

*N° d'ordre :*

REPUBLIQUE ALGERIENNE DEMOCRATIQUE & POPULAIRE  
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR & DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITE DJILLALI LIABES  
FACULTE DES SCIENCES EXACTES  
SIDI BEL ABBES

## **THESE DE DOCTORAT**

*Présentée par*

*MEKKAOUI Kheireddine*

*Spécialité : Informatique  
Option : Intelligence artificielle*

*Intitulé*

**SENSOR TECHNOLOGY :  
Un clustering centralisé et dynamique basé sur les AGs  
pour une consommation d'énergie minimale dans les  
réseaux de capteurs sans fil**

*Soutenue le : 07 Janvier 2016*

*Devant le jury composé de :*

*Président : Pr. El-Berrichi Zakaria - Professeur à l'université de SBA*

*Examineurs : Dr. Abdelmalek Amine - MC-A à l'université de Saida*

*Dr. Hamou Reda Mohamed - MC-A à l'université de Saida*

*Dr. Boukfi Hacène Sofiane - MC-A à l'université de SBA*

*Directeur de thèse : Pr. Rahmoun Abdellatif - Professeur à l'université de SBA*

UNIVERSITÉ DJILLAI LIABES - SIDI BEL-ABBES  
FACULTE DES SCIENCES EXACTES  
DEPARTEMENT D'INFORMATIQUE

# T H È S E

pour obtenir le titre de

**Docteur en Sciences**

de l'Université de Djillali Labès - Sidi Bel-Abbès

**Option : INFORMATIQUE ARTIFICIELLE**

Présentée et soutenue par

MEKKAOUI Kheireddine

## SENSOR TECHNOLOGY : UN CLUSTERING CENTRALISÉ ET DYNAMIQUE BASÉ SUR LES AGS POUR UNE CONSOMMATION D'ÉNERGIE MINIMALE DANS LES RÉSEAUX DE CAPTEURS SANS FIL

Thèse dirigée par Pr. RAHMOUN Abdellatif

**Jury :**

<i>Rapporteurs :</i>	PR. RAHMOUN Abdellatif	-	Professeur à l'université de SBA
<i>Président :</i>	PR. EL-BERRICHI Zakaria	-	Professeur à l'université de SBA
<i>Examineurs :</i>	DR. ABDELMALEK Amine	-	MC-A à l'université de Saida
	DR. HAMOU Reda Mohamed	-	MC-A à l'université de Saida
	DR. BOUKLI Hacène Sofiane	-	MC-A à l'université de SBA



## REMERCIEMENTS

Je souhaite remercier, profondément, en premier lieu mon directeur de thèse, Pr. RAHMOUN Abdellatif, professeur à l'école supérieure d'informatique de l'université de Sidi Bel-Abbès pour m'avoir fait l'honneur d'accepter de m'encadrer durant la réalisation de cette thèse. Je suis, également, très reconnaissant pour le temps qu'il m'a accordé, son précieux aide, ses remarques pertinentes et ses qualités pédagogiques et scientifiques, sa franchise, sa générosité et sa sympathie. J'ai beaucoup appris à ses côtés.

Je tiens à remercier Pr. El-Berrichi Zakaria d'avoir accepté d'être président du jury. Je remercie, également, tous les membres du jury, Dr Abdelmalek Amine, Dr. Hamou Reda et DR. Boukli Hacène Sofiane, d'avoir accepté d'assister à la présentation de ce travail, je les remercie, également, pour l'intérêt qu'ils ont montré pour ma thèse ainsi que pour leurs temps qu'ils ont consacré pour la lire.

Je voudrais remercier aussi le professeur Gianluigi Ferrari, Professeur à l'université de Parme en Italie, pour son aide et pour l'intérêt qu'il a porté à mon travail.

Je tiens à remercier tous les membres de ma famille qui m'ont apporté le soutien et le confort, qui m'ont accompagné durant ce parcours.

Ce travail n'a pu atteindre ses objectifs sans la contribution de près ou de loin de plusieurs personnes auxquelles j'adresse mes chaleureux remerciements.

**K. Mekkaoui**

## **abstract**

In order to minimize the energy consumption involved by communications in wireless sensor networks, two main contributions are presented in this thesis. The first one, focuses on the analysis of hop lengths, Indeed the network lifetime depends strongly on how the data are communicated in the network by using either short-hops or long-hops. The simulations provided show that using an ideal length hops reduces the energy consumption. In the second contribution, the clustering is investigated. The use of clustering has proven to be overwhelming. The problem consists of determining the number of cluster-heads and their distribution in the network to ensure minimal energy consumption and better networks coverage. The Low-Energy Adaptive Clustering Hierarchy algorithm fixes in advance the number of cluster-heads, thus this does not guarantee the coverage of the entire network.

In this thesis, we propose a genetic centralized dynamic algorithm (GA)-based clustering approach to optimize the clustering configuration (the number of cluster-heads, their distribution and the cluster-members) to limit node energy consumption and the best coverage.

Experiments have been conducted on well-known WSN data benchmark to test the performance of the proposed methodology. The obtained simulation results show that the proposed technique overcomes the Low-Energy Adaptive Clustering Hierarchy algorithm.

### **Keywords :**

Wireless sensor networks, Clustering, energy efficiency, network lifetime, genetic algorithms.

# Table des matières

<b>I</b>	<b>Introduction sur le domaine de recherche</b>	<b>5</b>
<b>1</b>	<b>les réseaux de capteurs sans fil</b>	<b>7</b>
1.1	Introduction . . . . .	7
1.2	Les réseaux Ad-hoc . . . . .	7
1.2.1	Caractéristiques des réseaux Ad-hoc . . . . .	8
1.3	Les réseaux de capteurs sans fil (RCSFs) . . . . .	8
1.4	Architecture d'un noeud capteur . . . . .	10
1.4.1	Micro-contrôleur . . . . .	11
1.4.2	Convertisseur Analogique/Numérique (CAN) . . . . .	13
1.4.3	Horloge temps réel . . . . .	13
1.4.4	Module de communication sans fil . . . . .	14
1.4.5	Système de localisation . . . . .	14
1.4.6	Capteurs/actionneurs . . . . .	15
1.4.7	Mémoires . . . . .	15
1.4.8	L'unité de contrôle d'énergie . . . . .	15
1.4.9	Batterie / système de récupération d'énergie . . . . .	16
1.5	Les communications dans les RCSFs . . . . .	16
1.6	Architecture d'un réseau de capteurs sans fil . . . . .	21
1.6.1	Les types d'architecture des RCSFs . . . . .	22
1.7	Similitudes et différences entre les réseaux MANETs et les RCSFs . . . . .	23
1.7.1	Similitudes entre les réseaux MANETs et les RCSFs . . . . .	23
1.7.2	Différences entre les réseaux MANETs et les RCSFs . . . . .	24
1.8	Conclusion . . . . .	26

---

<b>2 Domaines d'application des RCSFs</b>	<b>27</b>
2.1 Introduction . . . . .	27
2.2 Applications militaires . . . . .	27
2.3 Applications environnementales . . . . .	28
2.4 Applications de santé . . . . .	29
2.5 Applications domotique . . . . .	31
2.6 Applications industrielles . . . . .	32
2.7 Applications commerciales . . . . .	33
2.8 Conclusion . . . . .	34
<b>3 Défis et challenges dans les RCSFs</b>	<b>35</b>
3.1 Introduction . . . . .	35
3.2 Contraintes matérielles . . . . .	35
3.3 Tolérance aux pannes . . . . .	36
3.4 évolutivité . . . . .	37
3.5 Coût de production . . . . .	38
3.6 La topologie utilisée . . . . .	38
3.6.1 La phase de pré-déploiement et de déploiement . . . . .	39
3.6.2 La phase après déploiement . . . . .	39
3.6.3 Phase de re-déploiement des noeuds supplémentaires . . . . .	40
3.7 Supports de transmission . . . . .	40
3.8 Consommation d'énergie . . . . .	41
3.8.1 La capture (Sensing) . . . . .	43
3.8.2 Traitement des données . . . . .	43
3.8.3 Communication . . . . .	43
3.9 Conclusion . . . . .	45
<b>4 Le facteur énergie dans les RCSFs</b>	<b>47</b>

---

4.1	Introduction . . . . .	47
4.2	La consommation d'énergie par un noeud capteur . . . . .	48
4.2.1	Énergie consommée suite à la collecte des données . . . . .	48
4.2.2	Énergie consommée suite au traitement des données . . . . .	48
4.2.3	Énergie consommée suite à la transmission des données . . . . .	49
4.3	Les sources de gaspillage d'énergie des les RCSFs . . . . .	49
4.3.1	L'écoute passive (idle listening) . . . . .	49
4.3.2	Les collisions . . . . .	50
4.3.3	La puissance de transmission . . . . .	50
4.3.4	Les distances de transmission . . . . .	50
4.3.5	L'écoute abusive (Overhearing) . . . . .	51
4.3.6	Le sur-coût des paquets de contrôle (Overhead) . . . . .	51
4.4	Protocoles d'économie d'énergie dans les RCSFs . . . . .	51
4.4.1	Protocoles dédiés à la couche réseau . . . . .	52
4.4.2	L'approche à chaîne (Chain-based approach) . . . . .	60
4.4.3	Protocoles dédiés à la couche liaison de données (sous couche MAC) . . . . .	62
4.5	Conclusion . . . . .	64
<b>5</b>	<b>Le clustering dans les RCSFs</b> . . . . .	<b>65</b>
5.1	Introduction . . . . .	65
5.2	Les objectifs de conception de clusters dans les RCSFs . . . . .	68
5.3	Les défis de conception de clusters dans les RCSFs . . . . .	69
5.4	Classification des algorithmes de clustering . . . . .	71
5.4.1	Paramètres de clustering . . . . .	71
5.5	Taxonomie des protocoles de clustering . . . . .	73
5.5.1	Les algorithmes de clustering pour les réseaux homogènes ou hétérogènes . . . . .	73

---

5.5.2	Les algorithmes de clustering centralisés ou distribués . . . . .	74
5.5.3	Les algorithmes de clustering statiques ou dynamiques . . . . .	75
5.5.4	Les algorithmes de clustering probabilistes ou non probabilistes	75
5.6	Les systèmes immunitaires artificiels pour les RCSFs . . . . .	86
5.6.1	Les AIS pour les RCSFs . . . . .	86
5.7	Conclusion . . . . .	87
<b>II Contributions dans le domaine</b>		<b>89</b>
<b>6 Analyse énergétique : Longs-sauts Vs Courts-Sauts</b>		<b>91</b>
6.1	Résumé . . . . .	91
6.2	Introduction . . . . .	91
6.3	Travaux connexes . . . . .	94
6.4	Modèle de propagation utilisé . . . . .	95
6.5	Analyse des courts-sauts Vs Long-sauts . . . . .	97
6.6	Comparaison entre les courts-sauts et les longs-sauts . . . . .	99
6.7	Simulations et validation des résultats . . . . .	100
6.8	Conclusion . . . . .	102
<b>7 Un clustering génétique, centralisé et dynamique pour les RCSFs</b>		<b>105</b>
7.1	Résumé . . . . .	105
7.2	Introduction . . . . .	105
7.3	Optimisation multi-objectifs . . . . .	107
7.3.1	Formulation . . . . .	109
7.3.2	Notion de dominance . . . . .	110
7.4	Source de l'efficacité des méta-heuristiques . . . . .	111
7.5	L'historique de clustering . . . . .	112
7.6	Le model réseau utilisé . . . . .	113

---

7.7	L'approche proposée . . . . .	114
7.7.1	La présentation du problème . . . . .	114
7.7.2	L'algorithme proposé . . . . .	115
7.8	L'analyse des performances . . . . .	118
7.9	Conclusion . . . . .	126
	<b>Conclusion générale</b>	<b>129</b>
	<b>Bibliographie</b>	<b>133</b>



# Table des figures

1.1	Exemple d'un réseau de capteur sans fil . . . . .	9
1.2	Exemple de capteurs . . . . .	10
1.3	Architecture de base d'un noeud capteur . . . . .	11
1.4	Architecture matérielle d'un noeud capteur . . . . .	11
1.5	La pile protocolaire dans les RCSFs . . . . .	17
1.6	Exemple de RCSF. . . . .	21
1.7	Architecture Plat des RCSFs. . . . .	22
1.8	Architecture hierarchique des RCSFs. . . . .	23
2.1	Exemple d'utilisations d'un RCSF dans les applications militaires . . . . .	28
2.2	Exemple d'utilisations des RCSFs dans la detection des feux de forêts . . . . .	29
2.3	Exemple d'utilisations des RCSFs dans l'arrosage goûte à goûte . . . . .	30
2.4	Exemple de capteurs bio-médicaux . . . . .	30
2.5	Exemple d'application de santé : <i>CodeBlue</i> . . . . .	31
2.6	Exemple d'application domotique . . . . .	32
2.7	Exemple d'applications industrielles . . . . .	33
3.1	Architecture matérielle générale d'un noeud de capteur . . . . .	36
3.2	La répartition de la consommation d'énergie d'un noeud MicaZ . . . . .	42
4.1	Protocoles de routage dans les RCSFs . . . . .	53
4.2	Le routage data-centric . . . . .	54

---

4.3	Fonctionnement du protocole SPIN. Un noeud A commence par annoncer ses données vers le noeud B (a). Noeud B répond en envoyant une demande au noeud A (b). Après avoir reçu les données demandées (c), le noeud B envoie ensuite des annonces pour ses voisins (d), qui à leur tour envoient des requêtes vers B (e-f). . . . .	56
4.4	Le routage hiérarchique . . . . .	58
4.5	Construction de chaînes en utilisant l'algorithme avide . . . . .	61
4.6	le mécanisme du Duty-cycling . . . . .	64
5.1	La communication de données dans un réseau formé en cluster . . . . .	67
5.2	Le clustering à deux niveaux . . . . .	67
5.3	Algorithme de routage LEACH . . . . .	77
5.4	Processus de formation des clusters par LEACH . . . . .	80
6.1	La distance de transmission pour un saut . . . . .	96
6.2	Transmission avec : (1)n-sauts et (2) m-sauts . . . . .	97
6.3	Mica2 : La consommation d'énergie Vs longueur de sauts . . . . .	102
6.4	Mica2dot : La consommation d'énergie Vs longueur de sauts . . . . .	104
7.1	Exemple de RCSF structuré en 3 clusters . . . . .	106
7.2	Allure d'une fonction objectif d'un problème d'optimisation difficile .	111
7.3	Module de transmission/réception . . . . .	115
7.4	Réseau de capteurs sans fil avec 100 noeuds . . . . .	119
7.5	Réseau de capteurs sans fil avec 1000 noeuds . . . . .	120
7.6	Un RCSF de 100 noeuds après l'application de LEACH . . . . .	121
7.7	Un RCSF de 100 noeuds après l'application de GCDC . . . . .	121
7.8	L'énergie résiduelle du réseau en fonction de temps de simulation (en termes de nombre d'évènements). Le nombre de noeuds dans le réseau est : (a)100. . . . .	122

---

7.9	L'énergie résiduelle du réseau en fonction de temps de simulation (en termes de nombre d'évènements). Le nombre de noeuds dans le réseau est : (b)1000. . . . .	123
7.10	(a)-L'évolution de la connectivité du réseau , en termes de $NNN$ , en fonction du temps de simulation (en termes de nombre des évènements). Le nombre de noeuds dans le RCSF est 100 noeuds . . . . .	124
7.11	(b)-L'évolution de la connectivité du réseau , en termes des noeuds morts, en fonction du temps de simulation (en termes du nombre d'évènements). Le nombre de noeuds dans le RCSF est 100 noeuds . . . . .	125
7.12	(a)-L'évolution de la connectivité du réseau , en termes de $NNN$ , en fonction du temps de simulation (en termes de nombre d'évènement). Le nombre des noeuds dans le RCSF est 1000 noeuds . . . . .	126
7.13	(b)-L'évolution de la connectivité du réseau , en termes des noeuds morts, en fonction du temps de simulation (en termes de nombre d'évènement). Le nombre des noeuds dans le RCSF est 1000 noeuds . . . . .	127



# Liste des tableaux

6.1	Valeurs typiques de l'exposant de perte du signal . . . . .	96
6.2	Caractéristiques expérimentales des capteurs <i>Mica</i> et <i>Mica2dot</i> . . . .	100
6.3	Consommation d'énergie Vs longueur de sauts . . . . .	101
6.4	Consommation d'énergie Vs longueur de sauts . . . . .	103



# Introduction générale

La première décennie du 21<sup>ème</sup> siècle a été appelée, par certains chercheurs, la *décennie des capteurs*. Ceci est due au nombre important des domaines utilisant les capteurs. En effet, les capteurs sont entrain de déclencher une révolution semblable à celle vécue par les micro-ordinateurs dans les années 1980. Dans nos jours, les besoins de surveiller l'environnement et de détecter les évènements sont de plus en plus élevés, ceci a conduit aux énormes progrès dans la technologie des capteurs. Cependant, il reste de grands défis à soulever dans ce domaine.

Le besoin de collecter les informations depuis l'environnement (captage d'informations), l'évolution rapide de la micro-électronique et les technologies sans fils, ont permis l'apparition de petits dispositifs électroniques avec un coût très réduit et des ressources très limitées, capables de collecter et de traiter l'information d'une manière autonome et flexible. Ces appareils peuvent être déployés et interconnectés à grande échelle, donnant naissance, à un nouveau type de réseaux appelé Réseaux de Capteurs Sans Fil (*RCSF*). Le développement des RCSFs était originalement motivé par les applications militaires (surveillance des champs de bataille, localisation de l'ennemie, etc). Néanmoins, leurs performances remarquables en termes de fiabilité et le faible coût ont permis de propager leur utilisation dans d'autres domaines d'application civil (exemple : la surveillance de l'environnement, l'industrie, la domotique, la santé, etc).

Un réseau de capteurs sans fil est constitué, généralement, de milliers voir de millions de capteurs sans fil, ces capteurs sont conçus pour fonctionner en groupe, d'une façon autonome, en utilisant des sauts multiples (*Multi-hops*) afin de transmettre les données collectées à un point central appelé la station de base (*Sink*).

Chaque noeud capteur est équipé d'un micro-processeur d'une faible puissance de calcul, d'une petite batterie, d'une antenne radio et d'un ou de plusieurs capteurs

(exemple : température, humidité, pression, etc). Ainsi, les RCSFs doivent opérer en prenant toujours en compte leurs limitations de ressources. Les RCSFs sont déployés, généralement, d'une façon aléatoire dans des zones hostiles et inexplorées par un hélicoptère ou une lance roquette, et doivent s'auto-organiser à l'aide des communications sans fil. La station de base est le seul lien avec le monde extérieur et dispose de ressources moins limitées par rapport aux noeuds capteurs. Un réseau de capteurs est considéré comme un pont entre le monde physique et le système informatique, en fournissant des mesures et des propriétés physiques du monde réel.

Parmi plusieurs défis que connaissent les RCSFs, l'économie d'énergie représente l'un des plus grands défis à soulever pour le bon fonctionnement de ce type de réseaux. En effet, les noeuds capteurs sont généralement alimentés par le biais d'une petite batterie limitée en puissance, et le remplacement ou le changement de celle-ci est une tâche très difficile voire impossible. Par conséquent, l'épuisement des batteries des noeuds capteurs implique la mise hors service du réseau tout entier, ce qui signifie un investissement perdu. La sécurité représente un autre défi très important pour les RCSFs, étant donné que des décisions stratégiques peuvent être prises en se basant sur les informations reçues par les noeuds capteurs.

L'économie d'énergie est considérée comme la principale contrainte à prendre en compte dans les RCSFs, et elle est fortement liée à la sécurité. En effet, le concept de sécurité peut être formulé d'une autre manière dans les RCSFs, dans lequel on doit sécuriser le réseau contre les attaques externes et internes et contre les défaillances des ressources (épuisement des batterie par exemple). Ainsi, on doit proposer des protocoles de communication avec une gestion d'énergie efficace pour de prolonger la durée de vie des réseaux. Plusieurs recherches ont été conduites afin de proposer des protocoles de gestion d'énergie. Parmi ces techniques, le (*Clustering*).

Le clustering consiste à grouper des capteurs dans des (*Clusters*), tel que chaque cluster possède un seul chef (*Cluster-chef* ou *Cluster-head*) et au moins un membre

(*Cluster-membre*). Dans chaque cluster, les cluster-membres rassemblent les informations sur l'environnement et les envoient à leurs cluster-chef, à son tour le cluster-chef traite l'information reçue de ses cluster-membres, en éliminant les redondances, avant de l'envoyer au Sink.

Un autre point qui a fait couler beaucoup d'encre, dans la littérature, et le choix du chemin et de la longueur des sauts (*Hop-length*). En effet, dans la plus part des RCSFs, les capteurs n'ont pas l'énergie et la puissance de transmission nécessaires pour envoyer directement les données collectées au Sink, d'où la nécessité d'utiliser la technique des sauts multiples (*Multi-hops*). Sachant qu'un capteur perd de l'énergie en transmission qu'en réception, cela nous amène au dilemme de la longueur des sauts, car utilisé un grand nombre de sauts de petite distance (*Short-hops*) nous conduit à la perte d'énergie en réception tant que utilisé un nombre réduit de longs sauts (*Long-hops*) nous conduit à la perte d'énergie en transmission.

Notre travail, dans cette thèse, est organisé en deux grandes parties, la première partie consiste à présenter le domaine de recherche (l'état de l'art), tandis que la deuxième est consacrée à nos contributions en termes d'économie d'énergie. La première partie est structurée en 5 chapitres. Le premier chapitre est destiné à introduire les réseaux de capteurs sans fil. Ensuite, pour montrer l'importance des RCSFs, quelques domaines d'utilisation sont présentés dans le deuxième chapitre. Dans le troisième chapitre les défis et les challenges sont présentés. Le plus grand défi, qui porte sur la consommation d'énergie, est détaillé dans le quatrième chapitre. Dans le cinquième chapitre nous présenterons la technique de clustering, qui est utilisée pour conserver l'énergie. La deuxième partie est structurée en deux chapitres ; dans le premier on a présenté une analyse énergétique, où une comparaison est faite entre les longs-sauts et les courts-sauts. Le deuxième chapitre contient notre principale contribution dans cette thèse qui présente une technique de clustering basée sur les algorithmes génétiques pour minimiser la consommation d'énergie dans les RCSFs.



## Première partie

# Introduction sur le domaine de recherche



# les réseaux de capteurs sans fil

---

## 1.1 Introduction

Les réseaux de capteurs sans fil (*RCSFs* ou *WSNs : Wireless Sensor Networks*) sont devenus de plus en plus omniprésents dans notre vie quotidienne [1–3]. Les scientifiques et les industriels leur réservent plus d’attention du fait que ces réseaux peuvent être trouvés dans des divers applications. En effet, on peut les trouver dans plusieurs domaines, tels que : le domaine médical, commercial, militaire, environnemental, etc. Selon MIT’s Technology Review, il s’agit de l’une des dix nouvelles technologies qui vont changer notre vie et notre monde [4], et selon les auteurs dans [5] les RCSFs sont parmi les technologies clés de l’internet des choses (*Internet of Things*).

Les RCSFs sont des réseaux où des milliers, voir des millions, de capteurs sans fil appelés aussi noeuds sans fil ou motes, qui sont déployés aléatoirement dans un champs d’intérêt, dédiés à des applications spécifiques. Ils sont considérés comme un type particulier des réseaux Ad-hoc [3], dans lesquels chaque noeud est doté d’une mémoire limitée, d’un processeur de faible fréquence et d’une simple batterie.

## 1.2 Les réseaux Ad-hoc

Un réseau Ad-hoc, ou MANET (*Mobile Ad-hoc NETWORK*), est un réseau formé dynamiquement par un ensemble arbitraire de noeuds indépendants. Aucune pré-installation ou infrastructure relative à son fonctionnement n’est requise. Dans un réseau Ad-hoc, les noeuds sont supposés des routeurs et des clients à la fois, aussi, chaque noeud est libre de se déplacer et de s’organiser aléatoirement. Ainsi, la to-

pologie du réseau peut changer rapidement et de manière imprévisible [3, 6].

### 1.2.1 Caractéristiques des réseaux Ad-hoc

- **La mobilité des noeuds** : Dans un réseau Ad-hoc, la topologie du réseau est dynamique et peut changer assez rapidement.
- **La liaison sans fil** : Le seul moyen de communication dans les réseaux Ad-hoc est l'utilisation des interfaces sans fil.
- **L'équivalence des noeuds** : Contrairement aux réseaux classique, dans les réseaux Ad-hoc il n'existe pas de différence entre les noeuds, car tous les noeuds peuvent être amenés à assurer les mêmes fonctions.

## 1.3 Les réseaux de capteurs sans fil (RCSFs)

Les réseaux de capteurs sans fil sont considérés comme un type spécial des réseaux Ad-hoc. En conséquence, ils héritent des caractéristiques de ces réseaux, y compris l'architecture sans infrastructure et la communication sans fil. Généralement, les RCSFs sont composés d'un nombre potentiellement très grand de capteurs, déployés dans une zone à couvrir, communiquant selon le modèle « *sources multiples - destination unique* ». Chaque capteur est capable d'effectuer, d'une manière autonome, trois tâches complémentaires, qui sont :

1. Mesure d'une valeur physique (température, pression, humidité, etc),
2. Traitement de ses mesures ( élimination de redondance, agrégation de données, etc),
3. Communication par voie hertzienne (généralement Wifi).

Les capteurs sont définis comme étant des dispositifs électroniques miniaturisés déployés aléatoirement dans une zone géographique appelée champ de captage (ou champ de surveillance). Ces capteurs sont dotés d'une faible capacité de calcul, d'une source d'énergie limitée et d'une bande passante très limitées. Le champ de

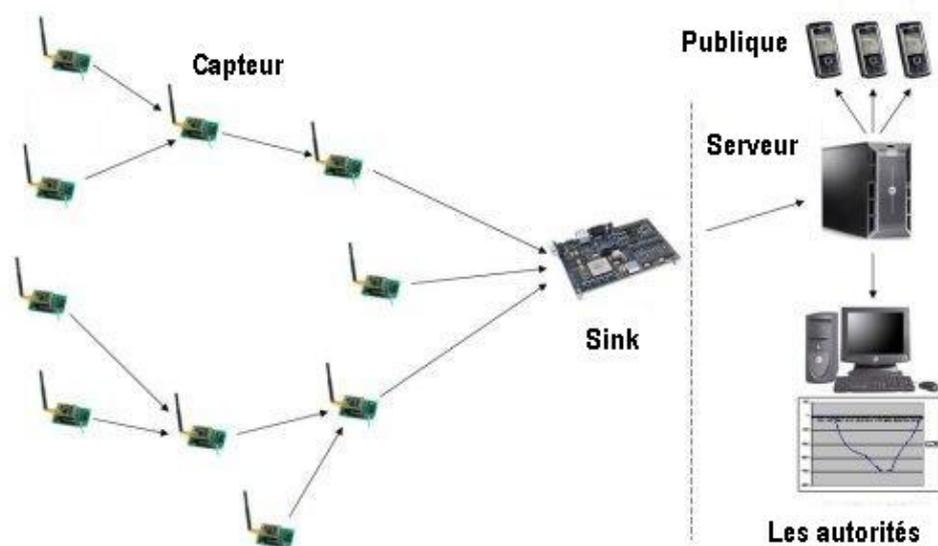


FIGURE 1.1 – Exemple d'un réseau de capteur sans fil

surveillance définit le terrain d'intérêt pour le phénomène à observer. Les données captées sont acheminées grâce à un routage multi-sauts vers un point de collecte appelé noeud puits (appelé aussi : *Sink*, ou *Station de Base*). La station de base est liée aux utilisateurs du réseau via Internet ou un satellite, elle permet de récolter les données environnementales captées et d'établir des requêtes aux noeuds du réseau en fournissant le type de données requis. La figure 1.1 présente un RCSF, composé d'un ensemble de noeuds, lié à un point de collecte qui est connecté à un serveur qui est, à son tour, connecté aux utilisateurs finals.

Les réseaux de capteurs sans fil sont souvent déployés dans des environnements fortement dynamiques et inaccessibles aux humains (à la différence des réseaux informatiques conventionnels), comme les volcans et le fond de la mer, etc [7], de ce fait ils doivent être tolérants à l'échec et autonomes avec une intervention humaine minimale.

Depuis leur apparition, les capteurs ont subi un développement rapide et remarquable, il existe dans nos jours des capteurs de petite taille, avec un prix réduit et des capacités, néanmoins, élevées. La figure 1.2 montre la taille de quelques capteurs

dans nos jours.



FIGURE 1.2 – Exemple de capteurs

## 1.4 Architecture d'un noeud capteur

De nombreuses techniques avancées dans les domaines des *MOEMS* (*Micro-Opto-Electro-Mechanical Systems*) et les technologies de communication sans fil ont permis le développement de petits dispositifs à coût réduit. Ces dispositifs, appelés noeuds ou capteurs ou motes, intègrent, généralement, une unité de calcul de faible puissance (un micro-contrôleur ou un microprocesseur), un ou plusieurs dispositifs de capture (capteur de température, d'humidité, de pression, de fumée, de mouvement, etc), une mémoire limitée, une unité de communication sans fil et une batterie d'alimentation ou un système de récupération d'énergie depuis l'environnement (énergie solaire) [8,9].

L'architecture de base d'un noeud capteur est composée, au minimum, de quatre unités fondamentales (voir figure 1.3), à savoir : l'unité de captage ou d'acquisition, l'unité de traitement, l'unité de communication et l'unité de contrôle d'énergie. Cette architecture peut, également, contenir d'autres modules, selon le domaine d'application, tel qu'un système de localisation (GPS), ou bien un système générateur d'énergie (cellule solaire), elle peut, aussi, inclure un système de mobilité chargé de déplacer le micro-capteur dans le champ de captage.

Un schéma d'une architecture typique d'un noeud capteur est présenté dans la figure 1.4. Le but de cette section est de détailler le fonctionnement de chaque

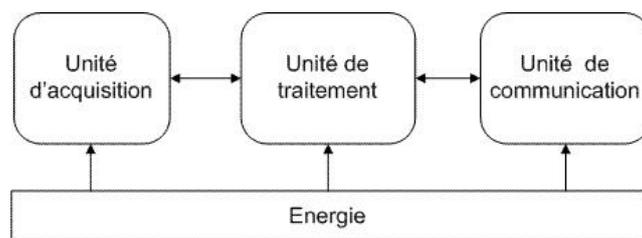


FIGURE 1.3 – Architecture de base d'un noeud capteur

composant d'un noeud capteur.

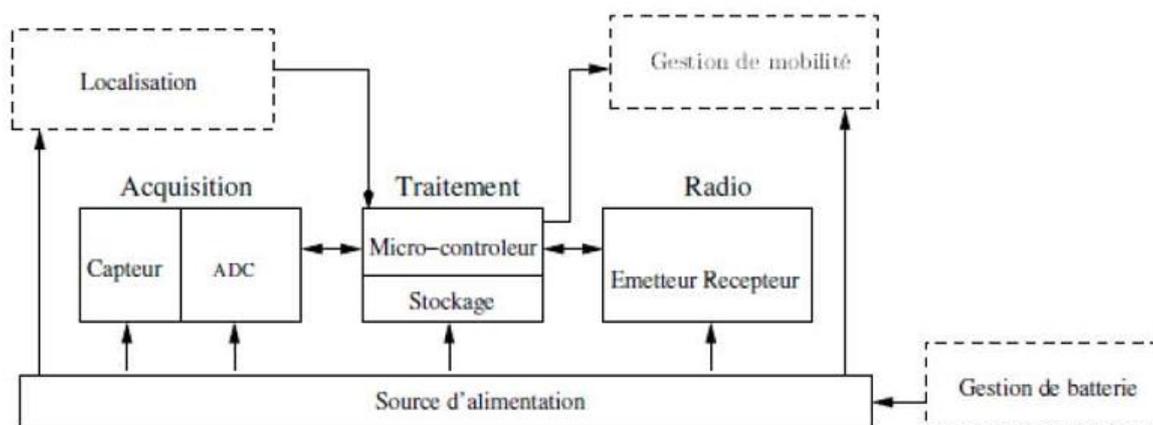


FIGURE 1.4 – Architecture matérielle d'un noeud capteur

### 1.4.1 Micro-contrôleur

Le micro-contrôleur représente le coeur du noeud. Il reçoit les données provenant des unités de captures, traite ces données, décide quand et où les envoyer et gère la commande des actionneurs (s'ils existent). Le micro-contrôleur reçoit aussi des paquets en provenance des noeuds voisins, il les analyse et répond à ces messages, si nécessaire. D'autres tâches plus complexes sont confiées au micro-contrôleur telles que la gestion de la mémoire, la gestion de l'énergie (alterné du mode actif, au mode veille ou veille prolongée et vice versa), l'ordonnancement des tâches, la gestion des périphériques, etc. Les plate-forme existantes des RSCFs utilisent une grande variété

de micro-contrôleurs à architectures différentes [10]. Nous présenterons ci-après les micro-contrôleurs les plus employés par les constructeurs des RCSFs.

Parmi les micro-contrôleurs les plus réponsus (dédiés pour les RCSFs), on trouve le MSP430 de Texas Instrument. Il s'agit d'un micro-contrôleur 16 bits réputé pour sa faible consommation d'énergie. Le MSP430 est intégré dans de nombreuses plateformes de réseaux de capteurs telles que *Telos*, *BSN*, *Wavenis* et *eyesIFX*. Deux familles de très faible consommation d'énergie sont disponibles sur le marché : la famille des MSP430F1xx, ayant une puissance dynamique inférieure mais fonctionne à une fréquence plus faible (ne dépasse pas 8 MHz), et la famille des MSP430F2xx qui peut atteindre une fréquence de fonctionnement de 16 MHz [11].

Intel Strong ARM est un processeur RISC 32 bits qui peut atteindre une fréquence de fonctionnement de 206 MHz, il est utilisé dans des applications nécessitant le traitement des flux importants de données (applications vidéos, applications temps réel, etc) [12, 13].

*Mica* est un micro-contrôleur pédagogique asynchrone, on le trouve souvent dans les laboratoires de recherche. Il a été conçu au laboratoire *TIMA* avec la collaboration de *France Telecom*, il est fabriqué par *ST-Microelectronics* d'une taille de  $0,25 \mu\text{m}$  avec une bibliothèque de cellules standards. Ce micro-contrôleur dispose d'un bus de données de 8 bits, un bus d'adressage de 16 bits, une mémoire RAM de 16 Ko, une ROM de 2 Ko et un périphérique permettant de gérer jusqu'à 6 ports parallèles de 8 bits et deux ports en série. La consommation de ce micro-contrôleur dépend de son mode de fonctionnement (varie de 0.8 mW sous 1 V à 77 mW sous 3.5 V) [11].

Le *Plutonium 8051* est un micro-contrôleur universitaire asynchrone conçu en technologie asynchrone au *California Institute of Technology*, et fabriqué en technologie CMOS  $0,18 \mu\text{m}$  de TSMC. La consommation de ce micro-contrôleur varie en fonction du mode de fonctionnement (de 0.17 mW sous 0.5 V à 100 mW sous 1.8 V) [9].

### 1.4.2 Convertisseur Analogique/Numérique (CAN)

Le Convertisseur Analogique/Numérique (*CAN*) est nécessaire pour convertir les signaux analogiques provenant des unités de captures en valeurs numériques exploitables par le micro-contrôleur. Le choix d'un CAN dépend des besoins de l'utilisateur et des contraintes imposées par l'application (vitesse de conversion, coût du convertisseur, énergie consommée par la conversion, la taille du CAN, etc). Les bénéfices en termes de réduction d'énergie des circuits asynchrones comparativement aux circuits synchrones ont poussé les chercheurs à concevoir des CAN asynchrones (un CAN asynchrone prend un échantillon uniquement lorsque la grandeur physique surmonte un seuil d'amplitude prédéfini). La technologie asynchrone souffre cependant à l'heure actuelle d'un manque de maturité et de fiabilité qui peut expliquer la rareté des CAN asynchrones commercialisés [9–11].

### 1.4.3 Horloge temps réel

Elle utilise souvent un quartz piézoélectrique afin de conserver l'information de temps dans le système. Les protocoles de communication se servent de l'horloge temps réel pour avoir une heure commune entre les noeuds (synchronisation des noeuds entre eux). Cette horloge peut aussi servir au système logiciel en tant que *Timer* pour réveiller le micro-contrôleur à des instants bien précises. Ce Timer peut, aussi, aider les noeuds à définir les intervals de sommeil et d'action (*Sleep/wake interval*).

Cette partie matérielle est active en permanence, sa consommation d'énergie doit donc être faible pour avoir une longue durée de vie du noeud. La consommation typique des horloges temps réel est de quelques centaines de nanoampères, ce qui rend sa consommation comparable aux consommations en veille des autres blocs fonctionnels du noeud [9, 11].

#### 1.4.4 Module de communication sans fil

Ce bloc représente la partie matérielle qui consomme le plus d'énergie. Le module de communication sans fil peut être un émetteur/récepteur optique, ultrasons, bluetooth ou Wifi, mais généralement il s'agit d'un émetteur/récepteur radio qui utilise principalement l'une des 3 bandes *ISM* (*Industrial Scientific and Medical bands*) suivantes : la bande des 400 MHz, la bande des 900 MHz et la bande *UWB* des 2.4 GHz. Plusieurs travaux de recherche ont permis de réduire la consommation du module radio afin de prolonger la durée de vie du noeud capteur. La plupart de ces études ont visé la réduction du phénomène de l'écoute passive ou l'écoute inutile ; mais malgré cela, la consommation due à cet composant reste importante [9, 11, 14].

Cette unité se charge de transmettre et de recevoir les paquets de données en utilisant un dispositif de communication sans fil. Ce dernier est, généralement, une antenne radio à faible portée afin d'économiser l'énergie des noeuds capteurs. L'unité de transmission doit aussi contenir des circuits de modulation, démodulation, filtrage et multiplexage, pour le traitement du signal radio [3].

#### 1.4.5 Système de localisation

Généralement, un capteur n'est utile que si une information relative à sa localisation géographique est fournie. Le *GPS* (*Global Positioning System*) est le système de localisation le plus connu et le plus utilisé au niveau mondial. D'autres systèmes de géolocalisation existent, parmi les quels on cite le système russe *GLONASS*, le système indien *IRNSS*, le système Beidou développé par la chine et le système de localisation européen Galileo. Le principe général de ces technologies se décrit comme suit : lorsqu'un module de localisation demande sa position, chaque satellite lui communique ses coordonnées et la distance qui le sépare de ce satellite. Au moins 3 satellites sont nécessaires pour une localisation dans un espace à deux dimensions, alors qu'au moins 4 satellites le sont pour un espace à trois dimensions [15].

### 1.4.6 Capteurs/actionneurs

Le rôle des capteurs (*unité de détection ou Sensors*) est de prendre en charge le captage et l'acquisition des données à partir de l'environnement surveillé.

On peut trouver plusieurs unité de détection ou capteurs embarqués sur un noeud. Ces capteurs transforment le ou les grandeurs physiques observées (pression, température, humidité, vitesse, mouvement, etc) en grandeur électrique (tension électrique, intensité, train d'impulsions, etc).

Certains modèles de réseaux des capteurs sont composées de noeuds équipés de plusieurs actionneurs. Un actionneur permet de convertir une énergie d'entrée (généralement une énergie électrique dans le cas des RCSFs) en une énergie de sortie (généralement un signal électrique de commande) utilisable pour commander ou contrôler un processus (par exemple déclencher une alarme suite à un incendie) [3, 11, 16].

### 1.4.7 Mémoires

Les noeuds capteurs sont équipés de plusieurs mémoires, de type : *RAM (Random Access Memory)*, *ROM (Read Only Memory)*, *EEPROM* ou *flash*. La mémoire du programme permet de sauvegarder la suite des instructions des programmes à exécuter par le système d'exploitation implémenté dans le noeud. La mémoire de données permet le stockage des données à traiter par le noeud capteur. Une mémoire de configuration est figée au sein du noeud (généralement de type ROM ou EEPROM), elle maintient les données de configuration nécessaires au bon fonctionnement de l'ensemble des blocs fonctionnels des noeuds. La taille et le type de ces mémoires varient en fonction du model des noeuds utilisés [11].

### 1.4.8 L'unité de contrôle d'énergie

A cause de leurs tailles très réduites et leurs faibles coûts de fabrication, les noeuds capteurs sont équipés d'une ressource énergétique à faible autonomie (géné-

ralement une batterie). De plus, les caractéristiques hostiles de l'environnement et du déploiement rendent généralement ces ressources d'énergie irremplaçables. Donc, l'énergie est la ressource la plus précieuse dans un réseau de capteurs puisqu'elle influe directement sur la durée de vie des micro-capteurs, voire du réseau tout entier. Ainsi, l'unité de contrôle d'énergie (*Co-processeur DVS*) représente l'une des unités les plus importantes dans un noeud capteur. En effet, celle-ci se charge d'alimenter les autres unités en énergie tout en minimisant la consommation énergétique du noeud capteur. Ainsi, cette unité peut mettre en veille les composants inactifs par exemple, afin de réduire le gaspillage d'énergie [3].

La technique DVS (*Dynamic Voltage Scaling* ou adaptation dynamique de tension) permet de contrôler efficacement la vitesse du micro-contrôleur afin d'optimiser la consommation d'énergie. Cette technique permet de ralentir le processeur (lorsque cela est possible) en abaissant la tension d'alimentation. En effet, dans un circuit *CMOS*, l'énergie consommée est proportionnelle au carré de la tension d'alimentation alors que la vitesse du micro-contrôleur varie linéairement en fonction de cette tension. Ainsi, une faible diminution de la tension ( $V_{dd}$ ) ralentit faiblement le circuit mais permet de réduire de manière importante la consommation énergétique [9, 11].

#### 1.4.9 Batterie / système de récupération d'énergie

Les noeuds disposent d'une batterie d'alimentation comme source principale d'énergie. Certains models utilisent des systèmes de récupération d'énergie de l'environnement. Dans nos jours, l'utilisation de l'énergie solaire est le moyen de récupération d'énergie le plus adéquat pour les noeuds d'un RCSF [17].

### 1.5 Les communications dans les RCSFs

Contrairement aux réseaux Ad-hoc, les réseaux de capteurs sans fil imposent des contraintes supplémentaires aux protocoles de communication. Par conséquent, le modèle traditionnel en couches (modèle OSI), ne répond pas aux exigences de ce

type particulier des réseaux. En effet, les RCSFs adoptent une version simplifiée du modèle OSI, dans laquelle sont ajoutées de nouvelles couches afin de remédier aux contraintes et aux limitations imposées. Ainsi, le nouveau modèle se compose de 5 couches similaires à celles du modèle OSI (physique, liaison, réseau, transport et application), et trois plans de gestion dédiés pour le contrôle d'énergie, de mobilité et des tâches particulières [3].

La pile protocolaire utilisée par les capteurs est représentée dans la figure 1.5. Cette pile combine entre l'énergie et le routage. La pile protocolaire est constituée de la couche physique, la couche liaison de donnée, la couche réseau, la couche transport et la couche application, ainsi que le plan de gestion d'énergie, le plan de gestion de mobilité et le plan de gestion des tâches [2].

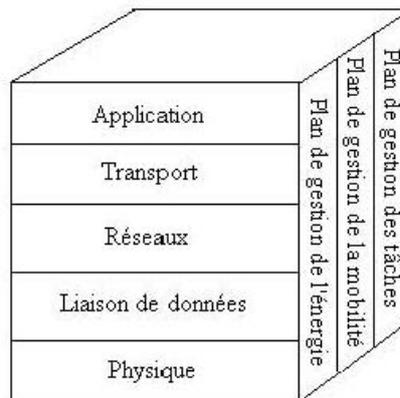


FIGURE 1.5 – La pile protocolaire dans les RCSFs

1. **Couche physique** : Comme celle du modèle OSI, cette couche est responsable de la modulation, la détection du signal, la sélection des fréquences porteuses et le cryptage des données [2].
2. **Couche liaison de données** : Cette couche est chargée du contrôle d'erreurs, du multiplexage des flux de données et le contrôle d'accès au média de transmission (*Medium Access Control*). Cette couche assure la connexion point-à-point ou point-à-multipoint dans les communications réseaux de cap-

teurs. plus spécifiquement, nous présentons les stratégies *MAC* et le control d'erreurs dans les sections suivantes.

– **MAC** : Le protocole *MAC* (*Medium Access Control*) dans un réseau de capteur sans fil doit atteindre deux objectifs. Le premier but est de créer l'infrastructure réseau, en assurant la synchronisation et l'auto-configuration des capteurs déployés. En effet, des milliers de noeuds capteurs peuvent être densément dispersés dans un champ de surveillance, le protocole *MAC* doit assurer la capacité de l'auto-organisation entre les noeuds et doit établir des liens de communication pour le transfert de données, ceci constitue l'infrastructure de base nécessaire pour la communication saut-à-saut (*Hop-by-Hop*). Le deuxième objectif est de partager de façon équitable et efficace les ressources de communication entre les noeuds capteurs. Ces ressources comprennent le temps, l'énergie et la fréquence de transmission. Plusieurs protocoles *MAC* ont été développés pour les réseaux de capteurs pour répondre à ces besoins au cours de la dernière décennie, nous citons, comme exemple : Sensor-MAC [18], B-MAC [19], MC-LMAC [20], etc.

– **Control d'erreur** : Une autre fonction importante de la couche de liaison de données est le contrôle d'erreurs dans la transmission de données. Deux modes importants de contrôle d'erreurs dans les réseaux de communication sont la correction avant l'erreur (*FEC*) et demande de répétition automatique (*ARQ*). Dans la conception d'un tel système, il est important d'avoir une bonne connaissance des caractéristiques du canal utilisé et des techniques de mise en oeuvre.

3. **Couche réseau** : L'objectif de cette couche est de trouver des chemins de routage à faible coût en terme d'énergie pour transmettre les données captées vers la station de base. Ainsi, les protocoles de cette couche doivent toujours prendre en compte les limitations des ressources des noeuds capteurs.

Les noeuds capteurs sont dispersés avec une grande densité dans une zone de surveillance. Les informations recueillies depuis l'environnement doivent être transmises au Sink, qui peut être située loin du champ de détection. Toutefois, la portée de communication limitée des capteurs empêche la communication directe entre les noeuds et le Sink. Cela nécessite des protocoles de routage efficace en utilisant des multi-sauts (*multi-hops*) entre le capteur source et le Sink en utilisant les noeuds intermédiaires. Les techniques de routage existantes, qui ont été développés pour les réseaux ad-hoc, ne correspondent généralement pas aux exigences des réseaux de capteurs sans fil [2,3]. La couche réseau des réseaux de capteurs est généralement conçu selon les principes suivants :

- L'efficacité énergétique est toujours un facteur important,
- En plus du routage, les noeuds intermédiaires peuvent agréger les données et éliminer la redondance provenant de plusieurs voisins à travers un traitement local,
- En raison du grand nombre des noeuds déployés dans un RCSF, des identifiants uniques pour chaque noeud ne peuvent pas être fournis, cependant les noeuds capteurs sont traités sur la base de leurs données ou leurs emplacements.

4. **Couche transport** : La couche de transport est particulièrement nécessaire lorsque le réseau est prévu pour être accessible à travers l'Internet ou d'autres réseaux externes. Son rôle est le contrôle de flux, le découpage, l'ordonnancement et le transport des paquets de données et la gestion des erreurs de transmission.

Contrairement aux protocoles tels que *TCP*, les systèmes de communication dans les réseaux de capteurs sans fil ne sont pas fondées sur l'adressage. Ces systèmes de communications doivent utilisés l'adressage basé sur les données

ou l'emplacement des noeuds.

Le développement de protocoles du couche de transport est une tâche difficile parce que les noeuds capteurs sont soumis à des contraintes matérielles telles que l'énergie et la mémoire limitée. En conséquence, chaque noeud capteur ne peut pas stocker de grandes quantités de données comme un serveur dans les réseaux classiques, ainsi que les accusés de réception sont trop coûteux pour un réseaux de capteurs sans fil [2].

5. **Couche application** : Afin de fournir une interface d'interaction avec l'utilisateur humain, les noeuds capteurs peuvent être dotés d'une couche application, dont le rôle est d'implémenter l'ensemble d'applications et de logiciels d'interaction [3].

La couche d'application comprend l'application principale, ainsi que plusieurs fonctionnalités de gestion. En plus du code qui est spécifique pour chaque application, les traitements des requêtes et les fonctionnalités de gestion du réseau résident également dans cette couche [2].

6. **Le plan de gestion d'énergie** : Les noeuds capteurs sont sévèrement limités en ressources d'énergie, qui influence directement sur la durée de vie du réseau. Ainsi, le plan de gestion d'énergie doit fournir des mécanismes de gestion efficaces pour réduire le degré de consommation d'énergie, et éliminer les sources de gaspillage de celle-ci [2].
7. **Le plan de gestion de mobilité** : Ce plan est responsable du contrôle du mouvement des noeuds capteurs dans le cas où ils sont mobiles. Il peut par exemple enregistrer les trajectoires d'un noeud capteur afin de l'aider à se localiser.
8. **Le plan de gestion des tâches** : Dans un réseau de capteurs, les noeuds peuvent effectuer des tâches qui se différencient en termes de consommation de ressources. Ainsi, un plan de gestion de tâche est souvent nécessaire afin de

répartir d'une manière équitable les tâches sur les noeuds capteurs, et offrir ainsi une gestion efficace des ressources disponibles.

## 1.6 Architecture d'un réseau de capteurs sans fil

Les noeuds capteurs sont, habituellement, dispersés dans une zone de surveillance. Chaque noeuds à la possibilité de collecter les données et de les router vers une ou plusieurs stations de base (*sink node*). Ce dernier est un point de collecte des données capturées. Il peut communiquer les données collectées à l'utilisateur final à travers un réseau de communication, éventuellement l'Internet [21], comme le représente la figure 1.6.

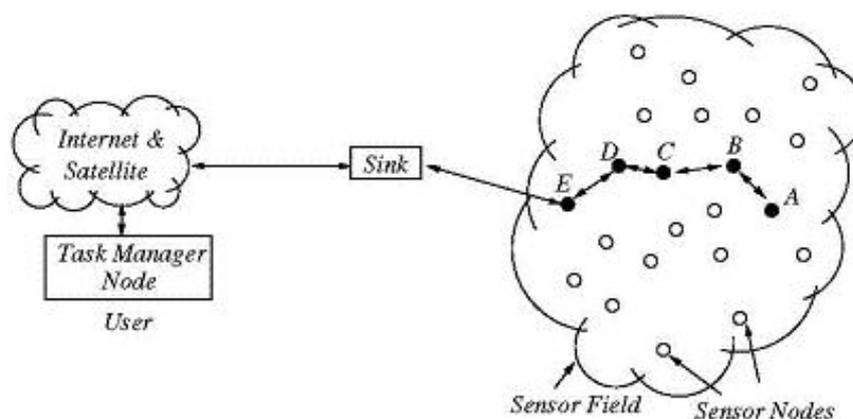


FIGURE 1.6 – Exemple de RCSF.

A un niveau plus élevé un RCSF peut être vu comme étant une combinaison de deux entités de réseaux [22] :

1. Un réseau d'acquisition de données : C'est l'union des noeuds capteurs et du sink, son rôle consiste à collecter les données à partir de l'environnement et de les rassembler au sink.
2. Un réseau de distribution de données : Son rôle est de connecter le réseau d'acquisition des données à un utilisateur.

### 1.6.1 Les types d'architecture des RCSFs

Deux architectures sont connus pour les RCSFs :

#### 1.6.1.1 Les réseaux de capteurs sans fil plats (*Flat Network*)

Un réseau de capteur sans fil plat est un réseau homogène, où tous les noeuds possèdent les mêmes capacités de communication, de traitement et de capture, etc. Le sink peut être différent puisque il joue le rôle de passerelle chargée de transmettre les informations collectées à l'utilisateur final, voir figure 1.7. Dans une telle architecture, les noeuds ne connaissent pas leurs successeurs dans le routage, les communications entre noeuds sont assurées, généralement, par une simple suite de *RTS/CTS/DATA/ACK*.

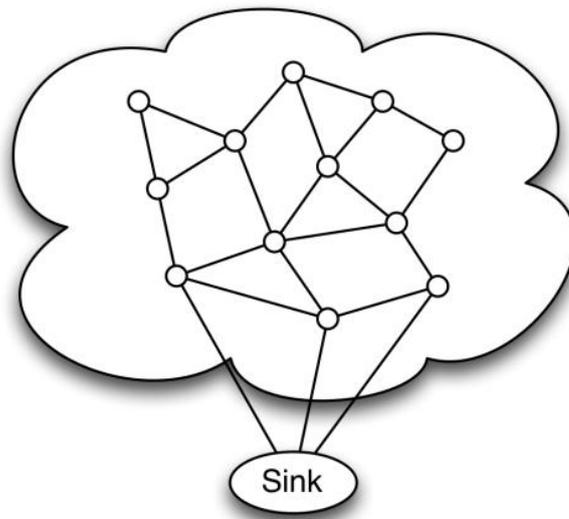


FIGURE 1.7 – Architecture Plat des RCSFs.

#### 1.6.1.2 Les réseaux de capteurs sans fil hiérarchiques (*Hierarchical Network*)

C'est un réseau hétérogène où les noeuds peuvent disposer des ressources (une source d'énergie, une portée de communication ou une puissance de calcul) différentes les unes des autres. Dans les réseaux hiérarchiques, certains noeuds sont

destinées à la collecte de données et d'autres au traitement local et au routage de ces données, ceci est représenté sur la figure 1.8. Dans ce type de réseaux on trouve la notion de noeuds chefs et noeuds membres, où les noeuds chefs sont considérés comme des pères des noeuds membres.

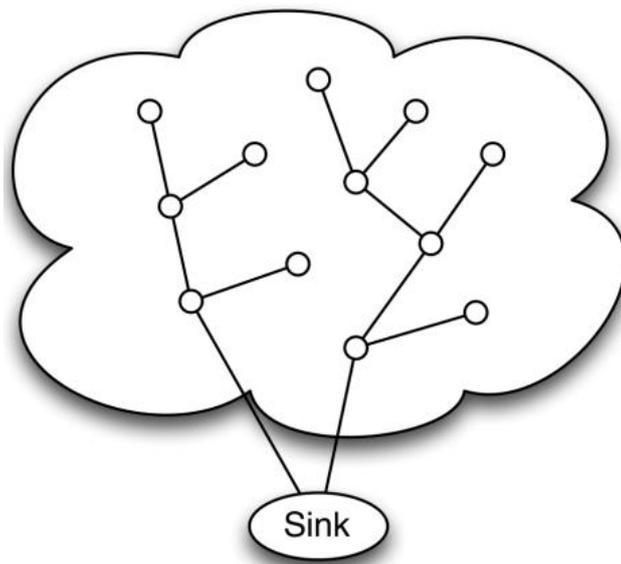


FIGURE 1.8 – Architecture hiérarchique des RCSFs.

## 1.7 Similitudes et différences entre les réseaux MANETs et les RCSFs

### 1.7.1 Similitudes entre les réseaux MANETs et les RCSFs

1. Les réseaux MANETs (*Mobile Ad-hoc NETWORKS*) et les RCSFs fonctionnent d'une manière Ad-hoc, où il n'y a pas une infrastructure fixe. Le réseau est auto-configurable, et les noeuds utilisent leurs voisins pour transmettre les données depuis une source vers la destination.
2. Les deux types de réseaux sont deux sous-classes des réseaux sans fil. Car, les noeuds dans MANET et RCSF utilisent des liaisons sans fil pour communiquer.

### 1.7.2 Différences entre les réseaux MANETs et les RCSFs

Bien que les MANETs et les RCSFs partagent certaines caractéristiques communes, ils ont encore de nombreux aspects qui les rendent différents [2, 23].

1. **La mobilité** : Bien qu'ils puissent parfois être mobiles, les noeuds d'un RCSF sont, dans la plupart des cas, stationnaire, en gardant la même position après le déploiement. Les dispositifs dans les MANETs, cependant, sont mobiles par définition, assurer de la mobilité est un grand défi, surtout dans le domaine de routage et la consommation d'énergie.
2. **Les capacités et les niveaux de puissance de calcul** : Dans les RCSFs, les capteurs sont généralement de petite taille ce qui les rend très limités en terme d'énergie et en terme de capacités de calcul. L'énergie devient donc l'un des principaux défis dans le domaine des RCSFs, et la nécessité pour développer des protocoles consommant moins d'énergie devient une tâche cruciale. Les faibles capacités de traitement et de mémoire des capteurs imposent également de nouveaux paradigmes et des techniques de communication, comme la fusion des données, le codage des données et la compression ; d'autre part les réseaux MANETs sont généralement moins limités en terme d'énergie et de capacités de calcul, puisque les appareils sont généralement des ordinateurs portables ou des appareils portatifs tels que les *PDA*, avec des processeurs plus puissants, avec plus de mémoire et les batteries sont rechargeables.
3. **évolutivité** : Typiquement, les RCSFs se composent de centaines ou de milliers, voir des millions, de noeuds (ou plus), tandis que les MANET sont moins denses. Le problème d'évolutivité devient très important dans les RCSFs, et les topologies prenant en considération l'évolutivité (par exemple les topologies hiérarchiques) pour ces réseaux sont d'une grande nécessité.
4. **Adressage** : Dans MANET, comme dans tout les réseaux traditionnels, les noeuds ont des identifiants uniques au monde comme les adresses *IP* et les

adresses *MAC*, rendant, ainsi, la communication basée sur l'adressage. D'autre part, dans les RCSFs nous ne sommes pas intéressés par les noms des capteurs ou leurs adresses IP. La question devient : "Quelle est la température de la pièce N13" Au lieu de "Quelle est la température fournie par le capteur N13". Ce mode de communication est appelé centré sur les données (*Data-centric*), où les utilisateurs sont intéressés par les données fournies par les noeuds.

5. **Le flux de données** : Dans les réseaux MANET, tous les noeuds sont des paires, et la communication pourraient survenir entre toute paire de noeuds. Il n'y a aucune forme particulière pour les flux de données, cela dépend des protocoles de communication.

Dans les réseaux RCSFs, il pourrait y avoir de nombreux rôles assumés par les noeuds, comme le Sink, qui représente la passerelle entre le réseau et le monde extérieur, et il pourrait y avoir des noeuds spéciaux qui appartiennent à la *backbone* du réseau ou des noeuds qui représentent des cluster-chefs (*Cluster-heads*). Le flux des données dans un RCSF est toujours le même, il est unidirectionnel démarrant d'un noeud feuille et atteindront le Sink. Cependant, les RCSFs semblent plus robustes et résistants aux erreurs (*Fault tolerance*), mais l'échec du Sink ou de l'un des noeuds spéciaux ne peut pas être ignoré dans un RCSF. [23].

6. **Modèle de trafic** : Les RCSFs sont conçus pour un type particulier d'applications, principalement pour la surveillance (par exemple l'environnement, les signes vitaux d'un patient, les structures critiques, etc), où le réseau produit un faible trafic pour de très longues périodes, et pourrait avoir de grandes données à faire communiquer quand quelque chose arrive (par exemple la détection d'intrusion ou la haute pression sanguine d'un patient). Les applications dans les MANETs d'autre part sont plus générique (par exemple, Web, voix, etc), et ils ont des modèles de trafic plus traditionnels.

## 1.8 Conclusion

Les réseaux de capteurs sans fil présentent un intérêt considérable et une nouvelle étape dans l'évolution des technologies de l'information et de la communication sans fil. La flexibilité, la tolérance aux fautes, le prix réduit et la mise en place rapides de ces réseaux de capteurs offrent des possibilités infinies de développement dans tous les domaines d'application. Ceci nous permet de penser que les réseaux de capteurs feront bientôt une partie intégrante de notre vie en satisfaisant sûrement les besoins de nos plus grands projets. Nous avons essayé à travers ce chapitre de donner une vue générale sur les capteurs et les RCSFs, en introduisant leurs architectures et leurs fonctionnements.

# Domaines d'application des RCSFs

---

## 2.1 Introduction

Les caractéristiques particulières des noeuds capteurs (le faible coût, la taille réduite, communication sans fil, etc), ont permis d'étendre rapidement leurs domaines d'application. Les domaines d'application des réseaux de capteurs sont diversifiés, on peut les trouver dans les applications : militaires, environnementales, de santé, la domotique, de l'industrie et de la sécurité [2]. En effet, les RCSFs peuvent être utilisés pour éviter les catastrophes naturelles, surveiller les infrastructures, renforcer les systèmes de surveillance militaire, conserver les ressources naturelles précieuses, augmenter la productivité, permettre de rendre les maisons intelligentes et beaucoup d'autres domaines. Dans ce qui suit nous présenterons quelques exemples d'application des RCSFs.

## 2.2 Applications militaires

Les RCSFs sont devenus une partie intégrante dans le commandement militaire, le contrôle, la communication, le renseignement, la surveillance, la reconnaissance et la détection. Le déploiement rapide, l'auto-organisation et la tolérance aux pannes de ces réseaux de capteurs en font une technique de détection très prometteuse pour les domaines militaires. Les réseaux de capteurs sont basés sur le déploiement dense de noeuds capteurs, généralement à usage unique et à faible coût, en outre, la destruction de certains noeuds par une action hostile n'interrompt pas une opération comme celle de la destruction d'un capteur traditionnel dans un réseau de capteurs filaires, ce qui rend le concept des réseaux de capteurs sans fil une meilleure approche

pour les champs de bataille.

Certaines applications militaires des réseaux de capteurs sont utilisées dans la surveillance des forces amies et les champs de bataille, la détection de l'épuisement des munitions, la reconnaissance des forces et des terrains ennemis, le ciblage, l'évaluation des dégâts et la détection des attaques nucléaire, biologique et chimique (*NBC attacks*) [24–26]. La figure 2.1 montre un exemple d'utilisation d'un RCSF dans les applications militaires, où les soldats, les armes et les engins peuvent être dotés avec des capteurs pour connaître leurs positions en temps réel et pour éviter, par exemple, la collision entre les forces amies.

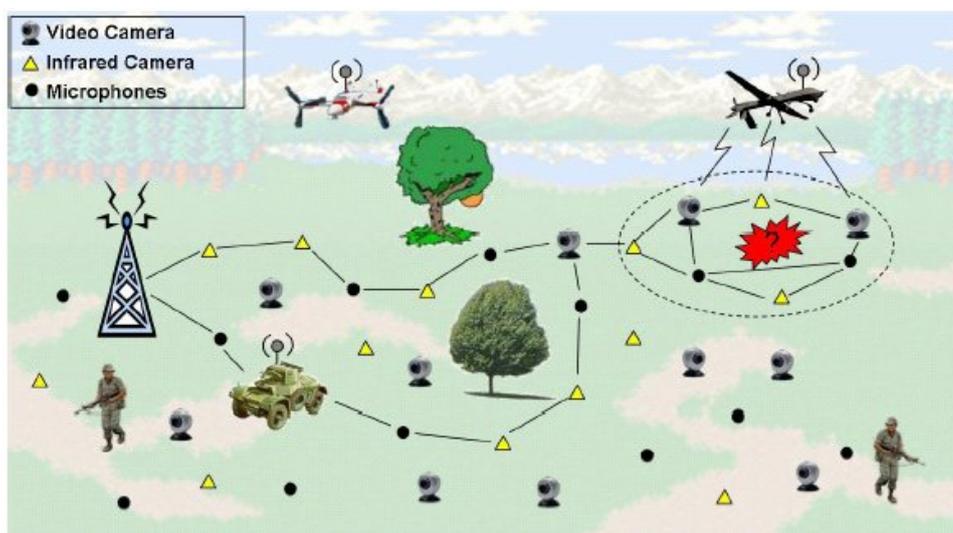


FIGURE 2.1 – Exemple d'utilisations d'un RCSF dans les applications militaires

### 2.3 Applications environnementales

Les capacités de coordination autonomes des capteurs, rendent l'utilisation des RCSFs possible dans la réalisation d'une grande variété d'applications environnementales. Certaines applications environnementales, des RCSFs, comprennent le suivi des mouvements des oiseaux, des animaux et des insectes, la surveillance des conditions environnementales qui affectent l'agriculture, le bétail et l'irrigation, la détection des feux de forêt, la recherche météorologique ou géophysique, l'études

de la pollution, la détection chimique et la surveillance de la terre d'agriculture et l'inondation.

Divers projets ont été développés durant cette dernière décennie pour protéger des espèces rares tel que *ZibraNet* pour protéger les Zèbres, la surveillance de quelques volcans actifs, la détection précoce des inondations [27–31]. La figure 2.2 montre un exemple d'utilisation des RCSFs dans la détection des feux de forêts où les services concernés sont avisés, même, avant le déclenchement d'un feu avec les coordonnées exactes de la zone susceptible aux incendies ; et la figure 2.3 montre un cas d'utilisation d'arrosage automatique pour connaître les parties de terre ayant besoin d'eau.



FIGURE 2.2 – Exemple d'utilisations des RCSFs dans la détection des feux de forêts

## 2.4 Applications de santé

Les développements des appareils biomédicaux et des capteurs intelligents rendent l'utilisation des réseaux de capteurs sans fil possible dans les applications biomédicales, voir figure 2.4. Certaines applications de santé pour les réseaux de cap-

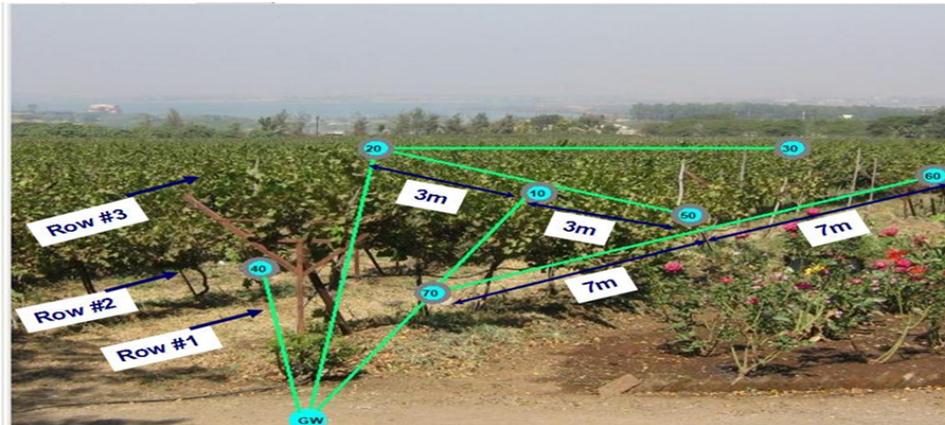


FIGURE 2.3 – Exemple d'utilisations des RCSFs dans l'arrosage goûte à goûte

teurs sont : la fourniture d'interfaces pour les personnes handicapées, la surveillance des patients, le diagnostic, l'administration de médicaments dans les hôpitaux, la télésurveillance de données physiologiques humaine et le suivi et la surveillance des médecins et des patients dans un hôpital [32–35].

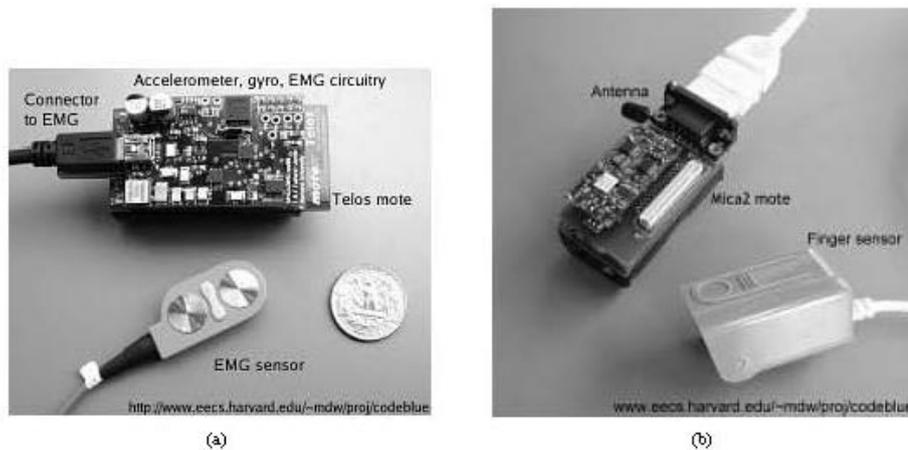
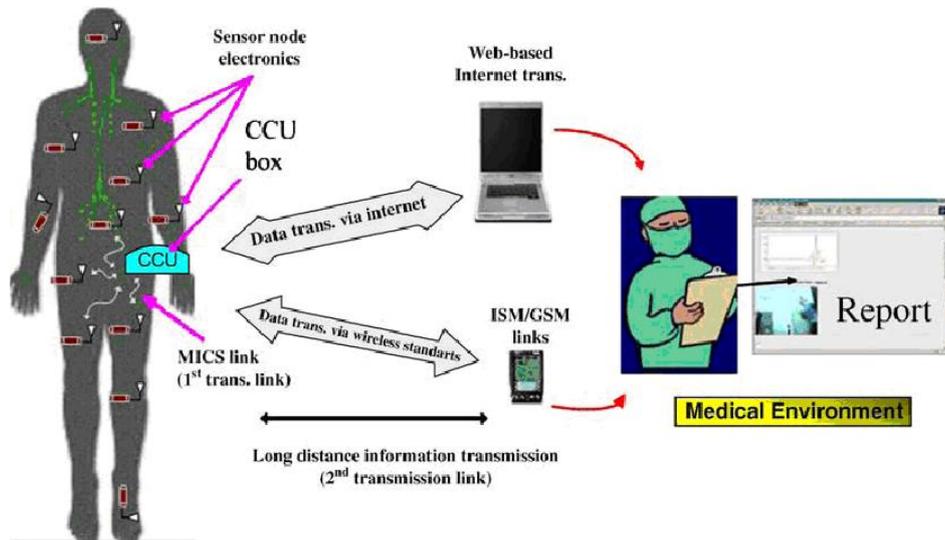


FIGURE 2.4 – Exemple de capteurs bio-médicaux

Le projet *CodeBlue*, présenté sur la figure 2.5 a été développé à l'Université de Harvard, ce projet se concentre sur les capteurs portables qui surveillent les signes vitaux des patients tout au long de leur vie quotidienne [34]. À cette fin, des capteurs : de l'oxymètre de pouls, d'électrocardiographie (*ECG*) et d'electromyo-

FIGURE 2.5 – Exemple d'application de santé : *CodeBlue*

graphiques (*EMG*) ont été conçus pour les capteurs *MicaZ* et *Telos*. En conséquence, la fréquence cardiaque, la saturation du sang en oxygène, les activités électriques du cœur, les mouvements du patient et l'activité musculaire peuvent être surveillés en permanence. La plate-forme logicielle *CodeBlue* permet à ces noeuds d'être exploités dans un réseau de capteurs sans fil, où le personnel médical peut surveiller les patients à travers un *PDA*.

## 2.5 Applications domotique

Avec la progression de la technologie, les noeuds avec capteurs intelligents peuvent être embarqués dans des appareils tels que les aspirateurs, les fours à micro-ondes, les réfrigérateurs et les lecteurs de DVD [36], ainsi que les systèmes de surveillance de l'eau et de Gaz [37]. Ces noeuds capteurs intégrés à l'intérieur des appareils domestiques peuvent interagir les uns avec les autres et avec le réseau externe via l'Internet ou par satellite. Ils permettent aux utilisateurs finaux de gérer plus facilement les dispositifs domestiques à la fois localement et à distance. En conséquence, les réseaux de capteurs permettent l'interconnexion des différents dispositifs sur les lieux de résidence avec le contrôle pratique de diverses applications à

la maison. La figure 2.6 montre un exemple de maison intelligente ou le propriétaire peut contrôler à distance sa maison.

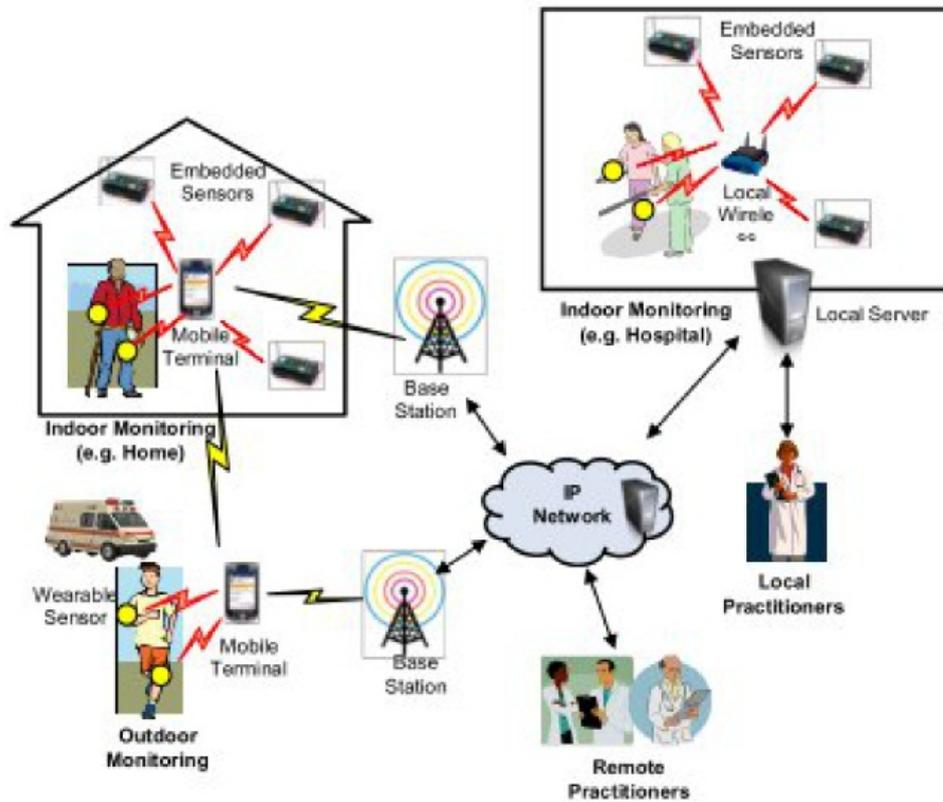


FIGURE 2.6 – Exemple d'application domotique

## 2.6 Applications industrielles

Les Réseaux de capteurs filaires ont été, pour longtemps, utilisés dans les domaines industriels pour la télédétection industrielle, l'automatisation du bâtiment et le contrôle d'accès. Cependant, le coût associé à l'installation des capteurs câblés limite l'applicabilité de ces systèmes. En effet, même le placement d'un nouveau capteur dans une installation existante coûterait presque autant que l'installation d'un nouveau système. Aussi, Le suivi d'un réseau de capteurs filaire traditionnel est assuré, généralement, manuellement par un personnel expérimenté utilisant des analyseurs portables à partir d'un emplacement central pour l'analyse [38]. Donc

les systèmes à base de capteurs filaire entraînent des coûts élevés, ainsi que les systèmes de contrôle manuels ont une précision limitée et nécessitent du personnel expérimenté. Au lieu de cela, les réseaux de capteurs sans fil sont une solution alternative prometteuse pour ce domaine en raison de leurs facilités de déploiement, leurs hautes précision et leurs faibles coût [39–41].

La figure 2.7 montre des exemples d'utilisations des RCSFs dans l'industrie comme le suivie des véhicules, le contrôle et l'automatisation des processus dans une usine, la commande de robot et l'orientation dans des environnements de fabrication automatiques, etc.



FIGURE 2.7 – Exemple d'applications industrielles

## 2.7 Applications commerciales

Certaines applications commerciales utilisent les RCSFs pour la la construction de bureaux intelligents, la surveillance de la qualité des produits et la sur-utilisation des matériaux, etc.

La lecture automatique des compteurs sans fil (AMR) est l'un des marchés les plus prometteux pour les capteurs sans fil. La collecte de données sans fil depuis des

compteurs d'électricité, d'eau ou de gaz est un moyen très rentable, permettant de recueillir des données de consommation pour le système de facturation [42].

## 2.8 Conclusion

Nous avons essayé à travers ce chapitre de mettre le point sur les principaux domaines d'application des RCSFs. En effet, Les réseaux de capteurs sans fil présentent un intérêt considérable et une nouvelle étape dans l'évolution des technologies de l'information et de la communication. La flexibilité, la tolérance aux fautes, le prix réduit et les caractéristiques rapides de déploiement de ces réseaux de capteurs offrent des possibilités infinies de développement dans tous les domaines d'application. Ceci nous permet de penser que les réseaux de capteurs seront bientôt une partie intégrante de notre vie et satisferont sûrement les besoins des plus grands projets. Dans le chapitre suivant, nous allons aborder le concept des défi de conception des RCSFs, qui constitue l'un des contraintes majeures qui confrontent le bon fonctionnement de ces derniers.

# Défis et challenges dans les RCSFs

---

## 3.1 Introduction

La conception des réseaux de capteurs sans fil nécessite une maîtrise d'un large nombre de domaines de recherche, y compris la communication sans fil et les protocoles de communication, les systèmes embarqués, le traitement numérique du signal, le génie logiciel et le fonctionnement des réseaux distribués. Ceci est due à la fusion entre plusieurs entités matérielles et logicielles dans ces dispositifs de détection. Par conséquent, plusieurs facteurs influent significativement sur la conception des réseaux de capteurs sans fil. Dans ce chapitre, les principaux facteurs sont décrits, y compris les contraintes matérielles, la tolérance aux pannes, l'évolutivité, le coût de production, la topologie des réseaux de capteurs, les dispositifs de transmission et la consommation d'énergie. Ces facteurs ont été abordés par de nombreux chercheurs dans une large classe de domaines concernant la conception et le déploiement des réseaux de capteurs sans fil. En outre, l'intégration des solutions à ces facteurs est encore un défi majeur en raison de la nature multi-disciplinaire de ce domaine.

## 3.2 Contraintes matérielles

L'architecture générale et les principaux composants d'un noeud capteur sans fil sont illustrées dans la figure 3.1. Un capteur sans fil est généralement composé de quatre éléments de base : une unité de détection, une unité de traitement, une unité d'émetteur-récepteur (radio) et une unité de puissance (batterie). En outre, des composants supplémentaires peuvent également être intégrés dans le noeud en fonction de l'application. Ces composants, comme indiqué par la figure 3.1 comprennent : un

système de localisation , un générateur de puissance et un mobilisateur [2, 43].

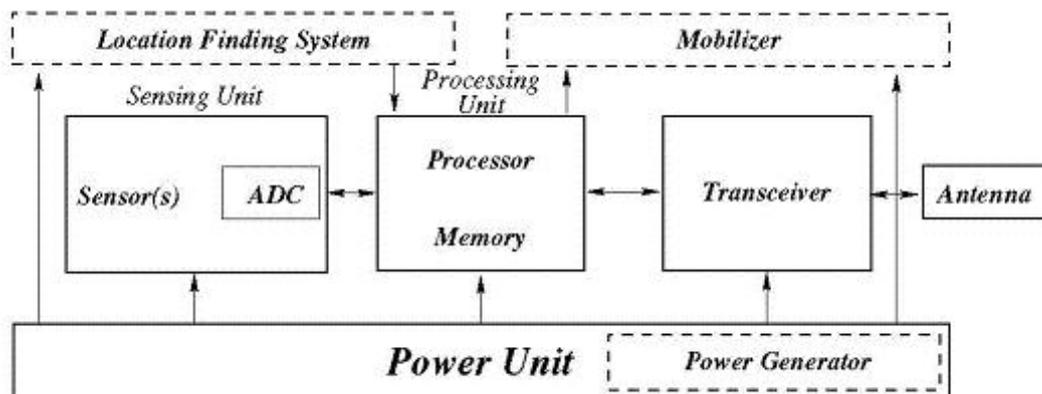


FIGURE 3.1 – Architecture matérielle générale d'un nœud de capteur

Ces composants devraient également être de petite taille [44]. Pour certaines applications, la taille réelle d'un capteur doit être plus petit qu'un centimètre cube [45], ainsi que le poids qui doit être à l'ordre d'un gramme. Ces fonctionnalités sophistiquées et les exigences de taille placent des contraintes supplémentaires sur la conception des nœuds capteurs sans fil. En plus de la taille, il y a aussi quelques autres contraintes strictes sur les nœuds capteurs ; Ces nœuds doivent : consommer une puissance extrêmement faible, opérer en haute densité, avoir un faible coût de production, être autonome et doivent fonctionner sans surveillance, ainsi ces capteurs doivent avoir une capacité d'adaptation à l'environnement [33].

### 3.3 Tolérance aux pannes

Les contraintes matérielles conduisent, souvent, les nœuds capteurs à l'échec ou au blocage après un certain temps de fonctionnement. Ces défauts peuvent se produire suite au manque de puissance, dommages physiques, interférences due à l'environnement ou aux problèmes logiciel. En effet, la défaillance d'un ensemble de nœuds se traduit par la déconnexion du réseau ; et comme un RCSF est intéressé

par des informations concernant le phénomène physique à la place de l'information à partir d'un capteur unique, la défaillance d'un seul noeud ne devrait pas affecter le fonctionnement global du réseau. Le niveau d'échecs qui permis au réseau de poursuivre adéquatement ses fonctions est défini par la tolérance aux pannes.

Plus précisément, la tolérance aux pannes est la capacité à maintenir les fonctionnalités des réseaux de capteurs sans interruptions due à des défaillances de noeuds capteurs [46, 47]. En premier lieu, les composants matériels et logiciels d'un noeud augmentent le taux d'échec. En effet, les noeuds capteurs sont des dispositifs à faible coût, ce qui fait que la majorité des pannes sont causées par des problèmes matériels. En outre, en raison de la mémoire et des capacités de traitement limitées, le logiciel peut entraîner l'arrêt ou le blocage d'un noeud, ceci est en plus des problèmes due par l'environnement et de déploiement qui peuvent également affecter la façon dont fonctionne un capteur. Les applications intérieures (par exemple à l'intérieur d'un bâtiment) produisent moins d'interférences entre les noeuds capteurs et peuvent ne pas augmenter le taux d'échec. D'autre part, dans les applications où les noeuds capteurs sont déployés à l'extérieur, les défaillances des noeuds capteurs se produisent aussi en raison de l'interférence avec l'environnement.

### 3.4 évolutivité

Le déploiement dense des noeuds capteurs dans un RCSF fournit une redondance de données et améliore la tolérance aux pannes du réseau, mais crée, également, des défis d'évolutivité. Le nombre de noeuds de capteurs déployés pour la détection d'un phénomène physique peut être de l'ordre de centaines ou de milliers de noeuds. Par conséquent, les protocoles développés pour ces réseaux doivent être capable de gérer un grand nombre de noeuds efficacement suite au déploiement aléatoire. La densité peut être à l'ordre d'une centaines de noeuds capteurs dans une région, qui peut être inférieure à 10 *m* de diamètre [48]. La densité des noeuds dépend de l'application

pour laquelle les noeuds capteurs sont déployés.

### 3.5 Coût de production

Étant donné que les réseaux de capteurs sont constitués, généralement, d'un grand nombre de noeuds capteurs, le coût d'un noeud unique est très importante, car il détermine le coût de l'ensemble du réseau. Si le coût d'un réseau de capteurs sans fil est plus cher que le déploiement des capteurs simples traditionnels, alors l'intérêt des réseaux de capteurs sans fil ne sera pas justifié. Par conséquent, le coût de chaque noeud capteur doit être maintenu faible. Dans nos jours, le coût d'un capteur utilisant le Bluetooth est généralement moins de 10 \$ [2]. Mais le coût d'un noeud capteur devrait être inférieur à 1 \$ pour les réseaux de capteurs à grande densité, afin d'être réalisable en pratique [40]. Les prix actuels des dispositifs de détection sont beaucoup plus élevé que même pour un Bluetooth. En outre, un noeud capteur peut aussi avoir des unités supplémentaires, par exemple, pour la détection et le traitement. En outre, il peut être équipé d'un système de localisation, un mobilisateur, ou un générateur de puissance en fonction des applications des réseaux de capteurs. Ces unités augmentent le coût de ces dispositifs de détection. En conséquence, le coût d'un noeud de capteur est un facteur très important ; étant donné qu'un RCSF peut contenir des millions de capteurs.

### 3.6 La topologie utilisée

Le grand nombre de noeuds capteurs inaccessibles et sans surveillance, et qui sont l'objet à de fréquentes pannes, font de l'entretien de la topologie utilisée une tâche difficile. Le défi majeur est le déploiement de ces noeuds capteurs dans un champ de tel sorte que le phénomène à observer doit être contrôlée efficacement. Ceci doit être défini dans la phase de pré-déploiement et dans la phase de déploiement.

L'entretien de la topologie est également important après le déploiement initial,

c-à-d la phase après-déploiement, où les paramètres et les protocoles doivent être adoptées selon la topologie du réseau. Enfin, une phase de re-déploiement peut être nécessaire si plusieurs noeuds échouent ou épuisent leurs énergie, ceci pour prolonger la durée de vie du réseau. Globalement, le déploiement d'un nombre élevé de noeuds nécessite une manipulation minutieuse de l'entretien de la topologie. Nous décrivons les notions suivantes liées à l'entretien de la topologie dans les trois phases suivantes :

### 3.6.1 La phase de pré-déploiement et de déploiement

Les noeuds capteurs peuvent être déployés soit en masse ou placés manuellement un par un dans un champ de surveillance. En effet, Ils peuvent être déployés par un hélicoptère, une lance roquettes, un missiles ou par un placement manuelle (un par un) soit par un humain ou un robot. les défis de déploiement sont :

1. Réduire le coût de l'installation ;
2. Éliminer la nécessité pour toute pré-organisation et une planification préalable ;
3. Accroître la flexibilité d'arrangement ;
4. Promouvoir l'auto-organisation et la tolérance aux pannes.

### 3.6.2 La phase après déploiement

Après la phase de déploiement, la topologie peut varier en raison des changements dans le réseau [49]. Dans un RCSF mobile, les mouvements des capteurs influent sur la topologie du réseau. Par conséquent, des importants changements peuvent se produire dans la topologie d'un instant à l'autre. En outre, la connectivité des noeuds peut changer en raison de brouillage, des interférences, de bruit ou des obstacles mobiles. En effet, ces facteurs affectent la topologie du réseau pendant un court instant. Une autre cause de changement de topologie après le déploiement est due aux défaillances des noeuds, qui se traduisent par des changements permanents. Finalement, la topologie du réseau, peut être modifié périodiquement selon les tâches

de détection et à la demande lorsque certains noeuds doivent être désactivé pour un laps de temps particulier.

Ces modifications se traduisent par un fonctionnement différent et inadéquat de celui du déploiement initial du réseau. En conséquence, les protocoles de réseau doivent être capables de s'adapter à ces changements à court terme, périodiques, et à long terme dans la topologie.

### **3.6.3 Phase de re-déploiement des noeuds supplémentaires**

Les changements de phase de post-déploiement peuvent nécessiter le déploiement des noeuds supplémentaires si la connectivité et la tolérance aux pannes du réseau sont gravement affectée par les changements de la topologie. En conséquence, des noeuds capteurs supplémentaires peuvent être redéployés à tout moment pour remplacer les noeuds avec un mauvais fonctionnement, exemple des noeuds avec des batteries épuisées ou en raison de changements dans la dynamicité des tâches. L'ajout de nouveaux noeuds représente une nécessité de réorganisation du réseau. Les changements, fréquents, de la topologie dans un réseau ad-hoc qui a des myriades de noeuds donne la naissance à des contraintes très strictes sur la consommation d'énergie, ceci nécessite aussi des protocoles de routage spéciaux.

## **3.7 Supports de transmission**

Le bon fonctionnement d'un RCSF repose sur une communication fiable entre les noeuds d'un réseau. Dans un réseau de capteurs à multi-sauts, les noeuds peuvent communiquer, entre eux, à travers des liens sans fil. Ces liens peuvent être des signaux radio, infrarouge, optique, acoustique ou magnéto-inductif. Pour permettre l'interopérabilité et le fonctionnement global de ces réseaux, le support de transmission choisi doit être disponible dans le monde entier. En plus le support de transmission doit être utilisé efficacement pour minimiser le gaspillage d'énergie.

### 3.8 Consommation d'énergie

Un noeud capteur sans fil ne peut être équipé que par une source de puissance limitée ( $<0.5\text{Ah}$ ,  $1,2\text{ V}$ ) en raison de plusieurs contraintes matérielles. De plus, pour la plupart des applications, le ré-apvisionnement des ressources énergétiques est impossible. Donc, un RCSF doit faire preuve d'une forte dépendance à la batterie. Ainsi, les sources qui consomment de l'énergie pendant le fonctionnement de chaque noeud doivent être analysés et utilisés de manière efficace.

Dans un réseau, Ad-hoc, de capteurs à multi-sauts, chaque noeud joue deux rôles distincts et complémentaires :

- **Source des données** : Le rôle principal de chaque noeud capteur est de recueillir des données provenant de l'environnement par le biais de ses différents capteurs. Les données générées à partir de la détection de l'environnement doivent être traitées et transmises à un noeud capteur à proximité (voisin), et ainsi de suite, jusqu'à la livraison de ces données au Sink par l'intermédiaire du technique multi-sauts.
- **Routeur de données** : En plus de ces propre données, chaque noeud est chargé de transmettre les informations reçus de ses voisins. Les techniques de communication à faible puissance dans les réseaux de capteurs limite la portée de communication d'un noeud. Dans un large réseau, la communication multi-sauts est nécessaire pour transmettre les données depuis un noeuds source vers un noeud destination, qui est généralement le Sink. Par conséquent, un noeud capteur est chargé de recevoir les données envoyées par ses voisins et transmettre ces données à l'un de ses autres voisins selon les décisions de routage.

Les opérations liées à chaque rôle détermine la quantité d'énergie consommée dans un noeud capteur. En outre, l'échec de quelques noeuds peut causer des changements topologiques importants et exige un ré-acheminement de paquets et de la

réorganisation du réseau. Par conséquent, la conservation d'énergie et la gestion de batterie sont une partie intégrante de tout protocole de communication dans les réseaux de capteurs sans fil. Par conséquent, la conception des protocoles et des algorithmes énergie-conscient (*Power-Aware*) pour les réseaux de capteurs sont d'une grande importance.

La tâche principale d'un noeud capteur dans un champ de détection est de détecter les événements, effectuer un traitement de données local puis transmettre ces données. La consommation électrique peut donc être divisé en trois domaines : la détection, le traitement et la communication des données, qui sont effectuées, respectivement, par le capteur, le CPU et la radio. La répartition de la consommation d'énergie d'un noeud capteur *MicaZ* est représentée sur la figure 3.2. On peut voir que, parmi ces trois consommations, un noeud capteur dépense de l'énergie maximale pour la communication des données. Les trois sources de consommation d'énergie sont discutés dans les sections suivantes.

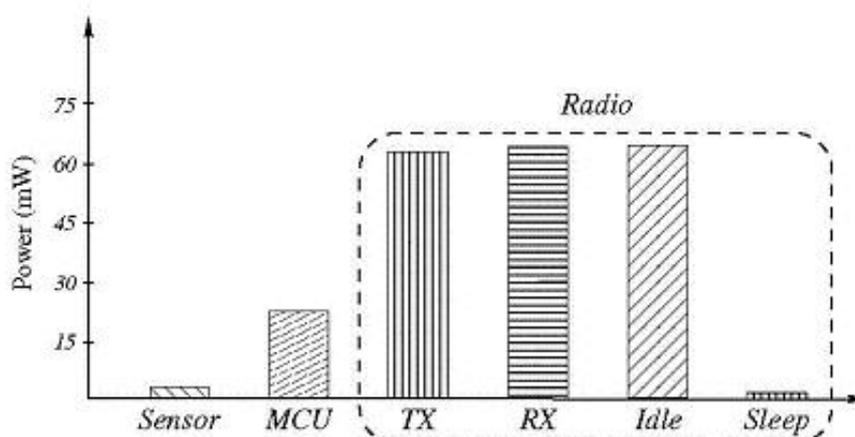


FIGURE 3.2 – La répartition de la consommation d'énergie d'un noeud MicaZ

### 3.8.1 La capture (Sensing)

L'unité de détection et ses composants ont été présentés plus haut dans cette thèse. la puissance de capture varie avec la nature des demandes et des capteurs spécifiques utilisés. La détection discrète peut consommer moins d'énergie que la surveillance permanente des événements. La complexité de la détection d'événements( en cas d'existence de chaos) joue également un rôle crucial dans la détermination de la consommation d'énergie. Les niveaux de bruit plus élevés pourraient entraîner à la corruption des signaux et l'augmentation significative de la complexité de détection.

### 3.8.2 Traitement des données

L'énergie consommée suite au traitement des données est similaire à celle de l'opération de détection. Cependant, le traitement nécessite beaucoup moins d'énergie par rapport à la communication de données comme le montre la figure 3.2. Par exemple, en prenant le capteur *Rayleigh*, le coût en terme d'énergie de la transmission d'un paquet de 1 *kb* sur une distance de 100 *m* est approximativement égal à l'exécution de 3.000.000 d'instructions par un microprocesseur typique [45]. Cette différence remarquable entre la communication et le calcul , en terme de consommation d'énergie, montre l'importance de traitement de données en local, comme la fusion et l'élimination de redondance, qui permis de minimiser la consommation d'énergie dans un réseau de capteurs à multi-sauts.

### 3.8.3 Communication

Parmi ses trois états (capture, communication et calcul), un noeud capteur dépense le maximum d'énergie dans la communication de données, pendant la réception et la transmission, comme le montre à la figure 3.2. La communication est effectué par l'antenne (émetteur-récepteur) pendant la réception et la transmission de données. La répartition de la consommation d'énergie par l'émetteur-récepteur représentée dans la figure 3.2 montre qu'un émetteur-récepteur dépense une quantité

similaire d'énergie pour émettre ou pour recevoir des données, ainsi que lorsqu'il est inactif (*Over-hearing*). En outre, une quantité importante d'énergie peut être sauvée en désactivant l'émetteur-récepteur en le posant à l'état de veille chaque fois que le noeud capteur n'a pas besoin d'émettre ou de recevoir des données. Cette technique permet d'économiser l'énergie de la batterie jusqu'à 97% (de 59.1 mW to 3  $\mu$ W) [50].

Le circuit émetteur-récepteur (antenne) est constitué d'un mélangeur, un synthétiseur de fréquence, d'un oscillateur commandé en tension (*VCO*), boucle à phase asservie (*PLL*), démodulateurs et amplificateurs de puissance; tous ces composants sont connus par leurs consommations d'énergie. La consommation d'énergie durant la communication de données par un émetteur-récepteur, est une combinaison de l'énergie consommée par ces composants et peut simplement être modélisée par :

$$P_c = P_o + P_{tx} + P_{rx} \quad (3.1)$$

Où  $P_o$  est la puissance d'émission et  $P_{tx}$  et  $P_{rx}$  sont, respectivement, l'énergie consommée durant la transmission et la réception. En d'autres termes, les deux premiers termes de la formule 3.1 modélisent la consommation de l'énergie pendant la transmission, tandis que le dernier terme représente la consommation de l'énergie pendant la réception.

La principale différence entre la transmission et la réception est l'utilisation d'amplificateurs de puissance ( $P_0$ ) pour la transmission de données. Toutefois, durant une communication à courte portée avec une puissance de rayonnement faible (-0 dBm), les coûts de transmission et de réception d'énergie sont presque les mêmes. En outre, avec le développement des systèmes de modulation (le décodage et le codage) l'énergie consommée durant la réception est supérieure à la puissance d'émission (comme exemple les noeuds de *MicaZ*). L'énergie de transmission consommée peut changer en fonction du niveau de puissance d'émission fixée par le noeud, qui peut aller de 0 jusqu'à -25 dBm. Par conséquent, si le niveau de l'énergie d'émission est

réglée à des valeurs très faibles, alors la puissance de réception domine la communication de données dans les réseaux de capteurs.

### 3.9 Conclusion

Dans ce chapitre, vous avons présenté une vue générale sur les grands facteurs qui influent pendant le déploiement d'un RCSF. Ces facteurs doivent être pris en considération pour avoir un réseaux robuste. La consommation d'énergie constitue l'une des plus importants facteurs dans les RCSFs, suite aux batterie non remplaçable et non rechargeable. donc le besoin de trouver des mécanismes pour minimiser la consommation d'énergie. Le chapitre suivant sera consacré à l'économie d'énergie dans les RCSFs.



# Le facteur énergie dans les RCSFs

---

## 4.1 Introduction

Dans les chapitres précédents, nous avons présenté les RCSFs ainsi que les besoins et les challenges pour les concevoir, en prenant en considération la limitation des ressources des noeuds capteurs. En conséquence, on peut considérer la mauvaise gestion des ressources comme une menace qui peut entraver le bon fonctionnement du réseau.

Étant donné que l'énergie est la ressource la plus précieuse dans les RCSFs, les protocoles de communication dédiés pour l'économie d'énergie représentent l'axe de recherche le plus important dans le domaine des RCSFs. En effet, les noeuds capteurs sont alimentés par de petites batteries à faible capacité, et le changement ou le rechargement de ces dernières est une tâche difficile (en raison de très grand nombre de noeuds capteurs déployés) voire même impossible (en raison de déploiement dans des zones hostiles). Ainsi, l'épuisement de l'énergie des noeuds capteurs implique, le plus souvent, la mise hors service du réseau tout entier. Par conséquent, les protocoles de communication dédiés aux RCSFs doivent prendre en compte le degré de consommation d'énergie afin d'offrir une gestion efficace de cette ressource très précieuse.

De notre point de vue, offrir des mécanismes de communication, ayant comme objectif une consommation minimale d'énergie, revient à proposer des protocoles de communication avec une gestion efficace de ressources. Dans ce chapitre, nous allons aborder le problème de l'économie d'énergie, nous commençons par mettre le point sur les formes de consommation et les sources de gaspillage d'énergie.

## 4.2 La consommation d'énergie par un noeud capteur

Les noeuds capteurs sont généralement destinés à surveiller l'environnement, collecter des informations, traiter et transmettre ces informations à un point central dans le réseau ( appelé *Sink* ou *Station de Base*). Ainsi, on peut distinguer plusieurs sources de consommation d'énergie à savoir : la collecte, le traitement et la transmission des données [51].

### 4.2.1 Énergie consommée suite à la collecte des données

la collecte de données se fait, généralement, par un ou plusieurs capteurs. Par conséquent, la consommation d'énergie durant cette phase peut être le résultat de l'échantillonnage et la conversion des signaux physiques en signaux électriques. Étant donné la diversité des capteurs, il n'y a pas de valeurs typiques de l'énergie consommée. Cependant, le niveau de consommation est en général très faible comparé aux autres sources de consommation d'énergie [1].

### 4.2.2 Énergie consommée suite au traitement des données

L'énergie nécessaire du traitement est l'énergie consommée par le processeur afin d'effectuer les calculs nécessaires sur les données collectées. Dans le but de réduire la consommation d'énergie, les processeurs des noeuds capteurs doivent basculer entre l'état actif, l'état d'écoute (*idle*) et l'état de sommeil dans le cas où il n'y a pas de données à traiter. En effet, ceci a prouvé son efficacité car la consommation de l'énergie dans ces deux derniers états est très réduite par rapport à l'état actif. Le degré de consommation d'énergie varie d'un processeur à un autre ; par exemple le processeur MSP430 (embarqué dans les capteurs *TmoteSky*) consomme 3 mW en mode actif, 98  $\mu$ W mode écoute et seulement 15  $\mu$ W en mode sommeil [9].

### 4.2.3 Énergie consommée suite à la transmission des données

la transmission des données (*Emission/réception*) est l'activité qui consomme plus d'énergie, étant donné que celle-ci est basée sur l'utilisation des antennes radio qui consomment jusqu'à 90% de la totalité de la batterie, durant le cycle de vie d'un noeud [1, 2, 52]. Afin de gérer efficacement sa consommation énergétique, l'antenne radio possède plusieurs modes de fonctionnement. Ainsi, pour réduire la dissipation d'énergie, l'antenne radio peut se fixer en mode sommeil, écoute, réception ou transmission selon les besoins. En outre, le passage d'un mode à un l'autre consomme aussi de l'énergie, puisque cela sollicitera une activité importante dans les circuits électroniques [9].

## 4.3 Les sources de gaspillage d'énergie des les RCSFs

La surconsommation de l'énergie est définie par la consommation d'énergie au-delà du seuil normal. Celle-ci peut être causée par différents phénomènes qui augmentent le niveau de consommation d'énergie des noeuds capteurs.

### 4.3.1 L'écoute passive (idle listening)

les noeuds capteurs changent périodiquement d'état (entre *actif*, *écoute* et *sommeil*), afin de détecter la présence d'un trafic dans le réseau, Cette mise en écoute permet de savoir s'il y'a des données à recevoir ou à router dans le réseau. Cependant, cette mise en écoute peut être très coûteuse dans le cas d'un réseau à faible trafic. En effet, le noeud capteur peut se trouver à l'état écoute (écouter un éventuel trafic) dans le cas d'absence de réception ou de transmission de données, entraînant, ainsi, un gaspillage inutile de ses réserves d'énergie. De plus, la transition périodique entre le sommeil et l'écoute causera une autre source de surconsommation d'énergie, surtout dans le cas où celle-ci n'est pas contrôlée.

### 4.3.2 Les collisions

à cause de l'environnement de communication sans fil, les réseaux de capteurs sont fortement exposés aux interférences et aux collisions. Ces dernières sont générées lorsque deux ou plusieurs noeuds adjacents transmettent leurs données en même temps. En effet, les collisions sont considérées comme étant une source de gaspillage d'énergie la plus importante, étant donné que celles-ci provoquent la retransmission des paquets, ce qui est très coûteux en terme d'énergie [51].

### 4.3.3 La puissance de transmission

la portée des antennes radio est directement liée à la puissance de transmission utilisée. La plupart des noeuds capteurs possèdent des antennes à portée statique dans lesquels la puissance de transmission est fixée précédemment par les concepteurs. Par conséquent, un noeud capteur peut gaspiller pas mal d'énergie en utilisant une grande puissance de transmission afin de communiquer avec un noeud très proche de lui. Pour remédier à ce problème, une autre famille de capteurs est commercialisée, celle-ci peut ajuster la puissance de sa transmission selon les besoins, ainsi un noeud voulant communiquer avec un noeud voisin, proche de lui, ne consommera que l'énergie due au transmission sur cette distance qui le sépare de son voisin [53].

### 4.3.4 Les distances de transmission

Afin d'économiser la consommation d'énergie, il est préférable de réduire les distances de transmission entre les noeuds capteurs. Ainsi, la communication multi-sauts est souvent utilisée dans les réseaux de capteurs sans fil, contrairement à celle basée sur un seul-saut dans laquelle la dissipation d'énergie est très élevée [54, 55].

### 4.3.5 L'écoute abusive (Overhearing)

Un noeud capteur peut recevoir toutes les données échangées entre les noeuds voisins, même si ces dernières ne sont pas destinées à lui. L'intensité de cette écoute abusive est proportionnelle à la densité du réseau. Ainsi, cela peut engendrer un grand gaspillage d'énergie, vu que la majorité des RCSFs sont déployés à grande échelle [51].

### 4.3.6 Le sur-coût des paquets de contrôle (Overhead)

Les paquets de contrôle sont les messages échangés dans un réseau pour la synchronisation de celui-ci. L'échange des paquets de contrôle peut être une autre source de gaspillage d'énergie, principalement si le nombre des paquets de contrôle est inutilement élevé [18, 56].

## 4.4 Protocoles d'économie d'énergie dans les RCSFs

Le routage dans les réseaux de capteurs est très difficile en raison de plusieurs caractéristiques qui les distinguent des réseaux sans fil Ad-hoc. Tout d'abord, il n'est pas possible d'utiliser un système d'adressage global à cause du déploiement d'un grand nombre de noeuds capteurs. Par conséquent, les protocoles classiques basés sur les adresses *IP* ne peuvent pas être appliqués aux réseaux de capteurs. Deuxièmement, contrairement aux réseaux de communication traditionnels, presque toutes les applications des réseaux de capteurs acheminent un flux de données détectés à partir de plusieurs régions (sources) vers un noeud particulier (Sink). Troisièmement, le trafic des données générées souffre d'une redondance importante puisque plusieurs capteurs peuvent générer les mêmes données dans le voisinage d'un phénomène. Cette redondance doit être exploitée par les protocoles de routage pour améliorer l'efficacité énergétique et l'utilisation de la bande passante. Quatrièmement, les noeuds capteurs sont largement limités en termes : de puissance de transmission,

d'énergie, de capacité de traitement et de stockage et donc nécessitent une gestion efficace des ressources.

L'économie d'énergie est considérée, toujours, comme un paramètre fondamental dans le contexte de robustesse des réseaux de capteurs sans fil. Malgré les progrès qui ont été réalisés, le prolongement de la durée de vie des noeuds capteurs est considéré toujours comme un défi majeur et un facteur clé, exigeant d'avantage des recherches sur l'efficacité énergétique et sur les protocoles de communication. Dans cette section, nous présenterons un état de l'art sur les recherches proposées dans le domaine d'économie d'énergie dans les RCSFs. En effet, la plus grande partie de ces recherches est réservée aux protocoles mono-couche, qui s'intéressent au problème de l'économie d'énergie au niveau d'une seule couche du modèle OSI [23]. Deux grandes familles de protocoles de communication dédiées à l'économie d'énergie pour les RCSFs sont présentées ci-après. La première famille est celle dédiée à la couche réseau où les protocoles s'intéressent à la construction des chemins de routage des données, la deuxième famille est dédiée à la couche liaison de données, elle opère dans la couche MAC (*Media Access Control*).

#### 4.4.1 Protocoles dédiés à la couche réseau

Les protocoles de routage destinés à la gestion efficace d'énergie, représentent la majorité des recherches proposées au niveau de la couche réseau. L'idée de base de cette catégorie de protocoles est la construction des chemins de routage à faible consommation d'énergie (*energy aware routing path*). De nombreuses stratégies de routage ont été proposées pour les réseaux de capteurs sans fil. Certaines sont des adaptations de stratégies qui existaient pour d'autres types de réseaux (principalement pour les réseaux sans fil au sens plus large), tandis que d'autres ont été conçues spécialement pour les réseaux de capteurs sans fil. Les algorithmes de routage, dédiés à la couche réseau, sont en fait classés en trois types d'algorithmes, voir figure 4.1 [57] :

1. les algorithmes de routage centrés sur données,
2. les algorithmes de routage géographiques,
3. les algorithmes de routage hiérarchiques.

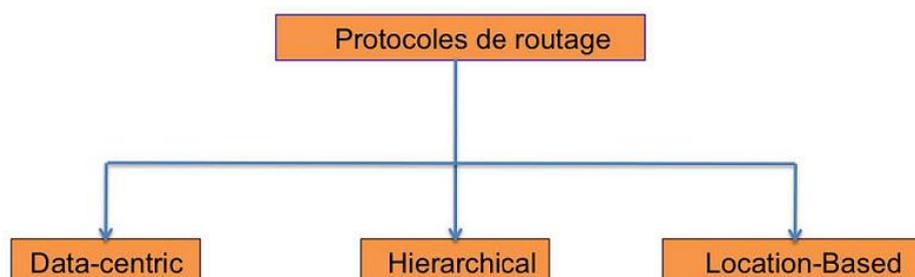


FIGURE 4.1 – Protocoles de routage dans les RCSFS

#### 4.4.1.1 Les protocoles de routage type centrés sur les données (*Data-centric protocols*)

Dans de nombreuses applications des réseaux de capteurs, vu le nombre élevé des noeuds déployés, il n'est pas possible d'attribuer des identificateurs globaux à chaque noeud. Cette absence d'identification globale avec le déploiement aléatoire de noeuds capteurs rendent difficile de sélectionner un ensemble spécifique de noeuds capteurs pour être interrogé. Par conséquent, les données sont généralement transmises de chaque noeud capteurs dans la région de déploiement avec une redondance importante. Cette réflexion a conduit au routage data-centric [58,59] qui est différent du routage traditionnel où les routes sont créées entre les noeuds adressables dans la couche réseau. Le destinataire, généralement le sink, envoie des requêtes à certaines régions et attend la réception des données provenant des capteurs situés dans les régions sélectionnées. Comme les données sont demandées à travers des requêtes, le nommage est nécessaire pour préciser les propriétés des données [57,59].

le routage centré sur les données est le modèle le plus simple où la station de base envoie des requêtes à certaines régions et attend les réponses des capteurs situés

dans les régions choisies. Puisque les données sont demandées par les requêtes, un adressage basé sur les attributs est utilisé pour préciser les propriétés des données. chaque noeud dans le réseau transmet ses données à la station de base. Les noeuds recevant les données redondantes applique la technique de l'agrégation et les envoient à leurs voisins, comme le montre l'exemple d'une approche data-centric dans la figure 4.2, les données provenant des deux sources sont agrégées au noeud B. Ensuite, la donnée combinée (1+2) est envoyée de B vers la destination. Plusieurs algorithmes centrés sur les données ont été développés, on peut citer : le protocole de propagation et de discussion (*flooding and gossiping*) présenté dans [60], le protocole de routage par négociation *SPIN* [56] et le protocole de routage par diffusion dirigée [44]. Par la suite nous détaillons l'algorithme *SPIN*, qui est le premier protocole centré sur les données, qui utilise la technique de négociation de données entre les noeuds afin d'éliminer les données redondantes et d'économiser l'énergie.

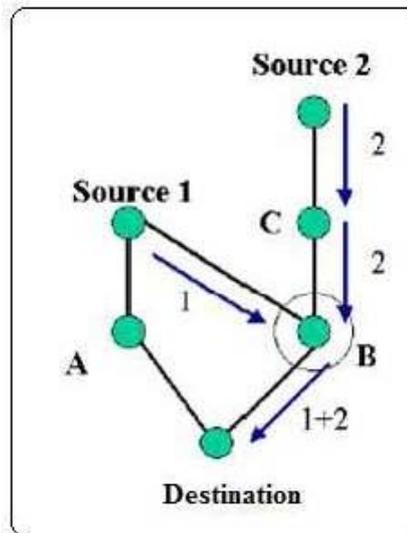


FIGURE 4.2 – Le routage data-centric

**Sensor Protocols for Information via Negotiation (SPIN) :**

Les auteurs dans [56] ont proposé une famille de protocoles appelée SPIN (*Sensor Protocols for Information via Negotiation*), qui repose sur un modèle de négociation afin de propager l'information dans un réseau de capteurs. Le but de SPIN est de pallier aux problèmes de l'inondation, qui sont :

1. L'implosion due à la duplication inutile des réceptions d'un même message.
2. Le chevauchement lié au déploiement dense des capteurs. En utilisant l'inondation, les capteurs d'une zone émettent tous la même donnée (ou presque).
3. L'ignorance des ressources, car l'inondation ne prend pas en considération les ressources des noeuds.

Ces trois problèmes affectent grandement la durée de vie et les performances du réseau. Pour les résoudre, SPIN adopte deux principes :

1. La négociation : pour éviter le problème d'implosion, SPIN précède l'émission d'une donnée par l'émission de sa description, en utilisant la notion de méta-données. Le récepteur aura le choix par la suite d'accepter la donnée ou non. Ce mécanisme permet aussi de régler le problème de chevauchement.
2. L'adaptation aux ressources : d'une manière continue, les noeuds contrôlent leur niveau d'énergie. Le protocole SPIN accommode son exécution suivant l'énergie restante du capteur, et modifie en conséquence le comportement du noeud.

Les communications dans SPIN se font en trois étapes :

1. Lorsqu'un noeud veut émettre une donnée, il émet d'abord un message ADV contenant une description de la donnée en question.
2. Un noeud recevant un message ADV, consulte sa base de données. S'il est intéressé par cette information, il émet un message REQ vers son voisin.

3. En recevant un message REQ, l'émetteur transmet à l'intéressé la donnée sous forme d'un message DATA.

La figure 4.3 illustre ces trois étapes :

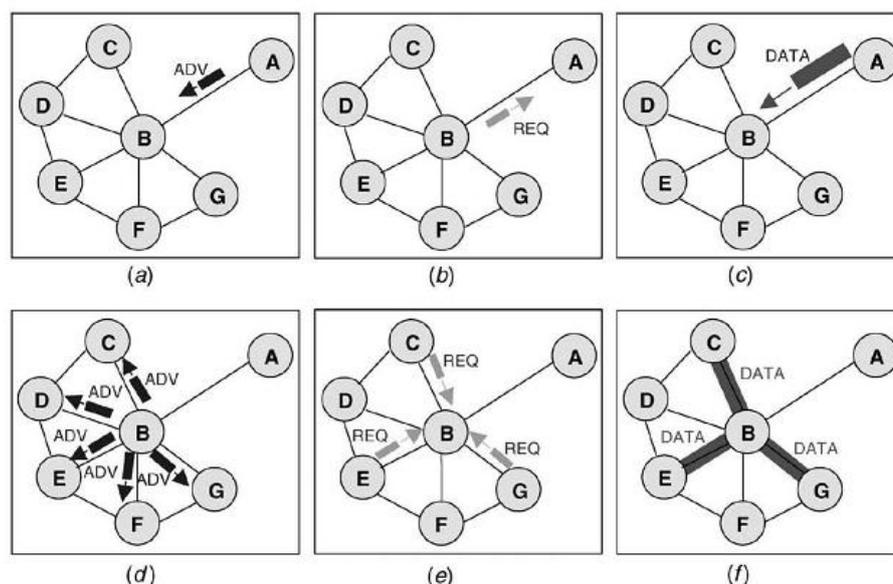


FIGURE 4.3 – Fonctionnement du protocole SPIN. Un nœud A commence par annoncer ses données vers le nœud B (a). Nœud B répond en envoyant une demande au nœud A (b). Après avoir reçu les données demandées (c), le nœud B envoie ensuite des annonces pour ses voisins (d), qui à leur tour envoient des requêtes vers B (e-f).

#### 4.4.1.2 Les protocoles de routage basés sur la localisation géographique

Dans ce type de routage les nœuds capteurs sont adressés en fonction de leurs localisations. La distance entre les nœuds voisins peut être estimée sur la base de l'intensité du signal [57, 59]. Des coordonnées, relatives, des nœuds voisins peuvent être obtenues en échangeant une telle information entre les voisins. Alternativement, la location des nœuds peut être disponible directement en communiquant avec un satellite en utilisant le *GPS* (système de positionnement global).

Dans la plupart des protocoles de routage géographique, l'information sur la localisation des nœuds est nécessaire afin de calculer la distance entre deux nœuds particuliers de telle sorte que la consommation d'énergie puisse être estimée. Par

conséquent, le coût de contrôle de l'algorithme est réduit et le routage est optimisé. De plus, avec la topologie réseau basée sur des informations de localisation des noeuds, la gestion du réseau devient simple. En effet, le routage géographique suppose que tous les noeuds connaissent leurs positions. Néanmoins, une solution basée sur le GPS peut être trop coûteuse, d'autant plus que le nombre de noeuds à équiper, par un module GPS, est très grand. Parmi ces protocoles géographiques, on peut mentionner les protocoles MECN [61], GAF [62] et GEAR [57]. Dans la section suivante nous présenterons l'algorithme *MECN*.

#### **Le protocole de routage (*MECN*) :**

*Minimum Energy Communication Network (MECN)* [59,63] est un protocole de routage qui cherche à établir et à entretenir une énergie minimale pour les réseaux de capteurs sans fil en utilisant des GPS de faible puissance. MECN utilise une station de base comme destination de l'information, ce qui est toujours le cas pour les réseaux de capteurs. MECN identifie une région de relais pour chaque noeud. La région de relais se compose de noeuds dans une zone périphérique où la transmission à travers ces noeuds est plus économe en énergie que la transmission directe. L'idée principale de MECN est de trouver un sous-réseau qui a moins de noeuds et qui nécessite moins d'énergie pour la transmission entre deux noeuds quelconques. Cela est effectué en utilisant une recherche localisée par chaque noeud en prenant en considération sa région de relais [57,59].

#### **4.4.1.3 Les protocoles hiérarchiques**

L'objectif principal du routage hiérarchique [57,59] est de maintenir efficacement la consommation d'énergie de noeuds de capteurs en les impliquant dans la communication multi-sauts au sein d'un cluster et en effectuant l'agrégation et la fusion des données afin de diminuer le nombre de messages transmis à la destination. La formation des clusters est, généralement, fondée sur la réserve d'énergie des capteurs

et sur les capteurs qui sont à proximité du cluster-chef.

le routage hiérarchique est considéré comme étant l'approche la plus favorable en termes d'efficacité énergétique [57,61,64]. Il se base sur le concept de noeud standard et noeud maître où les noeuds standards acheminent leurs messages à leur maître, ensuite ce dernier achemine les messages vers le Sink via d'autres noeuds maîtres comme le montre la figure 4.4.

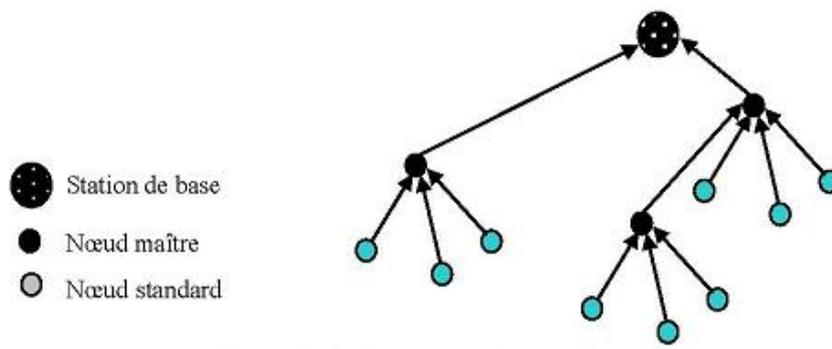


FIGURE 4.4 – Le routage hiérarchique

Le point fort de ce type de protocoles est l'agrégation et la fusion des données [65] afin de diminuer le nombre de messages transmis au sink, ce qui implique une meilleure économie d'énergie. Deux grandes approches sont dérivées de ce type de protocoles à savoir :

- L'approche à clusters (cluster-based approach).
- L'approche chaînée (chaine-based approach).

### L'approche à clusters (Cluster-based approach)

Elle consiste, de façon similaire aux réseaux téléphoniques cellulaires, à partitionner le réseau en groupes (*clusters*) où dans chacun d'entre eux, un seul capteur est sélectionné comme leader (*Cluster-chef*) pour jouer le rôle spécial de point de transfert et les autres noeuds sont considérés comme des cluster-membres. Ainsi,

chaque cluster-chef créera un plan de transmission pour tous les capteurs de son cluster, ce qui permet aux antennes radio de chaque cluster-membre d'être éteintes périodiquement, excepté pendant le temps de transmission. Les noeuds dont l'énergie est la plus élevée peuvent être employés dans le traitement et l'envoi d'informations tandis que ceux avec une faible énergie peuvent être employés uniquement dans la collection de données [66,67].

L'agrégation des capteurs en clusters permet de réduire la complexité des algorithmes de routage, d'optimiser l'utilisation de la ressource médium en la faisant gérer localement par un cluster-chef, de faciliter l'agrégation des données, de simplifier la gestion du réseau en particulier l'affectation d'adresses, d'optimiser les dépenses d'énergie et enfin de rendre le réseau plus évolutif (*scalable*). L'utilisation des clusters permet aussi aux noeuds d'effectuer des communications avec leurs cluster-chefs, sur des petites distances [68,69].

La rotation des cluster-chefs s'avère également un facteur important pour l'organisation des réseaux de capteurs hiérarchiques. Puisque la BS (*station de base ou sink*) est généralement loin du champ des capteurs, les cluster-chefs diffusent une quantité plus importante d'énergie pour la transmission de données à la BS. Par conséquent, les cluster-chefs mourront rapidement si le même noeud fonctionne continuellement comme un cluster-chef. Ainsi, pour ne pas épuiser la batterie d'un capteur simple, la plupart des algorithmes basés sur le clustering adoptent le concept de la rotation périodique des cluster-chef.

Plusieurs recherches ont été proposées afin d'améliorer la concept de clustering. Parmi ces recherches on peut citer les protocoles suivants : *LEACH* [70], *AROS* [71], *HCR* [68], *SPEAR* [72], *Energy-LEACH* et *MultiHop-LEACH* [73], *ECSA* [74], *E-LEACH* [75], *Improved-LEACH* [76], *STATIC-LEACH* [77], *Q-LEACH* [78] et *LEACH-TLCH* [69].

La notion de clustering est présenté avec plus de détaille dans le chapitre 5.

#### 4.4.2 L'approche à chaîne (Chain-based approach)

Dans cette approche, le principe du clustering est abandonné. Les noeuds du réseau sont organisés de façon à former une grande chaîne de proches voisins, dans laquelle un seul noeud est sélectionné pour transmettre au sink. En fait, l'idée de formation de chaîne a été proposée pour la première fois dans l'algorithme PEGASIS (*Power-Efficient Gathering in Sensor Information Systems*) [79].

Dans l'approche à chaîne, les noeuds vont être organisés de sorte qu'ils forment une chaîne, qui peut être soit calculée d'une manière centralisée par la BS et émise à tous les noeuds, ou accomplie par les noeuds capteurs eux-mêmes en employant un algorithme *avide* (*greedy algorithm*). Si la chaîne est calculée par les noeuds capteurs, ils doivent d'abord obtenir toutes les données sur l'emplacement des noeuds capteurs et calculent localement la chaîne en utilisant le même algorithme avide. Puisque tous les noeuds ont les mêmes données d'emplacement et exécutent le même algorithme, ils vont tous produire le même résultat [3, 80, 81].

#### **Le protocole PEGASIS (*Power-Efficient Gathering in Sensor Information Systems*) :**

Dans le but de prolonger la durée de vie du réseau, le protocole PEGASIS organise les noeuds de telle sorte à ce qu'ils forment une chaîne, et n'auront ainsi besoin de communiquer qu'avec leurs voisins les plus proches. En effet, PEGASIS a deux principaux objectifs, d'abord, augmenter la durée de vie de chaque noeud en employant les techniques de collaboration, ce qui permet d'augmenter la durée de vie du réseau. En second lieu, il permet la coordination locale entre les noeuds voisins de telle sorte que la largeur de bande consommée dans la communication soit réduite. PEGASIS évite la formation des clusters et n'utilise qu'un seul noeud dans une chaîne afin de transmettre au sink, au lieu d'en utiliser plusieurs. Ceci réduit l'énergie consommée pour transmettre des données, du moment que la consommation d'énergie est distribuée uniformément sur tous les noeuds. Pour localiser le

voisin le plus proche, chaque noeud utilise la force du signal pour mesurer la distance vers tous les noeuds voisins, et ajuster par la suite la force du signal de telle sorte qu'un noeud peut recevoir le signal.

Pour construire la chaîne, PEGASIS commence avec le noeud le plus éloigné de la BS. Le voisin le plus proche de ce noeud sera le noeud suivant dans la chaîne. Les voisins successifs sont sélectionnés parmi les noeuds non visités de cette manière, afin de former la chaîne de noeuds. La figure 4.5 montre le noeud  $c_0$  se reliant au noeud  $c_3$ , le noeud  $c_3$  se reliant au noeud  $c_1$ , et le noeud  $c_1$  se reliant au noeud  $c_2$ , dans cet ordre. Quand un noeud meure, la chaîne est reconstruite de la même manière pour dévier le noeud mort.

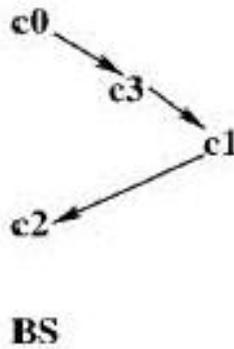


FIGURE 4.5 – Construction de chaînes en utilisant l'algorithme avide

Pour collecter des données des noeuds capteurs, chaque noeud reçoit les données d'un voisin, les fusionne avec les siennes, et les transmet à un autre voisin dans la chaîne. Il faut noter que ce noeud  $i$  serait à une certaine position aléatoire  $j$  sur la chaîne. Les noeuds se relient dans la transmission à la BS et PEGASIS va utiliser le  $\text{mod } N$  du numéro de noeud  $i$  ( $N$  représente le nombre des noeuds) afin de transmettre à la BS dans le cycle  $i$ . Ainsi, le leader dans chaque cycle de communication sera à une position aléatoire sur la chaîne. Chaque cycle de collecte de données peut être lancé par la BS avec un signal de balise qui synchronisera

tous les noeuds capteurs. Puisque tous les noeuds connaissent leurs positions sur la chaîne, PEGASIS peut employer une approche de slot de temps (*TDMA*) pour la transmission des données. Dans le  $i$ ème cycle de collecte de données, le noeud  $(i-1)$  sera leader.

PEGASIS exécute la fusion des données à chaque noeud excepté les noeuds de fin de chaîne. Chaque noeud va fusionner les données de ses voisins avec les siennes, afin de générer un paquet simple de la même longueur et les transmet par la suite à son autre voisin (S'il en a deux). Dans la formation de la chaîne, il est possible que certains noeuds puissent relativement avoir des voisins distants. De tels noeuds vont dissiper plus d'énergie dans chaque cycle, comparé à d'autres noeuds capteurs. Pour remédier à cela, les auteurs ont placé un seuil de distance entre les noeuds voisins, qui ne permet pas aux noeuds éloignés de devenir leaders.

Plusieurs travaux concernant l'approche à chaîne ont été publiés, nous citons une amélioration de *PEGASIS*, appelée *H-PEGASIS* (*PEGASIS* hiérarchique) [57], *CCS* [82] et *DS-PEGASIS* [83].

#### 4.4.3 Protocoles dédiés à la couche liaison de données (sous couche MAC)

Les sources de consommation d'énergie dans un noeud capteur sont le module radio, le microprocesseur et le capteur. La communication radio est considérée comme la source la plus consommatrice parmi les trois précédentes. Étant donné que la sous couche *MAC* est principalement concernée par l'utilisation du module radio, plusieurs recherches ont été proposées afin de gérer efficacement cette ressource et optimiser, ainsi, la consommation énergétique. En effet, un protocole MAC économise l'énergie en minimisant l'utilisation de la radio. L'utilisation inutile de la radio provient des 4 cas suivants [51] :

1. Le sur-écoute (*Overhearing*),
2. L collisions,

3. L'écoute passif (*Idle listening*),
4. Les messages de contrôle

Le moyen le plus efficace pour conserver l'énergie est de mettre la radio du noeud capteur en mode veille à chaque fois où la communication n'est pas nécessaire. Idéalement, la radio doit être éteinte dès qu'il n'y a plus de données à envoyer et/ou à recevoir, et devrait être prête dès qu'un nouveau paquet de données doit être envoyé ou reçu. Ainsi, les noeuds alternent entre périodes actives et sommeil en fonction de l'activité du réseau. Ce comportement est généralement appelé *Duty-cycling* ou *wake-up scheme*. Le mécanisme du Duty-cycling permet de réduire le temps de l'écoute passif (*Idle listening*), le sur-écoute (*overhearing*) et d'autres activités inutiles en mettant le noeud dans l'état de sommeil.

En effet, de nombreux protocoles *MAC* introduisent le mécanisme du Duty-cycling dans leurs conceptions afin de réaliser une faible consommation énergétique. Nous classons ces protocoles en deux grandes classes :

1. Les protocoles synchrones.
2. Les protocoles asynchrones.

Les protocoles basés sur le *Duty-cycling* synchrones sont typiquement équipés avec des calendriers d'activation (*wake-up schedules*) prédéterminées. Chaque période est divisée en une phase de sommeil et une phase d'activité. La synchronisation est effectuée par la transmission fréquente de petites trames de contrôle. Chaque noeud diffuse des trames d'activation une fois qu'il entre dans sa période d'activité. Ainsi il réveille tous ses noeuds voisins pour une éventuelle communication.

En ce qui concerne les protocoles basés sur le *Duty-cycling* asynchrones, on n'est pas obligé de synchroniser les horloges des noeuds capteurs. Dans ce cas, chaque noeud suit son plan d'activation afin de se mettre en état d'activité ou sommeil. Ainsi, le plan d'activation arrange les périodes d'activation des noeuds capteurs, de telle sorte à garantir que les noeuds émetteurs et récepteurs s'activent au même

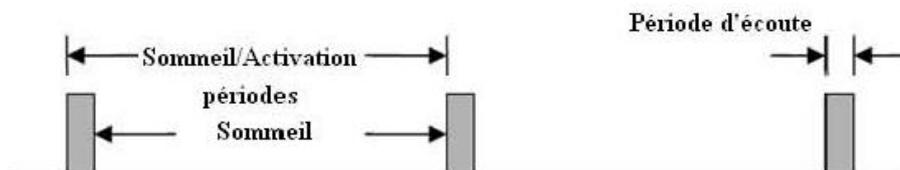


FIGURE 4.6 – le mécanisme du Duty-cycling

instant afin de communiquer leurs données.

Plusieurs protocoles *MAC* ont été proposés pour les réseaux de capteurs sans fil, nous citons : *Sensor-MAC* [18], *B-MAC* [19], *L-MAC* [84], *X-MAX* [85], *Wise-MAC* [86], *RI-MAC* [87] etc.

## 4.5 Conclusion

L'économie d'énergie constitue un défi très important pour la conception d'un RCSF robuste et sécurisé contre les défaillances de ressources. Vu que la disponibilité de ces derniers est directement liée à la durée de vie de leurs batteries, l'application des protocoles de communication à faible consommation d'énergie est plus que nécessaire. Les protocoles mono-couche ont prouvé leur efficacité en termes d'économie d'énergie au niveau de leurs couches respectives. Cependant, l'utilisation de l'un de ces protocoles au niveau de chaque couche de la plie protocolaire engendre une meilleure gestion d'énergie.

# Le clustering dans les RCSFs

---

## 5.1 Introduction

Dans les RCSFs, les protocoles dédiés au routage et à la collecte de données doivent consommer le minimum d'énergie, offrant, ainsi, une grande évolutivité et une prolongation de la durée de vie des réseaux à des niveaux acceptable. car, dans la plupart des applications des réseaux de capteurs sans fil, l'ensemble du réseau doit avoir la capacité de fonctionner sans surveillance dans des environnements où l'accès humain pour l'entretien ne peut pas être facilement programmé et efficacement géré et même parfois impossible [88]. En effet, dans de nombreuses applications importantes des RCSFs, les noeuds capteurs sont souvent déployés de façon aléatoire dans une zone de surveillance par des moyens relativement incontrôlés (exemple par un hélicoptère), cependant ces capteurs doivent former un réseau d'une manière Ad-hoc [89, 90].

En outre, d'autres facteurs doivent être pris en considération lors la formation d'un RCSF comme : la couverture entière de la zone de surveillance, la durée de vie limitée des batteries des capteurs, la possibilité d'avoir des noeuds endommagés lors du déploiement, en effet, un grand nombre de capteurs sont perdus, avec une grande possibilité que des centaines voire des milliers de noeuds de capteurs seront hors service.

Le regroupement des noeuds capteurs dans des clusters a été largement adopté par les chercheurs pour satisfaire l'objectif de l'évolutivité, l'efficacité énergétique et le prolongement la durée de vie des réseaux dans des environnements à grande échelle (des réseaux larges). Ces protocoles de routage hiérarchiques exigent une structu-

ration fondée sur des groupes de noeuds capteurs (*clusters*), afin que la fusion de données et l'agrégation seront possibles, ainsi conduisant à une économisation significative d'énergie. Dans cette structure de réseau hiérarchique chaque cluster a un leader, qui est aussi appelé un cluster-chef (*Cluster-Head*) qui assure, généralement, la tâches de la fusion et de l'agrégation des données, et plusieurs noeuds capteurs appelés cluster-membres (*Cluster-members*).

Le processus de formation de clusters conduit à une hiérarchie à deux niveaux où les noeuds cluster-chefs forment le niveau supérieur et les noeuds cluster-membres forment le niveau inférieur. Les noeuds capteurs cluster-membres transmettent périodiquement les données aux noeuds à leurs clusters-chefs correspondants.

Les noeuds cluster-chefs fusionnent les données, diminuant ainsi le nombre total de paquets à transmettre, et les transmettent à la station de base, soit directement par un seul-saut ou par l'intermédiaire d'autres noeuds cluster-chefs en utilisant la technique multi-sauts. Cependant, les noeuds cluster-chefs doivent envoyer toutes les données en temps réel et à des distances plus élevées que les noeuds cluster-membres, ainsi, ils consomment naturellement l'énergie à des taux plus élevés. Une solution pour équilibrer la consommation d'énergie entre tous les noeuds du réseau, est de ré-élire périodiquement de nouveaux cluster-chefs, en faisant, ainsi, tourner le rôle du cluster-chef entre tous les noeuds d'un cluster dans le temps. Un exemple typique d'un réseau à communication hiérarchique est illustré dans la figure 5.1.

La Station de Base (*BS*) ou le Sink est le point central de traitement des données reçues à partir des différents noeuds capteurs d'un RCSF, et c'est depuis le Sink que les données sont accessibles par l'utilisateur final. Le Sink est généralement considéré comme fixe et loin des noeuds capteurs. Les noeuds cluster-chefs agissent effectivement comme des passerelles entre les noeuds capteurs et la BS. La fonction de chaque cluster-chef, comme déjà mentionné, est d'exécuter des fonctions communes à tous les noeuds du cluster, comme l'agrégation des données avant de les envoyer à la BS.

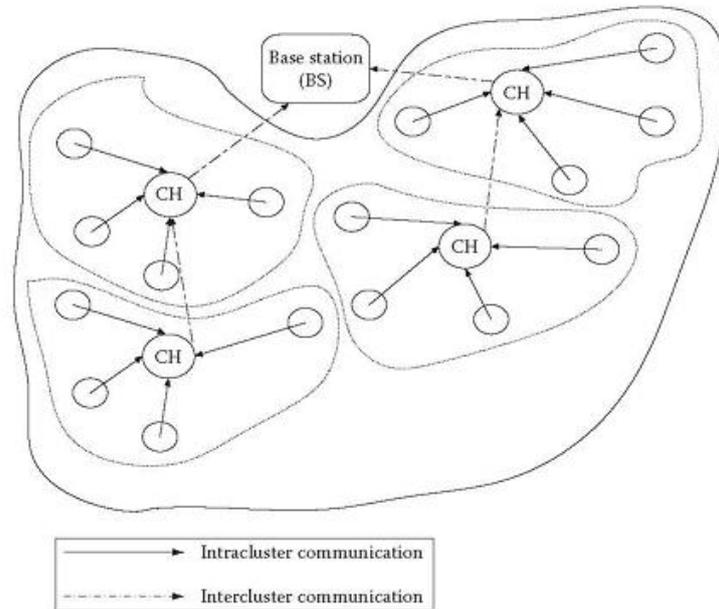


FIGURE 5.1 – La communication de données dans un réseau formé en cluster

D'une certaine manière, le cluster-chef est le Sink pour les noeuds de son cluster, et la BS est le sink pour l'ensemble des cluster-chefs. En outre, cette structure formée par les noeuds, les cluster-chefs et la station de base peuvent être reproduites autant de fois si cela est nécessaire, en créant des couches hiérarchiques multiples dans un RCSF (hiérarchie en clusters multi-niveau). La figure 5.2 représente un RCSF structuré en clusters à 2 niveaux.

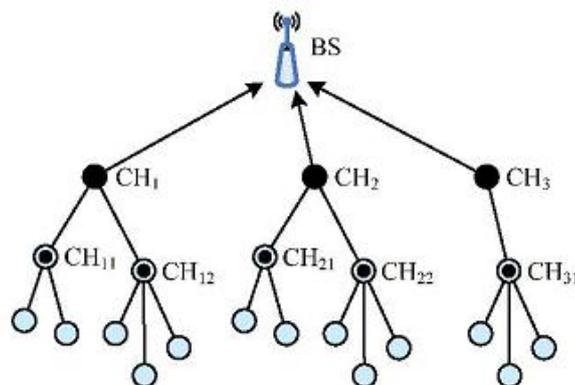


FIGURE 5.2 – Le clustering à deux niveaux

## 5.2 Les objectifs de conception de clusters dans les RCSFs

Le clustering dans les RCSFs peut largement renforcer l'évolution globale du système, la prolongation de la durée de vie du réseau et l'efficacité énergétique. Le routage hiérarchique est un moyen efficace pour réduire la consommation d'énergie au sein d'un cluster, car la portée de communication réduite dans un cluster (entre les cluster-members et leurs cluster-chef) permet de réduire l'énergie consommée de manière significative, aussi l'exécution de l'agrégation et la fusion des données permettent de diminuer le nombre de messages à transmettre à la BS. Contrairement, à un réseau plat, à un seul niveau, qui peuvent produire la congestion suite à l'augmentation de la densité des capteurs. Une telle congestion pourrait provoquer de la latence dans les communications et un suivi inadéquat des événements. En outre, l'architecture plat, n'est pas extensible pour un grand nombre de capteurs, parce que ces capteurs, généralement, ne sont pas capable d'effectuer des communications sur des longues distances. Le clustering est particulièrement utile pour les applications qui nécessitent une évolutivité à des centaines ou des milliers de noeuds. L'évolutivité dans ce contexte implique la nécessité d'équilibrage de charge et de l'utilisation efficace des ressources. Les applications exigeant une agrégation efficace de données sont également des candidats pour le clustering. Les protocoles de routage peuvent également employer le clustering [91, 92]. Dans [93], le clustering a également été proposé comme un outil utile pour repérer efficacement des emplacements d'objets.

En plus de soutenir l'évolutivité du réseau et de diminuer la consommation d'énergie, le clustering a de nombreux avantages et objectifs [88] :

1. Le clustering peut enregistrer les chemins au sein d'un cluster et donc réduire la taille de la table de routage stockée au niveau de chaque noeud.
2. Le clustering peut également conserver la bande passante, car il limite la portée des communications inter-clusters (la communication est entre les cluster-

membres et leurs cluster-chef) et évite l'échange de messages redondants entre les noeuds de capteurs.

3. En outre, le clustering peut stabiliser la topologie du réseau au niveau des capteurs, ainsi on gagne les frais de la maintenance de la topologie, car les capteurs ont une seule tâche qui est la connexion avec leurs cluster-chef, donc ils ne sont pas, généralement, affectée par les changements à l'extérieur du cluster.
4. Le cluster-chef peut aussi mettre en oeuvre des stratégies de gestion optimisés pour améliorer encore le fonctionnement du réseau et pour prolonger la durée de vie de la batterie des capteurs, ainsi la durée de vie du réseau. Un cluster-chef peut planifier des activités à l'intérieur de son cluster de tel sorte que les noeuds peuvent passer en mode de veille pour minimiser la consommation d'énergie. En outre, les capteurs peuvent être utilisés dans un tour de rôle dans le temps pour la transmission et la réception, ainsi la redondance peut être limitée, et la collision est évitée.

### 5.3 Les défis de conception de clusters dans les RCSFs

Les RCSFs présentent plusieurs défis particuliers en termes de conception et de mise en oeuvre. Dans les réseaux de capteurs sans fil, les capacités limitées (de la batterie, du processeur, le rang de transmission de l'antenne et la mémoire utilisée, etc) des noeuds capteurs font de l'efficacité énergétique et de l'évolutivité des facteurs importants. En outre, le défi de prolonger la durée de vie du réseau sous ces restrictions est difficile à atteindre en utilisant uniquement des techniques de clustering traditionnelles. Par conséquent, il devient inévitable de trouver de nouvelles techniques de clustering et de protocoles plus efficaces pour les RCSFs.

En plus des contraintes mentionnées (énergie limitée, capacités limitées, durée de vie du réseau) d'autres considérations importantes dans le processus de concep-

tion d'algorithmes de clustering pour les réseaux de capteurs doivent être pris en considération :

1. **La formation des clusters** : Les procédures de sélection des cluster-chefs et de formation des clusters doivent générer la meilleure combinaison possible (bien équilibrées, etc). Cependant, elles doivent préserver, aussi, le nombre réduit des messages échangés, ainsi que la complexité en temps doit (si possible) rester constante et indépendante de la croissance du réseau.
2. **La dépendance aux applications** : Lors de la conception des protocoles de clustering et de routage pour les RCSFs, la robustesse de l'application doit avoir une grande priorité et les protocoles conçus doivent être capable de s'adapter aux conditions des applications.
3. **Communication sécurisée** : Comme dans les réseaux traditionnels, la sécurité des données possède une grande importance dans les réseaux de capteurs. La capacité de clustering dans les RCSFs pour préserver des communications sécurisées est de plus en plus exigé par les différentes applications comme les applications militaires.
4. **La synchronisation** : les systèmes de transmission tel que *TDMA* permettent aux noeuds de programmer régulièrement des intervalles de sommeil, afin de minimiser l'énergie consommée. Ces systèmes nécessitent de la synchronisation entre les noeuds et l'efficacité énergétique.
5. **L'agrégation des données** : Ce processus permet l'économisation de l'énergie, cependant l'agrégation est un défi dans la conception des réseaux de capteurs. Toutefois, sa mise en oeuvre dans de nombreuses applications n'est pas une procédure simple et doit être optimisé en fonction des besoins des l'applications.

## 5.4 Classification des algorithmes de clustering

### 5.4.1 Paramètres de clustering

Avant d'entamer les algorithmes de clustering et les classifications possibles, il est intéressant de rapporter sur certains paramètres importants dans les réseaux de capteurs sans fil. Ces paramètres servent aussi comme des moyens de base pour la comparaison et la classification des protocoles de clustering [94] :

- **Le nombre de clusters** : Dans la plupart des algorithmes de clustering probabilistes et aléatoire le processus de l'élection des cluster-chefs et de formation des clusters conduit naturellement à un nombre variable de clusters. Dans certaines approches dans la littérature, le nombre des cluster-chefs est prédéterminés et donc le nombre de clusters est fixé. le nombre de clusters est généralement un paramètre critique à déterminer en ce qui concerne l'efficacité de l'ensemble du protocole de routage.
- **La communication intra-cluster** : Dans certains approches, la communication entre un capteur et son cluster-chef est supposé direct (communication en un seul saut). Toutefois, des communications à multi-sauts sont souvent nécessaires, à savoir, lorsque le rang de communication des noeuds capteurs est limitée ou le nombre de noeuds capteurs est très grand et le nombre des cluster-chefs est limité.
- **la mobilité des noeuds** : Si nous supposons que les cluster-membres et les cluster-chefs sont fixes (stationnaires), nous sommes, normalement, amenés à des clusters fixes avec une gestion facile du réseau. Contrairement, si les cluster-membres et les cluster-chefs sont supposé mobiles, l'appartenance au cluster pour chaque noeud change à travers le temps, forçant, ainsi, les clusters à évoluer au fil du temps et de se mettre à jours de manière continue.
- **Types et rôles des noeuds** : Dans certains modèles de réseaux proposés (réseaux hétérogènes), les clusters-chefs sont supposés être équipé par des res-

- sources moins limitées par rapport aux cluster-membres. Dans la plupart des réseaux habituels (les réseaux homogènes) tous les noeuds ont les mêmes capacités et un sous-ensemble de ces noeuds déployés est sélectionné pour être des cluster-chefs.
- **Méthodologie de formation des clusters :** Dans la plupart des approches récentes, où les cluster-chefs sont juste des noeuds capteurs réguliers et le temps de formation des clusters est un critère important, le clustering est exécuté de manière distribué et sans coordination entre les noeuds. Dans d'autres approches, dites centralisées, un noeud coordinateur ou plus sont utilisés pour partitionner l'ensemble du réseau et gérer l'appartenance des noeuds aux clusters.
  - **Sélection des cluster-chefs :** Les noeuds cluster-chefs, dans certains algorithmes proposés (principalement pour des environnements hétérogènes), peuvent être pré-attribués, cependant dans la plupart des cas (à savoir, dans des environnements homogènes), les cluster-chefs sont sélectionnés à partir des noeuds déployés, soit de manière probabiliste ou complètement aléatoire ou en fonction d'autres critères (énergie résiduelle, connectivité, etc).
  - **La complexité algorithmique :** Dans la plupart des algorithmes récents, le temps d'exécution du protocole de clustering est l'un des objectifs de conception les plus importants. Ainsi, la complexité temporelle ou le taux de convergence des procédures de formation des clusters proposées doivent être étudiés minutieusement, la complexité est souvent en fonction du nombre de cluster-chefs ou le nombre de sauts. Dans d'autres protocoles la complexité dépend du nombre total de capteurs.
  - **Le clustering à multi-niveaux :** Dans plusieurs approches proposées dans la littérature, le concept de clustering hiérarchique est proposé pour obtenir de la distribution de charges et une meilleure consommation d'énergie (au

lieu d'utiliser un seul niveau de clusters). Les améliorations offertes par le clustering à multi-niveaux sont importantes dans le cas des réseaux larges où la communication à l'intérieur des clusters est de grande importance.

- **Le chevauchement des noeuds** : Plusieurs protocoles réservent une grande importance aux noeuds produisant des chevauchements entre différents clusters (pour un routage efficace ou pour d'autres raisons). La plupart des protocoles connus, essaient d'avoir un minimum de chevauchement ou ne plus accepter le chevauchement.

## 5.5 Taxonomie des protocoles de clustering

Il y a plusieurs manières, basées sur les paramètres mentionnés ci-dessus, pour distinguer et classer les algorithmes de clustering utilisés pour les réseaux de capteurs sans fil [95]. On peut trouver, dans la bibliographie :

- Des algorithmes de clustering pour les réseaux homogènes ou hétérogènes,
- Des algorithmes de clustering centralisés ou distribués,
- Des algorithmes de clustering statiques ou dynamiques,
- Des algorithmes de clustering probabilistes ou non probabilistes,
- Des algorithmes de clustering basé sur les fonctions combinées.

### 5.5.1 Les algorithmes de clustering pour les réseaux homogènes ou hétérogènes

Cette classification est basée sur les caractéristiques et les fonctionnalités des capteurs dans les clusters. Dans les réseaux de capteurs hétérogènes, il y a généralement deux types de capteurs, on trouve des capteurs équipés par des capacités de traitement plus élevées avec des périphériques complexes. Ces capteurs sont, généralement, utilisés pour jouer le rôle des cluster-chefs dans les RCSFs, servant, ainsi, comme des points de collecte de données et des centres de traitement pour les données collectées par les autres noeuds du cluster, On trouve, aussi, des noeuds

capteurs avec des capacités plus faibles, utilisés dans la surveillance et la détection des évènements de l'environnement.

Dans les réseaux homogènes, tous les noeuds possèdent les mêmes caractéristiques, les mêmes périphériques et les même capacités de traitement. Dans ce type de réseau, qui est le plus utilisé, chaque capteur peut devenir un cluster-chef. En outre, le rôle du cluster-chef peut être un tour de rôle entre les noeuds du réseau pour assurer un meilleur équilibrage de charge et une consommation d'énergie uniforme [96,97].

### 5.5.2 Les algorithmes de clustering centralisés ou distribués

Quand tous les noeuds ont les mêmes capacités (cas des environnements homogènes), une technique d'élection des cluster-chefs de façon centralisés ou distribuée est exécutée pour former les clusters. Selon les auteurs dans [70] le processus distribué est la technique la plus appropriée pour obtenir une meilleure flexibilité et un temps d'exécution rapide, indépendamment du nombre de noeuds du réseau. Dans cette technique chaque noeuds exécute le même algorithme d'élection des cluster-chefs sur les mêmes données, qui en résulte forcément les mêmes résultats.

Il y a aussi d'autre approches utilisant des techniques centralisés [98,99] où un ou plusieurs noeuds coordonnateur ou la station de base qui sont responsable de partitionner l'ensemble du réseau en clusters. Dans cette technique l'algorithme de l'élection des cluster-chefs est exécuté au niveau de la station de base (le sink) et le résultat est communiqué par une simple diffusion sur le réseau. Cette technique à comme avantages de réduire la consommation d'énergie due à l'exécution de l'algorithme de clustering et d'utiliser un algorithme plus puissant pour déterminer les meilleures combinaisons possibles (cluster-chefs et cluster-membres) en utilisant plusieurs paramètres important dans la détermination de cette combinaison. Dans cette thèse, nous nous concentrons principalement sur les techniques centralisées pour les environnements homogènes, qui sont les plus efficaces, en particulier pour

les grands réseaux et les plus utilisés de nos jours [94].

### 5.5.3 Les algorithmes de clustering statiques ou dynamiques

Une autre classification, pour les algorithmes de clustering des RCSFs, connue par clustering statique et clustering dynamique. Dans le clustering statique, les cluster-chefs sont fixés au déploiement des noeuds et ne subissent aucun changement au fil de temps, ce type de clustering est généralement réalisé dans les réseaux hétérogènes. Ce type de clustering conduit à un épuisement rapide des ressources énergétiques des noeuds et donc une déconnexion rapide du réseau [97,100].

Une procédure de formation de clusters est considérée comme dynamique quand elle comprend une réélection périodique des cluster-chefs, soit pour réagir efficacement aux modifications de la topologie du réseau afin de réajuster les clusters, soit, tout simplement, pour viser une rotation du rôle de cluster-chef entre les noeuds pour renforcer l'efficacité énergétique. Le clustering dynamique est la meilleure technique pour les réseaux capteurs et conduit naturellement à une meilleure gestion de la consommation d'énergie et prolonge la durée de vie du réseau [101,102].

### 5.5.4 Les algorithmes de clustering probabilistes ou non probabilistes

Cette classification est la plus répandue dans la littérature. En effet, La plupart des algorithmes de clustering connus, pour les réseaux de capteurs sans fil, peuvent être divisés en deux catégories principales, en fonction des critères de formation des clusters et des paramètres utilisés dans la sélection des cluster-chefs, ainsi, nous trouvons :

1. Les algorithmes de clustering probabilistes,
2. Les algorithmes de clustering non probabilistes.

### 5.5.4.1 Approches de clustering probabilistes

Dans les algorithmes de clustering probabilistes, une probabilité est pré-affectée à chaque noeud capteur pour déterminer les cluster-chefs [103,104], cette probabilité est le facteur principal avec lequel chaque noeud décide, individuellement, s'il sera élu comme un cluster-chef ou non d'une manière souple, uniforme, rapide et complètement distribuée. Cependant, d'autres critères secondaires peuvent également être pris en considération soit pendant le processus de selection des cluster-chefs, comme l'énergie résiduelle, ou pendant le processus de formation des clusters, comme la proximité et le coût de la communication entre les cluster-membres et leurs cluster-chef, pour avoir une consommation d'énergie optimale et une longue durée de vie du réseau. Ce type d'algorithmes permet, généralement, d'obtenir une exécution rapide en terme de temps de convergence et une réduction de nombre de messages échangés.

L'utilisation efficace des réseaux de capteurs sans fil est une nécessité et d'une grande importance, pour cette raison plusieurs algorithmes de clustering probabilistes ont été développés pour répondre aux différentes exigences (augmenter la durée de vie des réseaux, réduire et répartir uniformément la consommation d'énergie, l'évolutivité, etc).

L'algorithme le plus populaire et le plus significatif pour ce type de clustering dans les RCSFS est *LEACH* (*Low Energy Adaptive Clustering Hierarchy*) [94,103]. *LEACH* est un algorithme de clustering probabiliste, son principal objectif est de réduire la consommation d'énergie et de prolonger la durée de vie des réseaux. La sélection des cluster-chefs dans *LEACH* est complètement aléatoire et distribuée, réduisant ainsi la consommation d'énergie. Cependant, *LEACH* contient plusieurs inconvénients [99,105]. Une étude détaillée de *LEACH* est présentée dans la section suivante, afin de dénombrer ses avantages et ses inconvénients. Cet algorithme est utilisé, dans la deuxième partie de cette thèse, dans la comparaison avec notre approche proposée, afin de prouver l'efficacité de celle-ci par rapport à *LEACH*.

### Low Energy Adaptive Clustering Hierarchy (LEACH)

*LEACH* est un protocole auto-organisateur basé sur le clustering adaptatif, qui utilise la rotation aléatoire des cluster-chefs, pour distribuer équitablement la consommation d'énergie entre les noeuds capteurs dans un réseau [70, 103]. Ce dernier est considéré comme étant l'une des premières approches de routage hiérarchique basées sur le clustering. LEACH est fondé sur deux hypothèses de base dans lesquelles :

1. La station de base est fixe.
2. Tous les noeuds du réseau sont supposés homogènes et limités en énergie.

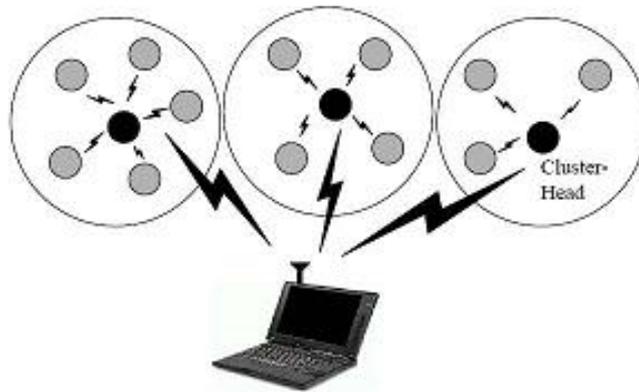


FIGURE 5.3 – Algorithme de routage LEACH

L'idée derrière LEACH est de former des clusters en se basant sur l'intensité du signal reçue, et d'utiliser les cluster-chefs comme des routeurs pour transférer les données à la station de base, voir figure 5.3. Le rôle principal de LEACH est :

- La coordination et le contrôle localisé : pour l'initialisation, le traitement et le routage de données.
- La rotation aléatoire du rôle de cluster-chef : effectué par les noeuds eux même.
- L'agrégation des données : les noeuds cluster-chefs fusionnent les données arrivant des noeuds appartenant à leurs clusters, et envoient un paquet agrégé

à la station de base, afin de réduire la quantité de messages à transmettre.

LEACH fonctionne sur des cycles de longueur constante, où chaque cycle commence par une phase d'initialisation suivie par une phase de transmission. Dans la première phase, les clusters sont organisés et les cluster-chefs sont sélectionnés. Cette sélection est basée sur le pourcentage désiré des cluster-chefs, en effet les auteurs dans LEACH ont fixé le nombre des cluster-chefs à sélectionner dans chaque cycle à 5% de la totalité des noeuds déployés. Un autre facteur important dans la sélection des cluster-chefs est un seuil appelé  $T(n)$ , défini dans l'équation 5.1. Ainsi, un noeud  $n$  prend une valeur aléatoire entre 0 et 1, si cette valeur est inférieure au seuil  $T(n)$  le noeud se déclare comme cluster-chef dans son voisinage.

$$T(n) = \begin{cases} \frac{P}{1-P(r \bmod \frac{1}{P})} & \text{Si } n \in G \\ 0 & \text{Sinon} \end{cases} \quad (5.1)$$

Où :

- $P$  : Le pourcentage désiré de cluster-chefs, qui est fixé par les auteurs à 5%.
- $r$  : Le numéro de l'itération actuelle.
- $G$  : L'ensemble des noeuds qui ont été sélectionnés comme des cluster-chefs durant les dernières  $(1/P)$  itérations.

Après la phase de la sélection, chaque nouveau cluster-chef diffuse un message de signalisation au reste des noeuds dans le réseau afin de les informer de son élection comme un nouveau cluster-chef. Chaque noeud cluster-membre (non cluster-chef), et après avoir reçu ces messages, décide à quel cluster il doit appartenir. Cette décision est basée sur l'intensité du signal du message reçu, car une intensité forte est équivalente à un cluster-chef proche. Les noeuds cluster-membres informent les cluster-chefs appropriés qu'ils seront membres dans leurs clusters.

Après la réception de tous les messages des noeuds qui voudraient être inclus dans le cluster, le nombre des messages dépend du nombre de noeuds dans ce cluster,

le noeud cluster-chef crée un plan de transmission et assigne à chaque noeud un slot de temps durant lequel il peut transmettre. Ce plan est émis à tous les noeuds dans le cluster. En effet, L'algorithme LEACH utilise la technique de multiplexage temporel *TDMA* (*Time division multiplexed access*) comme méthode d'accès au médium par les noeuds su sein du cluster, dans la quelle chaque noeud utilise la totalité de la bande passante allouée par le système de transmission durant son slot de temps. Le plus, les noeuds peuvent passer à l'état veille durant les slots inactifs. Ainsi, la perte d'énergie due aux états de sur-écoute (*overhearing*) et l'écoute passive (*idle*) est évitée.

Dans la deuxième phase : le transfert, des données collectées, à la station de base commence. La durée de la deuxième phase est plus longue que celle de la première, afin de réduire au minimum les problèmes de sur-écoute. La collection de données dans un cluster est centralisée et est exécutée périodiquement. Par conséquent, ce protocole s'avère le plus approprié quand on constate un besoin de surveillance constant par un réseau de capteurs.

Après un intervalle de temps donné, une rotation aléatoire des cluster-chefs est effectuée de sorte que la consommation d'énergie uniforme dans le réseau de capteurs est garantit, avec le même nombre des cluster-chefs, car les auteurs ont trouvé, en se basant sur leur modèle de simulation, que seulement 5% des noeuds sont suffisant pour agir comme des cluster-chefs. Le processus de formation de clusters par LEACH est représenté par l'organigramme dans la figure 5.4.

Malgré que LEACH permet de réduire la perte d'énergie, d'optimiser l'utilisation de largeur de bande et de prolonger la durée de vie des réseaux, un certain nombre d'inconvénients restent plus au moins apparents. Nous citons parmi eux :

1. La phase de formation de clusters augmente le nombre de messages échangés entre les noeuds, qui peut provoquer la consommation d'énergie. En effet, un RCSF peu contenir des milliers voir des millions de capteurs, ces capteurs

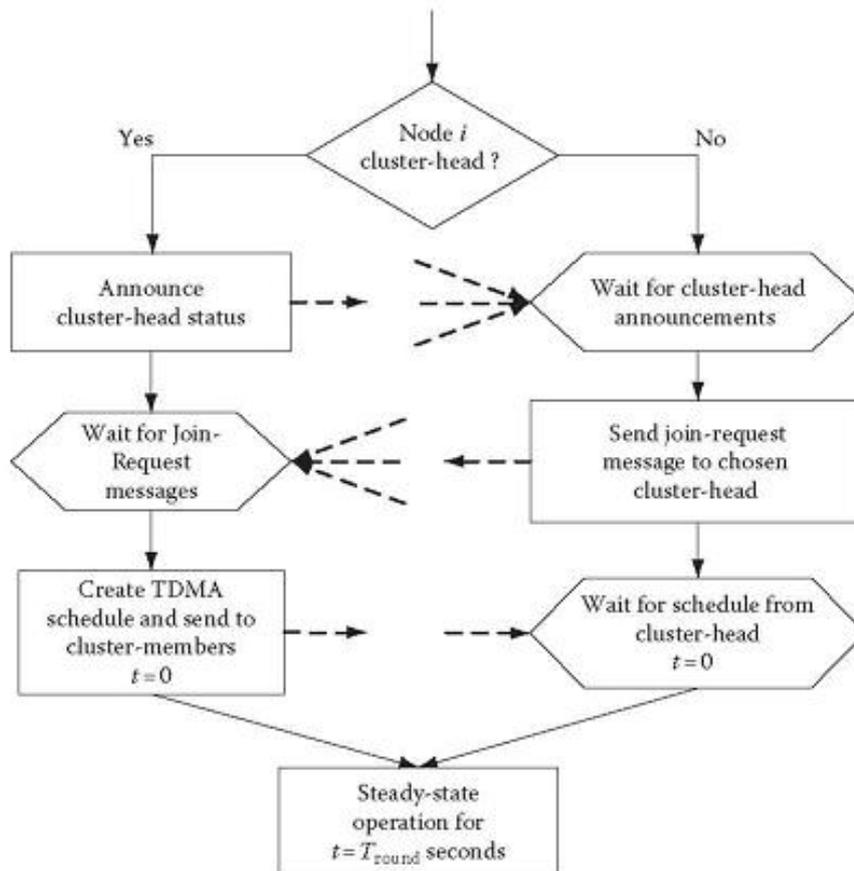


FIGURE 5.4 – Processus de formation des clusters par LEACH

doivent communiquer avec leurs cluster-chefs correspondants ce qui provoque de l'interférence, le sur-écoute, la congestion et la collision, etc.

2. Les noeuds les plus éloignés de leur cluster-chef meurent rapidement par rapport a ceux plus proches (communication a un seul-saut au sein du clusters), car LEACH ne garantit pas que les cluster-chefs soient aux centres des clusters,
3. Le nombre de messages reçus par le cluster-chef est égal au nombre des noeuds gérés dans son cluster. Ceci mène a l'épuisement rapide de sa reserve d'énergie,
4. l'utilisation de la technique un seul-saut au lieu de multi-sauts épuise rapidement l'énergie des noeuds et consomme d'avantage la largeur de bande,
5. LEACH suppose que tous les noeuds peuvent transmettre avec une énergie

assez suffisante pour atteindre la BS. Ce qui n'est pas toujours le cas.

6. LEACH ne garantit pas la couverture réseau, étant donné que seulement 5% des noeuds déployés seront élus comme des cluster-chefs, ainsi ces cluster-chefs peuvent se situer dans la même zone géographique, ceci est dû à la sélection aléatoire des clusters-chefs,
7. Il peut arriver à un certain moment qu'un noeud avec une très faible d'énergie peut être élu comme un cluster-chefs, ce qui causera sa mort avant la fin de la phase de transmission.
8. Les noeuds cluster-membres situés entre deux clusters peuvent provoquer de l'interférence, ce qui nécessitera la re-transmission qui est trop coûteuse en terme d'énergie et en terme de temps de réponse.
9. Il ne peut pas être appliqué à des applications avec une contrainte de temps de réponse du fait qu'il résulte en une longue latence.

Étant donné que LEACH est considéré comme étant le fondateur de l'approche de routage hiérarchique basée sur le clustering, plusieurs recherches ont été proposées afin d'améliorer ce dernier. Parmi ces recherches on peut citer les protocoles suivants : AROS [71], HCR [68], SPEAR [72], Energy-LEACH et MultiHop-LEACH [73], ECSA [74], E-LEACH [75], Improved-LEACH [76], STATIC-LEACH [77], Q-LEACH [78], LEACH-F [106] et LEACH-TLCH [69].

#### 5.5.4.2 Approches de clustering non probabilistes

Dans cette technique de clustering, appelée aussi clustering déterministe, des critères spécifiques (déterministes) pour l'élection des cluster-chefs et la formation de clusters sont pris en considération, qui sont principalement basées sur la proximité des noeuds et sur les informations reçues des noeuds voisins [107, 108].

La procédure de formation des clusters est principalement basée sur la communication de noeuds avec leurs voisins (les voisins à un-saut ou multi-sauts) et

nécessite, généralement, l'échange de messages. Ce type d'algorithmes construit, généralement, des clusters robustes et bien équilibrés [94]. En plus de la proximité des noeud, d'autres algorithmes utilisent une combinaison d'autres mesures telles que l'énergie restante dans les noeuds, la puissance de transmission, la mobilité, etc, formant ainsi une fonction multi-objectives combinée [109, 110].

Dans la même catégorie, nous abordons, également, deux nouvelles classes d'algorithmes de clustering pour les réseaux de capteurs sans fil :

1. Dans la première classe, le clustering est généralement calculé à la base de plusieurs critères ayant de différents priorités.
2. La deuxième classe est basée sur les inspirations biologique (les algorithmes génétique, colonie de fourmis, etc), qui sont probablement des approches alternatives et les plus prometteuses pour le clustering dans les réseaux de capteurs sans fil dans nos jours [111, 112].

L'approche proposée dans cette thèse est une combinaison de ces deux classe, où on utilisé les algorithmes génétique pour améliorer la fonction combinée utilisée. Dans ce qui suit nous présenterons en détail ces deux classes de clustering.

### 5.5.4.3 Les algorithmes de clustering basés sur les fonctions combinées

En plus de la proximité des noeuds, d'autres algorithmes utilisent une combinaison de plusieurs métriques telles que l'énergie restante, la puissance de transmission, la position, etc, formant ainsi une fonction combinée pour atteindre des objectifs plus généraux qu'un critère unique. Plusieurs algorithmes utilisant ce type clustering ont été initialement développés pour les réseaux Ad-hoc [109, 113]. Un exemple typique est l'algorithme *WCA* proposé dans [113], où la sélection des cluster-chefs est basée sur le calcul d'une fonction combinée  $W_n$  pour chaque noeud  $n$ , qui prend en compte différents paramètres du système tels que le degré de noeud, la puissance de transmission, la mobilité et l'énergie restante du noeud :

$$W_n = w_1 T_n + w_2 D_n + w_3 M_n + w_4 P_n \quad (5.2)$$

Où les  $w_i$  sont des poids de pondération à déterminer et  $T_n$ ,  $D_n$ ,  $M_n$  et  $P_n$  représentent respectivement la puissance de transmission, le degré, la mobilité et l'énergie d'un noeud  $n$ .

Le fonction combinée est calculée et le résultat est diffusé par chaque noeud. Le noeud ayant la plus petite valeur de poids dans son voisinage est choisi comme un cluster-chef. Cet algorithme a tenté de fournir un meilleur équilibrage de charge par la réduction du nombre de capteurs dans un cluster, mais l'exigence de l'horloge de synchronisation limite son application.

Plusieurs algorithmes de clustering de ce type ont été proposés dans la littérature ; nous citons : *DWEHC*, pour *Distributed Weight-Based Energy-Efficient Hierarchical Clustering*, présenté dans [114] et *TASC*, pour *Topology Adaptive Spatial Clustering*, présenté dans [110].

#### 5.5.4.4 Les algorithmes de clustering basés sur les inspirations biologique

Durant cette dernière décade, de nouveaux algorithmes de clustering ont été proposés sur la base des algorithmes méta-heuristiques. Ces algorithmes ont montré des résultats très prometteurs dans les simulations par rapport à plusieurs d'autres protocoles en ce qui concerne la durée de vie du réseau.

Nous avons cité dans les chapitres précédents les contraintes relatives aux réseaux de capteurs. Ces contraintes ont poussé les chercheurs à optimiser ces réseaux et à trouver des solutions possibles à leurs problèmes. D'autre part, les systèmes biologiques ont des propriétés attirantes et peuvent présenter des sources d'inspiration puissantes pour les chercheurs du domaine. Cette section est donc consacrée à la présentation des systèmes biologiques, leurs caractéristiques ainsi qu'une vue générale des inspirations qui ont déjà abouti à des solutions applicables dans le domaine

des réseaux de capteurs sans fil [115].

### **Caractéristiques des systèmes biologiques**

Les systèmes biologiques sont adaptés aux changements des environnements. Ils sont résistants aux pannes et peuvent avoir un comportement collaboratif réussi basé sur un ensemble limité de lois simples.

L'intelligence globale résultante d'un tel comportement est alors plus grande que la somme des intelligences individuelles des éléments qui constituent le système. A cela s'ajoutent quelques caractéristiques attirantes : l'auto-organisation, la survie, la distribution, et l'évolution [115, 116].

Les caractéristiques citées ci-dessus sont plus que suffisantes pour pousser les chercheurs à penser à innover des solutions pour les RCSFs à partir des comportements des systèmes biologiques. En fait, une grande majorité des qualités demandées dans un réseau futur sont présentes dans les systèmes biologiques.

### **Applications des solutions inspirées de la biologie dans les RCSFs**

Nombreuses sont les approches qui se sont inspirées des systèmes naturels pour essayer de suggérer des solutions pour les problèmes posés par les réseaux de capteurs, l'effet magique de l'application des principes biologiques aux RCSFs se montre par un comportement extrêmement complexe qui résulte d'un ensemble réduit de lois et d'interactions. Le modèle mathématique et le concept architectural sont faciles à comprendre, puisqu'on se contente souvent d'une vue globale et simplifiée du système biologique [115].

L'interaction entre les techniques de l'informatique et les systèmes biologiques est réciproque. L'inspiration biologique donne des solutions avancées aux problèmes de l'informatique mais un système biologique ne peut pas être modélisé sans les techniques de l'informatique. Le développement dans l'un des deux domaines aboutit donc à un développement dans l'autre [115, 117].

Selon les auteurs dans [116, 118], un regard attentif sur les études les plus matures dans le domaine des inspirations biologiques révèle que trois étapes communes sont toujours nécessaires pour le développement des méthodes bio-inspirés ayant un impact remarquable dans le domaine visé : l'identification d'analogies ; la compréhension et l'ingénierie. Selon les auteurs, les essais récents dans le domaine doivent investir au fond les avantages et les objectifs d'un système biologique.

Ces inspirations biologiques ont été, également, utilisées pour renforcer la sécurité des RCSFs, en se basant sur les systèmes immunitaires naturels. Les *AIS* (*artificial immune systems*) constitue un nouvel axe de recherche dans les RCSFs, nous présentons la notion des AIS avec plus de détaille dans la section suivante.

Pour les RCSF, beaucoup d'exemples peuvent être présentés pour argumenter la puissance des approches biologiques. Cette section était une introduction qui justifie cette direction de recherche. Parmi les algorithmes de clustering basés sur les inspirations biologiques nous citons : *ANTCLUST* présenté dans [111], où les auteurs ont proposé un algorithme de clustering basé sur les colonies de fourmis. Dans *ANTCLUS*, les noeuds avec les plus grandes réserves d'énergie deviennent des cluster-chefs. Chaque noeud avec une faible énergie choisit son cluster selon des critères strictes comme l'énergie résiduelle du cluster-chef, la distance au cluster-chef et l'estimation de la taille du cluster [94]. D'autres algorithmes de clustering de ce type, assurant une meilleure distribution des cluster-chefs et une consommation d'énergie minimale, on été, également, proposés et validés par des simulations dans la littérature dans [112, 119, 120] où leurs objectifs étaient de réduire la consommation d'énergie et les distances de communication intra-cluster.

Généralement, les algorithmes de clustering inspirés de la biologie ont montré leurs efficacité, en contrôlant dynamiquement la sélection des cluster-chefs, tout en réalisant une distribution, assez uniforme, des cluster-chefs et de la consommation d'énergie.

## 5.6 Les systèmes immunitaires artificiels pour les RCSFs

Le système immunitaire est un système de défense remarquablement adaptatif. Il est capable de créer une très grande variété de cellules et de molécules susceptibles de reconnaître et d'éliminer spécifiquement un nombre pratiquement illimité d'envahisseurs étrangers.

A partir de ce système est inspiré un système immunitaire artificiel (*Artificial Immune System, AIS*), dont les méta-heuristiques d'optimisation AIS font partie, selon les aspects de base d'un système immunitaire, à savoir l'apprentissage et mémoire [121].

### 5.6.1 Les AIS pour les RCSFs

Les AIS sont basées sur un mécanisme qui est présent dans le corps humain, à savoir, sur le Système immunitaire humain. Ils sont une partie prometteuses des avancées récentes dans les systèmes de détection d'intrusions dans les RCSFs.

En se basant sur les motivations présentées dans [121], nous avons entrepris une étude de la performance des AIS en se concentrant sur les réseaux de capteurs. Les conclusions générales basées sur la construction d'un AIS pour les réseaux de capteurs peuvent être résumées comme suit :

- Compte tenu des capacités de calcul limitées des capteurs sans fil, nous pouvons dire que la détection d'intrusions en fonction des AIS offre un taux de détection remarquable à un faible coût de calcul, de ce fait ils peuvent être une solution idéale pour les réseaux de capteurs.
- L'un des principaux défis dans la conception et l'exécution de l'AIS dans les réseaux de capteurs est l'ensemble des gènes. Les gènes sont nécessaires pour mesurer la performance d'un réseau à partir d'un noeud. Ces gènes doivent être faciles à calculer et robustes contre les intrusions.
- Pour accroître les performances de détection, un AIS doit utiliser les infor-

mations disponibles dans toutes les couches de la pile protocolaire OSI, cela comprend également la détection de performances en ce qui concerne la perte de paquets.

## 5.7 Conclusion

Le clustering dans les réseaux de capteurs a attiré une grande attention des chercheurs durant cette dernière décennie, ce qui a permis l'apparition de plusieurs travaux dans la littérature. A travers ce chapitre on a présenté le clustering, ses caractéristiques, ses objectifs, ainsi que les défis de conception des algorithmes de clustering. Comme il a été souligné dans ce chapitre, le groupement des noeuds en clusters dans le but d'un routage hiérarchique est l'approche la plus efficace pour soutenir l'évolutivité dans les réseaux de capteurs. Le clustering est un moyen efficace pour la collecte et l'agrégation des données et pour réduire l'énergie à dépensé durant les communications.

L'objectif principal des algorithmes de clustering existant reposent sur la prolongation de la durée de vie des réseaux et l'utilisation optimale des ressources critiques tel que la batterie, négligeant ainsi d'autres paramètres importants des réseaux de capteurs tel que la couverture réseaux. Dans cette these, nous avons présenté un nouvel algorithme de clustering basé sur une fonction combinée multi-objectifs et les algorithmes génétique [99]. l'algorithme de clustering proposé dans la deuxième partie de cette thèse, à pour but de réduire la consommation d'énergie, d'éviter l'extinction très tôt des noeuds et d'améliorer la couverture réseau.



Deuxième partie

Contributions dans le domaine



# Analyse énergétique : Longs-sauts Vs Courts-Sauts

---

## 6.1 Résumé

Les RCSFs sont composés de noeuds capteurs sans fil miniaturisés, chaque noeud capteur est doté d'une source d'énergie limitée et d'une capacité de traitement réduite. Généralement, ces réseaux de capteurs sont utilisés dans de nombreuses applications de surveillance dans des environnements inaccessibles (les champs de bataille, la surveillance du volcan, le suivi des animaux etc), d'où l'impossibilité de remplacer ou de recharger les batteries. Et comme, les capteurs sont généralement déployés dans des grandes surfaces, l'émetteur-récepteur radio est le plus grand consommateur d'énergie dans un noeud capteur, ce qui signifie que son utilisation doit être très efficace afin de maximiser la durée de vie des noeuds, ce qui nous amène à maximiser la durée de vie du réseau.

## 6.2 Introduction

Dans les réseaux de capteurs sans fil et afin de transmettre ses données, un noeud peut acheminer ses messages vers la destination, généralement la station de base, par la technique multi-sauts, ces sauts peuvent être des sauts courts (*short-hops*) ou des longs sauts (*long-hops*), ainsi, l'optimisation de la longueur du saut peut prolonger de manière significative la durée de vie du réseau. Dans ce chapitre nous présenterons notre première contribution dans cette thèse qui porte sur l'analyse des longueurs des sauts de communication dans les RCSFs [122]. Nous présenterons une condition simple à vérifier, qui rend la consommation d'énergie minimale en

choisissant une longueur de saut idéale.

Durant ces dernières années, les réseaux de capteurs sans fil ont été utilisés dans de nombreuses applications, comme la surveillance militaire, la gestion des catastrophes [123], la détection des incendies des forêts, la détection sismique [124], le suivi de l'habitat, la surveillance de la santé des patients dans un hôpital [125], le suivi des stocks, le suivi des animaux, les applications commerciales, les applications domestiques [2], etc. En effet, Business 2.0 liste les réseaux de capteurs sans fil comme l'une des six meilleures technologies qui vont changer le monde, et les magazines Technology Review de MIT et Global future présentent les réseaux de capteurs comme l'une des 10 nouvelles technologies qui vont changer notre vie [126].

Un réseau de capteurs sans fil se compose de centaines voire des myriades de noeuds capteurs, qui peuvent être déployés de manière aléatoire par une voiture, un avion ou une lance-roquettes. Chaque noeud a des strictes limitations sur l'usage de sa source d'énergie, son processeur et sa mémoire. Afin de transmettre leurs données, les noeuds capteurs utilisent, généralement, les communication sans fil. Par conséquent, les réseaux de capteurs doivent être bien structurés pour atteindre leurs objectifs et pour étendre leurs durées de vie. En effet, la façon dont le réseau est formé détermine sa durée de vie ainsi que la qualité de transmission des données.

Les noeuds capteurs sont, toujours, dotés d'une batterie avec une puissance limitée et souvent déployés, aléatoirement, dans des zones inaccessibles par les humains, d'où l'impossibilité, dans la plupart des cas, de recharger ou de remplacer la batterie. Ainsi, la batterie est considérée comme une ressource critique dans les RCSFs, et son utilisation doit être efficace. Un noeud capteur consomme l'énergie de sa batterie dans la détection, la réception, le traitement et l'émission des données.

Selon l'auteur dans [2] Le composant qui consomme plus d'énergie est le module radio qui fournit des communications sans fil . Par conséquent, parmi toutes les opérations des noeuds capteurs, c'est l'opération de l'émission et de la réception

des données qui consomme plus d'énergie. En effet, l'énergie consommée pour la transmission de 1 bit, par un capteur, sur le canal sans fil est équivalente à l'énergie requise pour exécuter des milliers de cycles d'instructions par son processeur [54]. Par conséquent, les protocoles de communication consomment une grande partie de l'énergie des noeuds, affectant, ainsi, la durée de vie des réseaux de capteurs sans fil.

Dans la plupart des RCSFs, les noeuds capteurs ne possèdent pas la puissance et la portée de communication nécessaire pour envoyer directement les données collectées, depuis l'environnement, à la station de base. Ainsi, le mode de communication multi-sauts, est généralement, utilisé pour transmettre les données et atteindre la station de base [2], les protocoles multi-sauts consistent à envoyer les données de chaque noeud à ses voisins, qui, à leurs tours envoient ces données à leurs voisins, et ainsi de suite, jusqu'à ce que la station de base soit atteinte. Ainsi, un noeud capteur typique ne transmet pas seulement ses propres données mais il joue, aussi, le rôle d'un routeur.

Dans nos jours, les capteurs sont équipés par des modules radio intelligents qui peuvent ajuster leurs puissances de transmission [50], afin d'atteindre leurs proches voisins ou leurs voisins éloignés ou le Sink, donc dans les réseaux de capteurs la station de base peut être atteinte avec soit un grand nombre de courts sauts ou avec un nombre réduit de grands sauts, en utilisant toujours les protocoles à multi-sauts. L'efficacité énergétique, dans ces deux approches, est basée sur la consommation d'énergie par la radio dans les deux modes de fonctionnement, à savoir la réception et l'émission des données.

L'idée concernant le nombre requis de sauts pour atteindre la station de base, vient du fait que chaque stratégie (routage avec des longs-sauts et des courts-sauts) a ses propres avantages. Car la transmission des données à la station de base avec plusieurs courts sauts minimise l'énergie de transmission qui augmente avec

l'augmentation de la distance de communication. Cependant, l'envoi des données avec des longs sauts réduit le coût de la réception (suite à la diminution du nombre de noeuds dans le chemin de routage) ce qui est ignoré dans de nombreux travaux de recherche [53].

En effet dans de nombreux papiers, l'énergie requise pour la réception dans les réseaux de capteurs est négligée [50], ce qui rend l'utilisation des courts sauts toujours préférable, c-à-d une faible énergie pour la transmission. Cependant, ceci n'est pas toujours le cas, parce que dans de nombreux travaux [1, 103], les auteurs ont montré que la puissance de réception ne peut pas être ignorée, et elle doit donc être incluse pour calculer l'énergie de transmission lors de l'utilisation des noeuds intermédiaires dans le routage à multi-sauts.

Dans ce chapitre et en se basant sur la comparaison entre les deux stratégies, nous allons fournir une condition simple sur la longueur des sauts utilisés, qui une fois respectée, nous conduit à prolonger la durée de vie des réseaux de capteurs, en optimisant la consommation d'énergie.

### 6.3 Travaux connexes

Au cours de ces dernières années, le problème de routage des paquets par des longs-sauts ou des courts-sauts a été traité par de nombreux auteurs et leurs conclusions sont variées selon les critères pris en compte et de l'approche adoptée [122, 127]

Certains travaux théoriques [127, 128] montrent que le routage avec des multi-sauts est plus efficace que le routage avec un seul-saut. Ceci est dû à des observations de certains RCSFs dans le monde réel, d'autres travaux montrent que le routage à un saut, peut être beaucoup plus efficace en terme d'économie d'énergie par rapport au routage multi-sauts [129, 130]. En outre le routage à un seul-saut peut aussi avoir des avantages pour d'autres paramètres de réseau, tels que le retard (*Latency*), la minimisation de la perte de paquets, évitant ainsi le besoin de la retransmission

qui est trop coûteux, etc [50].

Les auteurs dans [131] ont présenté deux stratégies de contrôle, l'une des méthodes consiste à diminuer la portée de transmission de chaque noeud. Selon les auteurs, cette diminution permettra de réduire la consommation d'énergie globale du réseau, en prouvant qu'une route avec de nombreux courts sauts est généralement plus efficace qu'une route avec des longs-sauts.

*Haenggi* dans [130] a précisé de nombreuses raisons pour utiliser le routage avec des longs-sauts. L'auteur a affirmé que l'énergie de transmission diminue considérablement avec la distance mais, selon lui, la diminution de la distance de transmission ne diminue pas, toujours, la consommation d'énergie totale .

Dans ce chapitre, nous allons fournir la longueur optimale des sauts qui rend la consommation d'énergie optimale. Pour cette fin, nous avons besoin d'utiliser le modèle mathématique de la consommation d'énergie, définie pour les réseaux de capteurs sans fil.

## 6.4 Modèle de propagation utilisé

Un lien radio entre l'émetteur et le récepteur ne peut être établie que lorsque la force du signal radio reçue est supérieure au seuil de la sensibilité du récepteur [50]. La réduction de l'intensité de puissance du signal, sur le chemin entre l'émetteur et le récepteur, est appelée perte de signal (*path loss*). Dans notre étude, nous utilisons un modèle de perte du signal, de telle sorte que la puissance reçue par un noeud distant de  $d$  mètres de l'émetteur peut être exprimée comme suit [50, 122, 127, 132] :

$$P(d) = P_0 \times \left(\frac{d_0}{d}\right)^\alpha \quad (6.1)$$

Où  $P_0$  représente la puissance du signal reçue à la distance  $d_0$  de l'émetteur et  $\alpha$  est l'exposant de perte de signal, qui est empiriquement mesuré dans des différents milieux de propagation [127, 132]. Des valeurs typiques de l'exposant de perte du signal dans de tels milieux sont présentés dans le tableau 6.1.

<i>Environment</i>	$\alpha$
Free-space	2
Urban area LOS	2,7 to 3,5
Urban area no LOS	3 to 5
Indoor LOS	1,6 to 1,8
Factories no LOS	2 to 3
Buildings no LOS	4 to 6

TABLE 6.1 – Valeurs typiques de l'exposant de perte du signal

Dans ce chapitre, nous utilisons l'équation 6.1 pour exprimer la puissance minimale requise pour communiquer sur une distance donnée et nous comparons les deux stratégies de routage (routage avec des longs-sauts et routage avec des courts-sauts).

Chaque noeud transmet avec une puissance minimale, en garantissant l'intensité du signal nécessaire au niveau du récepteur, qui est égale au seuil de sensibilité du récepteur [53, 122, 127]. Selon la figure 6.1, nous pouvons donc écrire :

$$P_t = P_x \times \left(\frac{d_0}{x}\right)^\alpha \quad (6.2)$$

À partir du quelle, nous pouvons obtenir l'énergie nécessaire pour atteindre la destination :

$$P_x = P_t \times \left(\frac{x}{d_0}\right)^\alpha \quad (6.3)$$

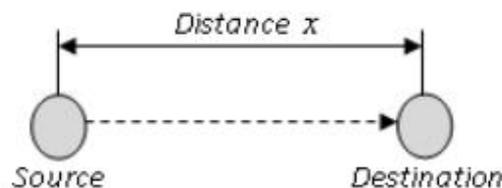


FIGURE 6.1 – La distance de transmission pour un saut

Dans certains modèles théoriques de transmission sans fil, les auteurs supposent que toute l'énergie consommée est effectuée par l'émetteur et le récepteur ne dépense

pas de l'énergie lors de la réception [50]. Selon eux, si on considère que la consommation d'énergie est due uniquement par l'émission, alors la stratégie à courts-sauts est la plus économe en énergie, mais puisque le coût de la réception ne peut pas être négligé [2, 52, 128], nous allons montrer dans ce chapitre que l'utilisation de la stratégie à long-sauts entre deux noeuds (source et destination) peut être une alternative optimale. Cela est dû au fait que la réduction de puissance de transmission par le routage à courts-sauts ne compense pas le coût de l'énergie de réception consommée. Le coût de l'énergie de la réception peut être équivalent à la transmission sur une distance  $t$  [128]. Ainsi, elle peut être exprimée par la formule :

$$P_r = P_t \times \left( \frac{t}{d_0} \right)^\alpha \quad (6.4)$$

## 6.5 Analyse des courts-sauts Vs Long-sauts

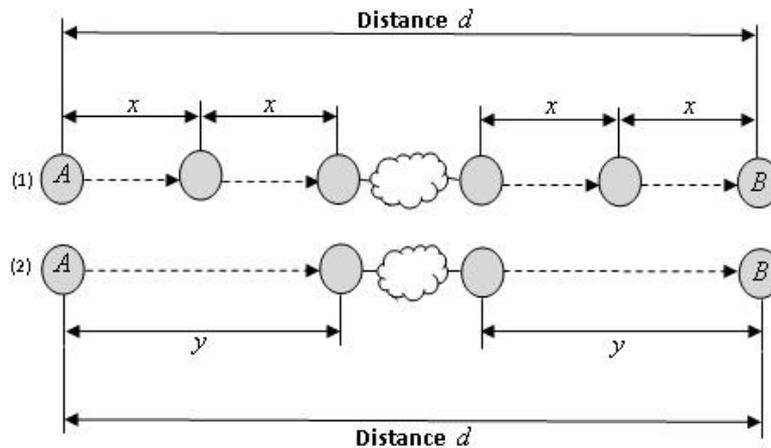


FIGURE 6.2 – Transmission avec : (1)  $n$ -sauts et (2)  $m$ -sauts

La figure 6.2 représente deux topologies de routage à multi-sauts entre deux noeuds distants, une source  $A$  et une destination  $B$ , séparés par une distance  $d$ . La première topologie utilise  $n$  sauts pour transmettre des données à partir de  $A$  vers  $B$  (en utilisant des courts-sauts de distance  $x$ ) ; tandis que la deuxième utilise  $m$  sauts (à l'aide des longs-sauts de distance  $y$ ), avec  $x = 2y$ , donc nous pouvons écrire :

$$P_t = P_x \times \left(\frac{d_0}{x}\right)^\alpha = P_y \times \left(\frac{d_0}{y}\right)^\alpha \quad (6.5)$$

Et puisque  $y = 2x$ , equation 6.5 devient :

$$P_y = P_x \times 2^\alpha \quad (6.6)$$

En utilisant l'équation 6.3 et l'équation 6.4, nous pouvons maintenant calculer l'énergie requise pour transmettre un message entre  $A$  et  $B$ . Pour la première topologie, nous avons un chemin avec  $n$  sauts. Ainsi, on peut exprimer la puissance requise pour transmettre des données à partir de  $A$  vers  $B$ , en utilisant  $n$ -sauts,  $P_{nh}$ , par :

$$P_{nh} = \overbrace{(0P_r + P_x)}^A + \overbrace{(P_r + P_x) \dots + (P_r + P_x)}^{(n-1) \text{ nodes}} + \overbrace{(P_r + 0P_x)}^B$$

elle peut être écrite, aussi de la forme :

$$P_{nh} = nP_x + nP_r \quad (6.7)$$

Qui peut être écrite comme :

$$P_{nh} = 2m \times P_x + 2m \times P_r \quad (6.8)$$

on connaît depuis equation 6.6, que  $P_x = \frac{P_y}{2^\alpha}$ , donc l'équation 6.8 devient :

$$P_{nh} = m \times P_y \left[ \frac{1}{2^{\alpha-1}} + 2 \frac{P_r}{P_y} \right] \quad (6.9)$$

De la même manière, on peut avoir l'énergie nécessaire pour transmettre des données de  $A$  vers  $B$ , en utilisant  $m$  sauts,  $P_{mh}$  :

$$P_{mh} = mP_y + mP_r \quad (6.10)$$

qui est égale à :

$$P_{mh} = m \times P_y \left[ 1 + \frac{P_r}{P_y} \right] \quad (6.11)$$

## 6.6 Comparaison entre les courts-sauts et les longs-sauts

Dans cette section, nous allons présenter notre résultat principal [122]. Calculons, maintenant  $P_{nh} - P_{mh}$  :

$$P_{nh} - P_{mh} = m \times P_y \left[ \frac{1}{2^{\alpha-1}} + 2\frac{P_r}{P_y} \right] - m \times P_y \left[ 1 + \frac{P_r}{P_y} \right]$$

Qui est égal à :

$$P_{nh} - P_{mh} = m \times P_y \left[ \frac{1}{2^{\alpha-1}} - 1 + \frac{P_r}{P_y} \right] \quad (6.12)$$

Et puisque  $m \times P_y$  est toujours positif, alors  $P_{nh} - P_{mh} > 0$  si et seulement si :

$$\frac{1}{2^{\alpha-1}} - 1 + \frac{P_r}{P_y} > 0 \quad (6.13)$$

Donc :

$$\frac{P_r}{P_y} > 1 - \frac{1}{2^{\alpha-1}} \quad (6.14)$$

Ce qui veut dire :

$$y < \frac{t}{\sqrt[\alpha]{1 - \frac{1}{2^{\alpha-1}}}} \quad (6.15)$$

Et comme  $y = 2x$ , equation 6.15 devient :

$$x < \frac{t}{2 \times \sqrt[\alpha]{1 - \frac{1}{2^{\alpha-1}}}} \quad (6.16)$$

Et puisque  $y > x$ , les inégalités 6.15 and 6.16 signifient que la consommation d'énergie optimale peut être obtenue lorsque la longueur des sauts est comprise dans l'intervalle :

$$\left] \frac{t}{2 \times \sqrt[\alpha]{1 - \frac{1}{2^{\alpha-1}}}}, \frac{t}{\sqrt[\alpha]{1 - \frac{1}{2^{\alpha-1}}}} \right[$$

## 6.7 Simulations et validation des résultats

Afin de valider la condition présentée dans la section précédente, nous présentons, dans cette section, des simulations avec deux réseaux de capteurs sans fil pour vérifier l'analyse théorique, le premier réseau est composé des capteurs *Mica2* et le deuxième est composé des capteurs *Mica2dot*.

Nous utilisons, dans notre simulation, les caractéristiques expérimentales de ces deux capteurs, afin d'évaluer la puissance consommée par *Mica2* et *Mica2dot* pour transmettre des données à partir d'un noeud à un autre, nous allons utiliser les caractéristiques physiques présentées dans [133] et décrites dans le Tableau 6.2 :

	<i>Mica2</i>	<i>Mica2dot</i>
Reception	16 mA	12mA
Transmission	18 mA	14mA
Transmission range	55 m	135m
Maximum range of transmission	70m	230 m

TABLE 6.2 – Caractéristiques expérimentales des capteurs *Mica* et *Mica2dot*

### Premier cas d'étude

Pour simuler le comportement des capteurs *Mica2*, nous supposons que les réseaux est composés par les noeuds *Mica2* alignés (Émetteur-Relais-Sink) avec une longueur de 760 mètres (distance entre l'émetteur et le Sink), et nous calculons l'énergie consommée par ces réseaux avec un nombre différent de sauts, ce qui signifie différentes longueurs de sauts, ce qui signifie aussi, un nombre différent de noeuds intermédiaires.

Le calcul de la puissance totale consommée pour ces différents réseaux, avec

des différentes longueurs de sauts (c'est-à-dire différent nombre de noeuds intermédiaires), donne les résultats présentés dans le tableau 6.3.

<i>Nombre de sauts</i>	<i>Longueur de saut m</i>	<i>Énergie consommée mA</i>
6	126.6666	662.7732
8	95.0	557.59
10	76.0	503.672
11	69.0909	488.4290
12	63.3333	475.3866
14	54.2857	466.9028
16	47.5	466.2968
18	42.2222	476.9244
20	38.0	491.836
21	36.1904	499.6533
22	34.5454	503.3204
24	31.6666	521.2308
26	29.2307	546.1027
28	27.1428	569.4514
30	25.3333	591.5625

TABLE 6.3 – Consommation d'énergie Vs longueur de sauts

On peut conclure depuis le tableau 6.3, que le nombre optimal des sauts qui garantit la consommation minimale d'énergie est de 16 sauts. (i.e longueur de saut équivalente à 47.5 m), ce qui est vérifié par notre condition qui limite les longueurs des sauts assurant une consommation minimale d'énergie dans l'intervalle ]36.6678, 73.3357[. Pour une lecture facile des résultats nous présenterons le tableau 6.3 par la figure 6.3.

### Deuxième cas d'étude

De la même manière, nous simulons le comportement des capteurs *Mica2dot*; nous calculons la puissance consommée pour des différents réseaux alignés, avec des différentes longueurs de sauts. Nous avons obtenu les résultats présentés dans le tableau 6.4.

D'après le tableau 6.4, nous pouvons obtenir le nombre de sauts qui mini-

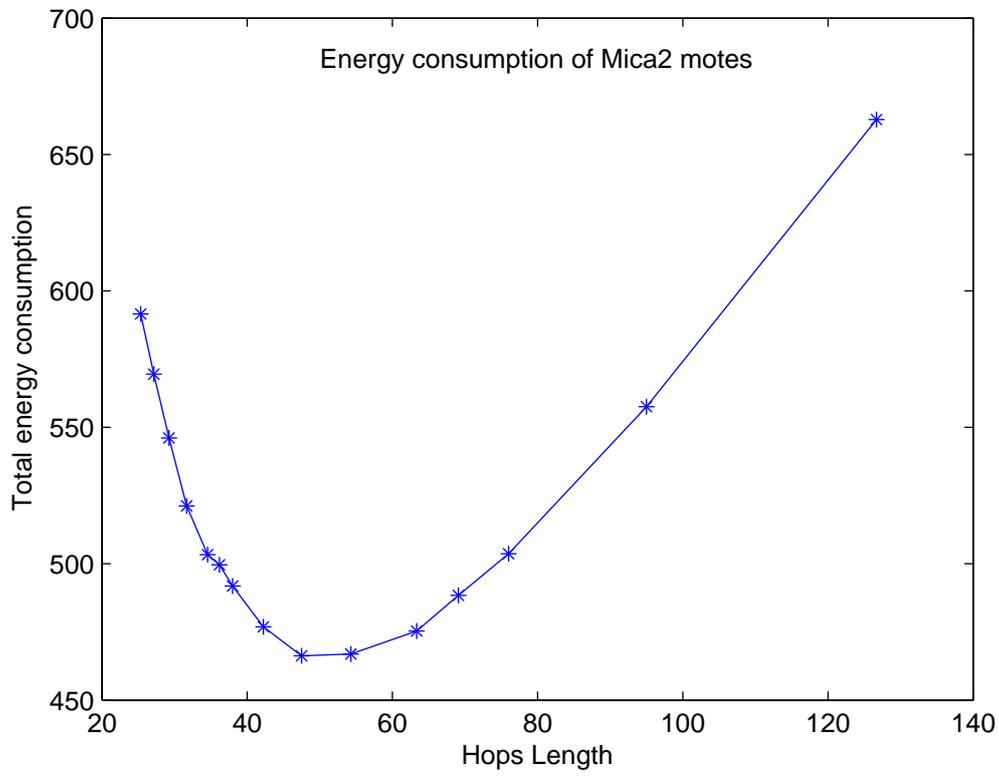


FIGURE 6.3 – Mica2 : La consommation d'énergie Vs longueur de sauts

mise la consommation d'énergie, qui équivaut à 6 sauts (c-à-d longueur de saut = 126,6666  $m$ ). Ce résultat est vérifié, également, par notre condition qui limite la longueur des sauts qui assurent la consommation d'énergie minimale dans l'intervalle  $]88.3883, 176.7766[$ . Pour faciliter la lecture des résultats, nous présentons les résultats sur la figure 6.4 :

## 6.8 Conclusion

Dans ce chapitre, nous avons analysé le problème de saut et de distance entre les noeuds dans les RCSFs et les résultats de simulation montrent l'impact du choix de la longueur des sauts. Nous avons montré, aussi, que la condition exprimée dans ce chapitre, garantit une plus grande efficacité énergétique, et nous aide à prolonger la

<i>Hops number</i>	<i>Hop length</i> m	<i>Total power consumed</i> mA
2	380.0	245.7984
4	190.0	158.8992
5	152.0	148.7193
6	126.6666	145.1566
8	95.0	151.4496
10	76.0	164.3596
11	69.0909	172.3269
12	63.3333	180.5783
14	54.2857	199.3528
16	47.5	219.1441
18	42.2222	240.3855
20	38.0	262.1798
22	34.5454	283.5317
24	31.6666	305.7131
26	29.2307	328.7930
28	27.1428	351.6764
30	25.3333	374.4000

TABLE 6.4 – Consommation d'énergie Vs longueur de sauts

durée de vie des réseaux. En outre, en utilisant la longueur minimale de sauts peut être plus efficace dans le cas de défaillances qui nécessite la retransmission. Dans un futur travail, nous allons nous concentrer sur le problème de routage des données par le chemin le moins coûteux en termes d'énergie, en utilisant la condition présentée.

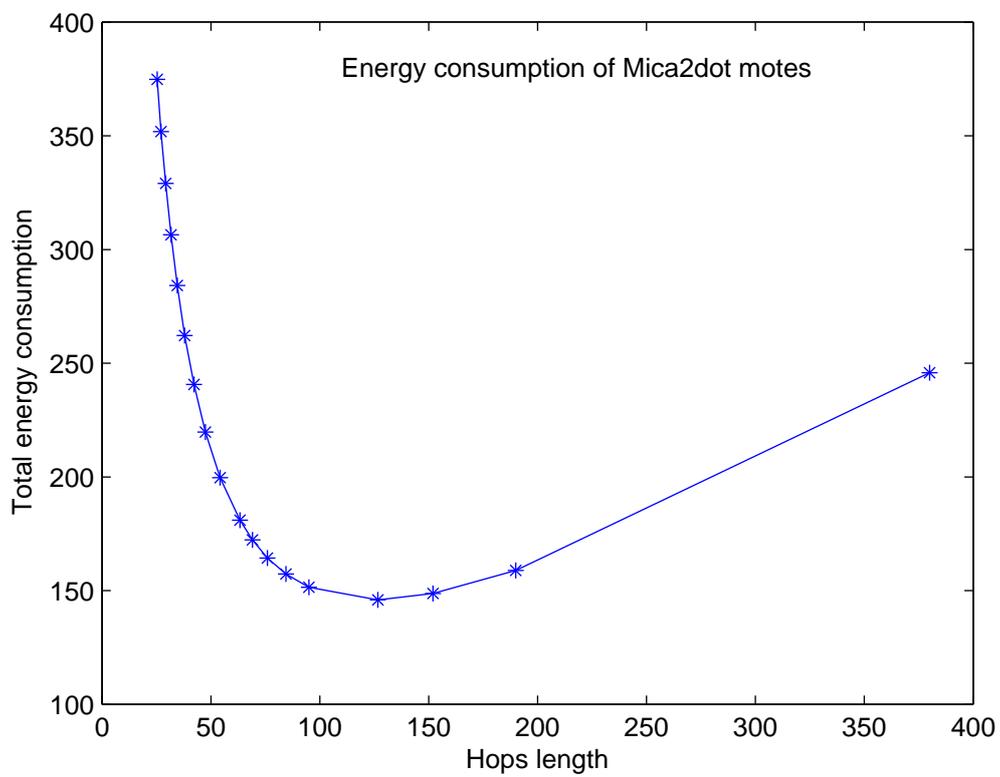


FIGURE 6.4 – Mica2dot : La consommation d'énergie Vs longueur de sauts

# Un clustering génétique, centralisé et dynamique pour les RCSFs

---

## 7.1 Résumé

Afin de minimiser la consommation d'énergie provoquée par les communications sans fil dans les réseaux de capteurs, l'utilisation de la technique du clustering a prouvé sa grande efficacité. Le problème reste à déterminer le nombre des cluster-chefs et leurs distributions dans le réseau pour assurer une consommation minimale d'énergie et une meilleure couverture réseau. Contrairement à l'algorithme LEACH (*Low-Energy Adaptive Clustering Hierarchy*) qui fixe à l'avance le nombre des cluster-chefs et qui ne garanti pas la couverture de l'ensemble du réseau, dans ce chapitre, nous avons proposé un algorithme de clustering centralisé et dynamique basé sur les AGs, afin d'optimiser la configuration du clustering (le nombre des cluster-chefs ainsi que leurs distributions) pour limiter la perte d'énergie et assurer une meilleure couverture. Les résultats de simulation obtenus montrent que la technique proposée surmonte ceux de l'algorithme LEACH.

## 7.2 Introduction

Le clustering consiste à grouper les capteurs dans plusieurs clusters, de telle sorte que chaque cluster a un seul cluster-chef et au moins un cluster-membre. Dans chaque cluster, les cluster-membres rassemblent des informations depuis le champ de surveillance et les envoient à leurs cluster-chef. A son tour, le cluster-chef traite les données reçues de ses membres et les envoie au sink. La figure 7.1 montre un réseau de capteurs sans fil structuré en trois clusters, chaque cluster contient un seul

cluster-chef et au moins un cluster-membre. Chaque noeud cluster-membre envoie les informations recueillies à son cluster-chef, qui à son tour les communique à la station de base.

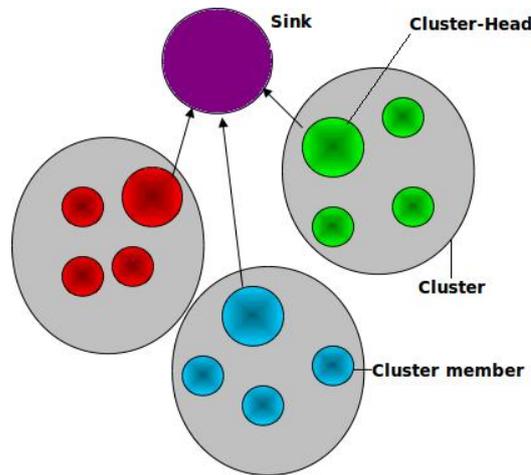


FIGURE 7.1 – Exemple de RCSF structuré en 3 clusters

Le clustering peut être statique ou dynamique. Dans un scénario statique, le nombre des cluster-chefs est fixé ce qui les amènent à épuiser rapidement leurs batteries, ce qui rend aussi le clustering statique impropre pour les RCSFs [70]. En effet, le réseau devient non fonctionnel dans l'absence de ses cluster-chefs. Dans le cas de clustering dynamique, les clusters changent au fil du temps, ainsi une consommation d'énergie uniforme est assurée entre les noeuds et par conséquent une prolongation de la durée de vie du réseau est garantie.

Notre objectif, dans cette thèse, est de maximiser la durée de vie du réseau (défini comme l'intervalle de temps qui sépare le moment du déploiement des noeuds à l'instant où un pourcentage donné de noeuds sont morts [134]) en minimisant la consommation d'énergie moyenne de tous les noeuds et assurer une bonne couverture réseau. Pour arriver à cette fin, chaque noeud du réseau peut devenir un cluster-chef. Pour atteindre cet objectif, nous avons utilisé les algorithmes génétiques comme outil pour trouver une solution optimale à une fonction multi-objectifs (Énergie, distance,

couverture réseau, etc), en déterminant dans chaque cycle, si un noeud peut être choisi, ou non, pour jouer le rôle d'un cluster-chef.

### 7.3 Optimisation multi-objectifs

Résoudre un problème d'optimisation consiste à trouver là où les meilleures solutions vérifiant un ensemble de contraintes et d'objectifs définis par l'utilisateur. Pour déterminer si une solution est meilleure qu'une autre, il est nécessaire que le problème introduise un critère de comparaison. Ainsi, la meilleure solution, appelée aussi solution optimale, est la solution ayant obtenu la meilleure évaluation au regard du critère défini. Les problèmes d'optimisation sont utilisés pour résoudre de nombreux problèmes comme le traitement d'images, la conception des systèmes, etc.

Il existe de nombreuses méthodes d'optimisation classiques pour résoudre de tels problèmes (méthode du gradient, méthode du gradient conjugué, méthode du simplexe, etc), applicables lorsque certaines conditions mathématiques sont satisfaites : ainsi, la programmation linéaire traite efficacement le cas où la fonction objectif est linéaire, ainsi que les contraintes s'expriment linéairement en fonction des variables de décision. Malheureusement, les situations rencontrées en pratique comportent souvent une ou plusieurs complications, qui mettent en défaut ces méthodes : par exemple, la fonction objectif peut être non linéaire, ou même ne peut pas s'exprimer analytiquement en fonction des paramètres ; ou encore, le problème peut exiger la considération simultanée de plusieurs objectifs non linéaire contradictoires [135]. En effet, La majorité de ces problèmes sont de qualités difficiles, car leur résolution nécessite l'utilisation des algorithmes évolutionnaires, et il n'est en général pas possible de fournir dans tous les cas une solution optimale dans un temps raisonnable [135].

Lorsqu'un seul critère est donné, par exemple un critère de minimisation de coût, la solution optimale est clairement définie, c'est celle qui a le coût minimal.

Malheureusement, dans de nombreuses situations rencontrées en pratique, un seul critère peut être insuffisant et le problème peut exiger la considération simultanée de plusieurs objectifs, comme le cas de définition des clusters, dans les RCSFs, qui repose sur plusieurs paramètres. L'optimisation multi-objectifs sur laquelle ce chapitre se repose, traite le cas de la présence des solutions simultanées de plusieurs objectifs influant sur la conception des clusters dans les RCSFs (tels que : l'énergie, la distance de communication, la couverture réseau, etc) [136].

On distingue en réalité deux types de problèmes d'optimisation : les problèmes combinatoires (ou problèmes discrets) et les problèmes à variables continues. Pour clarifier, citons quelques exemples : parmi les problèmes combinatoires, on trouve le célèbre problème du voyageur de commerce où on cherche à minimiser la longueur de sa tournée, en visitant un certain nombre de villes, avant de retourner à sa ville de départ. Comme problèmes continus, citons par exemple la restauration optimale d'une image à partir d'une image brouillée.

Dans le domaine de l'optimisation combinatoire, un grand nombre de méta-heuristiques, qui produisent des solutions proches de l'optimum, ont été développées [135]. Ces méthodes, qui comprennent notamment la méthode du recuit simulé, les algorithmes génétiques, la méthode de recherche tabou, les algorithmes de colonies de fourmis, etc, sont apparues, à partir des années 1980, avec une ambition commune : résoudre au mieux les problèmes d'optimisation difficile. Les méta-heuristiques ne s'excluent pas mutuellement : en effet, dans l'état actuel de la recherche, il est le plus souvent impossible de prévoir avec certitude l'efficacité d'une méthode donnée, quand elle est appliquée à un problème donné. De plus, la tendance actuelle est l'émergence de méthodes hybrides, qui s'enfoncent de tirer parti des avantages spécifiques d'approches différentes en les combinant.

Dans la plupart des problèmes du monde réel, il ne s'agit pas d'optimiser seulement un seul critère mais plutôt d'optimiser simultanément plusieurs critères et qui

sont généralement conflictuels. Dans les problèmes de conception, par exemple, il faut le plus souvent trouver un compromis entre des besoins technologiques et des objectifs de coût. L'optimisation multi-objectif consiste donc à optimiser simultanément plusieurs fonctions [137].

### 7.3.1 Formulation

Un problème multi-objectifs peut être défini comme suit [137] :

$$\min/\max F(x) = (f_1(x), f_2(x), \dots, f_m(x)) \quad \text{avec } m \geq 2 \quad (7.1)$$

où  $m$  est le nombre de fonctions à optimiser,  $x = (x_1, \dots, x_n)$  est le vecteur des variables de décisions,  $F(x) = (f_1(x), f_2(x), \dots, f_m(x))$  est le vecteur des critères à optimiser.

Les premières études concernant l'optimisation multi-objectifs transformaient une succession de problèmes mono-objectif les à un problèmes multi-objectifs. Pour cela, un ordre d'importance sur les objectifs pouvait être donnée, et l'optimisation consistait à optimiser un objectif sans dégrader les valeurs déjà obtenues pour les objectifs plus prioritaires. En effet, L'optimisation multi-objectif autorise des degrés de liberté qui manquaient en optimisation mono-objectif linéaire [136].

Les problèmes d'optimisation issus d'une agrégation linéaire des objectifs, chacun pouvant avoir un poids représentant son importance. Lorsque l'on se trouve dans un réel contexte multi-objectif, il n'est pas toujours possible de trouver un ordre d'importance sur les critères. Il est alors nécessaire de rechercher les solutions de meilleur compromis entre les objectifs [135]. Ainsi la fonction 7.1, peut introduire des degrés d'importance sur ses objectif, elle peut s'écrive :

$$\min/\max F(x) = (f_1^{\alpha_1}(x), f_2^{\alpha_2}(x), \dots, f_m^{\alpha_m}(x)) \quad \text{avec } m \geq 2 \quad (7.2)$$

où  $m$  est le nombre de fonctions à optimiser,  $x = (x_1, \dots, x_n)$  est le vecteur des variables de décisions,  $F(x)$  est la fonction non linéaire à optimiser. Les  $\alpha_i \geq 1$  avec  $i \in \{1, \dots, m\}$  sont les degrés d'importance de chaque objectif.

Ce chapitre, qui contient notre principale contribution, considère en particulier l'optimisation multi-objectifs ( quatre fonctions objectifs) afin d'assurer en fonction des noeuds :

1.  $f_1(\text{noeuds})$  =La couverture réseau,
2.  $f_2(\text{noeuds})$  =Le nombre optimal des cluster-chefs
3.  $f_3(\text{noeuds})$  =Le centrisme des cluster-chefs dans leurs clusters
4.  $f_4(\text{noeuds})$  =Une consommation d'énergie équilibrée entre les noeuds capteurs d'un réseau

Où les  $f_i$  sont les fonctions à optimisée et les noeuds sont les variables de décisions. D'où la fonction multi-objectifs non linéaire est définie par la fonction 7.4 :

$$\min/\max F(\text{noeuds}) = (f_1^{\alpha_1}(\text{noeuds}), f_2^{\alpha_2}(\text{noeuds}), f_3^{\alpha_3}(\text{noeuds}), f_4^{\alpha_4}(\text{noeuds})) \quad (7.3)$$

Où les  $\{\alpha_i\}_{i=1}^4$  représentent les degrés d'importance des fonctions mono-objectif, qui sont définis dans les sections suivantes.

Lorsque nous résolvons notre problème d'optimisation multi-objectifs, nous obtenons une multitude de solutions. Seul un nombre restreint de ces solutions va nous intéresser. Pour qu'une solution soit intéressante, il faut qu'il existe une relation de dominance entre la solution considérée et les autres solutions [135]. que nous allons définir dans la section suivante.

### 7.3.2 Notion de dominance

On dit que le vecteur  $x = (x_1, x_2, \dots, x_k)$  domine le vecteur  $y = (y_1, y_2, \dots, y_k)$  si :

- $x$  est au moins aussi bon que  $y$  sur tous les objectifs et,
- $x$  est strictement meilleur que  $y$  sur au moins un objectif.

D'une manière équivalente, nous avons : La solution  $x$  du problème ( $P$ ) est dite dominer une autre solution  $y$ , si les conditions suivantes sont vérifiées :

$$f_l(x) \leq f_l(y) \quad \forall l \in \{1, \dots, k\} \quad \text{et} \quad \exists l \in \{1, \dots, k\} \quad \text{tel que} \quad f_l(x) < f_l(y).$$

Si la solution  $x$  domine la solution  $y$ , nous allons écrire  $x < y$ .

## 7.4 Source de l'efficacité des méta-heuristiques

Lorsque l'espace des solutions possibles présente une structure aussi tourmentée, il est difficile de repérer le minimum global  $c_*$ . Les algorithmes itératifs classiques présentent, généralement, une grande faiblesse face aux problèmes combinatoires difficiles ceci est du au qu'ils conduisent, en général, à un minimum local.

Fonction objectif

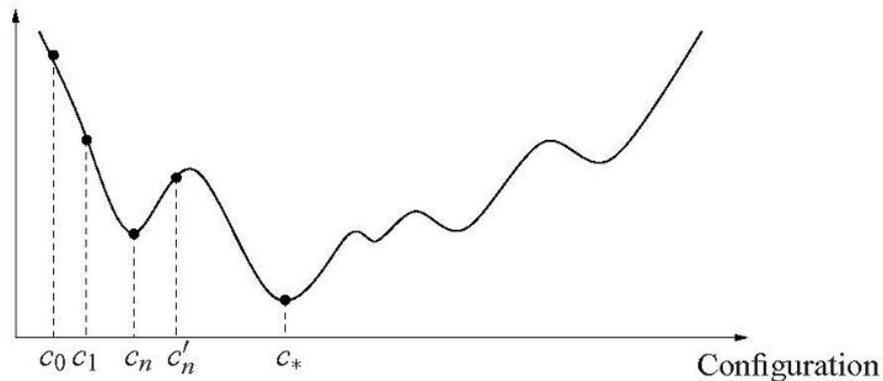


FIGURE 7.2 – Allure d'une fonction objectif d'un problème d'optimisation difficile

Pour surmonter l'obstacle des minima locaux, une autre idée s'est montrée très fructueuse, au point qu'elle est à la base de toutes les méta-heuristiques dites de voisinage (recuit simulé, méthode tabou, AGs), il s'agit d'autoriser, de temps en temps, des mouvements de remontée, autrement dit d'accepter une dégradation temporaire de la situation, lors du changement de la configuration courante. C'est le cas si l'on passe de  $c_n$  à  $c'_n$ , voir figure 7.2. Un mécanisme de contrôle des dégradations (spécifique à chaque méta-heuristique) permet d'éviter la divergence du procédé. Il

devient, dès lors, possible de s'extraire du piège que représente un minimum local, pour partir explorer une autre vallée plus prometteuse.

Les meta-heuristiques distribuées (telles que les algorithmes génétiques) ont elles aussi des mécanismes permettant la sortie hors d'un puits local de la fonction objectif. Ces mécanismes (comme la mutation dans les algorithmes génétiques) affectant une solution viennent, dans ce cas, seconder le mécanisme collectif de lutte contre les minima locaux, que représente le contrôle en parallèle de toute une population de solutions [135].

## 7.5 L'historique de clustering

L'idée d'utiliser le clustering a été adoptée par de nombreux auteurs. L'algorithme de clustering *LCA* a été l'une des premières approches [138]. Dans l'algorithme *LCA*, chaque noeud possède un *ID* unique, avec lequel un noeud devient un cluster-chef si son *ID* est le plus élevé dans son voisinage. Cependant ce clustering est statique ce qui épuise rapidement les batteries des noeuds choisis pour être des cluster-chefs.

*Leach* (Low-Energy Adaptive Clustering Hierarchy) est l'algorithme de clustering le plus populaire pour les réseaux de capteurs [88]. LEACH permet à un pourcentage fixe de noeuds pour devenir des cluster-chefs (à savoir, 5% du nombre total des noeuds déployés) et conduit à la création des clusters de manière distribuée, dans cet algorithme les noeuds prennent des décisions autonomes. Chaque noeud décide de devenir un cluster-chef avec une probabilité  $p$ . Un noeud qui ne devient pas un cluster-chef détermine son cluster d'appartenance en choisissant le cluster-chef le plus proche, en se basant sur l'intensité du signal reçu.

En général, LEACH fournit une faible consommation d'énergie et une distribution d'énergie uniforme entre les noeuds. Cependant LEACH a aussi quelques inconvénients. En raison de la sélection probabiliste des cluster-chefs, un noeud avec

une très faible énergie peut être sélectionné comme un cluster-chef. En plus, comme la sélection des cluster-chefs est aléatoire, les cluster-chefs choisis peuvent être situés dans la même zone, de telle sorte qu'une bonne couverture ne peut pas être garantie. En effet, certains noeuds seront déconnectés du réseau (par exemple, ils ne seront pas des membres dans aucun cluster). En outre, l'utilisation d'un pourcentage fixe de cluster-chefs peut conduire, l'ensemble du réseau, à une consommation d'énergie plus élevée, car le nombre des cluster-chefs dépend de plusieurs facteurs, tels que la densité des noeuds, l'énergie disponible, etc [64, 120].

*EEHC* est un algorithme de clustering hiérarchique aléatoire et distribué, dont le but est de maximiser la durée de vie du réseau [64]. Cet algorithme est exécuté en deux niveaux. Dans le premier niveau, appelé initial, des noeuds bénévoles, qui ne font partie d'aucun cluster, peuvent devenir des cluster-chefs avec une probabilité  $p$ , après avoir pris une décision ils annoncent leurs décisions à leurs voisins. Les noeuds qui ne reçoivent pas une décision, dans un intervalle de temps donné  $t$ , deviennent des cluster-chefs forcés. Dans le deuxième niveau, appelé étendu, l'algorithme de clustering est récursivement répété pour former le clustering hiérarchique, où de nouveaux cluster-chefs sont choisis parmi les cluster-chefs déjà choisis dans le premier niveau, jusqu'à ce que station de base est atteinte.

Dans [134], les auteurs ont proposé un protocole de routage basé sur les AGs (*GROUPE*) : en particulier, ils utilisent les AGs et le recuit simulé (à la place de la chaîne gourmande (*Greedy chain*)) pour sélectionner des chemins efficaces pour le routage, mais ces propositions restent très gourmande en mémoire et en processeurs pour être implémenter réellement dans les RCSFs.

## 7.6 Le model réseau utilisé

Les conditions et les hypothèses du modèle réseau utilisé, dans ce chapitre, sont considérés conformes et similaires à celles présentés dans [103]. Plus précisément, ils

peuvent être résumés comme suite :

- La station de base est fixe, elle n'a pas de contrainte d'énergie et elle a une grande capacité de calcul.
- Tous les noeuds déployés ont une puissance d'énergie limitée et sont homogènes.
- La puissance de traitement des données est très faible par rapport à la puissance nécessaire pour transmettre et recevoir des données.

Les spécifications des communications radio des noeuds sont définies comme dans [53, 99, 103, 122, 134] et présentées dans la figure 7.3. En particulier, nous supposons que le module radio dissipe :  $E_{elec} = 50$  nJ/bit dans la transmission et la réception ; et  $\varepsilon_{amp} = 100$  pJ/bit/m<sup>2</sup> par l'amplificateur de l'émetteur. On considère, aussi, que les communications se font dans un espace libre, ainsi, afin de transmettre un message de  $K$  bits sur une distance  $d$  (dimension :  $m$ ) un noeud consomme la quantité d'énergie présentée dans l'équation 7.4 :

$$E_{Tx}(k, d) = E_{elec} \times k + \varepsilon_{amp} \times k \times d^2. \quad (7.4)$$

Et en recevant un message de  $k$ -bit, un noeud consomme la quantité présentée dans l'équation 7.5 :

$$E_{Rx}(k) = E_{elec} \times k. \quad (7.5)$$

## 7.7 L'approche proposée

### 7.7.1 La présentation du problème

Dans un RCSF structuré en clusters, l'utilisation d'un nombre réduit de cluster-chefs, exige les noeuds cluster-membres à utiliser une grande portée d'émission radio, ainsi pour transmettre les données recueillies à leurs cluster-chefs l'énergie de leurs batteries est rapidement épuisée. En outre, Si un grand nombre de cluster-chefs est utilisé, ceci conduit dans la plupart du temps à un réseau à un seul-saut (la plupart

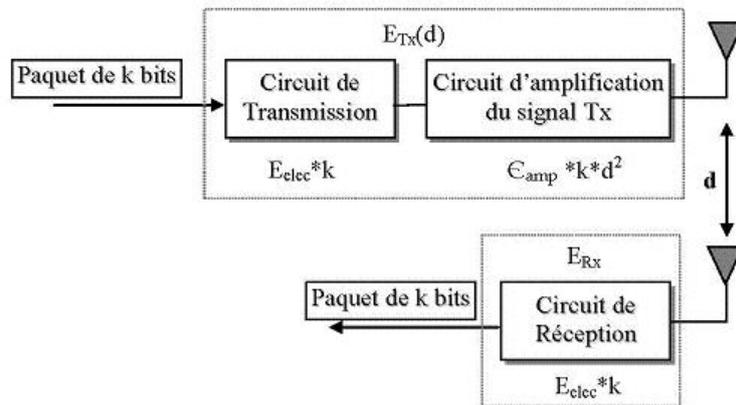


FIGURE 7.3 – Module de transmission/réception

des noeuds sont des cluster-chefs et doivent atteindre la station de base avec un seul-saut), ceci consomme aussi rapidement l'énergie de leurs batteries [53, 99, 122, 139]. Donc, la meilleure stratégie de clustering consiste à déterminer :

1. Le nombre des cluster-chefs
2. Les positions de ces cluster-chefs

En particulier, un noeud peut être promu à être un cluster-chef en fonction de plusieurs paramètres :

1. L'énergie résiduelle
2. Sa distance au Sink
3. La somme des distances qui le sépare à ses cluster-membres

Cela suggère l'utilisation des algorithmes génétique pour trouver la combinaison optimale de ces paramètres.

### 7.7.2 L'algorithme proposé

Dans ce chapitre, nous considérons le clustering dynamique, c-à-d le re-clustering est effectué pour éviter la mort précoce des noeuds cluster-chefs. L'algorithme génétique proposé est exécuté au niveau du sink (c-à-d il est centralisé), en raison de la

capacité de calcul nécessaire, et les résultats obtenus (les paramètres de clustering) sont communiqués aux noeuds. A chaque tour de re-clustering, chaque noeud peut être soit un cluster-chef ou un cluster-membre. Cette approche centralisée permettra d'éviter les principales limitations de LEACH, où le nombre des cluster-chefs est fixé et leur répartition est aléatoire, c-à-d il n'y a pas de coordination dans le choix des cluster-chefs [103].

Afin d'utiliser les algorithmes génétique, un RCSF doit être codifié. En particulier, nous utilisons une représentation binaire, dans laquelle chaque noeud est représenté soit par un 0 (s'il est un cluster-membre) ou par un 1 (s'il est un cluster-chef). Chaque réseau codifiée est appelé un chromosome. Un ensemble de chromosomes est appelé une génération.

L'algorithme génétique utilisé est basé sur l'exploration et l'exploitation de l'espace de recherche entier en utilisant une stratégie évolutive, en outre, il nous aide à trouver une combinaison optimale, des cluster-chefs et des cluster-membres ainsi que leurs distributions dans le champ de déploiement, parmi de nombreuses combinaisons existantes dans l'espace de recherche, en choisissant celle qui assure une consommation d'énergie optimale et une couverture totale du réseau. Chaque solution potentielle est caractérisé par une valeur appelée fitness, qui détermine l'optimalité des solutions. A chaque génération, l'AG conserve les meilleurs chromosomes et abandonne les autres selon leurs fonctions de fitness. Chaque chromosome représente une solution potentielle. L'AG applique ensuite les opérateurs génétiques suivants pour générer de nouveaux descendants [140] :

### 7.7.2.1 Sélection

Le processus de sélection est utilisée pour choisir les meilleurs chromosomes d'une génération. Dans notre simulation, l'algorithme de roulette est utilisé pour effectuer la sélection.

### 7.7.2.2 Croisement

Pour appliquer le croisement, nous choisissons arbitrairement deux chromosomes, nous choisissons, également, deux positions aléatoires dans chaque chromosome choisi et nous effectuons le croisement à deux points entre les chromosomes, pour générer deux nouveaux descendants qui appartiendront à la génération suivante.

### 7.7.2.3 Mutation

La mutation est utilisée pour éviter le problème de super chromosome. Cela signifie que si un chromosome est choisi à plusieurs reprises dans la même génération, le croisement ne peut pas produire de nouveaux chromosomes, étant donné que les parents sont les mêmes chromosomes. Par conséquent, la mutation est utilisée pour modifier arbitrairement des bits, dans chaque chromosome. Plusieurs essais ont été effectuées pour parvenir aux paramètres donnant une meilleure autonomie et une convergence rapide. Les meilleurs probabilités de croisement et de mutation, trouvés, sont respectivement 0.75 et 0.2.

Comme ci-mentionné, chaque chromosome est alors évalué avec une fonction d'adaptation (fonction multi-objectif) qui attribue une chance aux meilleures solutions de survivre. L'aptitude d'un chromosome candidat peut être exprimée, en fonction de plusieurs objectifs, ainsi elle peut être représentée par :

$$\text{Fitness} = f(NNN, NCH, DNCH, RECH) \quad (7.6)$$

Où :

- $NNN$  le nombre des noeuds réseautés ;
- $NCH$  le nombre des cluster-chefs ;
- $DNCH$  la somme des distance entre les cluster-membres (CMs) et leurs cluster-chefs (CHs), c-à-d,

$$DNCH = \sum_{i \in \{\text{CHs}\}} \sum_{j \in \{\text{CMs}\}} \text{Distance}(\text{CH}_i, \text{CM}_j);$$

- $RECH$  est la somme de l'énergie restante (cumulative) dans les cluster-chefs (dimension : [mW]), c-à-d.,

$$RECH = \sum_{i \in \{CHs\}} \text{l'énergie restante au } CH_i.$$

Afin d'optimiser le mécanisme de clustering proposé, nous considérons la fonction de fitness suivante :

$$\begin{aligned} \text{Fitness} = & (NNN)^{\alpha_1} + \left(\frac{NNN}{NCH}\right)^{\alpha_2} \\ & + (RECH)^{\alpha_3} + DNCH^{\alpha_4} \end{aligned}$$

Où la fraction  $\frac{NNN}{NCH}$  représente la dimension moyenne des clusters, et les paramètres exponentielles  $\{\alpha_i\}_{i=1}^4$  doivent être correctement optimisés ; la fraction représente la dimension moyenne du cluster.

La dimension de l'énergie. Par essais et erreurs, a donné la meilleure fonction de fitness (à savoir, la meilleure configuration des exposants) comme suit :

$$\begin{aligned} \text{Fitness} = & (NNN)^6 + \left(\frac{NNN}{NCH}\right)^5 \\ & + (RECH)^2 + DNCH. \end{aligned}$$

## 7.8 L'analyse des performances

Dans cette section plusieurs expériences ont été effectuées pour valider l'algorithme *GCDC* proposé (*GCDC* pour *Genetic Centralized Dynamic Clustering*). Afin de valider l'approche de clustering proposée, nous effectuons une analyse des performances basée sur la simulation, en prenant en considération plusieurs scénarios, dans lesquels on fait varier la densité spatiale des noeuds (le nombre des noeuds) et la position du sink. Dans chaque scénario, un certain nombre de capteurs est déployé de façon aléatoire dans un champ carré qui a comme surface  $100 \times 100 \text{ m}^2$ .

Le Sink, qui est placé au sein du réseau, exécute l'algorithme *GCDC* et informe les capteurs par la configuration des clusters. Après la réception de la décision,

chaque noeud connaît s'il est un cluster-chef ou un cluster-membre. L'algorithme GCDC est périodiquement exécuté par le sink afin d'éviter l'extinction des noeuds et pour ne pas affecter la connectivité du réseau. Nous supposons que tous les noeuds ont des batteries avec une énergie initiale égale à 0,25 J, déployés aléatoirement dans  $100 \times 100 \text{ m}^2$ . La dimension des messages à transmettre est fixé à 1000 bits, plus 50 bits pour les paquets de contrôle.

La figure 7.4 montre, le premier cas d'étude, celui d'un réseau de capteurs sans fil composé de 100 noeuds homogènes, dispersés aléatoirement dans un champ de surveillance. Le Sink est placé au milieu du champ.

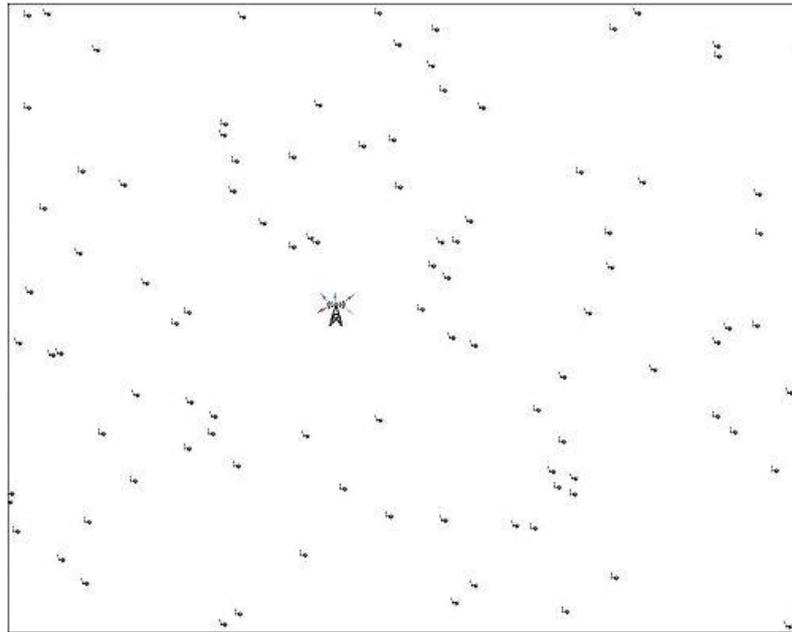


FIGURE 7.4 – Réseau de capteurs sans fil avec 100 noeuds

La figure 7.5 montre, également, un réseau de capteurs sans fil composé de 1000 noeuds homogènes, déployées de façon aléatoire dans une zone de surveillance. Le sink est placé au milieu du champ.

Nous supposons que des événements aléatoires (par exemple, la détection d'un signal acoustique, la détection de mouvement, etc.) se produit dans la zone surveillée. En particulier, chaque événement aléatoire est détecté par le plus proche voisin, qui

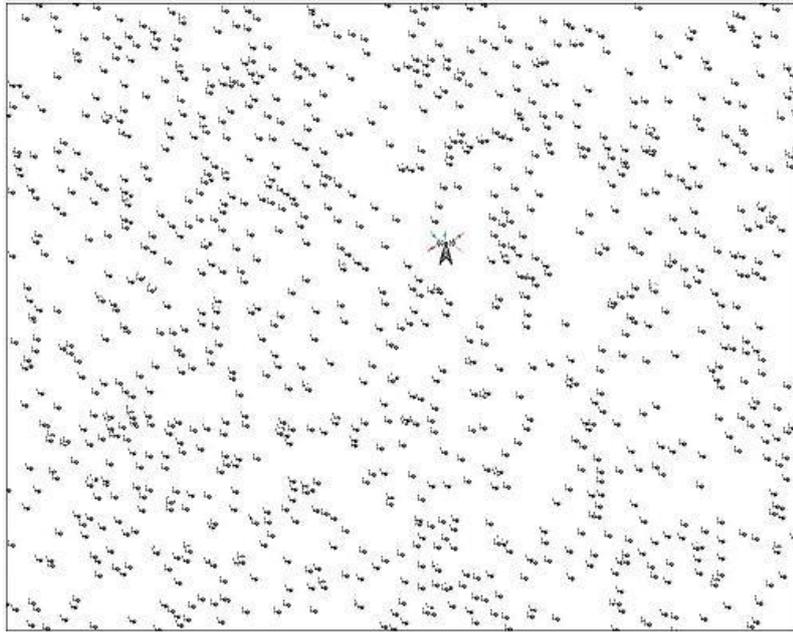


FIGURE 7.5 – Réseau de capteurs sans fil avec 1000 noeuds

doit rapporter cette observation au Sink.

Dans tous les scénarios envisagés, les performances de l'algorithme GCDC est comparée à celles de LEACH. On considère deux RCSFs avec des densités différentes : 100 noeuds (faible densité spatiale) et 1000 noeuds (forte densité spatiale).

La figure 7.6 et la figure 7.7 montrent le même réseau de 100 noeuds déployés après l'application de l'algorithme GCDC et l'algorithme LEACH , nous pouvons conclure à partir des figures que l'utilisation de l'algorithme GCDC offre une meilleure connexion réseau par rapport à l'algorithme LEACH, qui fixe à l'avance le nombre des cluster-chefs à 5% et qui ne garanti pas une répartition uniforme des cluster-chefs dans le réseau, contrairement au GCDC qui utilisent les AGs pour déterminer le nombre et la position appropriés des cluster-chefs pour couvrir l'ensemble du réseau et pour consommer moins d'énergie.

Dans la figure 7.8 et la figure 7.9, l'énergie résiduelle du réseau est représentée en fonction de temps de simulation (exprimée en nombre d'événements produits), compte tenu des deux valeurs pour le nombre de noeuds : (a) 100 et (b) 1000.

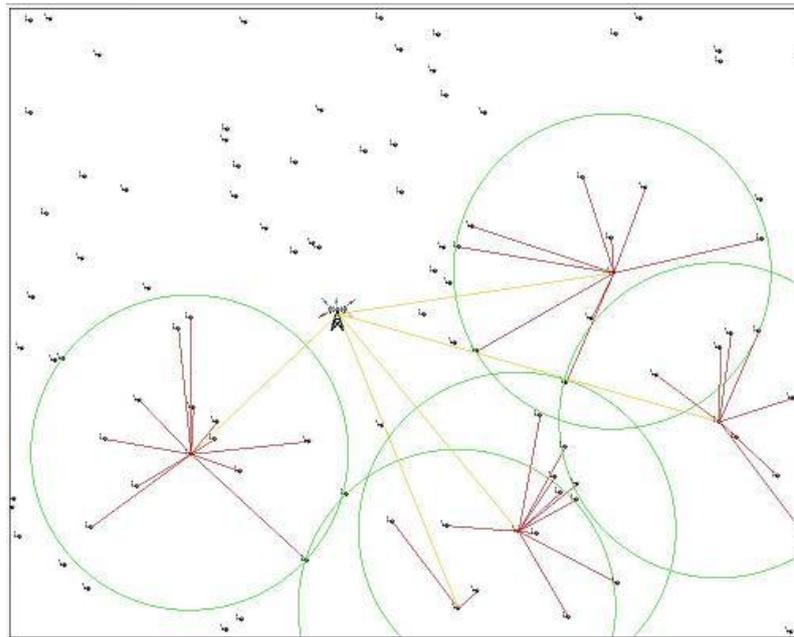


FIGURE 7.6 – Un RCSF de 100 noeuds après l'application de LEACH

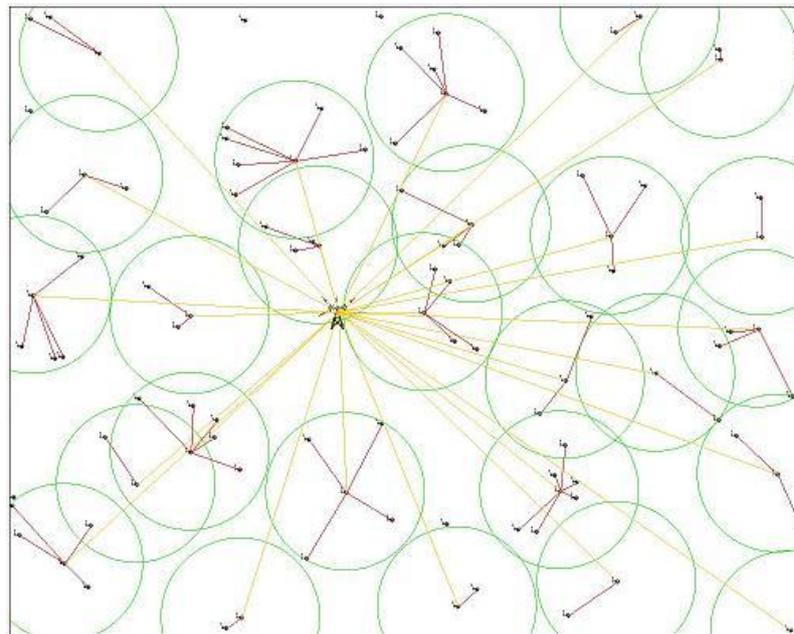


FIGURE 7.7 – Un RCSF de 100 noeuds après l'application de GCDC

On peut observer que l'algorithme GCDC permet de sauver plus d'énergie que LEACH. L'économie d'énergie n'est pas significative au début, mais elle devient plus importante avec le temps. Cela est dû au fait que l'algorithme GCDC met à jour

la topologie du réseau, en utilisant un clustering très efficace. Ce comportement est plus claire dans le scénario avec 100 noeuds (faible densité spatiale).

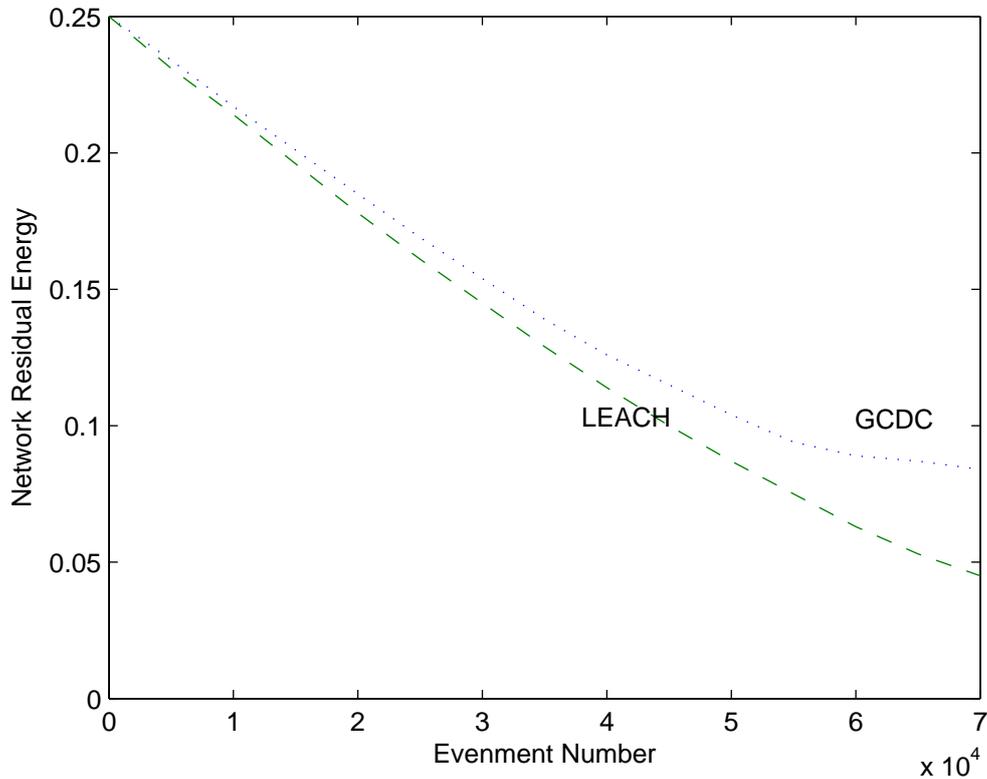


FIGURE 7.8 – L'énergie résiduelle du réseau en fonction de temps de simulation (en termes de nombre d'évènements). Le nombre de noeuds dans le réseau est : (a)100.

Dans la figure 7.10 et la figure 7.11, nous étudions l'évolution de la connectivité du réseau considérant (a) *NNN* (c-à-d, la couverture du réseau) et (b) le nombre des noeuds morts, en fonction du temps de simulation (en termes du nombre des évènements).

Dans les deux cas, le nombre de noeuds dans le RCSF est 100 noeuds (faible densité spatiale).

D'après les résultats de la la figure 7.10, on peut observer que le nombre des noeuds connectés (ie, les noeuds qui sont des cluster-membres ou des cluster-chefs)

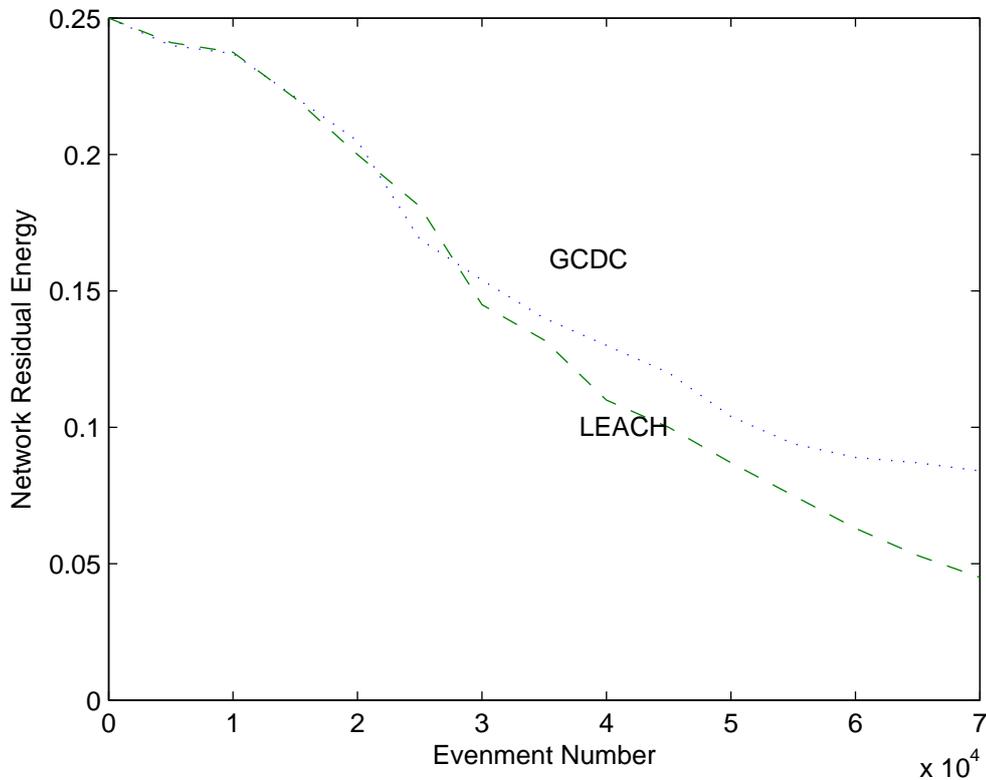


FIGURE 7.9 – L'énergie résiduelle du réseau en fonction de temps de simulation (en termes de nombre d'évènements). Le nombre de noeuds dans le réseau est : (b)1000.

par l'algorithme GCDC est plus grand que celui garanti par LEACH. Ceci est remarquable au début de la simulation, lorsque tous les noeuds (ayant l'énergies initiales) peuvent être connectés, tandis que l'amélioration apportée par GCDC réduit, avec l'avancement du simulation, le nombres des noeuds morts.

La différence de performance est due au fait que LEACH fixe, a priori, le nombre des cluster-chefs à 5% du nombre total des noeuds sans préciser leurs positions, cela conduit probablement aux chevauchements entre les clusters (par exemple, deux cluster-chefs peuvent être l'un près de l'autre), laissant les autres noeuds déconnectés (sans cluster-chefs proche d'eux) . L'algorithme GCDC ne détermine pas a priori le nombre des cluster-chefs, mais, plutôt, l'AG qui détermine le nombre optimal des

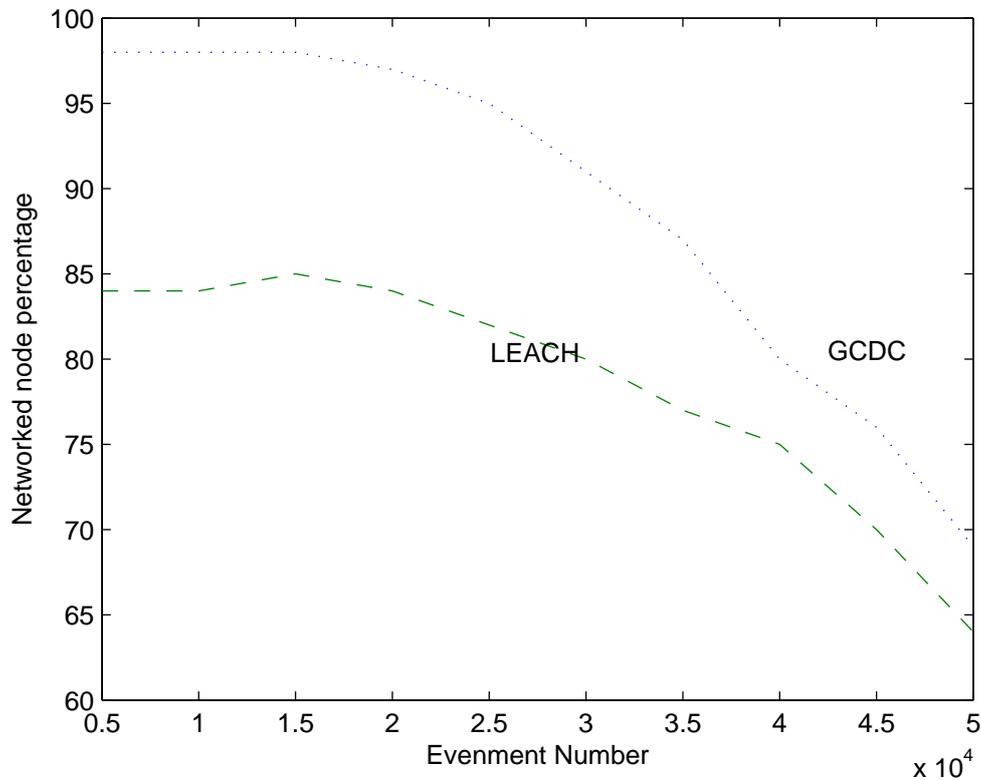


FIGURE 7.10 – (a)-L'évolution de la connectivité du réseau , en termes de  $NNN$ , en fonction du temps de simulation (en termes de nombre des évènements). Le nombre de noeuds dans le RCSF est 100 noeuds

cluster-chefs, avec leurs positions, pour couvrir toute la zone surveillée efficacement. Le nombre des noeuds morts (après l'épuisement de leurs batteries) est représenté sur la la figure 7.11, il est claire que le taux des noeuds morts en utilisant GCDC est inférieur à celui en utilisant LEACH, en raison de l'utilisation de GCDC qui prend en compte l'énergies résiduelles des noeuds et centrisme des cluster-chefs dans leurs clusters.

Dans les figures 7.12 et 7.13, l'évolution de la connectivité du réseau, considérant (a)  $NNN$  (ie, la couverture du réseau) et (b) le nombre des noeuds morts en fonctions du temps de simulation (en termes de nombre de d'évènements), est étudiée dans un scénario avec 1000 noeuds déployés (haute densité spatiale).

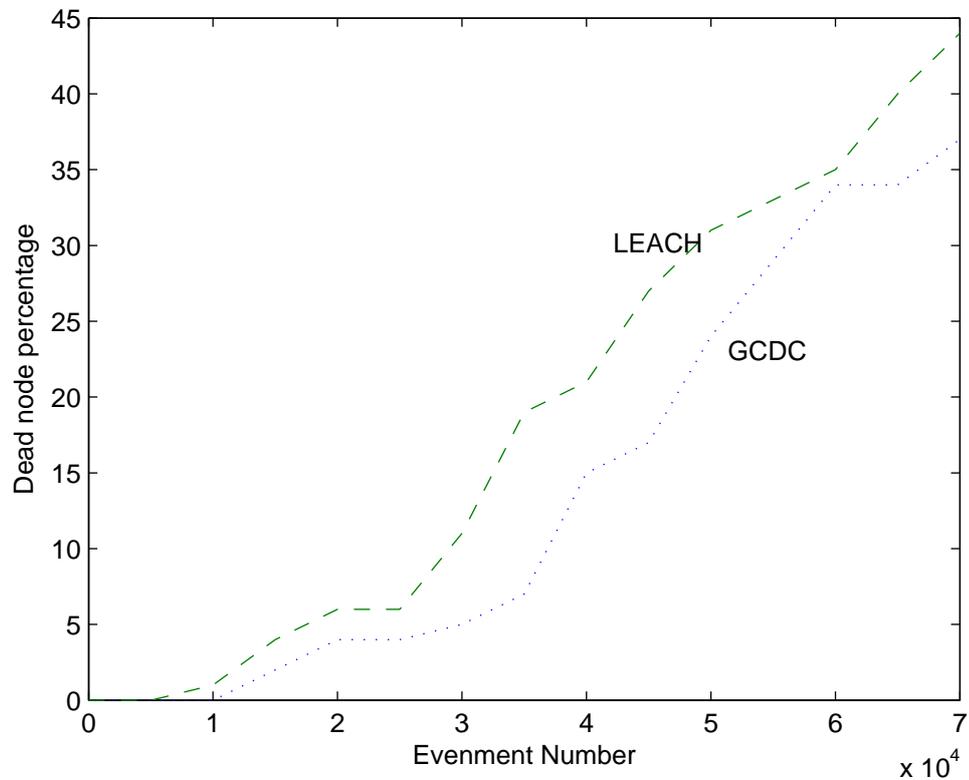


FIGURE 7.11 – (b)-L'évolution de la connectivité du réseau , en termes des noeuds morts, en fonction du temps de simulation (en termes du nombre d'évènements). Le nombre de noeuds dans le RCSF est 100 noeuds

En comparant les résultats de la figure 7.10 avec ceux de la figure 7.12, on peut conclure que l'amélioration des performances, en termes de  $NNN$ , fournit par GCDC est plus efficace dans les réseaux denses (avec une grande densité spatiale). En particulier, d'après les résultats de la figure 7.13, les taux des noeuds morts en utilisant LEACH et GCDC sont approximativement les mêmes dans le début de la simulation, et après un certain nombre d'évènement l'algorithme GCDC montre son efficacité par rapport à LEACH, cela signifie que l'algorithme GCDC est très efficace en re-clustering afin de garantir un haut niveau de connectivité aux noeuds survivants.

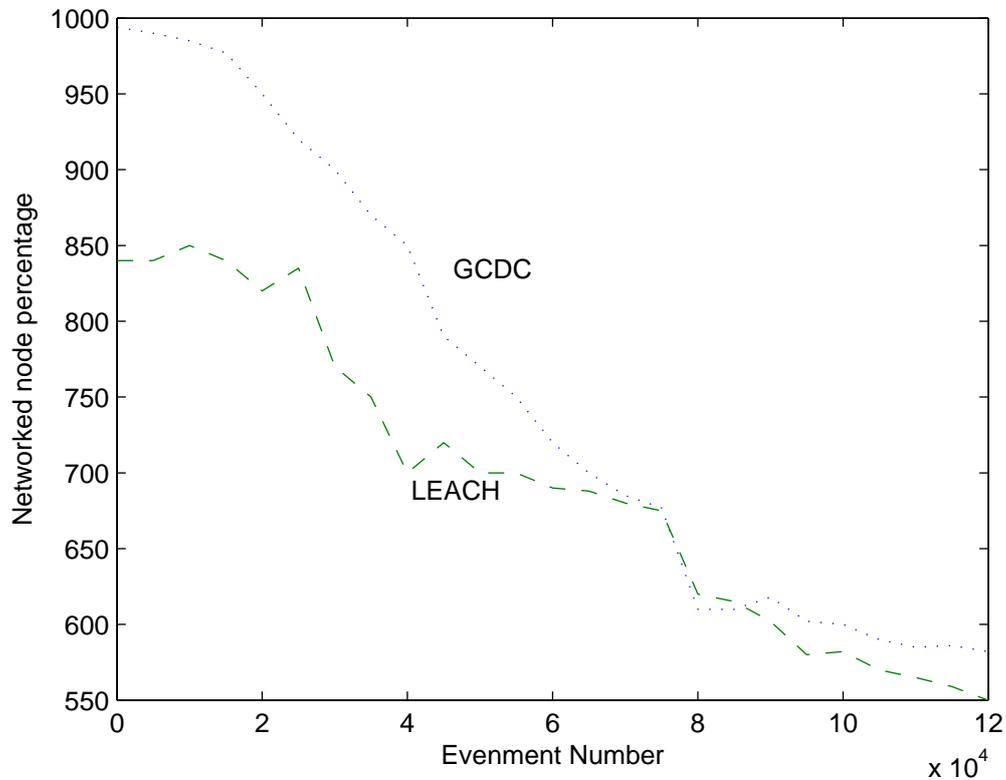


FIGURE 7.12 – (a)-L'évolution de la connectivité du réseau , en termes de  $NNN$ , en fonction du temps de simulation (en termes de nombre d'évènement). Le nombre des noeuds dans le RCSF est 1000 noeuds

## 7.9 Conclusion

Dans ce chapitre, nous avons présenté un nouvel algorithme de clustering, noté clustering génétique, centralisée et dynamique (GCDC), dont nous avons utilisé les algorithmes génétiques, comme outil, pour définir le meilleur nombre des cluster-chefs et leurs emplacements pour assurer une consommation d'énergie minimale et une meilleure couverture réseau. L'algorithme de clustering proposé est ré-exécuté périodiquement au niveau du Sink, en raison de la nécessité d'une grande capacité de calcul, afin de définir une nouvelle combinaison de cluster-chefs et cluster-membres, ceci pour éviter l'extinction précoce des noeuds choisis pour jouer le rôle

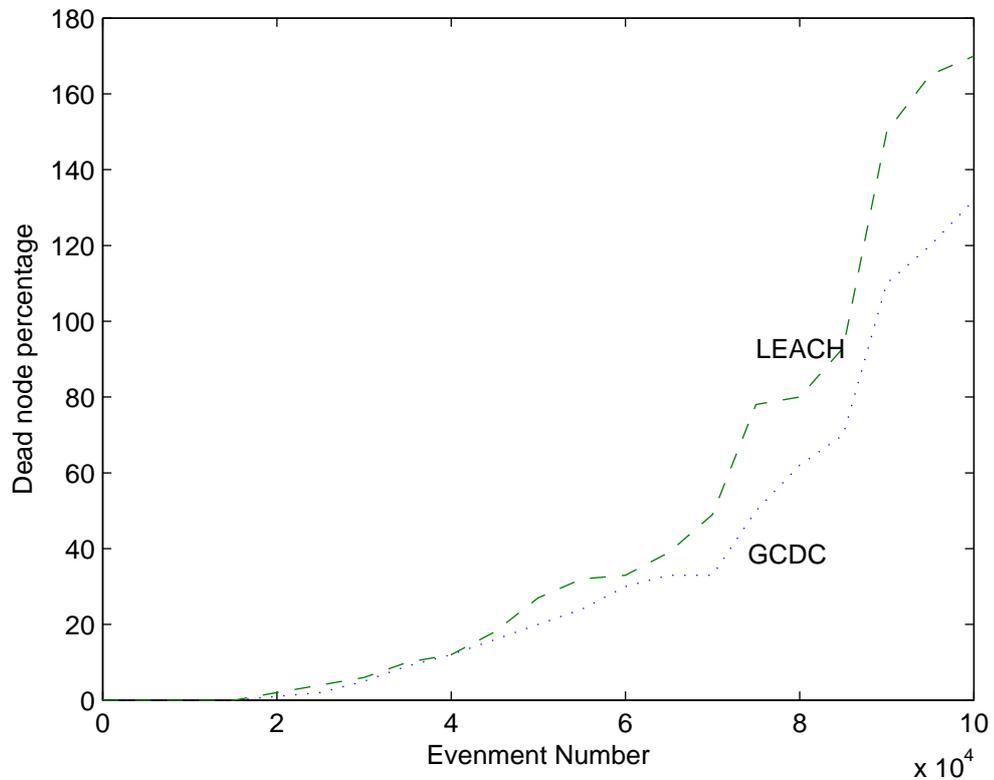


FIGURE 7.13 – (b)-L'évolution de la connectivité du réseau , en termes des noeuds morts, en fonction du temps de simulation (en termes de nombre d'évènement). Le nombre des noeuds dans le RCSF est 1000 noeuds

des clusters-chefs dans les tours précédents. Les performances de notre algorithme ont été comparées à celles de l'algorithme de clustering le plus utilisé (LEACH), qui fixe à l'avance le nombre des cluster-chefs et qui ne garanti pas une couverture totale du réseau.

Les résultats obtenus montrent que l'algorithme de clustering proposé réduit, de manière significative, le taux de consommation (sur l'ensemble du réseau) d'énergie et garanti une meilleure couverture du réseau, comparativement à LEACH. Une direction de recherche intéressante consiste à appliquer l'algorithme de clustering basé sur les AGs dans les réseaux de capteurs sans fil avec le duty cycling.



# Conclusion et perspectives

## Conclusion

Le succès des réseaux de capteurs sans fil est fondé sur la simplicité des noeuds capteurs (la faible puissance de calcul, la petite batterie, l'antenne radio à portée limitée, etc). Cependant, ce point fort des RCSFs représente également des contraintes imposantes. En effet, la limitation des ressources dans les noeuds capteurs engendre plusieurs défis de conception, dont l'économie d'énergie est la plus importante. L'objectif de notre travail a été de proposer une nouvelle approche de clustering qui offre une consommation d'énergie acceptable, car les architectures plats des réseaux ont montré leur inefficacité en termes d'économie d'énergie. Dans cette thèse nous avons montré les avantages de l'approche du clustering, ainsi que son application afin de remédier aux limitations des noeuds capteurs.

Dans cette thèse, nous avons proposé une technique de clustering, qui permet d'économiser l'énergie, nommé GCDC, dans ce protocole nous avons utilisé les AGs comme un outil pour trouver le nombre optimal des cluster-chefs ainsi que leurs emplacements afin de couvrir le réseau entier et de minimiser la consommation d'énergie. Les résultats obtenus par les simulations montrent que l'algorithme proposé offre une meilleure connexion réseau et une consommation minimale d'énergie par rapport à LEACH (le protocole de clustering le plus populaire). une deuxième contribution dans cette thèse consiste à une analyse d'énergie dans les RCSFs, en prenant en considération le bon choix de la longueur du saut qui assure une meilleure consommation d'énergie, une condition facile à vérifier a été présentée, si une fois respectée assure une consommation minimal d'énergie, ainsi la prolongation de la durée de vie du réseau est garantie.

## perspectives

Les travaux menés dans cette thèse ont traité la problématique de la consommation d'énergie dans les RCSFs. Ceux-ci ouvrent de nouvelles perspectives de recherche :

- L'implémentation de notre protocole sur des capteurs réels (tels que : *Tmote Sky*, *MICA*, *Imote* ou *BTnode*), représente une perspectives qui permettra d'évaluer les performances de nos contributions dans le mode réel.
- Explorer des nouvelles pistes où en doit prendre en compte plusieurs paramètres, à savoir le temps de latence, que nous n'avons pas pris en considération dans notre travail.
- Au cours de cette thèse, nous avons considéré une seule classe des RCSFs : les réseaux mono-objectifs composés de noeuds capteurs permettant de collecter une seule mesure physique de l'environnement correspondante à une application particulière. La généralisation de l'approche pour les RCSFs multi-objectifs où plusieurs mesures physiques sont à collecter simultanément et qui correspondent à de multiples applications au sein d'un même réseau est prometteuse.
- L'impact de la compression et de l'agrégation sur l'efficacité énergétique dans les RCSFs multi-objectifs représente encore une piste de recherche à explorer.
- Envisager le cas d'un double clustering, dans un réseau de capteurs dense.
- Étudier l'influence de la position de la station de base sur le clustering.
- Examiner le cas de plusieurs stations de base.
- Examiner le cas d'une station de base mobile.
- Des améliorations concernant la prise en compte des différents aspects tels que la mobilité des noeuds capteurs, dans un RCSF mobile, sont aussi à envisager.
- Étudier la Qos dans les RCSFs.
- Étudier le cas de routage des flux de données importants (voix, video, etc)

dans les RCSFs multimedia est aussi un axe de recherche important reste à explorer.

- La sécurité des RCSFs est aussi un point important qui nécessite une étude minutieuse.



# Bibliographie

- [1] I.F. Akyildiz, Weilian Su, Y. Sankarasubramaniam, and E. Cayirci. A survey on sensor networks. *Communications Magazine, IEEE*, Aug 2002. (Cité en pages 7, 48, 49 et 94.)
- [2] Ian Akyildiz and Mehmet Can Vuran. *Wireless Sensor Networks*. John Wiley & Sons, Inc., New York, NY, USA, 2010. (Cité en pages 7, 17, 19, 20, 24, 27, 36, 38, 49, 92, 93 et 97.)
- [3] Boubiche Djallel Eddine. *Une approche Inter-Couches (cross-layer) pour la Sécurité dans les R.C.S.F.* PhD thesis, Université de Batna, 2013. (Cité en pages 7, 8, 14, 15, 16, 17, 19, 20 et 60.)
- [4] Imad Mahgoub Mohammed Ilyas. *Handbook of Sensor Networks :Compact Wireless and Wired Sensing Systems*. CRC Press LCC, 2005. (Cité en page 7.)
- [5] Jianbin Xue, Ting Zhang, Yan Yan, Wenhua Wang, and Songbai Li. Cooperation-based ant colony algorithm in wsn. *JNW*, 8(4) :939–946, 2013. (Cité en page 7.)
- [6] Uwe Herzog. (Cité en page 8.)
- [7] Gerard C.M. Meijer. *Smart Sensor Systems*. John Wiley and Sons, Ltd, Publication, 2008. (Cité en page 9.)
- [8] Ludovic Samper. *Modélisations et analyses de réseaux de capteurs*. PhD thesis, Institut National Polytechnique de Grenoble, 2008. (Cité en page 10.)
- [9] Ahmed CHEFI. *Conception d'un micro-capteur d'image CMOS à faible consommation d'énergie pour les réseaux de capteurs sans fil*. PhD thesis, Université de Grenoble, 2006. (Cité en pages 10, 12, 13, 14, 16, 48 et 49.)

- 
- [10] Andreas Willing Holger Karl. *Protocols and Architectures for Wireless Sensor Networks*. John Wiley and Sons, Ltd, Publication, 2005. (Cité en pages 12 et 13.)
- [11] Aurélien Buhrig. *Optimisation de la consommation des noeuds de réseaux de capteurs sans fil*. PhD thesis, Institut National Polytechnique de Grenoble, 2008. (Cité en pages 12, 13, 14, 15 et 16.)
- [12] Amine Abbas. *Les réseaux sensoriels sans fil : Etat de l'art et perspectives de recherche*. PhD thesis, équipe de recherche AND du LIFC, Belfort, laboratoire d'informatique de l'université de Franche-compté, 2005. (Cité en page 12.)
- [13] François Ingelrest. *Protocoles localisés de diffusion et économie d'énergie dans les réseaux ad hoc et de capteurs*. PhD thesis, Université des sciences et technologies de Lille, 2006. (Cité en page 12.)
- [14] Odey A. J. and Li D. Low power transceiver design parameters for wireless sensor networks. *Wireless Sensor Network*. (Cité en page 14.)
- [15] Clément Saad. *Quelques contributions dans les réseaux de capteurs sans fil*. PhD thesis, Université d'Avignon et des Pays de Vaucluse, 2008. (Cité en page 14.)
- [16] Rahim Kacimi. *Techniques de conservation d'énergie pour les réseaux de capteurs sans fil*. PhD thesis, Institut National Polytechnique de Toulouse, 2009. (Cité en page 15.)
- [17] Hela Boussetta. *Modélisation multi-physiques et simulations globales de systèmes autonomes sur puce*. PhD thesis, Université de Grenoble et Faculté des sciences de Bizerte, 2009. (Cité en page 16.)
- [18] Wei Ye, John Heidemann, and Deborah Estrin. An energy-efficient mac protocol for wireless sensor networks. In *INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies*. Pro-

- ceedings. IEEE*, volume 3, pages 1567–1576. IEEE, 2002. (Cité en pages 18, 51 et 64.)
- [19] Joseph Polastre, Jason Hill, and David Culler. Versatile low power media access for wireless sensor networks. In *Proceedings of the 2Nd International Conference on Embedded Networked Sensor Systems*, SenSys '04, pages 95–107, New York, NY, USA, 2004. ACM. (Cité en pages 18 et 64.)
- [20] Ozlem Durmaz Incel, Lodewijk van Hoesel, Pierre Jansen, and Paul Havinga. Mc-lmac : A multi-channel mac protocol for wireless sensor networks. *Ad Hoc Netw.*, 9(1) :73–94, January 2011. (Cité en page 18.)
- [21] Makhoul Abdallah. *Réseaux de capteurs : localisation, couverture et fusion de données*. PhD thesis, 2008. (Cité en page 21.)
- [22] Leila Imane NIAR. *Analyse Graphique pour la surveillance dans un réseau de capteurs sans fils (RCSF) Simulateur : OMNETpp*. PhD thesis, Université d'Oran, 2012. (Cité en page 21.)
- [23] Wassim MASRI. *Dérivation d'exigences de Qualité de Service dans les Réseaux de Capteurs Sans Fil basés sur TDMA*. PhD thesis, L'UNIVERSITÉ DE TOULOUSE, 2009. (Cité en pages 24, 25 et 52.)
- [24] Shuguang Cui, Andrea J. Goldsmith, and Ahmad Bahai. Energy-constrained modulation optimization. *IEEE Transactions on Wireless Communications*, 4(5) :2349–2360, 2005. (Cité en page 28.)
- [25] D Nadig, SS Iyengar, and DN Jayasimha. A new architecture for distributed sensor integration. In *Southeastcon'93, Proceedings., IEEE*, pages 8–p. IEEE, 1993. (Cité en page 28.)
- [26] Marco Zuniga and Bhaskar Krishnamachari. Analyzing the transitional region in low power wireless links. In *Sensor and Ad Hoc Communications and Networks, 2004. IEEE SECON 2004. 2004 First Annual IEEE Communications Society Conference on*, pages 517–526. IEEE, 2004. (Cité en page 28.)

- 
- [27] Jon Agre and Loren Clare. An integrated architecture for cooperative sensing networks. *Computer*, 33(5) :106–108, 2000. (Cit  en page 29.)
- [28] Elizabeth A Basha, Sai Ravela, and Daniela Rus. Model-based monitoring for early warning flood detection. In *Proceedings of the 6th ACM conference on Embedded network sensor systems*, pages 295–308. ACM, 2008. (Cit  en page 29.)
- [29] Manish Bhardwaj, Timothy Garnett, and Anantha P Chandrakasan. Upper bounds on the lifetime of sensor networks. In *Communications, 2001. ICC 2001. IEEE International Conference on*, volume 3, pages 785–790. IEEE, 2001. (Cit  en page 29.)
- [30] Philippe Bonnet, Johannes Gehrke, and Praveen Seshadri. Querying the physical world. *Personal Communications, IEEE*, 7(5) :10–15, 2000. (Cit  en page 29.)
- [31] Tian He, Sudha Krishnamurthy, Liqian Luo, Ting Yan, Lin Gu, Radu Stoleru, Gang Zhou, Qing Cao, Pascal Vicaire, John A Stankovic, et al. Vigilnet : An integrated sensor network system for energy-efficient surveillance. *ACM Transactions on Sensor Networks (TOSN)*, 2(1) :1–38, 2006. (Cit  en page 29.)
- [32] Tia Gao, Christopher Pesto, Leo Selavo, Yin Chen, JeongGil Ko, Jong Hyun Lim, Andreas Terzis, Andrew Watt, James Jeng, Bor-rong Chen, et al. Wireless medical sensor networks in emergency response : Implementation and pilot results. In *Technologies for Homeland Security, 2008 IEEE Conference on*, pages 187–192. IEEE, 2008. (Cit  en page 30.)
- [33] Joseph M Kahn, Randy H Katz, and Kristofer SJ Pister. Next century challenges : mobile networking for smart dust. In *Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking*, pages 271–278. ACM, 1999. (Cit  en pages 30 et 36.)

- 
- [34] David Malan, Thaddeus Fulford-Jones, Matt Welsh, and Steve Moulton. Coblue : An ad hoc sensor network infrastructure for emergency medical care. In *International workshop on wearable and implantable body sensor networks*, volume 5, 2004. (Cité en page 30.)
- [35] Miklós Maróti, Branislav Kusy, Gyula Simon, and Ákos Lédeczi. The flooding time synchronization protocol. In *Proceedings of the 2nd international conference on Embedded networked sensor systems*, pages 39–49. ACM, 2004. (Cité en page 30.)
- [36] Emile M Petriu, Nicolas D Georganas, Dorina C Petriu, Dimitrios Makrakis, and Voicu Z Groza. Sensor-based information appliances. *Instrumentation & Measurement Magazine, IEEE*, 3(4) :31–35, 2000. (Cité en page 31.)
- [37] Younghun Kim, Thomas Schmid, Zainul M Charbiwala, Jonathan Friedman, and Mani B Srivastava. Nawms : nonintrusive autonomous water monitoring system. In *Proceedings of the 6th ACM conference on Embedded network sensor systems*, pages 309–322. ACM, 2008. (Cité en page 31.)
- [38] Lakshman Krishnamurthy, Robert Adler, Phil Buonadonna, Jasmeet Chhabra, Mick Flanigan, Nandakishore Kushalnagar, Lama Nachman, and Mark Yarvis. Design and deployment of industrial sensor networks : experiences from a semiconductor plant and the north sea. In *Proceedings of the 3rd international conference on Embedded networked sensor systems*, pages 64–75. ACM, 2005. (Cité en page 32.)
- [39] Nissanka B Priyantha, Anit Chakraborty, and Hari Balakrishnan. The cricket location-support system. In *Proceedings of the 6th annual international conference on Mobile computing and networking*, pages 32–43. ACM, 2000. (Cité en page 33.)
- [40] Jan M Rabaey, M Josie Ammer, Julio L da Silva Jr, Danny Patel, and Shad Roundy. Picoradio supports ad hoc ultralow power wireless networking. *Com-*

- puter*, 33(7) :42–48, 2000. (Cité en pages 33 et 38.)
- [41] Brett Warneke, Matt Last, Brian Liebowitz, and Kristofer SJ Pister. Smart dust : Communicating with a cubic-millimeter computer. *Computer*, 34(1) :44–51, 2001. (Cité en page 33.)
- [42] Anthony Wood, Gilles Virone, Thao Doan, Quihua Cao, Leo Selavo, Yafeng Wu, L Fang, Zhimin He, Shan Lin, and Jack Stankovic. Alarm-net : Wireless sensor networks for assisted-living and residential monitoring. *University of Virginia Computer Science Department Technical Report*, 2, 2006. (Cité en page 34.)
- [43] S Roundy, PK Wright, and JM Rabaey. Energy scavenging for wireless sensor networks : with special focus on vibrations. *Norwell, Massachusetts : Kluwer Academic Publishers*. (Cité en page 36.)
- [44] Chalermek Intanagonwiwat, Ramesh Govindan, and Deborah Estrin. Directed diffusion : a scalable and robust communication paradigm for sensor networks. In *Proceedings of the 6th annual international conference on Mobile computing and networking*, pages 56–67. ACM, 2000. (Cité en pages 36 et 54.)
- [45] Gregory J Pottie and William J Kaiser. Wireless integrated network sensors. *Communications of the ACM*, 43(5) :51–58, 2000. (Cité en pages 36 et 43.)
- [46] G Hoblos, M Staroswiecki, and A Aitouche. Optimal design of fault tolerant sensor networks. In *Control Applications, 2000. Proceedings of the 2000 IEEE International Conference on*, pages 467–472. IEEE, 2000. (Cité en page 37.)
- [47] Chien-Chung Shen, Chavalit Srisathapornphat, and Chaiporn Jaikaeo. Sensor information networking architecture and applications. *Personal communications, IEEE*, 8(4) :52–59, 2001. (Cité en page 37.)
- [48] SeongHwan Cho and Anantha P Chandrakasan. Energy efficient protocols for low duty cycle wireless microsensor networks. In *Acoustics, Speech, and*

- Signal Processing, 2001. Proceedings.(ICASSP'01). 2001 IEEE International Conference on*, volume 4, pages 2041–2044. IEEE, 2001. (Cité en page 37.)
- [49] M Cardei and J Wu. Coverage problems in wireless ad hoc sensor networks, 2004. (Cité en page 39.)
- [50] Uroš M Pešović, Jože J Mohorko, Karl Benkič, and Žarko F Čučej. Single-hop vs. multi-hop–energy efficiency analysis in wireless sensor networks. In *18th Telecommunications Forum, TELFOR*, 2010. (Cité en pages 44, 93, 94, 95 et 97.)
- [51] M.E. Pardeep Kumar. *Medium Access Control Protocols for Energy and Delay Efficient Applications of Wireless Sensor Networks*. PhD thesis, Institute of Computer Science, Freie Universität Berlin, 2012. (Cité en pages 48, 50, 51 et 62.)
- [52] Azzedine Boukerche. *Algorithms and protocols for wireless sensor networks*, volume 62. John Wiley & Sons, 2008. (Cité en pages 49 et 97.)
- [53] Mekkaoui Kheireddine and Rahmoun Abdellatif. Short-hops vs. long-hops : Energy-efficiency analysis in wsn. *Network Security Technologies : Design and Applications : Design and Applications*, page 74, 2013. (Cité en pages 50, 94, 96, 114 et 115.)
- [54] Jennifer Yick, Biswanath Mukherjee, and Dipak Ghosal. Wireless sensor network survey. *Computer networks*, 52(12) :2292–2330, 2008. (Cité en pages 50 et 93.)
- [55] Muhammad Saleem, Gianni A Di Caro, and Muddassar Farooq. Swarm intelligence based routing protocol for wireless sensor networks : Survey and future directions. *Information Sciences*, 181(20) :4597–4624, 2011. (Cité en page 50.)
- [56] Joanna Kulik, Wendi Heinzelman, and Hari Balakrishnan. Negotiation-based protocols for disseminating information in wireless sensor networks. *Wireless networks*, 8(2/3) :169–185, 2002. (Cité en pages 51, 54 et 55.)

- 
- [57] Kemal Akkaya and Mohamed Younis. A survey on routing protocols for wireless sensor networks. *Ad hoc networks*, 3(3) :325–349, 2005. (Cité en pages 52, 53, 56, 57, 58 et 62.)
- [58] Bhaskar Krishnamachari, Deborah Estrin, and Stephen Wicker. Modelling data-centric routing in wireless sensor networks. In *IEEE infocom*, volume 2, pages 39–44, 2002. (Cité en page 53.)
- [59] YOUSEF Yaser. *Routage pour la Gestion de l’Energie dans les Réseaux de Capteurs Sans Fil*. PhD thesis, Université de HAUTE ALSACE, 2010. (Cité en pages 53, 56 et 57.)
- [60] Ahmed A Ahmed, Hongchi Shi, and Yi Shang. A survey on network protocols for wireless sensor networks. In *Information Technology : Research and Education, 2003. Proceedings. ITRE2003. International Conference on*, pages 301–305. IEEE, 2003. (Cité en page 54.)
- [61] Jamal N Al-Karaki and Ahmed E Kamal. Routing techniques in wireless sensor networks : a survey. *Wireless communications, IEEE*, 11(6) :6–28, 2004. (Cité en pages 57 et 58.)
- [62] Ya Xu, John Heidemann, and Deborah Estrin. Geography-informed energy conservation for ad hoc routing. In *Proceedings of the 7th annual international conference on Mobile computing and networking*, pages 70–84. ACM, 2001. (Cité en page 57.)
- [63] Volkan Rodoplu and Teresa H Meng. Minimum energy mobile wireless networks. *Selected Areas in Communications, IEEE Journal on*, 17(8) :1333–1344, 1999. (Cité en page 57.)
- [64] Seema Bandyopadhyay and Edward J Coyle. An energy efficient hierarchical clustering algorithm for wireless sensor networks. In *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications*.

- IEEE Societies*, volume 3, pages 1713–1723. IEEE, 2003. (Cité en pages 58 et 113.)
- [65] Jamal N Al-Karaki, Raza Ul-Mustafa, and Ahmed E Kamal. Data aggregation in wireless sensor networks-exact and approximate algorithms. In *High Performance Switching and Routing, 2004. HPSR. 2004 Workshop on*, pages 241–245. IEEE, 2004. (Cité en page 58.)
- [66] Giljae Lee, Junguk Kong, Minsun Lee, and Okhwan Byeon. A cluster-based energy aware routing protocol for sensor networks. In *Parallel and Distributed Computing and Systems Proceedings of the 17 th IASTED International Conference, 2005*. (Cité en page 59.)
- [67] Mohamed Younis, Moustafa Youssef, and Khaled Arisha. Energy-aware routing in cluster-based sensor networks. In *Modeling, Analysis and Simulation of Computer and Telecommunications Systems, 2002. MASCOTS 2002. Proceedings. 10th IEEE International Symposium on*, pages 129–136. IEEE, 2002. (Cité en page 59.)
- [68] Sajid Hussain and Abdul W Matin. Hierarchical cluster-based routing in wireless sensor networks. *Proceedings of IPSN (Poster)*, 2006. (Cité en pages 59 et 81.)
- [69] Chunyao FU, Zhifang JIANG, Wei WEI, and Ang WEI. An energy balanced algorithm of leach protocol in wsn. *International Journal of Computer Science*, 2013. (Cité en pages 59 et 81.)
- [70] Wendi B Heinzelman, Anantha P Chandrakasan, and Hari Balakrishnan. An application-specific protocol architecture for wireless microsensor networks. *Wireless Communications, IEEE Transactions on*, 1(4) :660–670, 2002. (Cité en pages 59, 74, 77 et 106.)
- [71] Jonas Neander, Ewa Hansen, Mikael Nolin, and Mats Bjorkman. Asymmetric multihop communication in large sensor networks. In *Wireless Pervasive*

- Computing, 2006 1st International Symposium on*, pages 7–pp. IEEE, 2006. (Cité en pages 59 et 81.)
- [72] PTV Bhuvanewari, V Vaidehi, and S Shanmugavel. Spear : sensor protocol for energy aware routing in wireless sensor network. In *Wireless Communication and Sensor Networks, 2007. WCSN'07. Third International Conference on*, pages 74–78. IEEE, 2007. (Cité en pages 59 et 81.)
- [73] Fan Xiangning and Song Yulin. Improvement on leach protocol of wireless sensor network. In *Sensor Technologies and Applications, 2007. SensorComm 2007. International Conference on*, pages 260–264. IEEE, 2007. (Cité en pages 59 et 81.)
- [74] Mohamed Lehsaini, Herve Guyennet, and Mohammed Feham. An efficient cluster-based self-organisation algorithm for wireless sensor networks. *International Journal of Sensor Networks*, 7(1) :85–94, 2010. (Cité en pages 59 et 81.)
- [75] Jia Xu, Ning Jin, Xizhong Lou, Ting Peng, Qian Zhou, and Yanmin Chen. Improvement of leach protocol for wsn. In *Fuzzy Systems and Knowledge Discovery (FSKD), 2012 9th International Conference on*, pages 2174–2177. IEEE, 2012. (Cité en pages 59 et 81.)
- [76] Nitin Mittal, Davinder Pal Singh, Amanjeet Panghal, and RS Chauhan. Improved leach communication protocol for wsn. In *National Conference on Computational Instrumentation*, pages 153–156, 2010. (Cité en pages 59 et 81.)
- [77] Fei Hui, Xiao Le Wang, and Xin Shi. A static-leach wsns for hazardous materials monitoring. In *Advanced Materials Research*, volume 463, pages 261–265. Trans Tech Publ, 2012. (Cité en pages 59 et 81.)
- [78] Basit Manzoor, Nadeem Javaid, O Rehman, M Akbar, Q Nadeem, Adeel Iqbal, and M Ishfaq. Q-leach : A new routing protocol for wsns. *Procedia Computer Science*, 19 :926–931, 2013. (Cité en pages 59 et 81.)

- [79] S. Lindsey and C.S. Raghavendra. Pegasus : Power-efficient gathering in sensor information systems. In *Aerospace Conference Proceedings, 2002. IEEE*, volume 3, pages 3–1125–3–1130 vol.3, 2002. (Cité en page 60.)
- [80] Kemei Du, Jie Wu, and Dan Zhou. Chain-based protocols for data broadcasting and gathering in the sensor networks. In *Parallel and Distributed Processing Symposium, 2003. Proceedings. International*, pages 8–pp. IEEE, 2003. (Cité en page 60.)
- [81] Stephanie Lindsey, Cauligi Raghavendra, and Krishna M. Sivalingam. Data gathering algorithms in sensor networks using energy metrics. *Parallel and Distributed Systems, IEEE Transactions on*, 13(9) :924–935, 2002. (Cité en page 60.)
- [82] Sung-Min Jung, Young-Ju Han, and Tai-Myoung Chung. The concentric clustering scheme for efficient energy consumption in the pegasus. In *Advanced Communication Technology, The 9th International Conference on*, volume 1, pages 260–265. IEEE, 2007. (Cité en page 62.)
- [83] Jung-Eun Lee and Keecheon Kim. Diamond-shaped routing method for reliable data transmission in wireless sensor networks. In *Parallel and Distributed Processing with Applications, 2008. ISPA'08. International Symposium on*, pages 799–801. IEEE, 2008. (Cité en page 62.)
- [84] Gang Lu, Bhaskar Krishnamachari, and Cauligi S Raghavendra. An adaptive energy-efficient and low-latency mac for data gathering in wireless sensor networks. In *Parallel and Distributed Processing Symposium, 2004. Proceedings. 18th International*, page 224. IEEE, 2004. (Cité en page 64.)
- [85] Michael Buettner, Gary V Yee, Eric Anderson, and Richard Han. X-mac : a short preamble mac protocol for duty-cycled wireless sensor networks. In *Proceedings of the 4th international conference on Embedded networked sensor systems*, pages 307–320. ACM, 2006. (Cité en page 64.)

- 
- [86] A. El-Hoiydi and J.-D. Decotignie. Wisemac : an ultra low power mac protocol for the downlink of infrastructure wireless sensor networks. In *Computers and Communications, 2004. Proceedings. ISCC 2004. Ninth International Symposium on*, volume 1, pages 244–251 Vol.1, June 2004. (Cité en page 64.)
- [87] Yanjun Sun, Omer Gurewitz, and David B. Johnson. Ri-mac : A receiver-initiated asynchronous duty cycle mac protocol for dynamic traffic loads in wireless sensor networks. In *Proceedings of the 6th ACM Conference on Embedded Network Sensor Systems*, SenSys '08, pages 1–14, New York, NY, USA, 2008. ACM. (Cité en page 64.)
- [88] Ameer Ahmed Abbasi and Mohamed Younis. A survey on clustering algorithms for wireless sensor networks. *Computer communications*, 30(14) :2826–2841, 2007. (Cité en pages 65, 68 et 112.)
- [89] Katayoun Sohrabi, Jay Gao, Vishal Ailawadhi, and Gregory J Pottie. Protocols for self-organization of a wireless sensor network. *IEEE personal communications*, 7(5) :16–27, 2000. (Cité en page 65.)
- [90] Rex Min, Manish Bhardwaj, Seong-Hwan Cho, Eugene Shih, Amit Sinha, Alice Wang, and Anantha Chandrakasan. Low-power wireless sensor networks. In *VLSI Design, 2001. Fourteenth International Conference on*, pages 205–210. IEEE, 2001. (Cité en page 65.)
- [91] Chunhung Richard Lin and Mario Gerla. Adaptive clustering for mobile wireless networks. *Selected Areas in Communications, IEEE Journal on*, 15(7) :1265–1275, 1997. (Cité en page 68.)
- [92] Suman Banerjee and Samir Khuller. A clustering scheme for hierarchical control in multi-hop wireless networks. In *INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, volume 2, pages 1028–1037. IEEE, 2001. (Cité en page 68.)

- [93] Deborah Estrin, Ramesh Govindan, John Heidemann, and Satish Kumar. Next century challenges : Scalable coordination in sensor networks. In *Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking*, pages 263–270. ACM, 1999. (Cité en page 68.)
- [94] Rolf Clauberg. Rfid and sensor networks. In *Proc. RFID Workshop, St. Gallen, Switzerland*, pages 1–6, 2004. (Cité en pages 71, 75, 76, 82 et 85.)
- [95] Liliana MC Arboleda and Nidal Nasser. Comparison of clustering algorithms and protocols for wireless sensor networks. In *Electrical and Computer Engineering, 2006. CCECE'06. Canadian Conference on*, pages 1787–1792. IEEE, 2006. (Cité en page 73.)
- [96] E.I. Oyman and C. Ersoy. Multiple sink network design problem in large scale wireless sensor networks. In *Communications, 2004 IEEE International Conference on*, volume 6, pages 3663–3667 Vol.6, June 2004. (Cité en page 74.)
- [97] Wei-Peng Chen, J.C. Hou, and Lui Sha. Dynamic clustering for acoustic target tracking in wireless sensor networks. *Mobile Computing, IEEE Transactions on*, 3(3) :258–271, July 2004. (Cité en pages 74 et 75.)
- [98] Noritaka Shigei, Hiromi Miyajima, Hiroki Morishita, and Michiharu Maeda. Centralized and distributed clustering methods for energy efficient wireless sensor networks. 2009. (Cité en page 74.)
- [99] Mekkaoui Kheireddine, Rahmoun Abdellatif, and Gianluigi Ferrari. Genetic centralized dynamic clustering in wireless sensor networks. In *Computer Science and Its Applications*, pages 503–511. Springer, 2015. (Cité en pages 74, 76, 87, 114 et 115.)
- [100] F. Bajaber and I. Awan. Dynamic/static clustering protocol for wireless sensor network. In *Computer Modeling and Simulation, 2008. EMS '08. Second UKSIM European Symposium on*, pages 524–529, Sept 2008. (Cité en page 75.)

- 
- [101] Prashant Krishana. A study on dynamic and static clustering based routing schemes for wireless sensor networks. *International Journal of Modern Engineering Research (IJMER)*, 3(2) :1100–1104, March-April 2013. (Cit  en page 75.)
- [102] Prashant Krishana. Comparison and performance analysis of dynamic and static clustering based routing scheme in wireless sensor network. *International Journal of Advanced Research in Computer and Communication Engineering*, 3(4) :2319–59404, April 2013. (Cit  en page 75.)
- [103] Wendi Rabiner Heinzelman, Anantha Chandrakasan, and Hari Balakrishnan. Energy-efficient communication protocol for wireless microsensor networks. In *System sciences, 2000. Proceedings of the 33rd annual Hawaii international conference on*, pages 10–pp. IEEE, 2000. (Cit  en pages 76, 77, 94, 113, 114 et 116.)
- [104] Guanfeng Li and Taieb Znati. ReCa : a ring-structured energy-efficient clustering architecture for robust communication in wireless sensor networks. *International Journal of Sensor Networks*, 2(1-2) :34–43, 2007. (Cit  en page 76.)
- [105] Ilker Demirkol, Cem Ersoy, and Fatih Alag z. Mac protocols for wireless sensor networks : a survey. *Communications Magazine, IEEE*, 44(4) :115–121, 2006. (Cit  en page 76.)
- [106] Pankaj K Agarwal and Cecilia Magdalena Procopiuc. Exact and approximation algorithms for clustering. *Algorithmica*, 33(2) :201–226, 2002. (Cit  en page 81.)
- [107] Hongwei Zhang and Anish Arora. Gs 3 : scalable self-configuration and self-healing in wireless sensor networks. *Computer Networks*, 43(4) :459–480, 2003. (Cit  en page 81.)
- [108] Sangho Yi, Junyoung Heo, Yookun Cho, and Jiman Hong. Peach : Power-efficient and adaptive clustering hierarchy protocol for wireless sensor net-

- works. *Computer communications*, 30(14) :2842–2852, 2007. (Cité en page 81.)
- [109] Stefano Basagni. Distributed clustering for ad hoc networks. In *Parallel Architectures, Algorithms, and Networks, 1999. (I-SPAN'99) Proceedings. Fourth International Symposium on*, pages 310–315. IEEE, 1999. (Cité en page 82.)
- [110] Reino Virrankoski and A Savvidees. Tasc : topology adaptive spatial clustering for sensor networks. In *Mobile Adhoc and Sensor Systems Conference, 2005. IEEE International Conference on*, pages 10–pp. IEEE, november 2005. (Cité en pages 82 et 83.)
- [111] Junpei Kamimura, Naoki Wakamiya, and Masayuki Murata. A distributed clustering method for energy-efficient data gathering in sensor networks. *International Journal of Wireless and Mobile Computing*, 1(2) :113–120, 2006. (Cité en pages 82 et 85.)
- [112] NM Latiff, Charalampos C Tsimenidis, and Bayan S Sharif. Energy-aware clustering for wireless sensor networks using particle swarm optimization. In *Personal, Indoor and Mobile Radio Communications, 2007. PIMRC 2007. IEEE 18th International Symposium on*, pages 1–5. IEEE, 2007. (Cité en pages 82 et 85.)
- [113] Mainak Chatterjee, Sajal K Das, and Damla Turgut. Wca : A weighted clustering algorithm for mobile ad hoc networks. *Cluster Computing*, 5(2) :193–204, 2002. (Cité en page 82.)
- [114] Ping Ding, JoAnne Holliday, and Aslihan Celik. Distributed energy-efficient hierarchical clustering for wireless sensor networks. In *Distributed computing in sensor systems*, pages 322–339. Springer, 2005. (Cité en page 83.)
- [115] Nour EL MAWASS. *Amélioration de l'efficacité des Réseaux de Capteurs Sans Fil en s'inspirant des Réseaux de Régulation Génétique*. PhD thesis, UNIVERSITE LIBANAISE, 2012. (Cité en page 84.)

- 
- [116] Falko Dressler and Ozgur B Akan. A survey on bio-inspired networking. *Computer Networks*, 54(6) :881–900, 2010. (Cité en pages 84 et 85.)
- [117] Saket Navlakha and Ziv Bar-Joseph. Algorithms in nature : the convergence of systems biology and computational thinking. *Molecular systems biology*, 7(1) :546, 2011. (Cité en page 84.)
- [118] Falko Dressler and Ozgur B Akan. Bio-inspired networking : from theory to practice. *Communications Magazine, IEEE*, 48(11) :176–183, 2010. (Cité en page 85.)
- [119] Selvadurai Selvakennedy, Sukunesan Sinnappan, and Yi Shang. A biologically-inspired clustering protocol for wireless sensor networks. *Computer Communications*, 30(14) :2786–2801, 2007. (Cité en page 85.)
- [120] Ali Norouzi, Faezeh Sadat Babamir, Abdul Halim Zaim, et al. A new clustering protocol for wireless sensor networks using genetic algorithm approach. *Wireless Sensor Network*, 3(11) :362, 2011. (Cité en pages 85 et 113.)
- [121] Martin Drozda, Sven Schaust, and Helena Szczerbicka. Is ais based misbehavior detection suitable for wireless sensor networks ? In *Wireless Communications and Networking Conference, 2007. WCNC 2007. IEEE*, pages 3128–3133. IEEE, 2007. (Cité en page 86.)
- [122] Mekkaoui Kheireddine and Rahmoun Abdellatif. Analysis of hops length in wireless sensor networks. *Wireless Sensor Network*, 2014, 2014. (Cité en pages 91, 94, 95, 96, 99, 114 et 115.)
- [123] M Castillo-Effer, Daniel H Quintela, W Moreno, R Jordan, and W Westhoff. Wireless sensor networks for flash-flood alerting. In *Devices, Circuits and Systems, 2004. Proceedings of the Fifth IEEE International Caracas Conference on*, volume 1, pages 142–146. IEEE, 2004. (Cité en page 92.)
- [124] Geoffrey Werner-Allen, Konrad Lorincz, Mario Ruiz, Omar Marcillo, Jeff Johnson, Jonathan Lees, and Matt Welsh. Deploying a wireless sensor network

- on an active volcano. *Internet Computing, IEEE*, 10(2) :18–25, 2006. (Cité en page 92.)
- [125] Konrad Lorincz, David J Malan, Thaddeus RF Fulford-Jones, Alan Nawoj, Antony Clavel, Victor Shnayder, Geoffrey Mainland, Matt Welsh, and Steve Moulton. Sensor networks for emergency response : challenges and opportunities. *Pervasive Computing, IEEE*, 3(4) :16–23, 2004. (Cité en page 92.)
- [126] Mohammad Ilyas and Imad Mahgoub. *Handbook of sensor networks : compact wireless and wired sensing systems*. CRC press, 2004. (Cité en page 92.)
- [127] Szymon Fedor and Martin Collier. On the problem of energy efficiency of multi-hop vs one-hop routing in wireless sensor networks. In *Advanced Information Networking and Applications Workshops, 2007, AINAW'07. 21st International Conference on*, volume 2, pages 380–385. IEEE, 2007. (Cité en pages 94, 95 et 96.)
- [128] Mario Neugebauer, Joern Ploennigs, and Klaus Kabitzsch. Evaluation of energy costs for single hop vs. multi hop with respect to topology parameters. In *IEEE International Workshop on Factory Communication Systems*, 2006. (Cité en pages 94 et 97.)
- [129] Martin Haenggi and Daniele Puccinelli. Routing in ad hoc networks : a case for long hops. *Communications Magazine, IEEE*, 43(10) :93–101, 2005. (Cité en page 94.)
- [130] Martin Haenggi. Twelve reasons not to route over many short hops. In *Vehicle Technology Conference, 2004. VTC2004-Fall. 2004 IEEE 60th*, volume 5, pages 3130–3134. IEEE, 2004. (Cité en pages 94 et 95.)
- [131] Bolian Yin, Hongchi Shi, and Yi Shang. A two-level strategy for topology control in wireless sensor networks. In *Parallel and Distributed Systems, 2005. Proceedings. 11th International Conference on*, volume 2, pages 358–362. IEEE, 2005. (Cité en page 95.)

- 
- [132] Theodore S Rappaport et al. *Wireless communications : principles and practice*, volume 2. prentice hall PTR New Jersey, 1996. (Cité en page 95.)
- [133] Giuseppe Anastasi, Alessio Falchi, Andrea Passarella, Marco Conti, and Enrico Gregori. Performance measurements of motes sensor networks. In *Proceedings of the 7th ACM international symposium on Modeling, analysis and simulation of wireless and mobile systems*, pages 174–181. ACM, 2004. (Cité en page 100.)
- [134] Ayon Chakraborty, Swarup Kumar Mitra, and Mrinal Kanti Naskar. A genetic algorithm inspired routing protocol for wireless sensor networks. *International Journal of Computational Intelligence Theory and Practice*, 6(1) :1–8, 2011. (Cité en pages 106, 113 et 114.)
- [135] Patrick Siarry Collette Yan. *Optimisation multiobjectif*, volume 12. Editions Eyrolles, 2002. (Cité en pages 107, 108, 109, 110 et 112.)
- [136] Ouiza ZERDANI BOUARAB. *L’Optimisation non Linéaire Multiobjectif*. PhD thesis, Université de Tizi-Ouzou, 2013. (Cité en pages 108 et 109.)
- [137] Inès Alaya. *Optimisation multi-objectif par colonies de fourmis Cas des problèmes de sac à dos*. PhD thesis, Université de CLAUDE BERNARD LYON1, 2009. (Cité en page 109.)
- [138] Shiyuan Jin, Ming Zhou, and Annie S Wu. Sensor network optimization using a genetic algorithm. In *Proceedings of the 7th World Multiconference on Systemics, Cybernetics and Informatics*, pages 109–116, 2003. (Cité en page 112.)
- [139] Olutayo Boyinbode, Hanh Le, and Makoto Takizawa. A survey on clustering algorithms for wireless sensor networks. *International Journal of Space-Based and Situated Computing*, 1(2-3) :130–136, 2011. (Cité en page 115.)
- [140] John H Holland. Genetic algorithms. *Scientific american*, 267(1) :66–72, 1992. (Cité en page 116.)

## **ملخص:**

من أجل تقليل استهلاك الطاقة الناتج عن الاتصالات في شبكات الاستشعار اللاسلكية، تم عرض مساهمتين في هذه الأطروحة. المساهمة الأولى تركز على المقارنة بين طول مسافة المحادثة، في الواقع عمر شبكة يعتمد بشدة على كيفية إبلاغ البيانات في الشبكة باستخدام إما القفزات القصيرة أو القفزات الطويلة. المحاكاة المقدمة تبين أن استخدام الطول المثالي للمحادثة يقلل من استهلاك الطاقة. في المساهمة الثانية، أين يتم استعمال المجموعات. من أجل الحد من استهلاك الطاقة، حيث أن البحوث الحديثة قد أثبتت أن استخدام المجموعات وسيلة ساحقة للحد من تضييع الطاقة. المشكلة تكمن في تحديد عدد رؤساء المجموعات وتوزيعهم في الشبكة لضمان الحد الأدنى من استهلاك الطاقة وتحسين التغطية الشبكات. في هذه الأطروحة، نقتراح خوارزمية ديناميكية مركزية على أساس الخوارزمية الجينية معايير المجموعات و ذلك للحد من استهلاك الطاقة و ضمان أفضل تغطية. نتائج المحاكاة التي تم الحصول عليها تؤكد أن التقنية المقترحة تتفوق على خوارزمية LEACH.

**الكلمات الدالة:** شبكات الاستشعار اللاسلكية، المجموعات، كفاءة استخدام الطاقة، عمر شبكة، الخوارزميات الجينية.

## **Abstract:**

In order to minimize the energy consumption involved by communications in wireless sensor networks, two main contributions are presented in this thesis. The first one, focuses on the analysis of hop lengths, indeed the network lifetime depends strongly on how the data are communicated in the network by using either short-hops or long-hops. The simulations provided show that using an ideal length hops reduces the energy consumption. In the second contribution, the clustering is investigated. The use of clustering has proven to be overwhelming. The problem consists of determining the number of cluster-heads and their distribution in the network to ensure minimal energy consumption and better networks coverage. The Low-Energy Adaptive Clustering Hierarchy algorithm fixes in advance the number of cluster-heads, thus this does not guarantee the coverage of the entire network. In this thesis, we propose a genetic centralized dynamic algorithm (GA)-based clustering approach to optimize the clustering configuration (the number of cluster-heads, their distribution and the cluster-members) to limit node energy consumption and the best coverage. Experiments have been conducted on well-known WSN data benchmark to test the performance of the proposed methodology. The obtained simulation results show that the proposed technique overcomes the Low-Energy Adaptive Clustering Hierarchy algorithm.

**Keywords:** Wireless sensor networks, Clustering, energy efficiency, network lifetime, genetic algorithms.

## **Résumé:**

Dans le cadre de la minimisation de la consommation d'énergie produite par les communications au sein d'un réseau de capteurs sans fil, deux contributions sont présentés dans cette thèse. La première porte sur une analyse des longueurs des sauts, en effet la durée de vie du réseau dépend fortement de la manière de communication de données soit par des courts sauts ou des longs sauts. Les simulations présentés à la fin de cette thèse montrent l'efficacité de notre condition présentée, si une fois respectée elle réduit de manière significative la consommation d'énergie. Dans la deuxième contribution, la technique de clustering est utilisée pour réduire la consommation d'énergie, le problème reste à déterminer le nombre des cluster-chefs ainsi que leurs positions dans le réseau pour assurer une consommation minimale d'énergie et une meilleure couverture réseau. Pour surmonter ce problème, les algorithmes génétiques sont utilisés comme outils pour déterminer les paramètres du clustering. Les résultats de simulation, présentés dans cette thèse, montrent que notre approche présentée surmonte les résultats de l'algorithme LEACH.

**Mots clés:** Réseau de capteurs sans fil, Clustering, l'efficacité énergétique, La durée de vie du réseau, les algorithmes génétiques.