

الجمهورية الجزائرية الديمقراطية الشعبية  
*République Algérienne Démocratique et Populaire*  
وزارة التعليم العالي والبحث العلمي  
*Ministère de l'enseignement supérieur et de la recherche scientifique*

Université 20 Août 1955- Skikda  
Faculté des Sciences  
Département d'Informatique  
N° .....



جامعة 20 أوت 1955 سكيكدة  
كلية العلوم  
قسم الإعلام الآلي  
الرقم .....

## Thèse

En vue de l'obtention du diplôme de  
**Doctorat de 3<sup>o</sup> cycle (LMD) en Informatique**  
Option : *Systèmes Informatiques*

**Nouvelles approches pour l'optimisation de la  
consommation énergétique, l'efficacité du  
routage et pour l'agrégation des données  
dans les réseaux de capteurs sans fil**

Présentée par :

**Kenza REDJIMI**

Soutenue publiquement le : **22 Septembre 2022**

Devant le Jury composé de :

**Président :** Pr. Mazouzi Smaine Université 20 Août 1955- Skikda  
**Examineurs :** Pr. Maamri Ramdane Université Abdelhamid Mehri - Constantine 2  
Dr. Boulehouache Soufiane MCA Université 20 Août 1955- Skikda  
**Directeur de Thèse :** Dr. Boulaiche Mehdi MCA Université 20 Août 1955- Skikda

## **DEDICACES**

Je dédie ce travail à mon père, à ma mère et à ma sœur Lamia pour leur soutien sans faille et leurs encouragements.

## ***REMERCIEMENTS***

Je remercie avant tout, Dieu le tout-puissant pour m'avoir donnée la force et la volonté nécessaires pour mener à bout ce travail de thèse.

Je remercie mon directeur de thèse ; monsieur Boulaiche Mehdi MCA au département d'informatique de l'université 20 Août 1955- Skikda, pour avoir accepté de diriger cette thèse ainsi que pour tous ses conseils, recommandations et observations tout au long de ce parcours.

Je remercie monsieur Mazouzi Smaine professeur à l'université 20 Août 1955 – Skikda, pour avoir accepté de présider ce jury de thèse ainsi que pour toutes ses observations et remarques pertinentes et constructives.

Mes remerciements vont également à monsieur Maamri Ramdane Professeur à l'université Abdelhamid Mehri – Constantine 2, pour avoir accepté d'examiner le manuscrit de cette thèse.

Je remercie également, monsieur Boulahouèche Sofiane MCA au département d'informatique de l'université 20 Août 1955- Skikda pour avoir pris le soin d'examiner cette thèse.

## الملخص

---

إن التطورات العلمية والتكنولوجية في مجالات الإلكترونيات، الحوسبة، الاتصالات السلكية واللاسلكية وهندسة البرمجيات المرتبطة بالتقدم المحرز في تصغير حجم الآلات و الدارات جعلت من الممكن توفير أجهزة عالية الأداء ذات أسعار منخفضة و جودة عليا مثل الهواتف الذكية والأنظمة الأخرى التي يمكن تضمينها. يؤدي توزيع المهام وتنسيقها وتوازيها إلى تقديم عدة أنواع من الحلول من بينها تقسيم المعالجة بصفة موازية إلى مجموعات من المعالجات المحلية. في هذا النوع من الحالات؛ الحل الجماعي ينبثق من الحلول المحلية. في ترتيب الأفكار هذا ، تتكون شبكات الاستشعار اللاسلكية من عدد كبير من وحدات الاستشعار التي تستهلك طاقة منخفضة و هي متوفرة على إمكانات المعالجة و التخزين متواضعة إن لم نقول ضعيفة، وحيث يتمثل دورها في استخراج المعلومات الأولية ثم تحويلها وإرسالها إلى مراكز المعالجة البعيدة باستخدام الاتصالات اللاسلكية.

يندرج العمل موضوع هذا البحث ضمن هذا الإطار ويقترح مجموعة من الأفكار لضمان النقل بدون فقدان للبيانات التي يتم جمعها بواسطة عناصر الاستشعار الموزعة عبر الشبكة من المكان الذي يتم فيه جمع هذه المعلومات إلى المحطات الأساسية البعيدة. تقترح الخوارزمية الأصلية التي تمت محاكاتها وتنفيذها بروتوكول توجيه جغرافي (IEGGR) يضمن تسليم البيانات حتى في حالة عدم وجود عناصر استشعار في الاتجاه المطلوب أي بين العنصر المرسل للمعلومة و المحطة المركزية.

**الكلمات المفتاحية:** شبكات الاستشعار اللاسلكية (WSN) ، بروتوكولات التوجيه (هرمي ، مسطح ، جغرافي) ، رأس الكتلة (CH) ، المحاكات، البروتوكول التوجيه الجغرافي (IEGGR).

Les avancées scientifiques et technologiques dans les domaines de l'électronique, de l'informatique, des télécommunications et du génie-logiciel associées aux progrès consentis dans la miniaturisation des machines permettent de mettre à disposition des appareils très performants de taille réduite tels que les Smartphones et autres systèmes pouvant être embarqués. La distribution des tâches, leur coordination et parallélisations apportent plusieurs types de solutions parmi lesquelles la parcellisation de traitements globaux en des ensembles de traitements locaux. Dans ce type de cas ; la solution globale émerge des solutions locales. Dans cet ordre d'idées, les réseaux de capteurs sans fil sont constitués d'un grand nombre d'unités de captages dits nœuds capteurs disposant de faibles potentialités d'énergie et de traitement qui ont pour rôle d'extraire l'information brute, de la transformer et de l'envoyer vers des centres de traitement lointains grâce à des communications radio. Le travail entrepris ici, entre dans ce cadre et propose un ensemble d'idées permettant d'assurer le transfert sans perte des données recueillies par les nœuds capteurs depuis le lieu où ces informations sont recueillies vers les stations de base. Un algorithme original qui a été simulé et implémenté propose un protocole de routage géographique (IEGGR) qui permet de garantir la délivrance des données même dans le cas de l'existence de minima locaux (vide).

**Mots-clés :** Réseaux de capteurs sans fil, Protocoles de routage (Hiérarchiques, Plats, Géographiques), Cluster Head (CH), Simulateurs.

# Abstract

---

Scientific and technological advances in the fields of electronics, computing, telecommunications and software engineering associated with the progress made in the miniaturization of machines make it possible to provide high-performance devices with reduced sizes such as smartphones and other systems that can be embedded. The distribution of the tasks, their coordination and parallelization bring several types of solution among which the parceling of global processing into several sets of local processings. In this type of case; the global solution emerges from the local solutions. In this order of ideas, wireless sensor networks are made up of a large number of sensing units called sensor nodes with low energies and processing potentials, whose role is to extract information , to transform it and to send it to distant processing centers using radio communications. The work undertaken here falls within this framework and proposes a set of ideas to ensure the lossless transfer of the data collected by the sensor nodes from the place where this information is collected to the base station. An original algorithm that has been simulated and implemented proposes a geographic routing protocol (IEGGR) that guarantees the delivery of data even in the case of the existence of local minima.

**Keywords:** Wireless Sensor Networks (WSN), Routing Protocols (Hierarchical, Flat, Geographic), Cluster Head (CH), Simulators.

# Table des Matières

<b>Introduction Générale.</b>	
Contexte.	01
Motivations et Objectifs.	02
Organisation de la thèse.	03
Contributions scientifiques.	04
<b>Chapitre 1 : Généralités.</b>	
1.1-Introduction.	06
1.2-Les réseaux de capteurs sans fil.	07
1.3-Architecture générale d'un nœud capteur.	09
1.4-Propagation des ondes radio dans l'espace.	11
1.5-L'optimisation énergétique.	14
1.6-La communication dans un réseau de capteurs sans fil.	15
1.6.1-La pile protocolaire.	16
1.6.2-Système en couches.	17
1.7-Applications des réseaux de capteurs sans fil.	19
1.8-Les exigences de sécurité au niveau des capteurs sans fil.	21
1.9-Les menaces de sécurité et leurs contre-mesures dans la couche réseau.	23
1.10-Les modèles d'énergie.	24
1.10.1-La zone de captage ( Sensing range).	26
1.10.2-La zone de connectivité (communication range).	26
1.11-Les systèmes d'exploitation dédiés aux RCsFs.	28
1.11.1-Enjeux principaux d'un système d'exploitation WSN.	28
1.11.2-Modèles de programmation.	29
1.11.3-Tableau récapitulatif.	35
1.12-Conclusion.	36

<b>Chapitre 2 : Le routage dans les réseaux de capteurs sans fil.</b>	
2.1-Introduction.	37
2.2-Le routage dans les RCsF.	38
2.3-Les protocoles de routage.	39
2.3.1-Classification des protocoles de routage.	39
2.3.2-Les protocoles hiérarchiques homogènes.	41
2.3.2.1-Le protocole LEACH.	41
2.3.3-Les protocoles de routage hétérogènes.	45
2.3.3.1-Le protocole DEEC.	46
2.3.3.2-Le protocole EDEEC.	48
2.3.3.3-Simulation.	48
2.4-Cas d'application : Conception et mise en œuvre d'une application de gestion d'un parking intelligent (Smart Parking).	50
2.4.1-Conception de l'approche.	54
2.4.2-Modélisation avec RENEW.	55
2.4.3-La mise en œuvre utilisant la plateforme JESS du modèle.	56
2.4.4-Discussion.	57
2.5-Conclusion.	57
<b>Chapitre 3: Les simulateurs des protocoles de routage dans les réseaux de capteurs sans fil.</b>	
3.1-Introduction.	58
3.2-La modélisation/simulation dans les RCsFs.	59
3.2.1-Aspects liés à la simulation.	61
3.2.2-Intérêts de la Modélisation/Simulation.	63
3.2.3-Le processus de Modélisation/Simulation.	65
3.2.4-Outils pour la simulation.	67
3.2.5-Les paramètres de simulation.	73
3.3-La modélisation/simulation basée sur les systèmes multi-agents.	74

3.3.1-Les systèmes multi-agents.	75
3.3.2-L'approche proposée.	76
3.3.3-La plateforme multi-agent MADKIT.	79
3.3.4-Application à la simulation des protocoles de routage dans les RCsFs : Cas des protocoles hiérarchiques.	80
3.3.5-Mise en œuvre de l'approche.	80
3.3.6-La structure organisationnelle du simulateur.	81
3.4-Cas d'application.	83
3.4.1-Quelques fonctionnalités du simulateur.	84
3.5-Conclusion.	86
<b>Chapitre 4: Contribution : Le protocole de routage géographique IEGGR.</b>	
4.1-Introduction.	87
4.2-Le routage géographique.	87
4.3-Le modèle d'énergie.	90
4.4-Contribution : Le protocole géographique proposé IEGGR.	91
4.4.1-Aperçu de l'approche.	92
4.4.2-Formation des sous-graphes.	94
4.4.3-Evaluation des performances.	101
4.5-Conclusion.	108
<b>Conclusion et travaux futurs.</b>	109
<b>Références</b>	112

# Liste des figures

---

Figure 1.1- Exemple de réseau de capteurs sans fil.	07
Figure 1.2-Schéma simplifié d'un capteur.	07
Figure 1.3-Conversion analogique digitale	08
Figure 1.4-Architecture sommaire d'un système basé autour d'un microprocesseur.	09
Figure 1.5-Architecture générique d'un nœud capteur.	10
Figure 1.6-Zone de couverture d'une antenne.	12
Figure 1.7-Pile protocolaire dans un réseau de capteurs sans fil.	19
Figure 1.8-Quelques applications courantes des réseaux de capteurs sans fil.	21
Figure 1.9-Modèle d'énergie (adapté de Heinzelman et al., 2002).	25
Figure 1.10-Localisation de capteurs.	28
Figure 1.11-Architecture de TyniOS.	32
Figure 1.12-Architecture générale de Contiki.	32
Figure 1.13-Architecture de MANTIS.	33
Figure 1.14-Architecture de Nano RK.	34
Figure 1.15-LiteOs.	35
Figure 2.1- routage de l'information : (a) mono-saut, (b) multi-saut. (c) ajout d'un nœud capteur, (d) ajout d'un relai.	39
Figure 2.2-Classification de protocoles de routage	41
Figure 2.3-Le clustering dans LEACH.	42
Figure 2.4-Phases dans LEACH.	43
Figure 2.5- Exemple de déploiement d'un réseau de capteurs sans fil	49
Figure 2.6-Nombre de nœuds vivants.	50
Figure 2.7-Nombre de paquets transmis.	50
Figure 2.8- Architecture de communication dans un parking intelligent	52
Figure 2.9-Les réseaux dans les réseaux (Nets within Nets).	55
Figure 2.10-Modélisation du système avec Renew.	56
Figure 3.1-Cadre conceptuel pour la modélisation/simulation.	65
Figure 3.2-Le processus de décomposition.	77
Figure 3.3-Le modèle Agent/Groupe/Rôle (AGR).	78
Figure 3.4-Exemple de modélisation AGR du simulateur.	79

Figure 3.5- Le modèle AGR correspondant à l'approche proposée.	81
Figure 3.6-Organisation du simulateur pour le routage hiérarchique.	84
Figure 3.7-Exemple de valeurs concernant un nœud capteur.	85
Figure 3.8-Durée de vie du réseau.	86
Figure 4.1-Face routing.	89
Figure 4.2-Zone de routage.	93
Figure 4.3-Zone de récupération.	93
Figure 4.4- Sous-graphe local G' construit à partir de la zone de routage.	98
Figure 4.5- Sous-graphe local G' construit à partir de la zone de récupération.	100
Figure 4.6-Energie moyenne consommé par rapport au nombre de noeuds du réseau.	104
Figure 4.7-Moyenne de l'énergie consommée en fonction du temps.	104
Figure 4.8-Comparaison des taux de succès.	105
Figure 4.9-Durée de vie du réseau : nombre de nœuds actifs.	106
Figure 4.10-Délais de transmission.	107

## *Liste des tableaux*

---

Tableau 1.1-Comparaison entre les systèmes d'exploitation des RCsF.	35
Tableau 2.1-Paramètres de simulation.	49
Tableau 3.1-Comparaisons de quelques simulateurs des RCsF.	72
Tableau 3.2-Affectation AGR au sein du simulateur.	82
Tableau 3.3-Exemples de Paramètres de simulation.	85
Tableau 4.1-Paramètres utilisés dans la simulation.	102

# *Liste des acronymes et abréviations*

---

<b>Acronyme</b>	<b>Signification</b>
ABMS	Agent Based Modeling and Simulation
ADV	Advertising
AGR	Agent/Group/Role
AODV	Ad hoc On-demand Distance Vector
API	Application Programming Interface
BEENISH	Balanced Energy Efficient Network Integrated Super Heterogeneous
BS	Base Station
CAN	Convertisseurs Analogiques Numériques
CH	Cluster Head
CNA	Convertisseurs Numériques Analogiques
CPS	Cyber Physical Systems
CPU	Central Processing Unit
CSMA	Carrier Sense Multiple Access
Db	Décibel
DDEEC	Developed DEEC
DEEC	Distributed Energy Efficient Clustering
DEVS	Discrete Event System specification
DoS	Denial of Service attacks
DSDV	Destination Sequenced Distance Vector
DSR	Dynamic Source Routing
DSSS	Direct Sequence Spread Spectrum
EDEEC	Enhanced Distributed Energy Efficient Clustering
EGGR	Energy-aware and Delivery Guarantee Geographic Routing
FAQ	Foire Aux Questions
GPL	General Public License
GPS	Global Positioning System
GPSR	Greedy Perimeter Stateless Routing
HEED	Hybrid Energy-Efficient Distributed Clustering
IEGGR	Improved Energy-aware and Delivery Guarantee Geographic Routing
IoT	Internet of Things
IP	Internet Protocol
IPv6	Internet Protocol Version 6
LEACH	Low Energy Adaptive Clustering Hierarchy
LGPL	Lesser General Public License
M BEENISH	Modified Balanced Energy Efficient Network Integrated Super Heterogeneous
M S T	Minimal Spanning Tree
MAC	Medium Access Control
MADKIT	Multi Agent Development KIT
MEMS	MicroElectroMechanical Systems

NEMS	NanoElectroMechanical Systems
OSI	Open System Interconnection
PEGASIS	Power-Efficient Gathering in Sensor Information Systems
RCSF	Réseaux de Capteurs Sans Fil
RFID	Radio Frequency IDentification
SEP	Stable Election Protocol
SMA	Systèmes Multi-Agents
SNR	signal to noise ratio
TCP	Transmission Control Protocol
TDMA	Time Division Medium Access
WSN	Wireless Sensor Networks

# Introduction Générale

## 1- Contexte

Les réseaux de capteurs sans fil tirent leurs puissances des progrès sans cesse croissants dans divers domaines scientifiques et technologiques. Ainsi, plusieurs disciplines concourent à optimiser les performances de ces composants. La miniaturisation des composants électroniques et l'intégration très poussée dans le domaine de la micro-électronique permet d'enfouir des circuits systèmes complexes dans des puces de silicium dont des surfaces n'excédant pas le mm<sup>2</sup>. Les récents progrès dans les domaines de l'architecture des ordinateurs et ceux consentis dans celui du génie-logiciel dotent les nœuds capteurs de grandes capacités pour l'adaptation de leurs architectures ainsi que d'une grande profusion d'applications logicielles prêtes à être utilisées. De même, le domaine des communications à distance connaît lui-aussi une grande révolution et les applications et les ressources deviennent de plus en plus partagées. Le réseau Internet permet de relier toute la planète grâce à une toile géante dont la composition et les ramifications s'étendent d'heure en heure. Les communications sans fil, bien que de portée et de bande passante limitées permettent à toute la population de la planète de communiquer instantanément. Nous sommes à l'âge de la digitalisation et le monde tend vers le tout-digital. Ces progrès touchent tous les domaines techniques désignés sous les noms de MicroElectroMechanical Systems (MEMS) ou NanoElectroMechanical Systems (NEMS) qui prônent l'intégration de composants électriques et mécaniques à des niveaux microscopique et nanoscopiques (Sun et al., 2018 ; Lee et al., 2008 ; Lam et al., 2011).

Les systèmes cyber physiques CPS (Cyber Physical Systems) (Serpanos, 2018) sont des systèmes qui intègrent des composants informatiques et physiques pour implémenter des processus dans le monde réel. Les CPS constituent une technologie innovante pour de nombreuses industries en raison de leur potentiel d'intégration des technologies multidisciplinaires. Ces systèmes recueillent des grandeurs physiques concernant l'information à traiter. Les réseaux de capteurs sans fil forment, très souvent, le système d'acquisition de l'information sous sa forme physique pour les CPS et jouent un rôle très important dans diverses activités des CPS.

Les réseaux de capteurs sans fil sont formés de plusieurs nœuds capteurs (souvent des centaines) dont le but est de recueillir l'information sous sa forme physique (grandeur physique continue : humidité, lumière, température, détection d'objets, tension artérielle, ...) (Koh et al. , 2016). Cette information est ensuite convertie en grandeur numérique qui subit des traitements locaux puis elle est envoyée vers une station de base distante pour être acheminée vers les utilisateurs finaux. Chaque nœud capteur dispose de ressources matérielles, énergétiques et logicielles limitées et il ne peut, ainsi, qu'effectuer des traitements simples. Ces nœuds sont souvent placés dans des endroits hostiles voire inaccessibles et étant donné qu'ils sont alimentés par des batteries leur énergie est limitée et il est, alors, indispensable d'en optimiser les dépenses afin d'assurer la longévité du réseau.

## **2- Problématiques :**

Le problème de la gestion énergétique dans les RCSFs a été et demeure au centre des préoccupations de plusieurs sujets de recherche et plusieurs approches ont été proposées pour apporter des solutions à ce problème. De ce fait, plusieurs protocoles sont implémentés à l'heure actuelle. Cependant, d'autres problèmes cruciaux sont posés dans les RCSFs, dont celui de la sécurité du réseau, surtout dû au fait que l'information est partagée de façon ubiquitaire et qu'elle peut être, ainsi, potentiellement accessible par des intrusions malveillantes. La sécurité de l'information échangée est, dans ce cas, menacée de même que sa confidentialité entre autres problèmes pouvant survenir quant à son acheminement sain vers sa destination.

Le nombre d'informations (paquets) et leur fréquence sont aussi à considérer en fonction de la bande passante et de la latence du réseau de connexion utilisé. Plusieurs analyses statistiques et des algorithmes sont utilisés pour pallier ce problème. Des fonctions d'agrégation tels que le calcul du minimum, du maximum, de la moyenne, ... sont utilisées afin d'assurer la transmission des informations pertinentes. L'information est souvent compressée et encryptée avant d'être envoyée, ce qui conduit à une consommation énergétique supplémentaire mais nécessaire.

C'est dans cette optique qu'est conduite cette thèse, il est question d'apporter quelques réflexions à ces problèmes posés dans les réseaux de capteurs sans fil et à

proposer quelques solutions qui ont été publiées dans des revues et des conférences scientifiques.

La principale contribution de cette thèse concerne la proposition d'une nouvelle approche dénommée IEGGR (Redjimi et al., 2021a ; Redjimi et al. , 2021b) dévolue aux protocoles géographiques qui prend en charge le problème du vide dans le réseau ; lorsque l'algorithme proposé ne trouve pas de nœuds capteurs dans la direction de la station de base pour lui envoyer l'information. Un autre problème abordé et dont des éléments de réponses ont fait l'objet de communications dans une conférence (Redjimi et al., 2022a) et publiés dans une revue (Redjimi et al, 2022b) concerne la comparaison des protocoles DEEC et EDEEC aux niveaux gestion de l'énergie et paquets envoyés dans les réseaux de capteurs sans fil hétérogènes. Enfin, nous nous sommes penchés sur les problèmes des simulateurs des RCSFs pour proposer le squelette de nouveaux simulateurs basés sur les systèmes multi-agents (SMA) (Redjimi et al., 2021c)

Toutes ces contributions vont dans le sens de répondre à quelques problématiques soulevées par les RCSFs.

### **3- Organisation du manuscrit de thèse**

Ce manuscrit de thèse est articulé autour des points suivants :

- Le premier chapitre concerne une étude de l'état de l'art concernant le domaine des RCSFs.
- Le second chapitre est dédié à la présentation des protocoles de routage de l'information dans les RCSFs.
- Le troisième chapitre concerne une étude comparative des simulateurs des RCSFs ainsi qu'à la présentation d'une nouvelle approche basée sur le modèle AGR (Agent/Groupe/Rôle).
- Dans le quatrième chapitre, il est question des protocoles géographiques et nous y présenterons et discuterons en détail l'approche proposée qui constitue la contribution principale de ce travail.

## 4- Contributions scientifiques

### 4.1 Articles parus dans des revues internationales

- 1- Redjimi, Kenza., Boulaiche, Mehdi., & Redjimi, Mohammed. (2021). IEGGR: Improved Energy-aware and Delivery Guarantee Geographic Routing. *Journal of Communications Software and Systems*, 17(2), pp 143-153. DOI: [10.24138/jcomss-2020-0019](https://doi.org/10.24138/jcomss-2020-0019)
- 2- Redjimi, Kenza., & Redjimi, Mohammed. (2022). The DEEC and EDEEC Heterogeneous WSN Routing Protocols. *International Journal of Advanced Networking and Applications*, 13(4), pp 5045-5051.
- 3- Redjimi, Kenza. (2022). Modelling and Simulation for Smart Parking. *Quest Journals Journal of Software Engineering and Simulation Volume 8 ~ Issue 1* pp 25-30 ISSN(Online) :2321-3795 ISSN (Print):2321-3809 [www.questjournals.org](http://www.questjournals.org)

### 4.2 Articles parus dans des revues nationales

- 1- Redjimi, K & Redjimi Mohammed. (2021) Modeling Based Multi - Agent System of an Industrial System. *Revue Algérienne des Sciences A*, volume 07, pp 7-13.

### 4.3 Communications internationales

- 1- Redjimi, Kenza., Boulaiche, Mehdi., & Redjimi, Mohammed. (2021) Agent Based Modeling and Simulation for Geographic Routing Protocol in the Wireless Sensor Networks. In *9th (Online) International Conference on Applied Analysis and Mathematical Modeling (ICAAMM21)* June 11-13, 2021, Istanbul-Turkey (p. 78). [https://www.ntmsci.com/Areas/Conferences/FilesAndImages/19/ICAAMM2021\\_AbstractBook.pdf](https://www.ntmsci.com/Areas/Conferences/FilesAndImages/19/ICAAMM2021_AbstractBook.pdf)
- 2- Redjimi, Kenza., Boulaiche, Mehdi., & Redjimi, Mohammed. (2022). DEEC and EDEEC Routing Protocols for Heterogeneous Wireless Sensor Networks: A Brief Comparative Study. In: Troiano, L., Vaccaro, A., Kesswani, N., Díaz Rodriguez, I., Brigui, I. (eds) *Progresses in Artificial Intelligence & Robotics: Algorithms & Applications. ICDLAIR 2021. Lecture Notes in Networks and Systems*, vol 441. Springer, Cham. [https://doi.org/10.1007/978-3-030-98531-8\\_12](https://doi.org/10.1007/978-3-030-98531-8_12)

- 3- Redjimi, Kenza., & Redjimi, Mohammed. (2022). A Multi-Agent System for Industrial Simulators Design. In *Advances in Deep Learning, Artificial Intelligence and Robotics*. pp. 129-140. Springer, Cham.
- 4- Redjimi, Kenza., & Redjimi, Mohammed. (2020, October). Multi-agent-Based Approach for Complex Industrial Process Modeling. In *Brazilian Technology Symposium*. pp. 598-608. Springer, Cham.
- 5- Redjimi, Kenza., Redjimi, Mhammed., & Boulaiche, Mehdi. (2021, January). Improved Geographic Routing Protocol for Wireless Sensor Networks. In *International Conference on Digital Technologies and Applications*. pp. 1135-1145. Springer, Cham.
- 6- Redjimi, Kenza., Boulaiche, Mehdi., & Redjimi, Mohammed. (2021). A Multi-Agent Based Simulation Approach for Hierarchical Routing Protocol in the Wireless Sensor Networks. In *9th International Conference on Software Engineering and New Technologies (ISCENT2021) 22 – 23 July 2021, Annaba- Algeria*.
- 7- Redjimi, Kenza., Boulaiche, Mehdi., & Redjimi Mohammed. (2021) Devs-suite Simulation of Industrial Systems. In *International Symposium of Scientific Research and Innovative Studies (ISSRIS 21)- 22-25 February 2021, Turkey*.
- 8- Redjimi, Kenza., Boulaiche, Mehdi., & Redjimi Mohammed. (2021). Multi-agent Approach for Modelling and Simulation of Industrial Processes. In *International Symposium of Scientific Research and Innovative Studies (ISSRIS 21)- 22-25 February 2021, Turkey*.

#### **4.4 Communications nationales**

- 1- Redjimi Kenza, & Redjimi Mohammed. (2021) Modélisation Basée Multi-agent d'un processus industriel. In *The First National Conference on Artificial Intelligence and Information Technologies (CNIATI'21)- May 24th, 2021, El-Tarf – Algeria*.

- 1.1- Introduction
  - 1.2- Les réseaux de capteurs sans fil
  - 1.3- Architecture générale d'un nœud capteur
  - 1.4- Propagation des ondes radio dans l'espace
  - 1.5- L'optimisation énergétique
  - 1.6- La communication dans un réseau de capteurs sans fil
  - 1.7- Applications des réseaux de capteurs sans fil
  - 1.8- Les exigences de sécurité dans un réseau de capteurs sans fil
  - 1.9- Les menaces de sécurité et leurs contre-mesures dans la couche réseau
  - 1.10- Les modèles d'énergie
  - 1.11- Les systèmes d'exploitation dédiés aux RCsFs
  - 1.12- Conclusion
- 

## 1.1 Introduction

Les progrès très importants consentis dans les domaines technologiques et particulièrement en informatique ont mis à disposition de nouveaux outils qui ont révolutionné notre vie de tous les jours, nos façons de vivre, nos habitudes voire même influencé nos comportements et nos façons de penser. Aujourd'hui, la technologie est partout ; dans les voitures que nous utilisons tous les jours, les téléphones portables, les fours à micro-ondes, les usines, l'administration, la télévision,... L'information prend de plus en plus de place et elle est considérée comme étant le 'vrai' pouvoir, elle est partagée de façon ubiquitaire. Tout système de communication est basé autour d'émetteurs et de récepteurs qui se partagent différents canaux permettant la propagation de l'information. Dans le cas d'un canal de propagation sans fil, l'échange des informations est effectué non pas à l'aide d'un support de connexion matériel spécifique (fil) mais par l'utilisation de l'espace ambiant ; soit par des moyens acoustiques (voix, ...), optiques (fumée, infrarouge, Laser, etc.) ou ondes radio. Nous nous intéressons, ici, à la *communication sans fil* utilisant des liaisons hertziennes (radio). Les caractéristiques des liaisons radio (comportements, qualité du signal reçu par le récepteur, bande passante, débit,...) dépendent fortement du canal de propagation ainsi que des performances techniques et des qualités de l'émetteur et du récepteur.

La technologie sans fil permet l'échange de l'information sans se préoccuper du câblage classique souvent très contraignant. Le support véhiculant l'information est, dans ce cas, l'espace ambiant dans lequel se propagent les ondes radio qui acheminent cette information codée sous les contraintes de portée et de bande passante. Dans un réseau de capteurs sans fil, des centaines, voire des milliers de nœuds capteurs recueillent l'information et s'échangent les données pour les transmettre depuis leur endroit où elles sont produites et recueillies vers leur destination finale.

Ce premier chapitre est consacré à la présentation de quelques notions de base concernant les réseaux de capteurs sans fil.

## 1.2 Les réseaux de capteurs sans fil

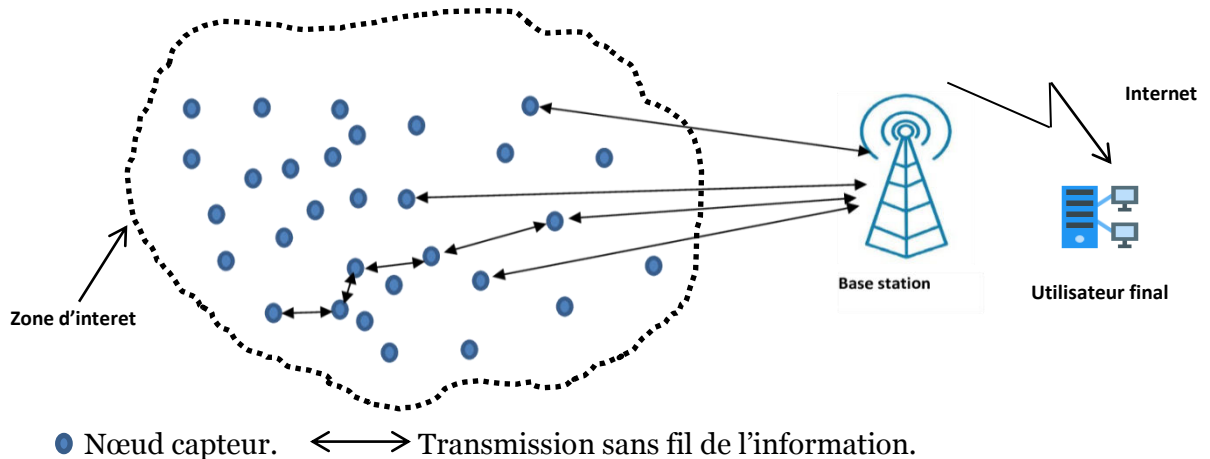


Figure 1.1- Exemple de réseau de capteurs sans fil.

Un réseau de capteurs sans fil est constitué d'un ensemble, souvent très important de nœuds capteurs (pouvant être des milliers) disséminés dans une zone géographique dite zone d'intérêt dans laquelle ils recueillent les informations pertinentes qui sont acheminées vers une station de base éloignée pour y être traitées (Akyildiz et al., 2002, Borges et al., 2014 ; Karray et al. , 2018 ). Les principaux rôles d'un nœud capteur sont : le recueil de l'information, ses traitements locaux et sa transmission (figure 1.1).

L'information brute recueillie par un nœud capteur est initialement sous sa forme physique (grandeur physique : température, humidité, lumière, détection de présence, feux de forêt, ....). Il y'a lieu, ainsi, de transformer cette information de sa forme initiale vers une forme électrique correspondante. Ce premier traitement est effectué par un dispositif spécifique appelé capteur (figure 1.2). Un capteur est un élément qui permet la transformation d'un signal représentant une grandeur physique très souvent analogique vers une grandeur électrique (ou autre grandeur utilisable) correspondante. Un capteur n'est pas un instrument de mesure ni un transducteur car il ne s'agit pas d'afficher les mesures effectuées mais il s'agit de systèmes d'acquisition de données qui seront par la suite manipulées par d'autres appareils électriques.

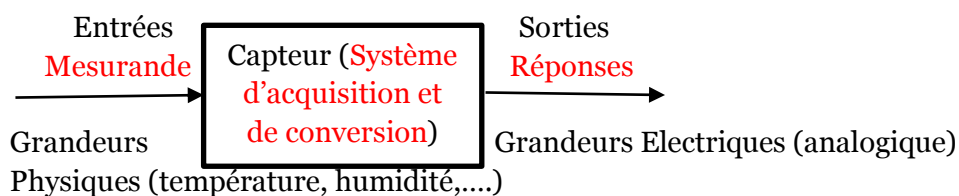


Figure 1.2 – Schéma simplifié d'un capteur.

Il existe plusieurs types de classification pour les capteurs. On parle de capteurs passifs, lorsque ces derniers fonctionnent sans aucun apport en énergie extérieure (exemple : thermistance, potentiomètre,...). De tels capteurs peuvent être modélisés par un dipôle dont l'impédance varie en fonction de la grandeur mesurée. Dans le cas de capteurs actifs, ces derniers sont constitués d'un ensemble de composants (transducteurs) alimentés. Ils agissent comme des générateurs qui génèrent des courants ou des tensions en fonction de l'intensité du phénomène physique considéré.

Une fois que le signal électrique est généré par le capteur, un deuxième élément traduit ce dernier en un signal numérique pouvant être traité par une machine digitale. Il s'agit là de Convertisseurs Analogiques Numériques (CAN) dont la fonction est de transformer un signal électrique analogique (continu) en un signal numérique (Shannon, 1984) (figure 1.2).

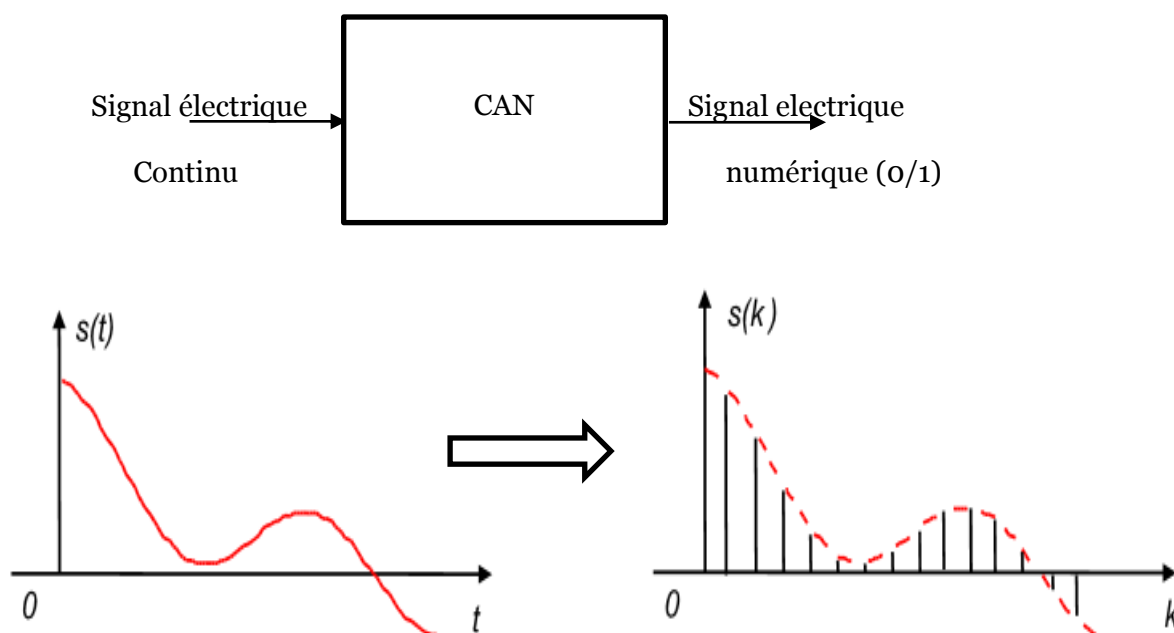


Figure 1.3- Conversion analogique digitale

Le signal analogique est échantillonné à des intervalles de temps périodiques. La fréquence d'échantillonnage doit respecter le théorème de Nyquist-Shannon (1.1) afin que le signal numérique obtenu en sortie du CAN soit fidèle au signal d'entrée. L'énoncé de ce théorème stipule que pour que le signal d'entrée (continu) soit entièrement reconstruit à partir des échantillons, il est nécessaire que la fréquence d'échantillonnage ( $f_e$ ) soit strictement supérieure à deux fois la plus grande fréquence ( $f_{max}$ ) présente dans le spectre du signal continu (aussi connue sous le nom de condition de Nyquist-Shannon) (Por et al., 2019) :

$$f_e > 2 f_{max} \quad (1.1)$$

$f_e$  : fréquence d'échantillonnage,  $f_{max}$  fréquence maximale du signal à échantillonner.

Une fois digitalisé, le signal obtenu peut à présent être traité par une machine digitale (automate, microprocesseur). Après que le traitement soit effectué, un ensemble de sorties et de décisions peuvent être exploitées. La figure (1.4) présente une architecture simplifiée de traitement basée autour d'un microprocesseur.

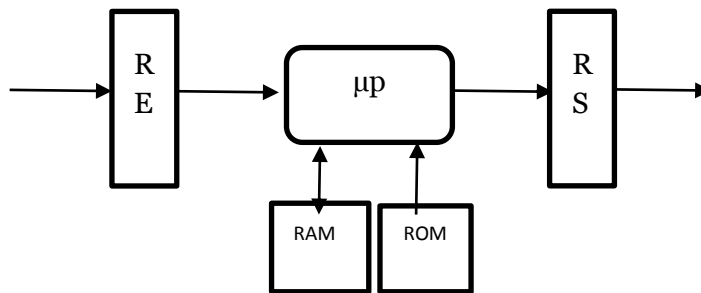


Figure 1.4- Architecture sommaire d'un système basé autour d'un microprocesseur.

Le système est alimenté en entrée par des registres d'interface (RE : Registre en Entrée) qui lui permettent d'une part de recueillir différents signaux issus des convertisseurs analogiques numériques (CAN), ces registres lui permettent aussi différentes autres informations telles que des signaux de consignes de commandes et de contrôle. Le microprocesseur exécute des programmes depuis ses mémoires vive (RAM) et morte (ROM). Un noyau du système d'exploitation spécialisé résidant dans ces mémoires permet de gérer ce système. Les interfaces de sorties (RS : Registres en Sortie) assurent le transfert des informations issues du microprocesseur vers l'environnement extérieur.

### 1.3 Architecture générale d'un nœud capteur

Un composant élémentaire du réseau appelé nœud capteur est un système intégrant plusieurs technologies : mécaniques, énergétiques, électroniques, physiques et informatiques. Les progrès consentis dans les circuits Micro Electro Mechanical Systems (MEMS) ou Nano Electro Mechanical Systems (NEMS) qui désignent l'intégration de composants électriques et mécaniques à des niveaux microscopique et nanoscopiques (Sun et al., 2018 ; Lee et al., 2008 ; Lam et al., 2011) font que ces unités offrent plusieurs fonctionnalités à des coûts appréciables avec un encombrement spatial moindre et de grandes robustesses et fiabilité.

Les fonctions attendues d'un nœud capteur s'articulent autour des besoins suivants :

- Recueil de l'information brute sous forme d'un signal physique et sa transformation en un signal électrique correspondant (rôle du capteur).
- Conversion du signal électrique analogique en un signal électrique numérique (CNA).

- Transfert de l'information numérique vers un organe de traitement (CPU, mémoires, registres, logiciels et systèmes dédiés, ...).
- Système de traitement du signal numérique représentant l'information recueillie et sa transmission vers sa destination. Cette transmission se fait sans fil, d'où la nécessité de l'intégration de modules pouvant assurer ce genre de communication (antennes, circuiterie adéquate).
- Source d'énergie pour alimenter ces composants.

La figure 1.4 suivante présente une architecture générique d'un nœud capteur :

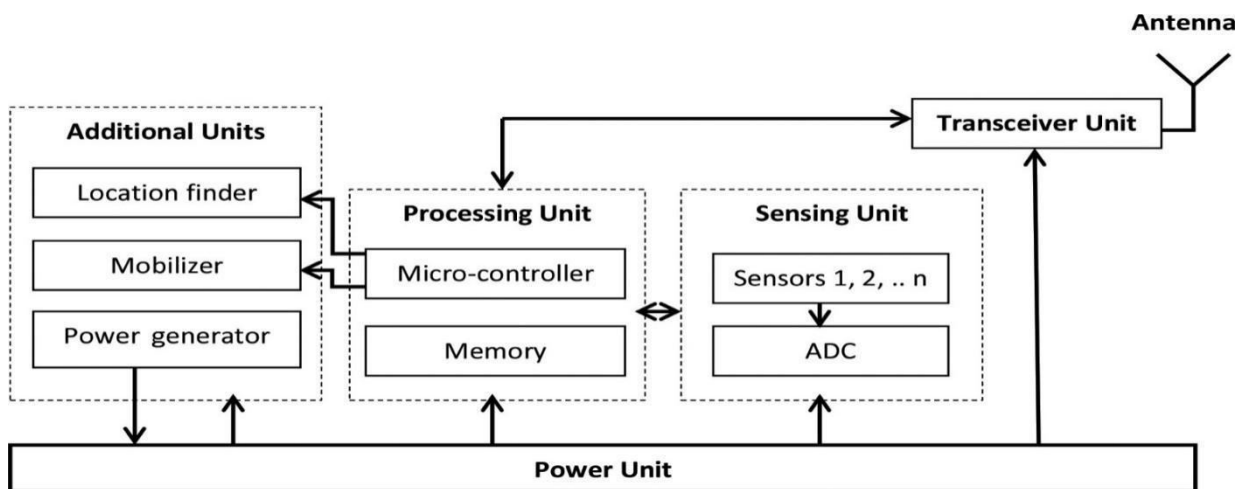


Figure 1.5- Architecture générique d'un nœud capteur.

Un nœud capteur est composé des quatre principaux composants suivants (Figure 1.5) : Un ou plusieurs capteurs (On utilisera avantageusement un capteur multifonctions), des unités de traitement et de calcul associés à des mémoires, des unités de communication sans fil ainsi qu'une ou plusieurs sources d'énergie. Généralement, d'autres composants sont associés aux nœuds capteurs tels que des modélisateurs et des systèmes de géolocalisation.

- Les unités de captage : Ces éléments regroupent les capteurs proprement dits et les convertisseurs analogiques digitaux.
- Les unités de calcul et de traitement : Ces unités traitent les données qui leur sont transmises par les unités de captage en exécutant des applications préalablement chargées dans les mémoires ROM qui leur sont associées. Les résultats sont stockés dans les mémoires du nœud capteur et/ou acheminés directement vers d'autres destinations. Ces unités (CPU et mémoires) sont des ressources à capacités très limitées étant données les tailles réduites des nœuds capteurs ainsi que leurs nombres (des milliers). Augmenter les performances d'un nœud, reviendrait à ajouter des coûts supplémentaires qui seraient exorbitants.

- Les unités de communication : Leurs rôles résident dans la réception et l'envoi d'information. Ces opérations s'effectuent de façon non filaire et à cet effet, des antennes de réception/transmission par ondes radio sont associées aux nœuds capteurs.
- Les ressources énergétiques : Il s'agit le plus souvent de batteries non rechargeable et dont l'énergie est limitée. La gestion optimale de l'énergie dans les RCSF est l'un des défis majeur auquel les chercheurs tentent d'apporter des solutions. Très souvent, ces nœuds capteurs sont situés des endroits hostiles ou inaccessibles et le remplacement des batteries devient alors impossible.
- Les générateurs d'énergie : Il s'agit de transformations énergétiques le plus souvent naturelles et non polluantes telles que l'énergie solaire, les marées, le vent,.... Ces générateurs fournissent des énergies d'appui très appréciables pour les nœuds capteurs, seulement elles ne sont pas toujours accessibles dans des situations particulières de placement des nœuds (exemple : enfouissements souterrains).
- Les systèmes de géolocalisation : Ces systèmes permettent aux nœuds de communiquer avec le GPS, ce qui présente un grand intérêt pour la localisation de ces nœuds, cependant ce genre de communication peut entraîner une dépense énergétique très grande pour le nœud capteur.
- Le mobilisateur : Ce circuit est utilisé dans le cas des nœuds capteurs mobiles.

### 1.4 Propagation des ondes radio dans l'espace

Un élément important dans un système sans fil concerne la portée d'un lien radio (Signal range). C'est la distance maximale dans laquelle un récepteur peut détecter le signal émis par un émetteur. Cette couverture d'un émetteur est généralement déterminée par un disque de rayon  $R_c$  (*Rayon de communication*).  $R_c$  désigne cette distance au-delà de laquelle la communication est fortement atténuée voire même inexistante. En pratique, trois paramètres principaux déterminent la portée d'un lien radio : la puissance du signal émis par l'émetteur, les conditions de l'environnement dans lequel se propage le signal radio et la sensibilité du récepteur. La figure 1.6 illustre la zone de couverture (en rouge) d'une antenne prédite en fonction de la puissance moyenne reçue, tandis que la zone en vert indique la région qui est couverte réellement car la puissance reçue varie autour de la moyenne à cause des obstacles et différents bruits pouvant intervenir entre l'émetteur et le récepteur.

La puissance  $Pr[dBm]$  du signal reçu à la distance  $R_c$  est déterminée par l'équation suivante (1.2) :

$$Pr[dBm] = Pe[dBm] + Ge[dBi] + Gr[dBi] - PL[dB] + M[dB] \quad (1.2)$$

où  $Pe$  est la puissance d'émetteur en  $dBm$ ,  $Ge$  et  $Gr$  sont respectivement des gains des antennes émetteur et récepteur en  $dBi$ , et  $PL$  est l'affaiblissement de parcours en  $dB$ .

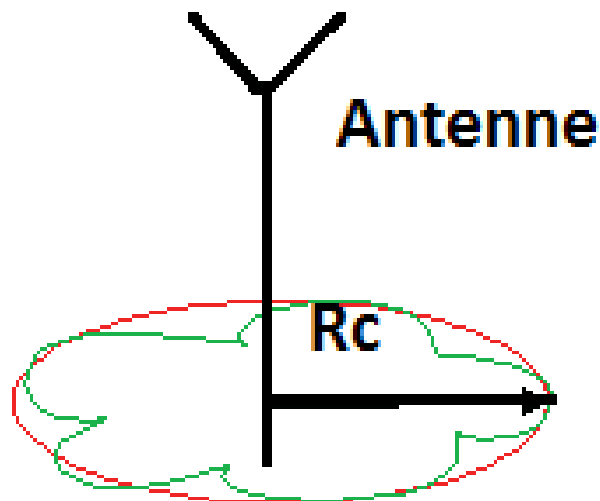


Figure 1.6 – Zone de couverture d'une antenne.

Dans un système à communication par ondes radio, l'information est transportée sous la forme d'un signal électromagnétique qui transite à travers l'espace. Parmi les caractéristiques importantes de ce signal, on trouve sa fréquence et sa puissance d'émission. L'information envoyée est modulée, amplifiée et adaptée au canal utilisé. Le récepteur reçoit une information sous forme d'un signal atténué. Ces atténuations sont dues aux différents phénomènes physiques que traverse le signal émis. Parmi les causes perturbant le signal émis, on trouve : l'éloignement entre l'émetteur et le récepteur cause des affaiblissement du signal (On parle de *path Loss*, pour perte en chemin), l'évanouissement ( *fading* ), le masquage (*shadowing*), la diffusion et la réflexion (*scattering*) : Dans ce dernier cas ; le récepteur reçoit différentes répliques du même signal transitant par des chemins différents.

Une des contraintes principales pour qu'un signal soit détecté et exploité de avec succès par un récepteur radio réside dans le fait que le rapport signal sur bruit  $S/N$  ( $SNR$  : *signal to*

*noise ratio*), mesuré à la réception est supérieur à un seuil donné, ce seuil caractérise la *sensibilité* du récepteur. Le rapport signal sur bruit (S/N) (1.3) mesure la quantité de bruit contenue dans le signal. Il s'exprime par le rapport des puissances du signal ( $P_S$ ) et du bruit ( $P_N$ ) et il est souvent exprimé en décibels (dB).

$$\left(\frac{S}{N}\right)_{dB} = 10 \log \frac{P_S}{P_N} \quad (1.3)$$

Les origines du bruit peuvent être diverses, nous en donnons ci-dessous quelques exemples :

- **Les bruits d'origine thermique** : Les électrons dans un conducteur s'agitent de façon désordonnée lorsque ce dernier est soumis à une température supérieure au zéro absolu. Ceci provoque la création de courants électriques aléatoires qui s'ajoutent au courant traversant ce conducteur. Il s'agit là d'un bruit blanc. La puissance du bruit relative à l'agitation des électrons est proportionnelle à la température du conducteur (en kelvins) et à la fréquence considérée (Equation 1.4).

$$P_{TH} = k.T_k.B \quad (1.4)$$

Avec  $P_{TH}$  puissance du bruit (en W),  $k$  constante de Boltzmann =  $1,381.10^{-23}$  (en J/K),  $T$  température (en K) et  $B$  : largeur de la bande de fréquence (en Hz). *La densité spectrale de bruit* correspond à un bruit dont la largeur de bande est de 1 Hz ; elle est très souvent représentée par la lettre 'S'. Elle correspond au produit  $k.T_k$  et elle est utilisée dans le calcul de la sensibilité du récepteur et elle est directement proportionnelle à la valeur de la température.

- **Les bruits propres au récepteur** : Un récepteur relié à une antenne pour l'écoute d'un signal sur une fréquence donnée reçoit, en même temps que le signal utile, d'autres signaux indésirables (bruits). Parmi ces bruits, on distingue le bruit de fond radioélectrique pouvant provenir d'origines diverses : résultant d'activités humaines, venant de l'espace ou produite par des phénomènes naturels (orages, soleil,...). Ce niveau de bruit de fond peut varier en fonction de plusieurs facteurs : saison, météo, situation géographique, position de l'antenne, ... Cependant le facteur déterminant reste la fréquence. Les études montrent que ce bruit devient plus préoccupant lorsque la fréquence dépasse 30 MHz.

- **Autres bruits** : Dans un circuit semi-conducteur (transistor, diode,...) le passage des charges électriques au niveau des jonctions provoque un bruit (grenaille) dont la puissance diminue lorsque la fréquence augmente. On distingue d'autres bruits présents dans les semi-conducteurs tels que les bruits de scintillation (flicker noise) ainsi que les bruits de phases de divers oscillateurs. D'autres bruits sont produits par le récepteur lui-même et l'on peut

entendre des bruits de fond ( bruit blanc, souffle,...) même si ce dernier n'est raccordé à aucune antenne.

## **1.5 L'optimisation énergétique**

L'optimisation énergétique compte parmi les voies les plus exploitées et les plus importantes dans un réseau de capteurs sans fils. Ceci est d'autant plus crucial qu'il s'agit là, le plus souvent de réseaux déployés dans de vastes environnements parfois très hostiles voire même inaccessibles fonctionnant avec des batteries et dont le fonctionnement doit être assuré durant plusieurs années. L'optimisation énergétique dépend à la fois de l'électronique équipant les nœuds capteurs, du moyen de transport des données, de l'organisation du réseau ainsi que des quantités énergétiques dépensées lors des traitements et du stockage des données et de leurs transmission /réception. Les cinq grandes catégories suivantes peuvent être dégagées :

- 1- Optimisation de la communication radio : Il s'agit là de méthodes permettant d'optimiser (réduire) la consommation des nœuds capteurs tout en gardant une bonne connectivité. Parmi celles-ci, on peut citer :
  - Intégration de fonction de contrôle et d'adaptation de la puissance de transmission.
  - Utilisation de fonction de communication coopérative se basant sur le principe de sur-écoute dans le but d'augmenter et d'améliorer la portée et la qualité du signal,
  - Utilisation d'antennes directionnelles ayant pour effet d'émettre dans quelques directions à la fois,
  - Intégration de la radio cognitive pour adapter de façon dynamique des paramètres de transmission tels que la fréquence porteuse, la puissance de transmission, la modulation ou la bande passante.
- 2- Fonctionnement par cycle d'activité : Ce type de fonctionnement consiste à utiliser différents modes de fonctionnement au cours de l'activité des nœuds. Le nœud passera notamment en mode sommeil lors des phases d'inactivité durant les périodes où le nœud ne communique pas. Ainsi, idéalement, le nœud ne se réveille qu'au moment de ses activités de traitement (transmission/réception de données, élection des cluster heads (CHs), phases d'initialisation,...). Un tel fonctionnement permet aux nœuds un grand gain d'énergie et par voie de conséquence des économies importantes des batteries.
- 3- La réduction des données : Ceci a pour objectif de minimiser la quantité de de donnée à transmettre et, par voie de conséquence, optimiser l'énergie nécessaire à la

transmission et à la réception des messages. Plusieurs méthodes peuvent être utilisées à cette fin :

- L'agrégation des données, le nœud effectue des fonctions de calcul et des statistiques sur ses propres données et celles éventuellement transmises par d'autres nœuds, telles que : minimum, maximum, moyenne, somme,... selon le contexte de fonctionnement du réseau.
  - La compression des données permettant ainsi d'en réduire la taille en utilisant des algorithmes de compression de données adéquats (Huffman, RLE, Wavelets, JPEG pour les images,...) et de bénéficier de gains énergétiques conséquents.
  - La prédiction des données, pour éviter les transmissions redondantes des données.
- 4- Le routage efficace de l'information : Le routage de l'information désigne son acheminement à travers le réseau vers la base station (BS). Ce point est certainement parmi ceux qui ont reçu le plus d'attention de la part des chercheurs et, ainsi, plusieurs protocoles de routage ont été conçus et la plupart ont été implémentés. Un protocole de routage efficace permet une bonne optimisation énergétique du réseau de capteur sans fil (RCSF). Un protocole de routage approprié doit permettre d'éviter ou du moins de limiter les risques de collision de données ainsi que le coût énergétique de chaque transmission. Ainsi, plusieurs stratégies de routage ont été mises en œuvre. On en distingue, principalement trois types : les protocoles plats où tous les nœuds jouent les mêmes rôles. Les protocoles géographiques où chaque nœud communique avec son voisinage immédiat en direction de la base station et les protocoles hiérarchiques où des nœuds sont élus à chaque tour pour se voir attribuer un rôle qui va déterminer l'architecture du réseau.
- 5- Réapprovisionnement de la batterie : Des transformateurs d'énergie (solaire, éolienne, hydraulique,...) sont souvent intégrés dans les nœuds capteurs afin de réapprovisionner les batteries quand celles-ci sont placées dans des environnements naturels adéquats.

### **1.6 La communication dans un réseau de capteurs sans fil**

Les réseaux de capteurs sans fil sont des systèmes communicants par essence même. La démarche de conception et de développement de tels systèmes obéit à des impératifs spécifiques. Ces réseaux de capteurs sont déployés pour des applications particulières et leur but est, de ce fait, bien déterminé : collecter les informations nécessaires, leur faire

subir des traitements et des enregistrements locaux puis de transmettre les données vers la base station.

Parmi, les caractéristiques principales des réseaux de capteurs sans fil, on peut citer :

- L'absence d'infrastructure et de centralisation.
- La taille importante du réseau qui peut dépasser les milliers de nœuds capteurs.
- La topologie qui peut être statique dans le sens où les nœuds sont immobiles ou dynamiques et dans ce deuxième cas de figure, les nœuds capteurs peuvent être attachés à des objets ou des animaux qui se déplacent de façon aléatoire, ce qui rend la topologie du réseau difficile à maîtriser.
- Des transmissions peuvent interférer si elles sont émises sur les mêmes fréquences.
- Limitation de la bande passante : Les réseaux de capteurs sans fil utilisent un médium partagé. Plus le nombre de nœuds capteurs est élevé plus la bande passante est limitée.
- Les ressources énergétiques des RCSFs sont limitées est une caractéristique critique dont dépend la durée de vie du réseau.

### 1.6.1- La pile protocolaire

La pile protocolaire des réseaux de capteurs sans fil tient compte des caractéristiques et spécificités de ce type de réseaux. Ainsi, il est question de prise en charge de la consommation énergétique, des protocoles de routage et du traitement des données transmises. De ce fait, la majorité des réseaux de capteurs sont modélisés selon le modèle en couches OSI (*Open System Interconnection*). Cinq couches concernent particulièrement les réseaux de capteurs : application, transport, réseau, liaison de données, physique (Benzerbadj, 2018). De même, trois plans composent ces modèles : le niveau de gestion d'énergie, le niveau de gestion des tâches et le de niveau gestion de la mobilité (Pruthi et al., 2019).

#### - *Plan de gestion de l'énergie*

C'est le niveau responsable de la gestion de la consommation énergétique du nœud capteur. Ce plan est dédié à l'optimisation de l'énergie dépensée dans le nœud capteur. Un nœud capteur peut, par exemple se mettre en veille quand il est en

inactivité, il peut s'isoler dès qu'il reçoit un message d'un nœud voisin pour éviter la réception de messages dupliqués.

### - *Plan de gestion de la mobilité*

C'est au niveau de ce plan qu'est assurée la détection et l'enregistrement de la mobilité des nœuds capteurs. Ceci permet de garder du chemin parcouru et de conserver une cartographie à jour des nœuds voisins. Ceci permet d'équilibrer l'exécution des tâches et la consommation d'énergie.

### - *Plan de gestion des tâches*

C'est au niveau de cette couche que sont attribuées les tâches aux nœuds capteurs telles que la détection de données, leur traitement, etc. Il prend en compte tous les nœuds de travail et libres et répartit le travail de manière uniforme.

Les différents nœuds composant le réseau travaillent à des rythmes différents. Leurs tâches sont souvent différentes. C'est dans ce sens que ce niveau sert à l'équilibrage et à la distribution des tâches sur les différents nœuds dans le but d'assurer un travail concurrent, collaboratif et coopératif et efficace.

### 1.6.2- *Système en couches*

Comme souligné ci-dessus, les réseaux de capteurs sans fil sont modélisés selon le modèle en couches OSI (*Open System Interconnection*). Ces cinq couches concernant les réseaux de capteurs sont présentées brièvement ci-dessous :

- La couche application : Cette couche fournit à l'utilisateur une plate-forme pour accéder aux informations et communiquer avec le réseau de capteurs. La fonction de cette couche est de contrôler le trafic et de fournir un logiciel à des fins variées telles que dans les domaines militaires, les secteurs de la santé, etc. qui convertira les données détectées en requêtes pour obtenir des informations spécifiques. Ces informations peuvent ensuite être fournies à l'utilisateur. C'est la couche la plus visible par les utilisateurs et elle permet l'interfaçage avec les applications.
- La couche transport : Les informations fournies par la couche application sont divisés en paquets de données. On distingue différentes étapes : Constitution des paquets, contrôle de flux, ordonnancement des paquets (ordre d'envoi) ainsi que la gestion des erreurs de transmission et les reprises éventuelles. Ces paquets sont transférés à la couche réseau. Divers protocoles ont été conçus pour favoriser le flux de données en amont (utilisateur vers station de base) et en aval (station de base vers utilisateur).

- La couche réseau : Elle est responsable de l'acheminement des paquets à travers le réseau de capteurs. Les protocoles de routage sont naturellement intégrés au niveau de cette couche. Les attaques les plus fréquentes dans les réseaux de capteurs sans fil se situent au niveau de cette couche, ce qui implique une bonne protection et une manipulation sécurisée des paquets de données. Parmi les, on peut citer différents types d'intrusions dans la couche réseau sont : l'attaque Sybil, l'attaque Wormhole, l'attaque Grey Hole, l'attaque Hello Flood, l'attaque Black Hole, l'attaque Sink Hole, ...
- La couche liaison des données : Plusieurs fonctionnalités sont associées à cette couche :
  - Spécification de la manière avec laquelle sont expédiées les données dans le réseau.
  - Multiplexage des données, accès au canal de transmission, contrôle d'erreurs et vérification des messages.
  - Elle fournit une interface logique avec la couche réseau et la couche MAC (accès au médium) qui contrôle les signaux radios. Cette couche fonctionne sur la fusion des flux de données en un seul, en détectant si les données sont encapsulées avant de les transmettre à la couche physique, garantit l'authenticité des données en rectifiant les éventuelles erreurs. La couche MAC (contrôle d'accès au support) est une partie importante de la couche liaison de données. Il prend en charge le flux d'informations via des composants physiques tels qu'une carte d'interface réseau. Le brouillage de la couche de liaison, la collision et la manipulation du trafic sont tous des exemples d'attaques de couche de liaison.
- La couche Physique : C'est à ce niveau que sont précisées les spécifications matérielles de la communication (fréquences porteuses, modulation, etc.). Les techniques d'émission, de réception et de modulation de données de manière robuste sont déterminées au niveau de cette couche. Ce niveau est dépendant de l'appareil physique destiné à transférer et à recevoir les informations. Plusieurs fonctionnalités sont associées à cette couche : génération des fréquences de modulation, extraction des signaux utiles du bruit, encodage et cryptage des données,...

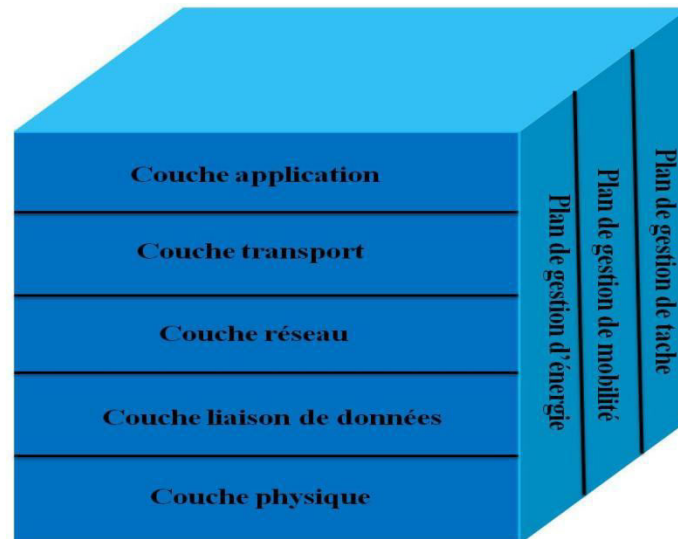


Figure 1.7 - Pile protocolaire dans un réseau de capteurs sans fil.

### 1.7 Applications des réseaux de capteurs sans fil

Les réseaux de capteurs sans fil agissent dans la chaîne d'acquisition, de contrôle et de distribution des données. Les informations recueillies depuis une source donnée sont transformées en grandeurs numériques au niveau des nœuds, elles subissent un premier traitement puis sont acheminées vers des centres de traitements grâce à des protocoles de routage. De nos jours, les applications des réseaux de capteurs sans fil sont multiples et diversifiées (Ramson et al, 2017, Kandris et al., 2020). Nous présentons ci-dessous, de façon sommaire, quelques-unes des applications parmi les plus courantes :

- Applications militaires : Les RCsFs ont été et continuent à l'être appliqués de façon intensive dans le domaine de la surveillance, du contrôle et des stratégies militaires. Il s'agit de déployer de façon dense des milliers de capteurs sans fil dans des zones à surveiller (Ali et al., 2020). Ainsi, tous les mouvements paraissant être suspects (intrusions, mouvement de troupes, objets d'espionnage tels que drones et autres,...) sont décelés en temps réel et envoyés aux centres de traitement. Tout comme, ces réseaux permettent de simuler des situations conflictuelles et permettre de contrôler les réactions face à ces situations.
- Applications médicales : Les RCsFs trouvent, là-aussi, un domaine où leur utilisation est d'une aide très considérable. On distingue la e-santé où les capteurs sont déployés au niveau des lieux de localisation des malades (maisons, voitures, vêtements,...). Dans ce cas, ces capteurs collectent différentes informations médicales concernant les patients : taux de glycémie, rythme cardiaque, pouls, ....ainsi que les états de ces patients : fatigue, stress, prise de médicament,... ces informations sont envoyées vers

des centres de traitement pour un suivi régulier des personnes (Vilela et al., 2020). Ce genre de situation est très bénéfique surtout pour les personnes âgées vivant seules ainsi que les malades chroniques et les personnes isolées. Un deuxième cas où les RCsFs revêtent une grande importance concerne l'exploration des organes humains. Il s'agit là d'introduire dans le corps humain des micro-senseurs sans fils afin qu'ils diagnostiquent l'état des tissus, vaisseaux sanguins et autres organes. Des cas similaires permettent de suivre à distance les évolutions de maladies chez des patients en déployant des micro-capteurs tout le long des organes malades ou des corps des patients. Cet effet est réalisé chez plusieurs espèces animales et des plantes.

- L'agriculture moderne est un domaine d'application très fertile des réseaux de capteurs sans fil (Ramaprasad et al., 2019 ; Nandal and Dahiya, 2021 ). Ces capteurs sont déployés dans de vastes zones agricoles afin de prélever différentes données concernant les conditions du sol, du climat, de l'irrigation,... Une telle configuration permet d'automatiser de façon intensive la production agricole et de faire face aux maladies des plantes et autres aléas, tout en garantissant un rendement optimal. On parle, alors, d'agriculture de précision où tout est calculé, contrôlé et surveillé.
- L'industrie est un autre secteur où l'application des RCsFs procure des bénéfices très appréciables. Dans ce cas de figure, les capteurs sont disséminés tout le long des unités de production à différents points pour collecter un ensemble de données concernant les états des éléments à contrôler (débits, pression, température, ...) (Aalsalem et al., 2019). Ces données sont envoyées vers des centres de commandes sociotechniques où elles sont analysées pour en tirer les décisions qui s'imposent.
- Dans les technologies émergentes telles que l'Internet des Objets (IoT : Internet of Things), les Systèmes Cyber-Physiques (CPS : Cyber-Physical Systems), les RCsFs constituent, de façon générale, le premier maillon de ces systèmes en permettant l'acquisition et la distribution de l'information brute vers les différents objets constituant le reste du système ( Behera et al., 2019 ; Yıldırım and Tatar, 2018).
- Les RCsFs sont utilisés aussi dans l'informatique ambiante et forment le support de base pour la distribution de l'information et du contrôle dans les villes intelligentes (Smart cities), les maisons intelligentes (Smart Homes), les parkings intelligents (Smart Parkings), (Redjimi et al., 2021 ; Sahfutri et al., 2018 )....



Figure 1.8 - Quelques applications courantes des réseaux de capteurs sans fil.

- Les feux de forêts sont un autre domaine où l'utilisation des réseaux de capteurs sans fil apporte une aide indéniable dans la surveillance de vastes régions forestières (Abdulsahib and Khalaf, 2018). Plusieurs mesures sont opérées par les capteurs, le but principal étant de déceler tout départ de feu, voire même les conditions propices aux incendies. Il est à rappeler, le nombre très important de par le monde d'incendies qui causent beaucoup de dégâts et des dommages aussi bien corporels que matériels. Malheureusement et malgré l'utilisation de ces technologies émergentes, nous assistons chaque été à des incendies immenses de par le monde, au sein même de pays très développés tels que les États-Unis d'Amérique.
- Beaucoup d'autres secteurs utilisent les réseaux de capteurs sans fil, il n'est pas possible de les citer tous ici.

### 1.8 Les exigences de sécurité au niveau des capteurs sans fil

Dans une communication radio, le signal traverse l'espace sous la forme d'ondes électromagnétiques pouvant être captée, mesurée et décodée par un récepteur fonctionnant sur des fréquences adéquates. Des intrusions peuvent, ainsi, survenir depuis n'importe quel émetteur/récepteur œuvrant dans les champs d'action de ce

signal et provoquant, ainsi, des menaces sur la sécurité des données échangées. Afin de permettre une traversée des données sûre et sécurisée, certains objectifs doivent être pris en compte tels que la confidentialité des données, l'authenticité, l'intégrité, etc. Ces exigences garantissent que les informations critiques transmises entre les nœuds sont protégées contre tout type de menace.

- La confidentialité des données : c'est l'une des exigences les plus importantes. Son but est de garantir que les informations sont sécurisées et qu'aucune tierce partie non autorisée ne peut accéder à ces informations. Dans les RCsF, les données transitent par de nombreux nœuds, ce qui augmente la probabilité de fuite de données. Un moyen pour assurer la confidentialité des données consiste en leur cryptage et elles ne peuvent être décryptées dans leur forme d'origine que par un destinataire disposant des autorisations suffisantes (clés de décryptage).

-L'intégrité des données : les données d'origine ne doivent en aucun cas être falsifiées ou modifiées, c'est ce qu'on appelle l'intégrité des données. Il est très important que le destinataire obtienne les informations d'origine et non celles modifiées. L'intégrité des données peut être compromise en raison d'un attaquant ou même des conditions environnementales difficiles. L'attaquant peut modifier les données ou injecter de fausses données et les transmettre au récepteur selon ses besoins.

- L'authentification des données : il s'agit du processus de confirmation si le nœud correspond exactement à ce qu'il prétend être. Chaque nœud capteur et chaque station de base doivent être capables de vérifier si les données reçues ont été réellement envoyées par un expéditeur autorisé ou si un attaquant a trompé le nœud autorisé en lui faisant transmettre de fausses données. L'authentification des données peut être réalisée par un code d'authentification de message lors de la transmission de données entre deux nœuds.

- La disponibilité des données : Le but est de garantir que les données sont disponibles et accessibles à chaque demande. Dans les réseaux de capteurs sans fil, il existe de nombreuses raisons pour lesquelles la disponibilité des données n'a pas pu être satisfaite, telles qu'une attaque par déni de service, un nœud de capteur capturé ou des nœuds à court de batterie. Le routage multivoie est une solution communément utilisée pour répondre à cette exigence.

- L'Auto-organisation : En l'absence d'infrastructure, les nœuds capteurs doivent s'organiser de manière significative pour participer à la formation du réseau afin d'assurer la communication avec la base station. Les nœuds s'adaptent aux changements qui se produisent autour d'eux, construisent des topologies, examinent des emplacements particuliers, etc.
- La fraîcheur des données : S'assurer que les données sont récentes et qu'il ne s'agit pas d'anciens messages qui sont rejoués. Une datation des paquets (un compteur de temps) est souvent rajoutée aux paquets pour éviter et résoudre ce problème.
- Sécurisation de la localisation des nœuds : Lorsque les nœuds sont installés de manière aléatoire, leur emplacement est inconnu, ce qui rend très difficile la sécurisation du réseau contre les adversaires. L'attaquant peut modifier les emplacements calculés des nœuds ou prétendre être un nœud légitime même s'il ne l'est pas. Cela peut être évité en installant un système de positionnement global (GPS) sur les nœuds de capteur.

### **1.9 Les menaces de sécurité et leurs contre-mesures dans la couche réseau**

Les réseaux de capteurs sans fil sont très vulnérables à plusieurs types d'attaques, de par leurs constitutions. Ces attaques peuvent survenir de sources distinctes et toucher des couches particulières du système. Il y'a lieu de localiser, analyser et prendre les mesures adéquates pour éviter et éradiquer ces menaces, nous présentons ci-après quelques grands types d'attaques sans pour autant être exhaustifs :

- Les attaques de déni de service (DoS : Denial of Service attacks) : Il s'agit d'attaques (Wood and Stankovic) qui ont pour but de diminuer ou éliminer la capacité d'un réseau dans l'exécution de ses fonctions [6]. Ces attaques peuvent être classées selon leurs impacts au niveau des couches du modèle OSI :
  - Au niveau de la couche physique : Au niveau de cette couche, l'attaque peut consister dans la transmission d'un signal radio qui interfère avec les fréquences radio employées.
  - Au niveau de la couche liaison : L'attaque peut consister à transmettre des messages pour provoquer des collisions avec ceux envoyés par le protocole de routage.
  - Au niveau de la couche réseau : un attaquant peut s'interposer dans un routage multi-saut et refuser de transmettre les messages ou envoyer des messages erronés.

- Au niveau de la couche transport : Cette couche est particulièrement vulnérable aux attaques. Un attaquant peut envoyer plusieurs demandes de connexions et saturer le réseau.

-Les attaques sur les protocoles de routage : Plusieurs attaques peuvent menacer les protocoles de routage pendant le processus d'établissement de la table de routage. On peut citer l'acheminement sélectif ( Selective forwarding ) dans lequel un nœud attaquant peut refuser d'acheminer certains messages voire même les détruire. Par exemple, dans l'attaque du trou noir (Black Hole), un nœud attaquant refuse le routage de tous les messages qui lui parviennent.

- L'attaque sybil : C'est l'une des attaques les plus dangereuses et les plus fréquentes dans les réseaux. Dans cette attaque, un nœud malveillant propage simultanément plusieurs fausses identités, ce qui affecte les protocoles de routage et de nombreuses autres opérations telles que les élections des cluster heads (CHs), la réputation, l'évaluation et l'agrégation de données.

Les attaques Sybil peuvent être classées en trois types principaux :

- a) Communication entre nœuds : On distingue le cas de la communication directe et celui où elle est indirecte. Dans le premier cas, les nœuds pirates peuvent s'interfacer directement avec les nœuds authentiques. Alors que dans le second cas, les nœuds défectueux ne peuvent pas interagir directement avec les nœuds authentiques. Les informations sont acheminées via un ou plusieurs nœuds défectueux, puis transmises à un nœud sybil.
- b) Le type de l'identité acquise par l'attaquant : On distingue les identités fabriquées et les identités volées : un nœud sybil peut soit forger une nouvelle identité, soit la voler à un nœud authentique. Dans le premier cas, l'attaquant marque une identité comme une identité légitime.
- c) La simultanéité : Dans l'attaque simultanée, l'attaquant introduit toutes ses identités à la fois à l'aide desquelles il peut provoquer un effet de vote ou de routage multivoie et tirer les résultats en sa faveur. En cas d'attaque non simultanée, l'attaquant introduit ses identités par lots. Afin de tromper le réseau, l'attaquant peut supprimer ses anciennes identités qui ont endommagé le réseau et acquérir de nouvelles identités.

### 1.10 Les modèles d'énergie

Plusieurs modèles de consommation d'énergie sont proposés dans la littérature. Nous en présentons, ici, deux modèles principaux. Le premier est celui représenté .5figure 1.9 ci-dessous. Ce modèle suit le même modèle radio utilisé dans (Heinzenman et al., 2002). La

quantité d'énergie dissipée par l'émetteur pour transmettre de k bits de données est calculée selon les équations (1.5) et (1.6) :

$$E_{tx}(k, d) = \begin{cases} E_{elec} * k + E_{fs} * k * d^2 & \text{si } d < d_0 \\ E_{elec} * k + E_{amp} * k * d^4 & \text{si } d \geq d_0 \end{cases} \quad (1.5)$$

$$d_0 = \sqrt{\frac{E_{fs}}{E_{amp}}} \quad (1.6)$$

Où  $E_{tx}(k, d)$  est l'énergie dissipée par l'émetteur,  $E_{amp}$  est l'amplificateur de puissance,  $E_{elec}$  est l'énergie des appareils électroniques,  $d$  est la distance de transmission,  $E_{fs}$  est l'énergie d'amplification dans l'espace libre,  $d_0$  est la distance seuil qui dépend de l'environnement et k représente le nombre de bits transmis.

L'équation (1.7) donne l'énergie dissipée par le récepteur pour k bits de données :

$$E_{rx} = kE_{elec} \quad (1.7)$$

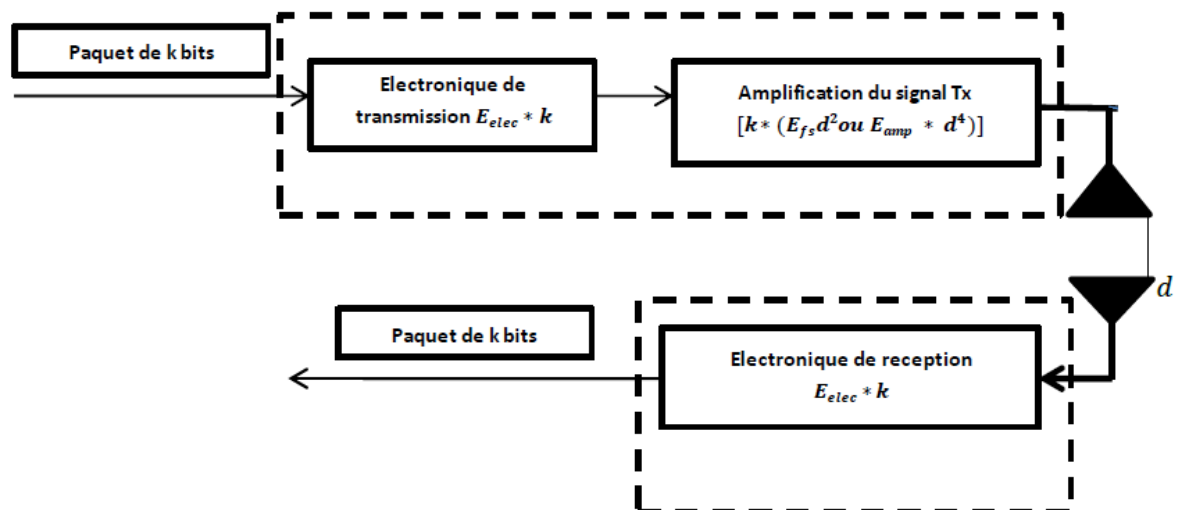


Figure 1.9- Modèle d'énergie (adapté de Heinzelman et al., 2002)

D'autres modèles d'énergie sont adoptés dans l'implémentation des réseaux de capteurs sans fil. Le modèle énergétique proposé par Ivan Stojmenovic (Stojmenovic and Lin, 2001) généralise le modèle de (Rodoplu and Meng 1999) en ajoutant un

facteur linéaire  $a$  et en supposant que la puissance nécessaire pour l'émission et la réception d'un signal est :

$$u(d) = ad^\alpha + c \quad (1.8)$$

Où  $d$  est la distance entre les deux nœuds (nœud émetteur et nœud récepteur),  $\alpha$  est le facteur d'atténuation du média (entre 2 et 6),  $c$  une constante (qui peut aussi inclure l'énergie consommée dans le traitement informatique et le codage/décodage pour chaque nœud), et  $a$  peut être ajusté en fonction de l'environnement physique, l'unité de longueur, la taille unitaire d'un signal,...

### 1.10.1- La zone de captage (Sensing range)

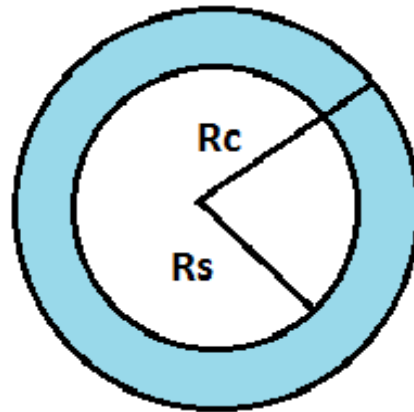
Soient  $n$  nœuds  $S = \{s_1, s_2, \dots, s_n\}$  déployés dans une zone d'intérêt  $Z$ . Chaque nœud capteur est localisé dans l'espace à deux dimensions par ses coordonnées  $(X_i, Y_i)$  où  $i = 1, 2, \dots, n$  et chaque nœud capteur peut détecter les informations situées dans son proche voisinage situé dans un rayon  $R_{s_i}$ . On dira qu'une zone est  $k_{couverte}$  si chaque point de cette région est couvert par au moins  $k$  ( $k > 1$ ) capteurs.

Ainsi toute information située à un point  $P$  éloigné d'une distance  $d$  du capteur  $S_i$  dans le disque de rayon  $R_{s_i}$  peut être détectée par le capteur  $S_i$  si elle satisfait l'équation (1.9) qui détermine les délimitations du rayon d'action du disque  $C_{x,y}(S_i)$  :

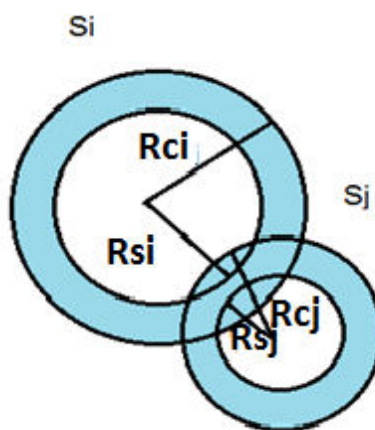
$$C_{x,y}(S_i) = \begin{cases} 1 & \text{si } d(S_i, P) \leq R_{s_i} \\ 0 & \text{sinon} \end{cases} \quad (1.9)$$

### 1.10.2- La zone de connectivité (communication range)

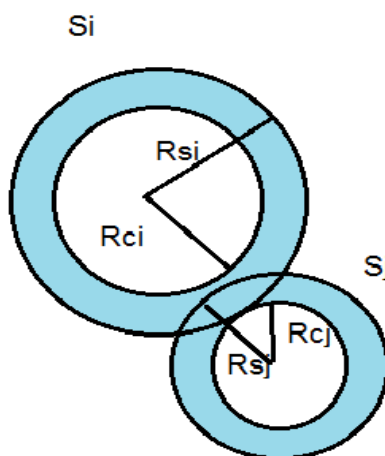
La zone de communication (émission et réception) d'un nœud capteur est comprise dans le disque de rayon  $R_c$  qui délimite cette zone. Ainsi deux nœuds peuvent se connecter si la distance euclidienne les séparant est inférieure ou égale à cette valeur  $R_c$ , soient deux nœuds  $R_{s_i}$  et  $R_{s_j}$ . On peut alors distinguer les différentes situations suivantes :



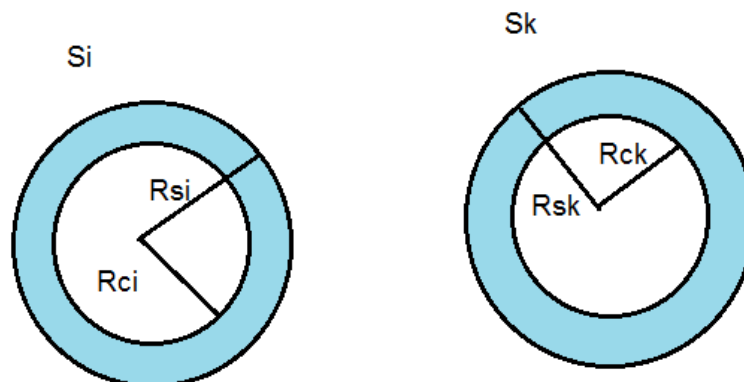
(a) Zones de captage et zone de communication.



(b)  $S_i$  et  $S_j$  peuvent communiquer mais leurs zones de captages sont disjointes.



(c) Les deux capteurs peuvent communiquer ensemble et accéder à la même zone de captage.



(d) Aucune communication n'est possible entre les deux capteurs, leurs zones de captage sont-elles-aussi disjointes.

Figure 1.10- Localisation de capteurs.

### 1.11- Les systèmes d'exploitation dédiés aux RCsFs

Un système d'exploitation est généralement vu comme un gestionnaire de ressources d'un système informatique (Benhaddou and Al-Fuqaha, 2015). Typiquement il s'agit de la gestion des processeurs, des mémoires, des bus, des périphériques de stockage et des entrées/sorties ainsi que d'autres ressources matérielles ou logicielles telles que des réseaux ou autres. La tâche principale d'un système d'exploitation consiste à gérer l'allocation de ces ressources aux utilisateurs de manière ordonnée et contrôlée. Les programmeurs d'applications peuvent ensuite invoquer différents services du système d'exploitation via des appels système. Un système d'exploitation multiplexe les ressources système de deux manières : Dans le temps et dans l'espace. Le multiplexage temporel implique que différents programmes peuvent utiliser à tour de rôle les ressources. Le multiplexage spatial implique que différents programmes accèdent à des parties de la ressource, éventuellement en même temps. Compte tenu des contraintes de ressources des nœuds de capteurs typiques dans un réseau de capteurs sans fil, un système d'exploitation 'classique' ne peut pas être utilisé tel quel mais de nouvelles configurations adaptées à la taille de ces réseaux doit être adoptée et de nouvelles approches sont requises pour la conception des systèmes d'exploitation dans les réseaux de capteurs sans fil.

#### 1.11.1- Enjeux principaux d'un système d'exploitation WSN

L'architecture d'un système d'exploitation a une influence sur la taille de son noyau ainsi que sur la manière dont il fournit des services aux programmes d'application. Parmi les architectures de système d'exploitation les plus courantes, on trouve l'architecture monolithique, l'architecture à micro-noyau, l'architecture de machine virtuelle et l'architecture en couches.

Une architecture monolithique n'a en fait aucune structure. Les services fournis par le système d'exploitation sont implémentés séparément et chaque service fournit une interface pour d'autres services. Une telle architecture permet de regrouper tous les services requis dans une seule image système, ce qui réduit l'empreinte mémoire du système d'exploitation. Un avantage de l'architecture monolithique est que les coûts d'interaction des modules sont faibles. Les inconvénients associés à cette architecture sont les suivants : le système est difficile à comprendre et à modifier, peu fiable et difficile à entretenir. Ces inconvénients associés aux noyaux monolithiques en font un mauvais choix de conception de système d'exploitation pour les nœuds de capteurs contemporains.

La seconde alternative concerne les architectures à micro-noyau. Dans ce cas, un minimum de fonctionnalités est fourni à l'intérieur du noyau. De ce fait, la taille du noyau est considérablement réduite. La plupart des fonctionnalités du système d'exploitation sont fournies via des serveurs de niveau utilisateur, tels qu'un serveur de fichiers, un serveur de mémoire, etc. Si un serveur tombe en panne, l'ensemble du système ne plante pas. L'architecture micro-noyau offre une meilleure fiabilité, une facilité d'extension et de personnalisation. L'inconvénient associé à un micro-noyau est sa faible performance en raison de l'utilisation fréquente des franchissements des limites du noyau. Un micro-noyau est le choix de conception pour de nombreux systèmes d'exploitation embarqués en raison de la petite taille du noyau. Ce genre d'implantation est, ainsi, très favorable pour la gestion des ressources dans un réseau de capteurs sans fil.

Une machine virtuelle est un autre choix architectural. L'idée principale est d'exporter des machines virtuelles vers des programmes utilisateur, qui ressemblent à du matériel. Une machine virtuelle possède toutes les fonctionnalités matérielles nécessaires. Le principal avantage est sa portabilité et son inconvénient réside, généralement, dans une mauvaise performance du système.

Dans le cas particulier des réseaux de capteurs sans fil, le système d'exploitation doit avoir une architecture ayant une petite taille de noyau, donc une faible empreinte mémoire. L'architecture doit permettre des extensions au noyau si nécessaire. L'architecture doit être flexible, c'est-à-dire que seuls les services requis par l'application sont chargés sur le système. Le système d'exploitation doit pouvoir gérer des applications temps réel et être adapté aux exigences de gestion de ces réseaux de capteurs.

### **1.11.2- Modèles de programmation**

On distingue deux modèles de programmation parmi les plus populaires fournis par les systèmes d'exploitation des réseaux de capteurs sans fil typiques, à savoir : la programmation

événementielle et la programmation multithread. Le multithreading est le modèle de développement d'applications le plus familier aux programmeurs, mais dans son vrai sens plutôt gourmand en ressources, il n'est donc pas considéré comme bien adapté aux dispositifs de contrainte de ressources dans les nœuds capteurs. La programmation pilotée par les événements est considérée comme plus utile pour les appareils informatiques équipés de ressources rares, mais n'est pas considérée comme pratique pour les développeurs d'applications traditionnels. Par conséquent, les chercheurs ont concentré leur attention sur le développement de modèles de programmation multithreading légers pour les systèmes d'exploitation des réseaux de capteurs sans fil.

L'ordonnement de l'unité centrale détermine l'ordre dans lequel les tâches sont exécutées. Dans les systèmes informatiques traditionnels, l'objectif d'un planificateur est de minimiser la latence, de maximiser le débit et l'utilisation des ressources, et d'assurer l'équité.

La sélection d'un algorithme d'ordonnement approprié pour les réseaux de capteurs sans fil dépend généralement de la nature de l'application. Pour les applications ayant des exigences en temps réel, un algorithme adéquat doit être utilisé. Pour d'autres applications, des algorithmes d'ordonnement non temps réel suffisent. De plus, les algorithmes d'ordonnement doivent être économes en mémoire et en énergie.

Dans un système d'exploitation traditionnel, la gestion de la mémoire fait référence à la stratégie utilisée pour allouer et libérer de la mémoire pour différents processus et threads. Deux techniques de gestion de la mémoire couramment utilisées sont la gestion de la mémoire statique et la gestion de la mémoire dynamique. La gestion de la mémoire statique est une technique utile lorsqu'il s'agit de ressources mémoire rares. Cependant, il en résulte des systèmes rigides car l'allocation de mémoire en cours d'exécution ne peut pas se produire. Par contre, la gestion dynamique de la mémoire fournit un système plus flexible car la mémoire peut être alloué et libérée au moment de l'exécution. La protection de la mémoire de processus fait référence à la protection de l'espace d'adressage d'un processus contre un autre. Dans les premiers systèmes d'exploitation de réseau de capteurs comme TinyOS (Amjad et al., 2016), il n'y avait pas de gestion de la mémoire disponible. Les systèmes d'exploitation initiaux pour les réseaux de capteurs sans fil supposaient qu'une seule application s'exécutait sur un nœud capteur, il n'y a donc pas besoin de protection de la mémoire. Avec l'émergence de nouveaux domaines d'application pour les réseaux de capteurs sans fil, les systèmes d'exploitation contemporains prennent en charge plusieurs threads d'exécution, par conséquent la gestion de la mémoire devient un problème plus complexe.

Un réseau de capteurs sans fil fonctionne dans un environnement distribué, où les nœuds capteurs communiquent avec d'autres nœuds du réseau. Tous les systèmes d'exploitation des

réseaux de capteurs sans fil fournissent des interfaces de programmation d'application (API) qui permettent aux programmes d'application de communiquer. Il est possible qu'un réseau de capteurs sans fil soit composé de nœuds capteurs hétérogènes, par conséquent le protocole de communication fourni par le système d'exploitation doit également tenir compte de l'hétérogénéité. Dans la communication basée sur le réseau, le système d'exploitation doit fournir des implémentations de protocole de couche transport : réseau et MAC.

L'allocation et le partage des ressources comptent parmi les fonctions les plus importantes que doit assurer un système d'exploitation. La majorité des systèmes d'exploitation des réseaux de capteurs sans fil fournissent, aujourd'hui, une sorte de multithreading, nécessitant un mécanisme de partage des ressources. Cela peut être effectué dans le temps, par exemple, l'ordonnancement d'un processus/thread sur le CPU et dans l'espace, par exemple, l'écriture de données dans la mémoire système. Dans certains cas, on a besoin à un accès sérialisé aux ressources et cela se fait grâce à l'utilisation de primitives de synchronisation.

Un réseau de capteurs sans fil est souvent utilisé pour surveiller des systèmes critiques. Par conséquent, le système d'exploitation utilisé doit fournir des implémentations d'algorithmes d'ordonnancement en temps réel pour respecter des délais de temps de réponse. Les concepteurs des systèmes d'exploitation doivent s'efforcer de fournir des implémentations de protocoles de communication en temps réel au niveau des couches réseau et transport. De plus, un système d'exploitation pour les RCFs doit fournir une interface de programmation d'application (API) au programmeur d'application qui lui permet d'implémenter des protocoles de communication personnalisés au-dessus de la pile des protocoles de communication prise en charge par le système d'exploitation.

Parmi les systèmes d'exploitation dédiés aux réseaux de capteurs sans fil les plus courants, on peut citer les systèmes suivants :

- **Tiny OS** : TinyOS est un système d'exploitation open source, flexible, basé sur des composants et spécifique à l'application, conçu pour les réseaux de capteurs. TinyOS peut prendre en charge des programmes simultanés avec des besoins en mémoire très faibles. Le système d'exploitation a une empreinte qui tient sur 400 octets. La bibliothèque de composants TinyOS comprend des protocoles réseau, des services distribués, des pilotes de capteur et des outils d'acquisition de données (Amjad et al., 2016 ; Patil and Biradar, 2016 ; Chandanashree et al., 2017)

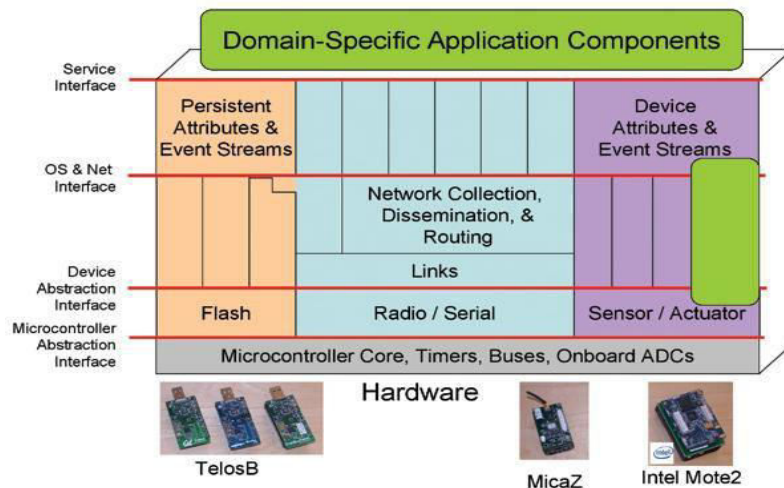


Figure 1.11- Architecture de TyniOS.

- **Contiki :** Contiki (Kurniawan, 2018 ; Sesli and Hacıoğlu, 2017) est un système d'exploitation open source léger écrit en C pour les nœuds de capteurs dans les RCsFs. Contiki est un système d'exploitation hautement portable et il est construit autour d'un noyau piloté par les événements. Contiki fournit un multitâche préemptif qui peut être utilisé au niveau de processus individuels. Une configuration Contiki typique consomme 2 kilo-octets de RAM et 40 kilo-octets de ROM. Une installation complète de Contiki comprend des fonctionnalités telles que : noyau multitâche, multithreading préemptif, proto-threads, réseau TCP/IP, IPv6, une interface utilisateur graphique, un navigateur Web, un serveur Web personnel, un client telnet simple, un économiseur d'écran et un réseau virtuel.

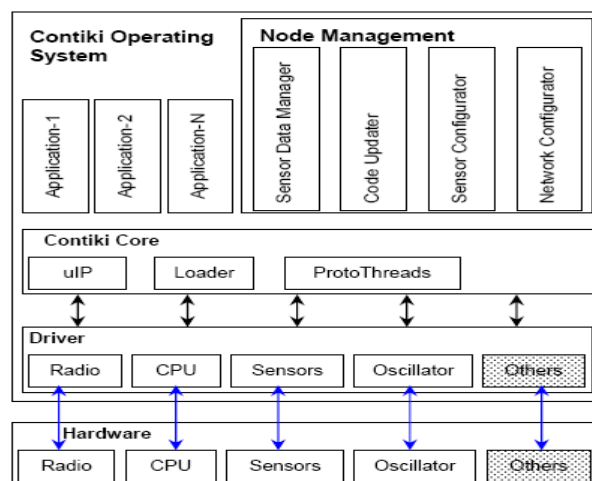


Figure 1.12- Architecture générale de Contiki

- **MANTIS :** Le système multimodal pour les réseaux de capteurs sans fil in situ (Multimodal system for NeTworks of In-situ: MANTIS) est un système d'exploitation multithread pour les RCsFs. MANTIS est un système d'exploitation léger et économe en

énergie. Il a une empreinte de 500 octets, qui comprend le noyau, l'ordonnanceur et la pile réseau (Ranjan et al., 2015 ; Bala et al., 2018).

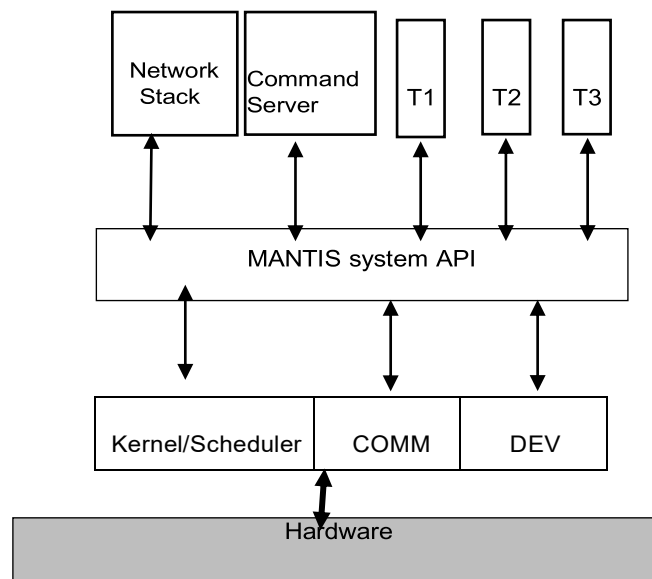


Figure 1. 13- Architecture de MANTIS

La principale caractéristique du système d'exploitation MANTIS (MOS : MANTIS Operating System) est qu'il est portable sur plusieurs plates-formes, c'est-à-dire que nous pouvons tester les applications MOS sur un PDA ou un PC. Ensuite, l'application peut être portée sur le nœud du capteur. MOS prend également en charge la gestion à distance des nœuds de capteurs grâce à la programmation dynamique. MOS est écrit en C et prend en charge le développement d'applications en C.

**Nano-RK :** Nano-RK (Eswaran et al., 2005 ; Patel and Shah, 2021) est un système d'exploitation en temps réel multitâche fixe et préemptif pour les RCsFs. Les objectifs de conception de Nano-RK sont le multitâche, la prise en charge de la mise en réseau multi-sauts, la prise en charge de la planification basée sur les priorités, la rapidité et la capacité de planification, la durée de vie étendue du réseau, les limites d'utilisation des ressources d'application et la faible empreinte. Nano-RK utilise 2 Ko de RAM et 18 Ko de ROM. Nano-RK prend en charge les réservations de CPU, de capteurs et de bande passante réseau. Nano-RK prend en charge les applications en temps réel dures et logicielles au moyen de différents algorithmes d'ordonnancement en temps réel, par exemple, l'ordonnancement monotone et l'ordonnancement harmonisé. Nano-RK fournit un support réseau via une abstraction de type socket. Nano-RK prend en charge les plates-formes de détection FireFly et MicaZ.

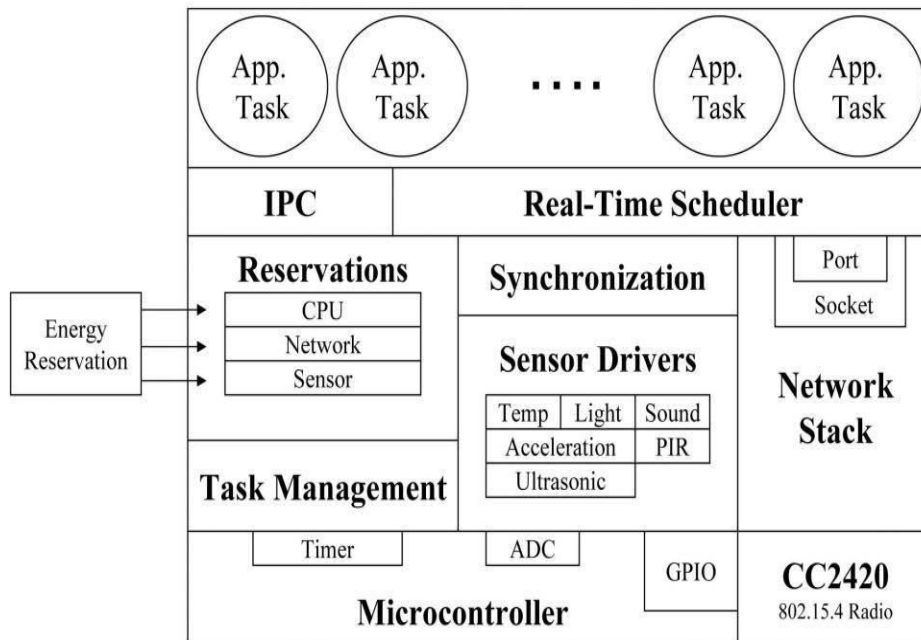


Figure 1.14- Architecture de Nano RK.

- **LiteOS** : LiteOS (Yaqoob et al., 2019 ; Burhan et al., 2018 ; Samridhi and Liscano, 2020) est un système d'exploitation de type Unix conçu pour les RCsFs à l'Université de l'Illinois à Urbana-Champaign. Les motivations derrière la conception de ce nouveau système d'exploitation de type Unix est de fournir aux programmeurs système un paradigme de programmation familier (mode de programmation basé sur les threads, bien qu'il fournisse un support pour enregistrer les gestionnaires d'événements à l'aide de rappels), un système de fichiers hiérarchique, prise en charge de la programmation orientée objet sous la forme de LiteC++ et d'un shell de type Unix. L'empreinte de LiteOS est suffisamment petite pour fonctionner sur des nœuds MicaZ dotés d'un processeur de 8 MHz, de 128 octets de programme flash et de 4 Ko de RAM. LiteOS est principalement composé de trois composants : LiteShell, LiteFS et le noyau.

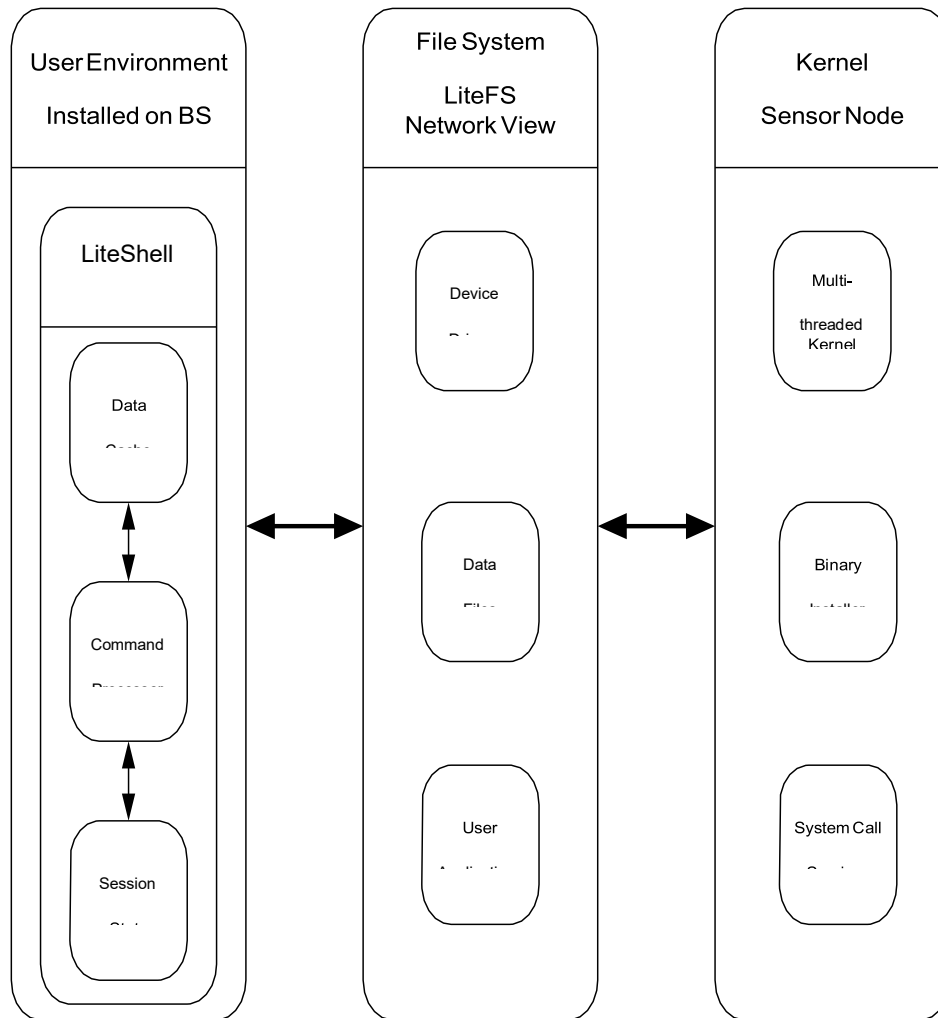


Figure 1. 15- LiteOs

### 1.11.3- Tableau récapitulatif

Le tableau suivant (Tableau 1.1 ) résume les caractéristiques des systèmes d’exploitation présentés ci-dessus :

Tableau 1.1- Comparaison entre les systèmes d’exploitation des RCsF.

Système d’exploitation	Architecture	Gestion mémoire et protection	Partage de ressources	Temps réel
TinyOS	Monolithique	Gestion statique de la mémoire avec protection.	Virtualisation et evenements	Non
Contiki	Modulaire	Gestion dynamique de la mémoire avec liens sans protection	Accès sérialisés	Non
MANTIS	En couche	Gestion dynamique de la mémoire avec liens sans protection	Utilisation de sémaphores	Dans une certaine mesure Au niveau de l’ordonnancement
Nano-RK	Monolithique	Gestion statique de la mémoire sans protection	Accès sérialisé via mutuelle exclusion et sémaphores. Fournir une	Oui

			implémentation de l'algorithme de plafond de priorité pour l'inversion de priorité	
LiteOS	Modulaire	Gestion dynamique de la mémoire avec protection des processus.	A l'aide de primitives cde synchronisation	Non

## 1.12- Conclusion

Les réseaux de capteurs sans fil sont composés de nœuds capteurs équipés de ressources réduites pour le traitement et le stockage de l'information. Employé séparément, chaque nœud capteur ne présente que de très faibles capacités de traitement, de stockage et de transmission de l'information, mais dans le cas d'un emploi massif de tels unités et profitant de la distribution du réseau de capteurs, le rendement de ces systèmes est très appréciable. Les applications des réseaux de capteurs sans fil dans divers domaines de la vie courante domestique et professionnelle apportent un grand confort et une grande aide dans les entreprises de tous les jours. Ces réseaux fonctionnent invariablement en mode sans surveillance et dans de nombreux scénarios, il est impossible d'accéder à ces nœuds capteur après leur déploiement, donc un objectif fondamental est d'optimiser la durée de vie des batteries de ces capteur. Ces caractéristiques imposent des défis supplémentaires à la conception des systèmes de gestion de ces réseaux. Le but de ce premier chapitre est de présenter et mettre en évidence quelques notions de base concernant les principales caractéristiques de ces systèmes. Le prochain chapitre tentera de présenter un ensemble d'aspects concernant les protocoles de communication dans les réseaux de capteurs sans fil.

2.1-	Introduction
2.2-	Le routage dans les RCsF
2.3-	Les protocoles de routage
2.3.1-	Classification des protocoles de routage
2.3.2-	Les protocoles hiérarchiques homogènes
2.3.2.1-	Le protocole LEACH
2.3.3-	Les protocoles de routage hétérogènes
2.3.3.1-	Le protocole DEEC
2.3.3.2-	Le protocole EDEEC
2.3.3.3-	Simulation
2.4-	Cas d'application : Conception et mise en œuvre d'une application de gestion d'un parking intelligent (Smart Parking).
2.4.1-	Approche proposée
2.4.2-	Le modèle RENEW obtenu
2.4.3 -	La mise en œuvre utilisant la plateforme JESS du modèle
2.4.4-	Discussion
2.5-	Conclusion

---

### **2.1 Introduction**

Dans un réseau de capteurs sans fil, l'information est collectée à la source sous sa forme physique brute (température, humidité, pression, lumière, ...), elle est numérisée puis subit des traitements locaux et acheminée vers le centre de traitement (la station de base).

Plusieurs contraintes limitent ces machines :

- Capacités de traitement et de stockage : Les composants matériels d'un nœud capteur (processeurs dédiés, mémoires, registres, bus,...) sont, généralement , assez rudimentaires avec des performances limitées.
- Énergie : Généralement des batteries non rechargeables et non remplaçables alimentent ces nœuds capteurs.
- Propagation du signal : Le rayon de propagation du signal est limité et il est atténué en fonction de la distance parcourue.
- Bruit : Des bruits divers peuvent s'ajouter au signal transmis, ce qui peut en perturber la qualité.
- Obstacles : Divers obstacles pouvant se trouver entre une station émettrice et une autre réceptrice peuvent empêcher, atténuer ou détourner le rayonnement du signal (bâtiments, humidité, eau,...)

- De par la nature même des liens radios, il n'existe pas d'architecture prédéfinie et fixe du réseau, c'est une infrastructure AD-HOC qui change selon l'évolution des besoins de communication des nœuds qui peuvent être mobiles.
- Les communications sans fil peuvent être piratées et écoutées facilement par des tiers mal intentionnés, la sécurité de ces réseaux compte parmi les défis majeurs pour les RCsF. Cependant, vu les capacités de traitement et de stockage limitées de ces nœuds capteurs, des algorithmes complexes ne peuvent pas y être logés et exécutés de façon adéquate. Ceci peut limiter les choix d'implémentation aux algorithmes de complexités réduites en longueur de code, en temps d'exécution et en nombre de ressources requises pour leurs exécutions.
- La nature même des données récoltées est un élément déterminant du nombre de capteurs à déployer dans une zone d'intérêt ainsi que de leur mode de déploiement, de gestion, d'organisation et de fonctionnement.

### **2.2 Le routage dans les RCsF**

Il s'agit, là, de considérer l'itinéraire de l'acheminement de l'information depuis un nœud capteur vers la station de base. Des contraintes principales sont à prendre en considération dans ce cas :

- L'éloignement du nœud capteur de la station de base : La station de base peut être située hors de la portée du signal émis par un nœud capteur. Tout comme, cette station peut être fixe ou mobile.
- Des obstacles peuvent empêcher le signal émis par le nœud capteur d'arriver à sa destination.
- Le coût énergétique du signal émis est fonction de l'éloignement entre la station de base et le nœud capteur.

Plusieurs solutions peuvent être envisagées comme l'ajout de relais (figure 2.1 (d)) ou de nœuds capteurs supplémentaires (figure 2.1 (c)) ce qui induit des coûts supplémentaires. Les solutions les plus avantageuses résident dans la mise en œuvre d'algorithmes permettant des configurations dynamiques du réseau et l'emploi de stratégies optimisant les coûts du routage de l'information. Ainsi, au lieu que l'information soit transmise en utilisant un seul saut (one hop) figure 2.1 (a), l'utilisation de routages multi-sauts (multi-hop) figure 2.1 (b) est très souvent plus avantageux.

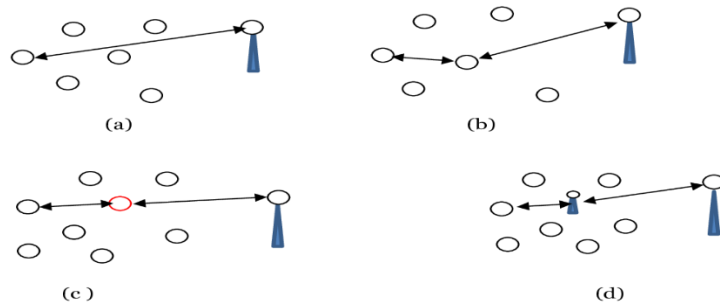


Figure 2.1 routage de l'information : (a) mono-saut, (b) multi-saut. (c) ajout d'un nœud capteur, (d) ajout d'un relai.

Dans un RCsF, la propagation des données et leur délivrance depuis une station émettrice vers une station de destination comptent parmi les plus importantes fonctionnalités. Toutes les caractéristiques statiques et dynamiques du réseau visant à assurer les meilleures performances en termes de durée de vie, de sécurité, de temps de réponse, de fiabilité,... doivent être prises en considération. Cette thématique concernant la conception et l'implémentation de protocoles de routage dans les RCsF donne lieu à un nombre impressionnant de recherches.

Afin de gérer de façon optimale les configurations dynamiques en l'absence de toute infrastructure dans les RCsF, plusieurs protocoles de routage utilisant des algorithmes opérant de façons spécifiques sont régulièrement proposés.

### 2.3 Les protocoles de routage

Un protocole de routage pour les réseaux de capteurs sans fil consiste à trouver le chemin optimal et à établir la route correcte et efficace entre une paire de nœuds en respectant les facteurs de consommation d'énergie, de durée de vie du réseau ainsi que la garantie de l'acheminement correct de l'information (Yan and Ding, 2016). La communication directe entre un nœud capteur et la station de base (BS) n'est pas, de façon générale, une solution efficace car elle consomme beaucoup d'énergie et la station de base peut être située en dehors de la portée du signal du nœud. Ainsi, plusieurs protocoles ont été proposés et nombre d'entre eux sont mis en œuvre.

#### 2.3.1 Classification des protocoles de routage

On distingue, généralement, trois classes importantes de protocoles : les protocoles de routage plats, hiérarchiques et basés sur la localisation. On peut distinguer deux catégories dans les protocoles de routage plats qui sont les protocoles à la demande (tels que Dynamic

Source Routing (DSR) (Johnson et al., 2001) et Ad hoc On-demand Distance Vector (AODV) (Perkins and Royer, 1999) ou les protocoles Table-Driven tels que Destination Sequenced Distance Vector (DSDV) (Abdulrahman, 2019). Dans le cas des protocoles hiérarchiques, les nœuds sont regroupés sous forme de clusters et un cluster head (CH) est élu périodiquement pour recevoir les données des différents constituants du cluster, les agréger puis les acheminer vers la station de base. Le protocole LEACH (Low Energy Adaptable Clustering Hierarchy) (Heinzelman et al., 2001) et ses variantes (Singh et al., 2017) sont parmi les implémentations typiques de cette classe de protocoles. Une autre classe intéressante de protocoles est le routage basé sur la localisation. ( protocoles dits géographiques). Dans ce cas, chaque nœud transmet ses données à son voisin le plus proche qui est en direction de la station de base. Par exemple, Greedy Perimeter Stateless Routing (GPSR) (Karp and Kung, 2000) et Improved Energy-aware and Delivery Guarantee Geographic Routing (IEGGR) (Redjimi et al., 2021) sont des protocoles géographiques. Dans le routage géographique, chaque nœud ne s'intéresse qu'à ses voisins et effectue des routages à un seul saut.

On peut considérer deux catégories de nœuds capteurs dans un réseau de capteurs sans fil : les nœuds homogènes et les nœuds hétérogènes. Dans la première catégorie, tous les nœuds ont une même quantité d'énergie initiale et les mêmes composants matériels, alors que pour les réseaux hétérogènes, il existe différents niveaux de nœuds équipés de différentes quantités d'énergies initiales, les ressources matérielles peuvent être, elles-aussi, différentes.

De nombreux protocoles de réseaux de capteurs sans fil homogènes et hétérogènes sont développés. Parmi les premiers protocoles homogènes figurent LEACH (Heinzelman et al., 2001), Power-Efficient Gathering in Sensor Information Systems (PEGASIS) (Lindsey and Raghavendra, 2002) et Hybrid Energy-Efficient Distributed Clustering (HEED) (Younis and Fahmy, 2004). Les protocoles homogènes ne sont pas efficaces dans les réseaux hétérogènes. Dans les WSN hétérogènes (HWSN), les nœuds ont des niveaux d'énergie initiaux différents, les composants logiciels et matériels des nœuds peuvent être différents ou la force des signaux radio des nœuds peut être différente. La quantité initiale d'énergies des nœuds capteurs est le cas d'hétérogénéité le plus étudié. Parmi ce type de protocoles, on peut distinguer les suivants : Distributed Energy Efficient Clustering (DEEC) (Qing et al., 2006) et ses variantes (Qureshi et al., 2012) : Developed DEEC (DDEEC) (Elbhiri et al., 2010), Enhanced Developed Distributed Energy-efficient Clustering (EDDEEC) (Javaid et al., 2013) et Enhanced DEEC (EDEEC) (Saini and Sharma, 2010). Le protocole d'élection stable «Stable Election Protocol : SEP (Aderohunmu and Deng, 2009) , Balanced Energy Efficient Network Integrated Super Heterogeneous (BEENISH) (Qureshi et al., 2013) et Modified

BEENISH (MBEENISH) (Preethi et al., 2016) sont d'autres protocoles pour les WSN hétérogènes.

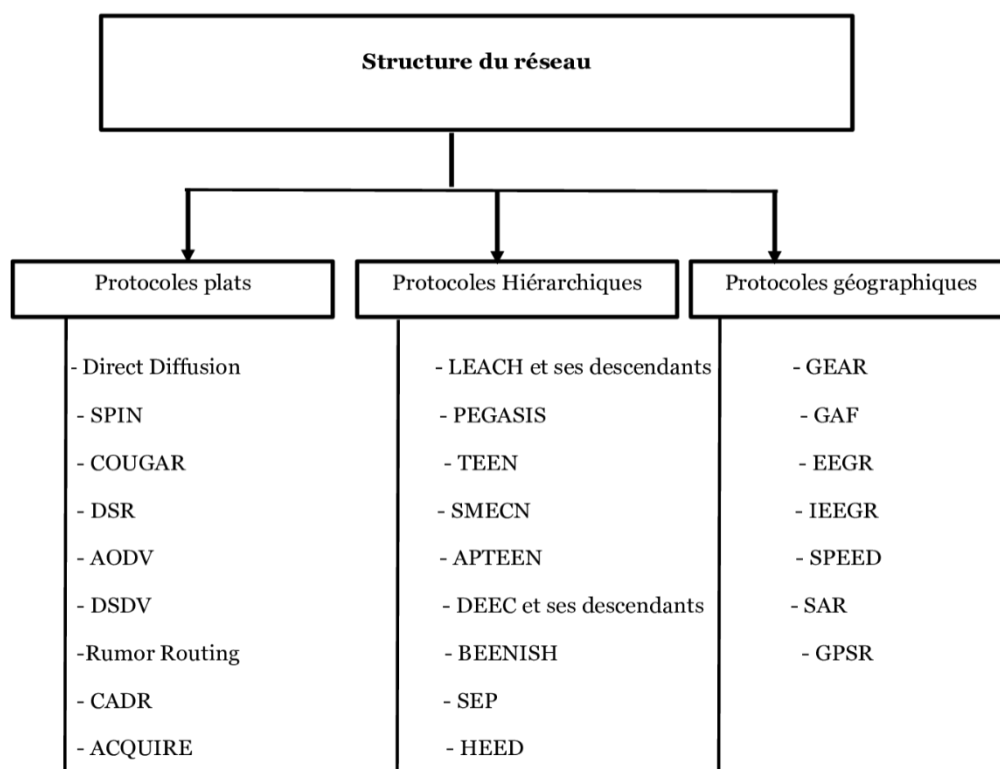


Figure 2.2- Classification de protocoles de routage

### 2.3.2- Les protocoles hiérarchiques homogènes

On parle de réseau de capteurs homogènes lorsque tous les composants du réseau ont des caractéristiques 'identiques' : Tous les nœuds capteurs sont disposent de la même quantité initiale d'énergie, ils ont tous des ressources matérielles et logicielles identiques (mémoire, CPU, ..., software), leurs champs d'action sont, eux-aussi, identiques. Cependant, le critère qui semble être le plus utilisé dans ce genre de classement repose sur les quantités d'énergie détenues par chaque nœud capteur. Il semble évident de dire que selon les dépenses énergétiques dues surtout à la transmission des informations, des variations quantitatives en réserves d'énergie affectent de façon inégales les nœuds du réseau ce qui ne permet pas de garantir une distribution énergétique équitables entre les différents nœuds. Une telle constatation implique que cette notion d'homogénéité dans la distribution énergétique ne peut être possible que suite à des efforts d'optimisation très conséquents.

#### 2.3.2.1- Le protocole Low Energy Adaptive Clustering Hierarchy (LEACH)

LEACH (Low Energy Adaptive Clustering Hierarchy) compte parmi les premiers protocoles hiérarchiques (Heinzelman et al.2002). LEACH est un protocole conçu pour les réseaux de capteurs sans fil homogènes qui fonctionne par rounds. A chaque round, les nœuds capteurs

sont regroupés en 'clusters' et un cluster head (CH) est élu parmi les nœuds capteurs n'ayant pas servi comme clusters dans les rounds précédents. Initialement, une proportion  $p$  de clusters est prédéterminée pour l'ensemble du réseau. Au premier round tous les nœuds capteurs sont des candidats potentiels pour devenir clusters head avec une chance de  $\frac{1}{p}$  chacun. Lors des prochains rounds les nœuds ayant été déjà élus ne seront plus considérés comme candidats potentiels pour devenir CHs. Un cluster se compose d'un seul CH et d'un nombre quelconque de membres qui ne communiquent qu'avec leur chef de cluster. Le clustering est une approche qui facilite la gestion du réseau et l'agrégation des données qui se font au niveau du CH réduisant ainsi la quantité de données devant être transmises à la station de base.

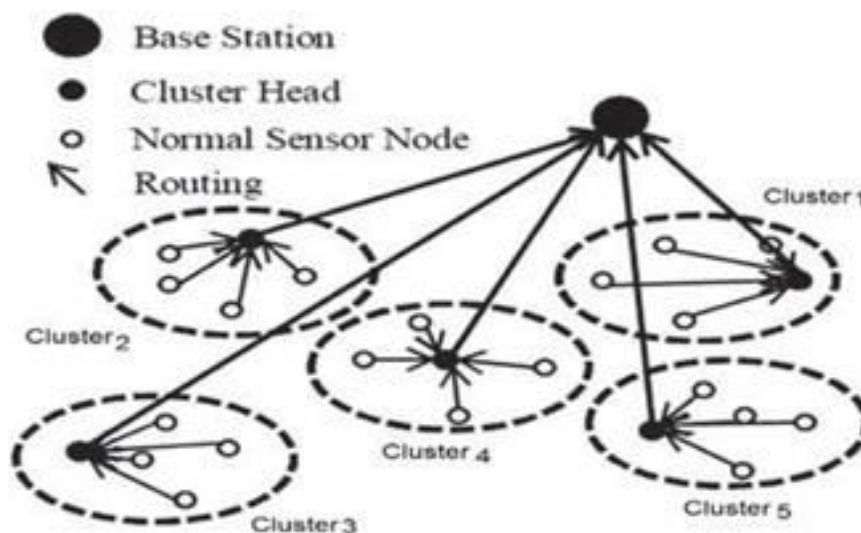


Figure 2.3- Le clustering dans LEACH.

LEACH fonctionne selon des cycles composés de deux phases : une phase de configuration (*setup phase*) et une phase de régime permanent (*steady-up phase*). La première phase est consacrée à la construction des clusters : élection du CH et constitution des membres du cluster. A cet effet, une fois le CH élu, ce dernier envoie des messages aux membres et tous les nœuds capteurs qui reçoivent ce message répondent au CH, lequel CH désigne les membres de son cluster selon la force du signal reçu. La deuxième phase consiste dans la transmission des paquets depuis les membres des clusters vers leurs CHs respectifs. LEACH combine une communication sans contention de style TDMA (Time Division Medium Access) où le CH alloue à chaque nœud capteur un quantum de temps pour qu'il envoie ses données, avec un algorithme de clustering. Ces deux phases sont exécutées à chaque round ce qui provoque une dépense énergétique répétitive.

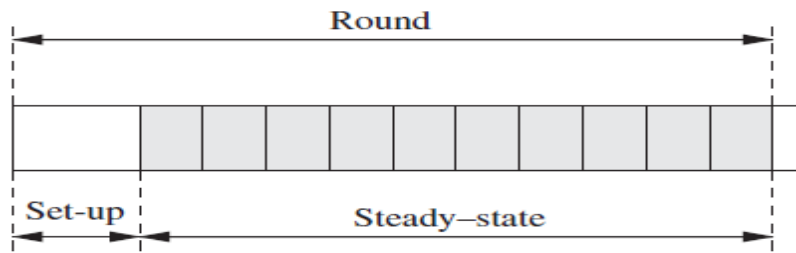


Figure 2.4- Phases dans LEACH.

### a- La phase de configuration

Au cours de la phase de configuration, les CHs sont déterminés et les tranches de temps de communication au sein de chaque cluster sont établies. Étant donné que le CH est responsable de la coordination de l'activité du cluster et de la transmission des données à la station de base, ses besoins énergétiques seront considérablement importants par rapport aux autres nœuds capteurs. Par conséquent, LEACH fait tourner la responsabilité du chef de cluster entre les nœuds de capteurs pour répartir uniformément la charge énergétique. En effet, au début de chaque tour, chaque capteur  $i$  candidate pour être élu CH avec une certaine probabilité  $P_i(t)$ . Dans un réseau avec  $N$  nœuds et un nombre souhaité de chefs de cluster de  $k$ , les probabilités peuvent être choisies pour satisfaire (2.1) :

$$\sum_{i=1}^N P_i(t) = k \quad (2.1)$$

L'équation (2.2) ci-dessous détermine la méthode de calcul de  $P_i(t)$  au round  $r$  :

$$P(S_i) = \begin{cases} \frac{k}{N - k(\text{mod}(r, \frac{N}{k}))} & \text{si } C_i(t) = 1; \\ 0 & \text{si } C_i(t) = 0; \end{cases} \quad (2.2)$$

Cette approche utilise une fonction indicatrice  $C_i(t)$  qui sera égale à 1 lorsque le nœud capteur n'a pas été CH auparavant (dans les tours précédents :  $r \text{ mod}(N/k)$ ) d'un cluster auquel cas elle sera égale à 0 et le nœud en question ne peut candidater à être cluster head. Cette approche de sélection des chefs de cluster vise à répartir équitablement la responsabilité du chef de cluster, et, par conséquent, le surcoût énergétique, entre tous les nœuds capteurs. Cependant, cela ne tient pas compte de la

quantité réelle d'énergie disponible pour chaque nœud. Par conséquent, une approche alternative pour déterminer la probabilité de devenir chef de cluster peut être utilisée (2.3):

$$P_i(t) = \min \left\{ \frac{E_i(t)}{E_{total}(t)} k, 1 \right\} \quad (2.3)$$

Dans cette équation  $E_i(t)$  représente l'énergie résiduelle (actuelle) du nœud  $i$  et  $E_{total}(t)$  est la somme des niveaux d'énergie de tous les nœuds. Un inconvénient de cette approche est que chaque nœud doit connaître (ou estimer)  $E_{total}(t)$ .

Une fois qu'un nœud capteur a déterminé qu'il servira de chef de cluster pour le tour suivant, il informe les autres nœuds capteurs de son nouveau rôle en diffusant un message publicitaire (ADV) à l'aide d'un protocole CSMA non persistant. Chaque nœud capteur rejoint un cluster en sélectionnant la tête de cluster qui peut être atteinte avec la plus petite quantité d'énergie de transmission (basée sur la force du signal reçu des messages ADV des têtes de cluster) et en transmettant un message de demande de jointure (Join-REQ) au chef de cluster choisi (encore une fois, en utilisant CSMA). Le cluster head établit un planning de transmission pour son cluster et transmet ce planning à chaque nœud de son cluster.

### **b- La phase de régime permanent**

Un nœud capteur communique uniquement avec le chef de groupe et est autorisé à transmettre des données uniquement pendant les créneaux alloués indiqués par le programme reçu du chef de groupe. Il est alors de la responsabilité du chef de cluster de transmettre les données de capteur provenant de l'un de ses nœuds de capteur à la station de base. Pour préserver l'énergie, chaque membre du cluster utilise la puissance de transmission minimale requise pour atteindre le chef de cluster et éteint la radio sans fil entre ses emplacements désignés. D'autre part, le chef de cluster doit être éveillé à tout moment pour recevoir les données des capteurs des membres de son cluster et pour communiquer avec la station de base.

Alors que la communication intra-cluster est sans contention à l'aide de trames et d'emplacements de type TDMA, la communication se produisant dans un cluster peut toujours interférer avec la communication dans un autre cluster. Par conséquent, les nœuds capteurs utilisent la technique DSSS (Direct Sequence Spread Spectrum) pour limiter les interférences entre les clusters, c'est-à-dire que chaque

cluster utilise une séquence d'étalement différente de la séquence d'étalement utilisée dans les clusters voisins. Une autre séquence réservée est utilisée pour la communication entre les chefs de cluster et la station de base. La communication entre les chefs de cluster et la station de base est basée sur ce code d'étalement fixe et la technique CSMA. Avant qu'un chef de cluster ne transmette des données, il détecte d'abord le canal pour voir s'il y a une transmission en cours utilisant le même code d'étalement.

Plusieurs variantes du protocole LEACH ont été proposées et implémentées ; on parle de descendants de LEACH. Ces variantes ont pour but d'améliorer ce protocole ou d'en combler certaines lacunes ou encore certaines sont destinées à des applications particulières. Une variante de ce protocole, appelée LEACH-C, s'appuie sur la station de base pour déterminer les chefs de cluster. Cela se produit pendant la phase de configuration, où chaque nœud capteur transmet son emplacement et ses niveaux d'énergie à la station de base. Sur la base de ces informations, la station de base détermine les chefs de cluster et informe les chefs de cluster de leur nouveau rôle. D'autres nœuds capteurs peuvent alors rejoindre des clusters en utilisant des messages de jointure comme décrit dans le protocole LEACH d'origine.

En résumé, LEACH utilise une variété de techniques pour réduire la consommation d'énergie (énergie de transmission minimale, évitant l'écoute inactive des membres du cluster) et pour obtenir une communication sans contention (communication basée sur la planification, DSSS). Alors que la communication intra-cluster est sans contention et que les interférences entre les clusters sont évitées, la communication entre les têtes de cluster et la station de base est toujours basée sur CSMA. De plus, LEACH suppose que tous les nœuds sont capables d'atteindre la station de base, ce qui affecte l'évolutivité de ce protocole. Cependant, cela peut être résolu en ajoutant la prise en charge du routage multi-sauts entre la station de base et tous les chefs de cluster.

### **2.3.3- Les protocoles de routage hétérogènes**

Les protocoles de routage hétérogènes pour les RCsF sont classés selon plusieurs critères tels que l'efficacité énergétique, le nombre de niveaux d'hétérogénéité, la stabilité et la sélection du chef de clusters (CH). Une gestion efficace de l'énergie doit être utilisée pour choisir efficacement le chef de cluster approprié, car elle améliore la durée de vie du réseau dans une

certaine mesure. Dans ce qui suit, nous présentons deux protocoles de routage pour les RCSFs hétérogènes DEEC et EDEEC.

### 2.3.3.1- Le protocole hétérogène 'Distributed Energy Efficient Clustering (DEEC)' pour les RCSFs

Li Qing, Qingxin Zhu et Mingwen Wang (Qing et al, 2006) ont proposé le protocole hétérogène Distributed Energy Efficient Clustering Protocol for Heterogeneous Wireless Sensor Networks (DEEC), qui est un algorithme de clustering à deux niveaux pour les RCSF hétérogènes. DEEC utilise deux niveaux d'énergie : normal pour les nœuds à faible énergie et avancé pour les nœuds équipés de niveaux d'énergie plus élevés. Pour sélectionner un CH, DEEC utilise les niveaux d'énergie initiale et résiduelle des nœuds.

Les CHs sont sélectionnés selon une probabilité basée sur le rapport entre l'énergie résiduelle de chaque nœud et l'énergie moyenne du réseau. Il existe différentes quantités d'énergies initiales pour les nœuds du réseau de capteurs.

Soient  $E_0$  et  $(1 + a)E_0$  les énergies initiales d'un nœud normal et d'un nœud avancé ( $a$  est un coefficient multiplicateur qui la proportion d'énergie des nœuds avancés par rapport aux nœuds normaux ( $a > 1$ ). Soit  $N$  le nombre total de capteurs du réseau (2.4).

$$N = N_{nml} + N_{advc} = N(1 - m) + Nm \quad (2.4)$$

Où  $N_{nml}$  et  $N_{advc}$  représentent respectivement le nombre total de nœuds normaux et avancés et  $m$  la fraction des nœuds avancés ( $0 < m < 1$ )

Les énergies premières totales des nœuds normaux et avancés sont données respectivement par les équations (2.5) et (2.6)

$$E_{nml} = N_{nml}E_0 \quad (2.5)$$

$$E_{advc} = N_{advc}E_0 + aN_{advc}E_0 = (1 + a)N_{advc}E_0 \quad (2.6)$$

Où  $a$  est un facteur multiplicatif de l'énergie initiale  $E_0$  et  $a > 1$ .

L'équation (2.7) calcule l'énergie première totale :

$$\begin{aligned} E_{total} &= E_{nml} + E_{advc} = (1 - m)E_0 + Nm(E_0 + aE_0) = NE_0(1 - m + m + ma) \\ &= NE_0(1 + ma) \end{aligned} \quad (2.7)$$

L'équation (2.8) donne l'énergie moyenne du réseau pour le  $r^{ième}$  tour :

$$E_{avg} = \frac{1}{N} E_{total} \left(1 - \frac{r}{R}\right) \quad (2.8)$$

$E_{total}$  est l'énergie totale de tous les nœuds (les  $N$  nœuds),  $R$  est le nombre de tours pour le réseau vivant.  $E_{round}$  représente l'énergie consommée pour chaque tour (2.9).

$$R = \frac{E_{total}}{E_{round}} \quad (2.9)$$

Soit  $P_i$  la probabilité qu'un nœud  $S_i$  devienne CH au tour  $r$ ,  $G$  l'ensemble des nœuds éligibles au tour  $r$ . Au début de chaque tour, chaque nœud propose un nombre probabiliste aléatoire (entre 0 et 1) et si ce nombre est inférieur au seuil ( $T(S_i)$ ) alors  $S$  est éligible pour devenir CH (équation 2.13).

Pour le protocole DEEC, la probabilité pour les nœuds normaux et avancés est donnée par l'équation suivante (2.12).  $P_{opt}$  est une valeur déterminée qui représente la fraction du nombre de clusters dans le réseau ( $0 < P_{opt} < 1$ ).  $P_{opt}$  peut être calculé en utilisant (2.10) et (2.11), où  $N$  et  $M$  sont respectivement le nombre de nœuds du réseau et la taille de ce réseau.

$$k_{opt} = \sqrt{\frac{N}{2\pi}} d_0 \frac{M}{d^2} \quad (2.10)$$

$$p_{opt} = \frac{k_{opt}}{N} \quad (2.11)$$

$$P_i = \begin{cases} \frac{E_i(r)P_{opt}}{(1+am)E_{avg}} & \text{if normal node} \\ \frac{E_i(r)P_{opt}(1+a)}{(1+am)E_{avg}} & \text{if advanced node} \end{cases} \quad (2.12)$$

$$T(S_i) = \begin{cases} \frac{P_i}{1 - P_i(\text{mod}(r, \frac{1}{P_i}))} & \text{if } S_i \in G \\ 0 & \text{otherwise} \end{cases} \quad (2.13)$$

### 2.3.3.2- Le protocole EDEEC (Enhanced Distributed Energy Efficient Clustering) pour les RCsF hétérogènes

Parul Saini, Ajay.K. Sharma [18] ont proposé le protocole Enhanced Distributed Energy Efficient Clustering (EDEEC). Ce protocole est un algorithme de clustering à trois niveaux destiné à augmenter la durée de vie et la stabilité du réseau. Le principe de fonctionnement de ce protocole est similaire à celui de DEEC. Cependant, il existe trois types de nœuds : les nœuds normaux, les nœuds avancés et les super nœuds selon les différents niveaux d'énergies initiales (équations (14-17)) :

$$E_{nml} = NE_0 (1 - m) \quad (2.14)$$

$$E_{adv} = NmE_0(1 - m_0)(1 + a) \quad (2.15)$$

$$E_{sup} = Nmm_0E_0(1 + b) \quad (2.16)$$

$$E_{total} = NE_0(1 + m(a + m_0b)) \quad (2.17)$$

Où a,b sont des facteurs multiplicatifs de l'énergie initiale E<sub>0</sub> et a,b et b>a, m et m<sub>0</sub> des fractions des nœuds super et avancés du réseau respectivement

La probabilité P<sub>i</sub> est calculée selon l'équation (2.18) :

$$P_i = \begin{cases} \frac{E_i(r)P_{opt}}{(1 + m(a + m_0b))E_{avg}} & \text{if normal node} \\ \frac{E_i(r)P_{opt}(1 + a)}{(1 + m(a + m_0b))E_{avg}} & \text{if advanced node} \\ \frac{E_i(r)P_{opt}(1 + b)}{(1 + m(a + m_0b))E_{avg}} & \text{if super node} \end{cases} \quad (18)$$

E<sub>avg</sub> est calculé selon l'équation (2.8).

Le seuil T(S<sub>i</sub>) pour que S<sub>i</sub> soit éligible à devenir CH est le même que dans l'équation (2.11).

### 2.3.3.3- Simulation

Pour la simulation du système, les valeurs des paramètres radio mentionnés dans le tableau 1 sont utilisées dans les protocoles DEEC et EDEEC. Les nœuds avancés ont a = 1,5 fois plus d'énergie et les super-nœuds ont b = 3 fois plus d'énergie que les nœuds normaux. Dans cet article, nous nous intéressons essentiellement à la durée de vie du réseau et à la transmission des paquets. Les figures 3 et 4 montrent clairement que le protocole EDEEC est

significativement plus efficace que DEEC en termes de durée de vie du réseau et de paquets transmis à la BS.

Les propriétés suivantes sont utilisées pour simuler la région du capteur :

- Les nœuds capteurs et la station de base sont supposés stationnaires une fois qu'ils sont déployés dans l'environnement et que leur emplacement est fixé.
- Le réseau de capteurs sans fil comprend des nœuds de capteurs hétérogènes.
- La station de base n'est pas limitée en termes d'énergie, de mémoire et de puissance de calcul.
- Les nœuds sont éligibles pour déterminer leur niveau d'énergie actuel et les informations de localisation via le service GPS.
- Tous les nœuds capteurs sont immobiles et ont un ID de nœud fixe.
- L'agrégation des données est effectuée uniquement aux niveaux des nœuds CH.

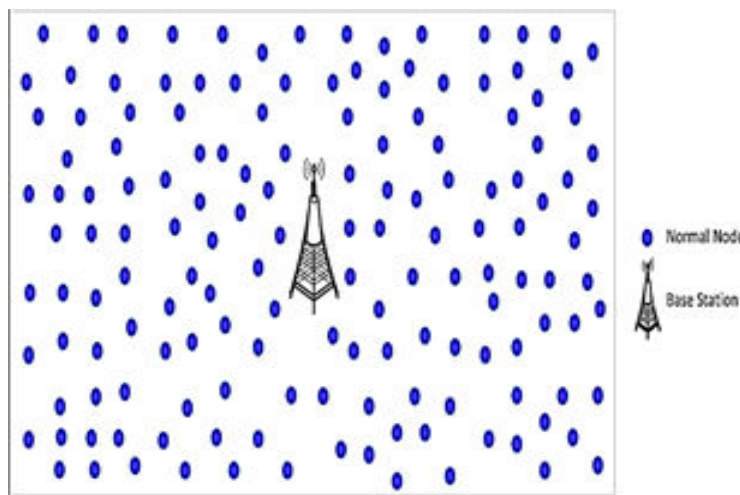


Figure 2.5- Exemple de déploiement dans un réseau de capteurs sans fil.

Tableau 2.1. Paramètres de simulation.

Parameters	Values
Network size	(100,100) m <sup>2</sup>
Number of nodes	100
Normal nodes initial energy ( $E_0$ )	0.5 J
A	1.5
B	3
Eda (data aggregation)	5nj/bit
$E_{elec}$	50nj/bit
$E_{fs}$	10nj/bit/m <sup>2</sup>
$E_{amp}$	0.0013 pj/bit/m <sup>4</sup>
$d_0$	76m
$P_{opt}$	0.1
Sink position	(50,50) in the centre of the network
Fraction of super nodes (m)	0.3
Fraction of advanced nodes ( $m_0$ )	0.3

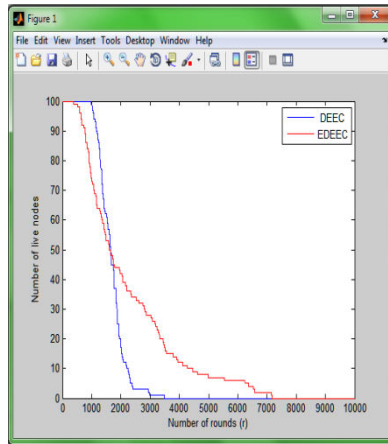


Figure 2.6- Nombre de noeuds vivants.

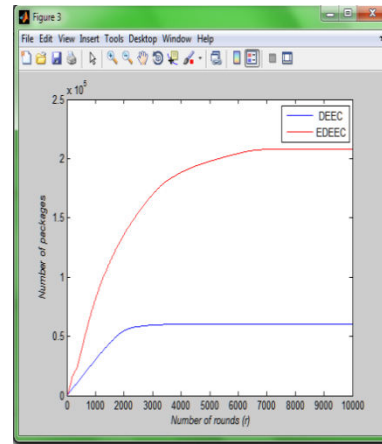


Figure 2.7- Nombre de paquets transmis

Les résultats de la simulation montrent que le protocole EDEEC est plus efficace que DEEC en termes de durée de vie du réseau et du nombre de paquets transmis à la BS. Le protocole EDEEC augmente la durée de vie du réseau. Dans cette simulation, la durée de vie du réseau et le nombre de paquets délivrés à la Station de Base sont multipliés par 2.

### 2.4 Cas d'application : Parking intelligent

Aujourd'hui, trouver une place pour stationner son véhicule dans les grandes villes est devenu un épineux problème. De nombreuses études ont prouvé que, lorsque les gens cherchent une place de parking, ils perdent du temps, consomment de l'énergie et participent à l'augmentation de la congestion du trafic (Arnott et al., 2005 ; Koster et al, 2013) sans compter les retombées écologiques (pollution) et économiques néfastes. Lorsque la recherche d'une place de parking se fait de façon naturelle, le conducteur cherche aléatoirement une place libre dans un parking proche de sa destination. Lors de cette recherche, le conducteur peut ne pas trouver l'itinéraire optimal pour atteindre le parking ou alors le parking désiré est complet à son arrivée. Le stationnement intelligent est une solution intéressante à ce problème et de nombreuses recherches ont été effectuées dans ce domaine dans le but d'améliorer les approches et les applications servant à aider les conducteurs. Les technologies émergentes et, principalement, les réseaux de communication sans fil ainsi que l'Internet des objets (IoT) et le cloud computing (Gupta et al., 2013) sont à la base de ces approches.

Pour modéliser une situation de stationnement intelligent, l'approche multi-agents (MAS) est considérée comme une solution efficace. Dans un système multi-agents, un ensemble d'agents pouvant être réactifs, cognitifs ou hybrides collaborent et coopèrent afin de résoudre un problème commun de manière parallèle et distribuée. Chaque agent n'a qu'une connaissance partielle de l'environnement global et peut exercer un ensemble de rôles (fonctions, plans, services). L'émergence des comportements des différents

agents tend à résoudre le problème global posé. Dans la suite de cette section, seuls les travaux récents concernant les applications SMA dans les parkings intelligents (2017-2021), sont brièvement présentés.

Dans (Belkhala et al., 2019), les auteurs proposent une architecture de stationnement intelligent basée sur les systèmes multi-agents. Un ensemble d'agents dédiés sont proposés à trois niveaux différents. Au niveau de l'interface se trouvent des objets connectés tels que des capteurs (caméras), des terminaux RFID et autres agents dédiés à la capture de l'environnement dans lequel évolue le système. Au niveau du contrôle, d'autres agents assurent le bon fonctionnement de cette architecture. Une base de connaissances est utilisée pour la sauvegarde et la consultation des données. Dans (Hassoune et al., 2020), les auteurs utilisent une approche basée sur l'algorithme Ant Colony Optimization (ACO) pour trouver le chemin optimal entre les conducteurs et le parking libre le plus proche. (Jioudi et al., 2019) propose un système basé sur une stratégie d'intelligence distribuée en essaim utilisant l'algorithme de colonie de fourmis, le système cloud et les systèmes multi-agents pour guider les conducteurs vers le parking libre le plus proche. Dans (Okoso et al., 2019), les auteurs proposent un système dit Cooperative Automated Valet Parking (Co-AVP) pour rechercher automatiquement les places de stationnement vacantes pour les véhicules. La recherche coopérative de chemin a été largement étudiée et considérée comme la recherche de chemin multi-agents (MAPF). Les auteurs dans (Sakurada et al., 2019) présentent dans leur article un système de stationnement intelligent Cyber-physical System (CPS) basé sur des agents. Ce système est composé d'agents logiciels interconnectés utilisant les technologies de l'Internet des objets. L'approche proposée a été mise en œuvre pour les vélos et pour les voitures. Dans (Zulficar et al., 2021), il est question d'un état de l'art très poussé de travaux récents concernant la gestion de parkings intelligents et de réservation de stationnement mobile. Le système présenté dans (Jioudi et al., 2021) prend en compte la position de la voiture et sa vitesse, et propose des places de stationnement disponibles en fonction du lieu de destination et de l'heure d'arrivée. Le système prend également en compte les frais de stationnement.

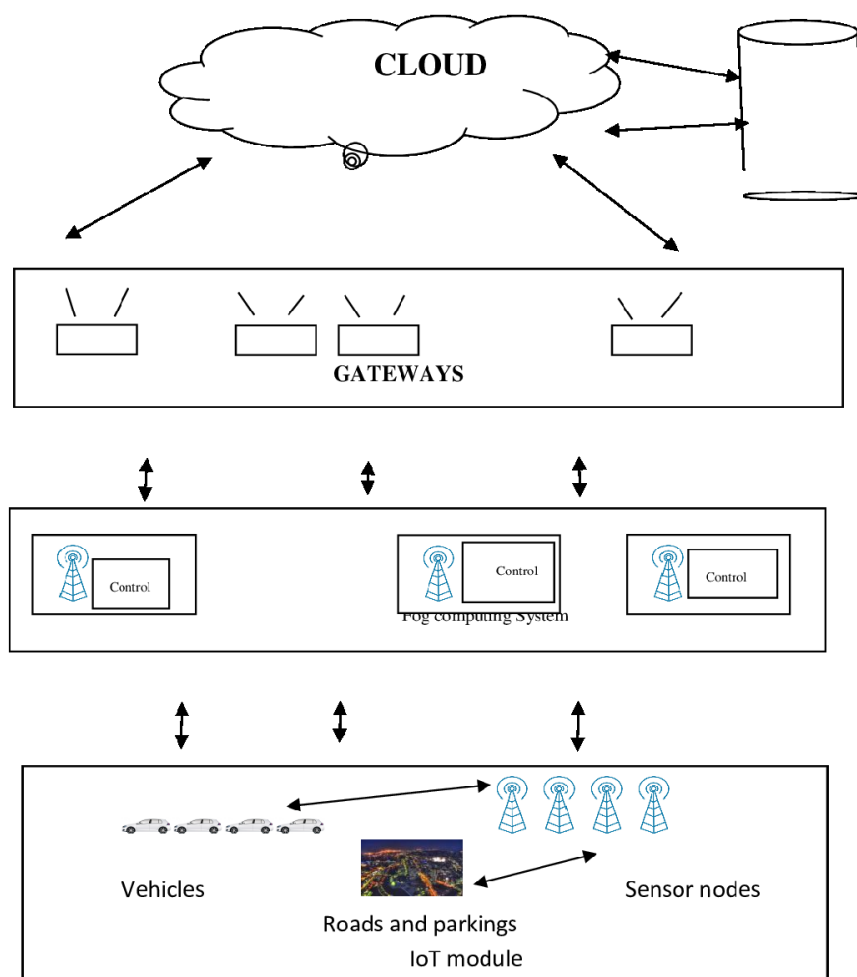


Figure 2.8- Architecture de communication dans un parking intelligent.

La figure 2.8 présente un schéma des niveaux de communication dans un système de stationnement intelligent :

- Au bas de cette architecture se trouve le module IoT, qui est chargé d'établir une communication sans fil entre les objets du système. Ces objets concernent les parkings, les plans de circulations routières et les véhicules (ou les chauffeurs). Le module IoT établit la communication avec les modules de Fog Computing. Les informations recueillies et véhiculées utilisent des RCsF.
- A un deuxième niveau se trouve le fog computing, qui vise à décentraliser le flux de calcul. Les informations issues du niveau IoT sont transmises à ce niveau. Il existe plusieurs modules de calcul. En réalité, chaque module comprend trois éléments : Un système de contrôle, une interface de balisage et une interface de communication.
- Les données sont acheminées via un ensemble de passerelles vers le module cloud. Ces derniers offrent de grandes et puissantes capacités de traitement et de stockage.

Dans l'approche proposée, le système intelligent aide les automobilistes à trouver une place de parking. Quatre agents sont proposés pour gérer le système de stationnement et chaque agent a des tâches de règles spécifiques à réaliser comme suit :

- L'Agent Chauffeur (DA) : recherche une place de parking en sollicitant un gestionnaire puis attend la réponse du gestionnaire (succès ou échec).
- L'Agent Gestionnaire (AG) : reçoit les demandes de DA avec leurs préférences (prix et emplacement) et travaille pour trouver l'espace de stationnement approprié selon les préférences des agents en envoyant la demande aux Agents du Secteur.
- L'Agent de Secteur (SA) : travaille pour trouver l'espace d'agent de stationnement approprié en fonction des prix et des préférences du DA.
- L'Agent de Place de Parking (AP) : observe et valide l'état de l'espace de stationnement et envoie un résultat par message à l'Agent Gestionnaire. Fig.2. présente un modèle de parking modélisée par le formalisme MAS.

Où:

- DA, MA, SA1, SA2, SA3, PA1, PA2, PA3 sont l'ensemble des agents qui participent à la gestion du stationnement,
- P1 : Driver Agent est prêt à demander ;
- P2 : l'agent chauffeur attend pour interagir avec l'agent gestionnaire,
- P3 : Driver Agent attend la réponse du Manager,
- P4 : Manager Agent est prêt à interagir,
- P5 : l'agent gestionnaire prépare la demande à l'agent de secteur,
- P6 : l'Agent Manager attend la réponse de l'Agent Secteur,
- P7 : les agents du secteur sont prêts à interagir,
- P8 : les agents du secteur préparent la demande à l'agent de stationnement,
- P9 : les Agents du secteur attendent la réponse de l'Agent de stationnement,
- P10 : les Agents de Place de Parking sont prêts à interagir ;
- P11 : L'agent de stationnement traite la demande de l'agent de secteur (message de réussite ou message d'échec ou choix d'un autre agent de stationnement),
- P12 : L'agent de stationnement attend d'interagir avec l'agent de gestion,
- P13 : Manager Agent prépare la réponse de Driver Agent,

- P14 : l'agent Manager attend pour interagir avec l'agent,
- P15 : L'agent chauffeur obtient une réponse à sa demande,
- T1 : l'agent chauffeur envoie une demande à l'agent gestionnaire avec ses préférences (lieu et prix),
- T2 : l'agent gestionnaire reçoit la demande de l'agent conducteur,
- T3 : l'Agent Manager envoie des requêtes à l'Agent Secteur,
- T4 : l'Agent Secteur reçoit la demande de l'Agent Gestionnaire,
- T5 : l'agent de secteur envoie une demande à l'agent de stationnement,
- T6 : L'Agent Place de Parking reçoit la demande de l'Agent Secteur,
- T7 : l'agent de stationnement choisit un autre secteur de stationnement,
- T8 : l'agent de stationnement envoie un message de réussite à l'agent de gestion,
- T9 : Parking Space Agent envoie un message d'échec à Manager Agent,
- T10 : Manager Agent reçoit le message Parking Space Agent,
- T11 : l'agent de gestion envoie un message à l'agent de conduite,
- T12 : l'agent de conduite reçoit le message de l'agent de gestion,

### 2.4.1- Approche proposée

Pour la conception de ce système, des réseaux de référence sont utilisés. Les réseaux de référence sont des réseaux de Petri de haut niveau orientés objet, dans lesquels les jetons peuvent être eux-mêmes des réseaux. Il s'agit de Réseaux dans les Réseaux qui assument une sémantique référentielle. Dans cette approche, les jetons à un endroit peuvent contenir un réseau d'objets. De plus, les réseaux d'objets eux-mêmes peuvent à nouveau contenir des jetons qui représentent des réseaux, et ainsi un système de réseaux imbriqués est obtenu. De plus, les transitions dans les réseaux peuvent activer des transitions dans d'autres réseaux, tout comme les appels de méthode d'objets, en utilisant des canaux synchrones [9]. Reference Nets Workshop (Renew : <http://www.renew.de>) est un simulateur de réseau de Petri de haut niveau basé sur Java qui fournit une approche de modélisation flexible basée sur des réseaux de référence.

Mulan (réseau de système multi-agents dans le réseau) [10] est implémenté dans Renew. Mulan a la forme universelle illustrée à la Fig. 2. Ce formalisme combine les outils MAS et Petri Nets. Les agents sont capables d'être intelligents, ils ont donc accès à une base de connaissances ; le comportement de l'agent s'exprime sous forme de protocoles. Les modèles de protocole peuvent être instanciés. Le protocole fait partie d'une conversation. Dans cette étude, la base de connaissances est implémentée dans Java Expert System Shell (JESS). La base de connaissances de l'agent est remplie de faits et de règles.

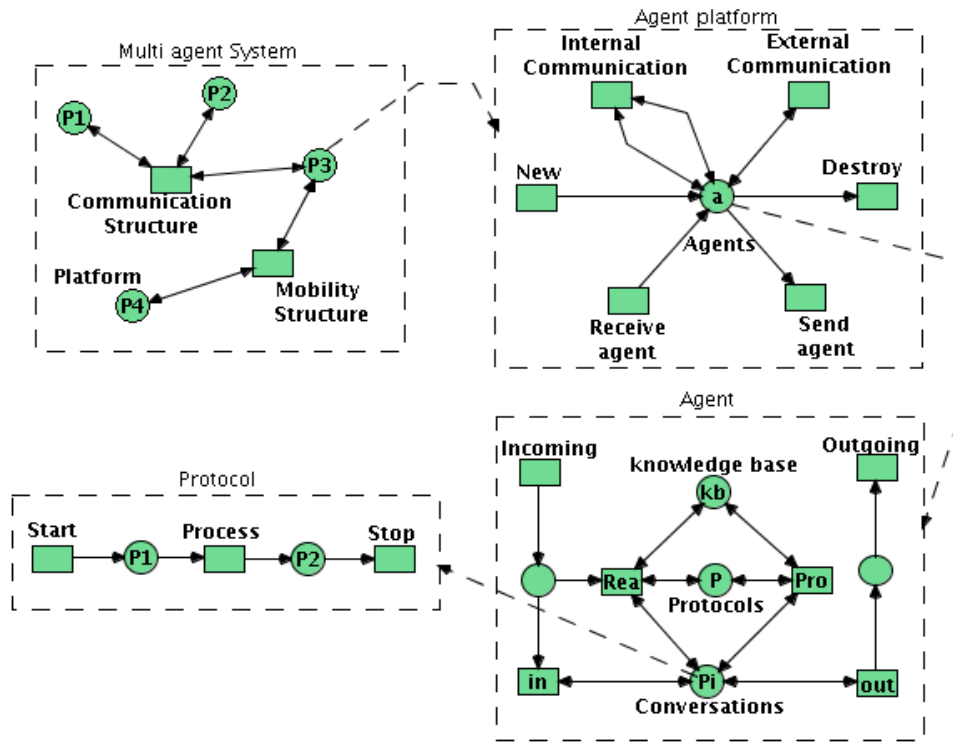
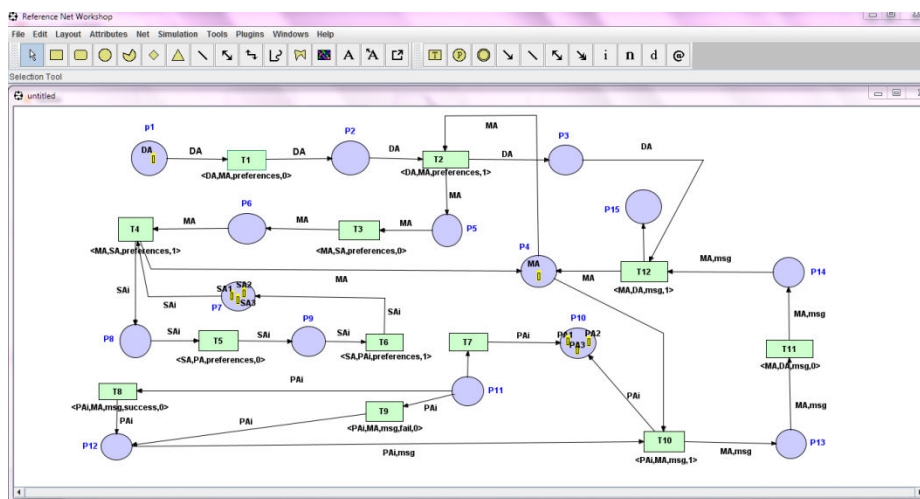


Figure 2.9- Les réseaux dans les réseaux (Nets within Nets).

### 2.4.2 Le modèle Renew obtenu

La figure 3 illustre le modèle de simulation Renew du parking intelligent. Il convient toutefois de noter que cela fait partie du système. Les différents lieux et transitions sont conformes aux propositions énoncées au début de cette section.



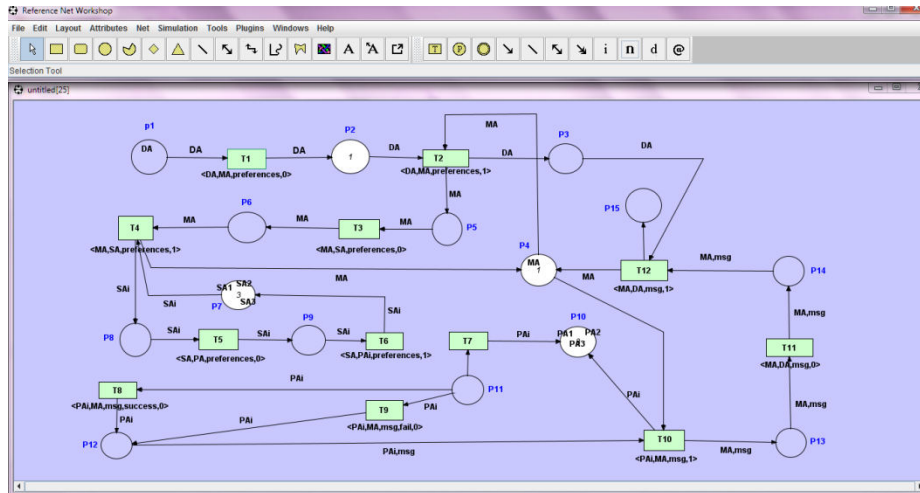


Figure 2.10 Modélisation du système avec Renew.

### 2.4.3 La mise en œuvre utilisant la plateforme JESS du modèle

Le système expert JESS est utilisé pour modéliser la base de connaissances des agents de stationnement intelligents. Quelques exemples de mise en œuvre sont donnés dans ce qui suit. Par exemple pour établir la communication au sein des agents, nous définissons le modèle de message ACL comme suit :

```
(deftemplateACL Message (slot communicative-act) (slot sender) (slot receiver) (slot conversation-id) (slot protocol) (slot language) (slot content) )
```

Le contexte de la mémoire de travail d'un agent automobiliste dépend de la route et de la position à laquelle il se trouve. Les modèles suivants décrivent le contexte :

```
(deftemplate car (slot position) (slot action) (slot speed) (slot weather) (slot fuel) (slot engine) (slot state_road) (slot content_road) (slot offer_platform))
```

Nous illustrons maintenant quelques règles JESS, qui montrent les changements de contexte. Dans le premier, dès que la voiture veut se déplacer vers le parking ou le centre-ville, elle capture le contexte de ces derniers.

```
(defrule A
  (car {(position == zone)// ( position == urban area)} && (content_road == position))
  =>
```

```
(assert (ACLMessage (communicative-act INFORM) (sender platform)
(receiver car) (content "position") (conversation-id "information") ))
(assert (car(deplacement )))
```

### **2.4.4 Discussion**

Les progrès très rapides dans le domaine des technologies de la communication et de l'information ont permis de concevoir des machines dotées de remarquables moyens d'analyse et de décision et pouvant apporter des bénéfices très appréciables dans tous les domaines. Dans ce chapitre, nous avons présenté une tentative de modélisation d'un parking intelligent qui peut aider les conducteurs à trouver une place dans un parking proche de leur destination. L'information est extraite et véhiculée à l'aide des nœuds capteurs sans fil. L'organisation en réseaux ADHOC de ces nœuds grâce à des protocoles de routage permet au système de dialogues, d'acheminer l'information et de gérer en temps réel un système complexe en distribuant les traitements et les accès aux données.

### **2.5 Conclusion**

Un réseau de capteurs sans fil comprend un ensemble, parfois impressionnant, de nœuds capteurs disséminés de façon aléatoire ou prédéterminée dans une zone géographique dite zone d'intérêt. Les communications entre les nœuds capteurs sont la source principale de la dissipation et de la consommation énergétique. Il y'a lieu, ainsi, d'effectuer un contrôle global de ces communications surtout lorsqu'il s'agit de la transmission de données qui est très couteuse en énergie. Plusieurs protocoles de routage ont été mis en œuvre pour gérer efficacement ces échanges d'informations. Dans ce chapitre, un certain nombre de protocoles a été passé en revue. Le chapitre suivant est consacré à la modélisation et la simulation des protocoles pour les réseaux de capteurs sans fil.

## *Chapitre 3 Les simulateurs des protocoles de routage dans les réseaux de capteurs sans fil.*

---

- 3.1- Introduction*
  - 3.2- La modélisation/simulation dans les RCsFs*
    - 3.2.1- Aspects liés à la simulation*
    - 3.2.2- Intérêts de la Modélisation/Simulation*
    - 3.2.3- Le processus de Modélisation/Simulation*
    - 3.2.4- Outils pour la simulation*
    - 3.2.5- Les paramètres de simulation*
  - 3.3- La modélisation/simulation basée sur les systèmes multi-agents*
    - 3.3.1- Les systèmes multi-agents*
    - 3.3.2- L'approche proposée*
    - 3.3.3- La plateforme multi-agent MADKIT*
    - 3.3.4- Application à la simulation des protocoles de routage dans les RCsFs : Cas des protocoles hiérarchiques.*
    - 3.3.5- Mise en œuvre de l'approche*
    - 3.3.6- La structure organisationnelle du simulateur*
  - 3.4- Cas d'application*
    - 3.4.1- Quelques fonctionnalités du simulateur*
  - 3.5- Conclusion*
- 

### **3.1- Introduction**

De nos jours, nous assistons à une prolifération de nouvelles technologies, qui induisent un changement radical dans nos habitudes, nos comportements et notre façon de vivre et de penser. Les objets familiers deviennent de plus en plus intelligents et acquièrent de plus en plus d'autonomie. On peut considérer que cette intelligence des objets est très liée à leurs interactions. Les communications sans fil permettent l'échange d'informations entre objets distants. Les réseaux de capteurs sans fil offrent des outils de partage d'informations entre ces objets. Cependant, plusieurs aspects technologiques, organisationnels et énergétiques doivent être considérés au sein de ces réseaux. Ainsi, et au cours des dernières décennies, plusieurs chercheurs se sont penchés sur la mise en place de protocoles permettant l'organisation et la gestion de ces réseaux sans fil. L'un des points qui a le plus retenu l'attention de ces chercheurs concerne l'économie d'énergie et son optimisation de

## *Chapitre 3 Les simulateurs des protocoles de routage dans les réseaux de capteurs sans fil.*

---

manière à assurer une disponibilité maximale du réseau et son maintien en vie le plus longtemps possible. En effet, ces réseaux disposent de ressources énergétiques réduites provenant de batteries qui ne peuvent pas être, le plus souvent, rechargées lorsque les nœuds capteurs sont hébergés dans des endroits hostiles ou inaccessibles. Les ressources matérielles et logicielles sont également réduites compte tenu de la taille des nœuds capteurs et de leurs réserves en énergie. Un autre aspect à considérer concerne les distances pour la transmission et la réception des données entre les nœuds. D'autres aspects liés aux réseaux de capteurs concernent la sécurité des échanges d'informations entre nœuds capteurs et la quantité d'informations à transmettre qui ne doit pas saturer le réseau. Dans le but de proposer de nouveaux protocoles de routage de données entre nœuds capteurs, prenant en compte tout ou partie des aspects précités, il est nécessaire de simuler leur comportement et de comparer leurs résultats avec ceux obtenus à l'aide d'autres protocoles existants. Pour cela, il existe plusieurs plateformes de simulation dont certaines sont open source et d'autres propriétaires. Ce chapitre passe en revue plusieurs de ces outils parmi les plus utilisés dans la littérature puis propose les principes de base d'un nouvel outil de simulation de protocoles de routage dans un réseau de capteurs sans fil basé sur des systèmes multi-agents. Le travail présenté ici se concentre sur la simulation des protocoles hiérarchiques à l'aide des concepts de rôle de groupe et d'agents. Dans ce cadre, le modèle organisationnel AALAADIN est utilisé pour développer le modèle de système semi-formel et le simulateur est implémenté en utilisant la plateforme multi-agent MADKIT.

### **3.2- La modélisation/simulation dans les RCsFs**

L'étape de modélisation d'un système est une action prioritaire qui consiste dans la construction d'un prototype concret, abstrait ou virtuel représentant fidèlement le système ou quelques-unes de ses parties. Ce prototype sera utilisé dans le cadre de la conception, de l'amélioration des performances ou dans un cadre purement pédagogique à des fins d'entraînement, d'assistance ou d'enseignement/présentation. Ainsi, au-delà du fait de retombées économiques en gain de coûts, on peut souligner l'apport tant en sécurité qu'en clarté d'une telle entreprise. La simulation consiste, elle, dans la soumission du modèle obtenu à un ensemble de cas de tests dont le but est d'analyser les comportements et réactions de ce modèle lorsqu'il est soumis à ces données de tests pour en tirer des conclusions. Il s'en dégage, ainsi, plusieurs théories concernant la modélisation simulation (Zeigler et al. , 2019 ; Bossel, 2018 ; Padilla et al ., 2018, Schweiger et al., 2020).

### *Chapitre 3 Les simulateurs des protocoles de routage dans les réseaux de capteurs sans fil.*

---

Pour analyser les performances des réseaux sans fil ainsi que des réseaux câblés, trois techniques principales peuvent être utilisées : les méthodes analytiques (mathématiques), les mesures au banc d'essai ou mesures physiques et la simulation par ordinateur et (Mehta et al., 2010).

Dans les méthodes mathématiques (Deng and al., 2015) , il s'agit de représenter le réseau par des ensembles d'équations et de fonctions mathématiques en fixant un ensemble de conditions initiales ainsi qu'un ensemble de paramètres. Il s'agit, cependant, de méthodes qui sont parfois très complexes et qui nécessitent des équations comportant, le plus souvent, de nombreux facteurs et dont la complexité peut devenir très vite non maîtrisable. La résolution de ces équations pour des jeux de données réels ou abstraits permet de prédire ou de représenter l'évolution du système. Il s'agit, très souvent, d'équations différentielles, dont les paramètres sont souvent simplifiés et optimisés pour maîtriser la complexité de résolution. De plus, il existe un fossé sémantique qui peut être, souvent, non négligeable entre le phénomène à modéliser et le système d'équation censé le représenter. L'utilisation et la manipulation d'un tel système d'équation sont, ainsi, une affaire de spécialiste. Dans cet ordre d'idée ; on peut citer la forme la plus basique de modélisation/simulation qui consiste à se faire une 'idée mentale' du système concerné et d'en faire des projections qui constituent, en quelque sorte, des plans mentaux d'expérimentation. C'est une forme mentale et intuitive qui constitue, certainement, la forme la plus utilisée et celle qui présente un coût nul. En extrapolant un peu sur ce domaine ; tout système provient de quelques idées de base qui en constituent l'essence même. Pour ne pas dire que toute modélisation/simulation a pour origine une abstraction mentale du phénomène à simuler.

La construction pratique de bancs d'essais ou de prototypes (maquettes) pour représenter un phénomène dans le but de le modéliser (Abuarqoub and al., 2016) compte parmi les méthodes assez simplistes mais les plus chères à mettre en œuvre. Le but est de construire une représentation pratique du phénomène puis d'agir sur cette maquette pour en tirer des conclusions (exemple : construction d'un prototype de réseaux de capteurs sans fil, et le faire évoluer en changeant un ensemble de conditions telles que simulation d'envois de paquets, vérification de la robustesse du réseau, changement de protocoles de routage, ...). Par conséquent, cette deuxième alternative peut être vue comme constituant un modèle complet pour les réseaux ad hoc sans fil et présente des intérêts non négligeables du point de vue clarté de

## Chapitre 3 *Les simulateurs des protocoles de routage dans les réseaux de capteurs sans fil.*

---

représentation. Cependant, d'une part la construction d'un banc d'essai pour tout réseau prédéfini est une tâche très coûteuse, souvent insupportable et nécessitant beaucoup d'efforts, de plus il y'a un problème de passage à l'échelle qui se pose et on pourrait difficilement prévoir ce qui adviendrait au système dans ce genre de situation.

De nos jours, la simulation par ordinateur est la méthode qui s'impose de plus en plus en s'avérant être un outil précieux pour développer et tester de nouveaux protocoles pour les réseaux de capteurs sans fil (Toor and Jain, 2017 ; Minakov et al., 2016). Cette méthode est nettement moins onéreuse que les bancs d'essai et plus souple et moins complexe que la simulation analytique. Parmi ses intérêts majeurs, il y'a le fait qu'il est possible d'effectuer la simulation en considérant une très large gamme de situations et de jeux d'essais. Cette technique peut être appliquée là où les méthodes analytiques ne sont ni applicables ni faisables. Les chercheurs utilisent couramment la méthode de simulation informatique pour analyser les performances d'un système. Ces simulateurs sont utilisés pour de nombreuses raisons telles que la réduction des coûts, la facilité de mise en œuvre et l'aspect pratique des tests de réseaux à grande échelle.

L'objectif principal des simulateurs est d'atteindre une situation "*aussi réelle que possible*" afin de rendre le résultat obtenu réaliste et adaptable. Dans la simulation de réseaux de capteurs sans fil, trois points importants doivent être pris en compte ; premièrement, les protocoles et les algorithmes doivent être exempts d'erreurs et avoir été mis en œuvre dans les détails, deuxièmement, l'environnement de simulation doit être réaliste et en plus de cela, une véritable méthode est utilisée pour analyser les données collectées. Cependant, il existe encore plusieurs pièges potentiels. Pour les surmonter, il est nécessaire de connaître les différents outils disponibles et de connaître leurs avantages et leurs inconvénients.

### **3.2.1- Aspects liés à la simulation**

Plusieurs aspects sont liés à la notion de simulation, nous présentons ci-dessous quelques-uns loin d'être exhaustif:

#### A- La simulation stochastique :

La simulation stochastique est l'un des outils de simulation les plus réalistes qui inclue un certain caractère aléatoire ainsi que des éléments qui s'écoulent dans le temps (Hui et al., 2022 ; Pereyra et al., 2015). Les principales applications de telles

## *Chapitre 3 Les simulateurs des protocoles de routage dans les réseaux de capteurs sans fil.*

---

simulations ont pour but de proposer un modèle d'observation du trafic dans un réseau particulier, des centres de service client, et bien d'autres.

### B- La simulation déterministe

Dans le cadre de la simulation déterministe des valeurs fixes et non aléatoires sont utilisées pour définir le modèle du système étudié (Ketshabetswe and al., 2019). La sortie du système est fixée en fonction de certaines entrées spécifiques en raison de l'absence de caractère aléatoire.

### C- La simulation dynamique

Les modèles de simulation dynamique sont des modèles qui entraînent des changements dans le système lorsque les signaux d'entrée changent. Le processus de simulation est évolutif et change dans le temps (Harish and Kumar, 2016).

### D- La simulation à événements discrets

Les modèles de simulation à événements discrets sont des modèles qui organisent les événements sur une base temporelle (Wainer, 2017 ; Silva, 2018 ; Wainer and Mosterman, 2018 ; Mesa, 2021). Dans ce type de simulations, une file d'attente d'événements est créée par le simulateur sur la base du temps dans lequel ils se produisent. Ensuite, le simulateur lit la file d'attente d'événements et un nouvel événement est entré au fur et à mesure que le précédent est exécuté. La plupart des outils de simulation relèvent de cette catégorie, comme les ordinateurs, les simulateurs d'arbre de défaillances et de tests logiques. Les simulateurs basés sur des agents sont un cas particulier de ce simulateur dans lequel les entités mobiles sont appelées agents. Cependant, dans le cas d'un modèle à événements discrets traditionnel, les entités n'ont que des attributs mais les agents sont dotés d'attributs et de rôles ou services analogues aux méthodes qui incluent des règles d'interaction avec d'autres agents. Les modèles de simulation à événements discrets sont de loin ceux qui ont donné lieu à la conception et à l'implémentation d'approches et de plateformes de modélisation simulation, notamment dans le cas de simulateurs pour les réseaux de capteurs sans fil (NS2, NS3,...) ainsi que pour les d'autres types de simulateurs pour l'IoT (Sehili, 2018), les CPs et autres technologies émergentes.

### E- La simulation continue

Les simulateurs continus sont de nature opposée aux simulateurs discrets. Dans ce cas de figure, la simulation prend en considération une base de temps continue où

## Chapitre 3 Les simulateurs des protocoles de routage dans les réseaux de capteurs sans fil.

---

l'évolution du phénomène simulé est considérée de façon ininterrompue (Bossel, 2018). Il s'agit, le plus souvent, de résoudre des systèmes d'équations différentielles qui montrent l'évolution du système. Ce type de simulation fonctionne en continu plutôt que par étapes discrètes.

### F- La simulation hybride

Dans le cas de la simulation hybride, les outils utilisés combinent à la fois les caractéristiques des simulateurs continus avec celles des simulateurs discrets (Klee and Allen, 2018 ; Onggo and al., 2018).

### G- La simulation parallèle et distribuée

La modélisation parallèle et distribuée considère un ensemble de modèles répartis sur un ensemble de sites qui sont simulés de façon concurrente (Fujimoto, 2015 ; Rashid and al., 2018). Une utilisation optimale de ce genre de simulation utilise des réseaux interconnectés, l'exécution se faisant sur un ensemble de machines interconnectées.

### 3.2.2- Intérêts de la Modélisation/Simulation

La modélisation/simulation est très employée dans les systèmes et les procédés industriels et reste un enjeu majeur. De nombreux travaux ont été consacrés à l'étude approfondie de ce domaine en considérant l'organisation des systèmes, la matière et l'énergie, les comportements collectifs et leurs propriétés émergentes inhérentes, etc. Parmi les approches de modélisation/simulation utilisées, la modélisation & simulation basées agent (Agent Based Modeling and Simulation (ABMS)) propose un ensemble d'outils théoriques et pratiques pour décomposer le système global en des sous-systèmes qui le constituent. L'impact attendu de cette démarche est dit phénomène d'émergence où le comportement du système aussi complexe soit-il émerge des comportements locaux des différentes entités en interaction qui le composent (Ferber et al., 2003 ; Savaglio et al., 2020). Il est à noter, cependant, que comme toute démarche de modélisation ; il y'a lieu de considérer les outils '*pour modéliser le système*' de ceux '*qui modélisent*' le phénomène lui-même. Dans le cadre de l'ABMS, une idée générale est de considérer que '*tout est agent*' à contrario de l'approche prônant l'hybridation d'outils de conception et de modélisation. Dans l'industrie moderne, des technologies de pointe sont utilisées pour développer des procédés industriels. La grande majorité de ces systèmes peuvent être considérés comme '*compliqués*' ou '*complexes*' dans le sens où ils sont composés d'un grand

### *Chapitre 3 Les simulateurs des protocoles de routage dans les réseaux de capteurs sans fil.*

---

nombre d'éléments en interaction. A notre connaissance, les frontières entre ces deux notions sont assez floues et dépendent en grande partie des évolutions des outils et avancées technologiques. En effet, on peut donner une image naïve de ce phénomène à travers des idées, somme toutes, assez simplistes ; le système d'exploitation Windows 95 comporte 15 million de lignes de code et était, à l'époque, qualifié de complexe en requérant une configuration matérielle qui semble de nos jours dérisoire (420 Mo de RAM).

L'objet du travail présenté ici porte sur une étude concernant les principaux outils et plateformes dédiés à modélisation et la simulation des protocoles de routage dans les réseaux de capteurs sans fil puis à la proposition d'un ensemble d'idées pour une approche basée SMA en vue de la conception d'un simulateur dans ce domaine. Notre équipe a mené divers projets similaires en utilisant différentes approches (DEVS, MAS, etc.) dans le cadre de la conception/réalisation de simulateurs industriels, spécialement pour les complexes pétrochimiques (Seddari et al, 2014 ; Seddari et al., 2021). L'utilisation d'outils informatiques de modélisation offre l'avantage de construire un modèle informatique très fidèle au système considéré ou à un ensemble de ses parties, puis de juger du comportement du modèle obtenu en fonction des jeux de données fournis en entrée. Cela permet de manipuler, d'observer et d'améliorer la compréhension de son comportement et permet d'éviter de nombreuses erreurs possibles concernant ses évolutions futures. L'arrêt de certains grands systèmes dits critiques peut entraîner des dommages matériels et financiers. Pour ces systèmes, l'utilisation de modèles les représentant est d'une aide substantielle. En effet, au lieu d'arrêter le fonctionnement de tout ou partie d'une usine, il suffit de considérer les modèles qui les représentent. C'est le cas, par exemple, des chaudières industrielles. L'arrêt de tels équipements peut causer des dommages financiers à l'usine et leur redémarrage peut prendre beaucoup de temps créant, ainsi, beaucoup de retard dans les systèmes de production.

La modélisation des systèmes a fait l'objet de nombreuses études et recherches compte tenu de son impact important sur les phénomènes mis en œuvre (Fishwick, 1995 ; Fishwick, 1997) considère la modélisation comme un ensemble de métaphores et d'analogies permettant de mieux appréhender tout phénomène. [7], définit la simulation informatique comme suit : « La simulation informatique est la discipline de conception d'un modèle d'un système physique réel ou théorique, l'exécution du modèle sur un ordinateur numérique et l'analyse de la sortie d'exécution ».

### 3.2.3- Le processus de Modélisation/Simulation

Trois phases fondamentales s'envisagent dans ce cadre : La création du modèle, l'exécution du modèle obtenu et les résultats obtenus (figure 3.1). De plus, (Zeigler et al. , 2019) détaillent le processus de simulation selon les critères suivants :

- (1) Le système source abstrait ou réel d'intérêt.
- (2) Les conditions d'expérimentation du système.
- (3) La spécification du système qui définit le modèle : un programme informatique, un ensemble de règles mathématiques, d'équations ou d'autres contraintes pour la génération du comportement du système.
- (4) Le système de calcul dit simulateur qui est un système capable d'exécuter le modèle obtenu pour générer son comportement.
- (5) La relation de modélisation, qui traite de la relation entre un modèle, un système et un cadre expérimental. Il définit la validité du modèle, sa capacité à capturer fidèlement le comportement du système dans la mesure exigée par l'objectif de l'étude de simulation.
- (6) La relation de simulation, qui définit la relation entre un simulateur et un modèle. Il s'agit de la capacité du simulateur à simuler correctement le modèle, à générer fidèlement la trajectoire de sortie du modèle compte tenu de son état initial et de sa trajectoire d'entrée.

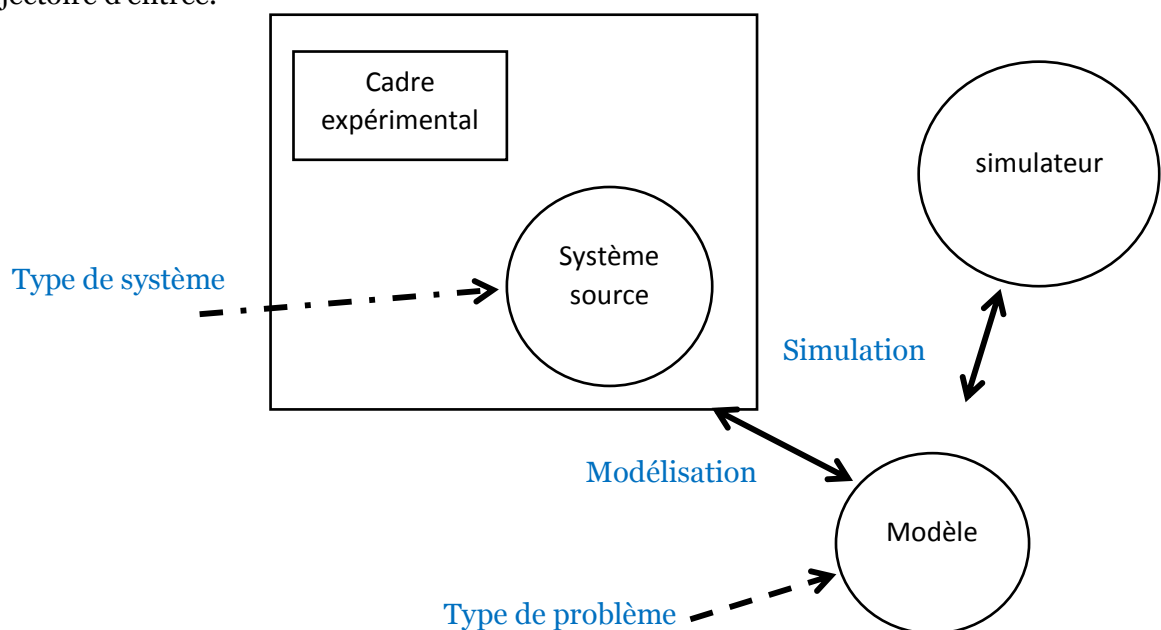


Figure 3.1 - Cadre conceptuel pour la modélisation/simulation

## *Chapitre 3 Les simulateurs des protocoles de routage dans les réseaux de capteurs sans fil.*

---

De ces définitions, on peut déduire que le modèle d'expérimentation est ce que l'on appelle sa simulation. Pratiquement, une simulation est la phase d'exécution du modèle. A partir d'un ensemble de données pertinentes en entrée du modèle, cette phase consiste à faire évoluer le modèle et à observer et analyser les résultats obtenus. Dans certains cas, la simulation de systèmes complexes peut prendre des heures voire des jours et consomme beaucoup de temps CPU. C'est pourquoi les développeurs ont recours à des méthodes de réduction de la complexité. Parmi ces méthodes, on peut citer la décomposition du phénomène en plusieurs entités de moindre complexité ou l'utilisation de CPU rapides et parallèles.

La recherche d'une plus grande fidélité, de précision et d'adaptation nécessite que les outils informatiques de modélisation et de simulation soient choisis avec le plus grand soin. Leur utilisation dans le domaine de la modélisation et de la simulation est très répandue assurant d'excellentes qualités dont l'amélioration est continue avec des coûts très raisonnables.

Plusieurs approches, méthodes et outils de modélisation ont été et continuent à être utilisés dans la littérature. Les réseaux de Petri (RdP ou Petri Nets : PN) (Juranić et al., 2017) comptent parmi les outils les plus efficaces et les plus largement utilisés. Ces réseaux, malgré leur simplicité, permettent de représenter la dynamique des systèmes. Plusieurs variantes et extensions de ces réseaux ont été développées telles que les réseaux de Petri colorés (Colored PN (CPN)) et les réseaux de Pétri temporisés (Timed PN) (Jensen et al., 2007). Malgré leurs soixante-deux ans d'existence (PN a été introduit par Adam Petri en 1962 dans sa thèse de doctorat), ils continuent à être pertinents aujourd'hui.

Les diagrammes d'états UML (Unified Modeling Language) sont d'autres outils semi-formels puissants capables de représenter le comportement et les interactions entre les acteurs d'un système (Sunitha and Samuel, 2019). De plus UML est normalisé par l'OMG (Object Management Group). Les systèmes multi-agents (MAS) sont d'autres outils de modélisation utilisés, qui offrent l'avantage d'une décomposition fine d'un système complexe réel en un ensemble de ses composants. Le comportement du système global émerge de celui des comportements de ses composants. La notion d'agent désigne une entité dont le fonctionnement est autonome et qui est capable de réaliser un ensemble de tâches (Mess and Guerrits, 2018). Ces systèmes sont de plus en plus utilisés dans les systèmes informatiques en raison de leur force d'expression, leur souplesse, leur simplicité, leur efficacité, et leur possibilité d'extension. De plus,

## *Chapitre 3 Les simulateurs des protocoles de routage dans les réseaux de capteurs sans fil.*

---

la description donnée par un SMA peut être plus facilement comprise par un observateur humain et s'adapte complètement au système représenté.

Dans ce travail, nous avons adopté une méthode descendante pour la conception de simulateurs industriels et utilisé une approche de modélisation et de simulation à base d'agents (ABMS) pour les avantages susmentionnés. L'ensemble du système est décomposé en sous-systèmes pertinents. Un agent réactif représente chacun d'eux. La communication entre les agents est assurée par le passage de messages. Pour concevoir le modèle du simulateur, nous avons opté pour le modèle organisationnel AALAADIN. Nous avons utilisé la plateforme multi-agents MADKIT (Multi Agent Development KIT) pour la mise en œuvre du modèle obtenu. AALAADIN et MADKIT ont été développés au Laboratoire d'informatique, de robotique, de mécanique et de micro-électronique (LIRMM) de Montpellier par l'équipe du professeur J. Ferber (Ferber et al, 2003 ; Souidi et al., 2016).

### **3.2.4- Outils pour la simulation**

Il est évident qu'avant d'implémenter ou de déployer un nouveau protocole pour les réseaux de capteurs sans fil, il est nécessaire d'évaluer et de tester les algorithmes proposés. La simulation informatique offre un moyen simple, adéquat, rapide, flexible, adaptable, robuste et relativement peu coûteux pour l'évaluation de tels protocoles.

De nos jours, il existe de nombreux simulateurs de réseaux de capteurs sans fil. Ces simulateurs présentent plusieurs caractéristiques concernant leurs exigences particulières, leur conception et les outils proposés. Les critères pour les simulateurs des réseaux de capteurs sans fil incluent, entre autres caractéristiques, la modélisation de la consommation d'énergie et l'évolutivité.

Ces dernières années, nous avons assisté à une large gamme d'outils et de plateformes dédiées à la simulation de réseaux filaires et sans fil. Le développeur peut choisir des outils particuliers en fonction de son application. La disponibilité de l'outil en question, l'expérience du développeur et sa capacité à manier ces outils sont des éléments clés pour mener à bien et réussir une bonne activité de simulation.

De plus, certaines plateformes sont en open source et d'autres commerciales. A titre d'exemple, nous présentons dans ce qui suit quelques plateformes dédiées à la simulation de réseaux de communications. Au choix de sélection des utilisateurs, plusieurs articles de recherche ont proposé des critères comparatifs pour évaluer les

## *Chapitre 3 Les simulateurs des protocoles de routage dans les réseaux de capteurs sans fil.*

---

simulateurs de réseaux. On trouvera dans la suite de ce chapitre plusieurs études comparatives pour ce genre de simulateurs (Patel et al. ,2018 ; Sharma et al., 2019 ; Raja, 2018)..

Un logiciel Open Source (FOSS : Free Open Source Software) est un logiciel à licence libre dans le sens où un utilisateur peut librement accéder au code source de ce logiciel, en effectuer des modifications, l'utiliser gratuitement et le distribuer (Maracke, 2019). Un tel logiciel peut être utilisé dans des applications diverses. Il existe plusieurs simulateurs de réseau qui sont Open Source, parmi ces logiciels on distingue : NS2, NS3, OMNET++, JSIM, ... Quelques détails de simulateurs assez utilisés sont mentionnés ci-dessous :

### **- Le Simulateur NS2 (Network Simulator 2)**

Ce simulateur (Balfaqui, 2020) compte parmi les outils qui ont été les plus utilisés. C'est un simulateur à événements discrets qui peut prendre en charge la simulation de TCP, les protocoles de multidiffusion et le routage sur des réseaux sans fil et câblés. NS2 a été développé en 1995 sous Virtual Inter Network Test-bed (VINT) avec l'effort conjoint de l'Institut des sciences de l'information de l'Université de Californie du Sud, de l'Université de Californie à Berkeley, du Centre de recherche Xerox Palo Alto et du Lawrence Berkeley National Laboratory. La Defense Advanced Research Projects Agency et la National Science Foundation sont les principaux sponsors de NS2. Les différents modules de simulation de réseau sans fil mobile, c'est-à-dire les modèles de propagation radio, le protocole IEEE 802.11 MAC, les protocoles de routage ad hoc (c'est-à-dire AODV, DSR), l'IP mobile et les modèles de mobilité sont fournis par Monarch Project. NS2 est encore étendu par SensorSim pour prendre en charge la simulation pour les réseaux de capteurs.

Caractéristiques principales :

- NS2 peut être utilisé pour la simulation parallèle et distribuée (PDNS) en présentant de bons résultats (assez proches de la réalité) pour la conception de simulation.
- NS2 est disponible gratuitement et possède une grande bibliothèque de modèles.
- Son code source peut être facilement modifié par les chercheurs.

Limites :

## Chapitre 3 *Les simulateurs des protocoles de routage dans les réseaux de capteurs sans fil.*

---

- L'évolutivité de NS2 est très faible en termes d'utilisation de la mémoire et de temps de simulation dans les réseaux de capteurs sans fil (WSN), les réseaux peer to peer et les réseaux où des centaines de nœuds sont utilisés.
- L'énergie consommée par le matériel, les logiciels et les composants des nœuds WSN ne peut pas être facilement mesurée.

### - **Le simulateur NS3 (simulateur de réseau version 3)**

Le simulateur NS-3 (Campanile et al., 2020) est également un simulateur de réseau à événements discrets lancé en 2006 à des fins de recherche et d'enseignement. Il s'agit d'un logiciel gratuit sous licence GNU GPLv2. Les principales caractéristiques de NS3 par rapport à NS2 sont les suivantes :

- Noyau modulaire et documenté.
- Programmes C++ avec script Python
- Plus proche des systèmes réels
- Intégration logicielle
- Système d'attributs
- Modèles mis à jour

Caractéristiques principales :

- Il présente de bonnes caractéristiques d'évolutivité ; car il est intégré aux caractéristiques architecturales et porte le code de GTNetS.
- Il est également capable d'effectuer des simulations de réseau à grande échelle de manière efficace.
- Il fonctionne bien en termes d'utilisation de la mémoire.
- Cela résout le problème de la crédibilité.

Limites :

- NS3 dispose de moins de modèles de simulation que NS2 et il est moins utilisé que ce dernier.
- De plus, il n'a pas les capacités telles que l'utilisation de l'adressage IP, la conception et l'alignement avec les protocoles Internet, la gestion de plusieurs interfaces, les détails sur les modèles 802.11, etc.
- NS3 intègre le banc d'essai et la virtualisation

## Chapitre 3 *Les simulateurs des protocoles de routage dans les réseaux de capteurs sans fil.*

---

- **OMNET++** (banc d'essai de réseau modulaire objectif en C++)

OMNET++ (Varga, 2019, Viridis and Kirsche 2019) est un simulateur de réseau à événements discrets à usage général, extensible, modulaire, basé sur des composants et à architecture ouverte. Il est autorisé sous sa propre licence publique académique uniquement dans des contextes non commerciaux. Il est le plus couramment utilisé pour la simulation de réseaux informatiques, la mise en file d'attente de simulations de réseaux, etc. Une simulation OMNET ++ contient autant de modules simples qui forment le comportement atomique d'un modèle. Il fonctionne sur Linux, d'autres systèmes de type UNIX et sur Windows (XP, Win2K) et MAC-OS.

Caractéristiques principales :

- C'est un simulateur très bien structuré et il ne se limite pas aux simulations de protocole réseau comme NS2.
- L'une des principales caractéristiques réside dans le fait que les modules de OMNET ++ peuvent être réutilisables et qu'ils peuvent être combinés de plusieurs façons.
- Il est basé sur le langage C++ pour modéliser les réseaux de communication et autres systèmes distribués.
- Il prend en charge certains protocoles MAC et certains protocoles localisés dans les réseaux de capteurs sans fil.

Limites :

- Certains scénarios et simulations ne peuvent pas être implémentés et exécutés dans ces simulateurs car certaines fonctionnalités ne sont pas présentes.
- Des protocoles limités sont disponibles.

- **JSIM ( Java Simulator)**

JSIM (Bakare and Enoch, 2019) comme son nom l'indique, c'est un système de simulation basé sur Java qui est utilisé pour l'analyse et la construction de modèles numériques quantitatifs par rapport aux données de référence des expériences. Il est disponible gratuitement avec le code source. Fondamentalement, JSIM fournit un environnement de développement d'applications avec une architecture basée sur des composants. JSIM a été développé par une équipe du DRCL (Distributed Real-time Computing Laboratory) qui a été parrainé par le DARPA's Information Technology Office, l'Air Force Office of Scientific Research's Multidisciplinary University Research, la National Science Foundation (NSF), l'Ohio State University et l'Université de l'Illinois à Urbana Champaign.

## *Chapitre 3 Les simulateurs des protocoles de routage dans les réseaux de capteurs sans fil.*

---

Caractéristiques principales :

- Il montre une bonne évolutivité par rapport aux autres simulateurs.
- Il fournit une plate-forme indépendante, réutilisable et extensible.
- Des protocoles et des composants de réseaux câblés et sans fil peuvent y être implémentés.

Limites :

- Plus compliqué à utiliser que NS2.
- Moins efficace et une surcharge se produit dans les modèles de communication en raison de l'utilisation du langage Java.

### **- OPNET (Outils d'ingénierie de réseau optimisés)**

OPNET (Chen et al., 2019) est un simulateur de réseaux à événements discrets écrit en C++ et proposé en 1986 par le MIT. Il s'agit d'un logiciel de simulation disponible dans le commerce bien établi et principalement utilisé. Cependant, il peut être utilisé gratuitement par les chercheurs dans les universités et les programmes éducatifs. Il prend également en charge divers matériels réseau tels que les antennes et les émetteurs-récepteurs. La caractéristique la plus importante d'OPNET est sa capacité à surveiller et à exécuter divers scénarios simultanément. Il permet aux utilisateurs, à l'aide d'une interface graphique, de développer des modèles et des graphiques. OPNET Modeler fournit diverses fonctionnalités telles que la conception, l'étude de protocoles, de réseaux, d'appareils et d'applications. Ce modélisateur fonctionne sur les systèmes Windows et Linux.

Caractéristiques principales :

- Les paramètres des modèles peuvent être modifiés.
- Fournit un riche ensemble de modules pour chaque pile de protocoles comme IEEE802.11, IEEE 802.15.4 et divers protocoles de routage, à savoir AODV et DSR.

Limites :

- Le problème d'évolutivité qui est dû à la conception orientée objet.
- Le nombre de protocoles disponibles est restreint.

### **- QUALNET**

Le simulateur QUALNET (Dorathy and Chandrasekaran, 2018) est un simulateur de réseau ad hoc commercial qui a été développé par Scalable Network Technologies et

## Chapitre 3 Les simulateurs des protocoles de routage dans les réseaux de capteurs sans fil.

---

qui est basé sur le protocole GloMoSim (Prabha, 2020). Qualnet fournit un environnement à l'utilisateur pour concevoir de nouveaux modèles de protocole, améliorer les protocoles nouveaux ou existants, concevoir des réseaux câblés et sans fil à l'aide de modèles existants et analyser leurs performances. Le code source présent dans le modèle aide les développeurs à créer de nouvelles fonctions ainsi qu'à modifier celles existantes. Les principales caractéristiques de Qualnet sont la vitesse, l'évolutivité, l'extensibilité, etc. Il est écrit en Parsec (basé sur C) car il est construit sur GloMoSim.

Caractéristiques principales :

- Facile à utiliser et à apprendre.
- Facile à stimuler un réseau large et hétérogène.

Limites :

- Les codes conviviaux sont limités en raison de la disponibilité de nombreuses fonctions intégrées.

Table 3.1- Comparaisons de quelques simulateurs des RCsF

<b>Simulateur</b>	<b>Language</b>	<b>Type</b>
Ns-2	C++, TCL	Open Source
OPNET	C	Commercial
OMNeT++	C / C++ and Java	Open Source
QualNet	C/C++	Commercial
NetSim	C / C++ and Java	Commercial
Ns-3	C/C++	Open Source
J-Sim	Java	Open Source
NetLogo	Java,TCL	Open Source
Castalia	C/C++	Open Source
CupCarbon	Java	Open Source
SensorSim	C/C++	Open Source

De plus, la plupart des langages de programmation proposent des extensions et des frameworks pour la simulation de systèmes. Parmi ces environnements de programmation, MATLAB (MATLAB signifie Matrix Laboratory) est un langage commercial de programmation et de simulation numérique développé par Mathworks. Ce langage possède des fonctionnalités puissantes et variées qui sont

## Chapitre 3 *Les simulateurs des protocoles de routage dans les réseaux de capteurs sans fil.*

---

continuellement améliorées. L'environnement MATLAB/SIMULINK fournit plusieurs outils avancés pour divers domaines de simulation ainsi que des interfaces graphiques très développées. Plusieurs scénarios de simulation, notamment dans le cas des RCsF, ont été implémentés à l'aide de MATLAB / SIMULINK.

### **3.2.5- Les paramètres de simulation**

Il existe des caractéristiques spécifiques et d'autres plus générales à prendre en compte lors de la simulation de protocoles WSN. Parmi les facteurs les plus utilisés, on retrouve les suivants :

- La taille du réseau : Il s'agit de la zone couverte par le WSN.
- Le nombre de nœuds du réseau : Dans le cadre de simulation dynamique ce nombre peut évoluer durant la simulation.
- L'énergie initiale : Désigne la quantité d'énergie initiale du WSN. En général, il s'agit de l'énergie totale du réseau. Dans ce cas, il est judicieux de penser que l'énergie initiale d'un nœud est le résultat de la division de cette énergie initiale par le nombre de nœuds (cas théorique).
- La durée de la simulation : désigne le temps imparti à la simulation du réseau. Ce temps peut être réajusté en fonction de l'évolution de la simulation.
- La latence : la latence est le temps nécessaire pour qu'un paquet soit transmis du nœud source vers la station de base.
- La latence moyenne est la somme des latences de tous les paquets divisée par le nombre total de paquets reçus.
- Le taux de réussite, qui représente le rapport entre le nombre de paquets reçus par la station de base et le nombre total de paquets envoyés par tous les nœuds du réseau.
- L'énergie moyenne consommée par rapport au nombre de nœuds dans le réseau : l'objectif principal du protocole de routage est de minimiser la consommation d'énergie du nœud, car elle affecte directement la durée de vie du réseau.

C'est pourquoi il peut être pratique de comparer un protocole spécifique avec d'autres protocoles en termes d'énergie moyenne consommée par tous les nœuds du réseau.

- La durée de vie du réseau : la durée de vie du réseau peut être définie de trois manières :

## *Chapitre 3 Les simulateurs des protocoles de routage dans les réseaux de capteurs sans fil.*

---

♣ FND (First Node Died) : est aussi appelé période de stabilité, c'est l'intervalle de temps entre le début de la simulation et la mort du premier nœud.

♣ HND (Half Node Died) : c'est le temps entre le début de la simulation et le moment de la mort des demi-nœuds.

♣ LND (Last Node Died) : c'est le temps entre le début de la simulation et l'heure de mort du dernier nœud.

-Le nombre de nœuds vivants par tour : qui mesure le nombre de nœuds vivants à chaque tour.

- L'énergie consommée : mesure l'énergie consommée par les nœuds à travers la simulation. L'objectif principal des protocoles de routage est généralement focalisée autour du problème de la gestion énergétique qui est le problème clé dans la vie du réseau.

- La sécurité : La sécurité est un aspect important dans les RCsF. Les données peuvent être facilement interceptées et peuvent être modifiées ou déviées de leur destination initiale.

Afin de réaliser des comparaisons efficaces entre un protocole donné et d'autres protocoles existants, le simulateur doit disposer de données pré-enregistrées concernant autant de protocoles existants que possible. Cette base de données doit être évolutive.

### **3.3- La modélisation/simulation basée sur les systèmes multi-agents**

L'importance de la simulation multi-agents a fait l'objet de plusieurs travaux de recherche. Plusieurs domaines d'application ont été abordés ces dernières années et il existe un grand nombre d'articles scientifiques consacrés à ce sujet.

L'idée intuitive et simpliste d'une organisation multi-agents est de répartir les composants principaux d'un système global en sous-ensembles de moindre complexité qui peuvent avoir des opérations autonomes et qui peuvent interagir pour réaliser une tâche commune.

Une étude rapportée dans (Abar et al., 2017) donne une idée précise de l'étendue des domaines couverts par la modélisation et la simulation multi-agents ainsi que des

## *Chapitre 3 Les simulateurs des protocoles de routage dans les réseaux de capteurs sans fil.*

---

outils logiciels dédiés à chaque domaine. (Bandini et al., 2009) décrit les problèmes de modélisation et de simulation à base d'agents et certains domaines d'application.

(Chen et al., 2020) présente la modélisation et la simulation du comportement de l'utilisation stochastique des pompes à chaleur dans les communautés résidentielles, qui est un domaine mécanique. (Mahmood et al., 2020) rapporte une simulation et modélisation à base d'agents du phénomène naturel lié au prototypage rapide de centrales éoliennes. (Macal et al., 2020) présente un système multi-agents dédié à la modélisation des problèmes d'ingénierie et de gestion des systèmes. On trouve dans (Herrera et al., 2020) une étude des applications des systèmes multi-agents dans l'ingénierie des systèmes. (Zhou et al., 2007) présente une étude des packages de simulation à base d'agents concernant les marchés de la consommation d'énergie. L'article a étudié un cadre de simulation à médiation par agent visant à faciliter le développement de modèles pour les marchés de l'électricité.

Le domaine de l'éducation est également investi dans la modélisation multi-agents. Une modélisation et une simulation basées sur MAS d'un système d'enseignement intelligent sont présentées dans (Yang, 2020).

De nombreuses contributions contemporaines à la modélisation à base d'agents pour l'IoT et le cloud-fog computing ont été recensées (Savaglio et al., 2020 ; Fellir et al., 2020). (Nanna et al., 2020) présente un système multi-agents pour simuler la propagation d'une maladie contagieuse.

### **3.3.1- Les systèmes multi agents**

Les systèmes multi-agents sont un paradigme conceptuel émergeant pour simuler l'interaction de plusieurs agents autonomes dans un ensemble d'environnements (Chen and Ren, 2019 ; Wang et al., 2016 ; Herrera et al., 2020). Les systèmes multi-agents ont de nombreuses applications ; nous nous intéressons ici à leur utilisation pour construire un simulateur opérationnel d'un système. En général, un système est dit multi-agent lorsqu'il contient un ensemble d'agents en interaction, qui évoluent dans certains environnements. Un agent est influencé par la situation perçue de l'environnement et il y réagit. Les systèmes multi-agents sont dans le domaine de la recherche depuis au moins 40 ans maintenant. À notre connaissance, les fondations des systèmes multi-agents (SMA) remontent aux années 1980 lorsque ces systèmes ont été introduits lors de l'atelier sur l'intelligence artificielle distribuée (IAD), qui s'est tenu au Massachusetts Institute of Technology (MIT) en juin 1980. MAS a été

## Chapitre 3 *Les simulateurs des protocoles de routage dans les réseaux de capteurs sans fil.*

---

identifié comme une branche de l'intelligence Artificielle Distribuée (IAD) (Weiss, 1999 ; Konolige and Nilsson, 1980).

Il n'y a pas d'accord commun sur une définition du terme agent. De manière générale, on peut dire qu'un agent est une entité (matérielle ou logicielle) qui se distingue d'un programme ordinaire en ce sens qu'elle est dotée d'une autonomie pour fonctionner dans des environnements partagés par d'autres agents (Ferber, 1999 ; Russell, and Norvig, 2010). Un agent est capable d'évoluer dans un environnement par l'acquisition d'informations émises par ce dernier et en agissant sur celui-ci. L'architecture d'un agent varie d'une simple structure formée de règles de production à des structures très complexes où l'agent peut être doté de mécanismes avancés de cognition et de raisonnement lui référant un caractère associatif et social.

Ferber [38] définit un système multi-agents en faisant la référence à un système composé des éléments suivants :

$\langle O, A, L, R, Op, U \rangle$

où,

O, A et L constituent des ensembles de l'environnement E.

O est un ensemble d'objets manipulables par les agents (perçus, créés, détruits et modifiés).

A est un ensemble d'agents. Les agents sont les entités actives du système. Ils sont capables d'effectuer des actions.

L (Localisation) est un ensemble de lieux. Il détermine les positions possibles des agents dans l'espace.

R est un ensemble de relations pour lier des agents et des objets.

Op représente l'ensemble des opérations pouvant être effectuées par un agent.

U est un ensemble d'opérateurs appelé loi de l'univers.

### **3.3.2- L'APPROCHE PROPOSÉE**

Les travaux concernés par cette présente étude sont liés au développement de simulateurs pour les protocoles de routage dans les réseaux de capteurs sans fil. Il s'agit de proposer un cadre logiciel permettant de modéliser et de simuler les constituants d'un tel réseau, ainsi que les interactions et les communications entre les

## Chapitre 3 Les simulateurs des protocoles de routage dans les réseaux de capteurs sans fil.

---

différents nœuds capteurs puis de simuler le comportement du réseau pour en tirer les résultats concernant son impact et ses performances. Un tel outil est d'une grande importance vu les intérêts qu'il apporte.

L'approche proposée est une approche descendante, qui part du processus global puis le découpe selon les unités fondamentales (voir Figure 3.2) après les avoir identifiées.

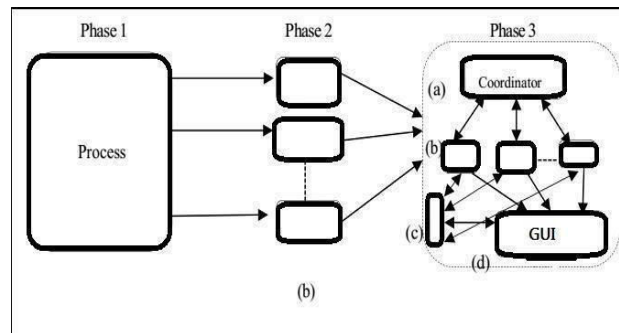


Figure 3.2- Le processus de décomposition.

Le processus de décomposition illustré figure 1 se décline selon les éléments suivants :

- (1) Dans la phase 1, le processus est étudié en détail. Le principe de son fonctionnement est clairement établi et tous ses composants atomiques sont identifiés. Cette phase concerne les investigations générales du processus. Il y'a lieu de noter, ici que cette phase est d'une importance cruciale pour le reste du processus de décomposition. L'intervention d'experts dans le domaine étudié est indispensable. Par exemple, s'il s'agit de la modélisation du processus de combustion dans un moteur diesel, l'intervention de physicien et de mécaniciens maîtrisant le domaine est indispensable pour l'explication de ce processus.
- (2) Dans la phase 2, les différentes unités composant le processus sont identifiées selon le schéma technique détaillé du procédé (b). Ces unités constitueront les groupes d'agents qui composent le système à modéliser.
- (3) Dans la phase 3, le modèle du simulateur est développé et chaque unité est isolée pour être modélisée individuellement. Les unités obtenues sont assemblées dans un schéma global.

Trois unités de contrôle et de communication supplémentaires sont ajoutées au modèle :

## Chapitre 3 Les simulateurs des protocoles de routage dans les réseaux de capteurs sans fil.

---

(a) Un coordinateur dont le rôle est de contrôler et de gérer l'exécution de l'ensemble du système.

(c) Une interface d'entrée à travers laquelle l'utilisateur peut agir sur le système.

(d) Une interface permettant de visualiser l'avancement du processus en fonction des entrées envoyées par l'utilisateur.

Nous avons adopté cette approche pour la conception de notre simulateur.

Pour la conception de ce type de simulateurs, nous avons opté pour une solution utilisant des systèmes multi-agents. Un agent représente chaque unité. Il en va de même pour le coordinateur, l'interface d'entrée ainsi que l'interface d'affichage. Notre choix s'est porté sur l'approche AGR (Agent-Group-Role). Cette approche a été adoptée au laboratoire d'informatique, robotique et mécanique de Montpellier (LIRMM). Le méta-modèle AALAADIN est un modèle d'organisation pour la modélisation des systèmes multi-agents basé sur des concepts AGR et il offre un ensemble d'outils conceptuels pour la mise en œuvre d'applications basées sur des systèmes multi-agents. Les principes de base de cette approche sont brièvement expliqués ci-dessous et illustrés dans la figure 3.3.

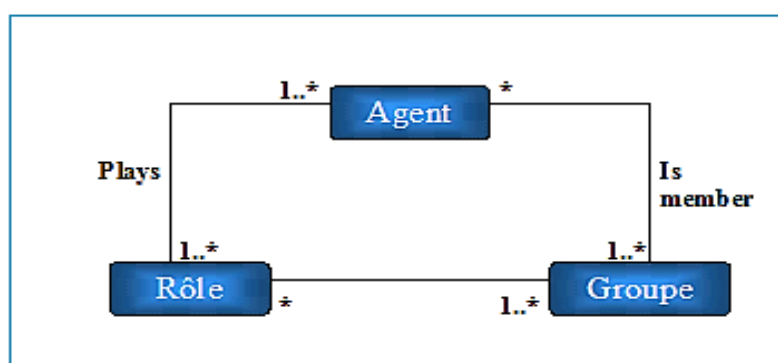


Figure 3.3- Le modèle Agent/Group/Role (AGR)

Dans le modèle AGR, un agent peut jouer un ou plusieurs rôles ; il peut être membre d'un ou de plusieurs groupes. Un groupe est composé d'au moins un agent. Un agent qui n'appartient pas à un groupe peut demander à rejoindre ce groupe. Plusieurs agents peuvent jouer le même rôle. Un agent ne peut communiquer avec un autre agent que si les deux agents sont dans le même groupe (Figure 3.4).

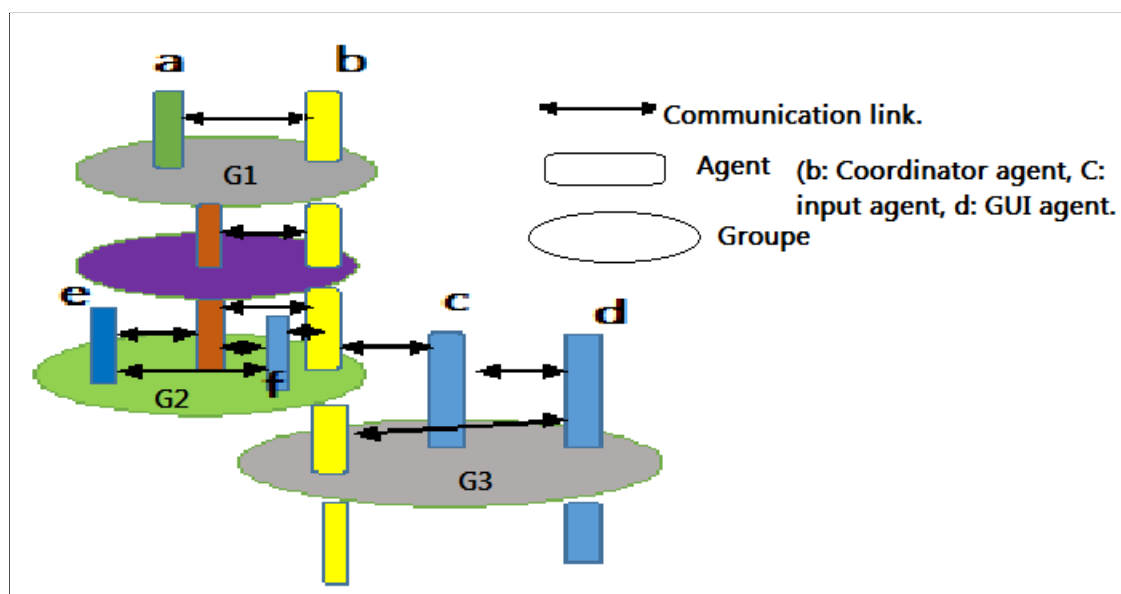


Figure 3.4- Exemple de modélisation AGR du simulateur

### 3.3.3- La plateforme MadKit

La plateforme Multi Agent Development Kit (MadKit) est un logiciel libre pour la conception et la simulation de Systèmes Multi-Agents (SMA) (Gutknecht, 2000 ; Chebbout et al., 2019). MadKit est une plate-forme multi-agents à usage général avec une couche de simulation à base d'agents sous licence GPL (General Public License) et LGPL (Lesser General Public License). Il existe plusieurs documentations en ligne, forum, FAQ et exemples sur cette plate-forme.

La plateforme MadKit est construite autour des concepts agent/groupe/rôle (AGR). Il y a trois principes de base dans cette plateforme :

- Architecture micro-noyau
- Agentification des services
- Modèle de composant pour l'interface graphique

MadKit est un ensemble de packages de classes Java. Il implémente un noyau d'agents de taille réduite et diverses bibliothèques. De plus, Madkit comprend un environnement de développement graphique et de nombreux autres systèmes de gestion d'agents. Madkit implémente différents outils de gestion des agents, des groupes et des rôles selon le modèle AGR. Cette plateforme est basée sur le méta-modèle AALAADIN.

### **3.3.4- Application à la simulation des protocoles de routage dans les RCsF**

La simulation des protocoles de routage dans les réseaux de capteurs sans fil revêt un caractère primordial et très important. En effet, l'analyse des résultats générés par la simulation permet d'évaluer l'approche proposée et de la situer par rapport à celles existantes. Elle permet également de raisonner sur un modèle abstrait tout en évitant les surcoûts de mise en œuvre de l'approche. Raisonner sur un prototype abstrait apporte un gain substantiel en termes de temps, de précision et d'argent. De plus, il est plus facile d'apporter des modifications sur un prototype abstrait que sur un produit final matériel. Un autre aspect sous-jacent de la simulation est éducatif. En effet, la simulation d'un système permet de visualiser son évolution et facilite sa compréhension. Un autre intérêt de cette démarche de modélisation/simulation concerne l'aide apportée dans le cadre du développement/implémentation de nouveaux protocoles. Il s'agit, ici, d'agir sur la représentation virtuelle du protocole à réaliser et de le 'corriger' tout le long de son évolution. Une telle approche est d'un grand intérêt pour la mise en œuvre de nouveaux protocoles.

Il existe aujourd'hui plusieurs plateformes logicielles dédiées à la simulation de protocoles de routage dans les réseaux de capteurs sans fil. Certaines de ces plateformes sont en libre accès et d'autres sont propriétaires.

### **3.3.5- Mise en œuvre de l'approche**

L'implémentation du travail que nous avons entrepris et que nous présentons ici, concerne la concrétisation d'un ensemble de concepts et d'idées de base menant à la mise en œuvre d'un simulateur de protocoles de routage basé sur les systèmes multi-agents (SMA). Les systèmes multi-agents sont un paradigme conceptuel émergent pour simuler l'interaction de plusieurs agents autonomes dans un environnement. Les systèmes multi-agents ont de nombreuses applications ; nous nous intéressons à leur utilisation pour construire un simulateur opérationnel pour les protocoles de routage dans les réseaux de capteurs sans fil. En général, un système est dit multi-agent s'il contient au moins un agent qui perçoit un environnement simulé par ses actions et réagit à cet environnement. MAS offre des solutions robustes pour les systèmes complexes et distribués. Les agents interagissent entre eux par tableau noir (mémoire partagée) ou échange de messages.

## Chapitre 3 Les simulateurs des protocoles de routage dans les réseaux de capteurs sans fil.

L'architecture proposée du simulateur repose sur le concept AGR. Ce modèle considère trois composantes principales :

- Les agents : Un agent est une entité autonome qui peut jouer divers rôles dans différents groupes. Chaque composant du simulateur est représenté par un agent. C'est l'entité atomique du système
- Les rôles : Un rôle correspond à un service exécuté par un ou plusieurs agents. Plusieurs agents peuvent jouer les mêmes rôles. Chaque unité du simulateur joue un ou plusieurs rôles à un moment donné. Les rôles attribués à une unité peuvent changer de façon dynamique.
- Les groupes : Un groupe est composé d'au moins un agent. Un agent qui n'appartient pas à un groupe peut demander à rejoindre ce groupe. Une unité du simulateur appartient à un ou plusieurs groupes et peut demander à changer de groupe de façon dynamique.

De plus, un agent est situé dans un environnement évolutif. Il peut percevoir cet environnement et y élaborer des réponses appropriées. La figure 1. Ci-dessous donne quelques éléments de base pour un système AGR.

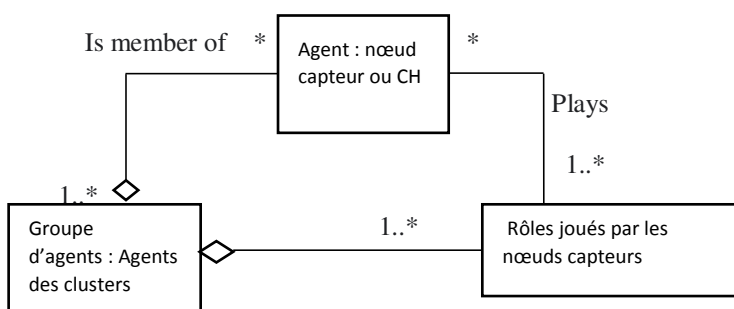


Figure 3.5. : Le modèle AGR correspondant à l'approche proposée.

### 3.3.6- La structure organisationnelle du simulateur

Pour modéliser le simulateur, nous avons utilisé le modèle organisationnel Aalaadin (Hettige et al., 2021 ; Souidi et al., 2016), qui constitue la base de notre conception. Ce modèle organisationnel se concentre sur l'analyse, la conception, la formalisation et la mise en œuvre de systèmes multi-agents basés sur les concepts AGR .

### Chapitre 3 Les simulateurs des protocoles de routage dans les réseaux de capteurs sans fil.

Par conséquent, concernant ce travail, chaque nœud et la BS sont représentés par un agent. De plus, un agent particulier : l'agent coordinateur assure la gestion des interfaces utilisateur ainsi que la supervision de l'ensemble du système.

Le simulateur est développé par l'utilisation de la plateforme multi-agents MadKit (Multi Agent Development Kit). Cette plate-forme combiné au langage JAVA est un environnement logiciel offrant des outils avancés et puissants pour la gestion des agents en utilisant les concepts AGR tels que définis dans le modèle d'organisation AALAADIN.

Concernant l'application et après une étude approfondie dans le cas de la mise en œuvre du simulateur dans le cadre des réseaux de capteurs sans fil hiérarchiques, la structure illustrée par le tableau 2 a été validée :

Tableau 3.2- Affectation AGR au sein du simulateur

Désignation	Concept AGR	Roles	Quantité
Cluster	Groupe d'agents	Tous les agents du groupe jouent des rôles identiques sauf le CH	Plusieurs clusters
Coordinateur	Agent coordinateur	Role : supervision et contrôle, assure l'interface avec l'utilisateur	Un seul agent
Cluster Head (CH)	Agent	Reçoit les données transmises par les nœuds capteurs (agents) et assure leur transmission vers la station de base	Un CH par cluster (évolutif : peut changer à chaque round)
Base station	Agent	Recueille les données issues des différents CH pour les transmettre à son tour vers l'utilisateur final en vue de leur	Une ou plusieurs stations.

## Chapitre 3 Les simulateurs des protocoles de routage dans les réseaux de capteurs sans fil.

---

		exploitation.	
Global Positioning System (GPS)	Agent	Indique les coordonnées de chaque élément du système	Un seul

### 3.4-Cas d'application

De nombreux protocoles de routage ont été spécifiquement conçus pour les réseaux de capteurs hiérarchiques. Low Energy Adaptive Clustering Hierarchy (LEACH) (Heinzelman et al., 2000) est parmi les premiers protocoles hiérarchiques le plus populaires. LEACH fonctionne par tours, à chaque tour deux opérations principales sont effectuées, la première implique la création de clusters et l'élection des cluster heads, cette phase est appelée le phase de configuration et la seconde concerne la transmission des données à la station de base par les Cluster-Heads (CH). Afin d'améliorer le protocole LEACH, plusieurs versions ont été mises au point, désignées souvent comme étant les descendants de LEACH (LEACH Family) (Khan and Marriwala 2021 ; Bhagat and Geetha, 2019). Le protocole de clustering Low Energy Adaptive Clustering Hierarchy-Centralized (LEACH-C) est une version centralisée de LEACH, les deux sont proposés par les mêmes auteurs. LEACH-C fonctionne par rounds comme LEACH et chaque round est divisé en deux phases : la phase de configuration et la phase de régime permanent. Il trouve son efficacité dans la conservation de l'énergie en particulier dans la phase de configuration en raison de la manière d'élection du CH où l'énergie résiduelle des capteurs est le facteur de sa sélection. Après ces deux protocoles populaires, de nombreuses propositions ont été faites jusqu'à nos jours, parmi lesquelles certaines sont basées sur LEACH comme Mobile-LEACH (M-LEACH) qui est une version multi-hop de LEACH qui utilise un routage multi-saut pour envoyer des données à la station de base. L'algorithme TL-LEACH dans [39] est un protocole hiérarchique à deux niveaux pour les WSN à faible énergie, l'idée principale de cette approche est l'utilisation de deux types de CH : un primaire et un secondaire. Dans [40], un protocole de communication de routage LEACH amélioré pour un réseau de capteurs sans fil propose un vice-cluster head pour chaque cluster pendant le processus de communication. DE-LEACH est un protocole de routage LEACH amélioré pour les WSN, qui divise le réseau en deux parties en fonction de la distance entre les capteurs et la station de base, l'énergie est le facteur de sélection des CH. DHCR est un algorithme de routage basé sur un

## Chapitre 3 Les simulateurs des protocoles de routage dans les réseaux de capteurs sans fil.

cluster hiérarchique à efficacité énergétique décentralisée pour les réseaux de capteurs sans fil.

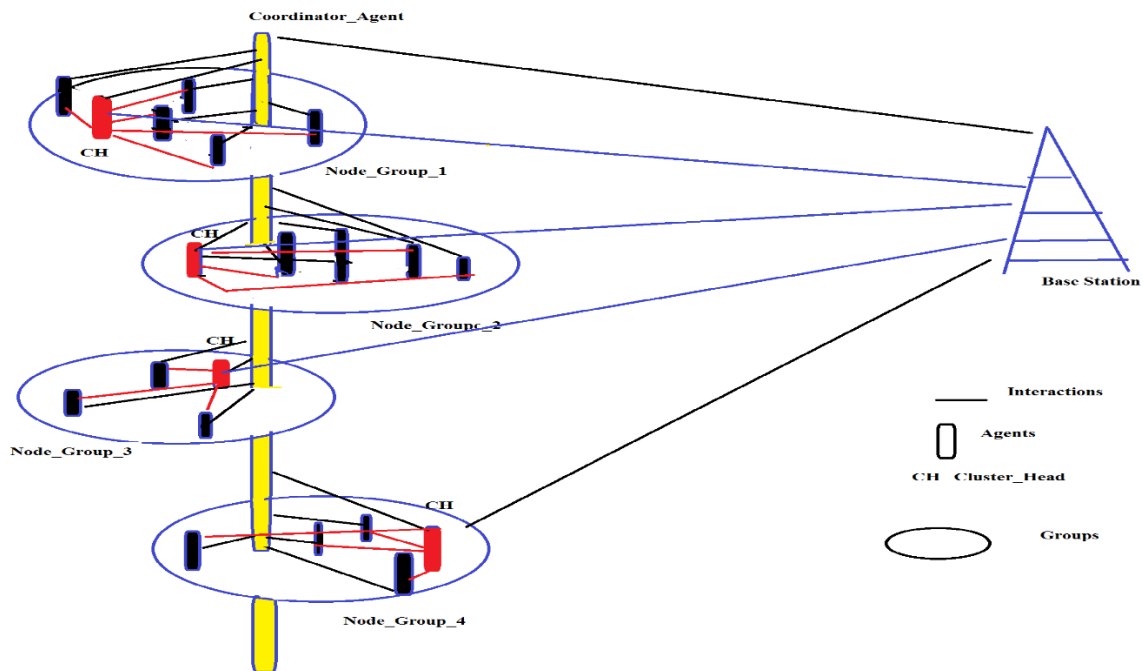


Figure 3.6- Organisation du simulateur pour le routage hiérarchique

Dans l'approche proposée, un agent représente chaque nœud capteur ; un groupe d'agents représente un cluster et un agent particulier simule le chef de cluster. Tous les agents du cluster jouent les mêmes rôles qui sont de se réveiller, d'élire le CH, de collecter les données et de les envoyer au CH puis le nœud se remet en veille. Après réception des données, le CH les agrège et les transmet à la station de base.

### 3.4.1 Quelques fonctionnalités du simulateur

Plusieurs fonctions relatives à la gestion des protocoles dans les WSN ainsi qu'à la visualisation des résultats de manière précise et ergonomique sont prévues dans le simulateur proposé. De plus, l'aspect évolutif de ce logiciel permet d'intégrer de nouveaux éléments et de nouvelles fonctions. La figure ci-dessous présente des exemples de paramètres de simulation.

## Chapitre 3 Les simulateurs des protocoles de routage dans les réseaux de capteurs sans fil.

---

TABLEAU 3.3- EXEMPLES DE PARAMÈTRES DE SIMULATION

Parameter	Value
Network size	100 sensors
Topology configuration mode	Randomized
Number of trigger nodes	05 trigger nodes
Network size	100 m x 100 m
Sensor node transmission range node	30 meters
Initial node energy	2 Joules
Radio Transmission power (Watt)	0.028 watts
Radio Reception power (Watt)	0.036 watts
MAC sub-layer	MAC IEEE 802.11
Packet size	25 Bytes
Minimum Energy	0.001 joule
Maximum duration of the simulation	3600 seconds

La figure 3.7 décrit les valeurs d'un nœud agent où : Temps de simulation Concerne le temps écoulé depuis le début de la simulation. Le numéro de cluster est le numéro du cluster actuel. Chaque nœud a un numéro d'identification (Node ID). Chaque nœud est situé dans la zone donnée et possède des abscisses et des ordonnées. Un nœud peut être dans l'un des deux états : actif ou en veille. Chaque nœud gère un tampon pour mémoriser les données avant de les transmettre au CH. Ce tampon peut être vide ou plein. La quantité d'énergie restante au niveau du nœud capteur est appelée énergie résiduelle.

```
Simulation time: 2 seconds
Cluster number: C_ID_4
Node ID: C_ID_4_6
Position coordinate: 40, 30
State of the node: Active
Data: 2 Kb
Residual energy: 0.2 J
Type: Hierarchical WSN.
```

Figure 3.7- Exemple de valeurs concernant un nœud capteur.

La figure 3.8 présente les résultats comparatifs de trois protocoles WSN hiérarchiques : LEACH-C, DE-LEACH et DHCR obtenus en utilisant notre simulateur.

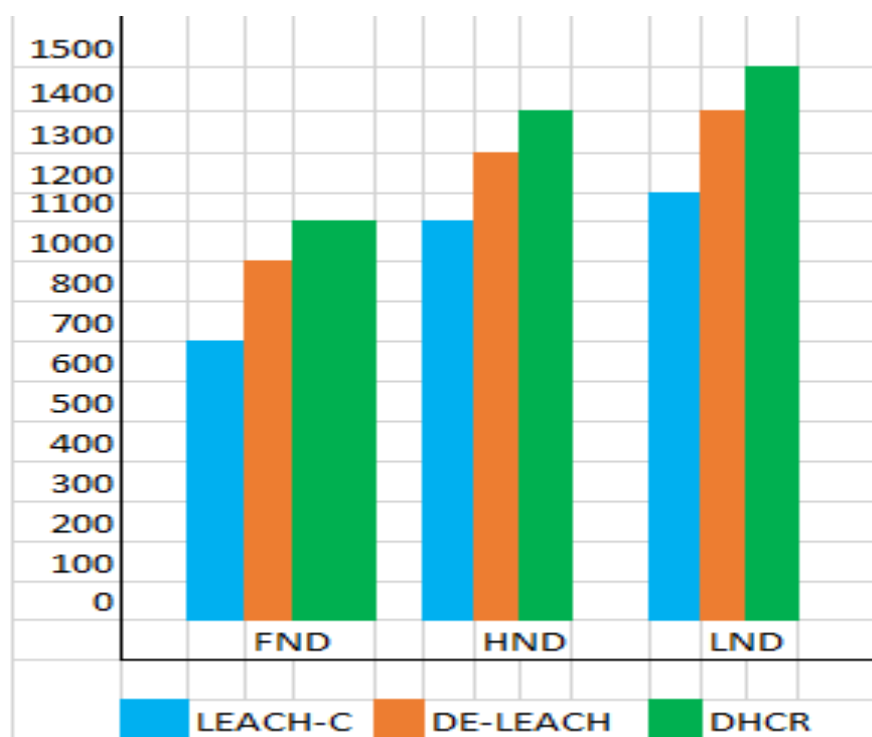


Figure 3.8- Durée de vie du réseau

### 3.4- Conclusion

Les réseaux de capteurs sans fil constituent une technologie émergente très prometteuse, extrêmement utile et largement utilisée. Les données obtenues sont acheminées depuis les nœuds capteurs vers les utilisateurs finaux distants à l'aide de protocoles de routage. Par conséquent, l'efficacité de ces dernières doit être testée en permanence. Les outils de simulation offrent des moyens d'analyse, d'évaluation et de comparaison des protocoles. Le travail présenté dans ce chapitre concerne particulièrement une contribution à la conception et à la mise en œuvre d'un simulateur dédié aux réseaux de capteurs sans fil basé sur la technologie Multi-agents. Dans ce contexte, nous avons présenté une nouvelle approche pour la conception, le développement et l'implémentation d'un tel simulateur. A l'heure de la rédaction de cette thèse, le travail projeté est encore à ses débuts et nous espérons nous pencher sur la finalisation de ce produit ainsi que son amélioration par l'ajout de nouvelles fonctionnalités dans des travaux futurs.

---

4.1-	Introduction
4.2-	Le routage géographique
4.3-	Le modèle d'énergie
4.4-	Contribution : Le protocole géographique proposé IEGGR
	4.4.1- Aperçu de l'approche
	4.4.2- Formation des sous-graphes
	4.4.3- Evaluation des performances
4.5-	Conclusion

---

## 4.1 Introduction

Les réseaux de capteurs sans fil sont formés d'un ensemble de nœuds de capteurs qui sondent leur environnement et envoient leurs données à une station de base (BS) pour de futures utilisations. Comme, énoncé dans le deuxième chapitre, de nombreux protocoles de routage ont été développés à cette fin. Une grande attention a été accordée aux protocoles de routage géographiques, qui s'appuient uniquement sur la connaissance locale des nœuds pour diffuser les données vers la BS. Dans la suite de ce chapitre, nous proposons un nouveau protocole de routage géographique économe en énergie appelé IEGGR (Enhanced Energy-aware and delivery Guarantee Geographic Routing protocol) (Redjimi et al., 2021). IEGGR est une version améliorée du protocole EGGR (Boulaiche and Bouallouche-Medjkoune, 2015). Le protocole proposé exploite les positions des nœuds voisins, qui sont les plus proches de la BS pour transmettre les paquets via le chemin le plus court en termes de consommation d'énergie. IEGGR introduit un mécanisme permettant de transmettre efficacement les paquets contournant la situation des vides. L'algorithme IEGGR considère le RCsF comme un graphe connecté et pour acheminer un paquet d'un nœud source vers la BS. IEGGR choisit l'optimum à chaque nœud intermédiaire dans l'espoir de trouver un chemin optimal global, qui vise à minimiser la consommation d'énergie. Pour trouver le chemin optimal, chaque nœud voulant effectuer un transfert de données exploite les positions de ses nœuds voisins les plus proches de la BS afin de construire itérativement le chemin optimal de bout en bout. Après avoir effectué des séries de simulation, les résultats montrent que IEGGR est efficace dans la gestion de l'énergie lors du transfert de paquets.

## 4.2 Le routage géographique

Dans le routage géographique (Kumar et al., 2017 ; Gupta et al., 2020), un nœud sélectionne son voisin plus proche de la destination comme prochain saut. Bien qu'elle soit simple et efficace, cette technique de routage échoue lorsqu'il n'y a pas de

voisin plus proche de la destination que le nœud courant. Ce problème est appelé le problème du vide (également connu sous le nom de problème des minima locaux).

Notre travail s'intéresse particulièrement à la conception d'un nouveau protocole de routage géographique qui répond à deux problèmes posés : la consommation d'énergie et la garantie de la livraison des paquets à la station de base. Nous présentons, ici, un nouveau protocole de routage géographique qui achemine les paquets par le chemin optimal en termes de consommation d'énergie entre le nœud source et la station de base basée uniquement sur la connaissance locale des nœuds (voisinage). Comme nous le verrons lors de la présentation des résultats de simulations, ce protocole augmente le ratio de livraison des paquets à la station de base.

Le routage géographique est devenu une solution attrayante pour les réseaux de capteurs sans fil où les nœuds ne conservent que des informations sur leurs voisins locaux à un saut. Dans le routage géographique, un nœud sélectionne un nœud de transfert suivant uniquement en fonction de son voisinage immédiat (à un saut) en direction de la station de base. Les informations de localisation peuvent être obtenues par GPS ou par tout autre système de localisation. Comme il n'utilise pas de paquets de contrôle pour établir un chemin, le routage géographique réduit la surcharge de contrôle de routage inondée dans le réseau pour maintenir la connectivité du réseau par rapport à d'autres types de protocoles de routage. Ainsi, les zones qui utilisent le mieux les communications entre nœuds voisins sont celles où les protocoles géographiques peuvent être appliqués avec succès ; c'est le cas de l'Internet des objets (IoT) (Lyu et al., 2019 ; Hameed et al., 2020) et des réseaux ad hoc véhiculaires (VANET) (Boussoufa-Lahlah et al., 2018) entre autres.

Les protocoles dits gloutons (*'greedy'*) (Al-Shugran et al., 2013) acheminent les paquets de telle sorte que leurs routes soient les plus proches du chemin à vol d'oiseau entre la source et la destination. Le protocole *'Nearest with Forward Progress (NFP)'* (Shivanka et al., 2016) sélectionne son voisin le plus proche parmi ceux dans la direction de la destination pour transmettre le paquet, alors qu'avec le protocole *'Most Forward within the transmission Range (MFR)'* (Luo et al., 2014), un nœud de transfert sélectionne son voisin le plus proche de la destination comme

prochain nœud de transfert. Le protocole '*Normalized ADVance (NADV)*' ( Hao et al., 2018) sélectionne le voisin avec le compromis optimal entre l'avance et le coût de la liaison. Bien que simples et générant peu de surcharge dans la maintenance de la structure de routage auxiliaire, les algorithmes de routage gourmand échouent lorsqu'il n'y a pas de voisin plus proche de la destination que le nœud actuel. Ce problème est appelé le problème du vide (également connu sous le nom de problème des minima locaux). Le problème du vide est un problème fréquent dans les réseaux de capteurs sous-marins et dans les réseaux à faible densité (Huang et al., 2017).

Le routage basé sur un graphe planaire est une stratégie de routage géographique, qui est capable de surmonter le problème de routage gourmand (problème des vides). Un graphe est dit planaire s'il peut être tracé dans le plan de sorte que ses arêtes ne se coupent qu'à leurs extrémités (Bondy and Murty, 1976). Un graphe planaire forme un ensemble de faces comme illustré figure 1 et le routage est effectué en transmettant des paquets le long des intérieurs d'une séquence de faces adjacentes, qui fournissent une progression vers la destination, en appliquant la règle de la main droite (ou gauche) (Bose et al, 2001). Les auteurs dans (Bose et al, 2001 ) et (Karp and Kung, 2000) ont proposé de combiner un algorithme glouton et un routage de faces (figure 4.1). Le principe est le suivant : initialement, le paquet est acheminé en mode glouton mais si le paquet atteint un vide, ce paquet est équilibré pour faire face au mode routage pour le faire passer autour du vide. Ensuite, le paquet est remis en mode gourmand dès que possible. (J. Kuruvila et al., 2006) ont proposé une extension de la méthode GFG, qui modélise la consommation d'énergie. Avant de transmettre un paquet à un nœud  $v$ , le nœud  $u$  vérifie s'il existe un nœud  $w$  voisin des deux nœuds  $u$  et  $v$ , de sorte que la consommation d'énergie est moindre si le paquet suit le chemin  $uwv$  plutôt que le chemin direct  $uv$ . Si c'est le cas,  $u$  envoie le paquet à  $w$ , qui traitera de la même manière.

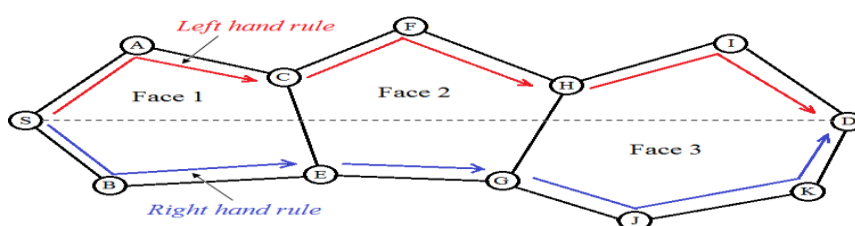


Figure 4.1 Face routing.

Dans les protocoles de routage géographique, les nœuds sélectionnent comme relais de saut suivant le voisin le plus prometteur (selon certaines métriques) parmi ceux qui sont plus proches de la destination. Si les nœuds connaissent les positions de tous les autres nœuds dans l'ensemble du réseau, le chemin optimal qui minimise la consommation d'énergie de bout en bout peut être calculé à l'aide de l'algorithme de chemin le plus court de Dijkstra (Dijkstra, 1983), où le coût de transmission entre deux nœuds représente le coût du bord entre ces nœuds. Cependant, les nœuds ne connaissent que leurs positions, les positions de leurs voisins et la position de la station de base. Dans ce travail, nous proposons un nouveau protocole de routage géographique économe en énergie. Le protocole proposé améliore le protocole EGGR (Boulaiche and Bouallouche-Medjkoune, 2015) sur les points suivants :

- 1) Plutôt que de supposer que la retransmission entre chaque voisin et la BS est optimale, IEGGR essaie de faire le choix optimal localement dans l'espoir de trouver un chemin optimal global sans aucune hypothèse.
- 2) Plutôt que de calculer le prochain saut pour chaque paquet, le prochain saut est calculé à chaque tour afin de réduire la consommation d'énergie et la surcharge de calcul.
- 3) Aucune information sur l'état du voisin n'est mise à jour et une autre stratégie est proposée afin de transmettre également les paquets autour d'un vide via le chemin optimal.
- 4) Le sous-graphe local  $G'$  est construit uniquement lors de la configuration du réseau car les nœuds de capteurs sont statiques afin de réduire la consommation d'énergie et les frais de calcul. Cependant, les poids des bords sont recalculés à chaque tour afin d'équilibrer le rôle du prochain voisin de transfert parmi plusieurs voisins.

### 4.3 Le modèle d'énergie

Nous considérons un réseau de capteurs sans fil dans lequel tous les nœuds sont statiques et limités en énergie. On suppose que chaque nœud connaît sa propre position, les positions de ses voisins, la position de la station de base et l'énergie restante de ses voisins. Les informations de localisation peuvent être obtenues en équipant les nœuds de GPS ou via tout autre système de localisation. Les

informations de localisation des voisins et les informations sur l'énergie résiduelle peuvent être obtenues par des messages de balise périodiques. Nous supposons également que les nœuds capteurs communiquent via le support sans fil et que toutes les radios sont bidirectionnelles, deux nœuds capteurs peuvent communiquer directement entre eux s'ils se trouvent dans leurs plages de transmission (portées). Chaque nœud a une portée de transmission maximale  $r$  qui peut être considérée comme étant la même pour tous les nœuds. Les nœuds capteurs effectuent la détection, la transmission et le relais des paquets vers la station de base. Nous représentons un réseau de capteurs sans fil sous la forme d'un graphe de disque unitaire connecté  $G = (E, V)$ , où l'ensemble d'arêtes  $E$  correspond aux liaisons sans fil entre les nœuds et l'ensemble de sommets  $V$  correspond aux nœuds du réseau. Dans notre protocole, nous adoptons le modèle énergétique d'Ivan Stojmenovic et Lin (Stojmenovic and Lin, 2001) où ils généralisent le modèle de Rodoplu et Meng (Rodoplu and Meng, 1999) en ajoutant un facteur linéaire 'a' et suppose que la puissance nécessaire à l'émission et à la réception d'un signal  $u(d)$  est calculée selon l'équation (4.1) :

$$u(d) = ad^\alpha + c \quad (4.1)$$

où  $d$  est la distance entre les deux nœuds,  $\alpha$  est le facteur d'atténuation du média (entre 2 et 6),  $c$  est une constante ( $c$  peut également inclure l'énergie consommée dans le traitement informatique et l'encodage/décodage à chaque nœud), et  $\alpha$  est un facteur pouvant être ajusté à l'environnement physique, unité de longueur, taille unitaire d'un signal,...

#### 4.4 Le protocole géographique proposé : IEGGR

L'idée principale de l'algorithme IEGGR est de considérer le réseau de capteurs sans fil comme un graphe connexe. Ensuite, pour acheminer un paquet d'un nœud source vers la BS, IEGGR essaie de faire le choix optimal local à chaque nœud intermédiaire dans l'espoir de trouver un chemin optimal global entre le nœud source et la station de base. Ce chemin optimal vise à minimiser la consommation d'énergie. Pour trouver ce chemin optimal, chaque nœud de transfert exploite les positions voisines les plus proches de la BS afin de construire itérativement le chemin optimal de bout en bout.

#### 4.4.1 Aperçu de l'approche

Dans le routage géographique, un nœud sélectionne un voisin, qui est plus proche de la station de base comme prochain saut. Bien qu'elle soit simple et efficace, cette technique de routage ne permet pas d'acheminer les paquets par le chemin optimal en termes de consommation d'énergie. De plus, cette technique échoue lorsqu'il n'y a pas de voisin plus proche de la destination que le nœud actuel. Ce problème est appelé le problème du vide (également connu sous le nom de problème des minima locaux).

L'idée principale d'IEGGR est d'initier un processus de calcul reposant sur l'algorithme Minimal Spanning Tree (MST). Fondamentalement, chaque nœud construit un sous-graphe local  $G' = (E', V')$ , incluant les voisins, qui sont dans la direction de la BS, c'est-à-dire les voisins qui sont plus proches de la BS que le nœud source. Nous nous référons à une telle zone comme étant la *zone de routage (routing area)*. Ensuite, un arbre de couverture minimale est calculé sur le sous-graphe local  $G'$  en utilisant l'algorithme de Prim (Prim, 1957). Cette étape fournit un ensemble de chemins partant du nœud courant vers un ensemble de feuilles. Le prochain voisin de saut est le premier nœud sur le chemin (dans la direction de l'une de ces feuilles) où la somme des poids est le minimum.

Un nœud qui rencontre un vide en direction de la BS va élargir sa zone de sous-graphe de  $\delta$  degrés pour inclure d'autres voisins. Nous nous référons à une telle zone comme étant la *zone de récupération (recovery area)*. Ensuite, il applique la même procédure et utilise le routage source pour transférer les paquets, c'est-à-dire que le chemin le plus court est inclus dans l'en-tête du paquet et le prochain saut transmet le paquet en fonction du chemin inclus dans l'en-tête du paquet. Toutes ces phases sont résumées dans l'algorithme 1. Deux problèmes se posent, à savoir (1) comment le sous-graphe local  $G'$  est-il défini, et (2) à quelle taille un nœud doit-il élargir sa zone de récupération ? et dans ce cas comment le paquet doit-il être transmis à travers cette zone pour éviter le vide ? Ces deux problèmes sont essentiellement liés et dépendent de la manière dont le mode greedy et le mode de récupération sont utilisés pour transmettre avec succès un paquet jusqu'à la BS. Ainsi, pour construire le sous-graphe local  $G'$ , seuls les liens permettant de faire avancer un paquet vers la BS doivent être considérés.

Ainsi, la taille de la zone de récupération est soumise à certains compromis. Du point de vue de la récupération, il est avantageux de rendre cette zone assez grande et large dans toutes les directions. Cependant, une zone de récupération très large peut éloigner le paquet de la BS et ainsi augmenter à la fois la consommation d'énergie et le délai de transmission de bout en

bout. Par conséquent, notre approche limite la largeur de la zone de récupération afin que la consommation d'énergie et le délai de transmission de bout en bout puissent être minimisés. Nous définissons l'ensemble A1-hop comme étant l'ensemble des nœuds qui peuvent atteindre le nœud A directement. En supposant un modèle de propagation radio en espace libre, un nœud  $X \in A1\text{-hop}$  si X est au plus éloigné de r unités du nœud A, où r est la plage de communication du nœud capteur. Ainsi, la zone de routage d'un nœud A, comme représenté sur la figure 4.2, comprend l'ensemble des nœuds qui sont les plus proches de la BS. Dans ce cas, tous les paquets reçus de l'extérieur de la zone de routage de A seront acheminés via des nœuds qui se trouvent à l'intérieur de la zone de routage de A. D'autre part, si la zone de routage est vide, c'est-à-dire qu'aucun voisin n'est plus proche de la BS que le nœud actuel ; tous les paquets reçus de l'extérieur de la zone de routage ( Routing\_Area) de A seront acheminés via des nœuds qui se trouvent à l'intérieur de la zone de récupération (Recovery\_Area) (voir Figure 4.3).

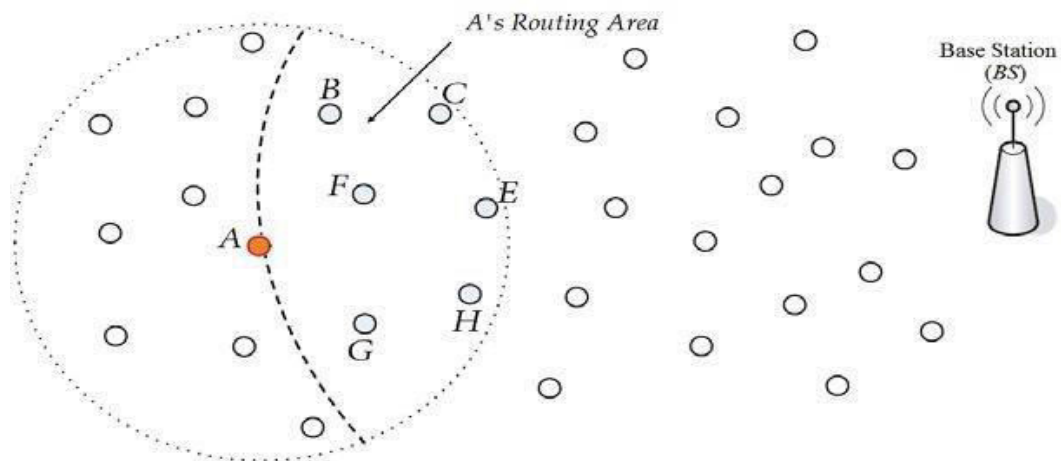


Figure 4.2- Zone de routage

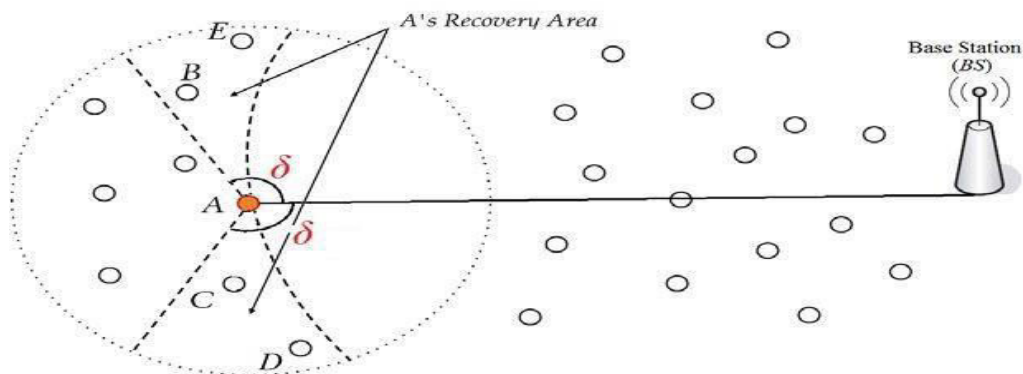


Figure 4.3- Zone de récupération

Le sous-graphe local  $G'$  construit à partir des nœuds de la zone de routage diffère de celui construit à partir des nœuds de la zone de récupération. De plus, la technique de routage appliquée à l'intérieur de la zone de routage diffère de celle appliquée à l'intérieur de la zone de récupération.

#### 4.4.2. Formation des sous-graphes

Dans les techniques de routage gloutonnes, chaque nœud est censé participer au processus d'acheminement des paquets vers la BS. Par conséquent, au moment de la configuration du réseau, chaque nœud lance le processus de sélection de son prochain voisin de transfert. Fondamentalement, ce voisin est utilisé pour transmettre tous les paquets reçus. Pour équilibrer la charge et éviter de surcharger un seul voisin, le prochain rôle de voisin de transfert est périodiquement resélectionné parmi plusieurs voisins, où un voisin sert de prochain voisin de transfert dans un tour ; par conséquent, le processus de sélection du voisin de renvoi suivant est réinitialisé à chaque tour. La durée d'un tour dépendrait naturellement de l'intensité du trafic sur le réseau. Un nœud peut également déclencher la rotation des rôles lorsque l'alimentation en énergie de son prochain voisin de transfert est faible. Chaque nœud maintient une table de ses voisins contenant : l'identifiant, la position et l'énergie résiduelle du voisin. Rappelons que chaque nœud diffuse périodiquement un message de balise contenant sa position et son énergie résiduelle. Le sous-graphe local est le graphe  $G' = (E', V')$  tel que :  $V'$  correspond à l'ensemble des voisins dans la zone de routage/La zone de récupération et  $E'$  l'ensemble des arêtes entre ces voisins. Soit  $\text{dist}(u, v)$  la distance entre les deux nœuds  $u$  et  $v$  et  $r$  la portée de communication du nœud capteur.

#### ***Algorithme 4.1: Algorithme des zones de routage et de récupération (Routing and Recovery area algorithm)***

---

$S$ : Le nœud manipulant le paquet  $P$ ;

$N(s)$ : l'ensemble des voisins de  $S$  proches de la BS;

$R(s)$ : L'ensemble des voisins de  $S$  se trouvant dans la zone de récupération ;

$H(s)$ : L'ensemble des voisins de  $S$  se situant à un seul saut de  $S$  ;

**For** each node  $u \in H(S)$  **Do**

$a = \text{dist}(s, u)$ ;

$b = \text{dist}(s, BS)$ ;

$c = \text{dist}(u, BS)$ ;

**If**  $\text{dist}(u, BS) \leq \text{dist}(s, BS)$  **Then**

        Add  $u$  to  $N(s)$ ;

```
EndIf
If  $[(b^2+c^2-a^2)/(2*b*c)] \leq \delta$  Then
    Add  $u$  to  $R(s)$ ;
EndIf
EndFor
```

---

1) Sous-graphe de la zone de routage

- Un noeud  $u \in V'$  ssi  $dist(u, BS) \leq dist(s, BS)$  et  $dist(s, u) \leq r$ .  
Un arc  $uv \in E'$  ssi  $u, v \in V'$  et  $dist(s, u) \leq r$  et  $dist(s, v) \leq r$  et  $dist(u, v) \leq r$  et  $dist(s, u) \leq dist(s, v)$ .
- Plus un arc entre le noeud courant  $S$  et chaque noeud  $u \in V'$ .
- $BS$  est la base station.

2) Sous-graphe de la zone de récupération

Soit  $a$  la distance entre  $S$  et  $u$ ,  $b$  la distance entre  $S$  et la  $BS$ , et  $c$  la distance entre  $u$  et la  $BS$ :

- Un noeud  $u \in V'$  ssi  $[(b^2+c^2-a^2)/(2*b*c)] \leq \delta$   
Un arc  $uv \in E'$  ssi  $u, v \in V'$  et  $dist(s, u) \leq r$  et  $dist(s, v) \leq r$  et  $dist(u, v) \leq r$  et  $dist(v, BS) \leq dist(u, BS)$ .
- Plus un arc entre le noeud courant  $S$  et chaque noeud  $u \in V'$ .

Où  $S$  est le noeud courant,  $BS$  la base station et  $\delta$  est la largeur de l'angle de la zone de récupération (entre la zone de routage et la zone de récupération).

Ces étapes sont résumées dans l'algorithme 4.1 présenté ci-dessus.

#### **$\alpha$ - Les poids des arcs**

Le poids de chaque arête représente l'énergie nécessaire à l'émission et à la réception entre les deux noeuds des extrémités de cet arc. Le coût de transmission/réception entre deux noeuds  $u, v \in V'$  et le coût de transmission/réception entre le noeud courant  $s$  et chaque noeud  $u \in V'$  est calculé selon l'équation (4.2).

Pour équilibrer la consommation d'énergie et la répartir régulièrement sur tous les voisins d'un noeud  $S$ , nous proposons d'ajouter une autre métrique, qui est l'énergie résiduelle  $R_{Eu}$  des noeuds voisins. Ainsi, le poids de chaque arête est basé sur deux paramètres : l'énergie d'émission/réception et l'énergie résiduelle des noeuds voisins :

$$Edge_{weight} = \frac{ad^{\alpha+c}}{R_{eu}} \quad (4.2)$$

où  $d$  est la distance entre les deux nœuds,  $\alpha$  un facteur d'atténuation du média compris entre 2 et 6,  $C$  est une constante, et  $a$  : peut être ajusté selon l'environnement physique.

---

**Algorithme 4.2 : Sous-graphe  $G'$  (Local sub-graph  $G'$  algorithm)**

---

$S$ : Le nœud manipulant le paquet  $P$ ;  
 $N(s)$ : l'ensemble des voisins de  $S$  proches de la BS;  
 $R(s)$ : L'ensemble des voisins de  $S$  se trouvant dans la zone de récupération ;  
 $H(s)$ : L'ensemble des voisins de  $S$  se situant à un seul saut de  $S$  ;  
**If**  $N(s) \neq \phi$  **Then**  
    **For** each node  $u \in N(s)$  **Do**  
        **If**  $dist(u, BS) \leq dist(s, BS)$  **Then**  
            Add Edge  $su$  to  $E'$  ;  
        **EndIf**  
    **EndFor**  
**For** each two nodes  $u$  and  $v \in N(s)$  **Do**  
    **If**  $dist(s, u) \leq dist(s, v)$  **Then**  
        Add Edge  $uv$  to  $E'$  ;  
    **EndIf**  
**EndFor**  
**For** each node  $u \in R(s)$  **Do**  
    **If**  $dist(u, BS) \leq dist(s, BS)$  **Then**  
        Add Edge  $su$  to  $E'$  ;  
    **EndIf** **EndFor**  
**For** each two nodes  $u$  and  $v \in R(s)$  **Do**  
    **If**  $dist(s, u) \leq dist(s, v)$  **Then**  
        Add Edge  $uv$  to  $E'$  ;  
    **EndIf**  
**EndFor**  
**EndIf**

---

Il convient de rappeler ici que le sous-graphe local  $G'$  n'est construit qu'au moment de la configuration du réseau car les nœuds capteurs sont supposés statiques. Cependant, les poids des arcs eux, sont recalculés à chaque tour afin d'équilibrer le rôle du prochain voisin de transfert parmi plusieurs voisins.

**b. Configuration du prochain voisin de transfert**

Comme indiqué précédemment, après avoir construit le sous-graphe local  $G'$ , un nœud applique l'algorithme de Prim afin de trouver l'arbre de couverture minimale (Minimum Spanning Tree : MST). Il est important de souligner que cet algorithme fournit la solution optimale au problème de recouvrement minimal. L'algorithme utilisé est dit algorithme de

Prim (Prim Algorithm) et sa structure pour le sous-graphe local  $G'$  est résumée dans l'algorithme 2. Un arbre couvrant est un sous-graphe d'un graphe non orienté qui contient tous les sommets et ne comporte pas de cycles. La somme des poids d'un MST est le minimum possible. Comme illustré dans l'algorithme 3, le MST consiste en un ensemble de chemins minimaux vers la BS en termes de consommation d'énergie. Le prochain voisin de transfert est le premier nœud sur le chemin avec la somme minimale des poids, comme illustré figure 4.4 b.

---

**Algorithme 4.3 : Algorithme de Prim (Prim Algorithm)**

---

1. Choose current node  $S$  to be the root node.
  2. Set  $V_t = \{S\}$  and  $E_t = \emptyset$ .
  3. Find an edge with the smallest weight such that one of its end points is in  $V_t$  and the other is in  $V \setminus V_t$ . Add this edge to  $E_t$  and its new vertex to  $V_t$ .
  4. If  $V \setminus V_t = \emptyset$ , then terminate. Otherwise, go back to Step 2.
  5. The minimum Spanning Tree is  $MST = (V_t, E_t)$ .
- 

**c. Exemple illustratif**

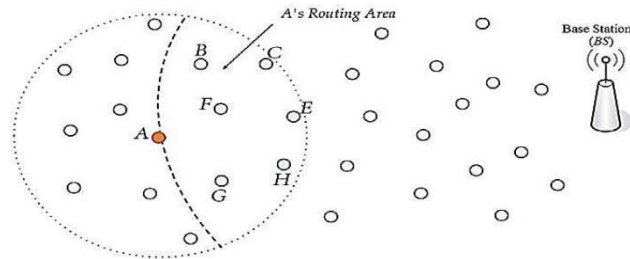
Pour illustrer le processus, considérons l'exemple de la topologie de la figure 4.4. Au moment de la configuration du réseau, le nœud A lance le processus de sélection de son prochain voisin de transfert vers la BS. La zone de routage du nœud A ( $A_{zone\_de\_routage}$ ) contient les voisins B, C, E, F, G et H. La première étape dans la sélection du prochain voisin de transfert est la formation de sous-graphes. Le sous-graphe formé par le nœud A est illustré sur la figure 4.4.b. Supposons que les poids des arcs correspondants soient tels qu'illustrés sur la figure. Après la formation du sous-graphe, le nœud A applique l'algorithme de Prim pour trouver le MST à partir du sous-graphe. Le MST calculé est illustré en bleu sur la figure 4.4.c. L'ensemble des chemins vers la BS obtenu est :

1. AFBC avec un total de poids :  $3+2+3 = 8$ .
2. AFBE avec un total de poids :  $3+2+2 = 7$ .
3. AGH avec un total de poids :  $4+2 = 6$ .

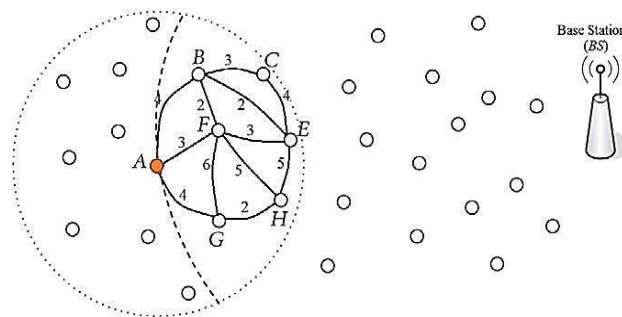
Le chemin ayant le poids minimum est AGH, de sorte que le prochain voisin de transfert du nœud A est G. Le nœud A transmet tous les paquets reçus via le nœud G jusqu'à l'expiration de la durée du tour.

En revanche, si le nœud A n'a pas de voisin plus proche de la BS, il utilisera sa zone de

récupération pour former le sous-graphe. Comme illustré sur la figure 4.5.a, la zone de récupération du nœud A ( $A_{zone\_de\_recuperation}$ ) contient les voisins B, C et D.



(a)



(b)

Figure 4.4- Sous-graphe local  $G'$  construit à partir de la zone de routage.

Dans ce cas, le sous-graphe local formé par le nœud A est illustré sur la figure 5. b. Supposons que les poids de bord correspondants sont comme illustré dans la figure 5.

Après application de l'algorithme de Prim, le MST obtenu est illustré en bleu sur la figure 5.c.

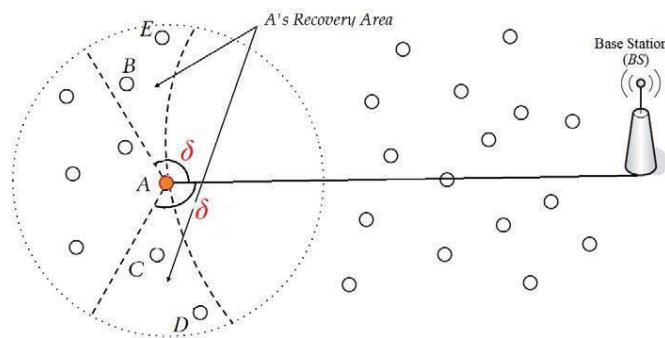
L'ensemble des chemins sur le MST sont :

1. ABE avec un total de poids :  $5+2=7$ .
2. ACD avec un total de poids :  $2+2 = 4$ .

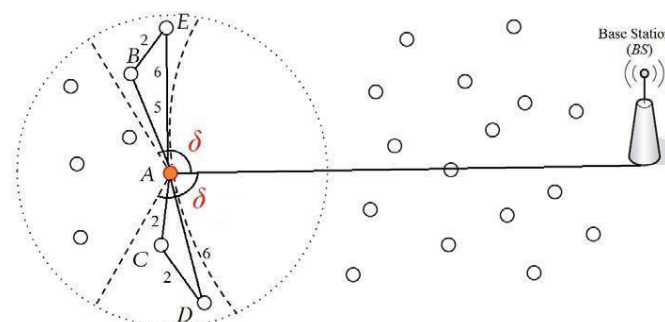
Le chemin ayant le poids minimum est ACD, de sorte que le prochain voisin de transfert du nœud A est C. Le nœud A transmet tous les paquets reçus en mode de récupération via le nœud C jusqu'à l'expiration de la durée du tour.

**d. La stratégie de routage**

Dans un réseau de capteurs sans fil, les paquets de données doivent être transmis des nœuds sources à la BS via des routes souvent multi-sauts. Un paquet a deux états dans IEGGR, à savoir le mode normal et le mode de récupération. Le mode normal permet de transférer des paquets via la zone de routage alors que le mode de récupération permet de les transférer via la zone de récupération. Initialement, un paquet est transmis en mode normal et chaque fois que le paquet rencontre un vide à un nœud  $p$ , il sera équilibré en mode de récupération pour le transmettre autour du vide. Lorsque le paquet arrive à un nœud  $q$  ayant au moins un voisin plus proche de la BS que le nœud  $p$ , le paquet est à nouveau renvoyé en mode glouton. Dans ce cas, le paquet est transmis à ce voisin. Pour ce faire, un nœud  $p$  qui équilibre un paquet en mode de récupération inclut dans l'en-tête du paquet sa position, c'est-à-dire la position où le paquet a été équilibré en mode de récupération. Le nœud  $p$  inclut également dans l'en-tête le chemin minimum par lequel le paquet sera acheminé en mode de récupération. Ensuite, chaque nœud  $q$  qui reçoit le paquet vérifie d'abord s'il a au moins un voisin plus proche de la BS que le nœud  $p$  en utilisant la position incluse dans l'en-tête du paquet. Si tel est le cas, il équilibre le paquet en mode normal et transmet le paquet à ce voisin. Sinon, il transmet le paquet en fonction du chemin minimum inclus dans l'en-tête. L'algorithme 4 résume la stratégie de routage selon notre approche. L'algorithme 4.4 ci-dessous décrit de façon approfondie l'approche que nous proposons.



(a)



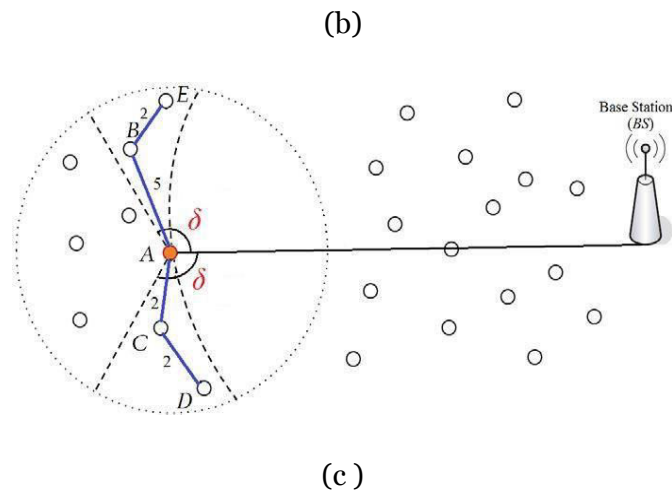


Figure 4.5- Sous-graphe local  $G'$  construit à partir de la zone de récupération.

**Algorithme 4.4: Routage des paquets (Route(Packet "P")); //Pseudo-code IEGGR.**

$S_i$  : the node that is handling packet  $P$ ;

$N(S_i)$ : the set of  $S_i$ 's neighbors which are closest to the

BS;  $R(S_i)$ : the set of  $S_i$ 's neighbors which are in the

recovery area;  $H(S_i)$ : the set of  $S_i$ 's one hop neighbors;

- 1) **If**  $N(S_i) == \phi$  **Then** // No neighbor is closer to the BS;
- 2)     **If**  $S_i$  is the source of the packet **Then**
- 3)          $Next\_Neighbor :=$  Nearest Neighbor in  $H(S_i)$ ;
- 4)     **Else**
- 5)         **If** ( $P.mode$  is in normal mode) **Then**
- 6)             Apply sub-graph  $G'$  Formation Algorithm from  $R(S_i)$ ;
- 7)             Apply Prim's Algorithm;
- 8)             Find the minimum path in the MST found;
- 9)              $Next\_Neighbor :=$  First node in the minimum path in the MST found;
- 10)            Include the minimum path in the packet's header;
- 11)         **Else** //  $P.mode$  is in recovery mode
- 12)             Check the address contained in the packet's header;
- 13)             **If** there is at least a neighbor closer to BS than  $P.Rec\_Addr$  **Then**

```
14)                                     Balance the packet to normal mode;
15)                                     Next_Neighbor := That_Neighbor;
16)                                     Else
17)                                     Next_Neighbor := first node in the path included in packet's
                                         header;
18)                                     EndIf
19)                                     EndIf
20)     EndIf
21) Else // there is neighbors closer to the BS;
22)     Apply sub-graph G' Formation Algorithm from  $N(S_i)$  ;
23)     Apply Prim's Algorithm;
24)     Find the minimum path in the MST found;
25)     Next_Neighbor := First node in minimum path in the MST found;
26) EndIf
27) Transmit P to Next_Neighbor;
```

---

Lemme 1 : La complexité d'exécution de IEGGR est  $(n \log m)$ , où  $m$  est le nombre de voisins et  $n$  est le nombre de liens entre ces voisins.

Preuve : Dans IEGGR, chaque nœud ne considère que ses voisins à un saut les plus proches de la BS pour construire le sous-graphe local où ces voisins représentent l'ensemble des sommets et les liens entre ces voisins représentent l'ensemble des arêtes. Avec une implémentation simple de l'algorithme de Prim, une recherche linéaire sur un ensemble d'arêtes pour trouver l'arête avec le plus petit poids nécessite une complexité d'exécution de l'ordre de  $m^2$ , où  $m$  est le nombre de sommets. Cependant, cette complexité d'exécution peut être réduite à  $(n \log n)$  où  $n$  est le nombre d'arêtes, en recherchant des arêtes dans un ensemble d'arêtes triées selon leurs poids. En recherchant les sommets au lieu des arêtes, la complexité d'exécution peut être encore améliorée à  $(n \log m)$ . Ainsi, la complexité d'exécution d'IEGGR est de l'ordre de  $(n \log m)$ ,  $m$  étant le nombre de sommets (nœuds voisins) et  $n$  le nombre de liens entre ces sommets.

#### 4.4.3- Evaluation des performances

L'efficacité de l'IEGGR est validée par simulation. Cette section traite des paramètres de simulation, des mesures de performance et des résultats.

##### a. Environnement de simulation et métriques de performances

Pour évaluer les performances de notre protocole, nous avons utilisé la plateforme J-Sim comme environnement de simulation. J-Sim est un simulateur très puissant particulièrement en ce qui concerne l'aspect gestion de la consommation d'énergie qui est le point le plus crucial dans les réseaux de capteurs sans fil. Les nœuds capteurs utilisés en simulation sont considérés comme homogènes : ayant la même quantité d'énergie initiale, la même capacité de calcul et de mémoire, la même portée de transmission, et équipés de la même interface de communication IEEE 802.11, l'énergie de la station de base est considérée comme illimitée. Ces nœuds capteurs sont déployés à l'intérieur d'une zone carrée de 100 sur 100 créant des topologies contenant de 20 à 160 nœuds. La position des nœuds générés dans les topologies est effectuée de manière aléatoire. Le facteur d'atténuation du support ' $\alpha$ ', le facteur linéaire ' $a$ ' et la constante ' $C$ ' utilisés pour calculer les poids des arcs sont initialisés aux valeurs suivantes :  $\alpha=2$ ,  $a= 1$  et  $C=1000$ . Le tableau suivant résume les paramètres utilisés :

Table 4.1- Paramètres utilisés dans la simulation.

Parametres	Valeurs
Nombre de nœuds capteurs	20, 40, 60, 80, 100, 120, 140, and 160
Topologie de configuration	Aléatoire
Nombre de nœuds trigger	05
Taille du réseau	100 m x 100 m
Portée de transmission des nœuds node	30 m
Energy initiale	0.5 Joules
Puissance radio de transmission (Watt)	0.028 watts
Puissance radio de réception (Watt)	0.036 watts
Sous-couche MAC	MAC IEEE 802.11
Angle $\delta$	$2\pi/3$

Pour évaluer les performances de notre protocole IEGGR et son efficacité énergétique, nous l'avons comparé avec l'étude réalisée précédemment dans [9]. La comparaison de notre protocole avec ces protocoles sera effectuée en utilisant les métriques suivantes :

- 1) **La consommation d'énergie** : l'objectif principal de notre protocole IEGGR est de minimiser la consommation d'énergie des nœuds, car elle affecte directement la durée de vie du réseau. C'est pourquoi nous comparons notre protocole avec d'autres protocoles en termes d'énergie moyenne EM consommée par tous les nœuds du

réseau (4.3):

$$EM = \frac{\sum_{i=0}^N (E_i - ER_i)}{\text{Nombre\_total\_paquets\_transmis}} \quad (4.3)$$

où  $E_i$  est l'énergie initiale du nœud  $i$ ,  $ER_i$  est l'énergie restante l'énergie du nœud  $i$ , et  $N$  est le nombre total des nœuds du réseau.

- 2) **Le taux de réussite :** Ce taux représente le rapport entre le nombre de paquets reçus par la station de base et le nombre total de paquets envoyés par tous les nœuds du réseau (4.4) :

Il est évident qu'un protocole de routage doit minimiser le nombre de paquets perdus lors de la transmission d'un nœud à la station de base, car les informations transportées peuvent être très importantes.

- 3) La latence moyenne : la latence est le temps nécessaire pour qu'un paquet soit transmis du nœud source à la station de base. Par conséquent, la latence moyenne est la somme des latences de tous les paquets divisée par le nombre total de paquets reçus.

$$\text{Latence}_{moyenne} = \frac{\sum_{i=1}^n (\text{temps\_reception}_i - \text{temps\_emission}_i)}{n} \quad (4.5)$$

où  $n$  représente le nombre de paquets reçus par la station de base.

### **b. Les résultats de la simulation**

Dans cette sous-section, nous présentons les résultats de la simulation. Les valeurs individuelles reflètent la moyenne de 30 scénarios indépendants et sont soumises à des intervalles de confiance de 90 % et restent à moins de 10 % de la moyenne de l'échantillon.

- 1) *La consommation énergétique :*

La Figure 4.6. présente l'énergie moyenne consommée par rapport au nombre de nœuds du réseau tandis que la figure 4.7 présente l'énergie moyenne consommée dans le temps. D'après les chiffres, nous pouvons noter que le protocole IEGGR améliore la consommation d'énergie moyenne dans le réseau par rapport aux autres protocoles. Cette amélioration peut être justifiée par le mécanisme appliqué par IEGGR qui achemine les paquets par le chemin optimal en termes de consommation d'énergie. Avec le protocole EGGR, le calcul du chemin

---

optimal est basé sur l'hypothèse que le coût de retransmission entre chaque voisin et la BS est optimal, ce qui n'est pas le cas dans la réalité. Cette hypothèse dégrade les performances d'EGGR par rapport au protocole IEGGR. D'autres protocoles (GPSR, NFP, MFR) sélectionnent un voisin le plus proche de la BS sans tenir compte de critères de consommation d'énergie. Ce mécanisme achemine les paquets par des chemins de bout en bout qui peuvent être coûteux en termes de consommation d'énergie surtout lorsqu'il y a un vide (mécanisme de récupération appliqué par le protocole GPSR pour sortir des vides).

De même, nous pouvons voir sur la figure 4.6 que le nombre de nœuds dans le réseau augmente l'énergie moyenne consommée pour tous les protocoles. Cela peut être justifié par le nombre de messages de contrôle et de mise à jour (messages périodiques envoyés pour mettre à jour les tables voisines) générés dans le réseau. D'autre part, le nombre de nœuds intermédiaires dans le chemin entre la source et la station de base augmente l'énergie moyenne consommée. En effet, plus il y a de nœuds dans le chemin de bout en bout, plus la consommation d'énergie est importante. Cela peut être noté avec le protocole NFP, qui sélectionne son voisin le plus proche pour transmettre les paquets.

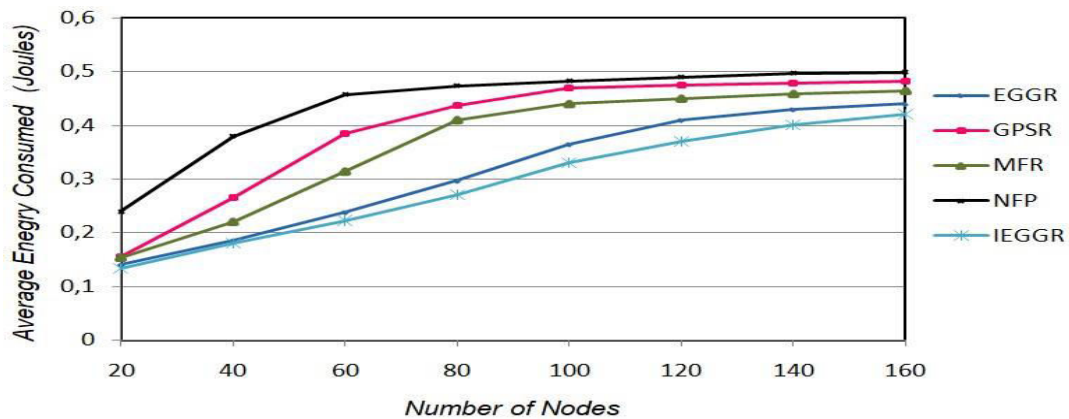


Figure 4.6-Énergie moyenne consommée par rapport au nombre de nœuds du réseau.

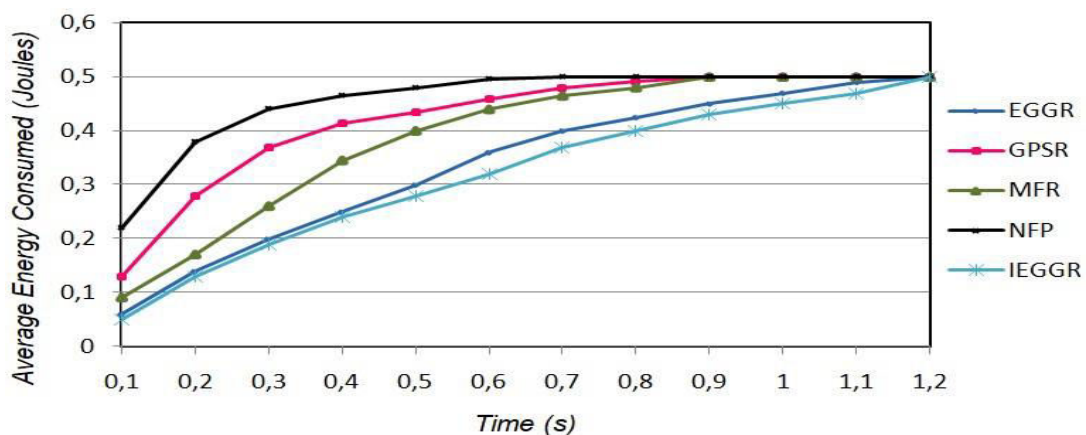


Figure 4.7- Moyenne de l'énergie consommée en fonction du temps.

On remarque sur la figure 7, qu'avec d'autres protocoles, les nœuds consomment leur énergie très rapidement par rapport au protocole IEGGR. Et cela prouve l'efficacité du mécanisme d'équilibre appliqué par notre protocole lors du calcul du plus court chemin qui distribue efficacement la consommation d'énergie sur tous les nœuds, ce qui permet aux nœuds de conserver leur énergie pendant une longue période. En effet, l'utilisation d'un critère fixe pour sélectionner le prochain voisin de transfert (qui est appliqué avec NFP, MFR et GPSR) donne le même voisin à chaque fois. L'utilisation du même voisin tout le temps pour acheminer des paquets vers la BS provoque l'épuisement rapide de son énergie.

2) *Le taux de succès*

La figure 4.8 illustre le taux de réussite marqué par tous les protocoles par rapport au nombre de nœuds dans le réseau. D'après la figure, nous pouvons voir que le taux de réussite de tous les protocoles varie entre 50 % et 99 %. Nous pouvons également voir que le nombre de nœuds dans le réseau affecte le taux de réussite ; bien sûr, plus il y a de nœuds dans le réseau, plus le taux de réussite est proche de 100 % pour tous les protocoles ; c'est-à-dire que plus il y a de nœuds dans le réseau, moins il y a de probabilité d'avoir des vides dans le réseau et moins il y a de nœuds dans le réseau, plus il y a de probabilité d'avoir des vides dans le réseau, ce qui affecte le taux de réussite. Nous notons également que le nombre de nœuds affecte le taux de réussite obtenu avec les protocoles MFR et NFP lorsque la densité du réseau est faible, ce qui prouve la nécessité d'utiliser un mécanisme de récupération pour garantir la livraison des paquets dans les WSN. Selon les résultats de la simulation, IEGGR et EGGR donnent de meilleurs taux de réussite.

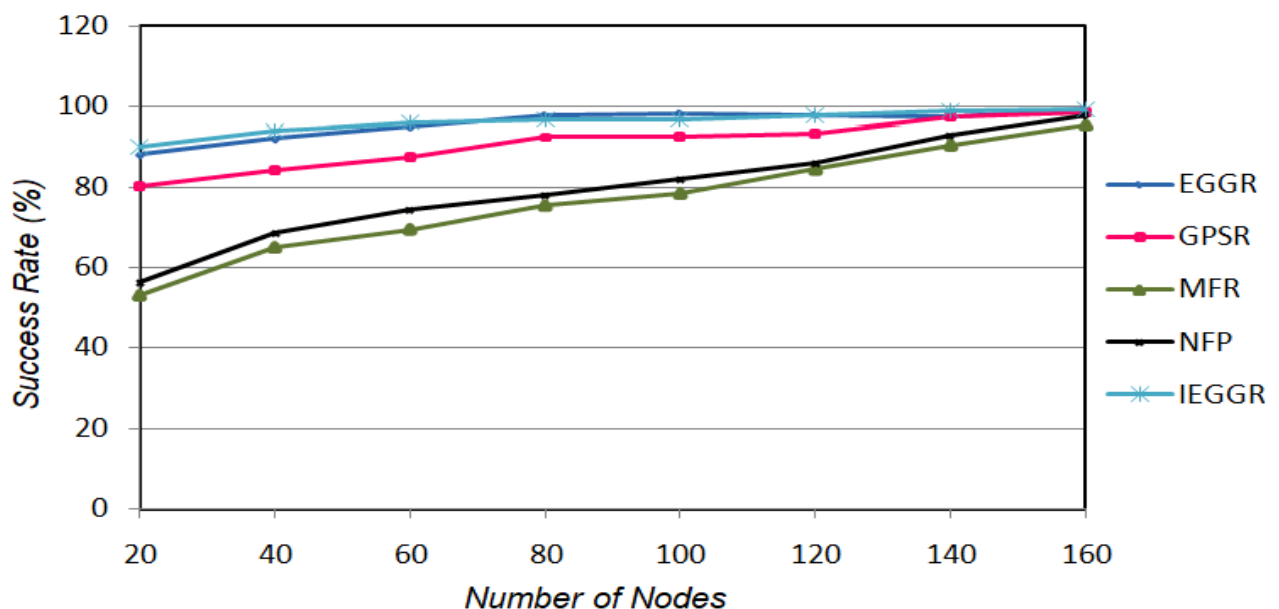


Figure 4.8- Comparaison des taux de succès.

3) La durée de vie du réseau

La figure 4.9 présente le nombre de nœuds vivants avec le temps. À partir de cette figure, nous pouvons voir qu'avec les nœuds MFR, NFP et GPSR, ils drainent leur énergie très rapidement par rapport aux protocoles EGGR et IEGGR. Cela montre que sans tenir compte de la consommation d'énergie lors du transfert de paquets, la durée de vie totale du réseau diminue. En revanche, IEGGR et EGGR, qui tentent de transmettre les paquets via un chemin optimal en termes de consommation d'énergie, augmentent la durée de vie totale du réseau.

Selon les résultats de la figure, il est clair que IEGGR permet d'améliorer la répartition du trafic par rapport à EGGR. En effet, avec EGGR, le calcul du chemin optimal est basé sur l'hypothèse que le coût de retransmission entre chaque voisin et la BS est optimal. Cette hypothèse dégrade les performances d'EGGR par rapport au protocole IEGGR où le chemin optimal est calculé avidement au fur et à mesure que le paquet progresse vers la BS.

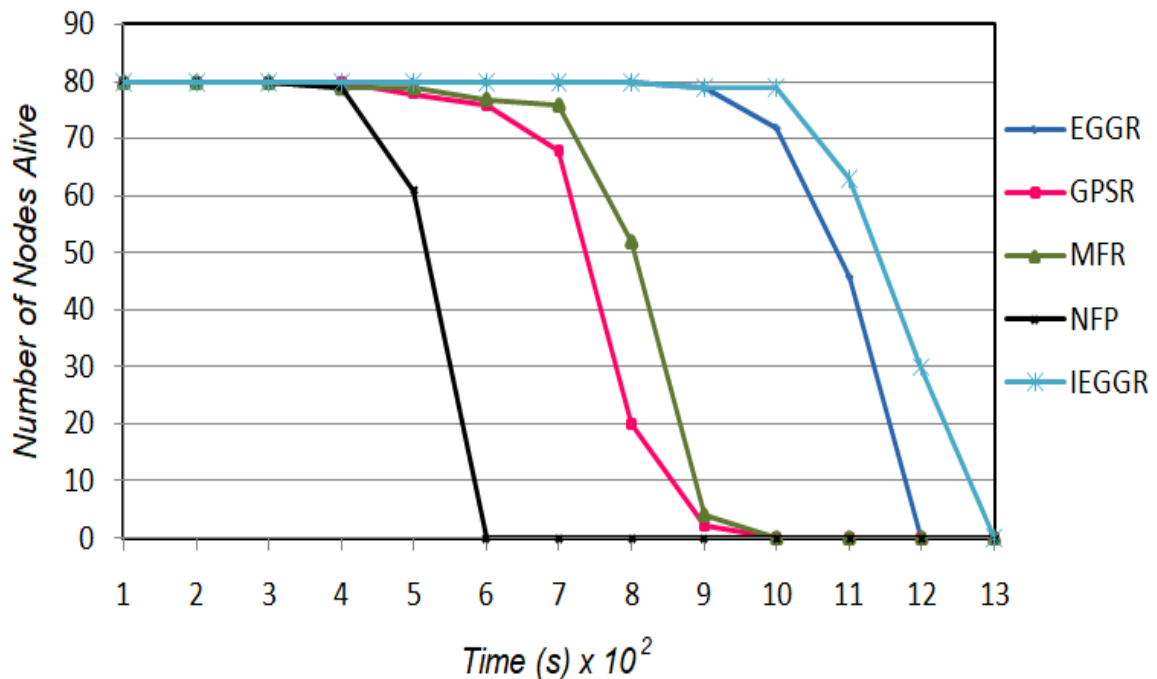


Figure 4.9- Durée de vie du réseau : nombre de nœuds actifs.

4) La latence moyenne

La Figure 4.10 montre le délai de transmission moyen de bout en bout de tous les

protocoles. Sur la figure, on peut noter que le délai de transmission de bout en bout obtenu avec le protocole de routage NFP est très élevé par rapport aux autres protocoles et cela peut être justifié par le nombre de nœuds intermédiaires sélectionnés par le protocole NFP pour transmettre les paquets du nœud source vers la destination. En effet, avec le protocole NFP, un nœud sélectionne son voisin le plus proche vers la destination comme prochain saut pour transmettre les paquets. Cette stratégie de sélection augmente le nombre de sauts le long du chemin entre la source et la BS, qui augmente le délai de transmission de bout en bout. Nous pouvons également voir que le délai de transmission de bout en bout obtenu avec le protocole MFR est meilleur que celui obtenu avec les autres protocoles et cela est dû au mécanisme de récupération appliqué par ces protocoles. Lorsque le nombre de nœuds dans le réseau est inférieur ou égal à 80, la latence moyenne obtenue avec le protocole IEGGR est similaire à celle obtenue avec MFR, EGGR et GPSR. Cependant, lorsque le nombre de nœuds est supérieur à 80, le délai de transmission de bout en bout obtenu avec le protocole IEGGR est légèrement supérieur à celui obtenu avec ces protocoles. Cela peut être justifié par le nombre de voisins qu'un nœud doit traiter pour construire le sous-graphe et MST afin de trouver le prochain voisin de transmission. Selon ces résultats, nous pouvons voir que le protocole MFR donne le meilleur délai de transmission de bout en bout.

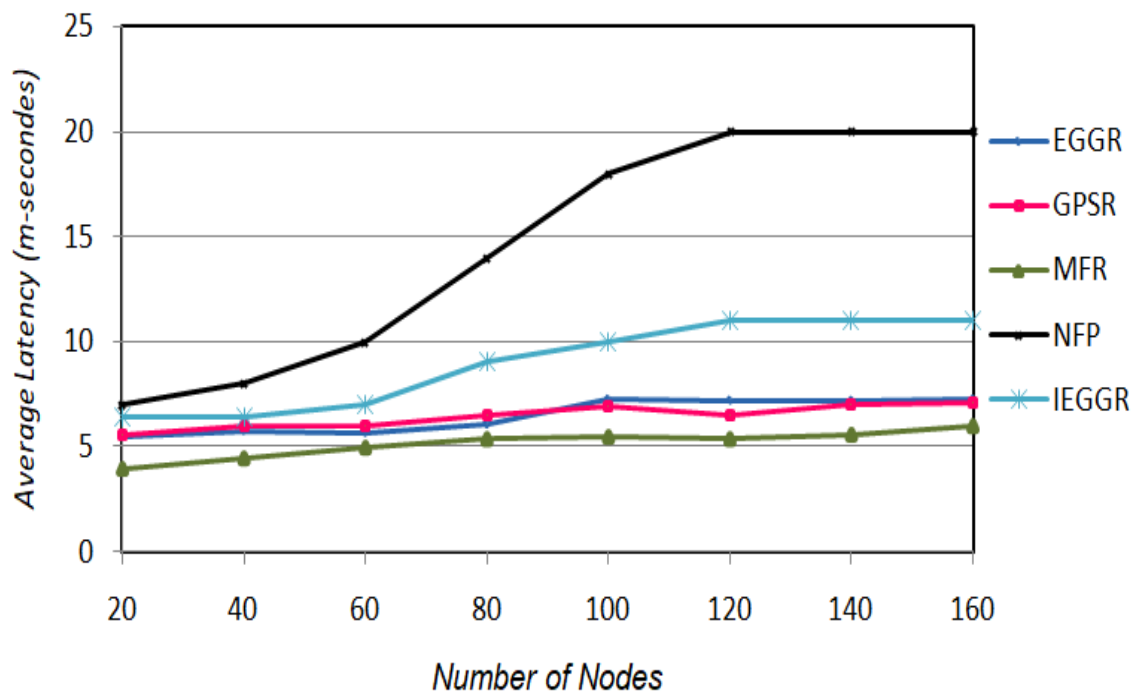


Figure 4. 10- Délais de transmission.

#### 4.5- Conclusion

Le fonctionnement d'un réseau de capteurs sans fil repose principalement sur une communication multi-sauts de nœuds qui coopèrent entre eux pour connecter les nœuds distants à la BS. De nombreux protocoles de routage géographique pour les réseaux de capteurs sans fil ont été proposés. Cependant, certains de ces protocoles ne tiennent pas compte de la consommation d'énergie. Dans cet article, nous avons proposé un nouveau protocole de routage géographique qui conserve l'énergie des nœuds et garantit la livraison des paquets dans les réseaux de capteurs sans fil basés uniquement sur la connaissance locale des nœuds (voisinage). Ce protocole, que nous avons baptisé IEGGR (Improved Energy-aware and delivery Guarantee Geographic Routing protocol) prend en considération les contraintes imposées par les nœuds capteurs, permettant une gestion efficace des ressources énergétiques lors de la communication des données dans le réseau, en introduisant un mécanisme pour calculer le chemin le plus court dans termes de consommation d'énergie lors du calcul du nœud suivant. D'autre part, la garantie de livraison est assurée en introduisant un mécanisme de récupération qui permet de transférer les paquets autour des vides. La comparaison des performances entre notre protocole et d'autres protocoles est réalisée à l'aide du simulateur open source J-Sim. Après avoir réalisé plusieurs simulations et analysé les résultats obtenus, nous avons constaté que notre protocole réduisait la quantité d'énergie consommée au niveau des nœuds de capteurs, ce qui allongeait toute la durée de vie du réseau. Des séries de simulations ont également prouvé que notre protocole (IEGGR) augmente le taux de livraison des paquets entre les nœuds et la base station en particulier dans les réseaux à haute densité. Cependant, notre protocole influence légèrement le délai de transmission de bout en bout dans les réseaux à haute densité. Néanmoins, l'approche proposée présente une limite, qui est le problème des « très grands vides ». Ce problème se produit lorsque le chemin complet inclus dans l'en-tête du paquet ne permet pas de rencontrer le vide en mode de récupération. Dans ce cas, le paquet peut retourner au nœud où le mode de récupération a été initié et retomber dans une boucle. Une solution potentielle à une telle situation consiste à empêcher le renvoi du paquet vers ce nœud et le transfert du paquet hors de cette zone. Une autre solution pour une telle situation consiste à transmettre le paquet vers la direction opposée par le nœud où le mode de récupération a été lancé. Des propositions pour éviter cette limitation fera l'objet de nos travaux futurs afin d'optimiser la consommation d'énergie lors de la transmission de paquets en mode de récupération lors de la rencontre de grandes zones blanches.

La technologie a façonné la face du monde et nous vivons, à l'ère du numérique. Les machines se dotent de plus en plus d'intelligence et de nouvelles fonctions et fonctionnalités sont sans cesse ajoutées. Nos comportements et nos gestes les plus simples, voire les plus intuitifs sont conditionnés par les machines. L'écriture manuscrite, qui était à l'origine des anciennes civilisations est, aujourd'hui recalée face à la concurrence d'éditeurs électroniques surpuissants et son utilisation se fait de plus en plus rare voire même inexistante dans les cadres officiels.

L'un des outils technologiques qui comptent parmi les plus puissants aujourd'hui concerne les échanges d'information sans support filaire. L'information est véhiculée à travers le milieu ambiant depuis une source d'émission vers une source destination de façon ubiquitaire. Ce partage d'information entre plusieurs unités leur permet de se synchroniser, de communiquer, de coopérer à l'exécution de tâches parallèles et distribuées.

Le monde a connu au cours des trois dernières années écoulées une crise sanitaire meurtrière due au coronavirus. Les relations physiques entre personnes ont été écartées. L'utilisation des nouvelles technologies de l'information et de la communication (NTIC) basées essentiellement sur les réseaux sans fil ont permis de remplacer une part importante des activités 'présentielles' par des communications à distance. C'est le cas de l'enseignement, du travail à distance et autres activités financières, gouvernementales, médicales et autres.

Bien entendu, les communications sans fil ne remplacent pas totalement le filaire mais elles le complètent très fortement. Les grands débits d'informations sont acheminés grâce à des câbles et autres fibres optiques, les réseaux sans fils prennent le relais pour acheminer l'information aux utilisateurs finaux tout comme une artère amène le flux sanguin qui est ensuite distribué par des milliers de fines capillaires.

Les réseaux de capteurs sans fil, objet du présent travail sont à la base de recueil de l'information source et de sa dissémination vers des stations de travail. Ces réseaux utilisent différentes ondes radio pour propager les signaux porteurs d'information. Cependant, si les bases technologiques matérielles permettent la mise en œuvre de centaines voire de milliers de nœuds capteurs disséminés de façon aléatoire ou prédéterminée dans de vastes zones d'intérêt, il y'a lieu de fournir des algorithmes performants permettant d'optimiser les dépenses énergétiques nécessaires au maintien vital de ces minces unités dont les ressources énergétiques sont assez limitées et parfois non remplaçable. D'autres aspects et non des moindres sont à

considérer s'agissant de ces systèmes, il s'agit de leur sécurité, de leur auto-organisation et bien d'autres qualité de service tels que l'assurance de l'acheminement correct des paquets à travers le réseau et de leurs contenus.

Ainsi et vue l'importance que revêt cette question, de nombreux protocoles de routage ont été imaginés et mis en œuvre offrant des mécanismes de plus en plus raffinés et de plus en plus évolués pour résoudre efficacement les impératifs énoncés ci-dessus.

Les outils et plateformes de modélisation et de simulation existants sont d'une aide cruciale pour le développement de nouvelles approches à moindre coût. Ainsi, des systèmes entiers peuvent être modélisés et leurs impacts et performances analysés avant qu'ils ne soient réalisés. Ceci permet de raisonner sur des prototypes virtuels tout en y apportant les correctifs éventuels de façon itérative avant leur réalisation tout en les comparant avec d'autres systèmes existants.

Ce travail de thèse entre dans cette optique très motivante à nos yeux. La problématique principale que nous avons eu à résoudre concerne la mise en œuvre d'une approche permettant d'éviter les 'vides' dans l'acheminement de l'information entre les nœuds capteurs et la station de base. Un nouveau protocole de routage géographique a été développé pour l'occasion. La modélisation et la simulation de ce protocole via la plateforme J-SIM montre l'efficacité de cet algorithme en le comparant à plusieurs autres approches existantes.

Bien d'autres aspects des réseaux de capteurs sans fil ont été abordés durant ce travail de thèse ; il s'agit de la proposition d'un simulateur basé sur les systèmes multi-agents (SMA) et plus particulièrement sur le concept Agent/Groupe/rôle (AGR). Le travail initié dans cette thèse est loin d'être terminé et beaucoup d'efforts restent à consentir dans ce sens. Au moment de l'écriture de cette thèse seul un noyau logiciel a été développé.

### **Travaux futurs :**

Comme tout travail de recherche ; l'engagement dans un axe ouvre la voie à plusieurs autres, le domaine des réseaux de capteurs sans fil présente des voies de recherche très fécondes et variées. La suite des travaux initiés dans cette thèse concernent la généralisation d'approches de recherche de chemins de propagation de l'information dans les réseaux de capteurs sans fil dans des zones peu denses, grises voire même blanches.

Un deuxième axe que nous voudrions bien tenter d'approfondir concerne l'approfondissement du développement de différents aspects du simulateur des réseaux de capteurs sans fil basé

SMA entrepris dans le cadre de cette thèse. En effet, plusieurs fonctions et fonctionnalités doivent être greffées au produit que nous avons entrepris de réaliser.

La technologie avance à grand pas et les systèmes deviennent de plus en plus intelligents, les machines apprennent de façon très rapide. Les algorithmes d'intelligence artificielle combinent bon nombre de concepts nouveaux et manipulent des quantités énormes d'informations véhiculées sans cesse sur le net. Certains pays ont difficilement accès à ces nouvelles technologies alors que le partage de l'information est devenu une nécessité absolue, les technologies sans fil associées à des dispositifs dont les coûts sont maîtrisables sont une bouée de sauvetage pour ces pays. Ainsi, le téléphone portable est devenu un outil à usage multiple à portée du pouvoir d'achat de ces populations. De cette problématique se dégage un axe de recherche dont l'idée est d'optimiser les coûts de communication dans les réseaux de téléphonie sans fil. Le problème fondamental considéré ici concerne surtout l'optimisation des débits des données, particulièrement les images qui nécessitent des grandes capacités de transmission et de stockage.

Aalsalem, M. Y., Khan, W. Z., Gharibi, W., Khan, M. K., & Arshad, Q. (2018). Wireless Sensor Networks in oil and gas industry: Recent advances, taxonomy, requirements, and open challenges. *Journal of network and computer applications*, 113, 87-97.

Abar, S., Theodoropoulos, G.K., Lemarinier, P., O'Hare, G.M. (2017). Agent based modelling and simulation tools: A review of the state-of-art software. *Computer Science Review*, 24: 13-33. <https://doi.org/10.1016/j.cosrev.2017.03.001>.

Abdulkarem, M., Samsudin, K., Rokhani, F. Z., & A Rasid, M. F. (2020). Wireless sensor network for structural health monitoring: a contemporary review of technologies, challenges, and future direction. *Structural Health Monitoring*, 19(3), 693-735.  
doi:10.1177/1475921719854528

Abdulrahman, E. K. G. (2019). *Performance Analysis of Ad-hoc On-demand Distance Vector and Destination Sequenced Distance Vector protocols* (Doctoral dissertation, Sudan University of Science and Technology).

Abdulsahib, G. M., & Khalaf, O. I. (2018). An improved algorithm to fire detection in forest by using wireless sensor networks. *International Journal of Civil Engineering & Technology (IJCIET)*, 9(11), 369-377.

Abuarqoub, A., Hammoudeh, M., Alfayez, F., Aldabbas, O., & Yurish, S. Y. (2016). A survey on wireless sensor networks simulation tools and testbeds. *Sensors, transducers, signal conditioning and wireless sensors networks advances in sensors: reviews*, 3(14), 283-302.

Adel, S. A., & Tijare, P. A. (2010). Performance Comparison of AODV DSDV OLSR and DSR Routing Protocols in MANET. *International Journal of Information Technology and Knowledge Management*, 2(2), 545-548.

Aderohunmu, F. A., & Deng, J. D. (2009). An enhanced stable election protocol (sep) for clustered heterogeneous wsn. *Department of Information Science, University of Otago, New Zealand*.

Ahmed, E. F., Omar, M. A., Wan, T. C., & Altahir, A. A. (2018). Work in Progress: LEACH-Based Energy Efficient Routing Algorithm for Large-Scale Wireless Sensor Networks. *Journal of Telecommunication, Electronic and Computer Engineering (JTEC)*, 10(1-5), 83-87.

Akyildiz, I. et al. A Survey on Sensor Networks. *IEEE Communications Magazine*, 2002.

Ali, A., Jadoon, Y. K., Changazi, S. A., & Qasim, M. (2020, November). Military operations: Wireless sensor networks based applications to reinforce future battlefield command system. In *2020 IEEE 23rd International Multitopic Conference (INMIC)* (pp. 1-6). IEEE.

Al-Shugran, M., Ghazali, O., Hassan, S., Nisar, K., & Arif, A. S. M. (2013). A qualitative comparison evaluation of the greedy forwarding strategies in Mobile Ad Hoc Network. *Journal of Network and Computer Applications*, 36(2), 887-897.

- Amjad, Muhammad; Sharif, Muhammad; Afzal, Muhammad; Kim, Sung (2016). *TinyOS-New Trends, Comparative Views and Supported Sensing Applications: A Review*. *IEEE Sensors Journal*, (), 1–1. doi:10.1109/JSEN.2016.2519924
- Arnott, R., Rave, T., & Schöb, R.: *Alleviating urban traffic congestion*. MIT Press Books (2005).
- Bakare, B. I., & Enoch, J. D. (2019). A review of simulation techniques for some wireless communication system. *International Journal of Electronics Communication and Computer Engineering*, 10(2), 60-70.
- Bala, T., Bhatia, V., Kumawat, S., & Jaglan, V. (2018). A survey: issues and challenges in wireless sensor network. *Int. J. Eng. Technol*, 7(2), 53-55.
- Balfaqih, Z. (2020). Design and development of network simulator module for distributed mobility management protocol. *Journal of Ambient Intelligence and Humanized Computing*, 1-15.
- Bandini, S., Manzoni, S., Vizzari, G. (2009). Agent based modeling and simulation: an informatics perspective. *Journal of Artificial Societies and Social Simulation*, 12(4): 4.
- Behera, T. M., Mohapatra, S. K., Samal, U. C., Khan, M. S., Daneshmand, M., & Gandomi, A. H. (2019). I-SEP: An improved routing protocol for heterogeneous WSN for IoT-based environmental monitoring. *IEEE Internet of Things Journal*, 7(1), 710-717.
- Belghith, A., & Obaidat, M. S. (2016). Wireless sensor networks applications to smart homes and cities. In *Smart cities and homes* (pp. 17-40). Morgan Kaufmann. doi:10.1016/b978-0-12-803454-5.00002-x
- Belkhala, S., Benhadou, S., Boukhdar, K., & Medromi, H.: Smart parking architecture based on multi agent system. *Int. J. Adv. Comput. Sci. Appl*, 10, 378-382. (2019).
- Benhaddou, Driss; Al-Fuqaha, Ala (2015). *Wireless Sensor and Mobile Ad-Hoc Networks || Introduction to Wireless Sensor Networks*. , 10.1007/978-1-4939-2468-4(Chapter 1), 3–32. doi:10.1007/978-1-4939-2468-4\_1
- Benzerbadj, A. (2018). *Approche inter-couches pour l'économie d'énergie et la fiabilité dans les Réseaux de Capteurs Sans Fil dédiés aux Applications Critiques de Surveillance* (Doctoral dissertation, Brest).
- Bhagat, A., & Geetha, G. (2019, November). A review of LEACH successors using single-hop and multi-hop communication model. In *International Conference on Futuristic Trends in Networks and Computing Technologies* (pp. 3-13). Springer, Singapore.
- Bondy, J. A., & Murty, U. S. R. (1976). *Graph theory with applications* (Vol. 290). London: Macmillan.

- Borges, L. M., Velez, F. J., & Lebres, A. S. (2014). Survey on the characterization and classification of wireless sensor network applications. *IEEE Communications Surveys & Tutorials*, 16(4), 1860-1890.
- Bose, P., Morin, P., Stojmenović, I., & Urrutia, J. (2001). Routing with guaranteed delivery in ad hoc wireless networks. *Wireless networks*, 7(6), 609-616.
- Bossel, H. (2018). *Modeling and Wetter, M. (2019). A view on future building system modeling and simulation. In Building performance simulation for design and operation (pp. 631-656). Routledge. simulation.* AK Peters/CRC Press.
- Boukerche, A., Turgut, B., Aydin, N., Ahmad, M. Z., Bölöni, L., & Turgut, D. (2011). Routing protocols in ad hoc networks: A survey. *Computer networks*, 55(13), 3032-3080.
- Boulaiche, M., & Bouallouche-Medjkoune, L. (2015). EGGR: Energy-aware and delivery Guarantee Geographic Routing protocol. *Wireless Networks*, 21(6), 1765-1774.
- Boussoufa-Lahlah, S., Semchedine, F., & Bouallouche-Medjkoune, L. (2018). Geographic routing protocols for Vehicular Ad hoc NETWORKS (VANETs): A survey. *Vehicular Communications*, 11, 20-31.
- Burhan, M., Rehman, R. A., Khan, B., & Kim, B. S. (2018). IoT elements, layered architectures and security issues: A comprehensive survey. *Sensors*, 18(9), 2796.
- C. Gharat, , S. Krishnan,. "Effects of Duplicate Packet Transmission in Timer based Co-ordination Opportunistic Routing Scheme," In 2019 International Conference on Smart Systems and Inventive Technology (ICSSIT) (pp. 401-405). IEEE, (2019, November).
- Campanile, L., Gribaudo, M., Iacono, M., Marulli, F., & Mastroianni, M. (2020). Computer network simulation with ns-3: A systematic literature review. *Electronics*, 9(2), 272.
- Cao, N., Liu, P., Li, G., Zhang, C., Cao, S., Cao, G., ... & Gupta, B. B. (2018). Evaluation models for the nearest closer routing protocol in wireless sensor networks. *IEEE Access*, 6, 77043-77054.
- Cao, N., Wang, Y., Ding, J., Zhou, C., Li, Y., Zhang, Y., ... & Li, H. (2017, July). The comparisons of different location-based routing protocols in wireless sensor networks. In *2017 IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC)* (Vol. 2, pp. 324-327). IEEE.
- Chandanashree, V. C., Bhat, U. P., Kanade, P., Arjun, K. M., Gagandeep, J., & Hegde, R. M. (2017, August). Tinyos based WSN design for monitoring of cold storage warehouses using internet of things. In *2017 International conference on Microelectronic Devices, Circuits and Systems (ICMDCS)* (pp. 1-6). IEEE.

- Chebout, M. S., Mokhati, F., Badri, M., & Babahenini, M. C. (2019). Monitoring open multi-agent systems: An aspect-oriented programming based approach. *Multiagent and Grid Systems*, 15(2), 155-177.
- Chen, D., & Varshney, P. K. (2007). A survey of void handling techniques for geographic routing in wireless networks. *IEEE Communications Surveys & Tutorials*, 9(1), 50-67.
- Chen, F., Ren, W. (2019). On the control of multi-agent systems: A survey. *Foundations and Trends in Systems and Control*, 6(4): 339-499. <http://dx.doi.org/10.1561/26000000019>
- Chen, M., Miao, Y., & Humar, I. (2019). Introduction to OPNET network simulation. In *OPNET IoT Simulation* (pp. 77-153). Springer, Singapore.
- Chen, S., Zhang, H., Guan, J., Rao, Z. (2020). Agent-based modeling and simulation of stochastic heat pump usage behavior in residential communities. In *Building Simulation*, 13(4): 803-821. <https://doi.org/10.1007/s12273-020-0625-2>
- Coutinho, R. W., Boukerche, A., & Loureiro, A. A. (2020). A novel opportunistic power controlled routing protocol for internet of underwater things. *Computer Communications*, 150, 72-82. doi:10.1016/j.comcom.2019.10.020
- Davis, R. (1980). Report on the Workshop on Distributed AI. [https://dspace.mit.edu/bitstream/handle/1721.1/41155/AI\\_WP\\_204.pdf?sequence=4](https://dspace.mit.edu/bitstream/handle/1721.1/41155/AI_WP_204.pdf?sequence=4).
- Deng, R., Yang, Z., Chow, M. Y., & Chen, J. (2015). A survey on demand response in smart grids: Mathematical models and approaches. *IEEE Transactions on Industrial Informatics*, 11(3), 570-582.
- Dijkstra, E. W. (1983). Solution of a problem in concurrent programming control (reprint). *Communications of the ACM*, 26(1), 21-22.
- Dorathy, I., & Chandrasekaran, M. (2018). Simulation tools for mobile ad hoc networks: a survey. *Journal of applied research and technology*, 16(5), 437-445.
- Elbhiri, B., Saadane, R., & Aboutajdine, D. (2010, September). Developed Distributed Energy-Efficient Clustering (DDEEC) for heterogeneous wireless sensor networks. In *2010 5th International Symposium On I/V Communications and Mobile Network* (pp. 1-4). IEEE.
- Eswaran, A., Rowe, A., & Rajkumar, R. (2005, December). Nano-rk: an energy-aware resource-centric rtos for sensor networks. In *26th IEEE International Real-Time Systems Symposium (RTSS'05)* (pp. 10-pp). IEEE.
- Faheem, M., Butt, R. A., Raza, B., Ashraf, M. W., Begum, S., Ngadi, M. A., & Gungor, V. C. (2019). Bio- inspired routing protocol for WSN- based smart grid applications in the context of Industry 4.0. *Transactions on Emerging Telecommunications Technologies*, 30(8), e3503.
- Fahmy, H. M. A. (2020). Wireless sensor networks essentials. In *Wireless Sensor Networks* (pp. 3-39). Springer, Cham. doi:10.1007/978-3-030-29700-8.

- Fellir, F., El Attar, A., Nafil, K., Chung, L. (2020). A multi-Agent based model for task scheduling in cloud-fog computing platform. In 2020 IEEE International Conference on Informatics, IoT, and Enabling Technologies (ICIOT), pp. 377-382. <https://doi.org/10.1109/ICIOT48696.2020.9089625>.
- Ferber, J. (1999). Multi-Agent Systems. An Introduction to Distributed Artificial Intelligence. Addison-Wesley.
- Ferber, J., Gutknecht, O., Michel, F. (2003). From agents to organizations: an organizational view of multi-agent systems. In International Workshop on Agent-Oriented Software Engineering, pp. 214-230. [https://doi.org/10.1007/978-3-540-24620-6\\_15](https://doi.org/10.1007/978-3-540-24620-6_15)
- Fishwick, P.A. (1995). Simulation Model Design and Execution: Building Digital Worlds. Prentice Hall PTR.
- Fishwick, P.A. (1997). Computer simulation: Growth through extension. Transactions of the Society for Computer Simulation, 14(1):13-24.
- Fujimoto, R. (2015, December). Parallel and distributed simulation. In 2015 Winter Simulation Conference (WSC) (pp. 45-59). IEEE.
- Gupta, N. K., Yadav, R. S., & Nagaria, R. K. (2020). 3D geographical routing protocols in wireless ad hoc and sensor networks: An overview. *Wireless Networks*, 26(4), 2549-2566. Lyu,
- Gupta, R., Pradhan, S., Haridas, A., & Karia, D. C.: Cloud based smart parking system. In 2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT) (pp. 341-345). IEEE. (2018, April).
- Gutknecht, O., Ferber, J., Michel, F. (2000). MadKit: une architecture de plate-forme multi-agent générique (Doctoral dissertation, Lirmm, University of Montpellier).
- Hameed, A. R., ul Islam, S., Raza, M., & Khattak, H. A. (2020). Towards Energy and Performance-aware Geographic Routing for IoT-enabled Sensor Networks. *Computers & Electrical Engineering*, 85, 106643. doi:10.1016/j.compeleceng.2020.106643
- Hao, K., Shen, H., Liu, Y., Wang, B., & Du, X. (2018). Integrating localization and energy-awareness: A novel geographic routing protocol for underwater wireless sensor networks. *Mobile Networks and Applications*, 23(5), 1427-1435.
- Harish, V. S. K. V., & Kumar, A. (2016). A review on modeling and simulation of building energy systems. *Renewable and sustainable energy reviews*, 56, 1272-1292.
- Hasan, M. Z., Al-Turjman, F., & Al-Rizzo, H. (2018). Analysis of cross-layer design of quality-of-service forward geographic wireless sensor network routing strategies in green internet of things. *IEEE Access*, 6, 20371-20389.
- Haseeb, K., Ud Din, I., Almogren, A., & Islam, N. (2020). An energy efficient and secure IoT-based WSN framework: An application to smart agriculture. *Sensors*, 20(7), 2081.

- Hassoune, K., Dachry, W., Moutaouakkil, F., & Medromi, H.: Dynamic Parking Guidance Architecture Using Ant Colony Optimization and Multi-agent Systems. *Journal of Advances in Information Technology*, Vol. 11(2), (2020).
- Heinzelman, W. B., Chandrakasan, A. P., & Balakrishnan, H. (2002). An application-specific protocol architecture for wireless microsensor networks. *IEEE Transactions on wireless communications*, 1(4), 660-670.
- Heinzelman, W. R., Chandrakasan, A., & Balakrishnan, H. (2000, January). Energy-efficient communication protocol for wireless microsensor networks. In *Proceedings of the 33rd annual Hawaii international conference on system sciences* (pp. 10-pp). IEEE.
- Herrera, M., Pérez-Hernández, M., Kumar Parlikad, A., Izquierdo, J. (2020). Multi-agent systems and complex networks: Review and applications in systems engineering. *Processes*, 8(3): 312. <https://doi.org/10.3390/pr8030312>
- Hettige, B., Karunananda, A. S., & Rzevski, G. (2021). MaSMT4: The AGR Organizational Model-Based Multi-Agent System Development Framework for Machine Translation. In *Inventive Computation and Information Technologies* (pp. 691-702). Springer, Singapore.
- Hilmani, A., Maizate, A., & Hassouni, L. (2018). Designing and managing a smart parking system using wireless sensor networks. *Journal of sensor and actuator networks*, 7(2), 24. <https://doi.org/10.3390/jsan7020024>
- <http://www.physiome.org/jsim/> Accessed on 8 Dec. 2020. [https://doi.org/10.1007/978-3-540-24620-6\\_15](https://doi.org/10.1007/978-3-540-24620-6_15)
- Huang, H., Yin, H., Min, G., Zhang, J., Wu, Y., & Zhang, X. (2017). Energy-aware dual-path geographic routing to bypass routing holes in wireless sensor networks. *IEEE Transactions on Mobile Computing*, 17(6), 1339-1352.
- Hui, Z., Lingli, Z., Yonghang, Y., & Linlin, C. (2022, February). A Survey of Multipath Load Balancing Based on Network Stochastic Model in MANET. In *2022 24th International Conference on Advanced Communication Technology (ICACT)* (pp. 336-341). IEEE.
- Javid, N., Qureshi, T. N., Khan, A. H., Iqbal, A., Akhtar, E., & Ishfaq, M. (2013). EDDEEC: Enhanced developed distributed energy-efficient clustering for heterogeneous wireless sensor networks. *Procedia computer science*, 19, 914-919.
- Jensen, K., Kristensen, L.M., Wells, L. (2007). Coloured Petri Nets and CPN Tools for modelling and validation of concurrent systems. *International Journal on Software Tools for Technology Transfer*, 9(3): 213-254. <https://doi.org/10.1007/s10009-007-0038-x>
- Jioudi, B., Amari, A., Moutaouakkil, F., & Medromi, H., e-parking: Multi-agent smart parking platform for dynamic pricing and reservation sharing service. *International Journal of Advanced Computer Science and Applications*, 10(11), 342-351, (2019).

- Johnson, D. B., Maltz, D. A., & Broch, J. (2001). DSR: The dynamic source routing protocol for multi-hop wireless ad hoc networks. *Ad hoc networking*, 5(1), 139-172.
- Juranic, J., Pavkovic, N., Naumann, T., Marjanovic, D. (2017). Modelling the design parameters dynamics with Petri nets. In DS 87-2 Proceedings of the 21st International Conference on Engineering Design (ICED 17) Vol 2: Design Processes, Design Organisation and Management, Vancouver, Canada, pp. 091-100.
- Kandris D, Nakas C, Vomvas D, Koulouras G. Applications of Wireless Sensor Networks: An Up-to-Date Survey. *Applied System Innovation*. 2020; 3(1):14. <https://doi.org/10.3390/asi3010014>
- Karp, B., & Kung, H. T. (2000, August). GPSR: Greedy perimeter stateless routing for wireless networks. In *Proceedings of the 6th annual international conference on Mobile computing and networking* (pp. 243-254).
- Karray, F., Jmal, M. W., Garcia-Ortiz, A., Abid, M., & Obeid, A. M. (2018). *A comprehensive survey on wireless sensor node hardware platforms*. *Computer Networks*, 144, 89–110. doi:10.1016/j.comnet.2018.05.010
- Ketshabetswe, L. K., Zungeru, A. M., Mangwala, M., Chuma, J. M., & Sigweni, B. (2019). Communication protocols for wireless sensor networks: A survey and comparison. *Heliyon*, 5(5), e01591.
- Khan, A., & Marriwala, N. (2021). A Literature Survey on LEACH Protocol and Its Descendants for Homogeneous and Heterogeneous Wireless Sensor Networks. In *Proceedings of the International Conference on Paradigms of Computing, Communication and Data Sciences* (pp. 631-643). Springer, Singapore.
- Khandelwal, A. (2018). A study on network simulator 2 ankit khandelwal. *International Journal of Current Research in Life Sciences*, 7(02), 1036-1039.
- Klee, H., & Allen, R. (2018). *Simulation of dynamic systems with MATLAB® and Simulink®*. Crc Press.
- Kleerekoper, A., & Filer, N. P. (2015). Perfect link routing for energy efficient forwarding in geographic routing. *Ad Hoc Networks*, 30, 46-62.
- Koh, K.; Kobayashi, T.; Lee, C. Investigation of piezoelectric driven MEMS mirrors based on single and double s-shaped PZT actuator for 2-D scanning applications. *Sens. Actuators A Phys.* **2012**, 184, 149–159. [[Google Scholar](#)] [[CrossRef](#)]
- Köhler, M., Moldt, D., & Rölke, H.; Modelling the structure and behaviour of Petri net agents. In *International Conference on Application and Theory of Petri Nets* (pp. 224-241). Springer, Berlin, Heidelberg. (2001, June). doi.org/10.1007/3-540-45740-2\_14
- Konolige, K., Nilsson, N.J. (1980). Multiple-agent planning systems. *Proc. AAAI 1980*, 80: 138-142.

- Koster, A., Koch, F. & Bazzan, A. L.: Incentivising crowd sourced parking solutions. In International Workshop on Citizen in Sensor Networks, Springer, Cham. (2013), September, pp. 36-43. doi:10.1007/978-3-319-04178-0\_4
- Kumar, A., Shwe, H. Y., Wong, K. J., & Chong, P. H. (2017). Location-based routing protocols for wireless sensor networks: A survey. *Wireless Sensor Network*, 9(1), 25-72..
- Kurniawan, A. (2018). *Practical Contiki-NG: Programming for Wireless Sensor Networks*. Apress.
- Kuruvila, J., Nayak, A., & Stojmenovic, I. (2006). Progress and location based localized power aware routing for ad hoc and sensor wireless networks. *International Journal of Distributed Sensor Networks*, 2(2), 147-159. DOI: 10.1109/71.969123
- Kuruvila, J., Nayak, A., & Stojmenovic, I. (2006). Progress and location based localized power aware routing for ad hoc and sensor wireless networks. *International Journal of Distributed Sensor Networks*, 2(2), 147-159.
- Lam, K. T., Stephen Leo, M., Lee, C., & Liang, G. (2011). Design evaluation of graphene nanoribbon nanoelectromechanical devices. *Journal of Applied Physics*, 110(2).
- Lee, C., Radhakrishnan, R., Chen, C. C., Li, J., Thillaigovindan, J., & Balasubramanian, N. (2008). Design and modeling of a nanomechanical sensor using silicon photonic crystals. *Journal of lightwave technology*, 26(7), 839-846.
- Lee, S., Bhattacharjee, B., Banerjee, S., & Han, B. (2010). A general framework for efficient geographic routing in wireless networks. *Computer Networks*, 54(5), 844-861.
- Lima, M. M., Oliveira, H. A., Guidoni, D. L., & Loureiro, A. A. (2017). Geographic routing and hole bypass using long range sinks for wireless sensor networks. *Ad Hoc Networks*, 67, 1-10. doi:10.1016/j.adhoc.2017.08.010
- Lindsey, S., & Raghavendra, C. S. (2002, March). PEGASIS: Power-efficient gathering in sensor information systems. In *Proceedings, IEEE aerospace conference* (Vol. 3, pp. 3-3). IEEE.
- Luo, J., Hu, J., Wu, D., & Li, R. (2014). Opportunistic routing algorithm for relay node selection in wireless sensor networks. *IEEE Transactions on Industrial Informatics*, 11(1), 112-121.
- Lyu, C., Zhang, X., Liu, Z., & Chi, C. H. (2019). Selective authentication based geographic opportunistic routing in wireless sensor networks for Internet of Things against DoS attacks. *IEEE Access*, 7, 31068-31082.
- Macal, C.M. (2020). Agent-based modeling and artificial life. *Complex Social and Behavioral Systems: Game Theory and Agent-Based Models*, 725-745. [https://doi.org/10.1007/978-1-0716-0368-0\\_7](https://doi.org/10.1007/978-1-0716-0368-0_7)

- Mahmood, I., Mobeen, M., Rahman, A.U., Younis, S., Malik, A.W., Fraz, M.M., Ullah, K. (2020). Modeling, simulation and forecasting of wind power plants using agent-based approach. *Journal of Cleaner Production*, 276: 124172. <https://doi.org/10.1016/j.jclepro.2020.124172>
- Maracke, C. (2019). Free and Open Source Software and FRAND- based patent licenses: How to mediate between Standard Essential Patent and Free and Open Source Software. *The Journal of World Intellectual Property*, 22(3-4), 78-102.
- Mehic, M., Fazio, P., & Voznak, M. (2019, October). Usability of destination-sequenced distance vector routing protocol routes. In *2019 11th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT)* (pp. 1-5). IEEE.. doi:10.1109/icumt48472.2019.8970752
- Mesa, B. J. F. (2021). *Exploration of Direct Synchronization Approaches for a High-Level and Unified Simulation of Discrete-Event/Continuous-Time Systems* (Doctoral dissertation, Université Grenoble Alpes).
- Mess, M., Guerrits, B. (2018). Multi-agent Systems. In: Zijm H., Klumpp M., Regattieri A., Heragu S. (eds) *Operations, Logistics and Supply Chain Management*. Lecture Notes in Logistics. Springer, Cham. [https://doi.org/10.1007/978-3-319-92447-2\\_27](https://doi.org/10.1007/978-3-319-92447-2_27)
- Milocco, R. H., Costantini, H., & Boumerdassi, S. (2014). Improved geographic routing in sensor networks subjected to localization errors. *Ad hoc networks*, 13, 476-486.
- Minakov, I., Passerone, R., Rizzardi, A., & Sicari, S. (2016). A comparative study of recent wireless sensor network simulators. *ACM Transactions on Sensor Networks (TOSN)*, 12(3), 1-39.
- Mohamed, R. E., Saleh, A. I., Abdelrazzak, M., & Samra, A. S. (2018). Survey on wireless sensor network applications and energy efficient routing protocols. *Wireless Personal Communications*, 101(2), 1019-1055. <https://doi.org/10.1007/s11277-018-5747-9>
- Mondal, M., & Rehena, Z. (2021). An IoT-based congestion control framework for intelligent traffic management system. In *Advances in Artificial Intelligence and Data Engineering* (pp. 1287-1297). Springer, Singapore.
- Moorthy, R., Bangera, V., Amrin, Z., Avinash, N. J., & NS, K. R. (2020, October). WSN in defence field: A security overview. In *2020 Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC)* (pp. 258-264). IEEE.
- Nandal, V., & Dahiya, S. (2021). IoT Based Energy-Efficient Data Aggregation Wireless Sensor Network in Agriculture: A Review. *PSYCHOLOGY AND EDUCATION*, 58(1), 2985-3007.
- Nanna, G.A., Quatraro, N.F., De Carolis, B. (2020). A multi-agent system for simulating the spread of a contagious disease. *Session 5: Agents & Actors for Data Science* 160, 1613: 119.

- Nazareth, P., & Chandavarkar, B. R. (2020). Void avoidance node deployment strategy for underwater sensor networks. In *Smart Systems and IoT: Innovations in Computing* (pp. 493-502). Springer, Singapore.
- Nguyen, K. V., Nguyen, C. H., Le Nguyen, P., Van Do, T., & Chlamtac, I. (2021). Energy-efficient routing in the proximity of a complicated hole in wireless sensor networks. *Wireless Networks*, 27(4), 3073-3089. <https://doi.org/10.1007/s11276-021-02569-3>
- Obaidat, M. S., & Misra, S. (2014). *Principles of wireless sensor networks*. Cambridge University Press. <https://doi.org/10.1017/CBO9781139030960>
- Okoso, A., Otaki, K., & Nishi, T., Multi-agent path finding with priority for cooperative automated valet parking. In 2019 IEEE Intelligent Transportation Systems Conference (ITSC) (pp. 2135-2140). IEEE, (2019, October).
- Onggo, B. S., Mustafee, N., Smart, A., Juan, A. A., & Molloy, O. (2018, December). Symbiotic simulation system: Hybrid systems model meets big data analytics. In *2018 Winter Simulation Conference (WSC)* (pp. 1358-1369). IEEE.
- Padilla, J. J., Diallo, S. Y., Lynch, C. J., & Gore, R. (2018). Observations on the practice and profession of modeling and simulation: A survey approach. *Simulation*, 94(6), 493-506.
- Pal, D., Funilkul, S., Charoenkitkarn, N., & Kanthamanon, P. (2018). Internet-of-things and smart homes for elderly healthcare: An end user perspective. *IEEE Access*, 6, 10483-10496.
- Patel, B., & Shah, P. (2021). Operating system support, protocol stack with key concerns and testbed facilities for IoT: A case study perspective. *Journal of King Saud University-Computer and Information Sciences*.
- Patel, R. L., Pathak, M. J., & Nayak, A. J. (2018). Survey on network simulators. *International Journal of Computer Applications*, 182(21).
- Patil, A., & Biradar, R. V. (2016, October). Scheduling techniques for TinyOS: A review. In *2016 International Conference on Computation System and Information Technology for Sustainable Solutions (CSITSS)* (pp. 188-193). IEEE.
- Peng, B., & Kemp, A. H. (2011). Energy-efficient geographic routing in the presence of localization errors. *Computer Networks*, 55(3), 856-872.
- Pereyra, M., Schniter, P., Chouzenoux, E., Pesquet, J. C., Tourneret, J. Y., Hero, A. O., & McLaughlin, S. (2015). A survey of stochastic simulation and optimization methods in signal processing. *IEEE Journal of Selected Topics in Signal Processing*, 10(2), 224-241.
- Perkins, C. E., & Royer, E. M. (1999, February). Ad-hoc on-demand distance vector routing. In *Proceedings WMCSA'99. Second IEEE Workshop on Mobile Computing Systems and Applications* (pp. 90-100). IEEE.

- Por, E., van Kooten, M., & Sarkovic, V. (2019). Nyquist–Shannon sampling theorem. *Leiden University*, 1, 1.
- Prabha, K. (2020). Performance assessment and comparison of efficient ad hoc reactive and proactive network routing protocols. *SN Computer Science*, 1(1), 1-7.
- Preethi, A., Pravin, E., & Sangeetha, D. (2016, April). Modified balanced energy efficient network integrated super heterogeneous protocol. In *2016 International Conference on Recent Trends in Information Technology (ICRTIT)* (pp. 1-6). IEEE.
- Prim, R. C. (1957). Shortest connection networks and some generalizations. *The Bell System Technical Journal*, 36(6), 1389-1401.
- Pruthi, V., Mittal, K., Sharma, N., & Kaushik, I. (2019). *Network Layers Threats & its Countermeasures in WSNs. 2019 International Conference on Computing, Communication, and Intelligent Systems. (ICCCIS)*. Doi:10.1109/icccis48478.2019.8974523
- Qing, L., Zhu, Q., & Wang, M. (2006). Design of a distributed energy-efficient clustering algorithm for heterogeneous wireless sensor networks. *Computer communications*, 29(12), 2230-2237.
- Qureshi, T. N., Javaid, N., Khan, A. H., Iqbal, A., Akhtar, E., & Ishfaq, M. (2013). BEENISH: Balanced energy efficient network integrated super heterogeneous protocol for wireless sensor networks. *Procedia Computer Science*, 19, 920-925.
- Raja, L. (2018). Study of Various Network Simulators. *International Research Journal of Engineering and Technology (IRJET)*, 5(12).
- Ramaprasad, S. S., Kumar, B. S., Lebaka, S., Prasad, P. R., Kumar, K. S., & Manohar, G. N. (2019, May). Intelligent Crop Monitoring and Protection System in Agricultural fields Using IoT. In *2019 4th International Conference on Recent Trends on Electronics, Information, Communication & Technology (RTEICT)* (pp. 1527-1531). IEEE.
- Ranjan, A., Sahu, H. B., & Misra, P. (2015). A survey report on operating systems for tiny networked sensors. *arXiv preprint arXiv:1505.05269*.
- Rashid, Z. N., Zebari, S. R., Sharif, K. H., & Jacksi, K. (2018, October). Distributed cloud computing and distributed parallel computing: A review. In *2018 International Conference on Advanced Science and Engineering (ICOASE)* (pp. 167-172). IEEE.
- Redjimi, K., & Redjimi, M. (2022). The DEEC and EDEEC Heterogeneous WSN Routing Protocols. *International Journal of Advanced Networking and Applications*, 13(4), 5045-5051.
- Redjimi, K., Boulaiche, M., & Redjimi, M. (2021). IEGGR: Improved Energy-aware and Delivery Guarantee Geographic Routing. *Journal of Communications Software and Systems*, 17(2), 143-153.

Redjimi, K., Boulaiche, M., & Redjimi, M. (2021). Agent Based Modeling and Simulation for Geographic Routing Protocol in the Wireless Sensor Networks. In *9th (Online) International Conference on Applied Analysis and Mathematical Modeling (ICAAMM21) June 11-13, 2021, Istanbul-Turkey* (p. 78).

Redjimi, K., Boulaiche, M., Redjimi, M. (2022). DEEC and EDEEC Routing Protocols for Heterogeneous Wireless Sensor Networks: A Brief Comparative Study. In: *Troiano, L., Vaccaro, A., Kesswani, N., Díaz Rodriguez, I., Brigui, I. (eds) Progresses in Artificial Intelligence & Robotics: Algorithms & Applications. ICDLAIR 2021. Lecture Notes in Networks and Systems, vol 441*. Springer, Cham. [https://doi.org/10.1007/978-3-030-98531-8\\_12](https://doi.org/10.1007/978-3-030-98531-8_12)

Redjimi, K., Redjimi, M., & Boulaiche, M. (2021, January). Improved Geographic Routing Protocol for Wireless Sensor Networks. In *International Conference on Digital Technologies and Applications* (pp. 1135-1145). Springer, Cham.

Rida, N., Ouadoud, M., & Hasbi, A. (2020). Traffic Signal Control for a Single Intersection-Based Intelligent Transportation System. In *Digital Transformation and Innovative Services for Business and Learning* (pp. 159-180). IGI Global. doi: 10.4018/978-1-7998-5175-2.ch009

Rodoplu, V., & Meng, T. H. (1999). Minimum energy mobile wireless networks. *IEEE Journal on selected areas in communications*, 17(8), 1333-1344.

Russell, S., Norvig, P. (2010). *Artificial Intelligence: A Modern Approach*, Pearson.

Sahfutri, A., Husni, N. L., Nawawi, M., Lutfi, I., Silvia, A., & Prihatini, E. (2018, October). Smart parking using wireless sensor network system. In *2018 International Conference on Electrical Engineering and Computer Science (ICECOS)* (pp. 117-122). IEEE.

Saini, P., & Sharma, A. K. (2010, October). E-DEEC-enhanced distributed energy efficient clustering scheme for heterogeneous WSN. In *2010 First international conference on parallel, distributed and grid computing (PDGC 2010)* (pp. 205-210). IEEE.

Sakurada, L., Barbosa, J., Leitão, P., Alves, G., Borges, A. P., & Botelho, P. (2019, October). Development of agent-based cps for smart parking systems. In *IECON 2019-45th Annual Conference of the IEEE Industrial Electronics Society* (Vol. 1, pp. 2964-2969). IEEE.

Samridhi, S., & Liscano, R. (2020, October). Performance comparison of a Software Defined and Wireless Sensor Network. In *2020 International Symposium on Networks, Computers and Communications (ISNCC)* (pp. 1-6). IEEE.

Savaglio, C., Ganzha, M., Paprzycki, M., Bădică, C., Ivanović, M., Fortino, G. (2020). Agent-based Internet of Things: State-of-the-art and research challenges. *Future Generation Computer Systems*, 102: 1038-1053. <https://doi.org/10.1016/j.future.2019.09.016>

Schweiger, G., Nilsson, H., Schoeggl, J., Birk, W., & Posch, A. (2020). Modeling and simulation of large-scale systems: A systematic comparison of modeling paradigms. *Applied Mathematics and Computation*, 365, 124713.

- Seddari, N., Boukelkoul, S., Bouras, A., Belaoued, M., Redjimi, M. (2021). A new transformation approach for complex systems modelling and simulation: Application to industrial control system. *International Journal of Simulation and Process Modelling*, 16(1): 34-48. <https://doi.org/10.1504/IJSPM.2021.113073>
- Seddari, N., Redjimi, M., Boukelkoul, S. (2014). Using of DEVS and MAS tools for modeling and simulation of an industrial steam generator. *Journal of Computing and Information Technology*, 22(3): 171-189. <https://doi.org/10.2498/cit.1002348>
- Sehili, S. (2018). *Discrete event modeling and simulation for IoT efficient design* (Doctoral dissertation, Corte).
- Serpanos, D. (2018). The cyber-physical systems revolution. *Computer*, 51(3), 70-73.
- Sesli, E., & Hacıoğlu, G. (2017). Contiki OS usage in wireless sensor networks (WSNs). *Turkish Journal of Electromechanics & Energy*, 2(2), 1-6.
- Shannon, C. E. (1984). Communication in the presence of noise. *Proceedings of the IEEE*, 72(9), 1192-1201.
- Sharma, S., Mahajan, A. N., & Poonia, R. C. (2019, February). An Inclusive survey of Network Simulators. In *Proceedings of International Conference on Sustainable Computing in Science, Technology and Management (SUSCOM)*, Amity University Rajasthan, Jaipur-India.
- Shivanka, P. C., Indu, S., Joshi, A., Singh, A., & Rajpal, R. (2016, March). Performance of static sink in wireless sensor networks when implementing geographic routing. In *2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom)* (pp. 3394-3398). IEEE.
- Silva, M. (2018). On the history of discrete event systems. *Annual Reviews in Control*, 45, 213-222.
- Singh, J., Kaur, R., & Singh, D. (2020). A survey and taxonomy on energy management schemes in wireless sensor networks. *Journal of Systems Architecture*, 111, 101782. doi:10.1016/j.sysarc.2020.101782
- Singh, M., & Kumar, S. (2017). A survey: Ad-hoc on demand distance vector (AODV) protocol. *International Journal of Computer Applications*, 161(1), 38-44.
- Singh, S. K., Kumar, P., & Singh, J. P. (2017). A survey on successors of LEACH protocol. *Ieee Access*, 5, 4298-4328.
- Slimani, A., Redjimi, M., & Slimani, D. (2017). Weighted Density Center (WDC-LEACH-C) Clustering Protocol for Wireless Sensor Networks (WSN). *Informatica*, 42(2).
- Slimani, A., Redjimi, M., & Slimani, D. (2020). An agent-based cluster head management strategy for hierarchical wireless sensor network. *International Journal of Systems, Control and Communications*, 11(4), 321-333.

- Souidi, M.E.H., Songhao, P., Guo, L., Lin, C. (2016). Multi-agent cooperation pursuit based on an extension of AALAADIN organisational model. *Journal of Experimental & Theoretical Artificial Intelligence*, 28(6): 1075-1088. <https://doi.org/10.1080/0952813X.2015.1056241>
- Stojmenovic, I., & Lin, X. (2001). Power-aware localized routing in wireless networks. *IEEE transactions on parallel and distributed systems*, 12(11), 1122-1133.
- Sun, Y., Dong, Y., Gao, R., Chu, Y., Zhang, M., Qian, X., & Wang, X. (2018). Wearable pulse wave monitoring system based on MEMS sensors. *Micromachines*, 9(2), 90.
- Sunitha, E.V., Samuel, P. (2019). Automatic code generation from UML state chart diagrams. *IEEE Access*, 7: 8591-8608. <https://doi.org/10.1109/ACCESS.2018.2890791>
- Tao, S., Ananda, A. L., & Chan, M. C. (2010). Greedy face routing with face identification support in wireless networks. *Computer Networks*, 54(18), 3431-3448.
- Tokala, M., & Nallamekala, R. (2018, April). Secured algorithm for routing the military field data using dynamic sink: Wsn. In *2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT)* (pp. 471-476). IEEE.
- Tomar, P., Kaur, G., & Singh, P. (2018). A prototype of IoT-based real time smart street parking system for smart cities. In *Internet of Things and Big Data Analytics Toward Next-Generation Intelligence* (pp. 243-263). Springer, Cham. [https://doi.org/10.1007/978-3-319-60435-0\\_10](https://doi.org/10.1007/978-3-319-60435-0_10)
- Toor, A. S., & Jain, A. K. (2017). A survey on wireless network simulators. *Bulletin of Electrical Engineering and Informatics*, 6(1), 62-69.
- Tyagi, S., & Kumar, N. (2013). A systematic review on clustering and routing techniques based upon LEACH protocol for wireless sensor networks. *Journal of Network and Computer Applications*, 36(2), 623-645.
- Valk, R. (1998, June). Petri nets as token objects. In *International Conference on Application and Theory of Petri Nets* (pp. 1-24). Springer, Berlin, Heidelberg.. [doi.org/10.1007/3-540-69108-1\\_1](https://doi.org/10.1007/3-540-69108-1_1).
- Varga, A. (2019). A practical introduction to the OMNeT++ simulation framework. In *Recent advances in network simulation* (pp. 3-51). Springer, Cham.
- Vilela, P. H., Rodrigues, J. J., Righi, R. D. R., Kozlov, S., & Rodrigues, V. F. (2020). Looking at fog computing for e-health through the lens of deployment challenges and applications. *Sensors*, 20(9), 2553.
- Virdis, A., & Kirsche, M. (2019). Recent Advances in Network Simulation. *EAI/Springer Innovations in Communication and Computing*.

- Von Mulert, J., Welch, I., & Seah, W. K. (2012). Security threats and solutions in MANETs: A case study using AODV and SAODV. *Journal of network and computer applications*, 35(4), 1249-1259.
- Wainer, G. A., & Mosterman, P. J. (Eds.). (2018). *Discrete-event modeling and simulation: theory and applications*. CRC press.
- Wang, B., Wang, J., Zhang, B., Li, X. (2016). Global cooperative control framework for multiagent systems subject to actuator saturation with industrial applications. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 47(7): 1270-1283. <https://doi.org/10.1109/TSMC.2016.2573584>
- Wang, B., Wang, J., Zhang, B., Li, X. (2016). Global cooperative control framework for multiagent systems
- Weiss, G. (1999). *Multiagent systems: a modern approach to distributed artificial intelligence*. MIT Press.
- Yan, J., Zhou, M., & Ding, Z. (2016). Recent advances in energy-efficient routing protocols for wireless sensor networks: A review. *IEEE Access*, 4, 5673-5686.
- Yang, Y. (2020). Research on modeling and simulation of agent-based intelligent teaching system. *Science Journal of Education*, 8(1): 27-31. <https://doi.org/10.11648/j.sjedu.20200801.15>
- Yaqoob, A., Ashraf, M. A., Ferooz, F., Butt, A. H., & Khan, Y. D. (2019, November). WSN Operating Systems for Internet of Things (IoT): A Survey. In *2019 International Conference on Innovative Computing (ICIC)* (pp. 1-7). IEEE.
- Yıldırım, G., & Tatar, Y. (2018). Simplified agent-based resource sharing approach for WSN-WSN interaction in IoT/CPS projects. *IEEE Access*, 6, 78077-78091.
- Younis, O., & Fahmy, S. (2004). HEED: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks. *IEEE Transactions on mobile computing*, 3(4), 366-379.
- Zeigler, B.P., Muzy, A., Kofman, E. (2019). Introduction to iterative system specification. *Theory of Modeling and Simulation*, 255-280. <https://doi.org/10.1016/b978-0-12-813370-5.00019-5>
- Zhou, Z., Chan, W.K.V., Chow, J.H. (2007). Agent-based simulation of electricity markets: A survey of tools. *Artificial Intelligence Review*, 28(4): 305-342. <https://doi.org/10.1007/s10462-009-9105-x>
- Zhu, X., Ding, B., Li, W., Gu, L., & Yang, Y. (2018). On development of security monitoring system via wireless sensing network. *EURASIP Journal on Wireless Communications and Networking*, 2018(1), 1-10. <https://doi.org/10.1186/s13638-018-1235-x>

Zulfiqar, H., Ul Haque, H. M., Tariq, F., & Khan, R. M. (2021). A survey on smart parking systems in urban cities. *Concurrency and Computation: Practice and Experience*, e6511.. doi:10.1002/cpe.6511