se déplaçant avec une vitesse moyenne de 4m/s. Le modèle de mobilité considéré est celui de Manhattan pour l'ensemble des véhicules. Ils se déplacent donc le long des rues avec des probabilités de tourner à gauche (0.25) ou à droite (0.25), continuer (0.4) et faire demi-tour (0.1) à chaque intersection. Ce scénario permet de mettre à l'épreuve nos propositions pour résoudre les contentions dans des environnements denses et mobiles irréguliers. Une capture instantanée est illustrée sur la Figure 3.8b.

Résultats

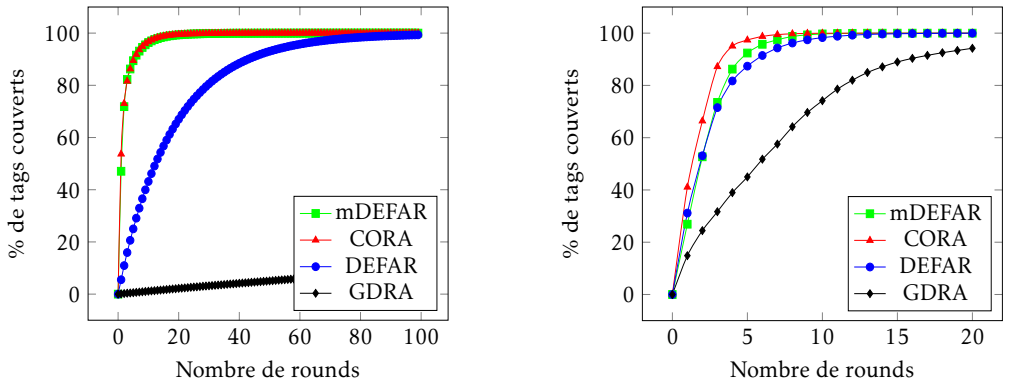
— **Débit de lecture** : Dans ce cas, le débit peut être considéré comme un gain de capacité de suivi pour les applications de sécurité, car plus le débit est élevé, plus la précision des données concernant les produits est élevée. La Figure 3.9a montre le débit des différents algorithmes dans le scénario d'entrepôt. Nous pouvons voir que CORA domine clairement avec une valeur de débit plus élevée. En effet, dans l'entrepôt où les lecteurs sont entre des rayons avec des tags à l'extrémité de la portée de lecture, la collision dans une zone sans tag n'est pas nuisible et CORA en profite, d'où les résultats. mDEFAR fonctionne également très bien grâce à sa portée d'interférence réduite. GDRA, cependant, offre de mauvais résultats en raison du nombre élevé de lecteurs en collision. Dans la Figure 3.9b, les résultats sont différents pour mDEFAR et DEFAR. En effet, dans le scénario urbain où nous avons des lecteurs mobiles et statiques, DEFAR fonctionne mieux puisqu'il est capable d'atteindre un état de convergence pour les lecteurs statiques situés dans les coins et de lire



(a) Entrepôt (b) Ville (c) Entrepôt

(d) Ville (e) Entrepôt (f) Ville

FIGURE 3.9 – Comparaison débit de lecture, collisions et efficacité dans les différents scénarios



(a) Entrepôt (b) Ville

FIGURE 3.10 – Résultats délai de couverture

activement les tags, se rapprochant ainsi des résultats obtenus avec CORA, contrairement à mDEFAR qui désactive plus de lecteurs que DEFAR car n'utilisant qu'un seul canal.

— **Collisions** : Une collision dans un environnement de tags mobiles peut signifier qu'un tag n'a pas été identifié, ce qui correspond pour une chaîne de production à un produit défectueux qui pourrait ne pas être identifié. Dans les Figures 3.9c et 3.9d, GDRA montre le plus grand nombre de collisions, ce qui est attendu au vu de sa mauvaise performance en débit de lecture. Il est à nouveau intéressant de voir que dans la Figure 3.9d, DEFAR a les valeurs de collision les plus faibles, ce qui s'explique par l'état de convergence qu'il est capable d'atteindre dans cet environnement mixant des lecteurs statiques et mobiles. mDEFAR et CORA offrent tous deux des résultats similaires.

— **Efficacité** : D'après les résultats précédents, en termes de débit de lecture et de collisions, les valeurs d'efficacité peuvent être prédites. Ainsi, dans l'environnement entièrement mobile de l'entrepôt (Figure 3.9e), GDRA affiche une efficacité remarquablement faible (environ 1%) par rapport à nos propositions, ce qui prouve qu'il n'est pas adapté aux environnements mobiles. Cependant, mDEFAR offre les meilleures performances malgré un débit inférieur à celui de CORA (voir Figure 3.9a), mais moins de collisions enregistrées le rendent légèrement plus efficace (75% pour mDEFAR et 72% pour CORA). Dans le scénario de la ville (Figure 3.9f), les résultats sont meilleurs pour GDRA (42%), puisque les lecteurs statiques aux coins de rue bénéficient mieux de l'algorithme, mais restent inférieurs à nos propositions. DEFAR atteint les meilleurs résultats (87%) tandis que mDEFAR et CORA sont encore assez similaires (respectivement 73% et 76%).

— **Délai de couverture** : Dans un environnement mobile, il peut être intéressant d'obtenir toutes les informations des tags le plus rapidement possible pour connaître l'état du système à chaque instant. La Figure 3.10 montre la proportion de tags couverts au cours du temps. Dans l'application d'entrepôt (Figure 3.10a), alors que mDEFAR et CORA ont des tracés qui se chevauchent et obtiennent une couverture presque totale après 20

rounds, GDRA de son côté peine à atteindre une couverture de 10% sur la durée de la simulation. DEFAR, cependant, atteint lentement la couverture totale après 80 *rounds*. Pour l'application urbaine (Figure 3.10b), les résultats sont meilleurs pour GDRA qui atteint 94% de couverture après 20 *rounds*, alors que nos propositions atteignent déjà 100%. CORA (12 *rounds*) atteint une couverture totale légèrement plus rapide que DEFAR (22 *rounds*) et mDEFAR (16 *rounds*). Même si DEFAR a eu une meilleure efficacité et moins de collisions, il n'était toujours pas aussi rapide que CORA puisque le gain d'efficacité de DEFAR (voir Figure 3.9d) est offert par les lecteurs statiques qui ne couvrent que les tags passant sous leur portée d'interrogation tandis que les lecteurs mobiles avec CORA et mDEFAR apportent le gain de délai de couverture.

Les faibles performances de GDRA par rapport à nos propositions peuvent être expliquées par la procédure de contention utilisée. En effet, alors que dans nos propositions nous pouvons toujours assurer qu'au moins un lecteur accède aux tags dans son voisinage en cas de collisions multiples, dans GDRA en cas de collision, non seulement tous les lecteurs concernés sont inactifs mais les lecteurs voisins sont également désactivés. Cette proportion importante de lecteurs désactivés sur un *round* explique le faible débit de lecture, les collisions et la latence élevées. D'après nos simulations, nous pensons que GDRA pourrait être plus adapté aux environnements moins denses où moins de lecteurs sont impliqués dans des procédures de contention.

Ces résultats nous confortent dans notre idée qu'un algorithme d'anti-collision unique ne peut convenir à différents déploiements et applications. En effet, les métriques ciblées (débit, collisions, efficacité, équité, latence, énergie ...), ainsi que la nature des lecteurs et tags déployés (mobiles et/ou statiques) devraient guider le choix de la configuration du système anti-collision. En ce qui concerne nos contributions, nous pensons que lorsque le multicanal est impératif, seul DEFAR doit être considéré car il est le seul conforme et offre des améliorations substantielles sur GDRA dans toutes les métriques considérées. Lorsque, dans le cas d'un déploiement mobile, tel qu'un entrepôt dynamique avec des lecteurs mobiles et des tags itinérants dans la zone de déploiement, mDEFAR devrait être choisi comme un protocole global plus performant. Si l'objectif est d'offrir

la couverture la plus rapide et un débit plus élevé, CORA est le meilleur choix. Alors que dans le cas d'un déploiement hybride de lecteurs mobiles et statiques, comme cela pourrait être le cas dans une ville intelligente avec des applications différentes fonctionnant en parallèle, DEFAR se révèle être le meilleur en termes d'efficacité, donc mieux adapté sur le long terme et la stabilité. Toutefois, CORA devrait être choisi si le système doit identifier rapidement tous les tags déployés et réduire la latence.

3.4 Conclusion

Le constat fait de l'état de l'art a su guider la conception des différents algorithmes proposés. En effet, le manque de solutions performantes pour des applications dynamiques avec des lecteurs et des tags déployés en grand nombre et mobiles nous a poussé à formuler nos contributions. Tous les 3 algorithmes proposés sont compatibles avec le standard ETSI [1] et offrent des performances intéressantes pour les scénarios considérés par nos applications.

Nous remarquons également des variations de performances entre nos différentes propositions en fonction des scénarios considérés (statiques, mobiles et hybrides). Ce qui valide l'hypothèse que le choix d'un algorithme d'anti-collision ou d'un autre doit être guidé par les contraintes environnementales de déploiement mais aussi les attentes applicatives.

Toutefois, la mise en place d'une solution d'anti-collision, même performante, à elle seule n'assure pas tous les rôles nécessaires à la surveillance d'une ville ou d'un entrepôt. Les informations récupérées depuis les tags par les lecteurs RFID doivent être collectées et remontées vers un centre de décision afin d'être exploitées et programmer une réaction. Il est donc impératif de mettre en place des techniques de collecte des données réutilisant l'architecture et la topologie présentes en évitant la perte d'informations et en maintenant une latence raisonnable.

Publications

Journal

Abdoul Aziz Mbacké, Nathalie Mitton et Herve Rivano. "RFID Reader Anticollision Protocols for Dense and Mobile Deployments" dans *MDPI Electronics Special Issue "RFID Systems and Applications", 2016*

Conférences

Abdoul Aziz Mbacké, Nathalie Mitton et Herve Rivano. "Distributed Efficient & Fair Anticollision for RFID Protocol" dans *WiMob 2016 - IEEE 12th International Conference on Wireless and Mobile Computing, Networking and Communications, Oct 2016, New York, Etats-Unis, 2016*

Arunabha Sen, Arun Das, Chenyang Zhou, Anisha Mazumder, Nathalie Mitton et Abdoul Aziz Mbacké. "Reader Scheduling for Tag Access in RFID Systems" dans *NoF 2016 - 7th International Conference on Network of the Future, Nov 2016, Buzios, Rio de Janeiro, Brésil, 2016*

Abdoul Aziz Mbacké, Nathalie Mitton et Herve Rivano. "Protocoles d'anticollision de lecteurs RFID pour des déploiements denses et mobiles" dans *CoRes 2017- 2ème Rencontres Francophones sur la Conception de Protocoles, l'Évaluation de Performance et l'Expérimentation des Réseaux de Communication, Mai 2017, Quiberon, France, 2017*

Abdoul Aziz Mbacké, Nathalie Mitton et Herve Rivano. "RFID Anticollision in Dense Mobile Environments" dans *WCNC 2017 - IEEE Wireless Communications and Networking Conference, Mar 2017, San Francisco, Etats-Unis, 2017*

Chapitre 4

Remontée d'informations RFID

Les villes intelligente, à travers les applications conçues, sont prévues pour améliorer le quotidien et le bien-être des citadins dans un environnement durable et plus sécurisé. A cet effet, plusieurs capteurs sans fil sont aujourd'hui déjà déployés dans plusieurs villes afin de surveiller les infrastructures urbaines (ponts, tunnels, routes, bâtiments publics, etc.) et d'autres équipements urbains (éclairage public, bennes à ordure, feux tricolores, etc) dans le but de recevoir des données cruciales pour leur entretien et maintenance. Cette thèse étudie l'utilisation de la technologie RFID en complément des réseaux de capteurs actuels dans le but de rendre les solutions plus fiables, facile à mettre en œuvre et à maintenir et surtout moins énergivores. La contrainte des collisions pouvant être levée grâce aux propositions d'algorithmes d'anti-collision, la nécessité de pouvoir remonter les données demeure et constitue l'enjeu majeur abordé dans ce chapitre.

L'idée est ici de pouvoir réutiliser la topologie et les caractéristiques des lecteurs RFID déployés afin de proposer une solution efficace et rapide de remontée des données issues des tags RFID. La solution proposée devra aussi prendre en compte non seulement la nature des informations remontées mais également les interrogations des lecteurs afin de permettre une intégration transparente.

4.1 RFID en complément des réseaux de capteurs classiques pour la ville intelligente

Initialement conçue dans le but d'identifier des biens, la technologie RFID a depuis ses débuts fort évolué et permet de nos jours, au delà de l'identification, de lire des données environnementales grâce des capteurs couplés aux tags. Ces nouveaux tags qu'on nommera "sensing tags" permettent d'effectuer des mesures jusqu'alors réservées aux capteurs et ce tout en gardant les avantages de la RFID passive.

Les solutions basées sur les réseaux de capteurs traditionnels rencontrent plusieurs défis et lacunes à savoir :

— **le coût** : le prix d'un capteur traditionnel peut vite devenir important en fonction de la précision des données voulues. De plus, afin de récupérer les données issues de ces derniers, il est généralement nécessaire de les équiper d'interfaces de communication soit de type cellulaires (GPRS, 3G, 4G, etc), soit de type longue portée avec faible consommation énergétique (Low Power Wide Area Network LPWAN) : LoRa, Sigfox, etc). Ces modes de communications peuvent être assez onéreux vu qu'ils nécessitent un abonnement à un fournisseur de service avec des paiements périodiques ou au fur et à mesure de l'utilisation. De plus, la couverture du service offert n'est pas toujours garantie en fonction des zones de déploiement prévues ;

— **la facilité d'utilisation** : quand bien même ils pourraient être considérés comme relativement petits, les capteurs n'en sont pas moins encombrants par leur taille pour certaines applications où il est désirable de les garder invisibles à l'œil nu des citadins. Ceci combiné au coût, avancé plus haut, impacte le nombre de capteurs pouvant être déployés, ce qui en retour affecte la précision des données ;

— **la durée de vie** : les capteurs déployés s'appuient sur une batterie pour leurs activités de détection et de communication. Afin d'allonger l'autonomie, soit la fréquence d'échanges de données est réduite, soit les systèmes utilisent des techniques de récupération d'énergie. Néanmoins, la durée

de vie des batteries limite leur utilisation avant qu'une campagne de remplacement des batteries ne soit demandée ;

— **l'environnement de déploiement** : les capteurs peuvent nécessiter d'être enterrés, noyés, rattachés à du métal ou dans du béton. De tels environnements affectent la fiabilité de la communication de ces appareils. De même, leur déploiement dans les infrastructures est compliqué et coûteux en termes d'ingénierie civile.

Fort heureusement, les sensing tags RFID permettent aujourd'hui de remédier à ces défis. Concernant le coût, les tags RFID reviennent à un dixième du prix de ces capteurs et sont basés sur une distribution de plusieurs tags avec des lecteurs placés à des endroits stratégiques. En ce qui concerne la facilité d'utilisation, la taille des tags RFID est généralement comprise entre quelques millimètres et quelques centimètres. Ils sont donc facilement dissimulables pour rester invisibles à un œil non-averti. Ceci combiné à leur faible coût permet d'entreprendre des déploiements importants qui améliorent la précision des données recueillies. La durée de vie quant à elle est pratiquement infinie étant donnée que ces tags fonctionnent grâce aux signaux d'interrogation des lecteurs à portée. Finalement, l'environnement de déploiement n'est pas un frein en ce qui concerne la RFID. En effet, des tests menés dans différents environnements pouvant être considérés comme hostiles (noyé dans l'eau [14], en compagnie de métaux [69], enterré [4] ou encore enfouis dans le béton [6]) se sont révélés concluants. Cela permet d'envisager le déploiement de ces solutions RFID en complément des solutions de réseaux de capteurs traditionnels pour mieux couvrir la ville et ses environs et offrir des services aux citadins toujours plus avancés et précis.

4.1.1 Sensing tags

En plus des propriétés d'identification prêtées à la RFID, le développement de la technologie permet aujourd'hui d'avoir accès à tout un panel d'applications et cas d'utilisation potentiels. Ce développement passe par la possibilité de récupérer plus que l'énergie nécessaire à la réponse du tag lors de l'interrogation par le lecteur. Trois types de sensing tags sont principalement utilisés.

Dans [50], les auteurs évaluent plusieurs solutions de récupération d'énergie et analysent en particulier l'opportunité donnée par la réutilisation de l'énergie d'interrogation en RFID. Suite à ces travaux, une première version d'un "tag passif augmenté" est proposée dans [88]. Ce tag enregistreur de donnée passif (Passive Data Logger PDL) permet de rattacher des capteurs à un tag RFID passif. Ces capteurs sont ensuite alimentés par un condensateur présent sur le tag qui lui-même se recharge lors des interrogations successives par les lecteurs à portée. Ainsi, lors des interrogations reçues, le condensateur emmagasine l'énergie qui sera plus tard réutilisée pour alimenter des capteurs à intervalles réguliers pour enregistrer des données environnementales. Ces données sont ensuite stockées sur une mémoire locale au tag et renvoyées lors des lectures suivantes. Des tests ont été menés avec différents capteurs tels que ceux de luminosité, température, orientation et tension électrique. Afin de valider leur solution, les auteurs ont testé l'installation de leur tag sur une brique de lait pour surveiller la température et le niveau de remplissage de la brique. Les tests se sont montrés positifs avec une portée de lecture allant jusqu'à 3 mètres.

Une alternative proposée au tag PDL est le "tag capteur amélioré" (Enhanced Sensor Tag ES-Tag) [15] avec des capteurs, possédant leur propre source d'alimentation, et connectés au tag pour la lecture des données. Ainsi, les mesures des capteurs sont stockées sur la mémoire du tag et seront envoyées aux lecteurs, en même temps que les données d'identification, lors de l'interrogation. Des tests ont été menés avec des capteurs de tension artérielle, taux de glycémie, température et humidité. Les résultats ont montré que le tag, bien que rattaché à un organisme vivant, permettait de communiquer sans interférence sur une portée allant jusqu'à 6.5 mètres.

D'autres exemples sont également disponibles, avec la proposition faite dans [93] d'utiliser des capteurs de température et de pression pour les applications industrielles. Ou encore dans [7], où les auteurs couplent un tag, noyé dans du béton, avec des capteurs de température, d'humidité et de niveau de pH pour la surveillance des constructions et de la structure du béton. Des applications médicales sont également prévues dans [81] avec des capteurs de température reliés aux patients et rattachés à des tags passifs pour la surveillance de leur état de santé. Ces derniers tags sont couplés avec des capteurs passifs

également. Le capteur est alimenté au moment de la lecture du tag. Les données environnementales sont acquises à la lecture et sont renvoyées en même temps que les données d'identification du tag.

4.1.2 Exemples d'applications utilisant la RFID

Nous retrouvons aujourd'hui plusieurs applications, basées sur la RFID, disponibles dans les villes pour améliorer le quotidien des citadins. Grâce aux propriétés d'identification couplées à celle de capture de données, la RFID se trouve être une des meilleures options pour les applications de logistique. Dans [49], plusieurs exemples sont donnés de l'utilisation de la RFID pour la traçabilité des biens entreposés et déplacés dans les entrepôts et usines. Ces même tags sont ensuite réutilisés pour la livraison dans les points de stockage et de vente. L'utilisation de la RFID pour le paiement du trajet par les voyageurs dans les bus est également étudiée dans [96] et mise en place aujourd'hui dans plusieurs villes. Les auteurs proposent le placement de lecteurs RFID dans les bus permettant une lecture sur une portée allant jusqu'à 3 mètres. Une telle solution permettrait d'avoir une idée du nombre de passagers dans les transports en commun et comment améliorer les services offerts. Une étude de plusieurs solutions de surveillance de la structure des équipements est proposée dans [95]. Les tags RFID peuvent être directement imprimés sur les structures à surveiller, passant totalement inaperçus, et permettent de récupérer des informations sur l'état des équipements urbains. L'utilisation de la RFID permet d'effectuer des mesures critiques dans des environnements où l'activité humaine serait trop dangereuse comme c'est le cas sur les centrales nucléaires, les usines à gaz et pétrole ou des infrastructures telles que les ponts, les routes ou les chemins de fer. La Figure 4.1 montre l'exemple d'un pont surveillé par des capteurs traditionnels et une alternative avec des "sensing tags" RFID. On peut voir que l'alternative permet de récupérer plus d'informations et d'obtenir une meilleure couverture. Des tags ont même été proposés pour la mesure de température dans les chambres de combustion des turbines à gaz avec des températures pouvant aller jusqu'à 1000°C [92]. L'amélioration de la gestion des ordures ménagères en utilisant la RFID est également proposée dans [16]. Une telle application

permet d'optimiser le passage des bennes à ordures par l'anticipation d'un trajet de récupération.

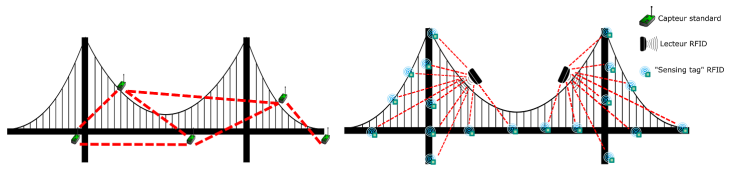


FIGURE 4.1 – Comparaison solutions capteurs traditionnels et "sensing tag" RFID

4.2 Remontée de données RFID

La plupart des solutions RFID existantes se basent sur l'utilisation de lecteurs RFID directement connectés à un réseau de communication (réseau mobile, Wi-Fi, câble Ethernet, relié à un ordinateur, etc) afin de remonter les informations lues depuis les tags. Cependant, comme vu dans l'exemple des transports en commun équipés de lecteurs RFID, et se déplaçant à travers la ville, il serait intéressant de pouvoir réutiliser le grand nombre de bus ainsi que leurs parcours pour faciliter la collecte et l'échange des données. Dans [71], les auteurs proposent une architecture avec des lecteurs communiquant entre eux pour la remontée des informations lues depuis les tags vers une station de base de manière sans fil. Toutefois, cette architecture et sa mise en place ne sont pas explorées au-delà de cette présentation.

Dans le cas des réseaux de capteurs traditionnels, plusieurs solutions de routage ont été proposées au fil des années. Ces dernières fonctionnent aussi bien sur des déploiements de capteurs statiques ou mobiles, avec des communications sans fil de pair-à-pair et/ou hybrides utilisant d'autres technologies de communications. Ces algorithmes ont généralement pour objectif principal d'arriver à mettre en place des chemins et les maintenir de manière efficace afin d'avoir des échanges de données fiables. Toutefois, au regard des problématiques de ces types de réseaux, relatifs aux points soulevés plus-haut, il est nécessaire d'arriver à produire un algorithme de routage qui permet d'allonger autant que

possible la durée de vie du réseau.

En général dans les réseaux de capteurs, il est supposé que les noeuds ne peuvent directement joindre la station de base et qu'il est donc nécessaire de mettre en place une solution permettant le relais des données de pair-en-pair. Dans [78], les auteurs proposent une classification des protocoles des réseaux de capteurs en fonction de 5 paramètres. La Figure 4.2 reprend la classification proposée.

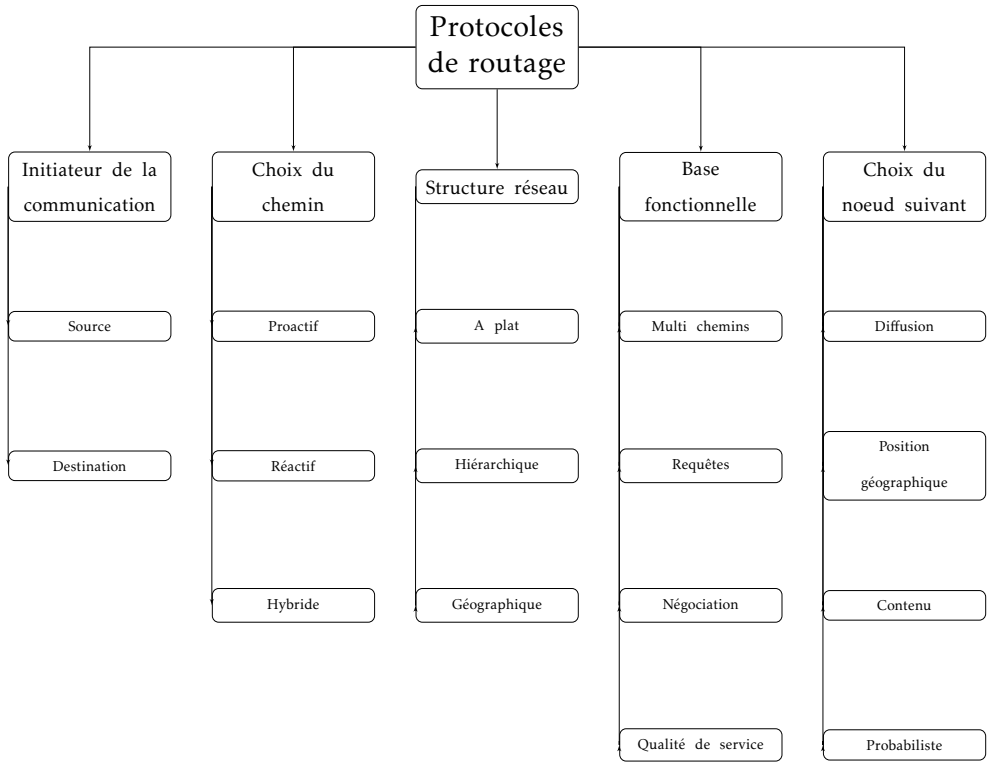


FIGURE 4.2 – Taxonomie des protocoles de routage [78]

Usuellement, les lecteurs RFID sont directement connectés à un ordinateur disposant d'une application chargée de traiter les données lues. Avec des lecteurs mobiles, les informations récupérées depuis les tags peuvent également être stockées au niveau du lecteur pour un traitement ultérieur. Le grand nombre de lecteurs qu'il est nécessaire déployer et la réactivité attendue des applications prévues font qu'il est impératif d'analyser et éventuellement déclencher un correctif le plus rapidement possible. Au vu de la densité de déploiement et de la

mobilité des lecteurs ainsi que des interfaces de communication qu'ils possèdent leur permettant déjà de coordonner leur activité de pair-à-pair, une solution de remontée des données de manière multi-sauts serait la plus opportune. La Figure 4.3 montre un exemple avec des lecteurs RFID mobiles qui peuvent se déplacer afin de lire les tags à portée et communiquant entre-eux pour remonter les données vers la station de base. En effet, en utilisant la forte proximité des lecteurs, relayer les informations entre eux vers une station de base permet de se passer d'interfaces de communication à longue portée ou dépendant d'un réseau tiers (3G, FAI, SigFox, LoRa, etc) rendant la solution moins onéreuse. La mise en place d'un algorithme de routage multi-sauts remontant les informations jusqu'à un puits qui sera lui plus tard chargé de renvoyer les données vers l'entité de traitement et de décision serait le plus adéquat.

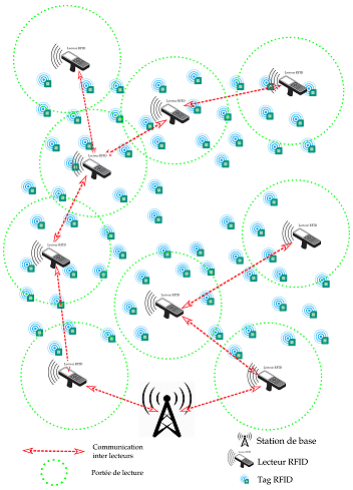


FIGURE 4.3 – Exemple d'un réseau multi-sauts RFID

Plusieurs protocoles issus des réseaux de capteurs, en particulier les réseaux AdHoc mobiles (Mobile Adhoc NETworks MANET), ont été proposées avec des schémas de routage multi-sauts haute performance [56]. Mais ces propositions ne peuvent malheureusement pas être directement appliquées à notre étude des systèmes RFID. En effet, dans les MANET, les nœuds communiquent leurs propres lectures issues des capteurs embarqués, alors que dans les solutions RFID considérées, les lecteurs couvrent plusieurs tags envoyant chacun leurs informations à transmettre. La quantité, la taille et le type de données à transmettre

sont donc différents. Toujours dans nos systèmes RFID, les lecteurs envoient des données en fonction de leur activité d'interrogation définie par l'algorithme anti-collision choisi, ce qui n'est pas le cas dans la plupart des propositions MANET. Concevoir conjointement les schémas d'ordonnancement et d'acheminement des activités permettrait au système d'être plus économe en énergie et d'éviter le chevauchement des phases d'interrogation et d'acheminement. Ce qui rendrait la solution plus efficace. Dans la mesure de nos connaissances et de l'état de l'art étudié, le routage multi-sauts pour les données issues de tags RFID en accord avec l'activité des lecteurs n'a toujours pas été considéré.

Pour répondre à ce manque, nous proposons par la suite Distributed & Adaptable Crosslayer Anti-collision and Routing (DACAR). Cet algorithme organise les lecteurs sous la forme d'un arbre menant à une station de base sensée récupérer les données en prenant en compte l'activité d'interrogation des lecteurs.

4.3 Distributed & Adaptable Crosslayer Anti-collision and Routing (DACAR)

DACAR est un protocole distribué de remontée de données RFID adaptable à différentes solutions d'anti-collision. Notre solution permet l'exploitation aussi bien de la forte densité de déploiement que de la mobilité des lecteurs utilisés. Il respecte les contraintes en termes d'efficacité et de latence des applications de villes intelligentes avec un faible taux de perte lors du relais et un délai de bout en bout compatible avec les applications considérées. Le relais des informations se fait grâce à une topologie sous forme de gradient avec les lecteurs auto-organisés autour d'un ou de plusieurs puits. Cette organisation en gradient nous permet de décaler les activités des lecteurs voisins réduisant les collisions. Le routage des informations est synchronisé sur l'activité des lecteurs pour réduire le délai de remontée des informations (les informations lues sont remontées au plus vite, réduisant la latence).

4.3.1 Positionnement

Ayant précédemment présenté des algorithmes d'anti-collision distribués, nous avons choisi de maintenir les mêmes caractéristiques pour la remontée de données. DACAR propose d'utiliser l'interface de communication dédiée des lecteurs pour l'organisation et la remontée des données. Pour rappel, cette interface dédiée, de portée notée $d_{Com} = 2 \times d_{CRT}$ dans le cas monocanal, permet aux lecteurs d'échanger des messages sans que ces derniers n'entrent en collision avec l'activité de lecture des tags.

Notre solution est également compatible avec la majorité des protocoles d'anti-collision TDMA disponibles de l'état de l'art. En effet, DACAR réutilise le découpage temporel en rounds et timeslots retrouvés dans la plupart des algorithmes TDMA centralisés et/ou distribués pour organiser la remontée des informations. Un lecteur pourra, durant un timeslot déterminé, lire les tags à portée sur son interface d_{CRT} et ensuite relayer les données vers ses voisins sur l'interface d_{Com}.

Les lecteurs sont organisés en fonction de leurs positions relatives au puits de manière à séparer leurs activités de lecture et de relais de données. Cette séparation et les différentes interfaces de communication utilisées permettront d'avoir un sous ensemble de lecteurs en interrogation de tags tandis que des sous ensembles voisins seront en train de renvoyer les informations vers le puits. La topologie choisie pour l'organisation des lecteurs est donc le **gradient**. Notre gradient pourra être représenté sous la forme d'un arbre où la racine est le puits et les noeuds et feuilles sont les lecteurs (voir Figure 4.4). Chaque noeud ou feuille sera numéroté en fonction d'un rang qui correspond à la distance relative le séparant du puits en nombre de sauts. Ainsi, plus un lecteur sera éloigné du puits, plus élevé sera son rang. On voit sur la Figure 4.4 que le lecteur R4, positionné à 2 sauts du puits, est de rang 2. Pareillement, le lecteur R10, situé à 3 sauts du puits, est de rang 3.

Afin de construire le gradient, le puits diffuse un message sur le canal de communication dédié des lecteurs. Ce message contient l'identifiant du puits id_{Sink} ainsi que son rang $rank_{Sink}$. Le puits étant la racine du gradient, son rang est égal à $rank_{Sink} = 0$ (Algo. 4 lignes 1-4 et voir Figure 4.4). Ce message sera

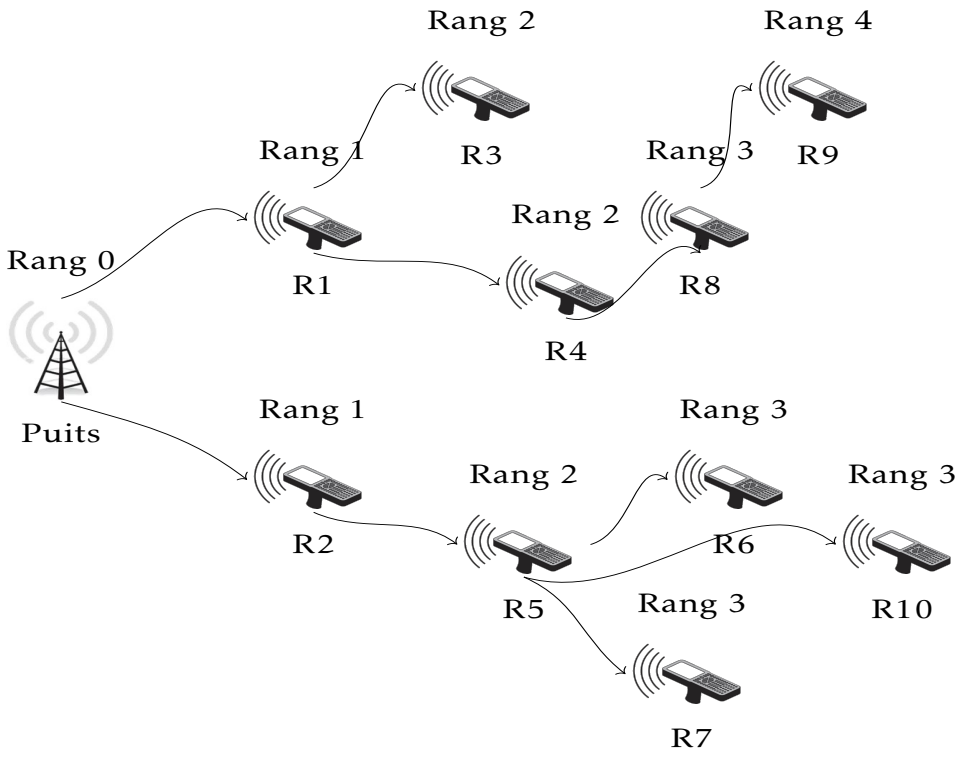


FIGURE 4.4 – Exemple d'un réseau multi-sauts RFID

reçu par les lecteurs à portée, qui, à la réception comparent leur rang $rank_i$, initialement fixé à une valeur infinie $rank_i = \infty$ (Algo. 4 lignes 5-6), à celui du message reçu $rank_{RX}$ (Algo. 4 ligne 8) :

— Si $rank_{RX} < rank_i - 1$, le lecteur définit son rang en fonction de celui reçu par incrémentation $rank_i = rank_{RX} + 1$ et enregistre la source du message comme son parent dans le gradient $parent_i = id_{RX}$. Le lecteur, après un temps aléatoire pour éviter les collisions, diffuse à son tour le message sur le canal de communication (Algo. 4 lignes 9-13) ;

— Sinon, le lecteur considère que le message provient d'un voisin de rang équivalent, supérieur et dans tous les cas n'en tient pas compte (Algo. 4 lignes 14-15).

Ce procédé classique [67, 48, 54] est répété par tous les lecteurs jusqu'à ce que chaque lecteur ait un rang dans la topologie.

Dans le cas d'utilisation de lecteurs mobiles, il est nécessaire de mettre à jour

Algorithm 4 Initialisation du gradient pour un noeud de rang $rank_i$

1: **if** $node == sink$ **then**
2: $rank_i = 0$,
3: $parent_i = 0$,
4: Diffusion message avec $rank_i$
5: **else**
6: $rank_i = \infty$
7: **end if**
8: A la réception d'un message avec $rank_{RX}$
9: **if** $rank_{RX} < rank_i - 1$ **then** ▹ message de rang inférieur
10: $rank_i = rank_{RX} + 1$,
11: $parent_i = id_{RX}$
12: Backoff(random) ,
13: Diffusion message avec $rank_i$ et id_i
14: **else** ▹ message de rang supérieur ou équivalent
15: Suppression message reçu
16: **end if**

le gradient vu que les lecteurs changent de positions et de rangs au cours de l'exploitation. Ainsi, à chaque fois qu'un lecteur devra envoyer une information, il diffusera un message auquel ses voisins répondront en renvoyant leurs valeurs $rank_j$ et id_j. Le lecteur utilisera ensuite ces valeurs pour mettre à jour son propre rang $rank_i$ et définir son parent dans le gradient $parent_i$ en suivant les règles établies (Algo. 4 lignes 8-16). Avec un déploiement dense de lecteurs, d'autres voisins pourraient profiter de cet échange pour recevoir de nouveaux paramètres et mettre également à jour leurs couples $rank_j$ et $parent_j$. Ces valeurs sont ensuite conservées avec une valeur de temporisation afin d'éviter trop d'échanges de messages et une consommation d'énergie inutile. En fonction du délai de tampon, un lecteur mettra à jour ses valeurs ou conservera celles déjà stockées. Cette valeur de délai tampon est définie en fonction de la vitesse des lecteurs et de la portée de communication. En fonction de l'algorithme anti-collision utilisé, les lecteurs se comportent différemment et le voisinage de chaque lecteur peut être affecté.

Une fois que tous les lecteurs ont obtenu un rang, les interrogations de tags se déroulent en fonction du protocole d'anti-collision en place. Notre al-

gorithme adaptable permet de conserver les timeslots utilisés dans la plupart des algorithmes TDMA. La seule contrainte qui sera imposée pour les lecteurs sera la plage de choix du timeslot en fonction du rang (Algorithme 5). Ainsi, un lecteur de rang pair ($rank_i \ mod \ 2 = 0$) choisira un timeslot pair dans l'intervalle $[0; max_slots[$ (Algo. 5 lignes 1-2), tandis qu'un lecteur de rang impair ($rank_i \ mod \ 2 \neq 0$) choisira un timeslot impair dans le même intervalle (Algo. 5 lignes 3-4). Un tel mécanisme permet de réduire par la même occasion le nombre de collisions de lectures des tags vu que deux lecteurs sur des rangs consécutifs ne pourront être en contention ou interroger les tags simultanément. Les lecteurs appartenant au même rang devront se soumettre au protocole d'anti-collision en exercice. Sur la Figure 4.5, on peut donc voir des lecteurs de rang pair en lecture tandis que d'autres de rang impair transmettent leurs données. Après contention, les lecteurs de rang pair R3, R5 et R9 accèdent au canal d'interrogation pour lire les tags tandis que les lecteurs R1, R8 et R10 transmettent leurs données lues précédemment car ils sont de rang impair.

Algorithm 5 Choix du slot pour un lecteur i

```
1: if Rang pair then
2:     slot_i ← nombre pair entre [0;max_slot]
3: else                                                    ▹ Rang impair
4:     slot_i ← nombre impair entre [0;max_slot]
5: end if
```

4.3.2 Fonctionnement

La collecte et le relais des données vers le puits sont faits en synchronisation avec la lecture des tags. Afin de limiter l'activité des lecteurs et réduire les collisions par la même occasion, nous définissons $max_slots = max_rank$. Ainsi, le plus haut rang dans le gradient définira aussi le nombre maximal de slots possibles pour le choix des lecteurs. Ce nombre sera connu grâce aux échange successifs lors de la construction du gradient. La collecte de données exploite la nouvelle planification temporelle basée sur le rang des lecteurs. La différence de temps observée entre les lecteurs pairs et impairs est utilisée pour effectuer l'acheminement des données pendant les périodes d'inactivité des lecteurs. Après avoir interrogé les tags lors du slot n, un lecteur va transférer les données

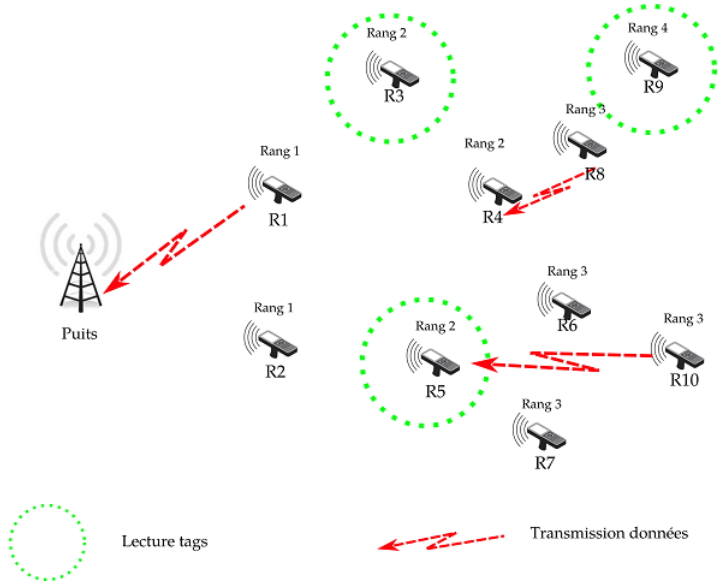


FIGURE 4.5 – Exemple d'organisation des lectures et transmission données avec DACAR

collectées vers son $parent_i$ lors du slot $n+1$. Puisque le canal de communication dédié est différent du canal d'interrogation, le transfert de données n'interférera pas avec l'activité de lecture. Sur la Figure 4.6, on peut donc voir que les lecteurs de rang impair effectuent leur contention pour ensuite accéder au canal d'interrogation, pendant que lecteurs de rang pair envoient leurs données. Ces phases s'alternent dans le temps jusqu'à la fin du *round* en cours.

Après avoir suivi la procédure de choix de slot décrite dans l'algorithme 5, les lecteurs entrent en contention pour l'accès au canal d'interrogation de tags. Cette contention se fera en fonction de l'algorithme d'anti-collision choisi. Ainsi, après une contention durant le slot n pour un lecteur i, différents résultats peuvent se produire (voir Algorithme 6) :

— le lecteur a des données à relayer. Ces dernières peuvent être soit issues de sa propre interrogation de tags, soit reçues depuis un lecteur voisin. Au slot $n+1$, le lecteur relaie les informations vers son $parent_i$ pendant la durée allouée à la lecture T_{CRT} et attend le prochain round. Si l'ensemble des données n'a pu être transmis durant le temps imparti, le lecteur continuera sa transmission à la prochaine occasion de relais. Afin d'éviter

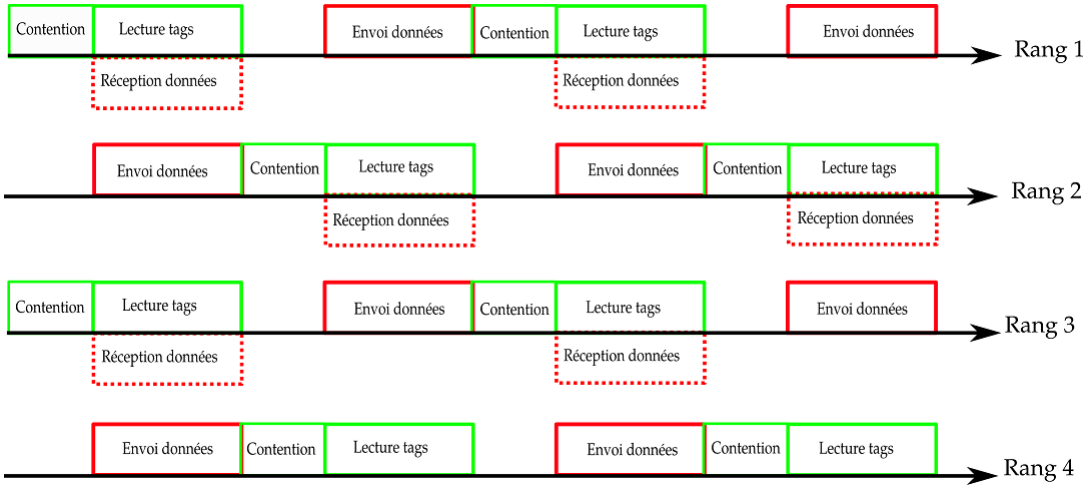


FIGURE 4.6 – Organisation de la contention, des lectures et de l'envoi des données avec DACAR

les collisions durant la transmission des données, les lecteurs tirent un temps aléatoire et suivent un mécanisme de Request To Send/Clear To Send (RTS/CTS) [79]avec leur $parent_i$ (Algo. 6 lignes 2-4) ;

— le lecteur n'a aucune information à transmettre. le lecteur n'a pas eu accès au canal d'interrogation des tags et/ou n'a pas reçu de données de ses voisins. Dans ce cas, le lecteur attend le prochain round pour entrer en contention à nouveau et avoir l'occasion de relayer ses données (Algo. 6 lignes 5-6) ;

— le lecteur a effectué sa contention durant le dernier slot disponible dans le round et donc ne peut relayer ses informations. Il attendra le round suivant pour participer à une nouvelle contention avec un nouveau $slot_i$ et alors relayer ses données (Algo. 6 lignes 7-8).

Lorsqu'un lecteur reçoit des données qui lui sont destinées, il les enregistre en attente de son slot de contention puis de relais (Algo. 6 lignes 10-13).

La Figure 4.7 représente un exemple avec une topologie de 7 lecteurs organisés autour d'un puits. Nous supposons ici que la construction du gradient a déjà été effectuée et que les lecteurs connaissent leur rang. Sur la Figure 4.7a, on peut voir pour chaque lecteur son $rank_i$ ainsi que son $parent_i$. Sur les figures suivantes, les petits cercles en pointillés désignent les activités d'interrogation de tags sur

Algorithm 6 Remontée de données avec DACAR

```
Après contention de lecture au slot n
if le lecteur a des données à relayer et slot_i pas le dernier then
    Envoi des données à parent_i
    Attente du round suivant
else if le lecteur n'a pas de données à transmettre then
    Attente du round suivant
else if Dernier slot du round then
    Attente du round suivant
end if
A la réception de données
if parent_RX == i then                                  ▹ Destinataire du paquet
    Stocke le paquet pour relais
end if
```

l'interface de portée d_{CRT} et les grands cercles désignent l'envoi des données sur l'interface de communication dédiée de portée d_{Com}. Nous supposons qu'un algorithme d'anti-collision est en cours, chargé de réguler les contentions d'accès au canal d'interrogation. Sur la Figure 4.7b, on est au slot 0 et donc les lecteurs de rang pair accèdent au canal. Ici les lecteurs F et G sont activés et lisent les tags à portée. Au slot suivant, illustré par la Figure 4.7c, les lecteurs impairs accèdent au cana, en l'occurrence ici les lecteurs A et B. Pendant ce temps, les lecteurs F et G qui avaient lu les tags à portée durant le slot précédent, relaient leurs données respectivement à $parent_F = A$ et $parent_G = E$. On peut remarquer que A reçoit les données de F pendant sa lecture de tags, cependant, l'usage de deux techniques de communication différentes sur des interfaces différentes nous permet ici de supposer que les activités n'entrent pas en collision. Par la suite, lors du slot 2, représenté par la Figure 4.7d, les lecteurs de rang pair interrogent les tags. Le lecteur C lit les tags à portée tandis que les lecteurs A et E relaient les données. En effet, le lecteur A ayant lu les tags à portée et reçu les données de F lors du slot 1, les relaie vers le puits. Le lecteur E lui renvoie les données reçues précédemment de G vers $parent_E = C$. Encore une fois l'activité d'interrogation de C n'entre pas en interférence avec la réception des données de E. Enfin, lors du dernier slot 3 (voir Figure 4.7e), le lecteur E interroge les tags à portée, tandis que le lecteur C relaie les données lues au slot précédent mais également celles reçues depuis E vers $parent_C = B$. E ayant effectué sa contention lors du dernier slot, attendra le prochain round pour à nouveau tirer un slot et participer à la contention et relayer ses données.

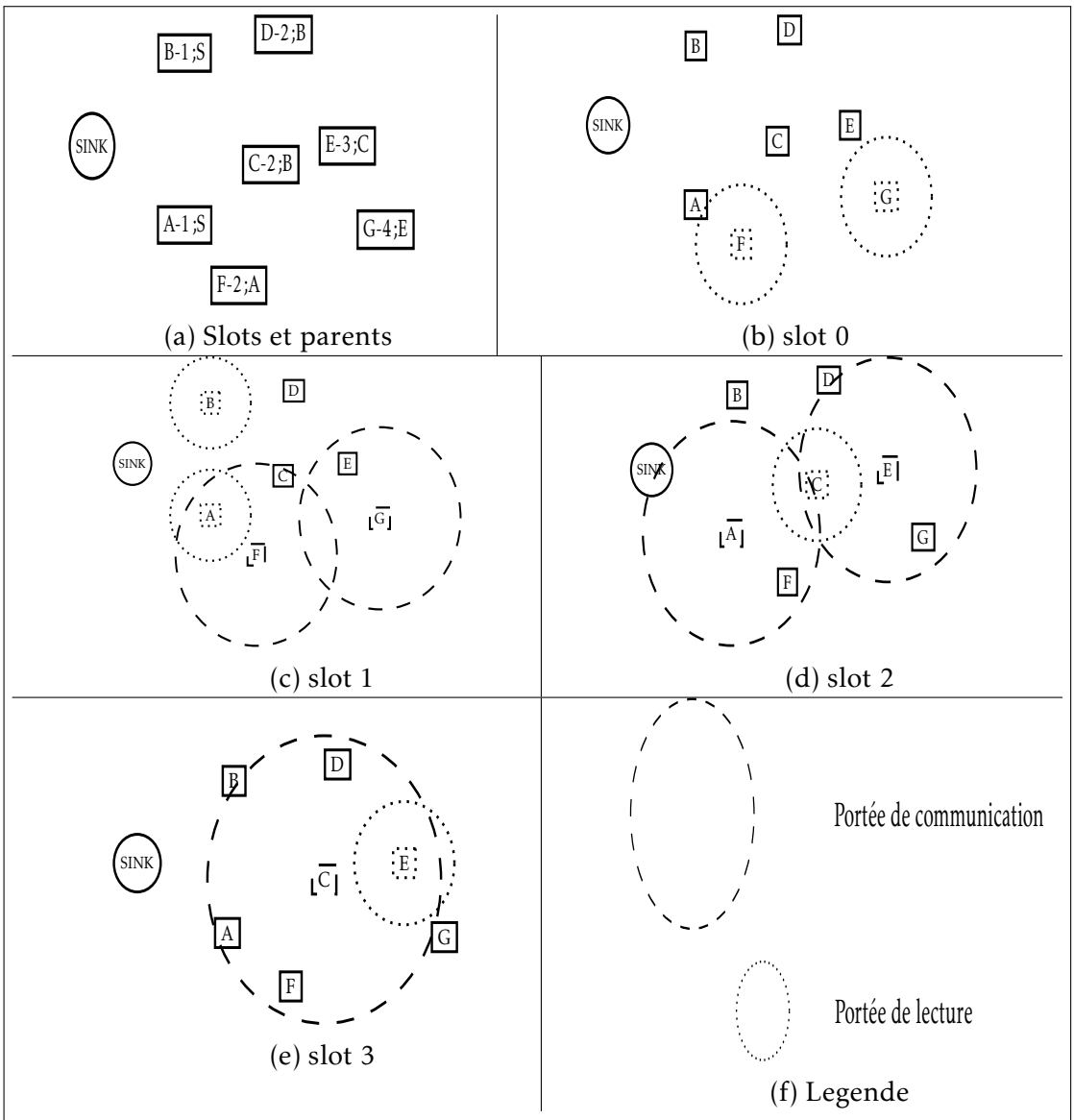
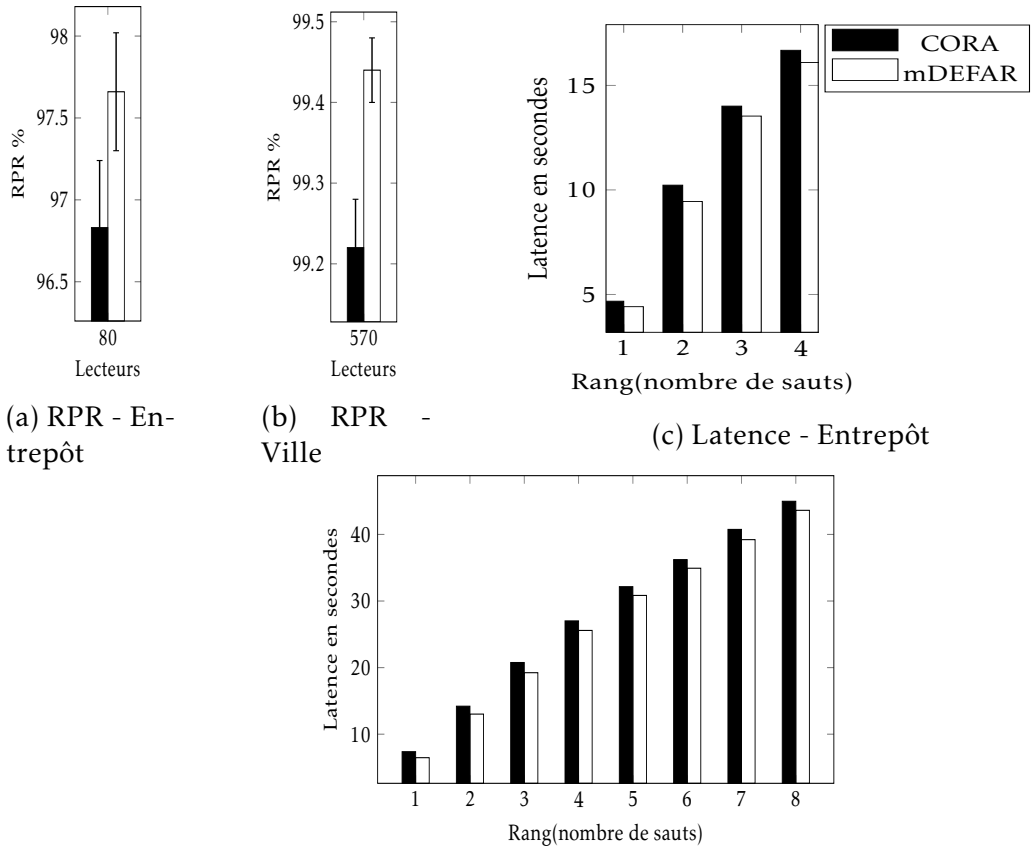


Figure 4.7 – Illustration DACAR

4.3.3 Résultats

L'évaluation de DACAR s'est faite par simulation toujours en utilisant notre simulateur WSNet. Nous avons là encore considéré les scénarios dynamiques introduits en Section 3.3. Nous avons placé pour le scénario d'entrepôt, un seul puits au centre et pour le scénario de la ville intelligente 5 puits. L'utilisation de plusieurs puits nous permet de valider le comportement de notre algorithme en déploiement large. Afin de gérer les collisions, nous avons implémenté DACAR avec les deux algorithmes d'anti-collision proposés (mDEFAR et CORA) qui avaient déjà montré des performances compatibles avec les déploiements denses et mobiles de lecteurs. Les résultats d'anti-collision ne seront pas présentés ici vu qu'ils restent en adéquation avec ceux déjà illustrés en Section 3.3.2.



(a) RPR - Entrepôt

(b) RPR - Ville

(c) Latence - Entrepôt

(d) Latence - Ville

FIGURE 4.8 – Résultats - DACAR

Les Figures 4.8a et 4.8b indiquent le Ratio de Paquets Reçus (RPR) avec succès depuis les lecteurs jusqu'à la (aux) station(s) de base. Il est calculé comme le ratio du nombre de messages reçus au(x) puits sur le nombre total de messages envoyés. Cette métrique permet d'évaluer la fiabilité de la technique de remontée. Un faible RPR désignera une remontée non fiable car avec beaucoup de paquets perdus. Dans le scénario d'entrepôt, le RPR de notre solution est supérieur à 96% alors que dans le scénario urbain, il est supérieur à 99% pour les deux algorithmes de contention. Les résultats plus faibles dans le scénario d'entrepôt sont dus à un plus grand nombre de tags et à un plus petit nombre de lecteurs les couvrant (4000 tags pour 80 lecteurs). Dans les deux cas également, nous pouvons voir que mDEFAR a des résultats légèrement meilleurs que CORA. Ceci est dû au fait que CORA ayant un débit plus élevé et un nombre supérieur de lecteurs actifs à chaque tour, a plus d'informations de tags à transmettre, générant ainsi plus de trafic. Plusieurs lecteurs impliqués entrent en collision, ce qui affecte le RPR. mDEFAR avec moins de lecteurs activés a un meilleur RPR. Cependant, avec de tels niveaux de RPR, les résultats sont assez fiables, puisque les tags peuvent être couverts par plusieurs lecteurs, donc même si l'information est perdue par un lecteur, elle sera récupérée par un autre voisin et transmise au(x) puits. Une fois de plus, la mobilité et la densité de déploiement deviennent un avantage pour les applications considérées.

La latence évalue le délai écoulé depuis la lecture d'un tag jusqu'à sa réception par l'un des puits. Dans DACAR, puisque les lecteurs sont organisés en topologie de gradient, la latence est donnée en fonction du rang relatif du lecteur source. Dans la Figure 4.8c, nous pouvons voir la latence, en secondes, pour les 4 rangs pendant les simulations. Nous pouvons voir que, en moyenne, le temps nécessaire pour qu'une information de tag atteigne le puits est d'environ 15 secondes, quel que soit l'algorithme d'anti-collision choisi. Cette durée semble être assez fiable pour l'application d'entrepôt concernée. En effet, dans le cas où un produit à l'intérieur de l'entrepôt est considéré défectueux en fonction des normes établies, en fonction de sa position, il faudra moins de 16 secondes pour que le système en soit notifié et que la contre-mesure appropriée puisse être appliquée. La Figure 4.8d montre les résultats de notre simulation en environnement urbain. En raison de la plus grande mobilité des lecteurs, les résultats sont

légèrement différents. Cependant, les résultats restent fiables avec en moyenne 44 secondes pour que les informations les plus éloignées atteignent un puits. Dans le cas où une infrastructure urbaine doit être évacuée en raison de mesures de sécurité, il faudra moins d'une minute pour que le système et les autorités en soient informés. En outre, dans le cas où un véhicule suspect fait l'objet de recherches, dans ce délai, il sera localisé et notifié aux autorités. Nous pouvons voir que dans les deux scénarios, mDEFAR a des résultats légèrement meilleurs que CORA, ceci est encore une fois dû au fait que mDEFAR, avec un débit de lecture inférieur et donc moins d'informations de tags à transmettre.

4.3.4 Discussion

Dans DACAR, la construction et la maintenance de la topologie sont basées uniquement sur le nombre de sauts entre les lecteurs et le puits. Si cette métrique peut être considérée comme fiable pour certaines applications où, par exemple, l'état du support radio n'est pas variable, plusieurs autres métriques peuvent être prises en compte pour proposer une solution qui pourrait être validée dans des scénarios industriels et urbains. En effet, ces environnements sont connus pour être instables et bruyants au niveau radio en raison du confinement ou de multiples autres sources d'émissions radios dans les villes. Nous pensons que la qualité de liaison entre les lecteurs, la latence, le taux de perte de paquets, les niveaux d'énergie entre voisins devraient également être considérés pour proposer une solution plus robuste. Sur la base de ce constat, nous avons donc proposé une solution améliorée de DACAR tenant compte de nouvelles métriques.

4.4 DACAR + Qualité de Service

Partant du constat précédent, nous avons décidé d'introduire une fonctionnalité de priorisation des données qui permet de définir si une information est soit critique, soit standard. En effet, toujours en restant dans l'exemple du scénario de l'entrepôt, dans le cas d'un produit déclaré comme impropre à la consommation dans un lot, il est important de pouvoir trouver le tag correspondant et remonter cette information au plus vite afin d'éviter des dommages ultérieurs.

Pareillement dans le cas de la ville intelligente, si un véhicule spécifique est recherché dans le cadre d'une enquête, si son tag correspondant est lu par un lecteur, l'information correspondante doit être remontée en priorité par rapport aux autres informations moins critiques.

L'utilisation comme nous l'avons faite précédemment du seul nombre de sauts vers le puits pour choisir le noeud relais peut affecter les performances du routage de données. En effet, le lecteur offrant le plus petit nombre de sauts n'est pas nécessairement le plus disponible en terme de charge ou le plus apte en terme d'activité radio à transmettre l'information. Surtout que dans le cas des déploiements denses de lecteurs considérés plusieurs lecteurs peuvent en même temps offrir un nombre de sauts équivalent et le choix d'un relais ou d'un autre devrait pouvoir être fait sur la base d'autres paramètres supplémentaires. A cet effet, nous proposons ici d'utiliser des techniques de **logique floue** permettant de combiner des métriques secondaires pour aider au choix du noeud relais et ainsi proposer un chemin alternatif pour les messages critiques. Le choix de cette technique est justifiée par sa simplicité qui nécessite donc très peu de puissance de calcul. Les lecteurs RFID pourront donc à la volée traiter les données reçues afin d'effectuer les étapes nécessaires au choix du noeud relais.

4.4.1 Positionnement

L'organisation en gradient ainsi que la technique de choix du slot décrite dans DACAR est maintenue ici. Les lecteurs sont donc toujours organisés selon leur rang relatif au puits et le choix du slot déterminé par la parité du rang. Afin de maintenir la topologie, les lecteurs continuent de s'échanger des messages régulièrement. Cependant, en plus du rang $rank_i$ et de l'identifiant id_i, les lecteurs envoient également 3 nouvelles valeurs :

— $load_i$: la charge du lecteur. Cette métrique définit la quantité d'informations de tags que le lecteur i a en mémoire et doit relayer vers le puits ;

— $radio_i$: l'activité radio dans le voisinage du lecteur. Elle renseigne sur la quantité de messages en échange autour du lecteur i (messages diffusés et informations de tags) et est relative à la densité de lecteurs déployés. Plus cette valeur sera élevé, plus la qualité du médium dans le voisinage du

lecteur sera affectée ;

— $backoff_i$: le nombre de tentatives consécutives d'accès au canal pour un lecteur avant l'envoi avec succès des données. A chaque fois qu'un lecteur doit relayer ses données vers son $parent_i$, il effectue un temps d'attente aléatoire avant d'écouter l'accès au canal. Si le canal est occupé, le lecteur tire un nouveau temps aléatoire et attend à nouveau dans la limite du temps imparti. Cette métrique renseigne sur le taux d'occupation du canal dans le voisinage d'un lecteur.

Pour aider au choix d'un noeud relais, il est nécessaire de pouvoir combiner ces 3 nouvelles métriques en plus du rang afin d'effectuer un choix adéquat. A cet effet, nous proposons d'utiliser la logique floue pour en déduire une métrique équivalente. La logique floue [91] vise à développer un cadre conceptuel pour des systèmes trop complexes ou mal définis pour produire une analyse quantitative précise. Elle permet la traduction et la combinaison de plusieurs éléments complexes en entrée en ensembles flous plus accessibles afin de produire une variable de sortie plus simple mais efficace. Un système de logique floue est construit en trois étapes :

Simplification des données ou Fuzzification

Cette étape convertit les données en entrées en ensembles flous en utilisant des fonctions d'appartenance (triangulaire, singleton, cloche, etc.). Dans notre cas, les lecteurs convertiront les valeurs de chaque métrique :

$$m = \{load_{RX}; radio_{RX}; backoff_{RX}\}$$

reçues des voisins en 3 ensembles flous selon les instructions suivantes :

— si $m < low_m$ la métrique est traduite comme Excellent ;

— si $low_m \leq m < up_m$ la métrique est traduite comme Bonne ;

— si $m \geq up_m$ la métrique est traduite comme Mauvaise.

low_m et up_m sont deux seuils définis pour chaque métrique considérée m (charge, radio et backoff) sur la base des observations effectuées ou du comportement attendu.

Ensembles flous			Sortie
Charge	Radio	Backoff	Métrique
Excellent	Excellent	Excellent	1
Excellent	Bon/Mauvais	Bon/Mauvais	2
Bon	Excellent	Excellent	2
Bon	Bon/Mauvais	Bon/Mauvais	3
Mauvais	Excellent	Excellent	4
Mauvais	Bon/Mauvais	Bon/Mauvais	5

Tableau 4.1 – Définition des règles floues

Définition des règles floues

Les règles floues combinent différents ensembles flous en utilisant des énoncés de type "**SI ... ALORS ... SINON**". Ces règles déterminent la relation entre les données en entrée et la variable de sortie désirée. En fonction des ensembles flous obtenus par la fuzzification, nous proposons les règles établies dans le tableau 4.1. L'objectif étant pour nous de réduire la charge des noeuds dans le réseau, nous avons défini cette dernière comme l'élément déterminant.

Obtention de la variable de sortie ou Defuzzification

C'est la dernière étape du processus durant laquelle la variable de sortie est convertie en une métrique utilisable. Dans notre cas, les valeurs $\{1;2;3;4;5\}$ du tableau 4.1 sont utilisées pour choisir le noeud relais en cas de données critiques à transmettre. Plus la métrique sera élevée, moins le noeud en question sera considéré comme relais potentiel.

La figure 4.9 reprend les différentes étapes menant à la définition des valeurs de métrique.

4.4.2 Fonctionnement

Après lecture des tags à portée, les lecteurs déterminent le niveau de criticité de l'information reçue. Si l'information est classée comme standard, non prioritaire, le lecteur suit la procédure de DACAR décrite plus-haut dans la Section 4.3.2 et envoie les données à son $parent_i$. Sinon, si les données sont dé-

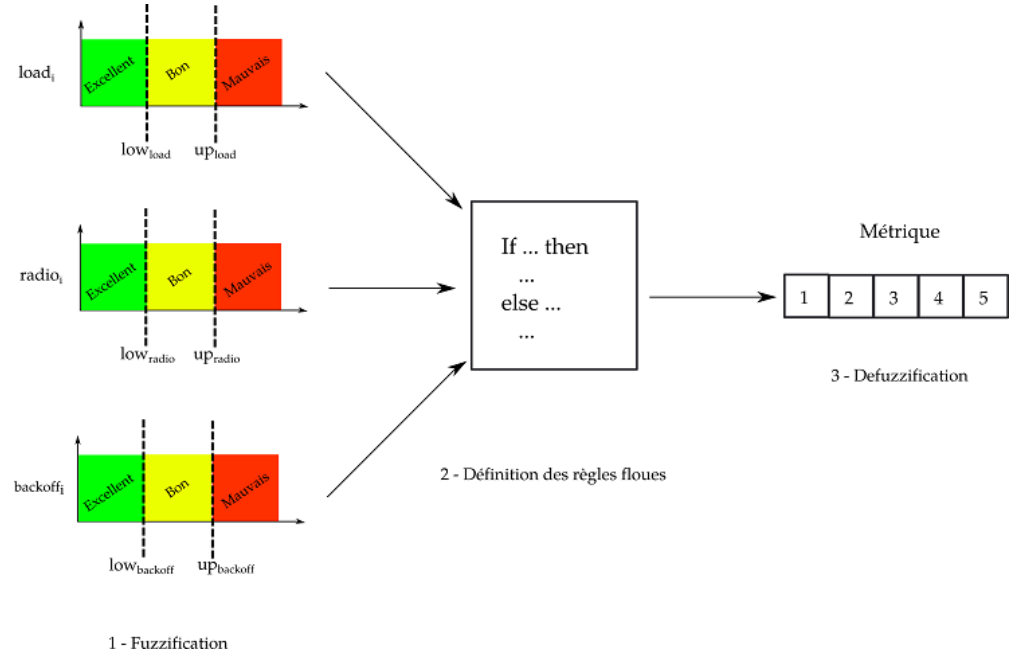


FIGURE 4.9 – Etapes de la logique floue

terminées comme prioritaires, le lecteur enverra les données vers le noeud relais $parentprio_i$ désigné par la combinaison des métriques issue de la logique floue (voir Figure 4.10). Dans cette nouvelle version chaque lecteur garde donc deux parents $parent_i$, pour les données standards, et $parentprio_i$, pour les données prioritaires à l'instar de la première version de DACAR qui n'utilisait qu'un seul $parent_i$ pour toutes les données indifféremment.

Envoi avec $parent_i$

Dans ce cas, la donnée est normale et peut être envoyée en se basant sur le nombre de sauts. Sur la base des messages reçus des voisins, le lecteur envoie ses données au noeud $parent_i$. En fonction de la densité de déploiement, les lecteurs peuvent avoir plusieurs voisins relais avec le même nombre de sauts. Néanmoins, seul l'un deux sera choisi comme $parent_i$.

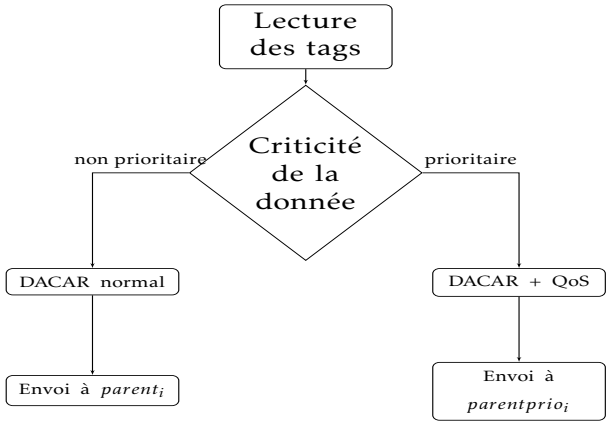


FIGURE 4.10 – Définition du noeud relais en fonction des tags lus

Envoi avec $parentprio_i$

Si la donnée est déterminée comme prioritaire, le lecteur utilise alors son $parentprio_i$ fixé par la combinaison des données issues de ses voisins. Sur la base des messages de maintenance de la topologie reçus des lecteurs voisins, le lecteur grâce aux règles de logique floue, détermine la variable de sortie correspondant à chacun de ses voisins à portée (voir Algorithme 7). Le voisin avec la métrique la plus faible sera donc choisi pour relayer les données prioritaires vers le puits (Algo. 7 lignes 5-8). A la réception d'un paquet contenant des informations prioritaires, le noeud relais accomplit lui aussi le même procédé pour définir son $parentprio_j$ et relayer les données.

Algorithm 7 Choix du $parentprio_i$ pour le lecteur R_i

1: $metric = \infty$
2: **for** tous les lecteurs R_j voisins de R_i **do**
3: **if** $rank_i > rank_j$ **then**
4: Estimation de la métrique suivant la logique floue
5: **if** $output < metric$ **then** ▹ Voisin avec métrique inférieure
6: $m \leftarrow output$
7: $parentprio_i = R_j$
8: **end if**
9: **end if**
10: **end for**

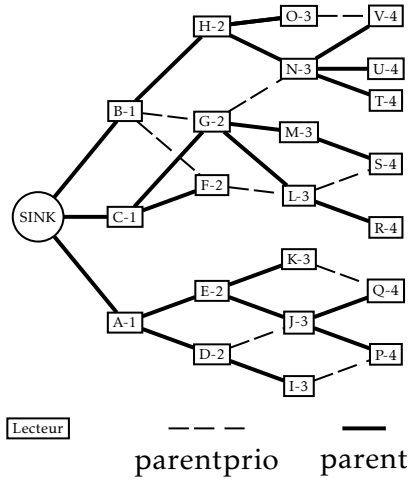


FIGURE 4.11 – Illustration DACAR + QoS

La Figure 4.11 représente une topologie de 22 lecteurs arrangés autour d'un puits. Le réseau est représenté sous la forme d'un graphe où les lecteurs et le puits sont les sommets et la nature du voisinage les liant sont les arêtes. Ainsi, si le trait est entier, le voisin est le $parent_i$ sinon, si le trait est en tirets, le voisin est le $parentprio_i$. On remarque que pour certains lecteurs le $parent_i$ et le $parentprio_i$ sont un même noeud ($parent_i == parentprio_i$) notamment pour les lecteurs autour du puits, c'est le cas des lecteurs $\{A;B;C\}$. En effet, ces noeuds n'ont d'autre alternative que d'envoyer les données au puits. C'est également le cas de certains autres lecteurs comme $\{D;E;H;I;K;M;O;R;T;U\}$, ces derniers sont peut être dans un voisinage où le seul lecteur à portée sera toujours le noeud relais, ce cas de figure est très fréquent dans des déploiements épais. Les autres lecteurs possèdent tous deux alternatives en fonction du niveau de criticité de la donnée à relayer. Par exemple, les lecteur F et G avec $parent_{F;G} = C$ pourront, en fonction des infirmations sur la charge, la radio et le backoff, choisir $parentprio_{F;G} = B$ pour relayer leurs données prioritaires.

4.4.3 Résultats

Pour le besoin de nos simulations, les valeurs de low_m et up_m pour chacune des métriques (charge, radio et backoff) sont fixées comme renseignées dans

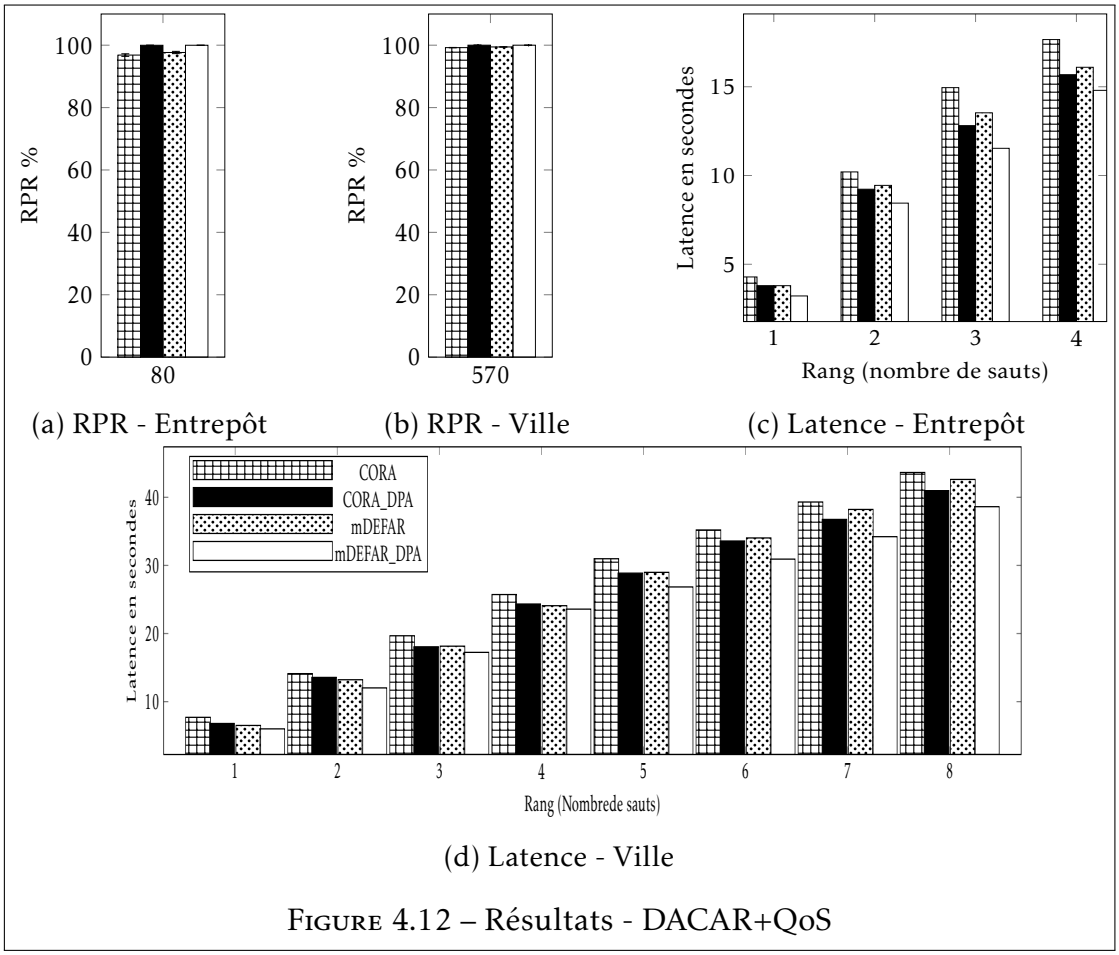
Métrique	low_m	up_m
Charge	3	6
Radio	10	20
Backoff	20	60

Tableau 4.2 – Définition des seuils low_m et up_m

le tableau 4.2. Ces valeurs ont été obtenu expérimentalement après plusieurs simulations qui ont démontré qu'elles permettaient d'obtenir les meilleurs résultats. Les Figures 4.12a et 4.12b montrent respectivement les résultats du ratio de paquets reçus avec succès dans les scénarios d'entrepôt et urbain definis en Section 3.3.2. Dans l'entrepôt (voir la Figure 4.12a), nous pouvons voir que les résultats sont supérieurs à 96% avec notre algorithme. En ce qui concerne les paquets standard CORA enregistre des performances légèrement inférieures (96,83%) en raison du nombre plus élevé de messages créés par rapport à mDEFAR (97,66%) expliqué par son débit de lecture plus élevé. Cela entraîne donc une perte de paquets. Les paquets prioritaires sont cependant livrés avec un taux proche de 100% aussi bien pour CORA (99,97%) que mDEFAR (99,98%). Dans le scénario urbain (voir la Figure 4.12b), les résultats sont meilleurs car les puits et les lecteurs sont plus disponibles pour les nœuds afin de transmettre leurs données. Quel que soit l'algorithme d'anti-collision utilisé, 100% des paquets prioritaires sont reçus avec succès aux niveaux des différent puits.

Les Figures 4.12c et 4.12d montrent respectivement les résultats de la latence relativement au rang du lecteur émetteur dans les scénarios d'entrepôt et urbain. Dans l'entrepôt (voir Figure 4.12c), nous pouvons voir qu'en utilisant nos métriques de logique floue, nous pouvons gagner environ 9 à 13% de latence sur chaque rang par rapport à la remontée standard (entre 300ms et jusqu'à 2 secondes). Le gain est plus faible aux rangs inférieurs car il y a moins de nœuds de relais potentiels et $parent_i = parentprio_i$ pour certains lecteurs (voir Figure 4.11). La même observation est faite dans le scénario urbain (voir Figure 4.12d) où le gain est de l'ordre de 8 à 12% par rapport à la remontée standard (entre 600ms et jusqu'à 5,4 secondes).

Nous ne comparons malheureusement pas nos algorithmes avec d'autres car aucune alternative n'a pu être trouvée dans la littérature.



(a) RPR - Entrepôt (b) RPR - Ville (c) Latence - Entrepôt

(d) Latence - Ville

FIGURE 4.12 – Résultats - DACAR+QoS

4.5 Conclusion

Les propositions faites ici permettent d'envisager des solutions IoT complètes basées sur la RFID pour la surveillance d'infrastructures urbaines. Les solutions permettront d'éviter au mieux les collision pour maximiser le débit de lecture, l'équité et le délai de couverture du système, mais également de remonter les données de manière rapide et fiable vers une ou plusieurs stations de base pour traitement. Nos solutions de remontée de données pourrons être adaptées au protocole d'anti-collision défini en fonction des contraintes de déploiement et de mobilité des lecteurs.

Les résultats obtenus en terme de fiabilité sont très satisfaisants. Quand bien

même ils n'atteignent pas 100% dans tous les cas, dans la mesure où les lecteurs sont en densité de déploiement, les données seront dupliquées par les lectures multiples et parviendront quand même à destination. La faible latence permet aussi d'assurer une très bonne réactivité en cas de problème détecté. La mise en place d'une solution traitant de manière différenciée les données en fonction de leur niveau de criticité améliore encore plus les performances attendues du système RFID.

Néanmoins, il serait intéressant de voir l'impact de la définition de règles floues différentes sur les performances de notre proposition. Aussi la considération de métriques telles que le niveau d'énergie des lecteurs peut être envisagée pour essayer de maintenir l'autonomie d'un système le plus longtemps possible.

Publication

Conférences

Abdoul Aziz Mbacké, Nathalie Mitton et Herve Rivano. "Using Fuzzy Logic for data priority aware collection in RFID sensing wireless networks" dans *PIMRC 2018 - IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, Sep 2018, Bologne, Italie, 2018*

Abdoul Aziz Mbacké, Nathalie Mitton et Herve Rivano. "DACAR : Distributed & Adaptable Crosslayer Anticollision and Routing protocol for RFID" dans *AdHoc-Now 2018 - 17th International Conference on Ad Hoc Networks and Wireless, Sep 2018, St Malo, France, 2018*

Abdoul Aziz Mbacké, Nathalie Mitton et Herve Rivano. "Une vision plus durable et fiable de la ville intelligente grâce à la RFID" dans *CORES 2018 - Rencontres Francophones sur la Conception de Protocoles, l'Évaluation de Performance et l'Expérimentation des Réseaux de Communication, Mai 2018, Roscoff, France, 2018*

Abdoul Aziz Mbacké, Nathalie Mitton et Herve Rivano. "Data Gathering Solutions for Dense RFID Deployments" dans *UIC 2017 - 14th International Conference on Ubiquitous Intelligence and Computing, Aou 2017, San Francisco, Etats-Unis, 2017*

Chapitre 5

Conclusion générale

L'objectif de cette thèse a été de pouvoir proposer des solutions complètes de surveillance des infrastructures urbaines à base de RFID passive avec collecte des informations via un réseau sans-fil multi-sauts réutilisant les lecteurs déployés. Les solutions proposées devaient pouvoir faire face non seulement aux contraintes environnementales de déploiement mais également aux besoins et exigences des applications considérées. Cela nous a poussé à nous intéresser à la RFID, à son mode de fonctionnement et les problématiques attenantes à la technologie notamment celles des collisions de lecteurs RFID. Par la suite, nous avons cherché à voir dans quelle mesure il était possible de réutiliser les mêmes équipements de la RFID pour non seulement lire les données environnementales mais également les collecter et les remonter vers un centre de décision de façon multi-sauts avec une gestion de la priorisation.

Nous avons donc pu établir une classification des différentes solutions d'anti-collision RFID et nous rendre compte du manque de solution adaptée aux déploiements denses et mobiles. Nous avons proposé des solutions adaptées à ces contraintes avec des résultats permettant une couverture rapide de l'ensemble des tags déployés tout en maintenant une bonne équité dans l'accès au canal entre les lecteurs et un faible taux de collisions par rapport aux algorithmes de la littérature. Pour la remontée de données, deux versions d'un algorithmes ont été proposées pour organiser la collecte des données issues des tags et leur renvoi vers une ou plusieurs stations de base en maintenant une fiabilité quasi

maximale et une latence adaptée aux applications destinées.

5.1 Conclusion sur les contributions de la thèse

La première phase de la thèse a été dédiée à l'étude des collisions de lectures RFID et de leur impact sur l'efficacité de la technologie. Nous avons donc pu identifier les différents types de collisions rencontrées et nous focaliser sur celles enregistrées lors de l'interrogation simultanée de 2 ou plusieurs lecteurs. Cette étude a permis de mettre en évidence le besoin évident de la mise en place d'une solution d'anti-collision afin d'améliorer les performances. Sur la base de ce constat nous avons parcouru l'état de l'art concernant cette problématique qui nous a révélé l'existence d'un grand nombre de solutions différentes aussi bien de par les mécanismes mis en place que par les objectifs finaux ciblés. De cette analyse nous avons pu déduire une classification en fonction des performances observées et des exigences applicatives qu'elles permettaient de résoudre. Cette analyse a également servi à l'affirmation de l'hétérogénéité des solutions RFID et la nécessité de guider le choix d'un protocole d'anti-collision ou d'un autre en fonction des objectifs attendus mais également des contraintes de déploiements des lecteurs RFID.

Sur la base de ce constat, nous avons entamé la deuxième phase, nous avons donc proposé deux solutions d'anti-collision de lecture pour les déploiements denses et mobiles de lecteurs RFID. Nos propositions baptisées DEFAR (et son dérivatif mDEFAR) et CORA ont été basées principalement sur les manquements observés lors de notre étude de la littérature sur la problématique. Ainsi, pour DEFAR, l'objectif a été de pouvoir récupérer au moins un lecteur lors de contention et maximiser l'équité et l'efficacité des lecteurs RFID en introduisant une gestion des priorités de lecture. D'un autre côté, CORA propose d'améliorer le débit de lecture quitte à avoir des collisions ce qui permet d'obtenir un meilleur délai de couverture totale des tags. De cette manière les lectures se font avec la prise en compte des voisins susceptibles de couvrir les zones en collision. Nos deux propositions ont montré des résultats convaincants et fiables pour le débit de lecture, les collisions, l'équité d'accès au canal et le délai de couverture pour les applications considérées malgré des simulations contraignantes en terme de

densité de déploiement de lecteurs RFID et mobilité.

Une fois les données issues des tags lues par les lecteurs, leur collecte et acheminement vers une ou plusieurs stations de base a été l'objectif de la troisième phase. Le but recherché était de pouvoir proposer une solution basée sur l'activité des lecteurs RFID afin de ne pas être trop demandant en énergie mais il était également de prendre en compte la nature et la quantité d'informations issues des tags. Notre revue de l'état de l'art a montré que malheureusement il n'existait pas à notre connaissance de solutions répondant aux objectifs visés tout en tenant compte des contraintes exprimées. De ce constat naquît DACAR qui propose d'utiliser la disposition des lecteurs comme topologie afin de les organise et de profiter des phases d'inactivité des lecteurs pour remonter les données vers la station de base. La première version de DACAR permettait aux lecteurs d'envoyer les données à un lecteur relais en fonction du nombre de sauts nécessaires pour atteindre la station de base. Toutefois, cette dernière ignorait les possibles données critiques qu'il est nécessaire de remonter au plus vite. Par ailleurs, le seul nombre de sauts ne reflète pas l'état du lien entre les deux lecteurs, ni entre le lecteur relais et la station de base. A cet effet, nous avons proposé une deuxième version, DACAR + QoS, qui permet aux lecteurs d'avoir plusieurs relais en fonction de la nature des données à transmettre. Ainsi, si les données à relayer sont prioritaires, le lecteur choisit un relais en fonction d'autres paramètres comme la charge du nœud relais, l'activité radio ou les échecs d'accès radio combinés grâce à des techniques de logique floues. Sinon, le lecteur utilise le nombre de sauts. Ces techniques ont permis d'améliorer les résultats en terme de perte de paquets lors de la remontée des données ainsi que de les acheminer avec une latence plus faible.

5.2 Travaux futurs

Malgré l'étendue des travaux menés, il reste nécessaire de prendre en compte d'autres éléments qui permettront non seulement d'améliorer les performances de nos propositions mais également de les rendre encore plus adaptées aux environnements de déploiement.

5.2.1 Modèle de propagation

Dans le but de modéliser nos algorithmes et simulations, nous avons utilisé un modèle de propagation isotrope représenté par des cercles concentriques autour du lecteur pour les plages de communication et de lecture. Cette hypothèse est faite dans toute la littérature, y compris les algorithmes présentés en Section 2.3, à des fins de simplification.

Ainsi, l'étude des environnements de déploiement prévus permet d'anticiper l'impact sur le canal de communication dédié, ainsi que son impact sur l'interrogation des tags. Rapporté aux scénarios considérés dans nos simulations, pour l'entrepôt où les tags peuvent être attachés à différents types de marchandises qui sont stockées de manière aléatoire, cette étude peut permettre de prévoir les effets de la réflexion et de la réfraction des signaux radio. Parallèlement, la prise en compte des différents types d'antennes RFID à faisceaux multiples trouvées sur le marché pourrait nous aider à mieux percevoir la performance de nos propositions dans un déploiement réel. Aussi, étant donné que l'interface de communication dédiée est différente de celle d'interrogation RFID, deux lecteurs en portée de communication pourraient ne pas réellement être en collision de lecture, résultant en des lecteurs inutilement désactivés, ou inversement, deux lecteurs pourraient être en collision mais ne pas se détecter sur le canal de communication dédié.

Par exemple, la Figure 5.1 montre un exemple avec 3 lecteurs A, B et C et deux tags T1 et T2. On peut y voir que le tag T2 est sous zone de collisions des lecteurs A et C cependant, la propagation de la portée de communication ne permet pas aux lecteurs en question de se détecter mutuellement.

5.2.2 Redondance et Compression des données

Bien que nous ayons décrit la redondance des informations de tags lus par plusieurs lecteurs comme un avantage, cela a un impact sur la consommation d'énergie des lecteurs et génère également une congestion inutile dans le réseau. Ceci pourrait être corrigé à 2 différents niveaux : interrogation des tags et/ou remontée des donnés. Éviter la redondance lors de l'interrogation permet d'empêcher les lecteurs voisins de lire les tags qui ont déjà été interrogées par d'autres

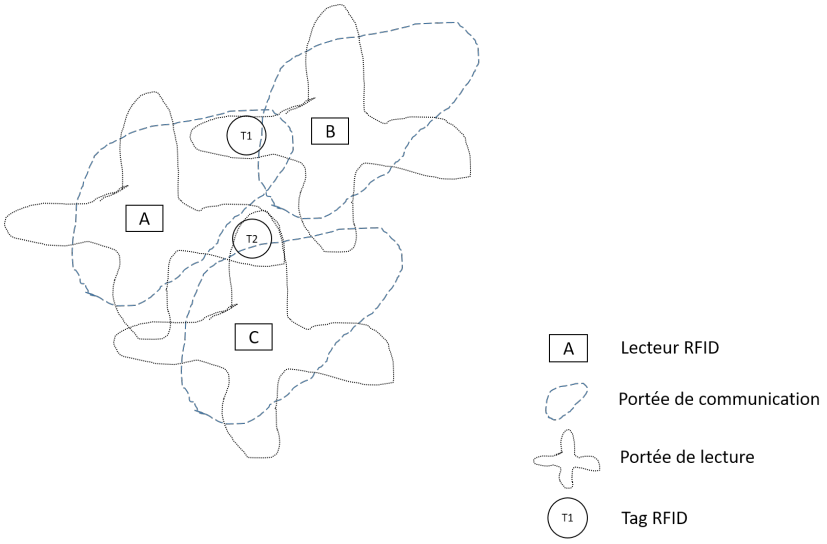


FIGURE 5.1 – Considération du modèle de propagation

voisins. Des travaux menés [58] proposent de tirer profit des collisions de tags afin d'éviter la redondance lors de l'interrogation. Pour éviter la redondance lors de la remontée des données, les lecteurs qui collectent des données à partir des dispositifs de rang supérieurs doivent s'assurer qu'ils ne transmettent pas les mêmes informations de tags et suppriment les informations en double. Cela a été discuté dans la littérature pour les réseaux de capteurs traditionnels et pourrait également être appliqué à la RFID. Une étude a d'ailleurs permis de valider l'efficacité énergétique de la compression ainsi obtenue des données collectées [70]. Dans ces travaux, les auteurs prouvent que dans plusieurs applications, la compression offre une meilleure efficacité énergétique que l'envoi de toutes les informations collectées. Une proposition consiste à compresser les données récupérées sur les nœuds de réseaux de capteurs en utilisant leur corrélation spatiale [57]. Alors que ce travail est fait pour les réseaux de capteurs traditionnels, il pourrait être appliqué à RFID et mis en œuvre avec la corrélation spatiale de la distribution des tags et les données qu'ils renvoient.

5.2.3 Consommation d'énergie

Au vu de la performance de nos propositions d'anti-collision en termes d'efficacité et de délai de couverture, il serait intéressant de mesurer l'énergie consommée pour leur bon fonctionnement. Bien que nous considérions des tags passifs qui n'ont pas besoin d'une batterie pour fonctionner, les lecteurs auront besoin d'être connectés à une source d'énergie pour lire les tags mais également coordonner et programmer leurs opérations pour éviter les collisions. Ainsi, dans DEFAR ou mDEFAR, les diffusions de messages servant à la phase de contention devraient être étudiées pour s'assurer que les lecteurs ne perdent pas trop d'énergie dans cette phase. En outre, il faudrait vérifier que l'énergie consommée par ces diffusions de messages est moins importante qu'un système basé sur un mécanisme CSMA où les lecteurs écoutent pendant des périodes de temps variables avant de transmettre. L'incidence des collisions entre lecteurs devrait également être étudiée pour vérifier si le compromis fait par CORA, concernant les collisions acceptées afin d'améliorer le délai de couverture, est rentable. Ces études permettront d'améliorer encore mieux la consommation d'énergie, le délai de couverture et les collisions. Délimiter la consommation d'énergie pourrait également permettre de se concentrer sur les solutions de récupération d'énergie pour les lecteurs. A ce titre, une application telle que celle de la ville intelligente pourrait faire bénéficier les lecteurs de sources de récupérations d'énergie de type freinage régénérateur, piézoélectrique pour bénéficier des vibrations créées par les véhicules, solaires, etc. Ces études et analyses permettront d'avoir des systèmes RFID entièrement autonomes, ce qui est un objectif pour les applications IoT, l'efficacité énergétique devant être améliorée afin d'allonger la durée de vie du réseau.

5.3 Perspectives

L'utilisation de solutions d'anti-collision, de collecte et d'acheminement des données comme présentées ici permet d'envisager la création de plusieurs applications basées sur la technologie RFID. Certes, des techniques supplémentaires mériteraient d'être étudiées afin d'améliorer les performances des applications

envisagées. Par exemple, nous pouvons imaginer la mise en place d'une solution de gestion des ordures ménagères basée sur la RFID. En effet, sur la base d'estimations de la Banque Mondiale [38], la production de déchet annuelle est d'environ 1.3 milliard de tonnes correspondant à 1.2 kg/personne/jour et il est prévu que ces chiffres grimpent à 2.2 milliard de tonnes/an et 1.42 kg/personnes/jour. Une telle quantité de déchets a un réel impact économique avec un coût de gestion estimé à 175.8 milliard € avec une prévision de 320.9 milliard € d'ici 2025. Dans le cas particulier d'une ville telle que Dunkerque dans le nord de la France, peuplée de 92000 habitants, la quantité de déchets est estimée à 132 282 tonnes avec un coût de 17 300 000 € pour la collecte et 9 600 000 € pour le tri et le recyclage [80]. Néanmoins, il est constaté que les populations restent encore réfractaires ou mal informées quant au tri correct de leurs déchets. Cela entraîne des coûts supplémentaires lors du tri sélectif en usine de valorisation revenant à environ 60% du coût total de la gestion des déchets [68] et malgré tout, seuls 39% de ces déchets sont effectivement recyclés [77] à cause du mauvais tri consommateur.

La mise en place d'une solution RFID pourrait permettre d'améliorer l'efficacité du tri et de la gestion des déchets ménagers par : la réduction des refus en centre de tri et coûts associés (transports additionnels, baisse de productivité, etc.) et la réduction des risques pour les centres de tri (gestion des déchets à risques comme les bonbonnes de gaz ou produits toxiques). Une telle solution passerait par l'agencement de 3 étapes :

Identification des déchets En utilisant des tags RFID passifs, il serait possible d'identifier les différents produits et donc déchets attenants au niveau des bornes de collecte. Les caractéristiques techniques de la RFID permettant la communication dans des environnements assez contraignants permettrait d'identifier toute sorte de déchets métalliques ou en verre. Certains produits, déjà disponibles dans le commerce, sont équipés de tags RFID et pourraient faciliter le traitement et donc être une bonne introduction à la solution proposée (voir Figure 5.2).

Pré-triage Afin de lire les tags rattachés aux déchets, on pourrait envisager d'équiper les conteneurs de lecteurs RFID. Ainsi en fonction de la classe de

Figure 5.2 – Exemple d'un produit du commerce avec tag RFID

déchet présenté, le conteneur pourrait soit signaler visuellement à l'aide de LED le bon compartiment ou grâce à un bras motorisé relever le couvercle du bac de destination. Ainsi, un tri amélioré sera effectué à la source avec une gestion des produits dangereux pour éviter les mélanges chimiques pouvant provoquer des dégâts importants [35]. La Figure 5.3 montre un exemple avec un utilisateur se présentant avec un déchet de type verre et qui sera donc guidé par le lecteur RFID en place.

Infrastructure L'ensemble de ces conteneurs RFID pourraient ensuite être considérés comme un réseau remontant les données jusqu'au centre de décision ou au camion de collecte. Cela permettra de pouvoir optimiser le trajet de passage des bennes à ordures afin de les orienter directement vers les bacs remplis ou organiser des collectes en fonction des classes de déchets. Les bacs pouvant être séparés par de grandes distances, il serait intéressant d'envisager la mise en place de petits robots drones chargés de maintenir la connectivité voire être utilisés comme data mules. Un exemple de l'architecture envisagée est illustrée en Figure 5.4.

Les travaux menés dans cette thèse permettent déjà de mettre en place une première version d'un telle application grâce aux algorithmes d'anti-collision nécessaire pour la lecture des tags sur les déchets mais également la remontée des données issues des bacs à déchets. Toutefois, il est nécessaire de porter une

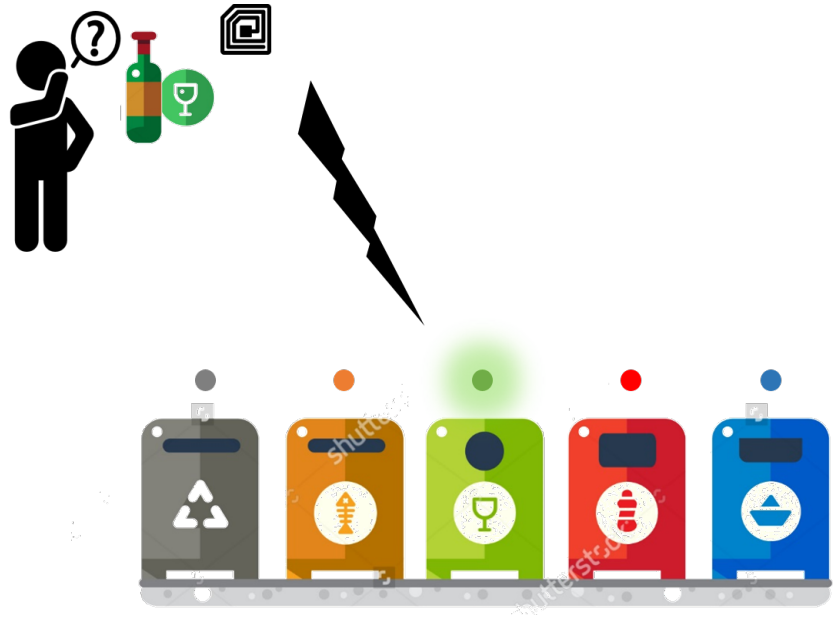


FIGURE 5.3 – Identification des déchets et instruction

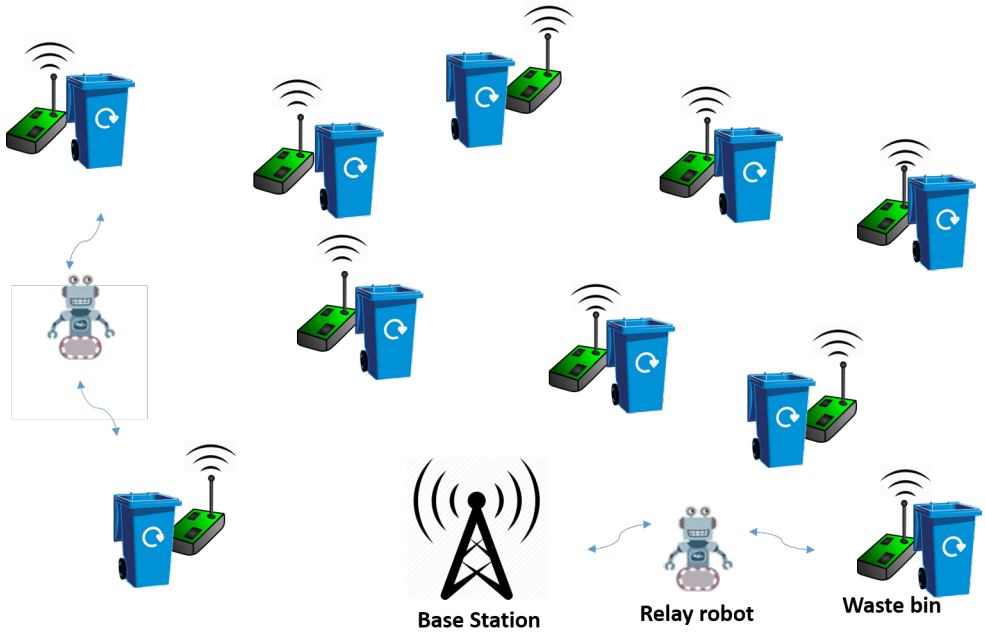


FIGURE 5.4 – Réseau de bacs à déchets

réflexion poussée sur d'autres aspects comme la **sécurisation des données**. En effet, des individus malveillants pourraient collecter les données issues des déchets stockés dans les bacs afin de les exploiter et produire des dégâts matériels. Dans le cas des applications de surveillance des infrastructures urbaines également, la sécurisation des données environnementales critiques est importante pour éviter une mauvaise utilisation de ces dernières. Des solutions soit basées sur des matériaux bloquants ou sur des techniques de cryptographie pourraient être envisagées afin de protéger les données d'accès tiers [74, 66].

Au vu de la grande quantité de données générées par les nombreux tags qui seront déployés, un autre aspect mérite une attention importante à savoir l'**apprentissage et la fouille de données dans le réseau et au niveau des lecteurs**. En effet avec l'ensemble des tags qui renvoient des informations au réseau de lecteurs déployés, il serait intéressant de pouvoir apporter des dispositifs de réaction immédiate en fonction des données collectées sans avoir besoin de les acheminer jusqu'au centre de décision. Ainsi les lecteurs pourraient bénéficier d'une intelligence collective, renforcée par leurs échanges pour améliorer le comportement et les réactions du système en cas de données critiques établies.

En combinaison avec les travaux futurs sur le modèle de propagation des signaux de lecture et de communication des lecteurs discutés en Section 5.2.1, l'**optimisation du déploiement et mobilité des lecteurs** est un axe de recherche qui nous semble pertinent. En effet, aussi bien dans le cadre du scénario de l'entrepôt, de la ville intelligente ou de la gestion de déchet, le déploiement des lecteurs devrait pouvoir se faire à des endroits stratégiques avec un nombre maîtrisé en fonction de leurs portée d'interrogation et de communication mais également de la quantité de tags prévue à l'endroit ciblé. Concernant la mobilité, le suivi de points d'intérêts identifiés à travers la zone de déploiement pourrait être une autre axe de recherche. Les lecteurs pourraient donc en fonction de leurs niveaux d'énergie et nombres de tags couverts, décider dans leur voisinage du lecteur le plus apte à poursuivre un (ou groupe de) tag(s) identifié(s) comme cible(s).

Bibliographie

[1] ETSI EN 302 208. *Radio Frequency Identification Equipment operating in the band 865 MHz to 868 MHz with power levels up to 2 W and in the band 915 MHz to 921 MHz with power levels up to 4 W; Harmonised Standard covering the essential requirements of article 3.2 of the Directive 2014/53/EU*. 2016.

[2] Ibrahim Amadou, Abdoul Aziz Mbacke et Nathalie Mitton. « How to improve CSMA-based MAC protocol for dense RFID reader-to-reader Networks ? » In : *Proceedings of ADHOC-NOW*. 2014.

[3] Ibrahim Amadou et Nathalie Mitton. « HAMAC : High Adaptive MAC Protocol for Dense RFID reader-to-reader Networks ». In : *Proceedings of AdHocNets*. 2015.

[4] Rafael V Aroca et al. « Application of Standard EPC/GEN2 UHF RFID Tags as Soil Moisture Sensors ». In : *Proceedings of Multidisciplinary Digital Publishing Institute*. T. 1. 2016.

[5] Youjing Bai et al. « An improved binary search RFID anti-collision algorithm ». In : *Proceedings of 12th International Conference on Computer Science and Education ICCSE*. IEEE. 2017, p. 435–439.

[6] Matthias Bartholmai et al. « Transmission characteristics of RFID sensor systems embedded in concrete ». In : *Proceedings of IEEE SENSORS*. IEEE. 2016.

[7] Matthias Bartholmai et al. « Two application examples of RFID sensor systems-identification and diagnosis of concrete components and monitoring of dangerous goods transports ». In : *ACTA IMEKO* 4 (2015).

[8] S. Birari et S. Iyer. « PULSE : A MAC Protocol for RFID Networks ». In : *Proceedings of Int. workshop on RFID and Ubiquitous Sensor Networks (USN)*. 2005.

[9] M. Bolic, D. Simplot-Ryl et I. Stojmenovic. « RFID Systems : Research Trends and Challenges ». In : John Wiley & Sons., 2010.

[10] M Victoria Bueno-Delgado et Pablo Pavon-Marino. « A centralized and aligned scheduler for passive RFID dense reader environments working under EPCglobal standard ». In : *Simulation Modelling Practice and Theory* 34 (2013).

[11] M Victoria Bueno-Delgado et Pablo Pavón-Mariño. « A maximum likelihood-based distributed protocol for passive RFID dense reader environments ». In : *The Journal of Supercomputing* 64 (2013).

[12] MV Bueno-Delgado et al. « A comparative study of RFID schedulers in dense reader environments ». In : *Proceedings of Int. Conf. on Industrial Technology (ICIT)*. IEEE. 2010.

[13] Victoria Bueno-Delgado et al. « A geometric distribution reader anti-collision protocol for RFID dense reader environments ». In : *IEEE T-ASE* (2013).

[14] Mathieu Cassel, Thomas Dépret et Hervé Piégay. « Assessment of a new solution for tracking pebbles in rivers based on active RFID ». In : *Earth Surface Processes and Landforms* (2017).

[15] L Catarinucci, R Colella et L Tarricone. « Enhanced UHF RFID sensor-tag ». In : *IEEE Microwave and Wireless Components Letters* 23 (2013).

[16] Belal Chowdhury et Morshed U Chowdhury. « RFID-based real-time smart waste management system ». In : *Proceedings of Australasian Telecommunication Networks and Applications Conference (ATNAC)*. IEEE. 2007.

[17] Li Da Xu, Wu He et Shancang Li. « Internet of things in industries : A survey ». In : *IEEE Transactions on industrial informatics* 10 (2014).

[18] Hong-Wei DENG, Hui WANG et Lang LI. « A New Hybrid Query Tree Anti-collision Algorithm ». In : *Artificial Intelligence* 10 (2016), 9789813206823_0082.

[19] K. Klair Dheeraj, Chin Kwan-Wu et Raad Raad. « A Survey and Tutorial of RFID Anti-Collision Protocols ». In : *IEEE Communications Survey & Tutorial* 12.3 (2010).

[20] United Nations DESA Population Division. *World urbanization Prospects*. Sous la dir. d'United Nations. 2018. url : `https://esa.un.org/unpd/wup/country-profiles/`.

[21] Litian Duan, Zizhong John Wang et Fu Duan. « Geometric Distribution-Based Readers Scheduling Optimization Algorithm Using Artificial Immune System ». In : *Sensors* 16.11 (2016).

[22] Guillaume Chelius et Elyes Ben Hamida. *WSNet - An event-driven simulator for large scale wireless sensor networks*. Sous la dir. d'INRIA CNRS. url : `http://wsnet.gforge.inria.fr/`.

[23] D. Engels et S. Sarma. « The Reader Collision Problem ». In : *Proceedings of IEEE Int. Conf. on Systems, Man and Cybernetics*. 2002.

[24] Jun-Bong Eom, Soon-Bin Yim et Tae-Jin Lee. « An efficient reader anticollision algorithm in dense RFID networks with mobile RFID readers ». In : *IEEE Transactions on Industrial Electronics* 56.7 (2009).

[25] *EPC TM radio-frequency identity protocols class-*1 *generation-*2 *UHF RFID protocol for communications at* 860 *MHz - 960 MHz version 1.2.0*. EPCglobal Standard specification, 2007.

[26] Szymon Fedor et Martin Collier. « On the problem of energy efficiency of multi-hop vs one-hop routing in wireless sensor networks ». In : *Proceedings of Int. Conf. on Advanced Information Networking and Applications Workshops*. IEEE. 2007.

[27] Renato Ferrero et al. « A fair and high throughput reader-to-reader anticollision protocol in dense RFID networks ». In : *IEEE Transactions on Industrial Informatics* 8.3 (2012).

[28] K. Finkenzeller. « RFID Handbook : Fundamentals and Applications in Contactless Smart Cards and Identification. » In : John Wiley & Sons, 2003.

[29] Klaus Finkenzeller. *Introduction*. Wiley Online Library, 2003.

[30] Filippo Gandino et al. « DCNS : An adaptable high throughput RFID reader-to-reader anticollision protocol ». In : *IEEE Transactions on Parallel and Distributed Systems* 24.5 (2013).

[31] Filippo Gandino et al. *Evaluation Criteria for Reader-to-Reader Anti-collision Protocols*. Rapp. tech. Politecnico di Torino, 2012.

[32] Filippo Gandino et al. « Probabilistic DCS : An RFID reader-to-reader anti-collision protocol ». In : *Journal of Network and Computer Applications* 34.3 (2011).

[33] David V Gibson, George Kozmetsky et Raymond W Smilor. *The technopolis phenomenon : Smart cities, fast systems, global networks*. Rowman & Littlefield, 1992.

[34] M. Golsorkhtabaramiri et al. « A Reader Anti-collision Protocol for RFID-Enhanced Wireless Sensor Networks ». In : *Wireless Personal Communications* (2015).

[35] Georges Guiochon. « On the Catastrophic Explosion of the AZF plant in Toulouse ». Global Congress on Process Safety. 2012.

[36] Essia Hamouda, Nathalie Mitton et David Simplot-Ryl. « Reader Anti-Collision in Dense RFID Networks With Mobile Tags ». In : *Proceedings of Int. Conf. on RFID-Technologies and App. RFID-TA*. 2011.

[37] MA Hannan et al. « A review on technologies and their usage in solid waste monitoring and management systems : Issues and challenges ». In : *Waste Management* 43 (2015), p. 509–523.

[38] Daniel Hoornweg et Perinaz Bhada-Tata. « What a waste : a global review of solid waste management ». In : *World Bank : Urban Development & Local Government* (2012).

[39] Yuxiao Hou et al. « PHY-Tree : Physical Layer Tree-Based RFID Identification ». In : *IEEE/ACM Transactions on Networking (TON)* 26.2 (2018), p. 711–723.

[40] Yu-Min Hu, Iang-Chong Chang et Jung-Shian Li. « Hybrid blocking algorithm for identification of overlapping staying tags between multiple neighboring readers in RFID systems ». In : *IEEE Sensors Journal* 15.7 (2015), p. 4076–4085.

[41] François Ingelrest, Nathalie Mitton et David Simplot-Ryl. « A Turnover based Adaptive HELLO Protocol for Mobile Ad Hoc and Sensor Networks ». In : *In Proceedings of IEEE MASCOTS*. 2007.

[42] S.E. Sarma J. Ho D.W. Engels. « HiQ : a hierarchical Q-learning algorithm to solve the reader collision problem ». In : *Proceedings of Int. Symposium on Applications and the Internet Workshops, SAINT*. USA, 2006.

[43] Kajal Jadhav et al. « Survey on Vehicle Tracking System ». In : *Imperial Journal of Interdisciplinary Research* (2016).

[44] Raj Jain, Arjan Durresi et Gojko Babic. « Throughput fairness index : An explanation ». In : (1999).

[45] Kyle Jamieson, Hari Balakrishnan et YC Tay. « Sift : A MAC protocol for event-driven wireless sensor networks ». In : *Wireless Sensor Networks*. Springer, 2006.

[46] Xiaolin Jia et al. « RFID technology and its applications in Internet of Things (IoT) ». In : *Proceedings of Int. Conf. on Consumer Electronics, Communications and Networks (CECNet)*. IEEE. 2012.

[47] Yi Jiang et al. « An efficient multi-channel reader collision avoidance protocol in RFID systems ». In : *Proceedings of Wireless Communications and Networking Conference (WCNC)*. IEEE. 2016.

[48] Daeho Kang et al. « ORGMA : Reliable opportunistic routing with gradient forwarding for MANETs ». In : *Elsevier Computer Networks* 131 (2018), p. 52–64.

[49] Kerem Kapucu, Jose Luis Merino Panades et Catherine Dehollain. « Design of a passive UHF RFID tag for capacitive sensor applications ». In : *Proceedings of 9th Conference on Ph. D. Research in Microelectronics and Electronics (PRIME)*. 2013.

[50] Sangkil Kim et al. « Ambient RF energy-harvesting technologies for self-sustainable standalone wireless sensor platforms ». In : *Proceedings of the IEEE* 102 (2014).

[51] Sukun Kim et al. « Wireless sensor networks for structural health monitoring ». In : *Proceedings of Int. Conf. on Embedded networked sensor systems*. ACM. 2006.

[52] Sung Hyun Kim et PooGyeon Park. « An efficient tree-based tag anti-collision protocol for RFID systems ». In : *IEEE Communications Letters* 11.5 (2007).

[53] Dheeraj K Klair, Kwan-Wu Chin et Raad Raad. « An investigation into thie energy eficiency of pure and slotted aloha based reid anti-collision protocols ». In : *Proceedings of IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks, WoWMoM*. IEEE. 2007, p. 1–4.

[54] RJ Kuo et Dwiyanti Yekti Nugroho. « A fuzzy multi-objective vehicle routing problem for perishable products using gradient evolution algorithm ». In : *Proceedings of 4th International Conference on Industrial Engineering and Applications (ICIEA)*. IEEE. 2017, p. 219–223.

[55] Yuan-Cheng Lai, Ling-Yen Hsiao et Bor-Shen Lin. « Optimal slot assignment for binary tracking tree protocol in RFID tag identification ». In : *IEEE/ACM Transactions on Networking* 23.1 (2015), p. 255–268.

[56] Jonathan Loo, Jaime Lloret Mauri et Jesus Hamilton Ortiz. *Mobile ad hoc networks : current status and future trends*. CRC Press, 2016.

[57] Chong Luo et al. « Compressive data gathering for large-scale wireless sensor networks ». In : *Proceedings of MOBICOM*. 2009.

[58] Martin Mayer, Norbert Görtz et Jelena Kaitovic. « RFID tag acquisition via compressed sensing ». In : *Proceedings of RFID-TA*. 2014.

[59] Abdoul Aziz Mbacke, Nathalie Mitton et Herve Rivano. « A survey of RFID readers anticollision protocols (under review) ». In : *IEEE RFID Journal* (2017).

[60] Abdoul Aziz Mbacke, Nathalie Mitton et Herve Rivano. « Distributed efficient & fair anticollision for RFID protocol ». In : *Proceedings of WiMob*. 2016.

[61] Abdoul Aziz Mbacke, Nathalie Mitton et Herve Rivano. « RFID anticollision in dense mobile environments ». In : *Proceedings of IEEE WCNC*. 2017.

[62] Abdoul Aziz Mbacke, Nathalie Mitton et Herve Rivano. « RFID Reader Anticollision Protocols for Dense and Mobile Deployments ». In : *MDPI Electronics Special Issue "RFID Systems and Applications"* (2016).

[63] Abdoul Aziz Mbacké, Nathalie Mitton et Hervé Rivano. « Data gathering solutions for dense RFID deployments ». In : *Proceedings of IEEE International Conference on Ubiquitous Intelligence and Computing (UIC)*. 2017.

[64] Abdoul Aziz Mbacké, Nathalie Mitton et Hervé Rivano. « RFID anticollision in dense mobile environments ». In : *Proceedings of IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE. 2017.

[65] United Nations. *By 2050, 68% of world population will live in urban areas - 2018 World Urbanization Prospects*. Sous la dir. d'United Nations. 2018. URL : https://www.youtube.com/watch?v=XN92srq5jwg.

[66] RK Pateriya et Sangeeta Sharma. « The evolution of RFID security and privacy : a research survey ». In : *Proceedings of Int. Conf. on Communication Systems and Network Technologies (CSNT)*. IEEE. 2011.

[67] Ying Qiu et al. « Multi-gradient routing protocol for wireless sensor networks ». In : *IEEE China Communications* 14.3 (2017), p. 118–129.

[68] Laurent Radisson. *Optimizing the collection of household waste involves simplifying the sorting process*. https://www.actu-environnement.com/ae/dossiers/collecte-dechets/simplification-geste-tri-jean-yves-menou-amf.php. 2014.

[69] Ramiro Ramirez, Eduardo Rojas-Nastrucci et Thomas Weller. « UHF RFID Tags for On/Off-Metal Applications Fabricated using Additive Manufacturing ». In : *IEEE Antennas and Wireless Propagation Letters* (2017).

[70] Mohammad Abdur Razzaque et Simon Dobson. « Energy-Efficient Sensing in Wireless Sensor Networks Using Compressed Sensing ». In : *MDPI Sensors* (2014).

[71] Hadiseh Rezaie et Medhi Golsorkhtabaramiri. « A fair reader collision avoidance protocol for RFID dense reader environments ». In : *Wireless Networks* (2017).

[72] Hadiseh Rezaie et Mehdi Golsorkhtabaramiri. « A fair reader collision avoidance protocol for RFID dense reader environments ». In : *Wireless Networks* (2017).

[73] Haidar Safa, Wassim El-Hajj et Christine Meguerditchian. « A distributed multi-channel reader anticollision algorithm for RFID environments ». In : *Journal of Computer Communications* 64 (2015).

[74] Dong-Her Shih, Chin-Yi Lin et Binshan Lin. « RFID tags : privacy and security aspects ». In : *International Journal of Mobile Communications* (2005).

[75] Dong-Her Shih et al. « Taxonomy and survey of RFID anti-collision protocols ». In : *Computer communications* 29.11 (2006).

[76] Haw–Yun Shin et Chung–Ming Own. « Implementing a green bicycle alliance using a wireless sensor network ». In : *International Journal of Ad Hoc and Ubiquitous Computing* (2014).

[77] Marie Simon. *Waste : what are the champion departments of recycling ?* `http://www.lefigaro.fr/economie/le-scan-eco/dessous-chiffres/2017/05/10/29006-20170510ARTFIG00005-dechets-quels-sont-les-departements-champions-du-recyclage.php`. 2017.

[78] Santar Pal Singh et SC Sharma. « A survey on cluster based routing protocols in wireless sensor networks ». In : *Procedia computer science* 45 (2015), p. 687–695.

[79] IEEE Computer Society. *IEEE Standard for Information Technology, IEEE 802.11 Standard*. 1999.

[80] Department of Urban Ecology of Dunkerque. *Summary of the annual report*. `https://www.communaute-urbaine-dunkerque.fr/fileadmin/documents/rapports/Synthese_dechets_2015.pdf`. 2015.

[81] Alexander Vaz et al. « Full passive UHF tag with a temperature sensor suitable for human body temperature monitoring ». In : *IEEE Transactions on Circuits and Systems II : Express Briefs* 57 (2010).

[82] James Waldrop, Daniel W Engels et Sanjay E Sarma. « Colorwave : an anticollision algorithm for the reader collision problem ». In : *Proceedings of ICC*. 2003.

[83] Roy Want. « Enabling ubiquitous sensing with RFID ». In : *IEEE Computer* 37 (2004).

[84] Haifeng Wu et al. « Binary tree slotted ALOHA for passive RFID tag anticollision ». In : *IEEE Transactions on Parallel and Distributed Systems* 24.1 (2013), p. 19–31.

[85] WWF. *Living Planet Report*. 2012.

[86] Minghua Xia, Qilin Yu et Zhonghua Li. « Relative density based anti-collision algorithm in RFID networks with dense readers ». In : *Proceedings of TENCON*. 2015.

[87] Ying Xu et Yifan Chen. « An improved dynamic framed slotted ALOHA anti-collision algorithm based on estimation method for RFID systems ». In : *Proceedings of IEEE International Conference on RFID*. IEEE. 2015, p. 1–8.

[88] Daniel J Yeager et al. « Wirelessly-charged UHF tags for sensor data collection ». In : *Proceedings of IEEE Int. Conf. on RFID*. IEEE. 2008.

[89] Tian Yuelong et al. « Achieving Collision-Free Communication by Time of Charge in WRSN ». In : *Mobile Networks and Applications* 21.3 (2016).

[90] Zhang YuJing et Cui Yinghua. « EDMC : An enhanced distributed multi-channel anti-collision algorithm for RFID reader system ». In : *Proceedings of American Institute of Physics Conf.* 2017.

[91] Lotfi A Zadeh. « Outline of a new approach to the analysis of complex systems and decision processes ». In : *IEEE TSMC* (1973).

[92] Bahareh Zaghari, Alex Weddell et Neil White. « Opportunities and challenges for energy harvesting sensor systems for harsh environments ». In : *Proceedings of the Fifth ACM International Workshop on Energy Harvesting and Energy-Neutral Sensing Systems*. ACM. 2017, p. 40–42.

[93] Ibon Zalbide et al. « Battery-free wireless sensors for industrial applications based on UHF RFID technology ». In : *Proceedings of IEEE SENSORS*. IEEE. 2014.

[94] Hong Zhang, Lei Han et Yu-Lu Li. « Design of hash-tree anti-collision algorithm ». In : *Proceedings of 3rd International Conference on Natural Computation ICNC*. T. 5. IEEE. 2007, p. 176–179.

[95] Jun Zhang et al. « A review of passive RFID tag antenna-based sensors and systems for structural health monitoring applications ». In : *MDPI Sensors* 17.2 (2017), p. 265.

[96] Lukas Zöscher et al. « Concept for a security aware automatic fare collection system using HF/UHF dual band RFID transponders ». In : *Proceedings of 45th European Solid State Device Research Conference (ESSDERC)*. IEEE. 2015, p. 194–197.

Table des matières

Résumé **xi**

Sommaire **xiii**

Liste des tableaux **xvii**

Table des figures **xix**

1 Introduction générale **1**

2 Anticollision de lecteurs RFID - Problématique et Etat de l'art **11**
2.1 Présentation de la problématique des collisions 11
2.1.1 Collisions de tags RFID 12
2.1.2 Collisions de lecteurs RFID 14
2.2 Présentation des métriques d'évaluation des protocoles d'anti-collision RFID 16
2.2.1 Débit de lecture 16
2.2.2 Collisions 17
2.2.3 Efficacité 17
2.2.4 Indice d'équité de Jain (IEJ) 18
2.2.5 Délai de couverture 18
2.3 Etat de l'art sur les protocoles d'anti-collision de lecteurs RFID 19
2.3.1 Catégorisation des protocoles 19
2.3.2 Protocoles TDMA 22
2.3.3 Protocoles CSMA 29
2.3.4 Évaluation de Performances des protocoles proposés . . 33
2.4 Conclusion 38

3 Anticollision de lecteurs RFID - Nos contributions **39**
3.1 Distributed Efficient & Fair Anti-collision for RFID (DEFAR) . 40
3.1.1 Positionnement 40

3.1.2 Fonctionnement 43
3.1.3 mobile-DEFAR (mDEFAR) 49
3.2 Coverage Oriented RFID Anti-collision (CORA) 49
3.2.1 Positionnement 50
3.2.2 Fonctionnement 52
3.3 Résultats obtenus 55
3.3.1 Déploiements statiques 57
3.3.2 Déploiements dynamiques 60
3.4 Conclusion 65

4 Remontée d'informations RFID 67
4.1 RFID en complément des réseaux de capteurs classiques pour la ville intelligente 68
4.1.1 Sensing tags 69
4.1.2 Exemples d'applications utilisant la RFID 71
4.2 Remontée de données RFID 72
4.3 Distributed & Adaptable Crosslayer Anti-collision and Routing (DACAR) 75
4.3.1 Positionnement 76
4.3.2 Fonctionnement 79
4.3.3 Résultats 84
4.3.4 Discussion 86
4.4 DACAR + Qualité de Service 86
4.4.1 Positionnement 87
4.4.2 Fonctionnement 89
4.4.3 Résultats 92
4.5 Conclusion 94

5 Conclusion générale 97
5.1 Conclusion sur les contributions de la thèse 98
5.2 Travaux futurs 99
5.2.1 Modèle de propagation 100
5.2.2 Redondance et Compression des données 100
5.2.3 Consommation d'énergie 102
5.3 Perspectives 102

Bibliographie 107

Table des matières 115

Collecte et remontée multi-sauts de données issues de lecteurs RFID pour la surveillance d'infrastructures urbaines

Résumé

La forte urbanisation dont le monde est témoin exige une meilleure gestion des villes. Cette gestion améliorée passe par la surveillance et la maintenance des équipements et infrastructures urbaines afin d'assurer plus de sécurité et bien-être aux habitants. Pour atteindre ces objectifs, un rôle clé a été confié aux technologies de l'information et de la communication (TIC) à travers les concepts de l'Internet des Objets (IoT) et des Villes Intelligentes. Nous avons donc observé un nombre grandissant de dispositifs de captation de données intégrés à différents éléments de la ville pour offrir des services et applications dédiés. Cependant, certaines contraintes matérielles, technologiques, économiques et environnementales relatives à la nature des dispositifs utilisés freinent encore les objectifs de couverture et de gestion totales des infrastructures urbaines. Cette thèse se positionne dans ce contexte et propose l'Identification par Radio Fréquence (RFID) en complément des techniques déjà utilisées afin de pallier ces manquements. Toutefois l'adoption de la RFID à grande échelle pour les centres urbains nécessite de résoudre deux principaux problèmes : les collisions de lecture et la collecte et remontée des données. A travers les travaux menés dans cette thèse, nous avons d'abord cherché à identifier les solutions déjà proposées dans la littérature pour réduire les collisions. Sur la base de cette étude, nous avons proposé deux algorithmes distribués d'anti-collision de lecture DEFAR et CORA. Ils permettent d'assurer un débit de lecture important tout en maintenant un taux de collisions et une latence de couverture faible comparés aux solutions de la littérature pour des déploiements denses et/ou mobiles de lecteurs RFID. Par la suite, nous avons proposé DACAR, un algorithme distribué de collecte des données issues des lecteurs RFID de manière multi-sauts. Ce protocole s'adapte en fonction du protocole d'anti-collision utilisé et de la position des lecteurs déployés pour fournir un ratio de délivrance des paquets fiable et un faible délai de bout-en-bout. Une version améliorée de DACAR est ultérieurement proposée qui prend en compte la priorisation des données pour offrir des chemins différents plus appropriés à l'aide d'une combinaison de différents paramètres grâce à la logique floue.

Mots clés : rfid, internet des objets, réseaux de capteurs sans-fil, villes intelligentes

---

Inria Lille - Nord Europe
Parc Scientifique de la Haute Borne – 40, Avenue du Halley – Bât A - Park Plaza – 59650 Villeneuve d'Ascq – France

Collection and multi-hop forwarding of RFID data for the monitoring of urban infrastructures

Abstract

The strong urbanization witnessed by the world requires better management of cities. This improved management involves the monitoring and maintenance of urban infrastructures and equipment to ensure greater safety and well-being for residents. To achieve these objectives, a key role has been given to Information and Communication Technologies (ICT) through the concepts of Internet of Things (IoT) and Smart Cities. As a result, we have seen a growing number of data capture devices integrated into different parts of the city to deliver dedicated services and applications. However, given material, technological, economic and environmental constraints related to the nature of the devices in use, the objectives of total coverage and management of urban infrastructures is still hindered. This thesis is positioned in this context and proposes Radio Frequency Identification (RFID) in addition to the techniques already in use to overcome these shortcomings. However, the adoption of large-scale RFID for urban centers requires solving two main issues: reading collisions and data collection and forwarding. Through the work carried out in this thesis, we first sought to identify the solutions already proposed in the literature to reduce collisions. Based on this study, we proposed two distributed anti-collision algorithms named DEFAR and CORA. They provide high throughput while maintaining low collision and latency rates compared to literature solutions for dense and/or mobile deployments of RFID readers. Subsequently, we proposed DACAR, a distributed algorithm for multi-hop data collection using RFID readers. This protocol is adaptable and according to the anti-collision protocol used and the position of deployed readers, provides a reliable packet delivery ratio and low end-to-end delay. An improved version of DACAR is later proposed that takes into account the prioritization of data to provide different, improved and more suitable paths using a combination of different parameters through fuzzy logic.

Keywords: rfid, internet of things, wireless sensor networks, smart cities