

MA-004-S14-1

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur de la Recherche Scientifique
Université Saad Dahleb Blida 1



Faculté des Sciences
Département d'Informatique

**Mémoire de fin d'études en vue d'obtention de
diplôme de Master en :**

Filière : Informatique

Option : Ingénierie de logiciel (IL)

Thème

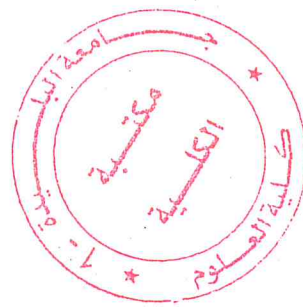
Déploiement d'une infrastructure RFID robotique cyber-physique et développement du système de contrôle

Présenté par :
BERKAT Lotfi
ABEB Abdelkrim

Soutenu le 01/07/2017

Mme Larousi
Mme Touahri

Dr HENTOUT Abdelfetah
Dr Chikhi Nacim



M. Président
M. Examineur
M. Examineur
Encadreur
Promoteur

Promotion
2016 / 2017

MA-004-514-1

Remerciement

Merci à Dieu de sa grâce, source de notre force et courage tout au long de nos études universitaires.

Nous tenons à exprimer nos vifs remerciements et notre profonde gratitude à Mr HENTOUT, de nous avoir encadré dans notre mémoire de fin d'étude et Mr TIBERKAK de nous avoir dirigé dans notre travail.

Nous remercions également Mr CHIKHI pour son aide précieuse et son encouragement qu'il n'a cessé de nous communiquer.

Nous remercions les membres du jury pour avoir accepté d'évaluer notre travail.

Un grand merci également à nos familles pour leur soutien aussi bien moral que financier et pour leurs sacrifices.

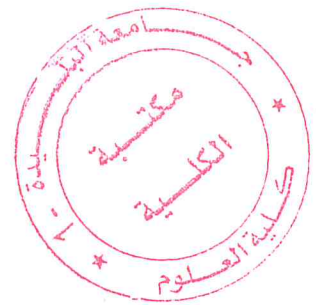
Nous ne pourrions terminer sans remercier tous ceux qui ont participé d'une manière ou d'une autre dans l'élaboration de ce projet de fin d'études.

Merci

Dédicaces

Je dédie ce modeste travail
A vous très cher mère et père, en témoignage de mon
frère Amine et ma sœur Amina
A toute ma famille
A mon encadreur Mr Hentout Abdelfetah
A mon promoteur Mr Chikhi Nacim Fateh
A tous mes collègues, plus particulièrement : Aymene, Badro,
Boualem, Mohamed

en témoignage De mon amitié sincère;
A tous mes amis, plus particulièrement : Moufida, Amina, Omar,
Elias, Fouzi, Dallel, Razika en
Témoignage de mon amitié sincère;
A tous ceux qui m'ont soutenu, qu'ils trouvent ici l'expression de
mon amour et ma profonde
Gratitude



MR L. BERKAT

Dédicaces

Je dédie ce modeste travail

A vous très cher mère et père, en témoignage de mes
frères

A toute ma famille

A mon encadreur Mr Hentout Abdelfetah

A mon promoteur Mr Chikhi Nacim Fateh

A tous mes collègues, plus particulièrement : Boualem, Mohamed,
Lotfi

Témoignage de mon amitié sincère;

A tous ceux qui m'ont soutenu, qu'ils trouvent ici l'expression de
mon amour et ma profonde
Gratitude

MR A. ABEB

Table des acronymes

SCP : Système cyber-physique.

SRCP : Système robotique cyber-physique.

RFID: Radio Frequency IDentification

NFC: Near Field Communication

Wi-Fi: Wireless Fidelity

RSSI: The Received Signal Strength Indicator

UML: Unified Modeling Language

CDTA : Centre de Développement des Technologies Avancées

ROS : Robot Operating System

DLL : Dynamic Link Library

UHF : Ultra haute fréquence

HF : haute fréquence

LF : Low frequenc

SHF : Super High Frequency

EPC : Electronic Product Code

TID : Tag identifier

CD : Compact Disk

TCP : Transmission Control Protocol

IP : Internet Protocol

Timestamp : L'horodatage décomptent à partir du 1er janvier 1970 à 0 h

NSF : national science foundation

Table des matières

Résumé

Abstract

ملخص

Introduction générale..... 1

Chapitre I

Etat de l'arts

1. Introduction.....	05
2. Système Robotique Cyber Physique.....	05
2.1 Définition des Systèmes Cyber Physique (SCP).....	05
2.2 Caractéristiques des systèmes cyber physique.....	05
3. Définition des systèmes robotiques cyber physiques (SRCP).....	06
3.1 Capacités de base d'un SRCP.....	06
3.2 Composants des SRCP.....	06
3.3 Technologies utilisés par les SRCP.....	06
3.3.1 Wifi.....	06
3.3.2 GPS.....	07
3.3.3 NFC.....	08
4 La technologie RFID.....	09
4.1 Définition.....	09
4.2 Fonctionnement des systèmes RFID.....	09
4.3 Composants des Systèmes RFID.....	10
4.3.1 Tag (étiquette).....	10
4.3.2 Lecteur RFID.....	11
4.3.3 Application de traitement des données.....	11
4.4 Caractéristiques d'un système RFID.....	12
4.5 La structure EPC (Electronic Product Code).....	13
4.6 Avantages du système RFID.....	13
4.7 Inconvénients.....	14
4.8 Domaines d'application des SRCP et de la technologie RFID.....	14
4.8.1 Le passeport biométrique RFID en Algérie.....	15
4.8.2 Des drones lecteurs RFID pour faire l'inventaire de stock.....	15

4.8.3 Surveillance Robotique Cyber Physique dans l'environnement.....	16
4.8.4 La RFID dans le secteur de la santé.....	17
4.8.5 La gestion de la chaîne d'approvisionnement.....,	17
5. Conclusion.....	18

Chapitre II

Description et prise en main du matériel IFM

1. Introduction.....	20
2. La société IFM	20
3. Les Tags RFID IFM	21
3.1 Types de mémoire des étiquettes RFID.....	21
3.1.1 Mémoire réservée.....	21
3.1.2 Mémoire TID.....	21
3.1.3 Mémoire EPC.....	21
3.1.4 Mémoire utilisateur.....	22
4. Le lecteur DTE800	22
4.1 Fournitures.....	22
4.2 Utilisation du Reader	23
4.3 Communication avec le Reader via la transmission Ethernet	23
4.4 Format de la trame	23
4.5 Port de communication	23
5. L'outil Wireshark	23
6. Fonctionnalités de base implémentées.....	24
6.1 Détail des commandes et leurs réponses	25
6.1.1 GetPortPower.....	25
6.1.2 GetExtResultFlag.....	26
6.1.3 SetExtResultFlag.....	27
6.1.4 Lecture des Tags.....	28
6.1.4.1 SyncGetEPCs	29
6.1.4.2 ASyncGetEPCs.....	31
6.1.5 ASyncStopCommand.....	33
6.1.6 SyncWriteDataSpecific	33
7. Conclusion	35

Chapitre III

Description de la solution proposée

1. Introduction.....	37
2. Localisation d'objets basée sur l'interaction omniprésente avec les étiquettes RFID.....	37
2.1 Interaction à base de phéromones.....	37
2.2 Estimation de déplacement à l'aide d'étiquettes RFID	38
2.3 Déploiement des phéromones avec tags RFID	39
2.3.1 Déploiement de phéromones	39
2.3.2 Processus d'évaporation progressive des phéromone.....	41
3. Déploiement de l'infrastructure RFID	42
4. Adaptation de la structure de Tag.....	43
5. Description de l'environnement de travail	44
6. Interaction entre les agents du système via Stigmergie	45
6.1 Propager phéromone	45
6.2 Optimisation du chemin	46
6.3 Accéder à un objet	48
7. Conception de l'outil de simulation.....	48
8. Conclusion	49

Chapitre IV

Teste et validation

1. Introduction.....	51
2 Environnement de développement	51
2.1 Langage de programmation choisi.....	51
2.2 IDE Eclipse.....	51
3. Présentation et validation de la librairie d'utilisation du lecteur DTE800.....	52
3.1 Connexion avec le Reader	52
3.1.1 Préparation du lecteur	52
3.1.2 Antennes	52
3.1.3.3 Port de communication	53
4. Interface graphique.....	53

4.1 Interface de communication.....	53
4.2. Interface de lecture	54
4.3 Ecriture dans les tags	56
4.4 Interface paramétrages.....	57
5. Validation de la librairie	58
6. Validation de l'algorithme de localisation.....	58
6.1 Simulation de l'environnement de travail.....	58
6.2 Simulation de déplacement d'objet.....	59
6.3 Elimination des boucles	60
6.4 Déplacement de plusieurs objets	61
6.5 Identification du meilleur chemin.....	61
7. Discussion des résultats obtenus.....	62
8. Conclusion.....	63
Conclusion générale et perspectives.....	64

Résumé

Les systèmes cyber-physiques sont considérés comme un domaine de recherche prioritaire. Ils occupent un rôle important dans de nombreux domaines, notamment la robotique. Ces systèmes sont capables de détecter le monde physique et de le reconfigurer en données informatiques. La détection, l'identification et la localisation d'objets dans un système robotique cyber-physique est une problématique importante ; ceci nécessite un ensemble d'approches pour effectuer cette tâche. Le but de notre travail est le déploiement d'une infrastructure cyber-physique utilisant la technologie RFID, de telle sorte que l'ensemble des lecteurs RFID, placés sur les robots, soient capables d'identifier des tags RFID situés dans l'environnement et localiser des tags RFID attachés à des objets en suivant des pistes de phéromones numériques déposées dans l'environnement.

Mot clés : Déploiement d'une infrastructure RFID, phéromone numérique, système robotique cyber-physique, RFID.

Abstract

Cyber-physical systems are an important research field especially in robotics. They systems are able to detect the physical world and to reconfigure it in computer data. The detection and identification in cyber-physical robotic system is a very important issue; this needs a set of approaches to perform this task. The goal of our work is to deploy an infrastructure using the RFID technology, in such a way that all the RFID readers, installed on the robots, be able to identify and localize the RFID tags of the environment attached to objects by tracking the deposited digital pheromone.

Keywords: Deployment of RFID infrastructure, digital pheromone, robotic cyber-physical system, RFID.

ملخص

الأنظمة السيبرانوفيزيائية تعتبر مجال أساسي للأبحاث، فهي تأخذ دورا أساسيا في عدة مجالات خصوصا مجال الروبوتات. هذه الأنظمة قادرة على اكتشاف العالم الفيزيائي وتحويله إلى معلومات إلكترونية. الاكتشاف والتعريف في النظام الروبوتي السيبرانوفيزيائي يعتبر مشكلة مهمة حيث يتطلب مجموعة من المناهج لتحقيق المطلوب. الهدف من عملنا هذا هو توزيع البنية التحتية باستعمال RFID حيث أن مجموعة من الأجهزة القارئة الموضوعية على الروبوتات تمكننا من تجميع وتعريف بطاقات RFID المثبتة على الأشياء المتواجدة في الوسط وتحديد مكانها باتباع مسارات الفرمونات الرقمية المتروكة.

كلمات مفتاحية: نشر البنية التحتية RFID، فرمون الرقمي، النظام الروبوتي السيبرانوفيزيائي، RFID.

Introduction générale

La robotique autonome devient de plus en plus importante dans le domaine de la recherche ; elle fait face à une croissance rapide dans la complexité des besoins et des exigences pour des robots chargés de tâches multiples, capables de se coordonner, et développés de telle manière que des garanties de sûreté et de sécurité puissent être vérifiées et certifiées pour remplacer l'homme [1]. En parallèle, une évolution similaire dans le domaine des systèmes temps-réel embarqués répartis a justifié l'émergence du domaine des « Systèmes cyber-physiques (SCP) » reflétant une montée similaire en complexité. À la différence des systèmes embarqués traditionnels, un SCP est généralement conçu comme un réseau d'éléments informatiques en interaction avec des entrées et des sorties physiques au lieu de dispositifs autonomes en interaction. Les SCP peuvent être trouvés dans divers domaines tel que l'aéronautique, la santé, l'automobile et les processus chimiques.

Ces systèmes sont capables de collecter des informations sur le monde physique et de les reconfigurer en données informatiques. « *En sciences informatiques, les données sont relativement limitées. En physique, à l'inverse, ce qui peut être observé est très divers* ». Un SCP doit donc être capable de prendre en compte le monde réel. Ces notions sont étroitement liées aux concepts des réseaux de capteurs et de la robotique [2]. Les systèmes robotiques cyber-physiques (SRCP) sont une classe importante de SCP. L'habilité des robots à interagir intelligemment avec le monde repose sur le calcul embarqué, la communication, le contrôle en temps réel et la perception du monde qui les entoure. Ceci est réalisé afin d'accomplir automatiquement des tâches souvent pénibles, répétitives, impossibles pour les humains ou dangereux (analyse de substances à l'intérieur d'un volcan, des centrales nucléaires, dans l'espace ou dans les profondeurs de l'océan) ; ou dans un but d'efficacité supérieure notamment dans la vie quotidienne (y compris le ménage domestique en assimilant les différents objets, soins des

malades, des personnes âgées) et la chirurgie médicale (avec des robots capables de réaliser des opérations, détecter et identifier les instruments médicaux, détecter des maladies).

En robotique, la capacité d'identifier et de localiser à tout moment des éléments fixes ou mobiles dans un environnement est une question importante. Il existe un ensemble de méthodes, de technologies, d'approches et de dispositifs qui permettent de répondre à cette question. On trouve des solutions basées sur l'exploitation de systèmes satellitaires GPS, GLONASS ou sur l'exploitation de réseaux terrestres de communications GSM/UMTS, LORAN, TNT ou encore sur l'exploitation de réseaux locaux WiFi, Bluetooth, Zigbee, Infra-rouge, ultrason, ULB, RFID.

Dans ce projet, nous nous intéressons à l'identification et la localisation d'objets dans une infrastructure RFID, plus précisément à l'interaction omniprésente basée sur les phéromones avec les étiquettes RFID. C'est une approche relativement récente de localisation basée sur la technologie RFID. Cette technologie utilise la radiofréquence pour identifier les objets étiquetés lorsqu'ils passent à proximité du champ d'un lecteur RFID. Dans ce contexte, notre travail est organisé en deux parties :

- La première partie est consacrée au développement des fonctions de base pour les lecteurs RFID (en particulier le lecteur *DTE800* de *IFM*) ; notamment la lecture et l'écriture sur les étiquettes RFID ainsi que des fonctions de paramétrage.
- La deuxième partie est consacrée aux aspects suivants :
 - Adaptation d'une approche existante à notre contexte qui est la localisation d'objets déposés par des robots dans une infrastructure RFID.
 - Proposition d'un algorithme qui permet d'optimiser des chemins en combinant les informations déposées dans l'environnement.
 - Validation de l'algorithme proposé à travers des scénarios en simulations.

Outre cette introduction générale, le présent document est organisé en quatre chapitres ; ils sont décrits comme suit :

- Dans le premier chapitre, nous allons présenter les SRCP sur lesquels est basé le déploiement de l'infrastructure RFID. Ensuite, nous allons présenter les différentes technologies et systèmes de détection et d'identification. Nous donnerons également des exemples d'applications de SCP dans divers domaines notamment dans le domaine de la robotique en utilisant la technologie d'identification RFID.

1. Introduction

L'intégration des robots dans les environnements humains combine les systèmes informatiques et le réseau avec les entités physiques ; ceci est appelé système robotique cyber-physique (SRCP). Actuellement, l'un des défis fondamentaux de la robotique dans le domaine de la gestion des entrepôts est la localisation des objets [3].

Dans ce chapitre, nous allons tenter de cerner le domaine des SRCP. Ensuite, nous allons détailler la technologie RFID, qui est l'une des technologies de détection utilisée par les SRCP. Enfin, nous citerons quelques domaines utilisant ce type de systèmes.

2. Système Robotique Cyber Physique

2.1 Définition des Systèmes Cyber-Physique (SCP)

D'après Lee & Seshia [4], le terme SCP a été inventé par Helen Gill en 2006 au National Science Foundation (NSF) aux États-Unis. Un système cyber-physique se réfère à plusieurs définitions selon le domaine d'utilisation. Il peut être défini comme étant l'intégration du calcul avec le processus physique (réseaux et ordinateurs embarqués) [6]. Dans [7], les auteurs définissent les SCP comme une technologie clé pour améliorer la prochaine génération d'applications intelligentes ; il s'agit de systèmes physiques surveillés/contrôlés par des lois sémantiques définies par l'utilisateur.

2.2 Caractéristiques des SCP

On peut dire que les applications des SCP ont contribué à la révolution informatique du XXe siècle. Les SCP se sont ainsi intégrés dans des systèmes et des dispositifs médicaux à haute confiance, dans le contrôle de circulations, de sécurité, dans la conservation d'énergie, dans les systèmes de défense, dans le contrôle de l'environnement, etc. Ils permettent [7] :

- Une distribution géographique étendue, où les composants peuvent être situés dans des endroits qui manquent de sécurité physique.
- Une utilisation à l'échelle mondiale.
- L'entrée et le retour d'informations via l'environnement physique.
- Un contrôle et une gestion distribués.

3. Définition des Systèmes Robotiques Cyber-Physique (SRCP)

On peut définir un SRCP comme une collection de robots autonomes qui évoluent dans un environnement intelligent ; ces robots peuvent communiquer et coopérer avec les autres robots et les autres entités de leurs environnements afin d'accomplir des opérations et des tâches précises.

3.1 Capacités de base d'un SRCP

La figure ci-dessous montre les trois éléments clés des SRCP, à savoir le Calcul, la Communication et le Contrôle.

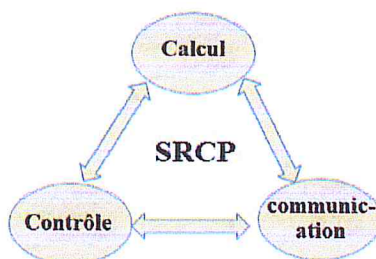


Figure 1 : Éléments principaux d'un SRCP [6].

3.2 Composants des SRCP

D'après [8], les SRCP se constituent d'entités physiques, d'un système informatique et d'un système de communication :

- *Entités physique* : elles peuvent être des humains, des robots, des capteurs, etc.
- *Système informatique* : cela inclut des ordinateurs, des systèmes d'exploitation, etc.
- *Systèmes de communication* : ils permettent de communiquer entre les différentes entités, comme le Wifi, NFC, Bluetooth, RFID, etc.

3.3 Technologies utilisés par les SRCP

Un SRCP utilise des technologies de communication, d'identification et de détection dans son environnement pour réaliser les tâches demandées. Parmi ces technologies, nous citons les plus importantes :

3.3.1 Wireless Fidelity (Wifi)

Le wifi est le mode de connexion le plus utilisé ; c'est une technologie de réseaux sans fil utilisant la norme IEEE 802.11 [9]. Il permet de relier plusieurs machines informatique (Ordinateur, téléphone, routeur, etc.) sur une dizaine de mètres.

a) Mode de fonctionnement

- *Connexion en étoile* : c'est le mode le plus utilisé et c'est le cas d'un routeur ; ce dernier est installé comme un point de contrôle de communication et chaque périphérique disposant d'un adaptateur wifi peut se connecter au routeur.
- *Point à point* : chaque périphérique dispose d'un adaptateur Wifi qui lui permet de se connecter à d'autres périphériques distants.

b) Avantages

- *Compatibilité* : la plupart des périphériques (Ordinateurs, tablettes, imprimantes, dispositifs de jeu, ...) et des systèmes d'exploitation sont compatibles avec la technologie wifi.
- *Facilité et maintenance* : les éléments du réseau wifi sont souvent faciles à installer et le coût de sa maintenance est généralement très réduit par rapport au réseau filaire.
- *Évolutivité* : les réseaux sans fil suivent quotidiennement l'évolution de la technologie pour couvrir les besoins.

c) Inconvénients

- *En terme de qualité* : le Wifi utilise à peu près la même bande de fréquence que d'autres types d'usage ce qui risque de causer des problèmes.
- *En terme de confidentialité* : pas toutes les méthodes de codage sont vraiment sécurisées. Certaines d'entre elles comme les WEP sont faciles à hacker avec des logiciels simples [06], et même les nouvelles méthodes de codage restent vulnérables.

3.3.2 Global Positioning System (GPS)

Le GPS est un système basé sur la radionavigation spatiale qui permet aux utilisateurs équipés d'un récepteur GPS de déterminer sur terre, mer et air leurs positions, vitesse et même l'heure à n'importe quel moment [10]. La figure ci-dessous montre la position exacte d'un emplacement grâce au GPS :

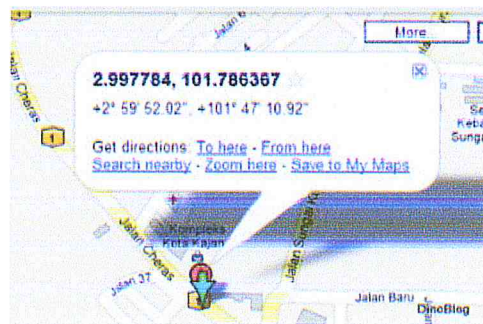


Figure 2 : position exacte d'un emplacement via un GPS

3.3.3 Near Field Communication (NFC)

Le NFC est une technologie de connectivité sans fil à courte-distance. Elle permet de faire des transactions simplifiées, l'échange de données, la connexion sans fil et le transfert des données entre deux objets proches l'un de l'autre (environ 10 cm) [11].

a) Modes de communication dans le NFC

Le NFC comporte trois modes de communication :

- *Mode Lecture/Écriture* : dans ce mode, l'objet peut lire et écrire des données avec les autres objets. Par exemple, avec un téléphone compatible avec la technologie NFC l'utilisateur peut se connecter à un site web, envoyer des SMS, etc. avec seulement une touche sur l'autre appareil.
- *Mode P2P* : ce mode permet aux appareils de partager des fichiers en toute sécurité.
- *Mode d'émulation de carte* : dans ce mode, un périphérique NFC peut remplacer une carte à puce sans contact. Par exemple, on peut l'utiliser dans le péage d'autoroutes, dans le contrôle d'accès, dans le paiement, etc. [11].

Dans la figure ci-dessous, le paiement d'une facture via un téléphone portable utilisant la technologie NFC :



Figure 3 : Paiement via NFC

b) Avantages de la NFC

- Possibilité d'utiliser des puces sans batterie ce qui permet d'avoir des étiquettes de très petite taille.
- Les puces NFC sont moins sensibles aux contraintes environnementales (température, etc.), et ne présentent pas les défauts des puces magnétiques [12].

c) Inconvénients

- Pour utiliser la technologie NFC, quelques fonctionnalités nécessitent de l'énergie en réserve comme le paiement mobile.
- Nécessite une courte distance [13].

4. Technologie Radio Frequency Identification (RFID)

4.1 Définition

Le système RFID ou tout simplement l'identification des objets à l'aide de la radiofréquence, est une technologie qui date de bien longtemps dans l'usage militaire [14]. Actuellement, la RFID est considérée par certains comme l'une des technologies informatiques les plus répandues. C'est une technologie qui permet de mémoriser et récupérer des informations à distance en utilisant des marqueurs appelés tag, étiquette, ou puce intelligente pour identifier des objets, personnes, animaux, etc. selon le domaine utilisé.

4.2 Fonctionnement des systèmes RFID

La technologie RFID devient de plus en plus populaire au point où presque tout peut être marqué. Elle offre la possibilité de détecter des personnes ou des objets sans contact et sans visibilité directe. Elle repose sur le principe suivant : un lecteur émet un signal radio et reçoit en retour les réponses des étiquettes (ou tags) qui se trouvent dans son champ d'action. Il existe une grande variété de systèmes RFID [12] ayant différents types de mémoires, différentes fréquences, différentes portées, différents types d'alimentation, etc.

Comme indiqué dans la figure 4, le principe est basé sur l'émission d'un champ magnétique par une antenne couplée à une puce électronique (transpondeur ou tag). Ces ondes magnétiques ont pour rôle de véhiculer l'information (entre la puce et son lecteur) et l'énergie nécessaire à l'activation de la puce. L'étiquette RFID ou tag est activé par les ondes émises par un lecteur composé lui-même d'une carte électronique et d'une antenne.

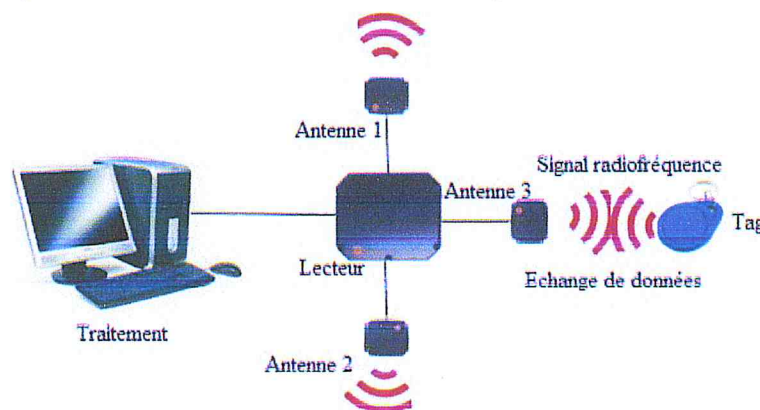


Figure 4 : Fonctionnement d'un Système RFID

4.3 Composants des Systèmes RFID

Pour équiper une entreprise avec un système de traçabilité RFID, celle-ci doit mettre en place un équipement de base spécifique composé de :

- *Tag (étiquette ou transpondeur)* : c'est un dispositif d'identification électronique qui est constitué d'une puce et d'une antenne.
- *Lecteur* : appelé aussi station de base qui a pour but d'identifier les transpondeurs.
- *Système de traitement des données* : c'est une infrastructure informatique qui sert à collecter et à exploiter les données.

4.3.2. Tag (étiquette)

L'étiquette électronique ou bien transpondeur (car elle comporte les deux fonctions émettre et répondre) est constituée d'un circuit intégré couplé à une antenne. Le tag interagit avec les ondes émises par le lecteur selon les caractéristiques qu'elle contient. L'antenne, généralement en cuivre, est déposée sur l'étiquette grâce à des ultrasons. Les étiquettes RFID fonctionnent à basse ou moyenne fréquence et utilisent un champ électromagnétique créé par l'antenne du lecteur. Le champ électromagnétique alimente l'étiquette en énergie et active la puce. Pour transmettre les informations qu'elle contient, elle va créer une modulation d'amplitude sur la fréquence porteuse. L'étiquette peut être apposée, portée, ou insérée dans un objet (colis, carte, véhicule, individu, etc.).

Les puces électroniques contiennent un identifiant de type *EPC (Electronic Product Code)*. C'est un identifiant unique permettant d'identifier un objet dans une chaîne de production. Les normes actuelles découpent cet EPC selon trois formats 64, 96 ou 125 bits [16]. On distingue trois types de tags : les tags passifs, actifs et semi-actifs :

a) Tags passifs

Les tags passifs fonctionnent à partir de l'énergie envoyée par le lecteur : l'antenne capte certaines fréquences provenant du lecteur qui fournissent suffisamment d'énergie pour lui permettre d'émettre à son tour son code d'identification unique. Les tags passifs ont une capacité qui varie entre 32 et 128 bits. Ils peuvent être fournis vierges ; l'utilisateur ou le fournisseur peut la munir d'une identification. Une fois que l'on souhaite poser un tag sur l'objet à tracer, l'utilisateur va écrire les données qui lui seront utiles par la suite. Ils sont bon marché et ont une durée de vie quasi illimitée. Ils constituent le gros du marché.

b) Tags actifs

Ils sont des tags comportant une pile interne leur permettant d'émettre un signal sans attendre le signal du lecteur pour l'activer. De ce fait, ils peuvent être lus depuis de longues distances, contrairement aux tags passifs. La présence d'une batterie permet également l'écriture de données, avec une mémoire allant jusqu'à 10 Kbits. Cependant les tags sont fournis vierges et pourront être écrits, effacés et lus plusieurs fois. Ces opérations peuvent dépasser les 500000 ou 1 million de fois mais avec une durée de vie ne dépassent pas 10 ans.

c) Tags semi-actifs

Les tags semi-actifs agissent comme des tags passifs au niveau de la communication et n'utilisent pas leur batterie pour émettre des signaux. Mais, elles leur permettent, par exemple, d'enregistrer des données lors du transport. Ces tags peuvent être utilisés pour enregistrer les changements de température lors du transport des marchandises [17].

3.3.2 Lecteur RFID

Un lecteur RFID est un dispositif qui fait la relation entre une application hôte et les tags RFID. Il est chargé de gérer la communication avec les tags et de transmettre leurs données à l'application hôte et vice-versa.

Une communication RFID est basée sur le principe de relation maître-esclave comme montré sur la figure 5 : le logiciel RFID commande le lecteur pour qu'il soit le maître et le tag joue le rôle d'esclave [25]. Les données contenues dans le tag sont alors lues et le lecteur peut également réécrire de nouvelles données dans le tag si nécessaire (lecture/écriture). Donc, le travail du lecteur dépend du type de l'étiquette.

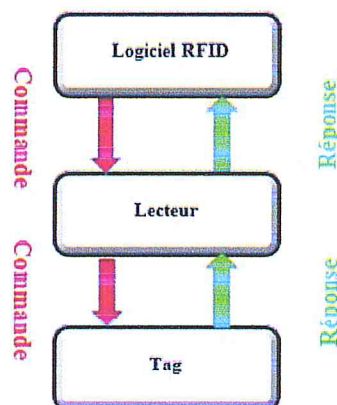


Figure 05 : Relation maître-esclave dans la communication RFID [23]

4.3.3 Application de traitement des données

Les données reçues par le lecteur sont traitées par un logiciel informatique qui traite les données selon les besoins.

4.4 Caractéristiques d'un système RFID

Un système RFID peut être classé en basse Fréquence (LF), Haute fréquence (HF) et Ultra Haute Fréquence (UHF) [18] [19].

4.4.1 La fréquence LF (Low Frequency)

Le système LF fonctionne avec une fréquence qui varie entre 125 et 134.2 KHz. La portée maximale de détection d'un tag répondant à cette fréquence est d'environ 50 cm. Ainsi, il pourrait être appliqué dans des environnements contenant du métal ou des liquides, en particulier dans le domaine de la gestion des animaux domestiques et du bétail, mais le prix de ces tags est très élevé [20]. Aussi, le temps de réponse d'une étiquette est relativement long (environ 1,6 ~ 5,8 ms). La figure 6 ci-dessous montre un exemple d'un tag LF.

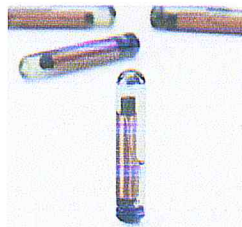


Figure 6 : exemple d'un tag LF

4.4.2 La fréquence HF (High Frequency)

Ils ont une fréquence de 13.56 MHz. La portée de détection du tag est d'environ 1 m. Ils sont moins chers que les tags LF et s'utilisent beaucoup avec les applications qui ne demandent pas de lire plusieurs tags en même temps et ils se caractérisent par une fréquence unique dans le monde (dans tous les pays). On peut les trouver dans la gestion des articles, chaîne d'approvisionnement, antivol, etc. (voir figure 7) [21].

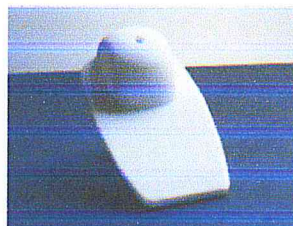


Figure 7 : exemple d'un tag HF

4.4.3 La fréquence UHF (Ultra High Frequency)

Le système RFID dans la bande de fréquence ultra-haute (UHF) est utilisé dans plusieurs applications et services, telle que l'identification des véhicules, l'accès, la sécurité,

les chaînes d'approvisionnement, etc. [22]. Ils sont moins chers que les HF et ils peuvent être détectés à une portée allant de 2m et 20 m. Exemple d'un tag UHF dans la figure ci-dessous.



Figure 8 : exemple d'un tag UHF

4.4.4 La fréquence SHF (Super High Frequency)

Elle est presque similaire à l'UHF dans ses performances ; sa portée typique maximale est de 1 m [23]. On peut l'utiliser dans l'identification des véhicules, l'accès, la sécurité, etc.

4.5 Structure de Electronic Product Code (EPC)

EPC est une norme d'identification universelle qui donne une identité unique à un objet physique spécifique. Chaque code est conçu pour qu'il soit unique par rapport aux autres objets physiques dans le monde et pour toujours. Les EPC sont généralement encodés sur des étiquettes RFID pour être utilisés dans les objets à suivre. Dans la figure 9, un exemple de norme EPC où le *Header* est la version du standard EPC utilisé, le *EPC Manager* est le code du fabricant, *Object Class* est le type de produit et le dernier *Serial Number* est le numéro unique. Cette figure montre la structure d'un code EPC de 96 bits. Ce code permet de représenter 79 milliards de milliards de milliards d'objets différents [24].

01	0000A89	00016F	000169DC0
Header	EPC Manager	Object Class	Serial Number
8 bits	28 bits	24 bits	36 bits

Figure 9 : exemple de norme EPC utilisé dans une canette de Coca-Cola

4.6 Avantages du système RFID

La technologie RFID possède plusieurs avantages, tels que :

- *La vitesse de marquage* : La possibilité de lire des dizaines de tags en une fraction de seconde, ce qui les rend très utiles dans certains domaines [26].
- *Possibilité de modification de données* : La plupart des tags offrent la possibilité de lire et écrire plusieurs fois, ce qui rend la modification et la mise à jour de données possible à tout moment pour des raisons de suivi et de traçabilité.

- *Grand volume de données* : Les tags RFID peuvent stocker un grand nombre de données et certains peuvent stocker un nombre de caractères allant jusqu'à 15000 caractères selon la capacité du tag.
- *Durée de vie* : Les étiquettes RFID peuvent avoir une durée de vie de dizaines d'années. Les données au cours de ces années peuvent subir des modifications plus d'un million de fois selon le type de l'étiquette avec un maximum de fiabilité.
- *Protection des contenus* : Possibilité de sécuriser les tags avec un mot de passe, pour les protéger lors de la lecture ou l'écriture. Avec cette protection contre l'accès des informations imprimées sur l'étiquette, la contrefaçon et le vol s'avèrent difficiles.
- *Meilleure accessibilité et résistance aux effets extérieurs* : Les étiquettes RFID sont insensibles à la poussière, aux tâches, aux frottements, à l'humidité. En plus, la lecture peut être effectuée en vrac « lecture simultanée de plusieurs tags ».

4.7 Inconvénients

Comme toutes les technologies de communication, la technologie RFID possède quelques limites, telles que :

- *Matériaux d'emballage* : Les ondes radio peuvent être perturbées par certains emballages, comme les métaux ferreux [24].
- *Orientation des antennes* : Les tags RFID ne nécessitent pas de liaison optique pour fonctionner, mais les lecteurs ne peuvent pas communiquer normalement avec un tag dont l'antenne est orientée perpendiculairement à l'antenne du lecteur.
- *Coûts* : Le coût de fabrication des systèmes RFID reste élevé dans certain cas.
- *Lecteurs multiples* : La majorité des lecteurs RFID sont conçus pour fonctionner avec un seul lecteur ce qui peut entraîner quelque problèmes en présence d'autres lecteurs.
- *Normes* : Les tags RFID fonctionnent selon des normes spécifique telles que EPCglobal 96-bits (915 MHz) ou l'ISO 14443 (13.56 MHz).

4.8 Domaines d'application des SRCP et de la technologie RFID

Les SRCP et la technologie RFID ne servent pas seulement à la gestion des stocks, des entrées et sorties de biens ; les performances de ces technologies ne cessent d'augmenter jour après jour en offrant plusieurs possibilités d'applications dans différents domaines.

4.8.1 Passeport biométrique RFID

Après les attentats du 11 septembre, les autorités américaines ont exigé la disposition d'un passeport biométrique pour l'entrée sur son territoire. Du coup, le phénomène a été suivi par la plupart des pays. Cette exigence a été en effet obligatoire depuis octobre 2005 [27].

La puce RFID diffuse un numéro qui est propre à la personne lorsqu'elle entre dans un champ radio. Le douanier utilise ce numéro pour consulter les informations personnelles conservées sur ses serveurs ; puis, il compare la personne qu'est devant lui à la fiche enregistrée afin de déterminer si elle doit être admise sur leur territoire ou non. La puce RFID ne contient aucune information personnelle (voir figure 10). Par la suite, l'on notera que sur le passeport, les données seront normalement blindées et seul un scanner possédant la clé d'authentification pourra accéder aux données enfermées. Puis elle renfermera en sus, les empreintes digitales du porteur afin de blinder les vérifications.

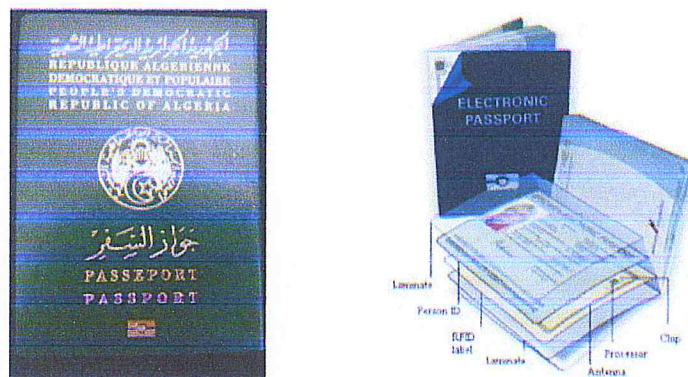


Figure 10 : exemple d'un passeport biométrique

4.8.2 Des drones dotés de lecteurs RFID pour faire l'inventaire de stock

L'inventaire dans les stocks des grands entrepôts deviendra différent en utilisant des drones [28]. En effet, l'Institut Fraunhofer développe InventAIRy (Figure 11) un nouveau drone volant et robotisé qui prend en charge l'inventaire, la tenue des dossiers et les rationalisations des opérations en entrepôt.

Ce drone est en mesure de compter le nombre de produits sur les étagères, réduire les coûts en éliminant les erreurs d'enregistrement, l'identification des goulets d'étranglement dans la production, et de garder un répertoire courant de l'inventaire. InventAIRy fonctionne avec un système RFID en installant des antennes sur les étagères ou des portiques à des endroits stratégiques. Une fois que le stock est déplacé, les antennes lisent les étiquettes et mettent à jour la base de données de l'inventaire.



Figure 11 : Un drone utilisé par InventAIRy

4.8.3 Surveillance Robotique Cyber Physique de l'environnement

La surveillance et la détection environnementale de la nature et de la faune par les humains est souvent pénible et dangereuse [29]. Le développement connu dans la technologie telle que la communication sans fil longue distance, le réseau de capteurs sans fil auto-organisé, les caméras robotiques en réseau, les systèmes multi-agents/robots et le cloud computing sur Internet permettent de développer une nouvelle génération de surveillance robotique cyber-physique (SR-CS) avec une architecture distribuée qui améliorent la capacité de fusion des réseaux de capteurs, pour relever le défi de la nature sauvage.

Le système de SR-CS devient une application typique et importante de SCP [30], dans lequel différents types d'actionneurs, de moniteurs et de capteurs sont adoptés pour la surveillance et l'analyse en temps réel. Les caméras, les enregistreurs et les capteurs, sont installés sur les tours et les arbres et peuvent être connectés à Internet via une communication sans fil à longue portée. Dans de tels systèmes de surveillance cyber-physiques, les actionneurs, les moniteurs et les capteurs sont le monde physique, tandis que le réseau de communication sans fil relie le monde physique et les systèmes informatiques [31].

Dans la figure ci-dessous quelques animaux captés dans différents temps à l'aide du système de la SR-CP dans l'environnement :

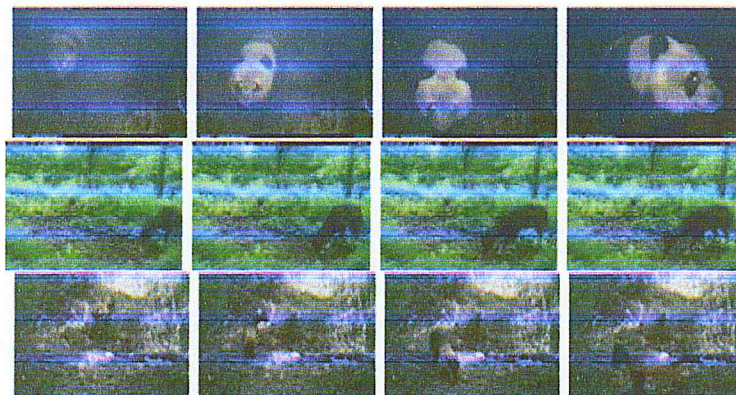


Figure 12 : animaux captés par les caméras infrarouges en forêt [31].

4.8.4 RFID dans le secteur de la santé

Les solutions informatiques modernes et la technologie RFID offrent de nombreux avantages pour les industries médicales et pharmaceutiques. Elles permettent d'améliorer l'identification du patient (figure 13), d'automatiser la gestion des marchandises et de faciliter le suivi des dispositifs médicaux. Elle augmente automatiquement la sécurité des patients dans les hôpitaux, la transparence des processus et la qualité de l'administration de soins de santé aux patients, l'identification automatique des consommables et accessoires, etc.



Figure 13 : Identification d'un patient à l'aide d'un bracelet RFID [38].

4.8.5 Gestion de la chaîne d'approvisionnement

En matière logistique, quatre niveaux d'applications peuvent être distingués [32] :

- *Expédition* : L'étiquette peut faciliter le picking des produits, la constitution des palettes, leurs tris par destination et le contrôle du chargement. L'étiquette peut contenir, outre l'identification du produit ou du contenu d'une palette, celle du numéro de lot de fabrication, l'identification du destinataire, le numéro de commande, des particularités de manutention, etc. Ces informations collectées au moment du chargement peuvent être stockées dans le tag RFID du conteneur ou du moyen de transport afin de faciliter le contrôle en cours de transit et douane.
- *Réception* : Les données pourront automatiquement être collectées pour effectuer les contrôles, mettre à jour les stocks et effectuer les rapprochements avec les documents commerciaux.
- *En transit* : L'étiquette permet de tracer le produit à chaque point de chargement et de déchargement ou simplement de passage. Ainsi, l'expéditeur peut être à tout moment informé du déroulement du transport.
- *Local* : Les étiquettes permettent l'inventaire du produit, mais aussi la gestion des supports de manutention et des équipements.

Chapitre II

Description et prise en main du matériel RFID de IFM

1. Introduction

Ces dernières années, le marché de la technologie RFID a connu un grand développement. Le chiffre d'affaire de vente des infrastructures RFID (lecteurs, étiquettes et logiciels) n'a cessé d'augmenter et les perspectives de croissance sont très bonnes pour toutes les entreprises du secteur [33]. Parmi les sociétés qui fabriquent des équipements RFID, la société allemande IFM est reconnue pour sa haute qualité des produits offerts.

Dans la Division Productique et Robotique (DPR) du CDTA, les chercheurs utilisent principalement le système d'exploitation Linux pour le contrôle des robots ; ces derniers tournent en général sous le middleware ROS (Robot Operating System).

Après une description du matériel qui nous a été confié au cours de ce travail, en particulier le lecteur **DTE800** d'IFM, nous présenterons les fonctions de base que nous avons développées. En effet, le lecteur DTE800 étant fourni avec des bibliothèques DLL (Dynamic Link Library) fonctionnant seulement sous le système Windows ; il est de ce impossible de les utiliser sous Linux. Il nous a donc fallu développer en Java une bibliothèque permettant d'utiliser le lecteur sous n'importe quelle plateforme.

2. Société IFM

La société IFM est une société spécialisée dans l'industrie de l'automatisation depuis sa création en 1969. Aujourd'hui, IFM est devenue un des leaders mondiaux dans le développement de capteurs, des systèmes de contrôle et de différents systèmes industriels [34]. IFM offre des produits de haute qualité et environ 70 % des produits sont développés et fabriqués en Allemagne.

IFM apporte également des solutions aux exigences particulières de différentes industries. Outre les détecteurs de position et les capteurs de processus, des détecteurs de contrôle de mouvement et la technologie de sécurité complètent l'offre IFM. De plus, IFM offre des produits pour le traitement d'images industrielles et la communication industrielle ainsi que des systèmes d'identification et des systèmes pour engins mobiles. Le groupe IFM (logo figure 1) a réalisé un chiffre d'affaires de 775 millions d'euros en 2016.

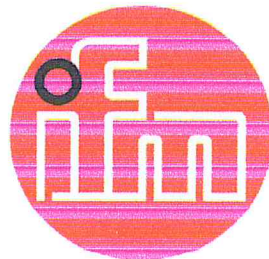


Figure 14 : Logo IFM

3. Tags RFID de IFM

Une vingtaine de tags IFM sont disponibles au niveau de la DPR pour la réalisation de ce travail. Les tags IFM disponibles sont des tags UHF existant sous différentes formes (voir figure 2) : cylindrique, rectangulaire, etc. Leur capacité de stockage peut aller jusqu'à 128 bits EPC.

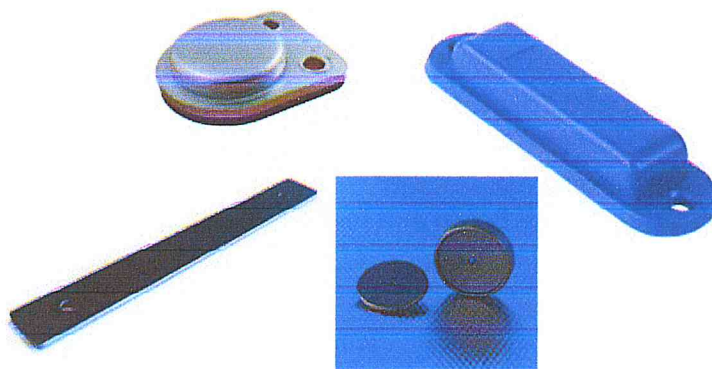


Figure 15 : Différents tags IFM disponibles dans la DPR

3.1 Types de mémoire des étiquettes RFID

La mémoire d'une étiquette RFID est divisée en quatre zones de mémoire :

3.1.1 Mémoire réservée

Cette zone de mémoire stocke le mot de passe kill et le mot de passe d'accès (chacun est sur 32 bits). Le mot de passe de suppression désactive définitivement la balise (très rarement utilisé) et le mot de passe d'accès est configuré pour verrouiller/déverrouiller les capacités d'écriture de l'étiquette. Cette zone de mémoire n'est accessible que si l'on souhaite spécifier un certain mot de passe.

3.1.2 Mémoire TID

Cette zone est utilisée uniquement pour stocker le numéro d'identification de la marque unique par le fabricant lorsque l'étiquette est fabriquée. Cette partie de mémoire ne peut pas être modifiée.

3.1.3 Mémoire EPC

Cette zone de mémoire stocke le code EPC ou le code électronique des produits. Il a un minimum de 96 bits de mémoire enregistrable. La mémoire EPC est ce qui est

généralement utilisé dans la plupart des applications si elles ne nécessitent que 96 bits de mémoire. Certaines étiquettes ont la capacité d'attribuer plus de bits à la mémoire EPC à partir de la mémoire utilisateur. La mémoire EPC est la première zone inscriptible.

3.1.4 Mémoire utilisateur

Si l'utilisateur a besoin de plus de mémoire que celle qui est disponible dans la section EPC, certaines étiquettes ont une mémoire utilisateur étendue qui peut stocker plus d'informations. En ce qui concerne la mémoire utilisateur, il n'y a pas de norme sur le nombre de bits de mémoire pouvant être écrits sur chaque étiquette. Typiquement, la mémoire étendue ne dépasse pas 512 bits ; mais, il existe des étiquettes de haute mémoire qui peuvent stocker jusqu'à 4K ou 8K octets de mémoire.

4. Lecteur DTE800

Le lecteur RFID DTE800 d'IFM (figure 3) est un lecteur multi-protocoles destiné à lire des tags RFID actifs et passifs dans une gamme de fréquences de 865 à 868 MHz.



Figure 16 : Reader IFM DTE800

Le lecteur possède jusqu'à quatre prises d'antennes externes permettant de raccorder les antennes émettrices/réceptrices nécessaires à la communication avec les tags RFID. L'appareil possède, suivant le modèle, différentes interfaces de communication pour l'intégration à différentes structures. Afin de le protéger, l'appareil doit être loin de l'humidité, des gouttes et des projections d'eau. Il ne doit pas être disposé à proximité d'un appareil de chauffage ni être exposé directement aux rayons du soleil ou être utilisé dans un environnement humide. Le lecteur ne doit être utilisé que dans un climat tempéré.

4.1 Fournitures

Le reader est fourni avec un CD contenant un logiciel pour une utilisation à titre de test, des notices d'utilisation et une interface de programmation. Cette interface est disponible sous forme d'une DLL pour la programmation orienté objet sous le système d'exploitation Microsoft Windows.

4.2 Utilisation du Reader

Les chercheurs de la DPR du CDTA travaillent généralement sous un environnement Linux notamment en raison des robots qui fonctionnent avec le middleware ROS. Pour pouvoir utiliser le lecteur sur un tel système, il nous a fallu développer les fonctionnalités de base en utilisant le langage Java. Nous décrivons dans ce qui suit la démarche suivie.

Concernant la communication avec le lecteur, nous avons choisi d'utiliser une connexion *Ethernet* avec le protocole de communication *TCP/IP*.

4.3 Communication avec le reader via la connexion Ethernet

La communication entre le programme et le système RFID s'effectue toujours comme suit :

- Appel : envoyer une commande au système RFID et recevoir une validation ; puis,
- Rappel : le résultat est envoyé via un rappel par le système RFID.

4.4 Format de la trame

La trame est donnée toujours sous le format suivant en hexadécimal : **Un code de début + Un bloc de données + Un code de fin**. Notons que le code de début de la trame se compose de : 0xAA 0xBB 0x01 0x01 ; le code de fin est : 0xAA 0xCC. Donc, pour envoyer une commande au reader, la trame aura la forme suivante : 0xAA 0xBB 0x01 0x01 + Code de la fonction demandée + 0xAA 0xCC. De même pour le rappel : 0xAA 0xBB 0x01 0x01 + Données renvoyées par le reader + 0xAA 0xCC.

4.5 Port de communication

Le port de communication TCP/IP utilisé est le port 4007.

5. Outil Wireshark

Wireshark est un analyseur de paquets gratuit et open source. Il est utilisé pour le dépannage du réseau, l'analyse, le développement de logiciels et de protocoles de

communication mais aussi pour l'enseignement. Il dispose de presque toutes les fonctionnalités standards d'un analyseur de protocole, et plusieurs d'autres fonctionnalités qui ne sont pas disponibles dans d'autres produits. Il fonctionne sur toutes les plateformes informatiques populaires (Unix, Linux et Windows) [35]. Les administrateurs réseau l'utilisent pour résoudre les problèmes de réseau ; les ingénieurs de sécurité de réseau l'utilisent pour examiner les problèmes de sécurité. Les développeurs l'utilisent aussi pour déboguer les implémentations de protocoles ; d'autres l'utilisent pour apprendre le protocole TCP/IP. *Wireshark* est très utile pour capturer des paquets et analyser la structure des paquets. Il permet d'analyser un trafic enregistré dans un fichier annexe, mais également et surtout le trafic en direct sur des interfaces réseau.

6. Fonctionnalités de base implémentées

Chaque commande est identifiée par un ID. Ci-dessous quelques fonctions de base implémentées avec leur code d'identification :

Commande	Code commande	Code réponse
GetPortPower	0x0005	0x8005
GetExtResultFlag	0x0018	0x8018
SetExtResultFlag	0x0019	0x8019
SyncGetEPCs	0x0101	0x8101
SyncBulkGetEPCs	0x0102	0x8102
SyncWriteDataSpecific	0x0106	0x8106
ASyncGetEPCs	0x0111	0x8111
ASyncStopCommand	0x0118	0x8118
GetAntennaList	0x0201	0x8201
SetAntenna	0x0202	0x8202
MeasureTxPwr	0x0208	0x8208

Tableau 1 : Liste des commandes DTE800

Pour bien traiter les réponses du reader, nous avons visualisé les trames envoyées par le lecteur à partir du logiciel fourni via *Wireshark*. La figure 4 montre un exemple de capture des trames avec *Wireshark*. En analysant le trafic on remarque que :

- La trame sélectionnée provient de l'adresse source 192.168.0.2 ; l'adresse de destination est 192.168.0.1 ce qui correspond respectivement à l'adresse de l'ordinateur et du reader.
- Le port de destination est 4007.
- Les données envoyées sont : « aa bb 01 01 01 01 aa cc » ; « aa bb 01 01 » correspond au code de début de la trame et « aa cc » représente le code de fin. La commande envoyée est « 01 01 » qui représente le code de la commande SyncGetEPCs.

No.	Time	Source	Destination	Protocol	Length	Info
46	10.602841	192.168.0.1	192.168.0.2	TCP	60	4007 → 1041 [ACK] Seq=83 Ack=82 Win=2048 Len=0
47	10.636407	192.168.0.1	192.168.0.2	TCP	63	4007 → 1041 [PSH, ACK] Seq=83 Ack=82 Win=2048 Len=9
48	10.652254	192.168.0.2	192.168.0.1	TCP	62	1041 → 4007 [PSH, ACK] Seq=82 Ack=92 Win=17048 Len=8
49	10.666747	192.168.0.1	192.168.0.2	TCP	60	4007 → 1041 [ACK] Seq=92 Ack=90 Win=2048 Len=0
50	10.668636	192.168.0.1	192.168.0.2	TCP	63	4007 → 1041 [PSH, ACK] Seq=92 Ack=90 Win=2048 Len=9
51	10.678142	192.168.0.2	192.168.0.1	TCP	62	1041 → 4007 [PSH, ACK] Seq=90 Ack=101 Win=17037 Len=8
52	10.684444	192.168.0.1	192.168.0.2	TCP	63	4007 → 1041 [PSH, ACK] Seq=101 Ack=98 Win=2048 Len=9
53	10.685960	192.168.0.2	192.168.0.1	TCP	62	1041 → 4007 [PSH, ACK] Seq=98 Ack=110 Win=17028 Len=8
54	10.701095	192.168.0.1	192.168.0.2	TCP	60	4007 → 1041 [ACK] Seq=110 Ack=106 Win=2048 Len=0
55	10.703737	192.168.0.1	192.168.0.2	TCP	63	4007 → 1041 [PSH, ACK] Seq=110 Ack=106 Win=2048 Len=9

Destination Port: 4007
 [Stream index: 0]
 [TCP Segment Len: 8]
 Sequence number: 90 (relative sequence number)
 [Next sequence number: 98 (relative sequence number)]
 Acknowledgment number: 101 (relative ack number)
 Header Length: 20 bytes
 Flags: 0x010 (PSH, ACK)
 Window size value: 17037
 [Calculated window size: 17037]
 [Window size scaling factor: -1 (unknown)]
 Checksum: 0x0176 [unverified]
 [Checksum Status: Unverified]
 Urgent pointer: 0
 [SEQ/ACK analysis]
 Data (8 bytes)
 Data: aabb010101aa
 [Length: 8]

```

0000  00 d0 55 04 05 49 d4 be d9 12 2b 1a 08 00 45 00  ..U..I.. ..+...E.
0010  00 30 07 b3 40 00 00 06 00 00 c0 a8 00 02 c0 a8  ..0..@.....
0020  00 01 04 11 0f a7 c9 fb fd d7 1f df 06 e4 50 18  .....P.
0030  42 8d 81 76 00 00 aa bb 01 01 01 01 aa cc      B..V.....
  
```

Figure 17 : capture des trames avec *Wireshark*

6.1 Détail des commandes et leurs réponses

6.1.1 GetPortPower

La commande *GetPortPower* permet de connaître la puissance du port de chaque antenne.

a) Structure du bloc de commande

I1 I2 PN

I1: ID de commande low byte

I2: ID de commande high byte

PN : Numéro du port d'antenne (Entre 1 et 4)

b) Structure du bloc de réponse

I1 I2 RF PN [PP]

I1: ID de réponse low byte

I2: ID de réponse high byte

RF: Result flag

PN : Numéro du port d'antenne (Entre 1 et 4)

PP: Puissance du port

c) result flag possibles

RRUI4RESULTFLAG_NOERROR

RRUI4RESULTFLAG_NOLICENSE

RRUI4RESULTFLAG_OUTOFRANGE

d) Puissance du port

La puissance du port est renvoyée si le *result flag* est RRUI4RESULTFLAG_NOERROR ; il indique le réglage du rayonnement de puissance du port d'antenne en fonction de l'atténuation du câble et du gain d'antenne.

6.1.2 GetExtResultFlag

La plupart des commandes renvoient un *result flag* ou un *extended result flag* ; chacun d'eux représente des informations sur l'état de la commande.

a) Structure du bloc de commande

I1 I2

I1: ID de commande low byte

I2: ID de commande high byte

b) Structure du bloc de réponse

I1 I2 EF

I1: ID de réponse low byte

I2: ID de réponse high byte

EF: Extended result flag

c) Extended result flags possibles

RRUI4ERF_0000

RRUI4ERF_000A

RRUI4ERF_00R0

RRUI4ERF_00RA

RRUI4ERF_0T00

RRUI4ERF_OT0A
RRUI4ERF_OTR0
RRUI4ERF_OTRA
RRUI4ERF_P000
RRUI4ERF_P00A
RRUI4ERF_P0R0
RRUI4ERF_P0RA
RRUI4ERF_PT00
RRUI4ERF_PT0A
RRUI4ERF_PTR0
RRUI4ERF_PTRA

6.1.3 SetExtResultFlag

a) Structure du bloc de commande

I1 I2 EF
I1: ID de commande low byte
I2: ID de commande high byte
EF: Extended result flag

b) Structure du bloc de réponse

I1 I2 RF
I1: ID du réponse low byte
I2: ID du réponse high byte
RF: Result flag

c) Result flags possibles

RRUI4RESULTFLAG_NOERROR
RRUI4RESULTFLAG_OUTOFRANGE

d) Explication des codes ExtendedResultFlag

Le *ExtendedResultFlag* renvoie des informations sur l'état de fonctionnement de l'antenne, de la valeur *RSSI* ou même du *TimeStamp*. Ci-dessous quelques résultats possibles pour la valeur renvoyée par le *ExtendedResultFlag* :

00, // Pas de transmission ni de l'antenne ni de RSSI, ni du Timestamp.
01, // Transmission de l'antenne.
02, // Transmission du RSSI.
03, // Une transmission de l'antenne et du RSSI.

- 04, // Une transmission du TimeStamp.
- 05, // Une transmission du TimeStamp et de l'antenne.
- 06, // Une transmission du TimeStamp et RSSI.
- 07, // Une transmission du Timestamp, RSSI et de l'antenne.

6.1.4 Lecture des Tags

Certaines commandes s'exécutent en deux modes de communication : synchrone et asynchrone. Lorsqu'une commande synchrone est envoyée, cette commande est exécutée une fois. Par exemple, lors de la lecture synchrone des EPC de tous les tags dans le champ d'antenne(s) (voir figure 5), ceux-ci sont lus une seule fois ; ils consistent en la liste des EPC de tous les tags lus est renvoyée comme résultat. Autrement dit, un résultat doit être fourni pour chaque appel de la commande.

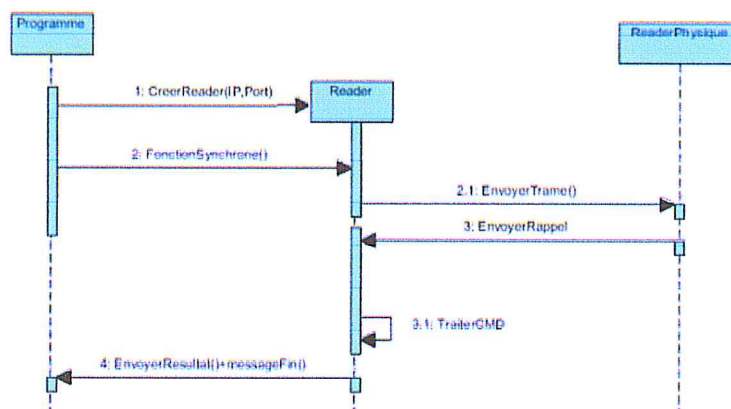


Figure 18 : Diagramme séquence Interaction synchrone

Lorsqu'une commande est envoyée pour la lecture asynchrone des EPC (voir figure 6), un résultat individuel est transmis pour chaque tag qui entre ou sort du champ d'antenne(s). Cela se produit jusqu'à ce qu'une commande d'arrêt est envoyée.

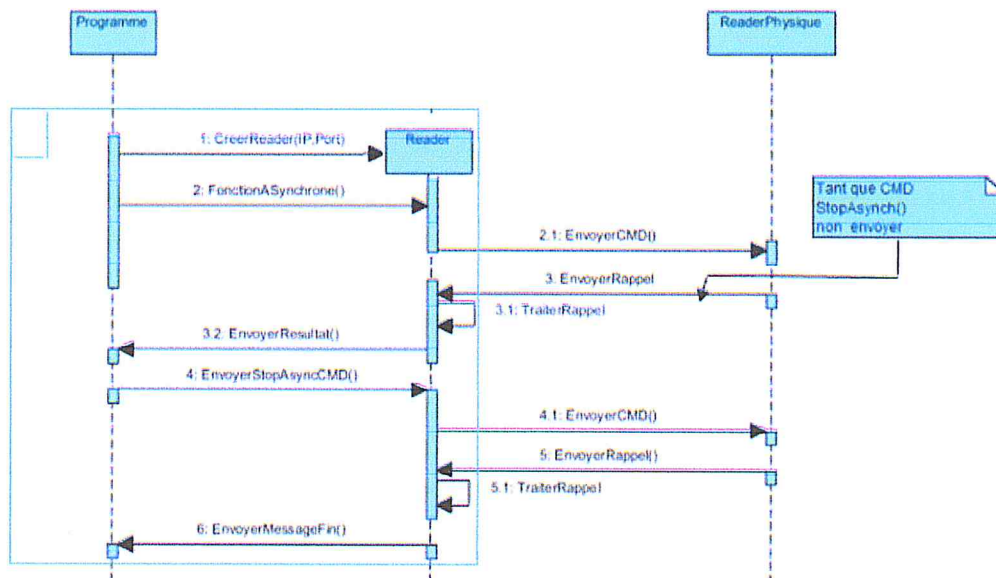


Figure 19 : Diagramme séquence Interaction Asynchrone

6.1.4.1 SyncGetEPCs

C'est la commande de lecture synchrone.

a) Structure du bloc de commande

I1 I2

I1: ID de commande low byte

I2: ID de commande high byte

b) Structure du bloc de réponse

I1 I2 RF [EF]

[[AP] [RS] [T1 T2 T3 T4] EW [E1 ... En]]

...

[[AP] [RS] [T1 T2 T3 T4] EW [E1 ... En]]

I1: ID du réponse low byte

I2: ID du réponse high byte

RF: Result flag

EF: Extended result flag

AP: Port d'antenne

RS: Valeur RSSI

T1: Time stamp LSB

T2: Time stamp 2nd byte

T3: Time stamp 3rd byte

T4: Time stamp MSB

EW: Nombre des mots EPC

E1 : EPC LSB

...

En : EPC MSB

c) Result flags possibles

RRUI4RESULTFLAG_NOERROR

RRUI4RESULTFLAG_NOSTANDARD

RRUI4RESULTFLAG_NOTAG

RRUI4RESULTFLAG_WRONGCFM

Pour effectuer une commande de lecture synchrone on envoie la trame suivante : « aabb010101aacc ». En exécutant cette commande, le lecteur renvoie la trame de réponse contenant tous les tags trouvés dans le champ d'antenne(s).

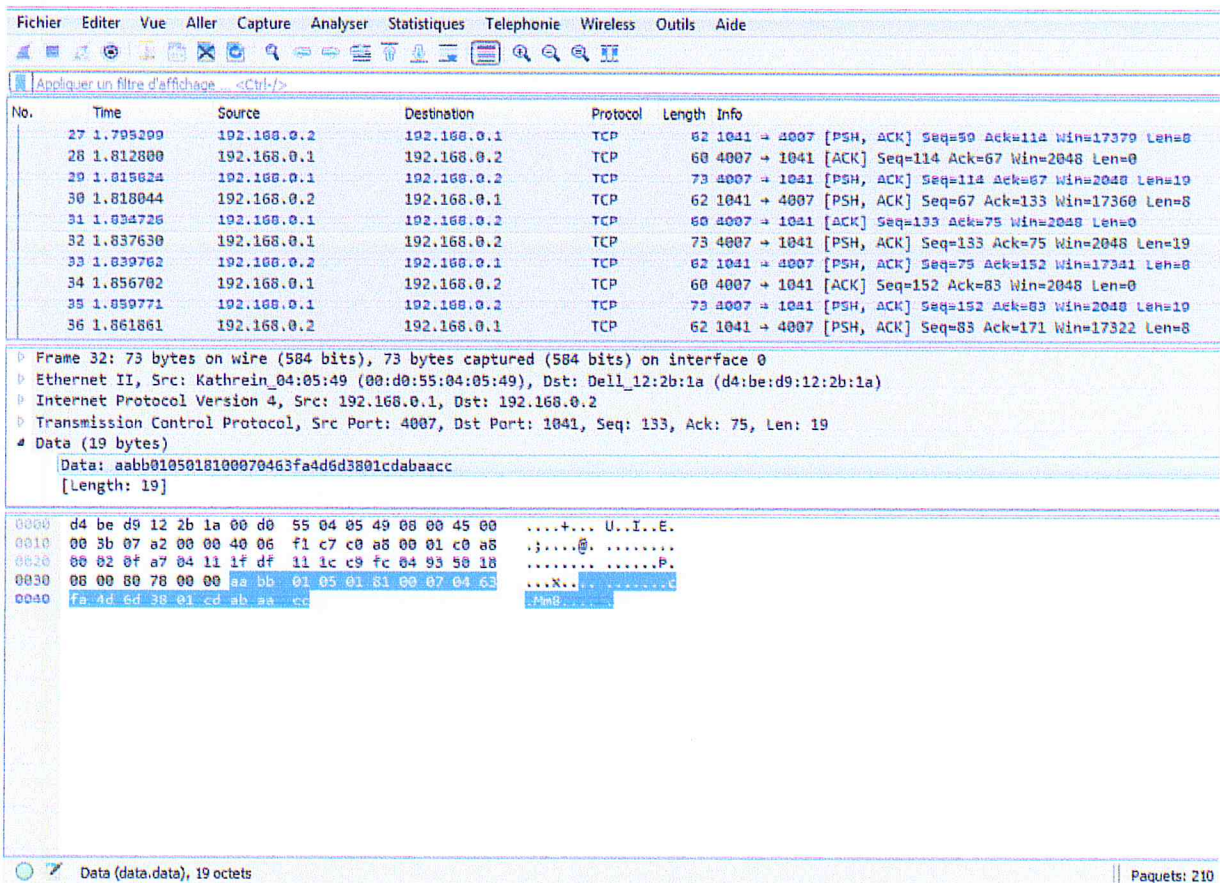


Figure 20 : Capture de la trame reçue par le reader

Afin de faciliter le traitement de la trame, nous avons utilisé *Wireshark*. L'analyse de la trame de la figure 7 montre que :

- La trame reçue est « **aa bb 01 05 01 81 00 07 04 63 fa 4d 6d 38 01 cd ab aa cc** » :
« **aa bb 01 05** » est le code de début de la trame et « **aa cc** » est le code de fin.

En comparant avec la structure du bloc :

- **01 81** : correspond à I1 I2 ce qui donne 8101 et c'est le code approprié à la réponse *SyncGetEPCs*.
- **00** : correspond au *ResultFlag* où cette valeur est renvoyée s'il n'y a pas d'erreur.
- **07** : correspond au *extendedResultFlag*, en informant qu'il y a eu transmission du *Timestamp*, *RSSI* et de l'antenne.
- **04** : est le numéro d'antenne, donc le tag est détecté avec l'antenne numéro 4.
- **63** : est la valeur *RSSI* reçue en hexadécimal, donc la valeur reçue est 99 en décimal.
- **fa 4d 6d 38** : est le *TimeStamp* du tag. Le *TimeStamp* est la valeur en secondes depuis le 01-01-1970 00h:00
- **01** : est la taille du premier tag ; si on l'inverse on trouve 10 en hexadécimal. Donc, nous avons 16 bits ce qui indique que le tag suivant a 2 octets.
- **cd ab** : est la valeur EPC du tag, en inversant les octets on trouve la valeur du tag **ab cd**.

6.1.4.2 ASyncGetEPCs

C'est la commande de lecture asynchrone.

a) Structure du bloc de commande

I1 I2

I1: ID de commande low byte

I2: ID de commande high byte

b) Structure du bloc de réponse

I1 I2 RF

I1: ID de réponse low byte

I2: ID de réponse high byte

RF: Result flag

c) Result flags possible

RRUI4RESULTFLAG_NOERROR

RRUI4RESULTFLAG_NOSTANDARD

RRUI4RESULTFLAG_WRONGCFM

d) Structure du bloc de réponse

I1 I2 CG EF [AP] [RS] [T1 T2 T3 T4] EW [E1 ... En]

I1: ID du réponse low byte

I2: ID du réponse high byte

CG: Indique si un tag est entré ou sorti du champ d'antenne

EF: Extended result flag

AP: Port d'antenne

RS: Valeur RSSI

T1: Time stamp LSB

T2: Time stamp 2nd byte

T3: Time stamp 3rd byte

T4: Time stamp MSB

EW: Nombre des mots EPC

E1 : EPC LSB

...

En : EPC MSB

Lorsqu'une commande asynchrone est envoyée, on reçoit toujours deux réponses. Ci-dessous un exemple de trame analysée avec *Wireshark* illustrant ce type de communication.

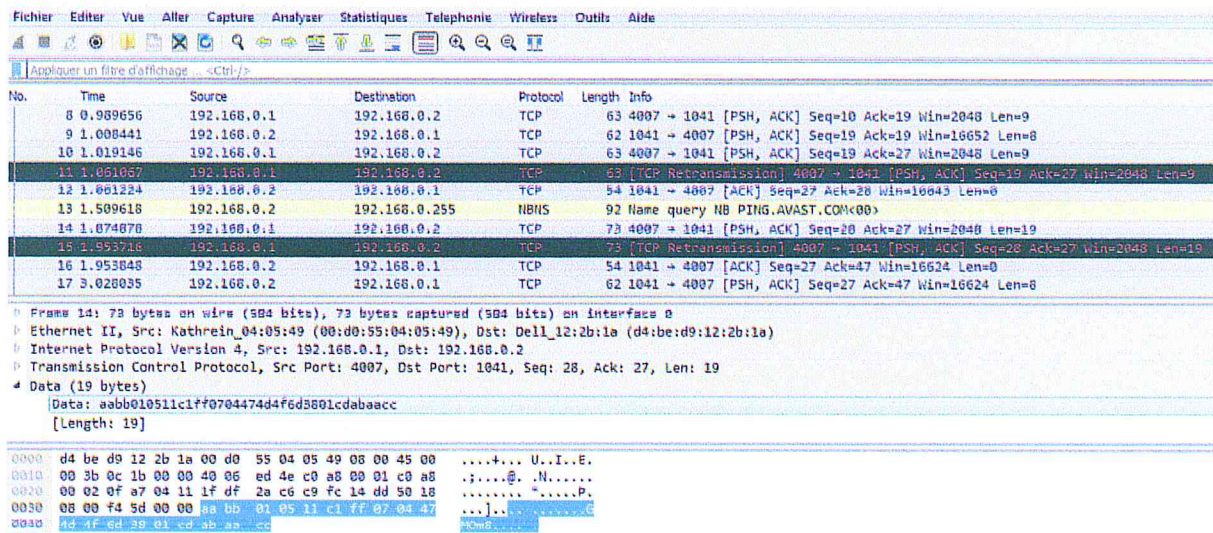


Figure 21 : Capture de la trame de lecture asynchrone

La figure 8 montre que :

- La trame reçue est « AA BB 01 05 11 C1 FF 07 04 47 4D 4F 6D 38 01 CD BA AA CC » : « AA BB 01 05 » est le code de début de la trame et « AA CC » est le code de fin.

En comparant avec la structure du bloc :

- **11 C1** : correspond à I1 I2 ce qui donne C111 et c'est le code approprié à la réponse *ASyncGetEPCs*.
- **00** : correspond au *ResultFlag* et 00 est renvoyé s'il n'y a pas d'erreur.
- **FF** : montre que le tag est entré dans le champ d'antenne.
- **07** : correspond au *extendedResultFlag* donc il existe une transmission du *Timestamp*, RSSI et de l'antenne.
- **04** : est le numéro d'antenne, donc le tag est détecté avec l'antenne numéro 4.
- **47** : est la valeur *RSSI* reçue en hexadécimal, donc la valeur reçue est 71.
- **4D 4F 6D 38** : est le *TimeStamp* du tag.
- **01** : est la taille du premier tag ; si on l'inverse on trouve 10 en hexadécimal. Donc, nous avons 16 bits ce qui indique que le tag suivant a 2 octets.
- **CD AB** : c'est la valeur EPC du tag ; en inversant les octets on trouve la valeur du tag **AB CD**.

6.1.5 ASyncStopCommand

La commande *ASyncGetEPCs* ne s'arrête pas jusqu'à qu'une commande d'arrêt est envoyée. La commande d'arrêt est *ASyncStopCommand*.

a) Structure du bloc de commande

I1 I2

I1: ID de commande low byte

I2: ID de commande high byte

b) Structure du bloc de réponse

I1 I2 RF

I1: ID du réponse low byte

I2: ID du réponse high byte

RF: Result flag

c) Result flags possible

RRUI4RESULTFLAG_NOERROR

RRUI4RESULTFLAG_WRONGCFM

6.1.6 SyncWriteDataSpecific

C'est une des commandes permettant d'effectuer une l'écriture dans le tag.

a) Structure du bloc de commande

I1 I2 EW [E1 ... En] W1 W2 W3 W4 MB P1 P2 P3 P4 WC DL DH [DL DH ... DL DH]:

I1: ID de commande low byte

I2: ID de commande high byte

EW: Nombre des mots EPC

E1: EPC LSB

...

En: EPC MSB

W1: Mot de passe octet bas

W2: Mot de passe 2eme octet

W3: Mot de passe 3eme octet

W4: Mot de passe haut octet

MB: 01 par default

P1 a P4 FFFFFFFF par default

WC: nombre de mots a écrire

DL: mots à écrire bas octet

DH: mots à écrire haut octet

b) Structure du bloc de réponse

I1 I2 RF [EF]

[[AP] [RS] [T1 T2 T3 T4] EW [E1... En] EC]

...

[[AP] [RS] [T1 T2 T3 T4] EW [E1 ... En] EC]

I1: ID du réponse low byte

I2: ID du réponse high byte

RF: Result flag

EF: Extended result flag

AP: Numéro du port d'antenne

RS: Valeur RSSI

T1: Time stamp LSB

T2: Time stamp 2nd byte

T3: Time stamp 3rd byte
T4: Time stamp MSB
EW: nombre de mots EPC
E1: EPC LSB
...
En: EPC MSB
EC: code d'erreur du Tag

c) Result flags possibles

RRUI4RESULTFLAG_NOERROR
RRUI4RESULTFLAG_OUTOFRANGE
RRUI4RESULTFLAG_NOSTANDARD
RRUI4RESULTFLAG_NOTAG
RRUI4RESULTFLAG_WRONGCFM

Notons que d'autres fonctionnalités ont été développées afin de répondre au besoin de la DPR du CDTA ; mais, nous nous sommes limités dans ce chapitre aux plus importantes.

7. Conclusion

La détection des objets à l'aide de la technologie RFID nécessite différents équipements. Le choix des lecteurs et des tags est très important afin d'assurer un bon fonctionnement de l'infrastructure RFID.

Dans ce chapitre, nous nous sommes intéressés au lecteur RFID DTE800 de IFM en détaillant la plupart des fonctionnalités de base que nous avons mises en œuvre. Ceci est dans le but de pouvoir les utiliser dans un environnement Linux (ou autre). Pour atteindre cet objectif, nous avons utilisé le langage Java ainsi qu'un outil d'analyse de trames (*Wireshark*).

Dans le chapitre suivant, nous allons aborder le problème de la localisation d'objets dans un environnement RFID.

Chapitre III

Approche proposée pour la localisation d'objets dans une infrastructure RFID

1. Introduction

Les tags RFID peuvent être utilisés dans divers domaines. Ils peuvent non seulement être considérés comme des outils d'identification mais aussi comme outils d'aide à la localisation et la navigation dans un environnement physique. En effet, il existe plusieurs applications où la localisation basée sur la RFID peut être appliquée, nous citerons la gestion des entrepôts de produits.

Dans ce chapitre, nous nous intéressons au problème de la localisation d'objets dans une infrastructure RFID. Nous commencerons par décrire l'approche proposée par Zambonili et Mamei [36] basée sur le comportement animal (section 2). Ensuite, nous adapterons cette approche à notre contexte, à savoir la localisation d'objets par des robots AGV qui déplacent des produits dans un entrepôt équipé d'une infrastructure RFID (sections 3 à 7).

2. Localisation d'objets basée sur l'interaction omniprésente avec les étiquettes RFID

2.1 Interaction à base de phéromones

Les fourmis et autres insectes sociaux interagissent en diffusant des marqueurs chimiques (c'est-à-dire des phéromones) lorsqu'ils se déplacent dans l'environnement et en se dirigeant vers leurs actions par les concentrations perçues de phéromones [37]. Ce mécanisme simple d'interactions locales et de coordination par l'environnement, appelé *stigmergie*, permet aux fourmis de s'organiser de manière globale en organisant leurs activités collectives d'une manière intelligente et d'agir de manière adaptative dans un environnement inconnu. Étant donné qu'un tel comportement adaptatif contextuel est exposé malgré les compétences très limitées des individus dans l'acquisition et le traitement cognitif de l'information contextuelle, les systèmes d'insectes sociaux se caractérisent par *l'intelligence de l'essaim*.

L'exemple classique pour montrer la puissance de l'interaction basée sur la phéromone est les fourragères. Des fourmis dans une colonie, à la recherche de nourriture, quittent le nid et commencent à errer. Quand elles trouvent de la nourriture, les fourmis commencent à répandre une phéromone et tentent de revenir au nid, créant ainsi un chemin menant à la source de la nourriture. Lorsqu'une fourmi cherche de la nourriture, elle peut exploiter indirectement l'expérience passée d'autres fourmis en suivant une piste de phéromone existante pour atteindre les sources de nourriture déjà découvertes. Cette action contribue à renforcer le chemin de phéromone, puisque la nouvelle fourmi répandra les phéromones à son tour. Dans une certaine mesure, l'environnement devient une sorte de dépôt

distribué d'informations contextuelles, contenant l'information sur tous les chemins vers les sources de nourriture découvertes.

La tendance naturelle des phéromones à s'évaporer (si elle n'est pas renforcée) permet au réseau de phéromones de rester à jour et de s'adapter aux conditions changeantes : lorsque certaines fourmis découvrent un chemin plus court vers la nourriture, les chemins plus longs ont tendance à être abandonnés et à disparaître. De manière analogue, lorsqu'une source de nourriture est éteinte, la piste de phéromones correspondante disparaît car elle n'est plus renforcée [37]. Malgré sa simplicité, l'interaction à base de phéromones présente plusieurs caractéristiques qui la rendent adaptée à une variété d'applications distribuées et omniprésentes :

- Elle découple complètement les interactions des agents (c'est-à-dire des fourmis), qui se produisent indirectement via l'intermédiaire des phéromones. C'est une caractéristique très souhaitable dans les scénarios ouverts et dynamiques où les agents ne se connaissent pas à l'avance et peuvent aller et venir à tout moment.
- Elle soutient naturellement la prise de conscience du contexte spécifique à l'application, dans la mesure où les phéromones fournissent aux agents une représentation expressive de leur environnement opérationnel (par exemple, les phéromones fournissent une représentation de l'environnement en termes de chemins conduisant à des sources alimentaires).
- Elle soutient naturellement l'adaptation des activités, dans la mesure où les phéromones représentent des informations contextuelles qui, lorsqu'elles ne sont plus mises à jour, ont tendance à disparaître.
- Les algorithmes sous-jacents à l'interaction basée sur la phéromone sont simples et impliquent uniquement des interactions locales (chaque fourmi dépose localement et suit des phéromones sans subir le fardeau d'être impliqué dans une tâche distribuée).

2.2 Estimation du déplacement à l'aide d'étiquettes RFID

Les tags RFID peuvent être utilisés pour détecter le mouvement de l'utilisateur. Fondamentalement, la différence dans les étiquettes d'emplacement détectées à deux instants différents indique à l'agent que l'utilisateur se déplace. Plus formellement, soit $L(t)$ l'ensemble des étiquettes de localisation détectées à l'instant t . Il est facile de voir qu'un agent peut déduire (voir figure 1) que l'utilisateur se déplace lorsque $L(t) \neq L(t-1)$. Il vaut la peine de remarquer que ce type de service nécessite un lecteur RFID avec une gamme de lecture plutôt

petite. Si, par exemple, le lecteur pouvait lire des balises dans un rayon de 100 m, la localisation serait extrêmement grossière.

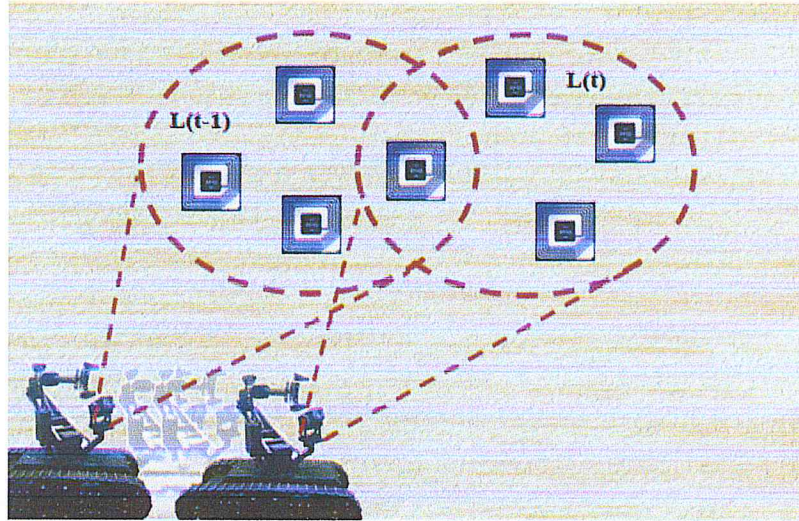


Figure 22 : Déplacement à l'aide des étiquettes RFID

2.3 Déploiement des phéromones avec tags RFID

Les phéromones sont créées au moyen de structures de données stockées dans des tags RFID. En d'autres termes, les étiquettes RFID dans l'environnement agissent comme une sorte de mémoire distribuée qui peut être utilisée pour stocker les phéromones et pour créer des pistes de phéromone (voir figure 2).

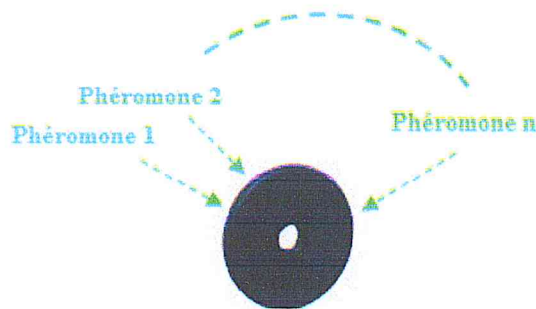


Figure 23 : Chaque tag peut contenir un certain nombre de phéromones

2.3.1 Déploiement de phéromones

Le déploiement de phéromones sur les tags RFID distribués dans un environnement s'effectue via un agent logiciel fonctionnant sur un ordinateur portable et qui contrôle le lecteur RFID associé. Chaque fois qu'il est demandé de commencer à diffuser une phéromone O (sous forme d'une structure de données consistant en une identification de phéromone et dans des informations supplémentaires détaillées dans ce qui suit), l'agent écrira O dans les

étiquettes d'emplacement dans la gamme de lecture, c'est à-dire l'agent écrira O dans tous les $L(t)-L(t-1)$ à mesure qu'il se déplace dans l'environnement (voir figure 3). Ce processus crée des pistes de phéromones numériques réparties sur les étiquettes d'emplacement que l'agent croise lors de la propagation de la phéromone. Une piste de phéromone constituée uniquement de l'ID de phéromone n'est pas très utile. En effet, les agents qui se succèdent des chemins de phéromones pour atteindre l'emplacement où les agents qui ont initialement déposé la phéromone ont été dirigés (ou, au contraire, pour atteindre l'emplacement d'origine).

Malheureusement, un agent qui traverse un chemin (constituée uniquement de l'ID) à un instant donné ne pourra choisir dans quelle direction aller. Ainsi, la structure de données de chaque phéromone O comprend également un compteur de sauts $C(O)$ associé à O ; lorsqu'il est chargé de répandre une phéromone O , l'agent initialise un compteur de sauts à 0. Chaque fois qu'un mouvement est détecté ($L(t) \neq L(t-1)$), l'agent lit la valeur actuelle de $C(O)$ dans $L(t)$.

Si les étiquettes appartenant à $L(t)$ n'ont pas de O , l'agent stocke la phéromone avec la valeur du saut ; s'ils ont un $C(O)$ inférieur au saut, l'agent écrase la valeur du $C(O)$ par la valeur du saut. Ensuite, il incrémente par 1, le résultat est une piste de phéromones avec un compteur de sauts toujours croissant. Il est utile de remarquer que le fait d'écraser le $C(O)$ inférieur crée des raccourcis dans le chemin de phéromone.

```

hop = 0;
while(true) {
  if(L(t) != L(t-1) {
    new = read(C(O));
    if(new == null || new < hop)
      write(C(O)=hop);
    else
      hop = new;
    hop++;
  }
}

```

Algorithme de propagation de phéromone proposé par *Mamei et Zambonelli* [36]

L'organisation globale de la mémoire des tags de localisation est illustrée dans la Figure 4. Nous avons en premier lieu l'identifiant unique immuable ; puis, les valeurs-clés décrivant des informations sur l'objet ou l'emplacement des tags, un bit *TagType* (utilisé pour indiquer si la balise est une étiquette d'emplacement), et un *Index* à 7 bits spécifie combien de phéromones sont actuellement stockées dans le reste de la mémoire.

Pour chaque phéromone, des emplacements de 3 octets sont attribués. Le premier octet code l'ID de phéromone, le second code le compteur C (saut) dans les 7 premiers bits, tandis que le dernier bit *Diff* spécifie comment propager la phéromone (comme décrit dans la

prochaine sous-section). En outre, le troisième byte *slot* stocke un *timestamp*, représentant l'heure à laquelle la phéromone a été écrite. Ce horodatage est utilisé pour supporter l'évaporation de la phéromone.

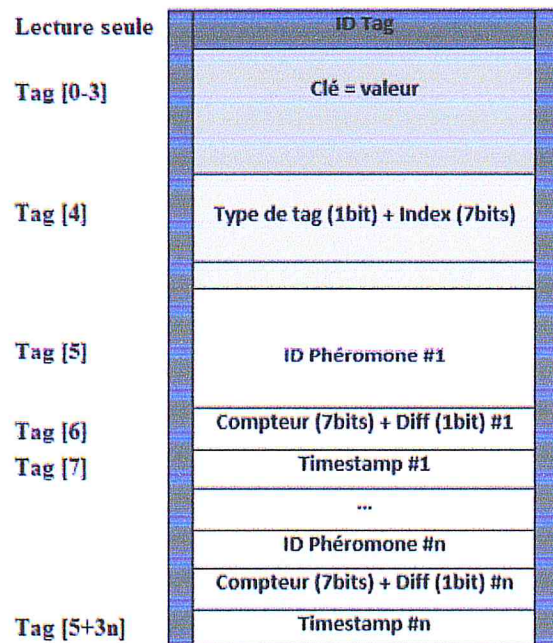


Figure 25 : Organisation de la mémoire d'un tag RFID proposé par *Mamei et Zambonelli* [36]

2.3.2 Processus d'évaporation progressive des phéromones

Chaque phéromone est créée avec une valeur d'horodatage associée $T(O)$ représentant l'heure à laquelle la phéromone O a été stockée. Pour coder le temps dans la limite d'un seul octet, il faut diviser le temps quotidien en 48 morceaux (1 morceau=30 minutes). Cela permet de ne pas déborder l'horodatage dans les 5 jours suivant l'utilisation. Lorsque les phéromones sont pro-activement (au lieu de parasite) déployées, la valeur de l'horodatage est réglée sur l'heure actuelle comme le prévoit l'horloge du robot. Cela représente naturellement le fait que la phéromone décrit une information à jour. Au contraire, lorsque les phéromones sont parasitées, l'horodatage reste le même que celui de la phéromone originale. Car lorsqu'un agent déploie la piste parasite, il n'a pas d'informations plus récentes et ne fait que repropager des données anciennes.

Après avoir lu une étiquette, un agent vérifie pour chaque phéromone O qu'il lit, si le *timestamp* associé $T(O)$ est en fonction de l'heure locale de l'agent. Il le compare ensuite à un certain seuil τ ; s'il est supérieur, l'agent supprime cette phéromone de la balise. Ce type d'évaporation de la phéromone pourrait conduire à deux avantages clés :

- Étant donné que l'espace de données dans les étiquettes RFID est très limité, il est utile d'avoir un mécanisme qui tente d'exploiter la mémoire au mieux ; par exemple, en supprimant les pistes de phéromone (et en libérant les emplacements de mémoire associés dans les balises) qui existent depuis longtemps et qui ne sont peut-être plus utilisées.
- Si un agent ne porte pas son PDA ou s'il a été éteint, il est possible que certaines actions soient entreprises sans étaler les pistes de phéromone correspondantes. Cela fait que les pistes anciennes de phéromone sont éventuellement périmées et finalement corrompues.

Dans ce contexte, il est fondamental de concevoir un mécanisme pour renforcer les phéromones pertinentes pour ne pas les laisser s'évaporer. À cet égard, un agent répandant la phéromone O activement, écrasera les phéromones ayant un $T(O)$ ancien. À partir de ces considérations, il devrait être clair que le seuil τ doit être réglé pour chaque application car le délai après lequel les phéromones sont jugées inutiles ou éventuellement corrompues dépend du contexte et de la tâche d'application donnés. Du point de vue des utilisateurs/agents, le seuil optimal dépend fortement de la recherche d'un objet récemment déplacé (auquel cas ils profiteraient d'un faible seuil d'évaporation) ou d'un objet qui n'a pas bougé depuis longtemps (auquel cas il y aurait des chances de trouver un chemin de phéromones conduisant à l'objet que si le seuil d'évaporation est assez élevé).

3. Déploiement de l'infrastructure RFID

La détection et l'identification des tags nécessite de bien préparer l'installation des lecteurs sur les robots comme une première étape, tel que chaque robot soit équipé d'un lecteur RFID. Chaque lecteur devra également être équipé de quatre antennes afin de couvrir 360° . La deuxième étape consiste à programmer les lecteurs de telle sorte qu'ils soient capables de détecter, et d'identifier tous les tags situés dans leurs voisinages. Nous considérons que les tags RFID sont de minuscules émetteurs-récepteurs sans fil qui peuvent être attachés à des objets telles qu'une table ou une chaise. Les étiquettes peuvent être placées sur des robots, des opérateurs humains, des murs, le sol, etc.).

Dans notre approche, nous considérons deux types de tags : des tags d'objets et des tags de localisation. Les tags d'objets sont ceux qui peuvent changer leurs positions ; tandis

que les tags de localisation sont les tags qui ont une position fixe et qui peuvent être utilisés comme des balises de localisation.

Les étiquettes RFID ont aujourd'hui un très faible coût, et ne présentent pas de problèmes d'épuisement. Chacune d'elles est marquée d'un identifiant unique et munie d'une minuscule mémoire (jusqu'à 1 Kb) permettant de stocker des données sous forme d'un ensemble d'octets ce qui rend la possibilité d'avoir un environnement riche en tags.

Dans la figure 5, un exemple de robot disponible au sein de la DPR auquel on peut installer un lecteur RFID IFM avec ces antennes.

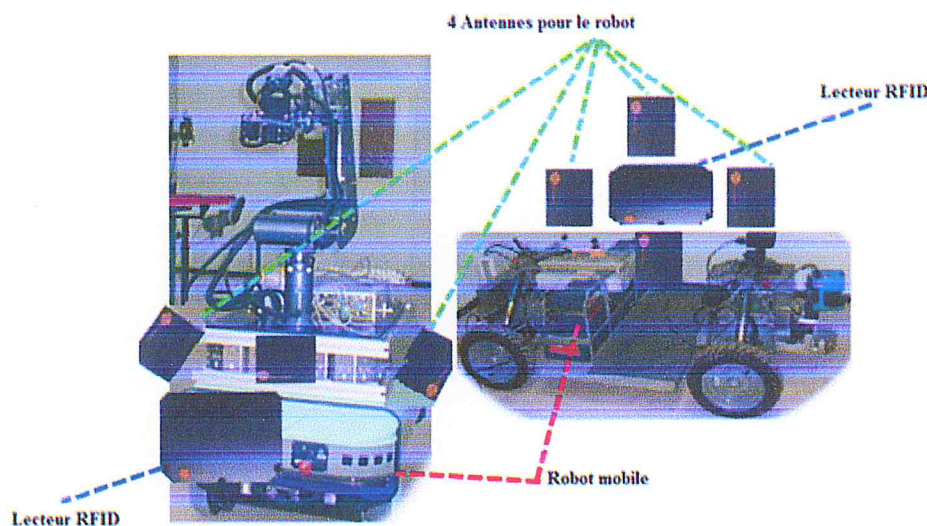


Figure 26 : Exemples de robots mobiles disponibles dans la division robotique

4. Adaptation de la structure des tags

Pour qu'un robot ait la possibilité de suivre une phéromone donnée, propagée préalablement par un autre robot, nous avons pensé à sauvegarder les positions de ce dernier dans la structure de phéromone lors de la propagation (figure 6). Ainsi, le premier robot à la recherche de cette phéromone pourra retracer les positions enregistrées (prochaines et précédentes) dans le but de retrouver l'objet recherché.

ID_Tag
Nbr_pheromone
ID_Pheromone #1
Saut_Pheromone #1
Timestamp_Pheromone #1
Position1 #1
Position2 #1

...
ID_Pheromone #n
Saut_Pheromone #n
Timestamp_Pheromone #n
Position1 #n
Position2 #n

Figure 27 : Adaptation de la structure des tags

5. Description de l'environnement de travail

Notre environnement consiste en un entrepôt logistique de stockage où l'on peut trouver des étagères de rangement, des racks, des armoires de stockage et des espaces au sol comme on peut trouver tout type de produits. La figure 7 illustre ce type d'entrepôts.



Figure 28 : Exemple d'un entrepôt de stockage

Autrement dit, nous sommes dans un environnement dynamique où les tags RFID sont non seulement déployés au sol, sur les murs ou sur des objets fixes (tag de localisation) ; mais aussi sur des objets d'intérêt où chaque tag indique le type ou le modèle d'objet pour aider à retracer l'endroit de stockage de cette quantité d'objet (tag d'objet).

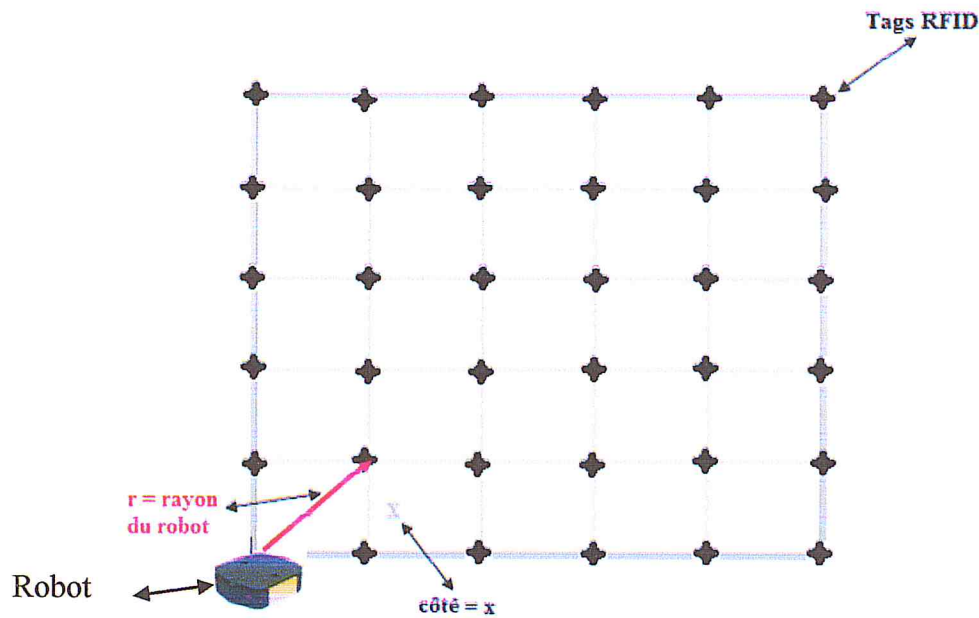


Figure 29 : Découpage de l'environnement sous forme de matrice.

Pour bien étaler les tags de localisation dans notre environnement, nous avons proposé de découper notre environnement sous forme de matrices tel que le rayon du champ du robot doit être supérieur ou égal au diamètres des carrés, comme illustré sur la figure 8. Donc, la condition est que : $r > x\sqrt{2}$.

6. Interaction entre les agents du système via une technique de stigmergie

Après avoir détaillé l'approche adaptée, nous décrivons dans cette section les différents comportements des agents (i.e. robots) dans notre environnement sous forme d'algorithmes. Le chemin emprunté par l'agent lors du premier déplacement pour mettre un objet à son endroit n'est pas forcément le plus court chemin ; le robot doit donc optimiser son chemin à chaque fois qu'il en trouve un meilleur. Les algorithmes développés sont décrits dans ce qui suit.

6.1 Propagation de phéromones

Lors du déplacement du robot, il y a un changement dans la liste des tags de localisation, c'est-à-dire $L(t) \neq L(t-1)$. Si la phéromone associée à l'objet o appartient à la liste des tags d'objets à deux instants différents, on peut déduire que le robot est en train de déplacer l'objet o .

```

Algorithme Propager_Phéromone(phéromone  $o$ ) ;
Début
  Saut  $\leftarrow 0$ ;
  Tant que (Navigation Robot) Faire
    Si (( $L(t) \neq L(t-1)$ ) && ( $o \in O(t)$  &&  $o \in O(t-1)$ )) //  $O(t)$  la liste de TAG objet
    d'intérêt détectés à un instant  $t$  et  $L(t)$  présente la liste des tags de localisation
    à un instant  $t$ 
    Alors
      Si(Saut==0) Alors
        Position_Actuelle  $\leftarrow$  Récupérer Position Robot() ;
        Ecrire_Pheromone( $o$ , saut,timestamp, Position_Actuelle, Position_Actuelle);
        Position_précédente  $\leftarrow$  Position_Actuelle ;
        saut++;
      FinSi
      Pour chaque TAG  $\in L(t)$  Faire
        Position_Actuelle  $\leftarrow$  Récupérer Position Robot() ;
        Ecrire_Pheromone ( $o$ , saut,timestamp, Position_Actuelle , Position_précédente);
        Position_précédente  $\leftarrow$  Position_Actuelle ;
        saut++;
      Fin Si;
    fait
  Fin Si;
Fait
Fin

```

Algorithme de Propagation de phéromones

6.2 Optimisation du chemin

Si un robot retourne à partir d'un objet d'intérêt à son emplacement initial, il peut optimiser et choisir le meilleur chemin pour revenir à cette zone en laissant une autre phéromone ; celle-là nous donnera un chemin meilleur la prochaine fois.

```

Algorithme Optimiser_Chemin(Phéromone o);
Début
  Tant que (posPrécédente != posActuelle) Faire
    Si (min (C(O)) < C(o)) // C(O) correspond à la liste des compteurs de saut
      associer à la phéromone
      // Propager la phéromone avec un saut descendant
    PropagerInv(o, C(o)- 1,timestamp,Postion_Actuelle(min(C(t)),Postion_Actuelle(C(t))));
    DéplacerRobot(positionPrécédente(min C(o)));
  Sinon
    DéplacerRobot(positionPrécédente(C(o)));
    Renforcer_Phéromone(o) ;
  Fin si ;
  Fin tant que
Fin
    
```

Algorithme pour l'optimisation du chemin

La figure 9 ci-dessus représente l'existence de deux chemins de deux objets différents.

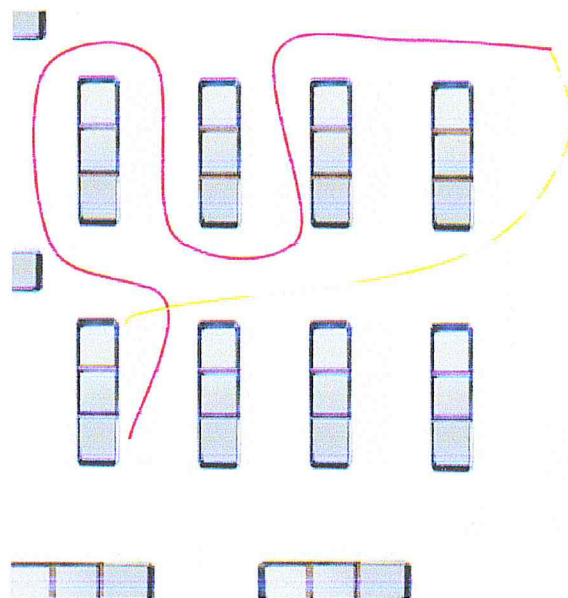


Figure 30 : Existence de deux chemins dans l'environnement.

Dans la figure 10, l'agent en revenant à partir de l'objet (chemin rouge), il trouve un meilleur chemin vers la position initiale, donc il suit ce chemin et propage une autre fois la phéromone mais en décrémentant le saut.

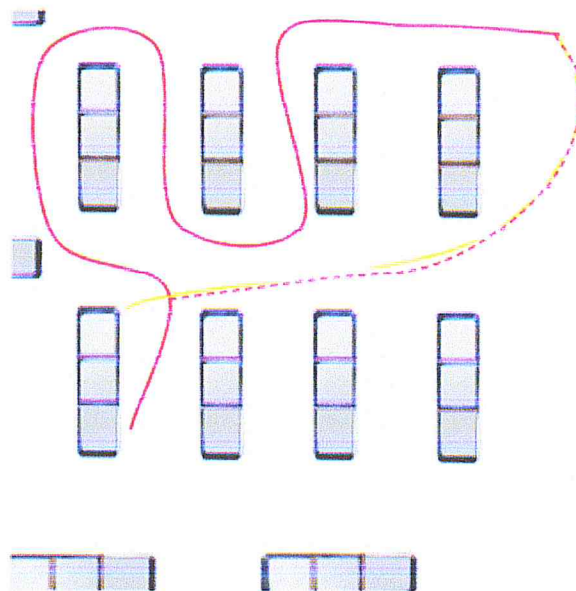


Figure 31 : Chemin optimisé

6.3 Accéder à un objet

Lorsque cette tâche est demandée à un agent, ce dernier cherche dans quelle zone se trouve l'objet ; puis, il commence à suivre la phéromone associée à ce type d'objet en cherchant le meilleur chemin à chaque fois.

Algorithme Accéder_Objet(Phéromone o);

Début

Accéder_a_la_zone_de_stokage();

*Tant que (o !∈ O(t)) // O(t) liste des TAG d'objet d'intérêt détecter a
l'instant t*

*// s'il rencontre plusieurs TAGs qui contient la même phéromone o il choisit la
position associer au compteur de saut le plus grand*

DeplacerRobot(posActuelle(max C(o))) ;

Renforcer_Phéromone(o) ;

Fin tant que

Fin

Algorithme pour accéder à un objet

7. Conception de l'outil de simulation

La validation de notre approche dans un environnement physique étant très compliquée ; nous avons décidé alors de développer un outil de simulation afin de faciliter les tests. Nous présentons ci-dessous le diagramme de classes (voir figure 11) du simulateur mis en œuvre. Plus de détails sur cet outil sont donnés dans le chapitre suivant.

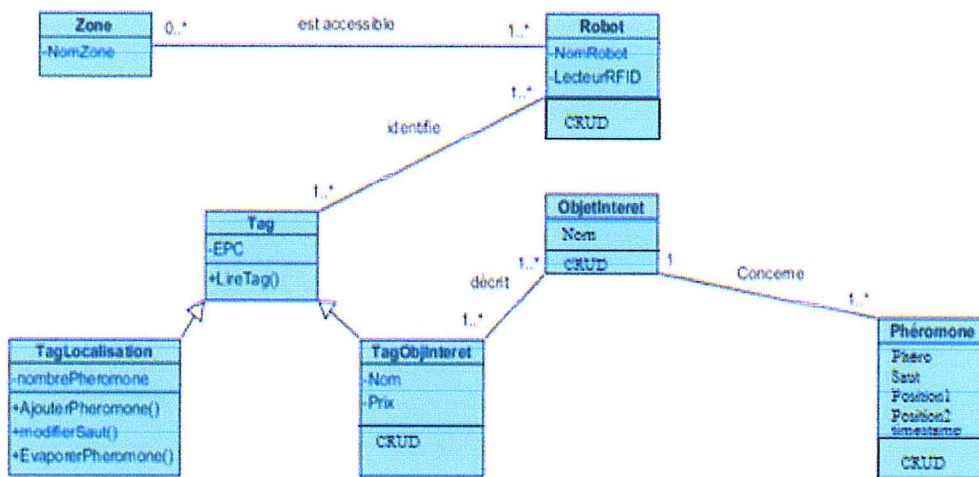


Figure 32 : Diagramme de classes

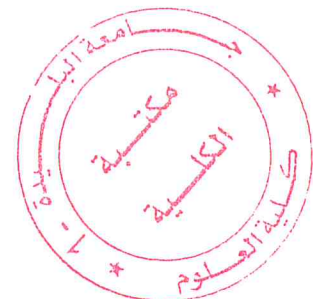
8. Conclusion

Dans ce chapitre, nous avons présenté l'approche proposée pour la localisation d'objets dans une infrastructure RFID. Il s'agit d'une approche inspirée des travaux de *Zambonelli* et *Mamei* [36] sur le comportement animal. Ensuite, nous avons proposé des algorithmes qui servent à la localisation et l'optimisation des chemins des objets d'intérêt. Enfin, nous avons abordé l'outil que nous avons mis en œuvre pour la simulation de notre infrastructure RFID et pour la validation de l'approche de localisation proposée.

Dans le chapitre suivant, nous présenterons les différentes réalisations mises en œuvre au cours de notre travail.

Chapitre IV

Tests et validation de l'approche proposée



1. Introduction

Dans les deux chapitres précédents, nous avons présenté les différentes fonctions développées pour le matériel IFM ainsi que la solution que nous avons proposée pour notre système. L'objectif de la première partie de dernier chapitre est de tester l'efficacité du fonctionnement du Reader DTE800 via la transmission Ethernet. Quant à la deuxième partie, son objectif est de simuler le comportement du robot qui exécute les différents algorithmes proposés via une interface graphique *Java*.

2. Environnement de développement

2.1 Langage de programmation Java

Java est un langage de programmation informatique orienté objet créé par *Sun Microsystems*. Il permet de créer des logiciels compatibles avec de (*Linux, Macintosh, Solaris*). *Java* donne aussi la possibilité de développer des programmes pour téléphones portables et PDA. Enfin, ce langage peut être utilisé sur internet pour des petites applications intégrées à la page web (applet) ou encore comme langage serveur (jsp). *Java* présente plusieurs caractéristiques, les plus importantes :

a) Orienté Objet

La conception orientée objet est une technique de programmation qui se concentre sur les données (les objets) et sur l'interaction des interfaces avec les objets.

b) Indépendant de l'architecture

Le bytecode généré n'est pas lié à un système d'exploitation particulier. De ce fait, il peut être interprété très facilement sur n'importe quel environnement disposant d'une JVM.

c) Multithreads

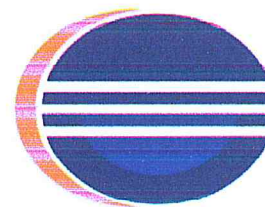
Un Thread est un flot d'instructions s'exécutant en concurrence avec d'autres threads dans un même processus.

2.2 IDE Eclipse

Eclipse est une plateforme open source basée sur *Java* qui permet à un développeur de logiciels de créer un environnement de développement personnalisé (IDE) à partir de composants de plugin construits par les membres d'Eclipse. Eclipse est géré et dirigé par le Consortium Eclipse.org.



Eclipse a débuté en 2001 lorsque IBM a fait don de trois millions de lignes de code à partir de ses outils *Java*. L'objectif initial d'Eclipse était de créer et de favoriser une communauté IDE open source qui compléterait la communauté qui entoure Apache. Bien que la plateforme Eclipse soit écrite en Java, elle prend en charge les plugins qui permettent aux développeurs de développer et de tester un code écrit dans d'autres langues.



3. Présentation et validation de la librairie d'utilisation du lecteur DTE800

Dans cette partie, nous allons tester le fonctionnement des différentes fonctions développées de lecture, d'écriture, etc.

3.1 Connexion avec le reader

La connexion avec le Reader se fait via des sockets en envoyant une trame spécifique et en recevant une autre trame contenant la réponse du reader.

3.1.1 Préparation du lecteur

La figure 1 ci-dessous représente notre lecteur RFID **DTE800** connecté via un câble *Ethernet*.

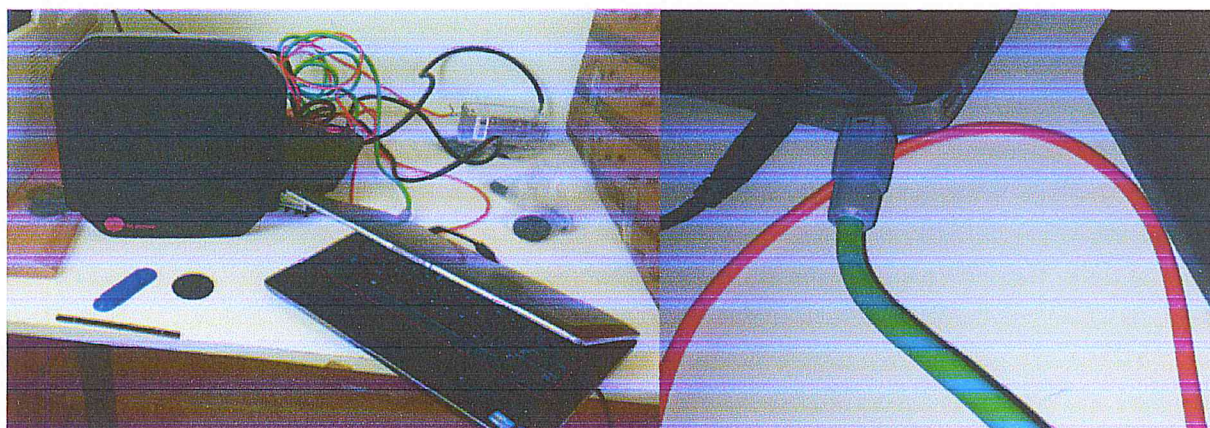


Figure 33 : Connexion via câble Ethernet

3.1.2 Antennes

Notre lecteur RFID peut utiliser jusqu'à 4 antennes connectées avec un câble coaxiale RG58, et l'antenne utilisée est de type **ATN830**. La figure 2 ci-dessous montre le câble de raccordement RG58 entre le Reader et l'antenne.



Figure 34 : Câble coaxiale RG58

3.1.3 Port de communication

La communication TCP/IP s'effectue toujours avec le port 4007 (voir figure 3 ci-dessous).

```
public Reader(String IPreader,int Port)
{

    try {
        Socket client = new Socket(IPreader, Port);
        IN = client.getInputStream();
        OUT = client.getOutputStream();
    } catch (IOException e) {
        e.printStackTrace();
    }
}
```

Figure 35 : Exemple d'utilisation de socket pour la connexion avec le reader.

4. Interface graphique

Dans les figures ci-dessous nous présentons quelques interfaces développées.

4.1 Interface de communication

Dans l'interface donnée par la figure 4 ci-dessous, une fois l'utilisateur accède à cette interface, il peut donner l'adresse IP du reader ; puis, il va choisir le numéro de port pour communiquer. De plus, nous avons introduit un message qui alerte l'utilisateur que son adresse IP doit être configurée sur son ordinateur.

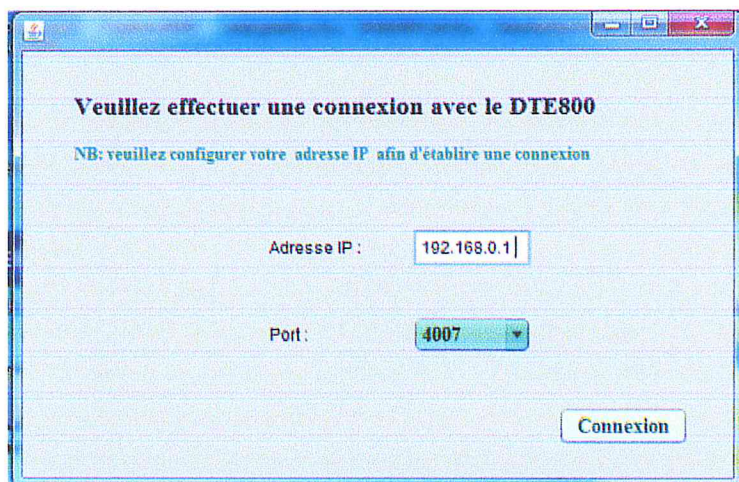


Figure 36 : Communication.

4.2 Interface de lecture

Pour choisir quel type de lecture effectuer, l'utilisateur peut choisir entre la lecture synchrone et la lecture asynchrone.

a) Lecture Synchrone des EPC

Dans ce mode de lecture, l'utilisateur lance l'opération de lecture (voir figure 5) ; une liste des tags détectés est affichée dans un tableau contenant le nombre de mots des EPC, le numéro d'antenne et la valeur RSSI si elle fonctionne sur le lecteur et l'identification EPC des tags détectés. La figure 5 illustre aussi le cas de la lecture asynchrone.

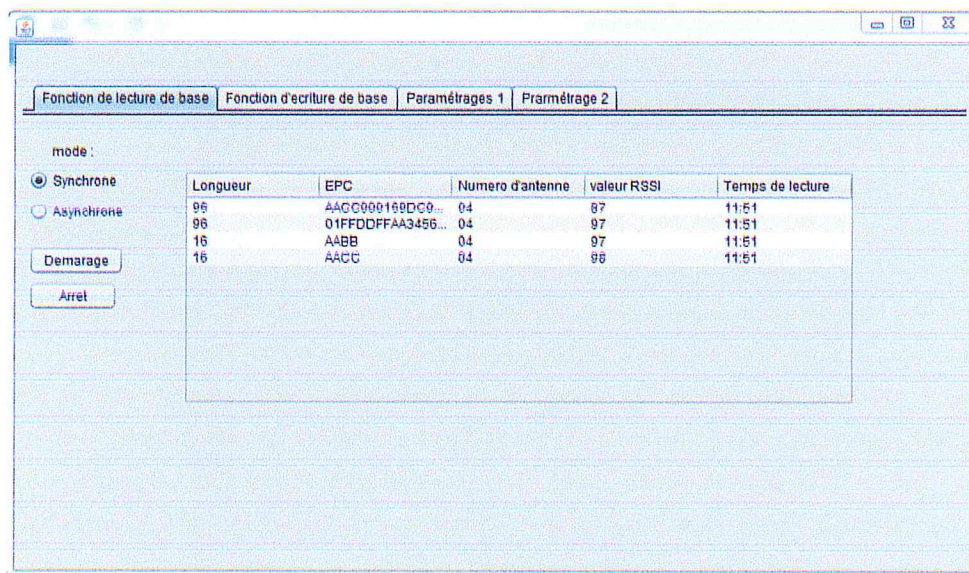


Figure 37 : Lecture Synchrone des EPC

b) Lecture des données utilisateur

Comme illustré dans la figure 6, le data User est la plus grande partie du mémoire des tags ; c'est là où on enregistre les phéromones.

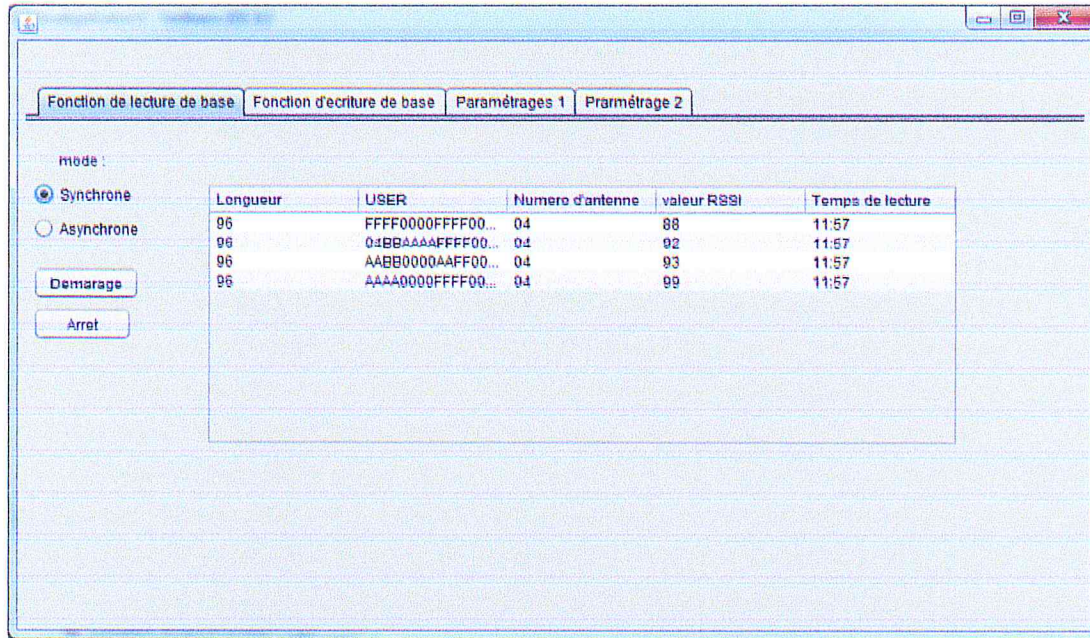


Figure 38 : Lecture des User Data

c) Lecture asynchrone

La figure suivante montre un exemple de lecture asynchrone.

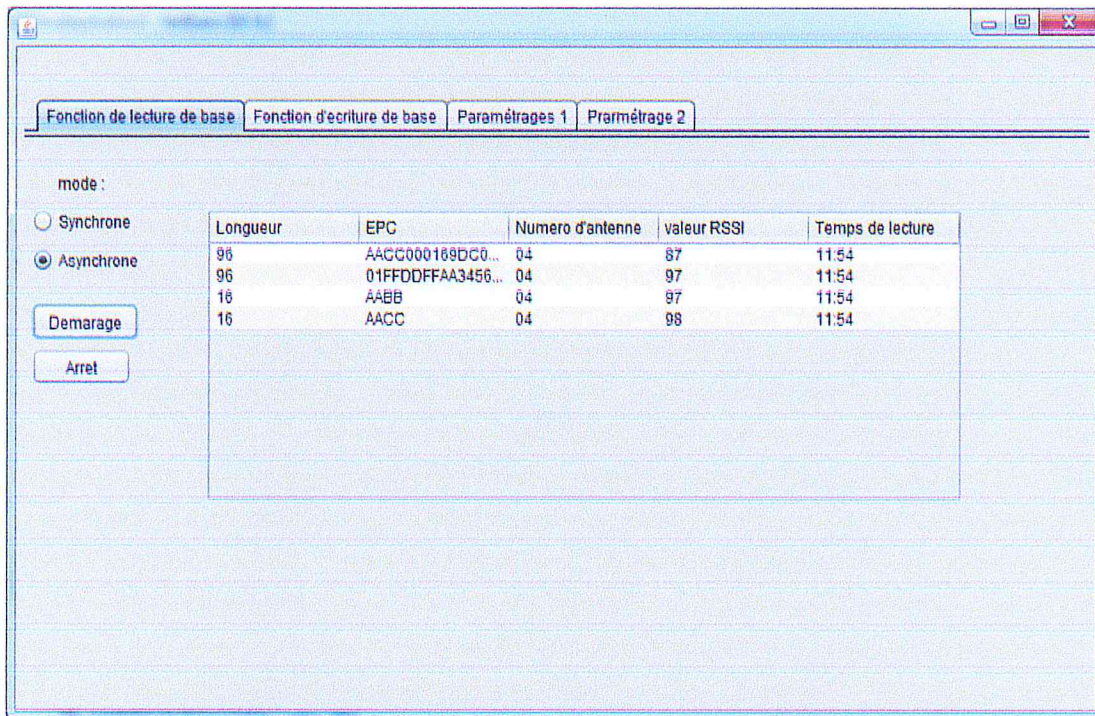


Figure 39 : Lecture Asynchrone des EPC.

4.3 Écriture dans les tags

a) Écriture des EPC

Comme présenté dans la figure 8 ci-dessous, pour effectuer l'écriture EPC dans les tags, l'utilisateur doit choisir le tag dont il souhaite modifier l'EPC. Puis, il écrit le mot de passe du tag pour accéder à cette mémoire et la modifier ; enfin, il fait entrer les données à écrire.

The screenshot shows a software window with a menu bar containing four tabs: 'Fonction de lecture de base', 'Fonction d'écriture de base', 'Paramétrages 1', and 'Paramétrage 2'. The 'Fonction d'écriture de base' tab is selected. The main area is titled 'Selectionner les tags'. It contains a 'Tag:' label followed by a dropdown menu showing 'Item 1' and an 'Actualiser' button. Below this are three input fields: 'Mot de passe:' with the value '00000000', 'Taille de données à écrire:' with the value '2', and 'Données à écrire:' with the value 'AABB'. An 'Ecrire EPC' button is located below the 'Données à écrire:' field. At the bottom of the window, there are labels for 'Resultat:' and 'Resultat R'.

Figure 40 : Écriture EPC

b) Écriture des données utilisateur

Dans ce type d'écriture (figure 9), l'utilisateur suit les mêmes étapes que l'écriture des EPC ; mais, il peut commencer du premier mot et non pas comme l'écriture des EPC où l'utilisateur commence du quatrième octet.

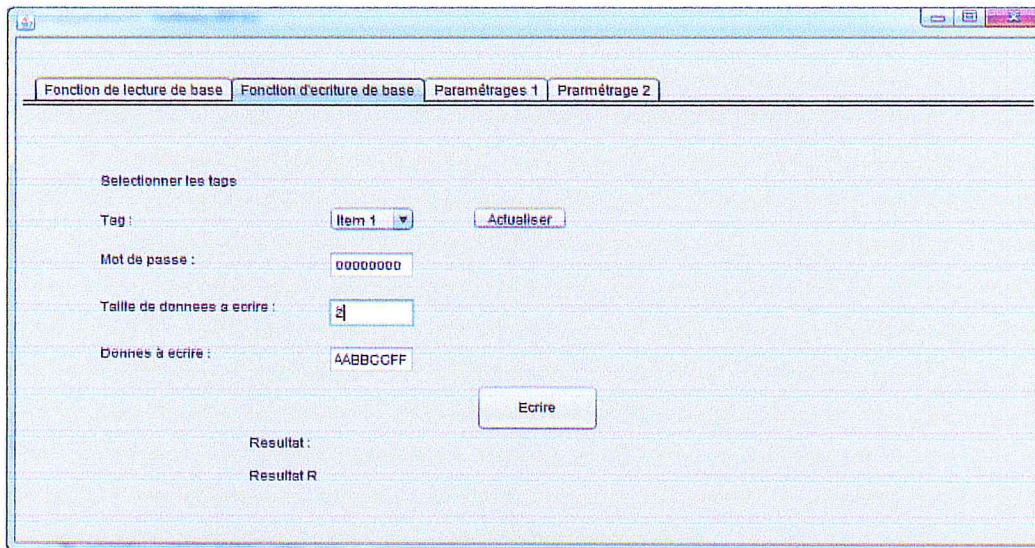


Figure 41 : Écriture des données utilisateur

4.4 Interfaces de paramétrage

Dans l'interface paramétrages 1 et paramétrages 2 (voir figure 10), l'utilisateur a le choix entre plusieurs tâches et fonctions existantes sur le matériel ; il peut aussi visualiser ou modifier les paramètres courante. Parmi ces fonctions :

a) **Get_Extended_Result_Flag**

On envoie cette commande pour récupérer le type de paramètres utilisé à cet instant sur le reader. Si le lecteur renvoie « 1 », il n'y a que le paramètre d'antenne qui fonctionne.

b) **Set_Extended_Result_Flag**

Avec le *SetExtendedResultFlag*, l'utilisateur peut changer un des paramètres du système. Par exemple, si l'utilisateur donne la valeur 3, le reader fonctionne avec les deux paramètres antenne et valeur RSSI.

c) **Get_Port_Power et Set_Port_Power**

Dans le *Get Port Power*, l'utilisateur doit choisir l'une des antennes entre 1 et 4 et il reçoit la puissance de ce port s'il n'y a pas de problème. L'antenne renvoie la puissance du port selon le gain de l'antenne et l'atténuation du câble. Le principe de *Set Port Power* est analogue.

d) **Get_Antenna_mode et Set_Antenna_mode**

Le *Get antenna mode* renvoie le mode utilisé dans l'antenne : soit mode synchrone ou mode asynchrone. Il permet de modifier le mode dans l'antenne en choisissant les antennes à configurer.

e) **Get_Cable_loss_and_Antenna_gain**

Avec cette fonction, on peut avoir le gain de l'antenne ainsi que le câble en dB.

f) Get_Modulation_Type

Permet d'avoir le type de modulation.

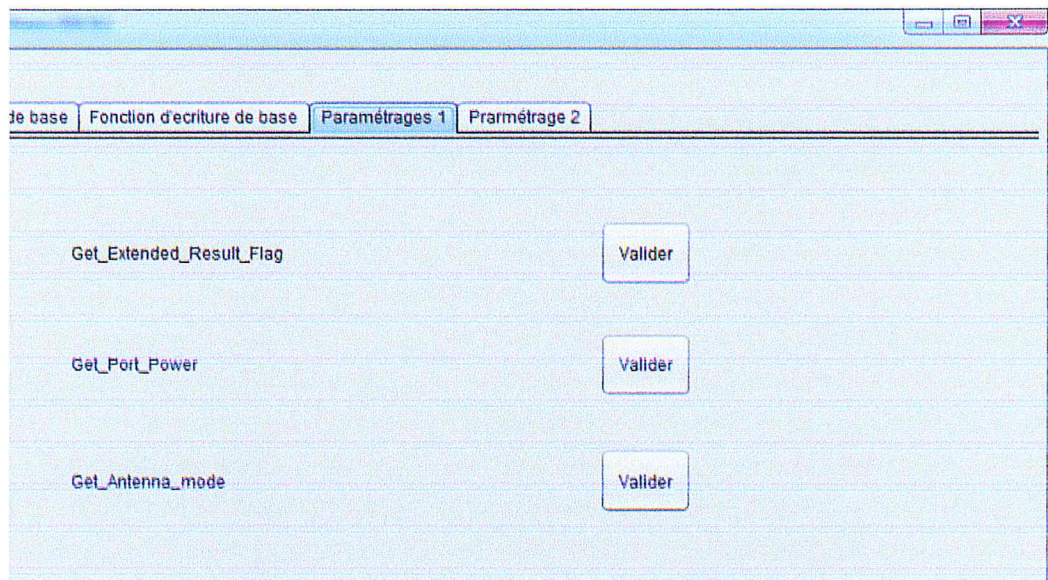


Figure 42 : Paramétrages

5 Validation de la librairie

Afin de valider notre librairie d'utilisation du lecteur DTE800, nous avons comparé les valeurs retournées par notre outil de lecture avec celles retournées par le logiciel fourni. Nous avons trouvé que les résultats étaient similaires.

6. Validation de l'algorithme de localisation

Afin de d'évaluer et de valider les algorithmes de localisation proposés, nous avons mis en œuvre un environnement de simulation. En effet, dans ce genre d'applications, il est important de valider le travail réalisé à travers des simulations avant de passer à la mise en œuvre sur une infrastructure RFID réelle. L'environnement réalisé permet de simuler le comportement des robots qui exécutent les algorithmes décrits dans le chapitre précédent via une interface graphique.

6.1 Simulation de l'environnement de travail

L'environnement de travail choisi consiste à simuler un robot dans un entrepôt où on retrouve des étagères et une zone d'entrée pour les robots (comme illustré dans la figure 11).

Il y a aussi les tags que le robot peut exploiter pour enregistrer et reconnaître les phéromones existantes afin de localiser les objets.

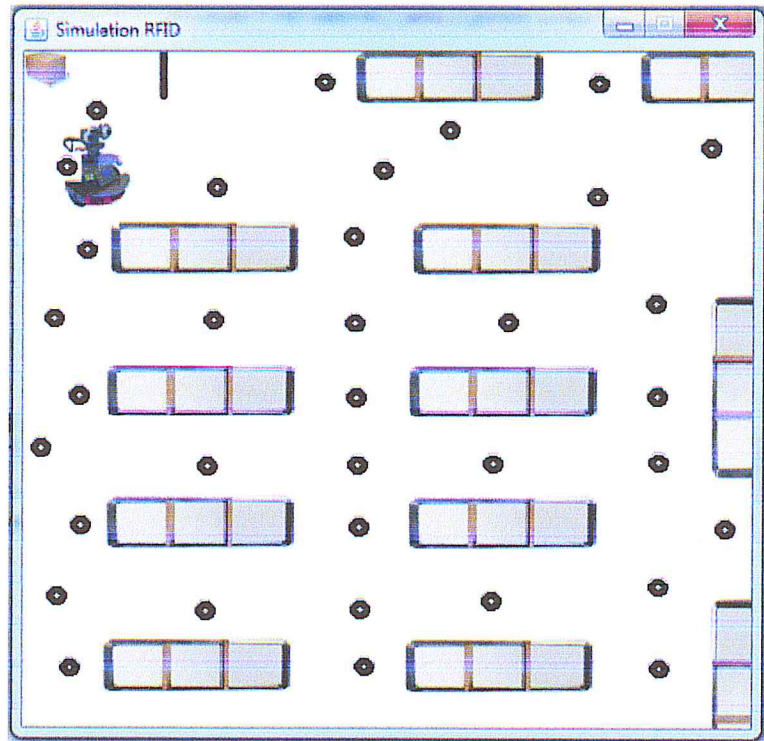


Figure 43 : Simulation de l'environnement de développement.

6.2 Simulation de déplacement d'objet

Chaque robot se déplace en aller/retour à la recherche d'un objet précis jusqu'à l'arrivée à l'emplacement de ce type de produit. Lors du premier déplacement, le robot propage des phéromones à chaque fois qu'il trouve un tag dans son champ jusqu'à l'endroit de stockage de ce produit (voir figure 12).

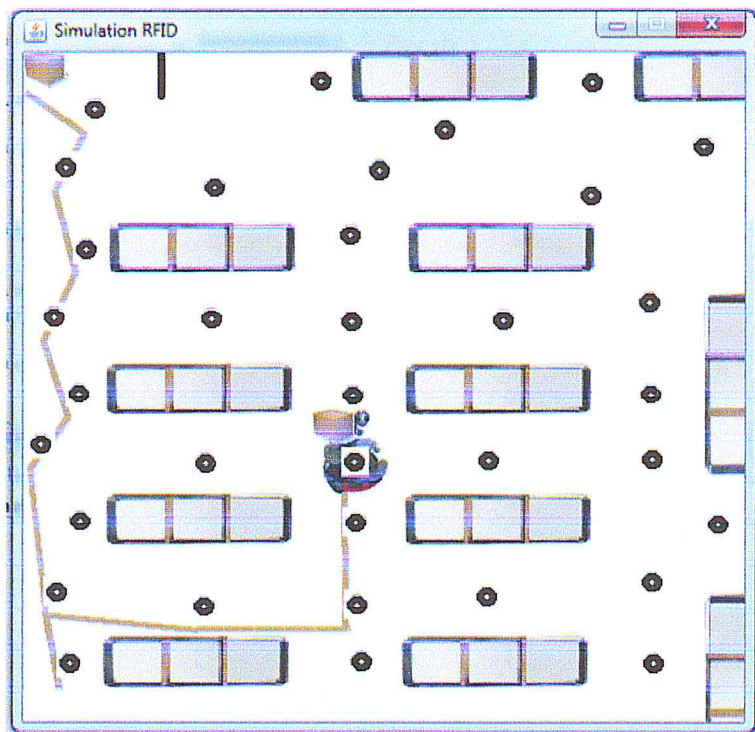


Figure 44 : Déplacement d'un objet.

6.3 Élimination des boucles

En revenant vers son emplacement initial, le robot retrace son chemin mais en éliminant cette fois-ci les boucles de son premier déplacement. L'évitement des boucles se fait en appliquant l'algorithme de *Mamei et Zamboneilli* [36] tel que montré par la figure 13.

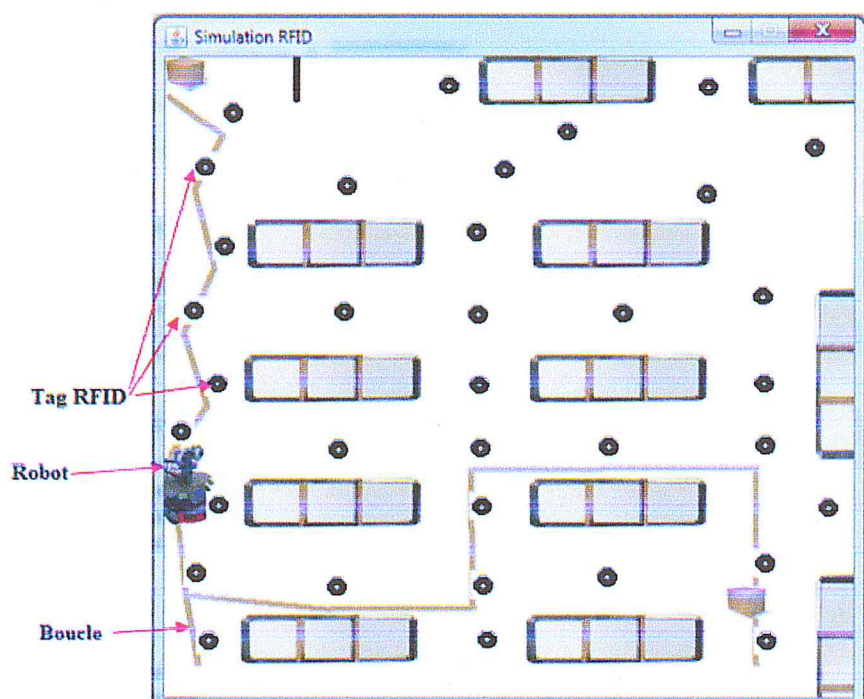


Figure 45 : Élimination des boucles

6.4 Déplacement de plusieurs objets

Plusieurs objets sont stockés dans l'entrepôt. Ils sont déplacés soit par le même robot soit par d'autres robots. À cet effet, chaque tag contient une ou plusieurs phéromones ; à chaque fois que le robot passe avec un nouveau produit, le nombre de phéromones augmente dans le tag (voir figure 14).

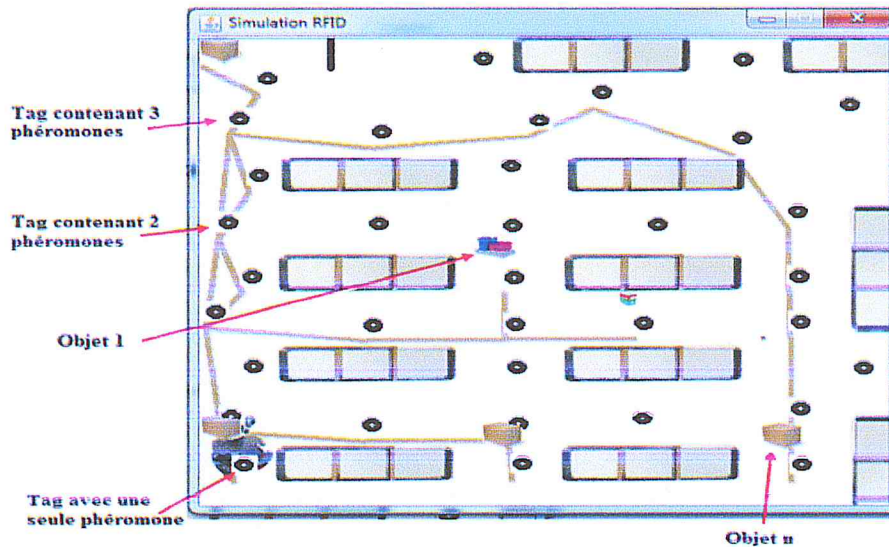


Figure 46 : Déplacement de plusieurs objets

6.5 Identification du meilleur chemin

Comme illustré dans la figure 15, chaque robot qui déplace un objet, le dépose et revient vers le point initial en cherchant le plus court chemin à chaque fois ; puis, il propage une autre fois la phéromone de l'objet dans le nouveau chemin trouvé.

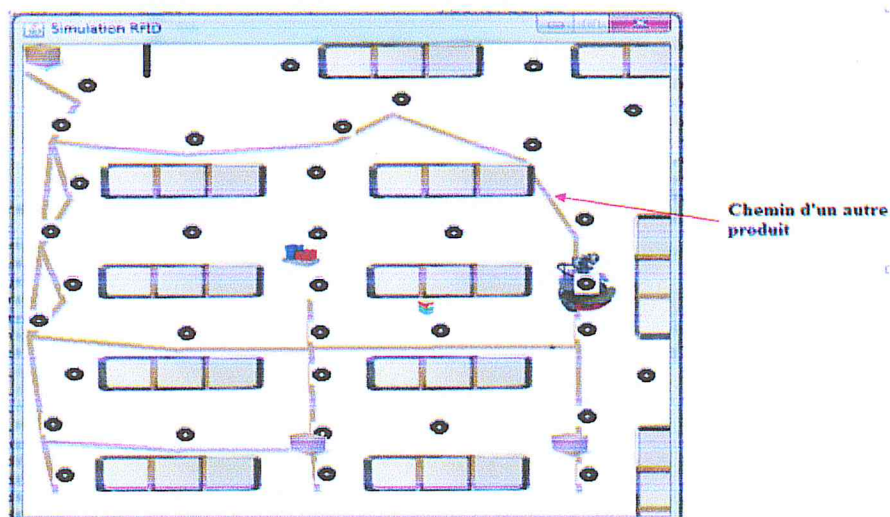


Figure 47 : Plus court chemin

7. Discussion des résultats obtenus

Nous avons réalisé une série de scénarios de simulations avec 5 objets. Les robots et les objets sont placés initialement dans un point de départ dans leur environnement. Nous avons effectué un certain nombre de tests (voir figure 16), à la fois en adoptant un cadre de simulation permettant de détecter et d'identifier tous les tags de localisation situés dans les voisinages des robots qui représentent des emplacements.

Le tableau ci-dessous récapitule les résultats obtenus pour la première expérience. Au cours de ce scénario, qui simule un seul robot mobile qui déplace des objets et revient vers son emplacement initial en appliquant l'algorithme de *Mamei et Zamboneilli*. Nous avons remarqué que le nombre d'emplacements visités par rapport au précédent (lors du déplacement de l'objet) a diminué.

Objet	Objet 1	Objet 2	Objet 3	Objet 4	Objet 5
Nombre d'emplacements visités lors du déplacement de l'objet	17	12	9	7	14
Nombre d'emplacements visités lors du retour vers l'emplacement initial	15	8	9	7	11

Tableau 1 : Tableau montrant le nombre de déplacement des robots sans et avec l'utilisation de l'approche de *Mamei et Zamboneilli 2007*.

Le deuxième tableau représente les résultats obtenus pour la deuxième expérience. Au cours de ce scénario qui simule un robot mobile qui déplace des objets et revient vers son emplacement initial en appliquant l'algorithme que nous avons proposé pour améliorer le chemin de chaque objet à chaque fois, nous avons remarqué que le nombre d'emplacements visités par le robot a énormément diminué par rapport aux résultats obtenus dans la première expérience.

Objet	Objet 1	Objet 2	Objet 3	Objet 4	Objet 5
Nombre d'emplacements visités lors du déplacement de l'objet	17	12	9	7	14
Nombre d'emplacements visités lors du retour vers l'emplacement initial	11	7	7	7	11

Tableau 2 : Tableau montrant la différence entre le nombre de déplacement des robots et d'autres qui utilisent l'approche proposée

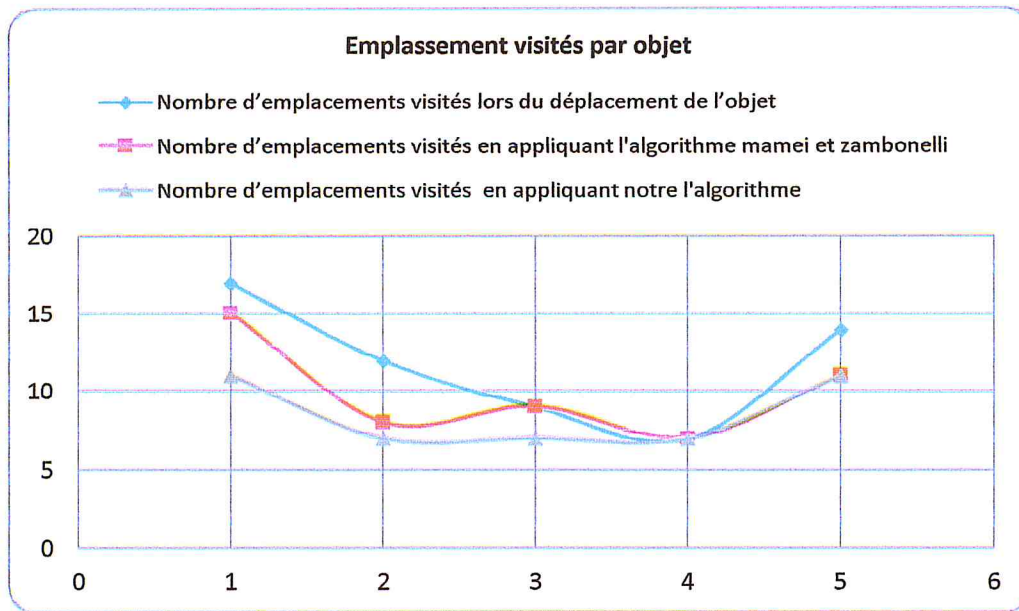


Figure 48 : Comparaison des emplacements visités

8. Conclusion

Tout au long de ce chapitre, nous avons testé et montré le fonctionnement du matériel IFM en effectuant des tests sur les tags disponibles dans la DPR notamment des tests de lecture et d'écriture et même quelques fonctions de réglage pour ce matériel (DTE800) afin de le préparer à fonctionner sur les robots.

Cependant, les robots utilisant un tel matériel de détection et de localisation d'objet nécessitent des algorithmes pour effectuer la localisation et la navigation. À cet effet, nous avons testé l'approche proposée avec différents scénarios pour extraire le plus court chemin amenant les robots d'une position initiale à une position finale.

Conclusion générale et perspectives

La détection, l'identification et la localisation d'objets pendant le déplacement d'un robot mobile est une tâche très importante dans la robotique mobile. Elle permet la récupération des différentes informations liées aux objets situés dans l'environnement afin de les utiliser dans différents contextes. À cet effet, de nombreux systèmes et technologies existent pour répondre à cette problématique ; le système d'identification par radiofréquence (RFID) s'impose comme un système très prometteur dans le domaine de l'identification et la localisation.

Dans ce travail, nous avons commencé par la présentation des systèmes cyber-physiques, leurs applications avec la technologie RFID, notamment dans le domaine de la robotique. Nous avons décrit différentes technologies d'identification et de localisation et nous avons cité différentes applications d'utilisation des systèmes RFID pour affirmer son efficacité et ses avantages. Ensuite, nous avons décrit le matériel qui nous a été confié, en particulier le lecteur RFID DTE800 et les fonctions de base développées. Après, nous avons détaillé l'approche adoptée pour la localisation d'objets dans une infrastructure RFID ainsi que son adaptation à notre contexte.

L'objectif de notre travail était de déployer une infrastructure cyber-physiques RFID dans un environnement pouvant contenir des robots et des objets quelconques (mur, salles, etc.). Ensuite, il s'agit de localiser des objets à manipuler par les robots. Pour ce faire, nous avons développé les fonctions de base pour le lecteur RFID DTE 800 à savoir, la lecture et l'écriture des/sur les étiquettes RFID. Nous avons par la suite simulé le système de localisation en exploitant une technique de stigmergie par un outil de simulation que nous avons développé en Java : interaction omniprésente des phéromones avec des tags RFID. Enfin, nous avons effectué des scénarios de test pour la validation des algorithmes proposés pour garder trace des différentes pistes de phéromones qui mènent aux objets d'intérêt. Ces phéromones ont été, exploitées afin d'optimiser (minimiser) le nombre de sauts séparant l'objet recherché du robot et le déplacer à son emplacement final.

Durant l'implémentation de notre solution, nous avons rencontré plusieurs problèmes. En premiers lieu, des problèmes liés au manque d'informations concernent le matériel RFID exploité ; les documents fournis avec le matériel ne donnent pas une description détaillée des commandes. Et l'un des inconvénients majeurs liés à cette approche est la saturation des tags. Un autre problème est lié à la simulation ; nous n'avons pas trouvé une solution qui permet de simuler le contexte de notre système RFID. Ceci nous a obligé à développer notre propre outil de simulation en Java où nous avons implémenté plusieurs scénarios qui ont été accompagnés par un ensemble de tests pour la validation de la solution proposée.

À l'issue du travail, différentes pistes formant les directions futures à cette recherche sont envisagées. Nous citons, dans ce qui suit, les plus importantes :

- Il sera intéressant, que notre travail soit appliqué sur un matériel RFID dans des environnements équipés d'une infrastructure RFID réelle où évoluent une équipe de robots mobiles.
- Une autre direction de recherche envisagée à travers ces études consiste à l'application du système développé avec d'autres travaux : cartographie, le positionnement, etc.

Références bibliographiques

- [1] « Control Architectures of Robots 2015 », *CAR'15*. [en ligne]. <http://www.car-conference.fr/>[dernière consultation le 01 juin 2017].
- [2] « Les Etats-Unis misent gros sur la cyber-physique », *L'Atelier-BNP Paribas*. [en ligne]. (03 septembre 2008). <http://www.atelier.net/trends/articles/etats-unis-misent-gros-cyber-physique> [dernière consultation le 04 janvier 2017].
- [3] L. Chunag, "Object localization strategy for a mobile robot using RFID", Master's thesis in Computational Science and Engineering, Umea University, pp.5-65, July 4, 2012.
- [4] E. A. Lee, S. A. Sheshia, "Introduction to Embedded Systems – A Cyber-Physical Systems Approach", LeeSeshia.org, 2011.
- [5] Lee, E. (2008). Cyber Physical Systems: Design Challenges. 2008 11th IEEE International Symposium on Object and Component-Oriented Real-Time Distributed Computing (ISORC).
- [6] Wang, H., Liu, J. and Han, J. (2015). RS-CPS: A distributed architecture of robotic surveillance cyber-physical system in the nature environment. 2015 IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER).
- [7] C. Neuman, "Challenges in Security for Cyber-Physical Systems", DHS: S&T. Workshop on Future Directions in Cyber-physical Systems Security, Newark, NJ, USA, 22-24 July 2009.
- [8] A. Huebner, C. Facchi, M. Meye et H. Janicke, "RFID Systems from a Cyber-Physical Systems Perspective", 2014.
- [9] Laurent. B. (2005). Monter un réseau Wi-Fi. Consulter le 19 avril 2017 de <http://christian.caleca.free.fr/wifi/>
- [10] Baviskar, J.J.; Mulla, A.Y.; Pandit, S.K.; Naik, R.D.; Baviskar, A.J, "GPS Based Real time Emergency Aid System with Analysis of Latency in Satellite Communication," Communication Systems and Network Technologies (CSNT), 2013 International Conference on , vol., no., 7-9 April 2017.
- [11] <https://www.nxp.com/documents/brochure/75017587.pdf> Dernière consultation : Mars 2017.
- [12] Gabriel. D. S. (2014), NFC – Near Field Communication, consulter Avril 2017.

- [13] http://monge.univ-mlv.fr/~dr/XPOSE2010/NFC_sur_mobile/concl_av_inconv.html
dernière consultation : Avril 2017.
- [14] Michel. D juin 2011. Les RFID quelle belle idée [en ligne] repéré a
<https://www.youtube.com/watch?v=FyN2jrltgvg&t=1454s>.
- [15] MezianiFadia & MesliF.Zohra, « Implémentation et application de la technologie RFID », Université ABOU-BEKR BELKAID-TLEMCEM, Septembre 2006.
- [16] "GS 1. GS 1 Global traceability standard 1.1.0[S]", 2009.
- [17] Grand. M & Malard. E & Pierre. N. F & Santiago. D (2008). Application pour la gestion d'un boîtier RFID : Systèmes Embarqués, Bordeaux.
- [18] Radio Frequency Identification device technology, a factfile provided by the Institution of Engineering and Technology. www.theiet.org/factfiles.
- [19] T. Flor, W. Niess, G. Vogler. RFID: The integration of contactless identification technology and mobile computing. Croatia: 7th International Conference on Telecommunications. 2003: 619.
- [20] <http://www.nexess-solutions.com/fr/choisir-un-tag-rfid-pour-des-applications-industrielles/> dernière visite Avril 2017.
- [21] *Yvan. D & Smail. T (2017) La RFID une Technologie Clé au Service de l'Humanité : RFID a Key Technology for Humanity. URSI-France.*
- [22] John, K. and Mathew, T. (2016). Series SRR Loaded UHF RFID Tag. *Procedia Computer Science*, 93, pp.238-242.
- [23] <http://www.centrenational-rfid.com/classification-des-tags-rfid-article-19-fr-ruid-17/>
dernière visite en mai 2017.
- [24] Nicolas. S (2005), Les systèmes d'identification radio (RFID) : fonctionnement, applications et dangers. Yverdon-les-Bains.
- [25] Delphine Bechevet, « Contribution au développement de tag RFID UHF et Microondes sur matériaux plastiques », Institut National Polytechnique de GRENOBLE, 9 décembre 2005.
- [26] Paulin. I. K. (2007) Technologie RFID (Radio Frequency Identification) : Concepts et Stratégie de mise en œuvre. Université Laval. Québec.
- [27] <http://www.rfid360.org/le-passeport-biometrique-rfid-arrive-en-algerie/> dernière consultation le 19 avril 2017.
- [28] <https://www.fraunhofer.de/en/press/research-news/2014/december/the-flying-inventory-assistant.html> dernière consultation le 19 avril 2017.

- [29] H. Wang, J. Liu and J. Han, "RS-CPS: A Distributed Architecture of Robotic Surveillance Cyber-Physical System in the Nature Environment". The 5th Annual IEEE International Conference on Cyber Technology in Automation, Control and Intelligent Systems June 8-12, 2015, Shenyang, China.
- [30] Edward A. Lee, Cyber Physical Systems: Design Challenges, 2008 11th IEEE International Symposium on Object Oriented Real-Time Distributed Computing (ISORC), Orlando, FL, May, 2008.
- [31] W. Kang, K. Kapitanova and S.H. Son, "RDDS: A Real-Time Data Distribution Service for Cyber-Physical Systems," IEEE Transactions on Industrial Informatics, vol. 8, no. 2, pp. 393-405, May 2012.
- [32] RODRIGUES Oliveira J.G, Bachoti. Y, Belhaj.S.B "Projet RFID", Projet de fin d'étude, option RMS, 25 janvier 2011.
- [34] Rfid market to reach 9.2 billion in 2014 and more than triple in next decad. Consulté le 19 avril 2017 de <http://www.mmh.com/article/rfid>
- [35] <https://www.ifm.com/de/en> derniere visite le 20 avril 2017
- [36] Mamei, M. and Zambonelli, F. (2007) 'Pervasive pheromone-based interaction with RFID tags', ACM Transactions on Autonomous and Adaptive Systems, Vol. 2, No. 2, p.4.
- [37] E. Bonabeau, M. Dorigo, G. Theraulaz, "Swarm Intelligence", Oxford University Press, 1999.
- [38] « Codes barres ou RFID dans les soins de santé et le secteur pharmaceutique », PHIDATA. [en ligne]. http://www.phidata.be/fr/marchs/soins_de_sant/[page consultée le 20 avril 2017].

