République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieure et de la Recherche Scientifique

Université Ahmed Draia - Adrar

Faculté des Sciences et de la Technologie

Département des Mathématiques et Informatique

Mémoire de fin d'étude, en vue de l'obtention du diplôme de Master en informatique

Option : Systèmes Intelligents

Thème

Une technique d'optimisation de la consommation d'énergie dans les réseaux de capteurs sans fil

Préparé par

KHALILI Zeyneb et BOUCHRA Meryem

Encadré par

Mr. KADDI Mohammed

Année Universitaire 2018/2019

Résumé

Les réseaux de capteurs sans fil représentent une révolution technologique des instruments de mesures, issue de la convergence des systèmes électroniques miniaturisés et des systèmes de communication sans fil. Il s'agit d'ensembles d'unités électroniques miniaturisées capables de mesurer certains phénomènes physiques dans l'environnement où ils sont déployés. En raison des contraintes de miniaturisation, et aussi de coût de fabrication, les nœuds de capteurs sont généralement dotés de ressources très limitées en termes d'énergie, de capacité du calcul, d'espace du stockage de données et de débit de transmission. Ces limitations motivent une grande partie des problématiques de recherche dans le domaine des réseaux de capteurs sans fil, en particulier, la contrainte de l'énergie qui est un problème fondamental. L'objectif de ce travail est de proposer un protocole de routage basé sur l'algorithme à évolution différentielle (AED) dans les réseaux de capteurs sans fils (RCSFs) pour minimiser la consommation d'énergie et prolonger la durée de vie du RCSF, appelé DEBCR (**D**ifferential **E**volution **B**ased **C**lustering and **R**outing protocol). La performance du protocole proposé est testée à l'aide du langage de programmation MATLAB.

Mots clés: Réseau de capteurs sans fils, consommation d'énergie, durée de vie, l'algorithme à évolution différentielle, DEBCR.

Abstract

Wireless sensor networks represent a technological revolution in measuring instruments, resulting from the convergence of miniaturized electronic systems and wireless communication systems. These are sets of miniaturized electronic units that can measure certain physical phenomena in the environment in which they are deployed. Because of the constraints of miniaturization, and also of manufacturing cost, the sensor nodes are generally endowed with very limited resources in terms of energy, computing capacity, data storage space and transmission rate. These limitations motivate a large part of the research problems in the field of wireless sensor networks, in particular, energy constraint which is a fundamental problem. The objective of this work is to propose a routing protocol based on the differential evolution algorithm (DE) in wireless sensor networks (WSN) to minimize energy consumption and extend the lifetime of the WSN, called DEBCR (**D**ifferential **E**volution **B**ased **C**lustering and **R**outing protocol). The performance of the proposed protocol is tested using the MATLAB programming language.

Key words: Wireless sensor network, energy consumption, lifetime, differential evolution algorithm, DEBCR.

Remerciements

En tout premier lieu, nous remercions le bon Dieu, tout puissant, de nous avoir donné la force et la patience pour accomplir ce travail.

Nous tenons de remercier grandement notre encadreur Mr. KADDI Mohamed pour son encadrement, sa disponibilité, son encouragement et ses précieux conseils.

Nous remercions également les membres du jury d'avoir voulu examiner et évaluer ce travail.

Enfin, nous adressons nos remerciements à toutes les personnes qui ont participé de près ou de loin à la réalisation de ce travail.

KHALILI Zeyneb et BOUCHRA Meryem

Dédicaces

Je dédie ce modeste travail

A mes chers parents

A toute ma famille.

Zeyneb

Dédicaces

Je dédie ce travail à tous ceux qui nous ont aidé à le réaliser.

Le plus grand merci A :

Source de vie…Ma mère

Soutien de vie…mon défunt père

Et tous les membres de ma famille.

Meryem

Table des matières

Résumé I

Remerciements III

Dédicaces IV

Table des matières VI

Liste des figures XIII

Liste des tableaux XV

Abréviations XVI

Introduction générale 1

Chapitre 1 : Réseaux de capteurs sans fil: Etat de l'art

1.1. Introduction 4

1.2. Capteur 4

1.2.1. Définitions 4

1.2.2. Architecture d'un capteur 5

1.2.2.1. Architecture matérielle 5

1.2.2.2. Architecture logicielle (Systèmes d'exploitation) 7

1.2.3. Classification des capteurs 10

1.2.3.1. Les capteurs Actifs 10

1.2.3.2. Les capteurs passive 11

1.2.4 Types des capteurs 13

1.2.4.1. Mica2Dot 15

1.2.4.2. MicaZ 15

1.2.4.3. Tmote Sky 16

1.3. Réseaux de capteurs sans fil 18

1.3.1. Définition 18

1.3.2. Architecture d'un RCSF 18

1.3.3. Caractéristiques d'un RCSF 19

1.3.4. Contraintes de conception des RCSF 20

1.3.4.1. La tolérance aux fautes 20

1.3.4.2. Le facteur d'échelle 21

1.3.4.3. Les coûts de production 21

1.3.4.4. L'environnement 21

1.3.4.5. La topologie de réseau 21

1.3.4.6. Les contraintes matérielles 21

1.3.4.7. Les médias de transmission 21

1.3.4.8. La consommation d'énergie 21

1.4. Pile protocolaire 21

1.4.1. Rôle des couches 23

1.4.2. Plans de gestion 24

1.5. Applications des RCSF 25

1.5.1. Découverte de catastrophes naturelles 25

1.5.2. Détection d'intrusions 25

1.5.3. Surveillance de l'environnement 26

1.5.4. Surveillance de la santé 26

1.5.5. Agriculture 27

1.5.6. Contrôle du trafic routier 28

1.6. Conclusion 29

Chapitre 2 : Techniques de conservation d'énergie dans les RCSFs

2.1. Introduction 31

2.2. Définition de durée de vie de réseaux 31

2.2.1. Durée de vie basée sur le nombre de nœuds vivants 31

2.2.2. Durée de vie basée sur la connectivité 32

2.2.3. Durée de vie basée sur la couverture 33

2.2.4. Durée de vie basée sur des les exigences de QoS 33

2.3. Consommation d'énergie dans les RCSF 33

2.3.1. Energie de capture 34

2.3.2. Energie de traitement 34

2.3.3. Energie de communication 34

2.4. Facteurs intervenants dans la consommation d'énergie 34

2.4.1. Etat du module radio 34

2.4.2. Accès au medium de transmission(MAC) 35

2.4.2.1. Retransmission 35

2.4.2.2. Ecoute active 35

2.4.2.3. Surécoute 35

2.4.2.4. Surcharge 36

2.4.4.5. Surémission 36

2.4.4.6. Taille des paquets 36

2.4.4.7. Modèle de propagation radio 37

2.5. Méthodes de conservation d'énergie dans les RCSF 37

2.5.1. Techniques du Duty-cycling 37

2.5.1.1. Protocole de contrôle de topologie 38

2.5.1.2. Protocoles de gestion d'énergie 38

a. Protocoles sleep/wakeup 38

b. Protocoles du niveau MAC 39

2.5.2. Approches orientées donnée 42

2.5.2.1. Technique de Réduire les données 42

2.5.2.2. Acquisition de données efficace en énergie 43

2.5.3. Mobilité 44

2.6. Comparaison des techniques de conservation énergétique 45

2.7. Conclusion 47

Chapitre 3 : Le routage dans les RCSFs

3.1. Introduction 49

3.2. Définition de routage 49

3.3. Classification des protocoles de routage 49

3.3.1. Classification selon la topologie du réseau 50

3.3.1.1. Les protocoles de routage à plat 50

3.3.1.2. Les protocoles hiérarchiques 51

3.3.2. Classification selon le fonctionnement du protocole 51

3.3.2.1. Routage basé sur la QoS 51

3.3.2.2. Routage basé sur la négociation 52

3.3.2.3. Routage basé sur les requêtes 52

3.3.2.4. Routage basé sur les multi-chemins 52

3.3.3. Classification selon le paradigme de communication 52

3.3.3.1. Centré-nœuds (Node-centric) 53

3.3.3.2. Centré-données (Data-centric) 53

3.3.3.3. Basé-localisation (location-based) 53

3.3.4. Classification selon l'établissement de la route 53

3.3.4.1. Protocoles proactifs 53

3.3.4.2. Protocoles réactifs 54

3.3.4.3. Protocoles hybrides 54

3.3.5. Classification selon l'initiateur de communication 54

3.3.5.1. Communication lancée par la source 55

3.3.5.2. Communication lancée par la destination..... 55

3.4. Exemples des protocoles de routage 55

3.4.1. Low Energy Adaptive Clustering Hierarchy (LEACH) 55

3.4.2. Power-Efficient GAthering in Sensor Information Systems (PEGASIS) 56

3.4.3. Threshold sensitive Energy Efficient sensor Network protocol (TEEN et APTEEN)..... 56

3.4.4. Le protocole de routage (MECN) 57

3.4.5. Sensor Protocol for information via negociation (SPIN) 57

3.4.6. Le protocole de routage (GAF) 58

3.4.7. La diffusion dirigée 58

3.5. Les protocoles hiérarchiques 59

3.5.1. L'approche basée sur le cluster 59

3.5.1.1. LEACH..... 61

3.5.1.2. Low Energy Adaptive Clustering Hierarchy Centralized (LEACH-C) ... 63

3.5.1.3. Fixed number of LEACH (LEACH-F) 64

3.5.1.4. Two level Low Energy Adaptive Clustering Hierarchy (TL-LEACH).... 64

3.5.1.5. Improved Low Energy Adaptive Clustering Hierarchy (I-LEACH)..... 65

3.5.1.6. Cell Low Energy Adaptive Clustering Hierarchy (Cell-LEACH) 65

3.5.1.7. Multi-hop Low Energy Adaptive Clustering Hierarchy (MH-LEACH).. 65

3.5.1.8. Comparaison entre LEACH et ses variantes 66

3.5.2. L'approche basée sur la chaine 67

3.5.2.1. PEGASIS 67

3.5.2.2. H-PEGASIS (Hierarchical PEGASIS) 69

3.2.2.3.Energy Balancing PEGASIS (EB-PEGASIS) 70

3.6. Conclusion 70

Chapitre 4: AED et travaux liés

4.1. Introduction 72

4.2. Métaheuristiques..... 72

4.3. L'algorithme à évolution différentielle 72

4.3.1. Définition 72

4.3.2. Principe 73

4.3.2.1. Mutation différentielle..... 74

4.3.2.2. Croisement différentiel..... 75

4.3.2.3. Sélection 75

4.3.2.4. Critère d'arrêt 76

4.3.3. Exemple 77

4.3.4. Variantes pour l'AED 77

4.4. Travaux liés 78

4.4.1. LEACH Protocol Improvement Based on the Differential Evolution Algorithm (LEACH_DE) 78

4.4.2. Differential Evolution based Clustering Algorithm (DECA) 81

4.4.3. Hybrid differential evolution and simulated anneling (DESA) 85

4.4.4. Switching-Differential Evolution(S-DE) for Cluster Head Election in Wireless Sensor Network 86

4.4.5. Comparaison entre les travaux 87
4.4.5.1. Comparaison suivant les paramètres de l'AED 88
4.4.5.2. Comparaison suivant autres paramètres 89
4.5. Conclusion 89

Chapitre 5 : Contribution

5.1. Introduction 91
5.2. Modèle du système 91
5.2.1. Modèle d'énergie 91
5.2.1. Modèle du réseau 92
5.3. Description du protocole proposé DEBRC 92
5.3.1. Phase de clustering 92
5.3.2. Phase de transmission 96
5.4. Implémentation du protocole DEBRC 97
5.4.1. Choix du langage de programmation 98
5.4.2. Description de l'interface de simulation 98
5.4.3. Résultats et discussion 101
5.4.3.1. Les paramètres de simulation 102
5.4.3.2. Métriques d'évaluation 102
5.4.3.3. Résultats 103
5.4.3.4. Analyse des résultats 103
5.5. Conclusion 106
Conclusion générale et perspectives 107
Bibliographique 108

Liste des figures

Figure 1.1 : Schéma interne d'un Capteur sans fil. 4

Figure 1.2 : Architecture d'un nœud de capteur. 5

Figure1.3 : Principe de fonctionnement d'un dispositif RFID 12

Figure 1.4: Principe de fonctionnement d'un capteur SAW 13

Figure 1.5 : Quelques exemples de capteurs. 14

Figure 1.6 : Capteur Mica2Dot 15

Figure 1.7 : Capteur MicaZ 15

Figure 1.8 : Capteur Tmote Sky. 16

Figure 1.9 : Réseau de capteurs sans fil 18

Figure 1.10: Architecture générale d'un réseau de capteurs sans fil..... 19

Figure 1.11 : La pile protocolaire des RCSF 23

Figure 1 .12 : Découverte de catastrophes naturelles par des RCSFs 25

Figure 1.13 : Application des capteurs dans la détection d'intrusions 26

Figure 1.14 : Application des RCSFs dans la surveillance de la santé. 27

Figure 1.15 : Application des RCSFs dans l'Agriculture 28

Figure 1.16 : Application des RCSFs pour la Contrôle du trafic routier 28

Figure 2.1: La surécoute dans une transmission 36

Figure 2.2 : Les méthodes de conservation d'énergie 37

Figure 2.3 : les techniques utilisées pour minimiser la consommation d'énergie 46

Figure 3.1: Classification des protocoles de routage 50

Figure 3.2 : Routage plat 51

Figure 3.3 : Le routage hiérarchique 59

Figure 3.4 : Organigramme de l'algorithme de formation de clusters distribué pour LEACH. 61

Figure 3.5 : Chronologie montrant le fonctionnement de LEACH. 61

Figure 3.6 : Algorithme de routage LEACH 63

Figure 3.7 : Protocole PEGASIS : La construction de la chaîne 68

Figure 4.1 : Mutation différentielle 74

Figure 4.2: Croisement différentiel : binomial à gauche, exponentiel à droite 75

Figure 4.3 : Un réseau de capteur sans fil avec des passerelles 83

Figure 4.4 : Affectation finale 87

Figure 5.1 : Le modèle de dissipation de l'énergie 92

Figure 5.2 : Organigramme du protocole DEBCR 95

Figure 5.3 : Organigramme de la phase de transmission des données 97

Figure 5.4 : L'interface « Simulateur DEBCR » 98

Figure 5.5 : Paramètres du réseau 99

Figure 5.6 : Paramètres de l'AED 99

Figure 5.7 : Zone de simulation 100

Figure 5.8 : Légende 100

Figure 5.9 : Boutons d'exécution. 100

Figure 5.10 : Zone des résultats 101

Figure 5.11: Comparaison en termes de la durée de vie 104

Figure 5.12: Comparaison en termes de nombre des noeuds morts par tour 105

Figure 5.13: Comparaison en termes de nombre des noeuds vivants pour chaque tour 105

Figure 5.14 : Comparaison en termes de l'énergie résiduel 105

Liste des tableaux

Tableau 1.1 : Comparaison entre les caractéristiques de quelques systèmes d'exploitation ... 7

Tableau 1.2. : Comparaison entre les caractéristiques de quelques capteurs 17

Tableaux 2.1: Comparaison entre les techniques d'accès au médium dans les RCSFs 41

Tableau 2.2 *:* Comparaison entre quelque approches de conservation d'énergie dans les RCSFs. 41

Tableau 3.1 : Comparaison entre LEACH et ses variantes. 66

Tableau 4.1: Affectation des nœuds de capteur à partir du vecteur de codage 83

Tableau 4.2 : Comparaison suivant les paramètres de l'AED 88

Tableau 4.3 : Comparaison suivant autres paramètres de l'AED 89

Tableau 5.1 : caractéristiques matérielles 98

Tableau 5.2 : Méthodes utilisées de l 'AED 101

Tableau 5.3: Paramètres de simulation 102

Tableau 5.4: Résultats des simulations 103

Liste des abréviations

ADC :	Analog Digital Converter
AE :	Algorithmes Évolutionnaires
AED :	Algorithme à Évolution Différentielle
APTEEN :	Adaptive Threshold sensitive Energy Efficient sensor Network protocol
CDMA :	Code Division Multiple Access
CH :	Cluster-Head
CSIP :	Collaborative Signal and Information Processing
CSMA :	Carrier Sense Multiple Access
Cell-LEACH :	Cell Low Energy Adaptive Clustering Hierarchy
DEBCR	Differential Evolution Based Clustering and Routing protocol
DVS :	Dynamic Voltage Scaling
EB-PEGASIS:	Energy Balancing PEGASIS
FEM :	Finite Element Method
GAF :	Geographic Adaptive Fidelity
GPS :	Global Positioning System
IDT :	InterDigital Transducer
ID :	Identifier
I-LEACH :	Improved Low Energy Adaptive Clustering Hierarchy
LEACH:	Low Energy Adaptive Clustering Hierarchy
LEACH-C :	Low Energy Adaptive Clustering Hierarchy Centralized
LEACH-F:	Fixed number of cluster Low Energy Adaptive Clustering Hierarchy
MANTIS OS :	MultimodAl NeTworks of In-situ micro Sensor Operating System
MAC :	Medium Access Control
MECN :	Minimum Energy Communication Network .
MH-LEACH :	Multi-Hop Low Energy Adaptive Clustering Hierarchy
NL :	Nœud Leader
PEGASIS :	Power-Efficient GAthering in Sensor Information Systems
QoS :	Quality of Service
RAM :	Random Access Memory
RCSF :	Réseau de Capteurs Sans Fil
SA :	Simulated Annealing

SB :	Station de Base
SPIN:	Sensor Protocol For Information via Negociation
S_MAC :	Sensor MAC
STEM :	Sparse Topology and Energy Management
TDMA :	Time Division Multiplexing Access.
TEEN :	Threshold sensitive Energy Efficient sensor Network protocol
TL-LEACH :	Two level Low Energy Adaptive Clustering Hierarchy
TinyOS :	Tiny Operating System
WSN	Wireless Sensor Network
Z_MAC :	Zebra MAC

Introduction générale

Les progrès réalisés dans les systèmes micro-électromécanique MEMS (Micro-electro-mechanical systems) d'une part et dans les technologies de communication sans fils d'autre part ont donné naissance à des nouveaux appareils exceptionnels appelés capteurs. Ces petits dispositifs servent à récolter et transmettre des données environnementales (luminosité, mouvement, température, pression barométrique, etc.) d'une manière autonome.

Un réseau de capteurs sans fil (RCSF) est un réseau ad hoc d'un grand nombre de nœuds de capteurs aléatoirement dispersés dans une zone géographique. Les RCSFs peuvent être utilisés dans divers domaines : militaire, médical, environnemental…etc.

Plusieurs contraintes sont posées dans le domaine des RCSFs (localisation, routage, alimentation,…etc), mais la principale contrainte est la source d'alimentation limitée et irremplaçable des nœuds de capteur.

Un grand nombre de méthodes ont été proposés et discutées dans le but de minimiser la consommation d'énergie dans un réseau de capteurs, cependant, le clustering est la technique la plus efficace pour la conservation d'énergie.

Dans une architecture basé-cluster, le réseau est partitionné en groupes dont un seul nœud appelé cluster-head (CH) est choisi comme nœud maitre, ce nœud reçoit les données des nœuds de son groupe, les agrège et puis les transmet à la station de base (SB).

Dans ce travail, nous implémentons un protocole de clustering basé sur l'algorithme à évolution différentielle (AED), appelé (DEBCR **D**ifferential **E**volution **B**ased **C**lustering and **R**outing protocol). Le protocole proposé vise à réaliser un choix raisonnable du CH approprié pour tous les nœuds des capteurs. Les résultats de l'implémentation sont discutés dans le chapitre 5.

Le travail est organisé comme suit :

Le premier chapitre introduit un état de l'art sur les RCSFs.

Le deuxième chapitre présente les techniques de conservation d'énergie dans les RCSFs.

Le troisième chapitre détaille une classification des protocoles de routage pour les RCSFs.

Le quatrième chapitre est consacré pour la description de l'AED et aux travaux liées utilisant cet algorithme dans le routage pour les RCSFs.

Dans le cinquième chapitre, l'implémentation du protocole DEBCR et les résultats de simulation sont détaillés.

Chapitre 1

Réseaux de capteurs sans fil: Etat de l'art

Sommaire

1.1. Introduction

1.2. Capteur

1.3. Réseaux de capteurs sans fil

1.4. Pile protocolaire

1.5. Applications des RCSF

1.6. Conclusion

1.1. Introduction

Les avancées technologiques récentes confortent la présence de l'informatique et de l'électronique au cœur du monde réel.

De plus en plus d'objets se voient ainsi équipés de processeurs et de moyens de communication sans fil et mobiles, leur permettant de traiter des informations mais également de les transmettre Les réseaux de capteurs sans fil (RCSFs) entrent dans ce cadre. En effet, ceux-ci sont constitués d'un ensemble de petits appareils, ou capteurs, possédant des ressources particulièrement limitées, mais qui leur permettent néanmoins d'acquérir des données sur leur environnement immédiat, de les traiter et de les communiquer à une station de base(SB) ou point de collecte.

Dans ce chapitre nous présentons une généralité sur les RCSFs . Nous commençons par une définition d'un capteur, son architecture, ses types etc. Ensuite nous présentons les RCSFs, leur architecture ,la pile protocolaire ,et enfin leurs domaine d'application .

1.2. Capteur

1.2.1. Définition

Un capteur sans fil est un petit dispositif électronique capable de mesurer une valeur physique environnementale (température, lumière, pression, etc.) et de la communiquer à un centre de contrôle via une SB. [1]

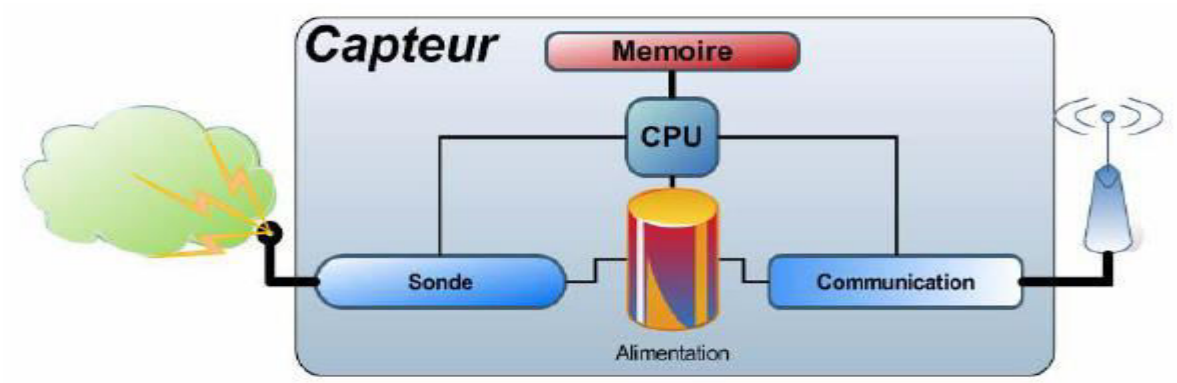

Figure 1.1 : Schéma interne d'un Capteur sans fil**.** [1]

1.2.2. Architecture d'un capteur

Dans cette section, nous distinguerons l'architecture matérielle ainsi que les systèmes embarqués dans un capteur.

1.2.2.1. Architecture matérielle

Un « *nœud capteur* » contient quatre unités de base : l'unité de captage, l'unité de traitement, l'unité de transmission, et l'unité de contrôle d'énergie. Selon le domaine d'application, il peut aussi contenir des modules supplémentaires tels qu'un système de localisation le **G**lobal **P**ositioning **S**ystem (GPS), ou bien un système générateur d'énergie (cellule solaire). Quelques micro-capteurs, plus volumineux, sont dotés d'un système mobilisateur chargé de les déplacer en cas de nécessité.

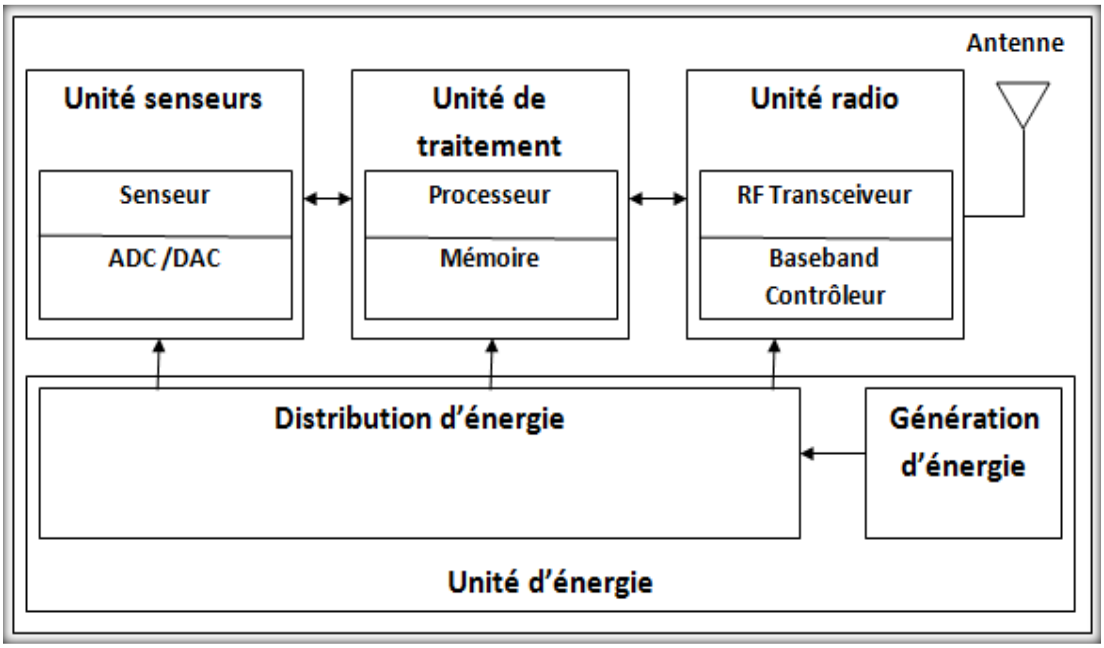

Figure 1.2 : Architecture d'un nœud de capteur [1].

❖ **L'unité de captage**

Elle est généralement composée de deux sous-unités : les capteurs et les convertisseurs analogique numérique ("ADC: Analog Digital Converter" en Anglais). Les capteurs permettent d'obtenir des mesures numériques sur les paramètres

environnementaux et les transforment en signaux analogiques que les ADC convertissent à leur tour en signaux numériques. [1]

- **L'unité de traitement**

Mote, processeur, Random Access Memory (RAM) et Flash : On appelle généralement Mote la carte physique utilisant le système d'exploitation pour fonctionner. Celle-ci a pour coeur le bloc constitué du processeur et des mémoires RAM et Flash. Cet ensemble est à la base du calcul binaire et du stockage, temporaire pour les données et définitif pour le système d'exploitation. Cette unité est chargée d'exécuter les protocoles de communications qui permettent de faire collaborer le noeud avec les autres noeuds du réseau. Elle peut aussi analyser les données captées pour alléger la tâche du noeud puits. [2]

- **L'unité de transmission**

Radio et antenne : les équipements étudiés sont donc généralement équipés d'une radio ainsi que d'une antenne. Cette unité est responsable d'effectuer toutes les émissions et réceptions des données sur un medium sans fil. Elle peut être de type *optique* (comme dans les noeuds Smart Dust), ou de type *radiofréquence*. Les communications de type optique sont robustes vis-à-vis des interférences électriques. Néanmoins, elles présentent l'inconvénient d'exiger une ligne de vue permanente entre les entités communicantes. Par conséquent, elles ne peuvent pas établir de liaisons à travers des obstacles.[2]

- **Unité de stockage (Mémoire)**

L'unité de stockage inclut la mémoire de programme (dont les instructions sont exécutées par le processeur) et la mémoire de données (pour conserver des données fournies par l'unité de captage et d'autres données locales). La taille de cette mémoire est souvent limitée essentiellement par les considérations économiques et s'améliorera aussi probablement au fil des années [3].

- **Unité de contrôle d'énergie**

Batterie : Un micro-capteur est muni d'une ressource énergétique pour alimenter tous ses composants. Cependant, en conséquence de sa taille réduite, la ressource énergétique dont il dispose est limitée et généralement irremplaçable. Cette unité peut

aussi gérer des systèmes de rechargement d'énergie à partir de l'environnement observé telles que les cellules solaires, afin d'étendre la durée de vie totale du réseau.[2]

1.2.2.2.Architecture logicielle (Systèmes d'exploitation):

Les avancées technologiques récentes ont permis de faire embarquer des systèmes d'exploitation (OS : Operating System) au sein des capteurs, mais leurs fonctionnalités restent toutefois limitées.

Les systèmes d'exploitation pour les RCSFs sont des interfaces informatiques spécifiques destinées au fonctionnement des capteurs dans les réseaux.

Le rôle du système d'exploitation pour un capteur en réseau est d'être l'interface entre les ressources matérielles et les applications distribuées. Il doit fournir une variété de services systèmes basiques comme la gestion de l'allocation des ressources sur les périphériques de matériels divers et la gestion et la planification des tâches. Le but du système d'exploitation est de faciliter la programmation des applications, mais aussi d'optimiser les utilisations des ressources.

Il existe plusieurs systèmes d'exploitation pour les RCSFs comme : TinyOS (Tiny Operating System), Contiki, MANTIS OS (MultimodAl NeTworks of In-situ micro Sensor OS), LiteOS, RETOS, Nano-RK [4], [5], [6]. Il y a certaines caractéristiques qui font la différence entre ces systèmes d'exploitation [4], par exemple : l'architecture, le modèle de programmation, la gestion de a mémoire, le langage de programmation. Le Tableau 1.1 fait une comparaison entre les caractéristiques de quelques systèmes d'exploitation.

Tableau 1.1 : Comparaison entre les caractéristiques de quelques systèmes d'exploitation [4]

Caractéristique/OS	Architecture	Modèle de programmation	Gestion de la mémoire	Langage de programmation
TinyOs	Monolithique	Evénementielle	Mémoire statique	NesC
Contiki	Modulaire	Evénementielle et multitâche	Mémoire dynamique	C
MANATIS	Sous forme des couches	Multitâche	Mémoire dynamique	C
Nano-RK	Monolithique	Multitâche	Mémoire statique	C
LiteOS	Modulaire	Evénementielle et multitâche	Mémoire dynamique	LiteC++

Parmi les systèmes d'exploitation actuels, nous décrivons les OS les plus utilisés dans le domaine scientifique qui sont : TinyOS, Contiki et MANTIS OS.

- **TinyOS**

TinyOS est un système d'exploitation open source pour les RCSFs qui trouve sa genèse au sein du laboratoire d'informatique de l'université de Berkeley et qui a été l'un des premiers systèmes d'exploitation conçus pour les réseaux de capteurs miniatures. En effet, TinyOS est le plus répandu des OS pour les réseaux de capteurs sans-fil. Il est capable d'intégrer très rapidement les innovations en relation avec l'avancement des applications et des réseaux eux-mêmes tout en minimisant la taille du code source en raison des problèmes inhérents de mémoire dans les réseaux de capteurs.

La librairie de TinyOS comprend des protocoles réseau, des applications de services distribués, des pilotes (drivers en anglais) de capteurs et des outils d'acquisition

des données. La contrainte énergétique due à l'autonomie des capteurs implique l'utilisation de puissance de calcul réduite. Cela entraîne le développement de logiciels contraint par la capacité de la mémoire et par la rapidité d'exécution. Les applications pour TinyOS sont écrites en langage de programmation NesC (Network Embedded System C), une extension du langage programmation C. L'utilisation du langage NesC permet l'optimisation du code et par suite réduit l'usage de la mémoire à accès aléatoire (RAM).

Un programme sous TinyOS ne doit comporter que les composants nécessaires à son exécution, ce qui réduit la taille du programme à insérer dans l'unité de traitement du capteur.

Un autre but de TinyOS est de prolonger la durée de vie du capteur. Dans cette optique, la programmation sous TinyOS est une programmation événementielle, c'est-à-dire que l'exécution des différentes instructions s'effectue en fonction des événements enregistrés par l'unité de traitement. Ce type de programmation est adapté aux capteurs car il n'y a pas de traitement que lors d'apparitions d'événements, ce qui permet au capteur de rester dans un état de veille le reste du temps afin de préserver son énergie.

Par contre, les programmes développés pour fonctionner sous le noyau TinyOS pourront être difficilement utilisables sous un autre système d'exploitation.

- **Contiki**

Contiki est également un système d'exploitation OS. C'est un système configurable modulaire pour les réseaux de capteurs. L'architecture hybride du noyau Contiki autorise deux modes de fonctionnement : soit multitâche, soit basé sur les événements. Contiki est un système d'exploitation conçu pour prendre le moins de place possible, avec une faible empreinte mémoire. Pour cela, le code est écrit en langage C.

Un système utilisant Contiki contient des processus, qui peuvent être des applications ou des services, c.à.d. un processus proposant des fonctionnalités à une ou plusieurs applications. La communication entre processus se fait par l'envoi d'événements. Le noyau Contiki reste, nativement, un système d'exploitation basé sur les événements. Pour obtenir le

mode multitâche, une bibliothèque doit être installée. Les fonctions associées à cette bibliothèque n'accèdent pas directement à l'ensemble des ressources du capteur sans fil. Elles doivent, dans certains cas, faire appel à la partie du noyau dédié à la gestion des événements. Cette structure à deux niveaux a pour conséquence une dégradation des performances du système quand le mode multitâche est activé.

- **MANTIS OS**

MANTIS (MultimodAl NeTworks of In-situ micro Sensor) OS apparu en 2005, a été conçu par l'université du Colorado [8]. C'est un système d'exploitation léger et multitâche pour les capteurs adapté aux applications où plusieurs traitements, chacun associé à un ou plusieurs processus, sont en concurrence pour accéder aux ressources du capteur sans fil.

Il dispose d'un environnement de développement Linux et Windows. La programmation d'application sur MANTIS OS se fait en langage C. Son empreinte mémoire est faible : 500 octets en mémoire RAM et 14 kilo-octets en mémoire flash. C'est un système modulaire dont le noyau supporte également des entrées/sorties synchrones et un ensemble de primitives de concurrence.

L'économie d'énergie est réalisée par MANTIS à l'aide d'une fonction de veille appelée sleep function qui désactive le capteur lorsque toutes les taches actives sont terminées. MANTIS est un système dynamique ; les modifications applicatives peuvent être réalisées pendant le fonctionnement. MANTIS apporte une compatibilité avec le modèle événementiel TinyOS à travers TinyMOS (MOS est la contraction de MantisOS (An Embedded Multithreaded Operating System for Wireless Micro Sensor Platforms)), dont son noyau est équipé.

1.2.3. Classification des capteurs

En adoptant une approche fondée sur la consommation énergétique, on peut classifier tous les capteurs sans fil en deux types: actif et passif.

1.2.3.1. Les capteurs Actifs

Généralement, un capteur actif est un système de mesure qui nécessite une source d'énergie embarquée, la plupart du temps assurée par une batterie, et ce pour la réalisation de la phase de traitement au cours de laquelle le signal est filtré (nettoyé), amplifié et converti dans un format compatible et exploitable. Dans ce cas, le capteur doit non seulement mesurer des propriétés physiques mais doit également effectuer des tâches additionnelles au travers de circuits de traitement et de communication intégrés [9]. Ce type de capteur est surtout utilisé pour assurer des mesures continues en temps réel.

1.2.3.2. Les capteurs passifs

Les capteurs passifs sont des dispositifs qui ne possèdent pas de source d'énergie embraquée et présentent l'avantage d'être facilement intégrables. Ce type de capteur est utilisé dans des applications spécifiques (surveillance environnementale [10], des instruments de suivis spatial et aéronautique , des applications liées à la santé) qui nécessitent des unités de mesure miniatures, passives, de grande précision et fiables. L'objectif est d'assurer des mesures à distance des grandeurs physiques [11]. Dans ce cas, deux différentes technologies peuvent être utilisées pour la transmission sans-fil de données : la transmission inductive et la transmission radio basée sur la réflexion (transpondeur passif). Dans ce qui suit, on présente brièvement les composants les plus répandus fondés sur ces types de technologies.

a. Les capteurs Inductifs [12]

Les capteurs inductifs sont des dispositifs composés d'un élément sensible et d'une antenne. Le capteur lui-même ne contient pas de source d'énergie, mais on y a intégré un condensateur et une inductance pour réaliser un circuit résonateur LC série pour la communication sans fil. Le capteur est interrogé à distance avec une ou deux antennes boucles moyennant un couplage inductif mutuel. Le couplage inductif est fondé sur un couplage magnétique entre la bobine interne et l'autre externe. Ces bobines forment un système de transformateur et de l'information peut être transférée par modulation de puissance. Le champ magnétique généré par la boucle d'émission induit

une force électromotrice au sein de la bobine reliée au capteur, qui à son tour génère une FEM vers la boucle extérieure. Vu que le capteur contient un circuit résonant LC série, la FEM du capteur est maximale à la fréquence de résonance. Donc, la FEM (Finite Element Method)de retour est aussi maximale à cette fréquence là. La fréquence de résonance du circuit LC change lorsque la valeur de la capacité change. Cette variation est récupérée à distance et se traduit par une variation dans les propriétés du couplage inductif. Le taux du transfert énergétique dépend aussi de la position et de l'orientation de la bobine interne par rapport à la bobine externe. Le transfert énergétique est maximal quand les axes des bobines coïncident. Il est détérioré si on a un mauvais alignement angulaire ou latéral. Dans le cas pratique, les deux boucles doivent être placées à des courtes distances (quelques centimètres à une dizaine de centimètre) l'une par rapport à l'autre.

b. Capteur radio frequency identification RFID

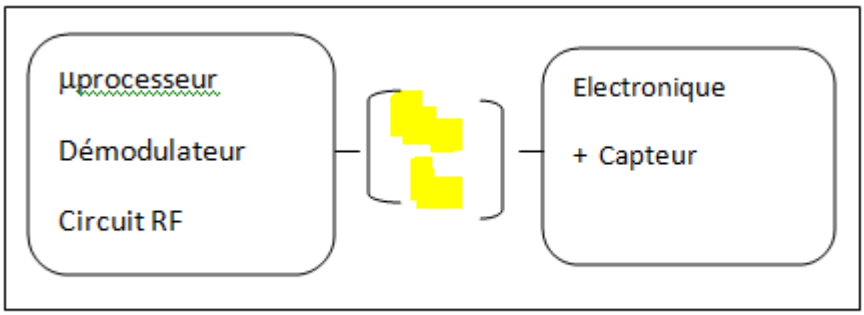

Figure 1.3 : Principe de fonctionnement d'un dispositif RFID

Les dispositifs RFID, connus essentiellement pour leur utilisation dans l'identification d'objets, peuvent être également exploités pour des applications de mesure. Le dispositif se présente sous la forme d'un microprocesseur (connecté à un élément sensible) doté d'une antenne (Figure 1.3). L'étiquette reçoit de l'énergie électrique par son antenne. Lorsqu'il reçoit suffisamment d'énergie pour être activé, le système RFID renvoie un message au lecteur, qui a la fonction de décodage.

Les performances d'un système RFID s'expriment en termes de distance et de vitesse de lecture. On peut classer les RFID en quatre catégories selon les bandes de fréquences dans lesquelles ils fonctionnent : - Les étiquettes BF (fréquence inférieure à 135 kHz) avec une distance de lecture de quelques centimètres. - Les étiquettes HF (fréquence de 13,56 MHz) avec une distance de lecture de quelques dizaines de centimètres. La plupart des puces passives utilisent cette bande de fréquences. - Les étiquettes UHF (868-950 MHz) avec une distance de lecture de l'ordre du mètre. - Les étiquettes UHF à 2,45 GHz.

c. Les **capteurs** *à ondes acoustiques de surface (SAW)*

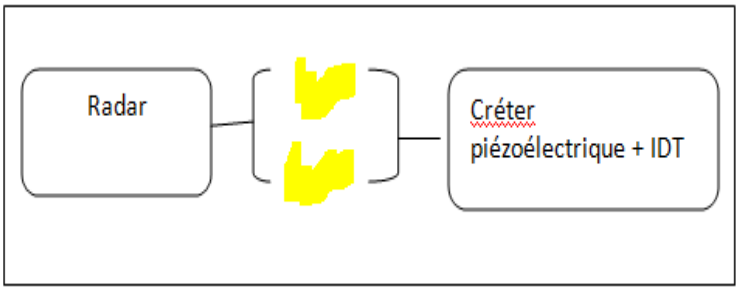

Figure 1. 4. Principe de fonctionnement d'un capteur SAW

Au début des années 1980, de nouveaux types de capteurs passifs ont été étudiés en utilisant des composants à ondes acoustiques de surface (SAW) [13] Le principe de fonctionnement (Figure 1.4) est le suivant : une onde électromagnétique (signal provenant d'un lecteur RF), transmise par l'antenne d'un système d'émission réception (E/R), est reçue par l'antenne du transducteur SAW passif. Ensuite, cette onde est transformée en une onde acoustique de surface via un transducteur interdigité (IDT) placé sur un substrat piézoélectrique connecté à l'antenne [14].

1.2.4. Types des capteurs

Il existe actuellement un grand nombre de capteurs, avec des fonctionnalités diverses et variées. Tous ces différents capteurs ne pourraient être décrits ici, cependant une liste exhaustive peut être trouvée sur le site The Sensor Network Museum [9].

La plupart des capteurs dépendent de l'application pour lesquels ils ont été conçus (capteur aquatique, sous-terrain, etc. . .). Il est plus intéressant de décrire les capteurs les plus utilisés et leur évolution au cours du temps.

En l'occurrence, la figure 1.5 illustre l'évolution des capteurs au cours de ces 20 dernières années. Cette représentation met en avant l'importance des travaux de recherche de l'université de Berkeley dans l'essor des réseaux de capteurs, surtout sachant que l'entreprise Xbow (aussi appelé Crossbow) qui fait jusqu'à aujourd'hui office de référence dans la fabrication de capteurs est née au sein de la célèbre université californienne.

Figure 1.5: Quelques exemples de capteurs [16]

Les capteurs fabriqués par Xbow au cours des dix dernières années (famille de capteurs Mica et Telos) sont sans aucun doute les plus utilisés dans les expériences et travaux de recherche. Ces capteurs sont capables de mesurer plusieurs métriques (température, humidité, luminosité, etc. . .) et s'articulent pour la plupart d'entre eux autour du Chipcon CC2420 qui est devenu le standard au niveau des modules de transmission utilisant le protocole de communication IEEE 802.15.4.

Dans ce qui suit, nous présentons différents modèles des capteurs **: [15]**

1.2.4.1.Mica2Dot

Figure 1.6 : Capteur Mica2Dot

Fabriquant : Crossbow

Caractéristiques

- Durée de vie des piles > 1 an, 1 pile AA
- Transceiveur radio multi canal 433, 868/916 ou 310 MHz
- Capteurs de lumière, température, humidité relative, pression barométrique, accélération / sismique, acoustique, magnétique, GPS et autres disponibles

1.2.4.2.MicaZ

Figure 1.7 : Capteur MicaZ

Fabriquant : Crossbow

Caractéristiques

- Durée de vie des piles > 1 an, 2 piles AA
- Radio compatible 2,4 GHz, IEEE 802.15.4
- Capteurs de lumière, température, humidité relative, pression barométrique, accélération / sismique, acoustique, magnétique, GPS et autres disponibles
- 58x32x7 mm

1.2.4.3.Tmote Sky

Figure 1.8 : Capteur Tmote Sky

1.2.4.4. Fabriquant : Moteiv

Caractéristiques

- 2 piles AA ou USB
- Microcontrôleur TI MSP430F1611 à 8 MHz maximum
- SRAM 10k, flash 48k + stockage en série 1024k
- Transceiveur sans fil 250kbps 2,4 GHz Chipcon CC2420 IEEE 802.15.4
- Capteurs d'humidité, de température et de lumière embarqués
- Consommation de courant ultra-faible

- Réveil rapide du sommeil (<6usec)
- Programmation et interface via USB
- 32x80 mm.

Tableau 1.2. : Comparaison entre les caractéristiques de quelques capteurs

Caractéristiques / Type capteur	Fabriquant	Durée de vie des piles	Radio compatible	Taille	Autre caractéristique
Tmote Sky	Moteiv	> 1 an ,2 piles AA ou USB	Microcontrôleur TI MSP430F1611 à 8 MHz maximum	32x80 mm	Capteurs d'humidité, de température et de lumière embarqués
Micaz	Crossbow	> 1 an, 1 pile AA	2,4 GHz, IEEE 802.15.4	58x32x7 mm	Capteurs de lumière, température, humidité relative, pression barométrique
Mica2Dot	Crossbow	> 1 an, 1 pile AA	radio multi canal 433, 868/916 ou 310 MHz	/	Capteurs de lumière, température, humidité relative, pression barométrique,

1.3.Réseaux de capteurs sans fil

1.3.1. Définition

Un Réseau de Capteurs Sans Fil (RCSF) est un ensemble de dispositifs très petits, nommés nœuds capteurs, variant de quelques dizaines d'éléments à plusieurs milliers.

Dans ces réseaux, chaque nœud est capable de surveiller son environnement et de réagir en cas de besoin en envoyant l'information collectée à un ou plusieurs points de collecte, à l'aide d'une connexion sans fil [17].

Dans le cas d'un réseau de petite échelle (corps humain par exemple), les capteurs seront dans le voisinage direct du puits (un réseau de type étoile à un saut).

Cependant, dans le cas d'un réseau à grande échelle (forêt, barrage, champs de batail…), les capteurs ne sont pas tous dans le voisinage du puits et les messages seront acheminés du nœud source vers le puits en transitant par plusieurs nœuds, selon un mode de communication multi-sauts comme le l'illustre la Figure 1.9.

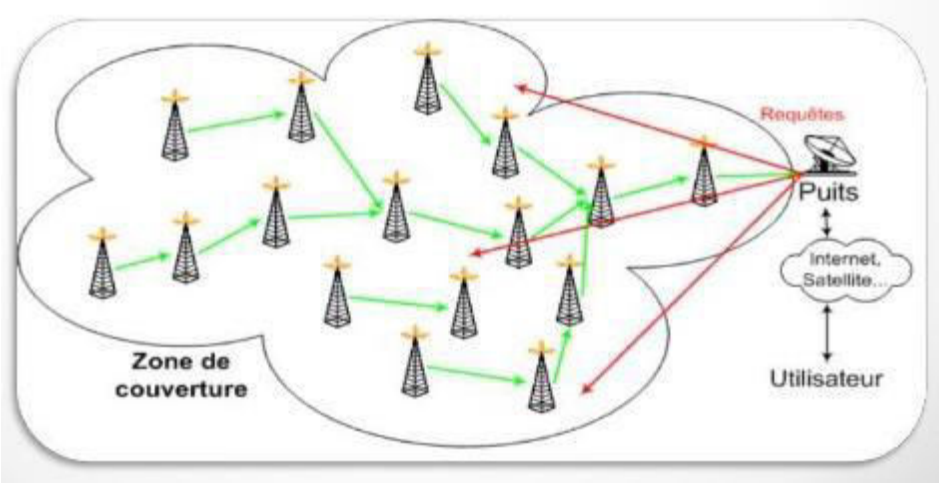

Figure 1.9 : Réseau de capteurs sans fil [18]

1.3.2. Architecture d'un RCSF

L'architecture des RCSFs utilise beaucoup de sources. Historiquement, beaucoup du travail relatif a été effectué dans le contexte des réseaux à auto-organisation, mobiles

et Ad Hoc Un réseau de capteurs est constitué essentiellement de : plusieurs nœuds capteurs, un nœud Sink et un centre de traitement des données.

❖ **Nœuds :** sont des capteurs, leur type, leur architecture et leur disposition géographique dépendent de l'exigence de l'application en question. Leur énergie est souvent limitée puisqu'ils sont alimentés par des piles.

❖ **Sink (station de base) :** c'est un nœud particulier du réseau. Il est chargé de la collecte des données issues des différents nœuds du réseau. Il doit être toujours actif puisque l'arrivée des informations est aléatoire. C'est pourquoi son énergie doit être illimitée. Dans un réseau de capteur sans fils plus ou moins large et à charge un peu élevée, on peut trouver deux sinks ou plus pour alléger la charge.

❖ **Centre de traitement des données :** c'est le centre vers lequel les données collectées par le sink sont envoyées. Ce centre a le rôle de regrouper les données issues des nœuds et les traiter de façon à en extraire de l'information utile exploitable. Le centre de traitement peut être éloigné du sink[19], alors les données doivent être transférées à travers un autre réseau, c'est pourquoi on introduit une passerelle entre le sink et le réseau de transfert pour adapter le type de données au type du canal (comme c'est illustré dans la Figure 1.10).

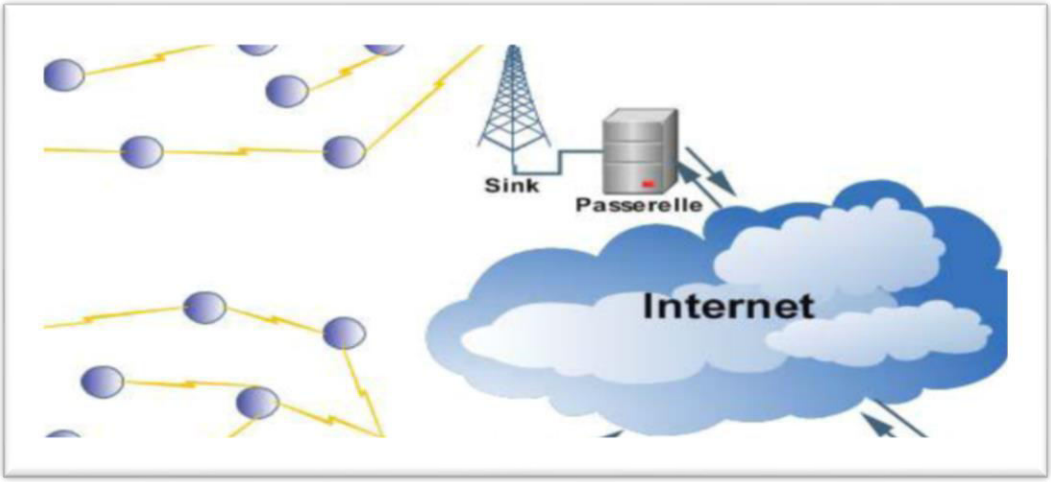

Figure 1.10: Architecture générale d'un réseau de capteurs sans fil. [19]

1.3.3. Caractéristiques d'un réseau de capteurs sans fil

Parmi les caractéristiques les plus importantes d'un réseau de capteurs sans fil, nous citons [20] :

1.3.3.1.La durée de vie limitée : Les nœuds capteurs sont très limités par la contrainte d'énergie, ils fonctionnent habituellement sans surveillance dans des régions géographiques éloignées. Par conséquent recharger ou remplacer leurs batteries devient quasiment impossible.

1.3.3.2. Ressources limitées : Habituellement les nœuds capteurs ont une taille très petite, ce facteur de forme limite la quantité de ressources qui peuvent être mises dans ces nœuds. En conséquence, la capacité de traitement et de mémoire est très limitée.

1.3.3.3.Topologie dynamique : La topologie des réseaux de capteurs change d'une manière fréquente et rapide car les nœuds capteurs peuvent être déployés dans des environnements hostiles (par exemple un champ de bataille), la défaillance d'un nœud capteur peut donc être très probable. De plus, les nœuds capteurs et les nœuds finaux où ils doivent envoyer l'information capturée peuvent être mobiles.

1.3.3.4. Agrégation des données : Dans les réseaux de capteurs, les données produites par les nœuds capteurs sont très reliées, ce qui implique l'existence de redondances de données. Une approche répandue consiste à agréger les données au niveau des nœuds intermédiaires afin de réduire la consommation d'énergie lors de la transmission de ces données.

1.3.3.5. La scalabilité : les réseaux de capteurs engendrent un très grand nombre de capteurs, ils peuvent atteindre des milliers voire des millions de capteurs. Le défi à relever par les RCSFs est d'être capable de maintenir leurs performances avec ce grand nombre de capteurs.

1.3.3.6.Bande passante limitée : En raison de la puissance limitée, les noeuds capteurs ne peuvent pas supporter des débits élevés.

1.3.3.7. Sécurité physique limitée: cela se justifie par les contraintes et limitations physiques qui minimisent le contrôle des données transmises.

1.3.4. Contraintes de conception des RCSFs

Les principaux facteurs et contraintes influençant l'architecture des réseaux de capteurs peuvent être résumés comme suit [1] :

1.3.4.1.La tolérance aux fautes [21] : la tolérance aux fautes est la capacité de maintenir les fonctionnalités du réseau en présence de fautes. La fiabilité des réseaux de capteurs

sans fil est affectée par des défauts qui se produisent à cause de diverses raisons telles que le mauvais fonctionnement du matériel ou à cause d'un manque d'énergie. Ces problèmes n'affectent pas le reste du réseau.

1.3.4.2.Le facteur d'échelle (Scalability) [22]: le nombre de nœuds de capteurs augmente sur un réseau sans fil et ce nombre peut atteindre le million. Un nombre aussi important de nœuds engendre beaucoup de transmissions entre les nœuds et peut imposer des difficultés pour le transfert de données.

1.3.4.3.Les coûts de production [23]: souvent les réseaux de capteurs sont composés d'un très grand nombre de nœuds. Le prix d'un nœud est critique afin de pouvoir concurrencer un réseau de surveillance traditionnel.

1.3.4.4.L'environnement : les capteurs sont souvent déployés en masse dans des endroits tels que des champs de bataille, à l'intérieur de grandes machines, au fond d'un océan, dans des champs biologiquement ou chimiquement souillés [24],... Par conséquent, ils doivent pouvoir fonctionner sans surveillance dans des régions géographiques éloignées.

1.3.4.5.La topologie de réseau [25]: le déploiement d'un grand nombre de nœuds nécessite une maintenance de la topologie. Cette maintenance consiste en trois phases : déploiement, post-déploiement (les capteurs peuvent bouger, ne plus fonctionner,...) et redéploiement de nœuds additionnels.

1.3.4.6.Les contraintes matérielles [26]: la principale contrainte matérielle est la taille du capteur. Les autres contraintes sont la consommation d'énergie qui doit être moindre pour que le réseau survive le plus longtemps possible, qu'il s'adapte aux différents environnements (fortes chaleurs, eau,..), qu'il soit autonome et très résistant vu qu'il est souvent déployé dans des environnements hostiles.

1.3.4.7.Les médias de transmission : dans un réseau de capteurs, les nœuds sont reliés par une architecture sans fil. Pour permettre des opérations sur ces réseaux dans le monde entier, le média de transmission doit être standardisé. On utilise le plus souvent l'infrarouge, le Bluetooth [27] et les communications radio Zig Bee [28].

1.3.4.8.La consommation d'énergie [29]: un capteur, de par sa taille, est limité en énergie (<1.2V). Dans la plupart des cas le remplacement de la batterie est impossible. Ce qui veut dire que la durée de vie d'un capteur dépend grandement de la durée de vie de la

batterie. Dans un réseau de capteurs (multi-sauts) chaque nœuds collecte des données et envoie/transmet des valeurs. Le dysfonctionnement de quelques nœuds nécessite un changement de la topologie du réseau et un re-routage des paquets. Toutes ces opérations sont gourmandes en énergie, c'est pour cette raison que les recherches actuelles se concentrent principalement sur les moyens de réduire cette consommation [30].

1.4. Pile protocolaire [1]

Le rôle de ce modèle consiste à standardiser la communication entre les composants du réseau afin que différents constructeurs puissent mettre au point des produits (logiciels ou matériels) compatibles. Ce modèle comprend 5 couches qui ont les mêmes fonctions que celles du modèle OSI ainsi que 3 couches pour la gestion de la puissance d'énergie, la gestion de la mobilité ainsi que la gestion des tâches (interrogation du réseau de capteurs). Le but d'un système en couches est de séparer le problème en différentes parties (les couches) selon leur niveau d'abstraction. Chaque couche du modèle communique avec une couche adjacente (celle du dessus ou celle du dessous). Chaque couche utilise ainsi les services des couches inférieures et en fournit à celle de niveau supérieur .

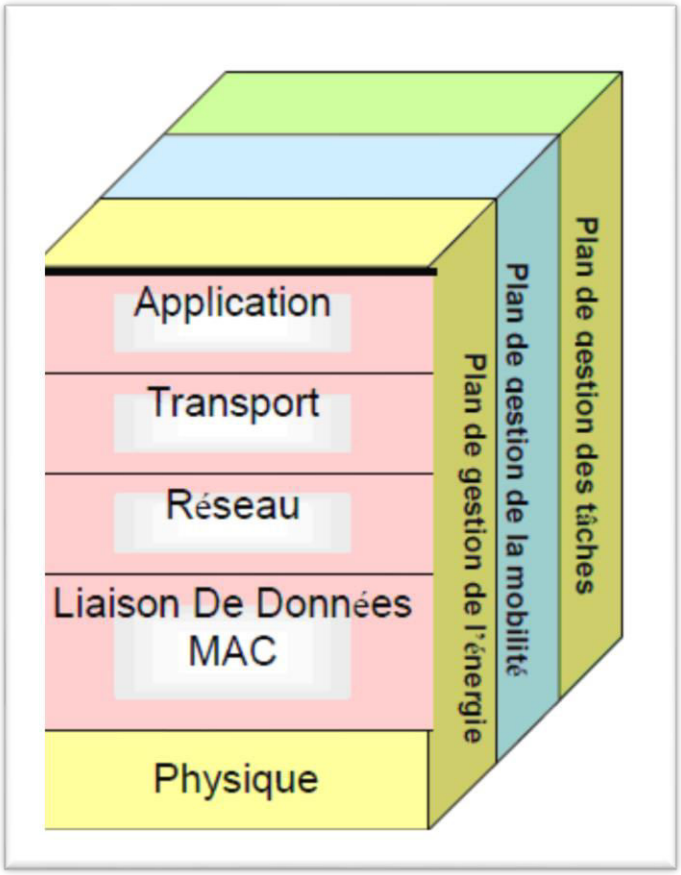

Figure 1.11 : La pile protocolaire des RCSFs [1]

1.4.1. Rôle des couches

- **La couche physique** : Spécifications des caractéristiques matérielles, des fréquences porteuses, etc.
- **La couche liaison** : Spécifie comment les données sont expédiées entre deux nœuds/routeurs dans une distance d'un saut. Elle est responsable du multiplexage des données, du contrôle d'erreurs, de l'accès au media,... Elle assure la liaison point à point et multi-point dans un réseau de communication.
- **La couche réseau** : Dans la couche réseau le but principal est de trouver une route et une transmission fiable des données, captées, des nœuds capteurs vers le puits "sink" en optimisant l'utilisation de l'énergie des capteurs. Ce routage diffère de celui des réseaux de transmission ad hoc sans fils par les caractéristiques suivantes:

  - Il n'est pas possible d'établir un système d'adressage global pour le grand nombre de nœuds.

- Les applications des réseaux de capteurs exigent l'écoulement des données mesurées de sources multiples à un puits particulier.
- Les multiples capteurs peuvent produire de mêmes données à proximité d'un phénomène (redondance).
- Les nœuds capteur exigent ainsi une gestion soigneuse des ressources. En raison de ces différences, plusieurs nouveaux algorithmes ont été proposés pour le problème de routage dans les réseaux de capteurs.

➢ **La couche transport** : Cette couche est chargée du transport des données, de leur découpage en paquets, du contrôle de flux, de la conservation de l'ordre des paquets et de la gestion des éventuelles erreurs de transmission.

➢ **La couche application** : Cette couche assure l'interface avec les applications.

Il s'agit donc du niveau le plus proche des utilisateurs, géré directement par les logiciels.

1.4.2. Plans de gestion

Les plans de gestion d'énergie, de mobilité et de tâche contrôlent l'énergie, le mouvement et la distribution de tâche au sein d'un nœud capteur. Ces plans aident les nœuds capteurs à coordonner la tâche de captage et minimiser la consommation d'énergie. Ils sont donc nécessaires pour que les nœuds capteurs puissent collaborer ensemble, acheminer les données dans un réseau mobile et partager les ressources entre eux en utilisant efficacement l'énergie disponible. Ainsi, le réseau peut prolonger sa durée de vie.

➢ **Plan de gestion d'énergie :** contrôle l'utilisation de la batterie. Par exemple, après la réception d'un message, le capteur éteint son récepteur afin d'éviter la duplication des messages déjà reçus. En outre, si le niveau d'énergie devient bas, le nœud diffuse à ses voisins une alerte les informant qu'il ne peut pas participer au routage. L'énergie restante est réservée au captage.

➢ **Plan de gestion de mobilité :** détecte et enregistre le mouvement du nœud capteur. Ainsi, un retour arrière vers l'utilisateur est toujours maintenu et le nœud peut garder trace de ses nœuds voisins. En déterminant leurs voisins, les nœuds capteurs peuvent balancer l'utilisation de leur énergie et la réalisation de tâche.

- **Plan de gestion de tâche :** balance et ordonnance les différentes tâches de captage de données dans une région spécifique. Il n'est pas nécessaire que tous les nœuds de cette région effectuent la tâche de captage au même temps ; certains nœuds exécutent cette tâche plus que d'autres selon leur niveau de batterie.

1.5. Applications des RCSFs

Au cours des dernières années, les RCSFs ont pris de l'ampleur en raison de la progression des unités et des améliorations dans les technologies radio de faible puissance. À l'origine, la motivation initiale derrière la recherche sur les RCSFs était des applications militaires.

Comme les coûts de nœuds de capteurs et des réseaux de communication ont été réduits, de nombreuses autres applications potentielles, y compris celles utilisées à des fins civiles et environnementales ont vu le jour. Dans ce qui suit, une explication plus détaillée des différentes applications est présentée.

1.5.1. Découverte de catastrophes naturelles : on peut créer un réseau autonome en dispersant les nœuds dans la nature. Des capteurs peuvent ainsi signaler des événements tels que les feux de forêts, les tempêtes ou les inondations. Ceci permet une intervention beaucoup plus rapide et efficace des secours [7].

Figure 1 .12 : Découverte de catastrophes naturelles par des RCSFs.

1.5.2. Détection d'intrusions *:* en plaçant à différents points stratégiques des capteurs, on peut ainsi prévenir des cambriolages ou des passages de gibier sur une voie de chemin de fer (par exemple) sans avoir à recourir à de coûteux dispositifs de surveillance vidéo.

Figure 1.13 : Application des capteurs dans la détection d'intrusions

1.5.3. Surveillance de l'environnement

Ce domaine couvre le plus grand champ d'application des RCSFs de nos jours. En effet, il englobe la surveillance de l'état de l'air ambiant, la qualité des eaux (par exemple le contrôle de leur degré de pollution, etc), le contrôle des environnements dangereux surtout ceux qui sont vulnérables aux incendies, inondation ou glissement de terrain. En outre, ce domaine d'application inclut les prévisions de la météo.[32]

1.5.4. Surveillance de la santé

L'implantation des capteurs à des fins de santé est un domaine interdisciplinaire qui a révolutionné le système de soins à distance en permettant la surveillance de la santé peu coûteuse et continue avec des mises à jour en temps réel des dossiers médicaux via Internet.

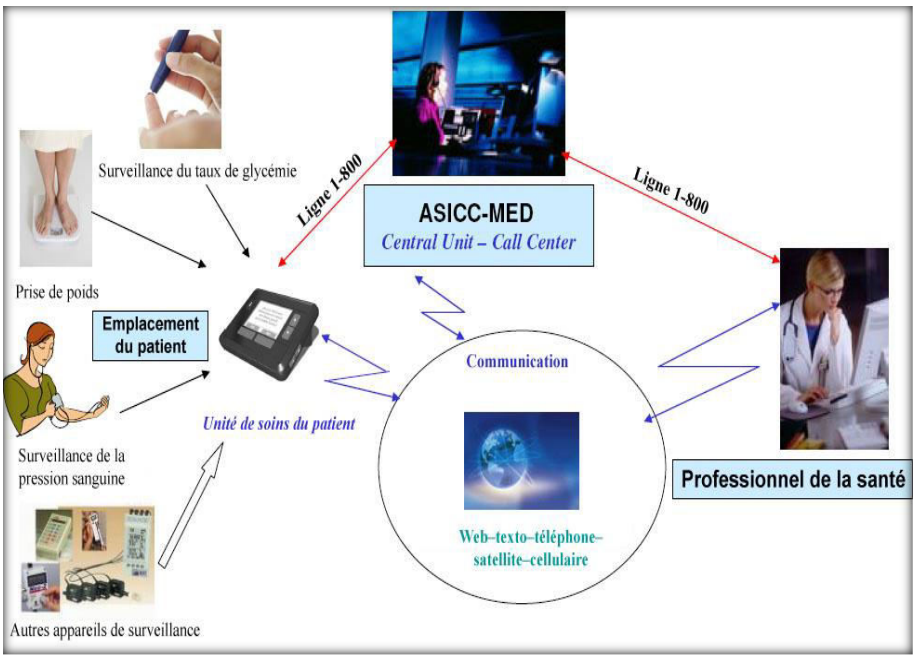

Figure 1.14 : Application des RCSFs dans la surveillance de la santé .

1.5.5. Agriculture : des nœuds peuvent être incorporés dans la terre et on peut interroger le réseau sur l'état du champ et déterminer par exemple les secteurs les plus secs afin de les arroser en priorité. On peut aussi imaginer équiper des troupeaux de bétail de capteurs pour connaître en tout temps, leur position ce qui éviterait aux éleveurs d'avoir recours à des chiens de berger.

En effet, les RCSFs peuvent être intégrés dans le bâtiment d'un hôpital pour suivre et surveiller les patients et toutes les ressources médicales. Il existe des types spéciaux de capteurs qui peuvent mesurer la pression artérielle, la température corporelle et des capteurs sous forme d'électrocardiographes qui peuvent même être tricotés en vêtements pour fournir des soins infirmiers à distance pour les personnes âgées. [33]

Figure 1.15 : Application des RCSFs dans l'Agriculture.

1.5.6. Contrôle du trafic routier :

Les réseaux de capteurs sont utilisés pour la surveillance des déplacements des véhicules ainsi que les feux de circulation. Les caméras vidéo sont également fréquemment utilisées pour surveiller les segments de route à fort trafic et permettent entre autres d'aider les conducteurs à trouver des places de stationnement inoccupées et éviter les embouteillages. [33]

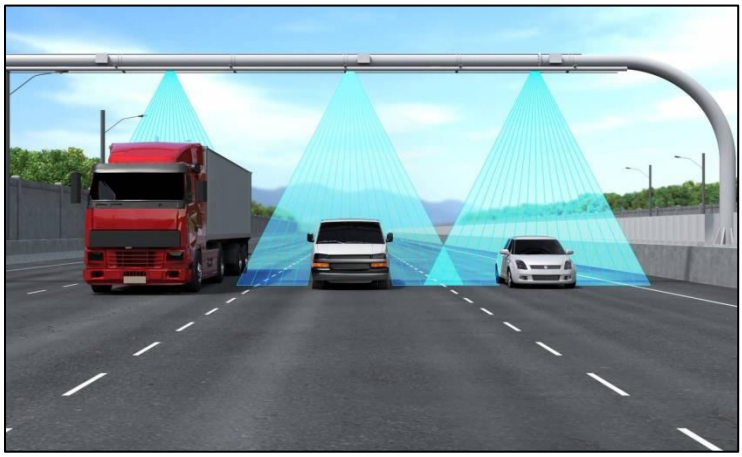

Figure 1.16 : Application des RCSFs pour la Contrôle du trafic routier [7]

1.6. Conclusion

Les réseaux de capteurs constituent un sujet de recherche relativement récent et pour tenter de résoudre tous les problèmes soulevés dans ce contexte de nombreuses directions en termes d'algorithmes, protocoles, etc.… Les aspects énergétiques et routage constituent les contraintes majeures.

Dans ce chapitre nous avons présenté une généralité sur les RCSFs. Nous commençons par une définition d'un capteur, son architecture, ses types etc. Ensuite nous présentons les RCSFs, leur architecture, la pile protocolaire et enfin leurs domaine d'application

Dans le chapitre suivant, nous introduirons en détail les techniques de conservation d'énergie dans les RCSFs.

Chapitre 2

Techniques de conservation d'énergie dans les RCSFs

Sommaire

2.1. Introduction

2.2 . Durée de vie d'un réseau de capteurs

2.3. Consommation d'énergie dans les RCSFs

2.4. Facteurs intervenants dans la consommation d'énergie

2.5. Méthodes de conservation d'énergie

2.6. Conclusion

2.1. Introduction

Dans les réseaux de capteurs, la consommation d'énergie est très importante puisque généralement les capteurs sont déployés dans des zones inaccessibles. Ainsi, il est difficile voire impossible de remplacer les batteries après leur épuisement. De ce fait, la consommation d'énergie au niveau des capteurs a une grande influence sur la durée de vie du réseau et plusieurs travaux se concentrent à la conservation d'énergie dans la communication en concevant des protocoles de contrôle d'accès au medium (MAC) spécialement pour ces réseaux.

Dans ce chapitre nous présentons les techniques de conservation d'énergie dans les RCSFs .Nous commençons par une définition de la durée de vie d'un réseau. Ensuite, la consommation d'énergie dans les RCSFs, Puis, les facteurs intervenants dans la consommation d'énergie et enfin les méthodes de conservation d'énergie.

2.2. Notion de durée de vie d'un réseau

La durée de vie d'un réseau est considérée comme la mesure la plus importante pour l'évaluation des réseaux de capteurs. Dans un environnement aux ressources limitées, la consommation énergétique de chaque nœud capteur doit être envisagée. Un réseau ne peut accomplir son objectif que tant qu'il est « en vie », mais pas au delà. La durée de vie prévue est critique dans tout déploiement de capteurs.

Il existe plusieurs définitions de cette notion dans la littérature. Dans cette section, nous en résumons les définitions les plus courantes. Nous en distinguons trois catégories principales qui sont détaillées ci-dessous :

2.2.1. Durée de vie basée sur le nombre de nœuds vivants

Dans cette définition, la durée de vie du réseau est définie par la durée jusqu'au moment où un certain nombre de nœuds épuise leur batterie. Il existe plusieurs définitions possibles dans cette catégorie selon le nombre de nœuds défaillants à considérer.

Dans [34], [35], [36] la durée de vie d'un réseau est définie par la durée jusqu'à ce que le premier nœuds épuise toute son énergie. Certains auteurs excluent les stations de base dans le calcul de la durée de vie du réseau puisque celles-ci sont généralement alimentées par le

courant continu [37].Ces définitions sont dites, durée de vie N-de-N (n-of-n lifetime). La durée de vie T_n^n est définie par l'équation suivante :

$$T_n^n = \underset{n \in v}{Min}\ T_v \qquad \textit{(Equation 2.1)}$$

T_v étant la durée de vie du nœud v et n étant le nombre de nœuds.

Dans un modèle de réseau hiérarchique, la durée de vie est définie par la durée jusqu'à ce que le premier clusterhead épuise sa batterie [38]. Une autre variante de la définition de la durée de vie d'un réseau peut aussi s'exprimer en fonction du nombre de nœuds vivants.

Dans [39], [40], la durée de vie est calculée par la période de temps écoulée jusqu'à ce qu'il reste au plus une fraction β de nœuds vivants dans le réseau. Cette définition est dite durée de vie K-de-N (k-of-n lifetime). Une autre définition consiste à considérer la durée jusqu'au moment où 50% des nœuds épuisent leur bénergie. [41]

Dans [42], la durée de vie d'un réseau est exprimée par la durée jusqu'à ce que tous les capteurs épuisent leur énergie. Bien que cette catégorie de définition soit la plus simple, certaines définitions de la durée de vie peuvent s'avérer imprécises. Prenons l'exemple de la définition durée de vie K-de-N et nous considérons le cas de k'< k nœuds qui sont à des positions stratégiques (autour de la SB par exemple) et qui épuisent leurs énergies. Le reste des nœuds ne peuvent désormais transférer leurs messages au Sink. Ainsi, le réseau est considéré comme non fonctionnel.

Pourtant la définition de la durée de vie ne reconnaît pas cette situation et attend jusqu'il y ait exactement k nœuds défaillants pour considérer le réseau comme non fonctionnel.

2.2.2. Durée de vie basée sur la connectivité

Dans [43], la connectivité dans un réseau de capteurs est définie par le pourcentage de nœuds qui ont des chemins vers la station de base(SB). Ainsi, la durée de vie du réseau est la durée pendant laquelle un pourcentage donné de nœuds possèdent un chemin vers la SB. Plusieurs travaux ont associé la métrique de couverture à celle de connectivité. Par exemple, [44] définit la durée de vie du réseau par la période jusqu'à ce que la couverture ou la connectivité du réseau devienne inférieur à un seuil prédéfini. Dans ce cas, la couverture est mesurée en termes de α_couverture. La connectivité est mesurée en termes de taux de messages délivrés au Sink .

2.2.3. Durée de vie basée sur la couverture

La couverture dans les RCSF est souvent considérée comme étant une mesure de performance très importante. Elle reflète la façon dont une zone donnée est surveillée (contrôlée), c'est-à-dire comment chaque point de la zone de surveillance est observée et suivie par l'ensemble des nœuds capteurs [45]. Ainsi, la notion de couverture dans les RCSF peut être vue comme une mesure de la qualité de service (QoS). Néanmoins, ce paramètre n'est pas suffisant pour la définition de la durée de vie puisqu'il n'est jamais garanti que les données mesurées soient transmises au nœud collecteur [44]. Parmi les définitions de durée de vie dans la littérature :

- **K_couverture** : la durée pendant laquelle la zone d'intérêt est couverte par au moins k nœuds [46] ;
- **100 %_couverture :**

- La durée pendant laquelle chaque cible est couverte par au moins un nœud [47 ,48] ;
- La durée pendant laquelle l'ensemble de la zone est couverte par au moins un nœud[49,50]

- **α_couverture :**

- La durée cumulée, au bout de laquelle au moins une portion de la région est couverte par au moins un nœud [51];
- La durée pendant laquelle la couverture tombe en-dessous d'un seuil prédéfinir [52] ;
- La durée de fonctionnement continu du système avant que la couverture ou la proportion de paquets reçus tombent en-dessous d'un seuil prédéfini [44] ;

2.2.4. Durée de vie basée sur des les exigences de QoS

Certains travaux définissent la durée de vie du réseau en termes de paramètre de QoS requise pour l'application. Dans [53], la durée de vie est définie par la période pendant laquelle le réseau satisfait la QoS requise.[42] définit la durée de vie du réseau comme étant la période jusqu'à ce que le réseau n'est plus capable de fournir un taux acceptable de détection d'événements. Cette définition est vague, puisqu'on ne spécifie pas le paramètre de QoS à considérer, et non précise.

2.3. Consommation d'énergie dans les RCSF

L'énergie consommée par un nœud capteur est due essentiellement aux opérations suivantes : la capture, le traitement et la communication de données [54].

2.3.1. Energie de capture

L'énergie de capture est dissipée pour accomplir les tâches suivantes : échantillonnage, traitement de signal, conversion analogique/numérique et activation de la sonde de capture. En général, l'énergie de capture représente un faible pourcentage de l'énergie totale consommée par un nœud.

2.3.2. Energie de traitement

L'énergie de traitement se divise en deux parties : l'énergie de commutation et l'énergie de fuite. L'énergie de commutation est déterminée par la tension d'alimentation et la capacité totale commutée au niveau logiciel (en exécutant un logiciel). Par contre, l'énergie de fuite correspond à l'énergie consommée lorsque l'unité de calcul n'effectue aucun traitement.

En général, l'énergie de traitement est faible par rapport à celle nécessaire pour la communication.

2.3.3. Energie de communication

L'énergie de communication se décline en deux parties : l'énergie de réception et l'énergie de l'émission. Cette énergie est déterminée par la quantité des données à communiquer et la distance de transmission, ainsi que par les propriétés physiques du module radio. L'émission d'un signal est caractérisée par sa puissance. Quand la puissance d'émission est élevée, le signal aura une grande portée et l'énergie consommée sera plus élevée. Notons que l'énergie de communication représente la portion la plus grande de l'énergie consommée par un nœud capteur.

2.4. Facteurs intervenants dans la consommation d'énergie

La consommation d'énergie dépend de plusieurs facteurs qui sont expliqués ci-dessous :

2.4.1. Etat du module radio

Le module radio est le composant du nœud capteur qui consomme le plus d'énergie, puisque c'est lui qui assure la communication entre les nœuds. On distingue quatre états des composants radio (transmetteur et récepteur) : actif, réception, transmission et sommeil **[55]**.

- **Etat actif***:* la radio est allumée, mais elle n'est pas employée. En d'autres termes, le nœud capteur n'est ni en train de recevoir ni de transmettre. Cet état provoque une perte de l'énergie suite à l'écoute inutile du canal de transmission.
- **Etat *sommeil:*** la radio est mise hors tension.
- **Etat transmission:** la radio transmet un paquet.
- **Etat réception:** la radio reçoit un paquet.

Il est aussi à noter que le passage fréquent de l'état actif à l'état sommeil peut avoir comme conséquence une consommation d'énergie plus importante que de laisser le module radio en mode actif. Ceci est dû à la puissance nécessaire pour la mise sous tension du module radio. Cette énergie est appelée l'énergie de transition. Il est ainsi souhaitable d'arrêter complètement la radio plutôt que de transiter dans le mode sommeil. Le changement d'état du module radio doit être géré par un protocole de la couche MAC.

2.4.2. Accès au medium de transmission(MAC)

La couche MAC joue un rôle important pour la coordination entre les nœuds et la minimisation de la consommation d'énergie. Dans cette section, nous allons analyser les principales causes de consommation d'énergie au niveau de la couche MAC **[56]**.

2.4.2.1. Retransmission

Les nœuds capteurs possèdent en général une seule antenne radio et partagent le même canal de transmission. Par ailleurs, la transmission simultanée des données provenant de plusieurs capteurs peut produire des collisions et ainsi une perte de l'information transmise. La retransmission des paquets perdus peut engendrer une perte significative de l'énergie.

2.4.2.2. Ecoute active

L'écoute active (*idle listening*) du canal pour une éventuelle réception de paquet qui ne sera pas reçu peut engendrer une perte importante de la capacité des nœuds en énergie. Pour éviter ce problème, il faut basculer les nœuds dans le mode sommeil le plus longtemps possible.

2.4.2.3. Surécoute

Le phénomène de surécoute (*overhearing*) se produit quand un nœud reçoit des paquets qui ne lui sont pas destinés (Figure 2.1).

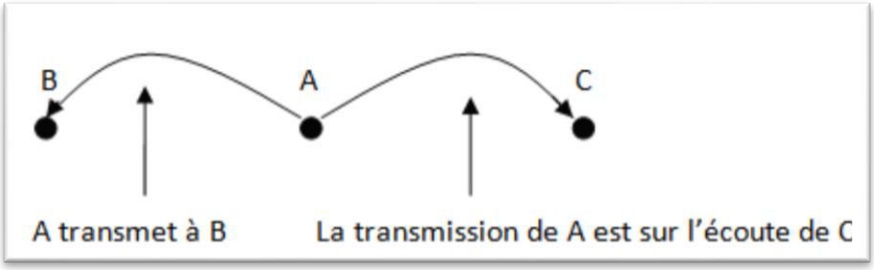

Figure 2.1: La surécoute dans une transmission.

La surécoute conduit à une perte d'énergie additionnelle à cause de l'implication des autres capteurs dans la réception des données.

2.4.2.4. Surcharge

Plusieurs protocoles de la couche MAC fonctionnent par échange de messages de contrôle (*overhead*) pour assurer différentes fonctionnalités: signalisation, connectivité, établissement de plan d'accès et évitement de collisions. Tous ces messages nécessitent une énergie additionnelle.

2.4.2.5. Surémission

Le phénomène de surémission (*overemitting*) se produit quand un nœud capteur envoie les données à un destinataire qui n'est pas prêt à les recevoir. En effet, les messages envoyés sont considérés inutiles et consomment une énergie additionnelle.

2.4.2.6. Taille des paquets

La taille des messages échangés dans le réseau a un effet sur la consommation d'énergie des nœuds émetteurs et récepteurs. Ainsi, la taille des paquets ne doit être ni trop élevée ni trop faible.

En effet, si elle est petite, le nombre de paquets de contrôle (acquittement) générés augmente l'*overhead*. Dans le cas contraire, une grande puissance de transmission est nécessaire pour des paquets de grande taille.

2.4.3. Modèle de propagation radio

Le modèle de propagation représente une estimation de la puissance moyenne reçue du signal radio à une distance donnée d'un émetteur. La propagation du signal radio est généralement soumise à différents phénomènes: la réflexion, la diffraction et la dispersion par divers objets.

Généralement, la puissance du signal reçue est de l'ordre de ($1/d^n$, où d est la distance entre l'émetteur et le récepteur, n un exposant de perte d'un chemin (Exemple : n=2 dans le vide, de 4 à 6 dans un immeuble) **[57]**.

2.5. Méthodes de conservation d'énergie dans les RCSF

nous distinguons trois méthodes de conservation d'énergie : Data-cycling, approches orientées données et la mobilité . Nous détaillerons dans ce qui suit chaque classe de méthodes :

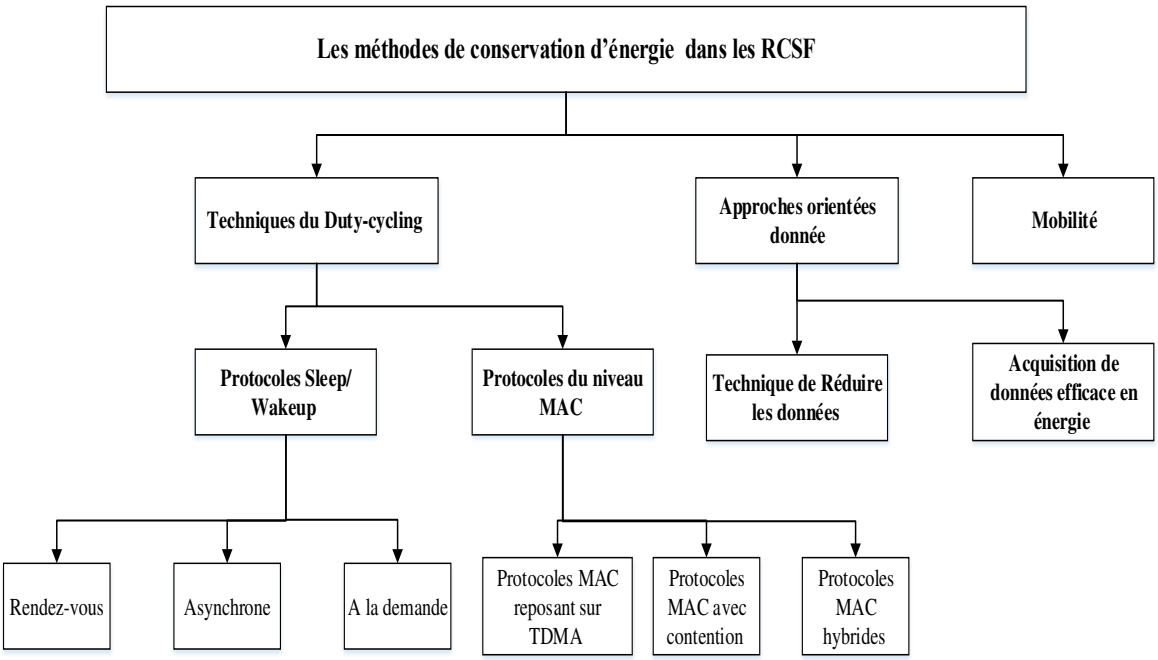

Figure 2.2 : Les méthodes de conservation d'énergie.

2.5.1. Techniques du Duty-cycling

Cette technique est principalement utilisée dans l'activité réseau. Le moyen le plus efficace pour conserver l'énergie est de mettre la radio de l'émetteur en mode veille (low-

power) à chaque fois que 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 nœuds alternent entre périodes actives et sommeil en fonction de l'activité du réseau Ce comportement est généralement dénommé Duty-cycling. Un Duty-cycle est dénommé

Comme étant la fraction de temps où les nœuds sont actifs. Comme les nœuds -capteurs effectuent des tâches en coopération, ils doivent coordonner leurs dates de sommeil et de réveil. Un algorithme d'ordonnancement Sommeil/Réveil accompagne donc tout plan de Duty-cycling. Il s'agit généralement d'un algorithme distribué reposant sur les dates auxquelles des nœuds décident de passer entre l'état actif et l'état sommeil. Il permet aux nœuds voisins d'être actifs en même temps, ce qui rend possible l'échange de paquets, même si les nœuds ont un faible duty-cycle (i.e., ils dorment la plupart du temps).

Le duty cycling peut être réalisé à travers deux approches différentes et complémentaires : le contrôle de topologie et la gestion d'énergie.

2.5.1.1. Protocole de contrôle de topologie

Le contrôle de topologie consiste à trouver le sous-ensemble optimal de nœuds qui assurent la connectivité du réseau. l'idée de base derrière le contrôle de topologie est d'exploiter la redondance du réseau pour prolonger au maximum sa durée de vie. Ceci permet d'augmenter, généralement, la durée de vie par un facteur de deux ou trois par rapport à un réseau dans lequel tous les nœuds sont simultanément actifs [58 ,59]

2.5.1.2. Protocoles de gestion d'énergie

les protocoles de gestion d'énergie se divisent en protocoles sleep/wakeup et d'autres intégrés au niveau MAC Le schéma suivant donne un aperçu global de ces mécanismes :

a. Protocoles Sleep/Wakeup

Comme mentionné précédemment, un régime sleep/wakeup peut être défini pour un composant donné (i.e. le module Radio) du nœud -capteur. On peut relever les principaux plans sleep/wakeup implantés sous forme de protocoles indépendants au-dessus du protocole MAC (i.e. au niveau de la couche réseau ou de la couche application). Dans le document [60], les protocoles sleep/wakeup sont divisés en trois grandes catégories : à la demande, rendez-vous programmés, régimes asynchrones.

❖ Les protocoles à la demande utilisent l'approche la plus intuitive pour la gestion d'énergie. L'idée de base est qu'un nœud devrait se réveiller seulement quand un autre nœud veut communiquer avec lui. Le problème principal associé aux régimes à la demande est de savoir comment informer un nœud en sommeil qu'un autre nœud est disposé à communiquer avec lui. À cet effet, ces systèmes utilisent généralement plusieurs radios avec différents compromis entre énergie et performances (i.e. une radio à faible débit et à faible consommation pour la signalisation, et une radio à 'haut' débit mais à plus forte consommation pour la communication de données). Le protocole STEM (Sparse Topology and Energy Management) [61], par exemple, utilise deux radios ;

❖ Une autre solution consiste à utiliser une approche de rendez-vous programmés. L'idée est que chaque nœud doit se réveiller en même temps que ses voisins. Typiquement, les nœud se réveillent suivant un ordonnancement de réveil et restent actifs pendant un court intervalle de temps pour communiquer avec leurs voisins. Ensuite, ils se rendorment jusqu'au prochain rendez-vous ;

❖ Enfin, un protocole sleep/wakeup asynchrone peut être utilisé. Avec les protocoles asynchrones, un nœud peut se réveiller quand il veut et tant qu'il est capable de communiquer avec ses voisins. Ce but est atteint par des propriétés impliquées dans le régime sleep/wakeup, aucun échange d'informations n'est alors nécessaire entre les nœuds. Quelques régimes sleep/wakeup asynchronnes sont proposés dans [62].

b. Protocoles du niveau MAC

Plusieurs protocoles MAC pour les RCSFs ont été proposés, et de nombreux états de l'art et introductions aux protocoles MAC sont disponibles dans la littérature (par exemple, [63, 64, 65]). Nous nous concentrons principalement sur les questions de gestion d'énergie plutôt que sur les méthodes d'accès au canal. La plupart d'entre eux mettent en ouvre un régime avec un faible duty-cycle pour gérer la consommation d'énergie. Nous avons recensés les protocoles MAC les plus communs en les classant en trois catégories : les protocoles fondés sur TDMA (Time Division Multiplexing Access.), les protocoles utilisant la contention et les protocoles hybrides.

❖ *Protocoles MAC reposant sur TDMA*

Dans les protocoles MAC fondés sur la méthode TDMA [66, 67, 68] le temps est divisé en trames (périodiques) et chaque trame se compose d'un certain nombre de slots de

temps. À chaque nœud est attribué un ou plusieurs slots par trame, selon un certain algorithme d'ordonnancement. Il utilise ces slots pour l'émission/réception de paquets de/vers d'autres nœuds.

Dans de nombreux cas, les nœuds sont regroupés pour former des clusters avec un clusterhead qui est chargé d'attribuer les slots de temps pour les nœuds de son cluster (par exemple, Bluetooth [68], LEACH [67] , et Energy-aware TDMA-based MAC [66]) .

❖ ***Protocoles MAC avec contention***

Les protocoles avec contention sont les plus populaires et représentent la majorité des protocoles MAC proposés pour les RCSFs . Ils assurent le duty-cycle par une intégration étroite des fonctionnalités d'accès au canal avec un régime sleep/wakeup. La seule différence est que, dans ce cas, l'algorithme sleep/wakeup n'est pas un protocole indépendant.

Parmi les protocoles MAC avec contention les plus populaires, nous citons :

- **Accès multiple avec écoute de la porteuse (CSMA : Carrier sense multiple access) ou LBT (listen before talk)**

L'accès multiple avec écoute de la porteuse est introduit en 1975 par *Kleinrock et Tobagi* [69]. Il existe deux versions de CSMA : *non-persistent* CSMA et *p-persistent* CSMA. Dans la première version, un nœud désirant transmettre un message, écoute le canal pour déterminer s'il est occupé. Dans ce cas, le noeud attend un temps aléatoire appelé *backoff* avant de tenter la retransmission. Quand le canal devient libre, le noeud transmet le message immédiatement. Dans *p-persistent* CSMA, quand un noeud écoute et trouve que le canal est occupé, il continue à écouter jusqu'à ce que le canal soit libre. A ce stade, le noeud envoie un message avec une probabilité *p* et retarde la transmission avec une probabilité *1-p*. L'écoute régulière du canal empêche les nœuds capteurs à utiliser CSMA sans modification car l'unité de communication consomme de l'énergie aussi rapidement. De plus, CSMA n'évite pas complètement les collisions maisréalise une utilisation de canal entre 50 et 80% [70].

- **S-MAC (Sensor MAC)** [65] : est un protocole similaire au protocole 802.11. Il utilise la méthode d'accès au médium CSMA/CA RTS/CTS (Request-To-Send, Clear-To-Send) qui permet d'éviter les collisions et le problème du nœud caché. Ce protocole est conçu pour assurer une méthode d'accès économe en énergie pour les RCSFs. Pour ce faire, les nœuds se mettent en mode sommeil pendant une certaine durée et se réveillent pour écouter le

médium pendant une autre durée. Les nœuds échangent leur calendrier de périodes d'écoute en le diffusant a leurs voisins a un saut. Ainsi, chaque nœud connait le calendrier de ses voisins et sait quand il faut se réveiller pour communiquer avec un nœud a sa portée. Plusieurs nœuds peuvent avoir le même intervalle de temps comme période d'écoute. Pour maintenir une synchronisation des horloges, les nœuds émetteurs envoient des messages de synchronisation SYNC au début de la période d'écoute de leurs voisins. S-MAC est particulièrement conçu pour les systèmes d'alerte dans lesquels les applications ont de longues périodes d'inactivité et peuvent tolérer la latence. Donc, son but est de maximiser la durée de vie du réseau en dépit des autres critères de performances.

Le tableaux suivante représente une Comparaison entre les techniques d'accès au médium dans les RCSFs **:**

Tableaux 2.1. Comparaison entre les techniques d'accès au médium dans les RCSFs

	Avantage	**Inconvénient**
TDMA	- économie de consommation d'énergie - l'élection aléatoire de l'émetteur peut entraîner des collisions (pas de collisions)	Complexes dans mise en œuvre . -Problèmes d'adaptation dans le cas de variations temporelles du canal
S-MAC	- l'économie de consommation d'énergie grâce aux séquences périodiques « sleep ». - SMAC est un protocole assez simple à implémenter. - facilite l'adaptation aux changements de topologies (par rapport à TDMA et TRAMA)	Dans le cas où deux nœuds voisins possèdent des séquences d'état (listen/sleep) différentes, cela peut entraîner une grande consommation d'énergie dûe à l'idle listening et à (l'overhearing ?).
CSMA	- elle impose à un émetteur de s'assurer que le canal est libre avant d'émettre	L'utilisation de CSMA oblige les noeuds d'être éveillés pour l'écoute du médium ce qui accroît leurs consommations d'énergie.

❖ ***Protocoles MAC hybrides***

L'idée de base des protocoles MAC hybrides (changement du comportement du protocole entre TDMA et CSMA en fonction du niveau de contention) n'est pas nouvelle. Concernant les RCSFs, Zebra MAC(Z-MAC) [71] est l'un des protocoles les plus intéressants. Afin de définir le schéma principal du contrôle de transmission, Z-MAC commence par une phase préliminaire de configuration. Chaque nœud construit une liste de voisins à deux sauts par le biais du processus de découverte de voisins. Puis, un algorithme distribué d'attribution des slots est appliqué pour faire en sorte que deux nœuds dans un voisinage à deux sauts ne soient pas affectés au même slot. Par conséquent, on est assuré qu'une transmission d'un nœud avec un de ses voisins à un saut n'interfère pas avec les transmissions.

2.5.2. Approches orientées donnée

Généralement, les plans Duty-cycling ne tiennent pas compte des données prélevées par les nouds. Par conséquent, des approches orientées données peuvent être utiles pour améliorer l'efficacité en énergie. En fait, la détection (ou prélèvement de données) affecte la consommation d'énergie de deux manières :

❖ Des échantillons inutiles : les données échantillonnées ont souvent de fortes corrélations spatiales et/ou temporelle [72], il est donc inutile de communiquer les informations redondantes à la SB . Un échantillonnage inutile implique une consommation d'énergie à son tour inutile. En effet, même si le coût de l'échantillonnage est négligeable, cela induit aussi des communications tout le long du chemin qu'emprunte le message.

❖ La consommation électrique du module de détection : réduire la communication ne suffit pas lorsque le capteur est lui-même très consommateur.

Des techniques orientées données sont conçues pour réduire la quantité d'échantillonnage de données en garantissant un niveau de précision acceptable dans la détection pour l'application.

2.5.2.1. Technique de Réduire les données

Réduire les données en terme de volume ou de nombre de paquets, dans le réseaux peut avoir un impact majeur sur la consommation d'énergie due à la communication. Parmis les méthodes de réductions de données, nous trouvons le In-network processing qui consiste à

réaliser de l'agrégation de données (par exemple, calculer la moyenne de certaines valeurs) au niveau des nœuds intermédiaires entre la source et le Sink. Ainsi, la quantité de données est réduite tout en parcourant le réseau vers le Sink. Une agrégation de données appropriée est spécifique à l'application. Un état de l'art sur les techniques In-network processing est présenté dans le document [72].

La compression de données peut être appliquée également pour réduire la quantité d'informations transmises par les nœuds sources. Ce régime implique l'encodage d'informations au niveau des nœuds qui engendrent des données, et le décodage au niveau du Sink. Il existe différentes méthodes de compressions de données citées dans [73].

2.5.2.2. Acquisition de données efficace en énergie

De nombreuses applications émergeantes ont d'applications a de réelles contraintes dues à la détection. Ceci va à l'encontre de l'hypothèse générale selon laquelle la détection n'est pas significative d'un point de vue consommation d'énergie. En fait, la consommation d'énergie du module de détection peut, non seulement être significative, mais encore supérieure à la consommation d'énergie de la radio ou même plus grande que la consommation d'énergie du reste du nœud-capteur [74]. Cela peut être dû à différents facteurs [75].

❖ ***Transducteur gourmand en énergie*** : Certains capteurs ont intrinsèquement besoin d'une forte puissance pour s'acquitter de leur tâche d'échantillonnage. Par exemple, des capteurs d'images CMOS (complementary metal-oxide-semiconductor), voire des capteurs multimédias [76] ont généralement besoin de beaucoup d'énergie. Les capteurs chimiques ou biologiques peuvent aussi être gourmands en énergie.

❖ ***Convertisseurs A/D gourmands*** : des capteurs tels que les transducteurs acoustiques et sismiques [77] nécessitent généralement des convertisseurs A/D à haut débit et à grande résolution. La consommation d'électricité des convertisseurs représente la part la plus importante de la consommation d'énergie du sous-système de détection.

❖ ***Capteurs actifs*** : Une autre classe de capteurs peut obtenir des données du phénomène perçu par l'utilisation de transducteurs actifs (par exemple, sonar, radar ou laser). Dans ce cas, les capteurs doivent envoyer un signal de sondage afin d'obtenir des informations sur la grandeur observée, comme dans [78].

❖ ***Temps d'acquisition long*** : Le temps d'acquisition peut être de l'ordre de plusieurs centaines de millisecondes, voire de quelques secondes. Par conséquent, l'énergie consommée par le sous-système de détection peut être élevé, même si la consommation d'énergie du détecteur reste modérée.

Dans ce cas, réduire les communications peut s'avérer insuffisant, mais les stratégies de conservation d'énergie doivent réellement réduire le nombre d'acquisitions (échantillons de données). Il faudrait également préciser que les techniques d'acquisition de données efficaces en énergie ne visent pas exclusivement à réduire la consommation d'énergie du module de détection. En réduisant les données prélevées par des nœuds sources, elles diminuent aussi le nombre de communications. En fait, beaucoup de techniques d'acquisition de données efficaces en énergie ont été conçues pour réduire au minimum l'énergie consommée par la radio, en supposant que la consommation de la radio est négligeable.

2.5.3. Mobilité

Au delà du duty cycling et des approches orientées données, la mobilité des nœuds, lorsque cela est possible, peut être une solution efficace pour réduire la consommation d'énergie. En effet, dans un réseau de capteurs statiques, les paquets en provenance des nœuds traversent le réseau vers le Sink en suivant des chemins multi-sauts. Lorsque le Sink est lui même statique, quelques chemins peuvent être plus chargés que d'autres, en fonction de la topologie du réseau et les taux de génération de paquets au niveau des nœuds sources.

Généralement, les nœuds proches du Sink sont chargés de transférer un plus grand nombre de paquets et donc verront leur niveau de batteries s'épuiser rapidement. Cela se produira même si des techniques de conservation d'énergie, comme listées ci-dessus, sont mises en place.

Des nœuds collecteurs mobiles, connus sous le nom de Data Mules [79], permettent de réduire la consommation d'énergie. En effet, il s'agit d'appareils mobiles dont nous pouvons contrôler le mouvement pour fournir une alternative à la collecte classique des données à partir des nœuds capteurs dispersés dans l'espace. Les data mules circulent dans la zone de supervision et entrent en communication avec chaque nœuds quand il se trouve à sa proximité. Ainsi les nœuds statiques attendront le passage du data mule pour lui transférer leurs données. Ce dernier se chargera du routage des données vers le Sink. En conséquence,

les nœuds capteurs statiques peuvent économiser l'énergie grâce à la réduction de la longueur des chemins de routage, de la contention et des overheads de diffusion.

2.6. Comparaison des techniques de conservation énergétique [80]

le tableau 2.2 , résume les caractéristiques des quelque approches de conservation d'énergie les comparant en termes d'avantages et d'inconvénients.

Tableau 2.2 *:* Comparaison entre quelques approches de conservation d'énergie dans les RCSFs.

Caractéristiques / **Technique de conservation énergétique**	**Approche utilisés**	**Avenages majeurs**	**Inconvénients majeurs**
Sleep –Week-up	Alternance entre périodes actives et inactives pour conserver l'énergie	-Efficace en termes de consommation énergétique	-Nécessitent des mécanismes de synchronisations des noeuds lors des phases de TX/RX -Collisions fréquentes dans le réseau
Protocoles MAC hybrides	Techniques combinant CSMA et TDMA	-Efficaces en termes de consommation énergétique -Permet de réduire les interférences inter-noeuds	- Plus complexes à mettre en œuvre
Techniques de réduction de données	Techniques d'agrégation et de compression des données	-Efficaces en termes de consommation énergétique.	« Overhead » plus important

		-Adaptées dans le cas de topologies avec redondances de couvertures	

Nous présentons dans ce qui suit les différentes techniques utilisées pour minimiser la consommation d'énergie dans les RCSFs . Ces techniques sont appliquées soit au niveau de la couche liaison soit au niveau de la couche réseau.

Le schéma suivant donne un aperçu global de ces mécanismes : Les techniques de conservation d'énergie dans un capteur visent à économiser l'énergie sur plusieurs niveaux : capture, traitement ou communication. [81]

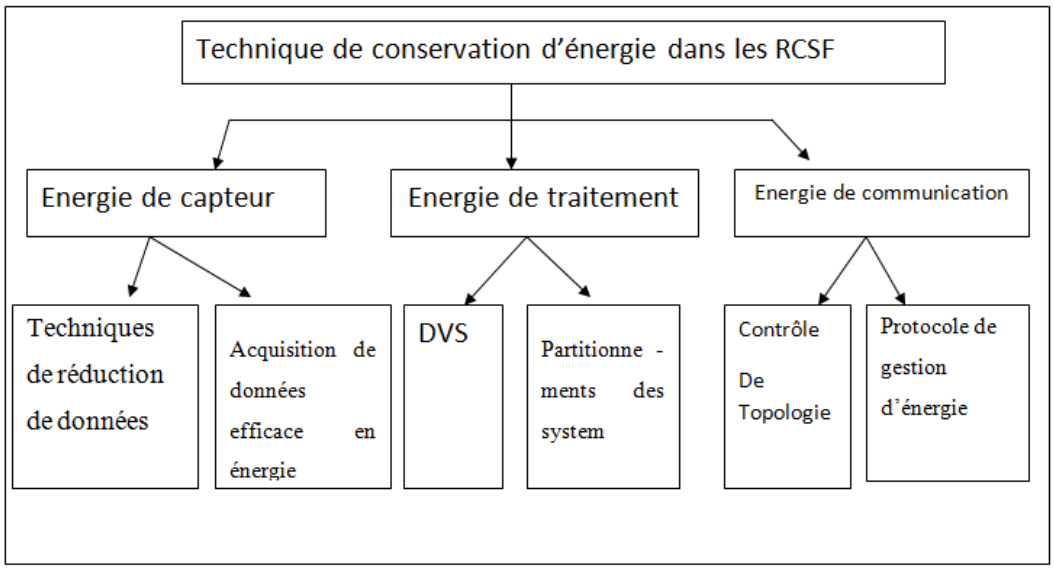

Figure 2.3 : Les techniques utilisées pour minimiser la consommation d'énergie [81]

L'énergie du capteur peut être économisée soit au niveau de la capture, au niveau de traitement ou au niveau de la communication.

a. La seule solution apportée pour la minimisation de la consommation d'énergie au niveau de la capture consiste à réduire les fréquences et les durées de captures.

b. L'énergie de traitement (calcul) peut être optimisée en utilisant deux techniques :

- L'approche DVS (Dynamique Voltage Scaling) [82] qui consiste à ajuster de manière adaptative la tension d'alimentation et la fréquence du microprocesseur pour économiser la puissance de calcul sans dégradation des performances.

- L'approche de partitionnement de système qui consiste à transférer un calcul prohibitif en temps de calcul vers une SB qui n'a pas de contraintes énergétiques et qui possède une grande capacité de calcul [83].

c. La minimisation de la consommation d'énergie pendant la communication est étroitement liée aux protocoles développés pour la couche réseau et la sous-couche MAC. Ces protocoles se basent sur plusieurs techniques : l'agrégation de données, la négociation et à la technique CSIP (Collaborative Signal and Information Processing). Cette dernière technique est une discipline qui combine plusieurs domaines [84] : la communication et le calcul à basse puissance, le traitement de signal, les algorithmes distribués, la tolérance aux fautes, les systèmes adaptatifs et la théorie de fusion des capteurs et des décisions. Ces techniques ont le but de réduire le nombre d'émission/ réception des messages.

2.7. Conclusion

Durée de vie est sans doute la métrique la plus importante dans l'évaluation des performances d'un réseau de capteurs. En effet, dans un environnement contraint, toute ressource limitée doit être prise en compte. Toutefois, la durée de vie du réseau, comme mesure de la consommation d'énergie, occupe une place exceptionnelle puisqu'elle constitue la borne supérieure de l'utilité de ce réseau [45].

Dans ce chapitre nous avons présenté les techniques les plus utilisées dans le but de prolonger la durée de vie d'un RCSF.

Chapitre 3

Le routage dans les RCSFs

Sommaire

3.1. Introduction

3.2. Définition de routage

3.3. Classification des protocoles de routage

3.4. Exemples des protocoles de routage

3.5. Les protocoles hiérarchiques

3.6. Conclusion

3.1. Introduction

En raison de la capacité limitée des chemins multi-sauts [85] et de la forte dynamique des liaisons sans fil, l'approche de routage simple à chemin unique ne parvient pas à assurer de manière efficace la transmission de données à haut débit dans les RCSFs. De nos jours, l'approche de routage multi-chemins est largement utilisée en tant que l'une des solutions possibles pour faire face à cette limitation et une panoplie de protocoles de routage multi-chemins a été élaborée**.**

Dans ce chapitre, nous présentons une aperçue globale sur le routage dans les RCSFs, nous commençons par une définition de routage et de protocole de routage, puis les classifications des protocoles de routage, Enfin nous intéressons aux différentes approches proposées pour le routage hiérarchique de données sur les réseaux RCSFs.

3.2. Définition de routage

Le routage est un processus qui permet de sélectionner des chemins dans un réseau pour transmettre des données depuis un expéditeur jusqu'à un ou plusieurs destinataires. On parle de routage dans différentes domaines : réseaux téléphoniques, réseaux électroniques (comme Internet), réseaux de transports… [86]

L'objectif principal d'un protocole de routage pour un réseau de capteurs sans fil est l'établissement correct et efficace d'itinéraires entre une paire de nœuds afin que des messages puissent être acheminés. Le protocole de routage permet aux nœuds de se connecter directement les uns aux autres pour relayer les messages par des sauts multiples et de transmettre les données vers un point de collecte.

3.3. Classification des protocoles de routage

Les protocoles publiés dans le domaine des RCSFs ont été largement étudiés et peuvent être classifiés suivant un certain nombre de critères. Dans cette section, une classification des protocoles de routage selon cinq critères est détaillée. La figure **3.1** représente la classification des protocoles de routage selon les critères suivantes: la topologie du réseau, le fonctionnement du protocole, le paradigme de communication, l'établissement de la route, l'initiateur de communication.

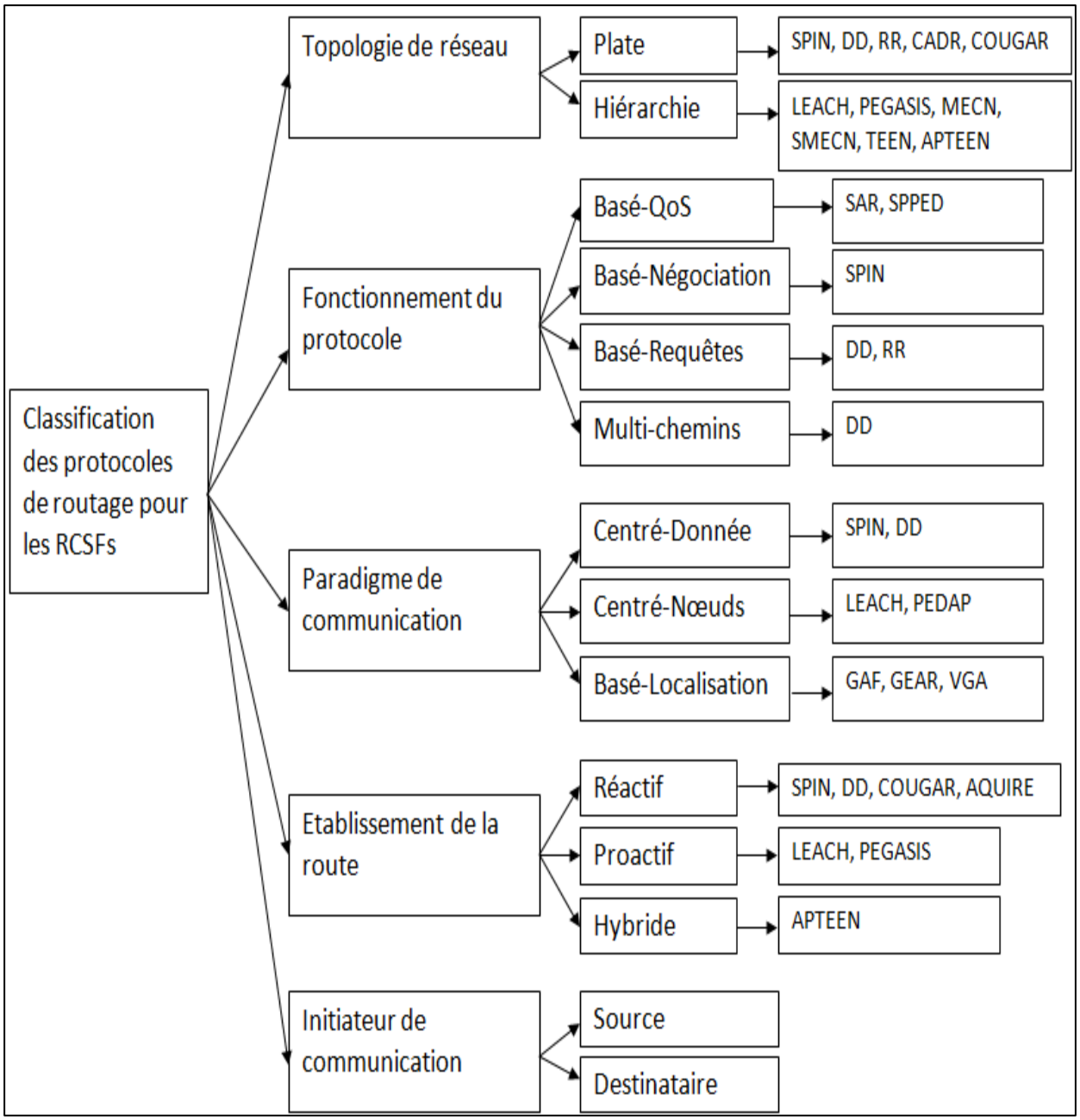

Figure 3.1: Classification des protocoles de routage

3.3.1. Classification selon la topologie du réseau

Les protocoles de routage basés sur la topologie du réseau peuvent être classifiés en deux catégories : protocoles de routage à plat**,** protocoles hiérarchiques**.**

3.3.1.1. Les protocoles de routage à plat

Ces protocoles considèrent que tous les nœuds sont identiques, c'est à dire ont les mêmes fonctions à exécuter sauf les nœuds de contrôle (*sink*) qui est chargé de collecter toutes les informations issues des différents nœuds capteurs pour les transmettre vers

l'utilisateur final. La décision d'un nœud de router des paquets vers un autre dépendra de sa position et pourra être remise en cause au cours du temps [87].

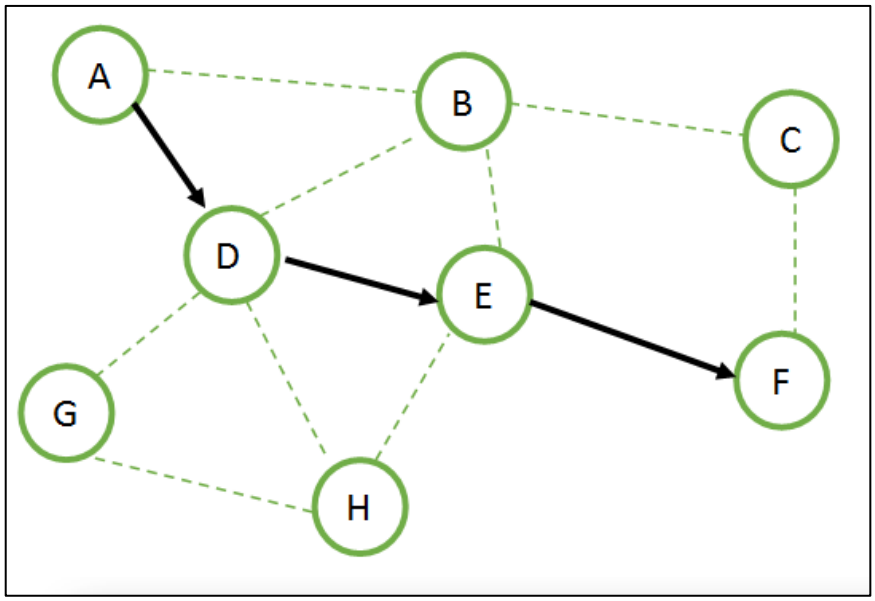

Figure 3.2 : Routage plat

3.3.1.2. Les protocoles hiérarchiques

L'objectif principal du routage hiérarchique [88, 98] est de maintenir efficacement la consommation d'énergie de nœuds de capteurs en les impliquant dans la communication multi-hop 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.

3.3.2. Classification selon le fonctionnement du protocole

Le mode de fonctionnement définit la manière avec laquelle les données sont propagées dans le réseau. Selon ce critère, les protocoles de routage peuvent être classifiés en quatre catégories : routage basé sur la Qualité de Service ("QoS" based routing), routage basé sur les requêtes (query-based routing), routage multi chemins (Multi-path routing), et routage basé sur la négociation (Negociation based routing) [89].

3.3.2.1. Routage basé sur la QoS

Dans les protocoles de routage basés sur la QoS, le réseau doit s'équilibrer entre la consommation d'énergie et la qualité de données. En particulier, le réseau doit satisfaire une certaine métrique de QoS, par exemple : largeur de la bande passante [86].

3.3.2.2. Routage basé sur la négociation

Dans les protocoles de routage basé sur la qualité de service, le réseau doit équilibrer entre la consommation d'énergie et la qualité de données. En particulier, le réseau doit satisfaire certaines métriques de QoS, par exemple, retard, énergie, largeur de bande passante, etc.

Les protocoles de cette approche sont très recommandés pour les applications de surveillance (centrales nucléaires, applications militaires, etc.) [90].

3.3.2.3. Routage basé sur les requêtes

Dans ce type de routage, le nœud puits (Sink) génère des requêtes afin d'interroger les nœuds à travers le réseau, les protocoles appartenant au routage basé sur les requêtes fonctionnent sur le principe de l'envoie et de la réception des requêtes pour les données.

De ce fait, les nœuds ayant les données correspondantes aux requêtes reçues doivent les envoyer au nœud initiateur de la requête à travers le chemin inverse de la requête. Les requêtes émises par le nœud puits peuvent aussi être ciblées sur des régions spécifiques de réseau [91].

3.3.2.4. Routage basé sur les multi-chemins

Dans cette catégorie, les protocoles de routage utilisent des chemins multiples plutôt qu'un chemin simple afin d'augmenter la performance du réseau. La fiabilité d'un protocole peut être mesurée par sa capacité à trouver des chemins alternatifs entre la source et la destination en cas de défaillance du chemin primaire. Pour cette raison, certains protocoles construisent plusieurs chemins indépendants, c.-à -d : ils ne partagent qu'un nombre réduit (voir nul) de nœuds. Malgré leur grande tolérance aux pannes, ces protocoles requièrent plus de ressources énergétiques et plus de messages de contrôle. [86]

3.3.3. Classification selon le paradigme de communication

Le paradigme de communication détermine la manière dont les nœuds sont interrogés. Dans les RCSFs, il existe trois paradigmes de communication : centré- nœuds, centré-données et basé sur la localisation [92].

3.3.3.1. Centré-nœuds (Node-centric)

Ce paradigme est celui employé dans les réseaux conventionnels, ou il est nécessaire de connaitre et d'identifier les nœuds communicants (comme l'adresse IP). Les réseaux ad hoc utilisent ce genre de paradigme, qui s'intègre bien avec l'utilisation de ce type d'environnement. Cependant, pour les réseaux de capteurs, un routage basé sur une identification individuelle des nœuds ne reflète pas l'usage réel de réseau. Pour cela, un autre paradigme a été introduit : data centric. Néanmoins, le paradigme node centric n'est pas à écarter totalement, car certaines applications nécessitent une interrogation individuelle des capteurs.

3.3.3.2. Centré-données (Data-centric)

Ce paradigme suppose qu'il est difficile d'avoir des identifiants comme les adresses MAC ou IP pour pouvoir communiquer entre les nœuds capteurs. Ainsi, le routage ne se fait pas en fonction d'une adresse de destination, mais suivant les données disponibles au niveau des capteurs. Ces données seront propagées de proche en proche pour arriver aux nœuds puits.

3.3.3.3. Basé-localisation (location-based)

Dans cette approche, les décisions de routage sont établies selon la position des nœuds. La distance entre les nœuds voisins peut être estimée sur la base de la puissance du signal arrivé. Un tel type de routage nécessite que les nœuds aient connaissance de leurs positions géographiques. Par conséquent, ce type de mécanismes nécessite un déploiement d'une solution de positionnement, dont le degré de précision requis dépend de l'application ciblée. L'utilisation du GPS reste trop couteuse pour un RCSF. Néanmoins, d'autres méthodes de localisation et de positionnement des capteurs ont été développées comme : la triangulation.

3.3.4. Classification selon l'établissement de la route

Les protocoles de routage basés sur l'établissement de la route peuvent être classifiés en protocoles : proactifs, réactifs, hybrides.

3.3.4.1. Protocoles proactifs

Le calcul de routes se fait à priori ce qui facilite l'acheminement des données. Les informations des chemins à suivre par chaque donnée source vers une destination sur le réseau sont stockées dans une table de routage. Les tables de routage doivent être mises à jour régulièrement afin de corriger certains chemins coupés en raison du changement de topologie

dus aux défaillances ou à la mobilité de certains nœuds capteurs. Cette mise à jour est assurée par la diffusion périodique des paquets de contrôle sur le réseau, ce qui n'est pas évident pour des réseaux de grande taille comme les réseaux de capteurs sans fil. L'établissement de routes se fait indépendamment des besoins réels de l'application et un bon nombre de ces routes est sauvegardé pour ne jamais être utilisées. Une autre limite concerne la taille des tables de routage, notamment pour des réseaux de grande taille, qui pourrait dépasser les capacités de stockage des nœuds capteurs [93].

3.3.4.2. Protocoles réactifs

Egalement appelé routage à la demande, le routage réactif permet de créer les routes selon les besoins de l'application. Lorsqu'une requête est diffusée sur le réseau, la procédure de découverte de routes est lancée par le nœud concerné par cette requête, et les réponses sont acheminées sur les routes créées. Cette procédure est lancée également pour des applications event-driven (applications orientées événements) pour chaque événement intéressant détecté. L'avantage d'établir des routes à la demande est la conservation d'énergie par rapport au routage proactif. La recherche de routes peut causer des lenteurs pour l'acheminement des données ce qui n'est pas approprié aux applications interactives et temps-réel [94].

3.3.4.3. Protocoles hybrides

C'est une combinaison des deux concepts de routage proactif et réactif. Des tables de routage sont stockées sur les nœuds capteurs de façon à établir des routes sur leur voisinage proche (généralement en deux sauts maximums). Au-delà de leur voisinage, le routage devient réactif et des procédures de recherche de routes sont lancées. Cette approche combine les avantages des deux autres approches proactive et réactive et réduit considérablement la taille des tables de routage ainsi que les délais d'établissement de routes [95].

3.3.5. Classification selon l'initiateur de communication

La communication dans un réseau de capteurs peut être lancée par les nœuds sources ou par les nœuds destinataires.

3.3.5.1. Communication lancée par la source

Dans un protocole ou la communication est initiée par un nœud source : les nœuds envoient des données à la station de base quand ils détectent une variation sensible des paramètres à surveiller. Ces protocoles utilisent des modèles de livraison de donnée dirigés par les événements ou dirigés par le temps (périodiques). Soit la donnée est envoyée à intervalle de temps régulier ou alors elle est envoyée quand les nœuds capturent une certaine valeur (détection d'un événement) [96].

- **Avantages**

✓ L'´etablissement de communication dans le réseau évite le problème d'overhead.

✓ L'efficacité énergétique due `a l'absence des requêtes qui consomment beaucoup d'énergie générées.

- **Inconvénients :**

✓ Les capteurs doivent avoir des informations sur les chemins qui conduisent au nœud puits.

✓ Pour cette approche, l'´energie n'est pas la seule préoccupation, des exigences de QoS doivent en général être respectées.

3.3.5.2. Communication lancée par la destination

Les protocoles ou la communication est initiée par les destinations, utilisent un modèle de livraison de donnée basé sur les requêtes. Les nœuds sources répondent aux requêtes envoyées par la station de base. Il y a un surcout dans ce type de protocoles, puisque les requêtes sont d'abord diffusées dans tout le réseau. Donc, à chaque fois qu'il y a une requête, il y a un flooding dans tout le réseau (inondation de tout le réseau) [96].

- **Avantage**

✓ L'envoie des requêtes décrivent les données requises par la station de base élimine les transmissions inutiles.

- **Inconvénients**

✓ Les délais entrainés par l'établissement des routes.

✓ La circulation de messages requêtes de grande taille tend à épuiser les batteries des capteurs.

3.4. Exemples des protocoles de routage

Dans ce qui suit, un certain nombre d'exemples de protocoles de routage pour les RCSFs est présenté.

3.4.1. Low Energy Adaptive Clustering Hierarchy (LEACH)

LEACH est un protocole basé sur les groupes, dans lequel les chefs de groupes élus collectent les données à partir de tous les nœuds capteurs appartenant à leur groupe, agrègent les données rassemblées par des procédures de fusion, et transmettent ces données directement à la station de base. Les chefs de groupes élus demeurent ainsi pour une période de temps appelée « *round* ». Au début de chaque *round*, chaque nœud détermine la possibilité d'être un chef de group pendant la période en cour, s'il décide de l'être, il annonce sa décision à tous ses nœuds voisins. Les autres nœuds qui décident de ne pas être un chef de groupe, se joignent à l'un des chefs élus après recevoir leurs décisions, la sélection du chef de groupe adéquat se base sur plusieurs paramètres prédéterminés tel que le rapport signal/bruit (SNR).

3.4.2. Power-Efficient GAthering in Sensor Information Systems (PEGASIS) [88]

PEGASIS est une version améliorée du protocole LEACH. PEGASIS forme des chaînes plutôt que des clusters de nœuds de capteurs afin que chaque nœud transmette et reçoive uniquement des données d'un voisin. Un seul nœud est sélectionné à partir de cette chaîne pour transmettre à la station de base. L'idée de PEGASIS est qu'il utilise tous les nœuds pour transmettre ou recevoir des données avec ses plus proches voisins. Il déplace les données reçues de nœud à nœud, puis les données seront agrégées jusqu'à ce qu'elles atteignent tous la station de base. Donc, chaque nœud du réseau est tour à tour un chef de file de la chaîne, ainsi que responsable pour transmettre l'ensemble des données recueillies et fusionnées par la chaîne de nœuds au niveau de la station de base [97].

3.4.3. TEEN et APTEEN

TEEN (Threshold sensitive Energy Efficient sensor Network protocol) est un protocole de routage basé sur les groupes et proposé par Manjeshwar et al. [123]. Ce protocole est similaire à LEACH décrit dans la section précédente, sauf que les nœuds ne sont pas supposés avoir un taux fixe de données à transmettre.

TEEN utilise la même stratégie que LEACH pour l'étape de formation des groupes, mais adopte une approche différente pour la phase de transmission des données. Durant cette étape,

TEEN utilise deux paramètres définis par l'utilisateur appelé *hard threshold* (*ht*) et *soft threshold* (*st*) et ceci, pour pouvoir déterminer le besoin de transmission de la donnée captée vers la station de base. Si la valeur captée excède *ht* pour la première fois, elle est stockée dans une variable et transmise durant le temps alloué par le chef du groupe au nœud concerné.

Si la valeur captée dépasse, par la suite, la valeur stockée par une magnitude st, le nœud décide de la transmettre et stocke cette nouvelle valeur dans son cache pour les comparaisons ultérieures.

APTEEN (Adaptive Threshold sensitive Energy Efficient sensor Network protocol) est une extension de TEEN qui fait à la fois la collection des captures périodique de données et qui réagit aux événements critiques. Quand la SB forme des clusters, les clusters head diffusent les attributs, les valeurs des seuils, ainsi que le calendrier de transmission à tous les nœuds. Le CH effectue également l'agrégation de données afin d'économiser l'énergie. [88]

3.4.4. Minimum Energy Communication Network (MECN) [98]

MECN est un protocole de routage qui cherche à établir et à entretenir une énergie minimale pour les réseaux 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 nœud. La région de relais se compose de nœuds dans une zone périphérique où la transmission à travers ces nœuds est plus économe en énergie que la transmission directe. L'idée principale de MECN est de trouver un sous-réseau qui a moins de nœuds et qui nécessite moins d'énergie pour la transmission entre deux nœuds quelconques. Cela est effectué en utilisant une recherche localisée pour chaque nœud en prenant en considération sa région de relais [88].

3.4.5. Sensor Protocol for information via negociation (SPIN)

Cette famille de protocoles adaptatifs [99] est proposée pour pallier au problème d'ignorance de ressources confronté dans la technique d'inondation en utilisant la négociation et l'adaptation aux ressources disponibles.

La conception des protocoles SPIN est basée sur deux idées :

- Si les nœuds capteurs se limitent, dans leurs rapports, sur l'envoi des données qui décrivent les informations captées et non pas l'envoi de ces dernières en entier (telle que les images), ils peuvent opérer avec plus d'efficacité et économiser ainsi une quantité considérable d'énergie.
- les nœuds capteurs doivent contrôler en permanence les changements dans leurs ressources énergétiques.

3.4.6. Geographic Adaptive Fidelity (GAF) [100]

GAF est un protocole de routage basé sur la localisation des nœuds. Il est conçu principalement pour les réseaux mobiles ad hoc, mais peut être applicable aux réseaux de capteurs. La localisation des nœuds dans GAF pourrait être fournie à l'aide d'un GPS ou d'autres techniques de localisation [88, 111, 102]. Il consiste à former des grilles virtuelles de la zone concernée en partitionnant cette zone où les nœuds sont déployés en de petites zones telles que, pour deux grilles adjacentes Gx et Gy, tous les nœuds de Gx peuvent communiquer avec tous les nœuds Gy. Ainsi, ce système de partitionnement GAF assure la fidélité du routage car il existe au moins un chemin entre un nœud et la SB.

GAF peut augmenter considérablement la durée de vie du réseau. En effet, un seul nœud dans chaque grille reste à l'état actif en faisant passer les autres nœuds de la grille à l'état de sommeil pour une certaine période de temps tout en assurant la fidélité du routage. Cependant, dans certains environnements où les nœuds sont fortement mobiles, la fidélité du routage pourrait être réduite si un nœud actif quitte la grille. Ainsi, le nombre de données perdues sera important.

3.4.7. La diffusion dirigée

La diffusion dirigée [103, 104] est un protocole important dans le routage data-centric des réseaux de capteurs. L'idée vise à diffuser des données aux nœuds en utilisant un schéma de nommage pour les données. La raison principale derrière l'utilisation d'un tel système est de se débarrasser des opérations inutiles de routage de couche réseau afin d'économiser l'énergie.

La diffusion dirigée suggère l'utilisation de paires attribut-valeur pour les données et les requêtes des capteurs. Afin de créer une requête, un nœud est défini à l'aide d'une liste de paires attribut-valeur comme le nom des objets, l'intervalle, la durée, la zone géographique, etc. Un paquet est diffusé par ce nœud vers la destination à travers ses voisins. Chaque nœud qui reçoit les paquets peut les stocker pour une utilisation ultérieure. Les paquets stockés sont ensuite utilisés pour comparer les données reçues. La requête contient aussi plusieurs champs de gradient. Un gradient est un lien réponse avec un voisin dont le paquet a été reçu et qui est caractérisé par le débit, la durée et la date d'expiration de données. Ainsi, en utilisant les intérêts et les gradients, les routes sont établies entre la destination et les sources. Plusieurs routes peuvent être établies de telle sorte que l'une d'elle est choisie par renforcement. La

destination renvoie le message d'intérêt initial à travers la route choisie. Un intervalle plus petit renforce donc le nœud source sur ce chemin pour envoyer des données plus fréquemment [88].

3.5. Les protocoles hiérarchiques

Le routage hiérarchique est considéré comme étant l'approche la plus favorable en termes d'efficacité énergétique et d'auto-organisation. Il se base sur le concept (nœud standard – nœud maître), où les données de tous les nœuds standard sont envoyées à leur maitre, qui les achemine vers la SB via d'autres nœuds maitres après l'application de techniques de *l'agrégation et de la fusion*, afin de diminuer le nombre de messages transmis au *SINK*, ce qui implique une meilleure économie d'énergie [105, 107, 108].

En fait, suivant la topologie du réseau, deux grandes approches sont dérivées de ce type de protocoles à savoir : une approche basée sur le cluster et une approche basée sur la chaine.

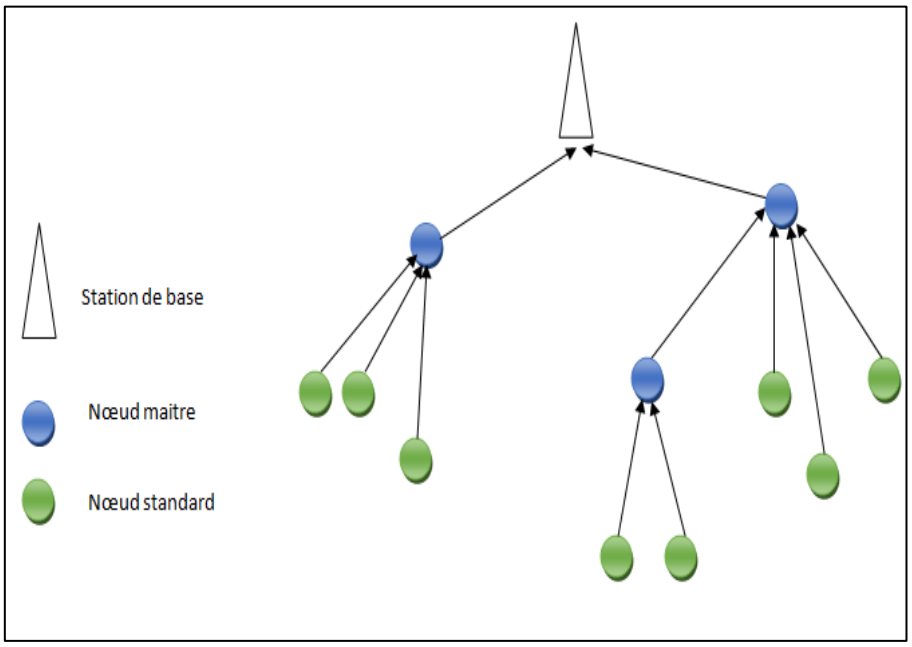

Figure 3.3 : Le routage hiérarchique

3.5.1. L'approche basée sur le cluster

Le réseau est partitionné en clusters, où dans chacun d'entre eux, un seul capteur est sélectionné comme leader (CH) pour jouer le rôle spécial de nœud maitre. La formation des clusters et la sélection de ses CHs, sont les deux préoccupations importantes des protocoles de routage hiérarchiques. D'ailleurs, chaque CH crée un plan de transmission pour les capteurs dans son cluster, ce qui permet aux antennes radio de chaque nœud non-CH d'être éteintes

toutes les fois, excepté pendant son temps de transmission. Les nœuds dont l'énergie est la plus élevée peuvent être employés dans le traitement et l'envoie d'informations tandis que ceux de basse énergie peuvent être employés dans la collection de données [109, 110].

La rotation des CHs s'avère également un facteur important pour l'organisation des RCSFs. Puisque la SB est généralement loin du champ de surveillance, les CHs diffusent une quantité plus importante d'énergie pour la transmission de données à la SB. Par conséquent, les CHs mourront rapidement si le même nœud fonctionne continuellement comme un CH. Ainsi, pour ne pas épuiser la puissance de batterie d'un capteur simple, les algorithmes de groupement (clustering) emploient une rotation de rôle des CHs entre les nœuds capteurs.

3.5.1.1. LEACH

LEACH est un protocole basé sur les groupes, dans le quel les chefs de groupes élus collectent les données à partir de tous les nœuds capteurs appartenant à leur groupe, agrègent les données rassemblées par des procédures de fusion, et transmettent ces données directement à la SB. Les chefs de groupes élus demeurent ainsi pour une période de temps appelée « *round* ». Au début de chaque *round*, chaque nœud détermine la possibilité d'être un chef de groupe pendant la période en cour, s'il décide de l'être, il annonce sa décision à tous ses nœuds voisins. Les autres nœuds qui décident de ne pas être un chef de groupe, se joignent à l'un des chefs élus après recevoir leurs décisions, la sélection du chef de groupe adéquat se base sur plusieurs paramètres prédéterminés tel que le rapport signal/bruit (SNR).

Les caractéristiques principales de l'algorithme LEACH se résument en :

- La coordination locale pour l'établissement des groupes et leur fonctionnement.
- La rotation aléatoire des CHs.
- La fusion locale des données pour réduire le taux de communications avec la SB.
- Ce protocole est organisé en périodes où chacune commence par une phase d'initialisation suivie par une autre phase d'une durée plus longue pour le transfert des données captées.

Dans ce qui suit, nous décrirons les sous phases impliquées par celles citées auparavant.

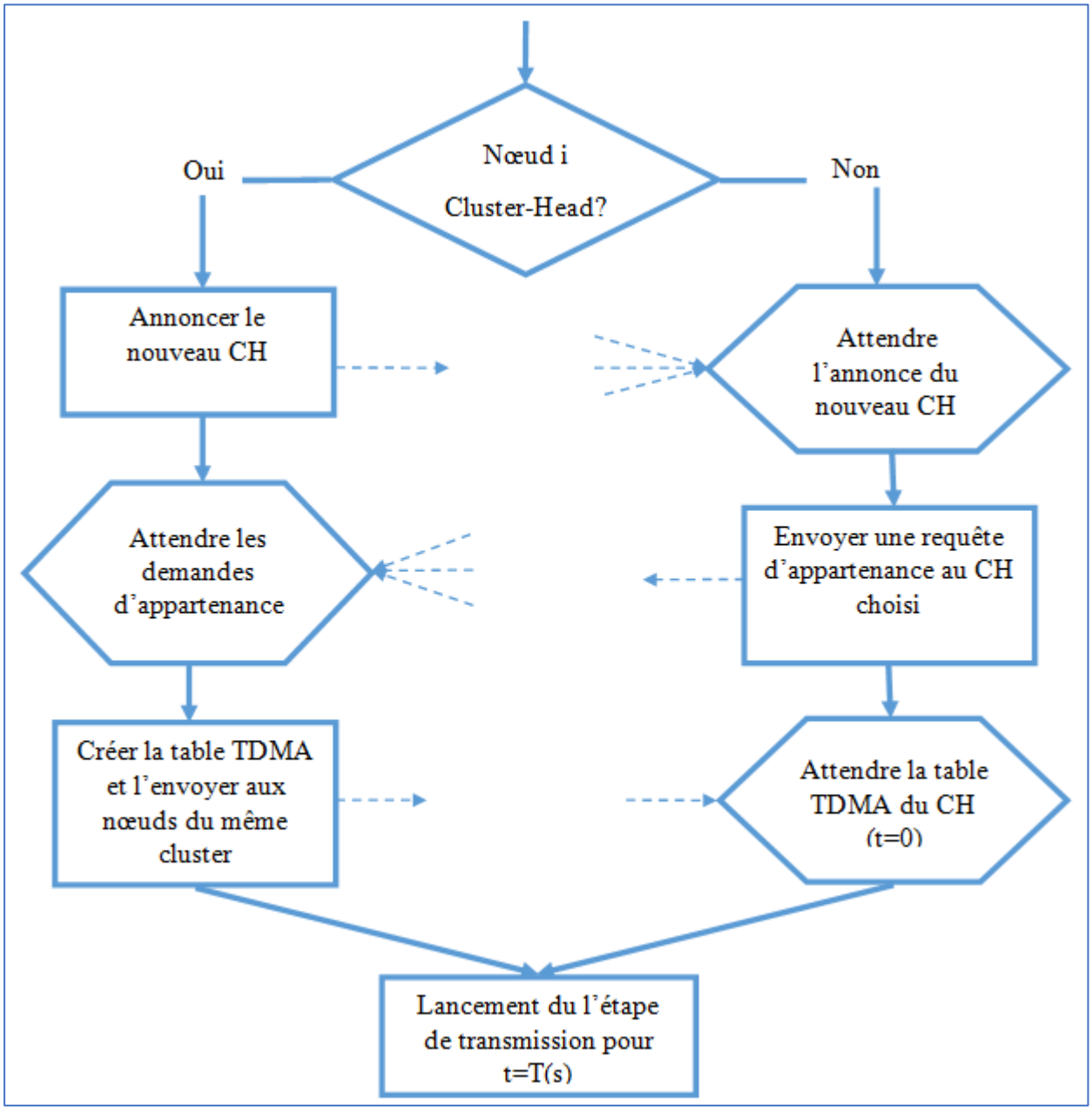

Figure 3.4 : Organigramme de l'algorithme de formation de clusters distribué pour LEACH.

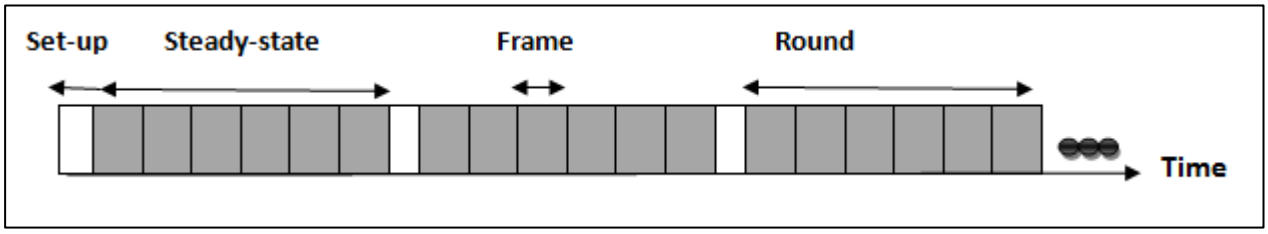

Figure 3.5 : Chronologie montrant le fonctionnement de LEACH.

Des clusters adaptive sont formés pendant la phase set up et les transferts des données qui se produisent pendant la phase set up.

- **Phase d'annonces**

Durant cette phase, une fraction prédéterminée de nœuds capteurs, soit *p*, s'élit comme étant chef de groupe. La décision d'être chef de group se fait par le choix d'un nombre aléatoire entre 0 et 1. Si le nombre généré est inférieur a un seuil T(*n*), le nœud est élu chef de groupe durant la période «*round*» courante. Le seuil T(*n*) est donné par l'expression suivante :

$$T(n) = \begin{cases} \dfrac{p}{1 - p[r mod(\frac{1}{p})]} & Si\ n \in G \\ 0 & Sinon \end{cases} \qquad \textit{(Equation 3.1)}$$

Tel que *r* est le numéro de la période courante, et G l'ensemble des nœuds n'ayant pas été sélectionnés chef de groupe durant les *1/p* dernières périodes.

Une fois la sélection des chefs de groupes et faite, ces derniers annoncent leurs décisions jointes avec leurs identificateurs à tous les nœuds du réseau en utilisant le protocole MAC appelé CSMA [111].

- **phase d'établissement de l'ordonnancement**

Chaque chef de groupe, basé sur l'approche TDMA, établit et diffuse un ordonnancement affectant à chaque membre du groupe un temps durant lequel il peut communiquer les données captés au chef [112], [113].

- **Transmission de données**

Une fois l'ordonnancement est connu par tous les nœuds du même groupe, chacun d'eux utilise le temps qui lui a été consacré pour transmettre au chef les données captées. Le chef de groupe exécute, par la suite, des algorithmes de fusion de données pour agréger les données reçues et minimiser ainsi la quantité d'information transmise à la station de base [114].

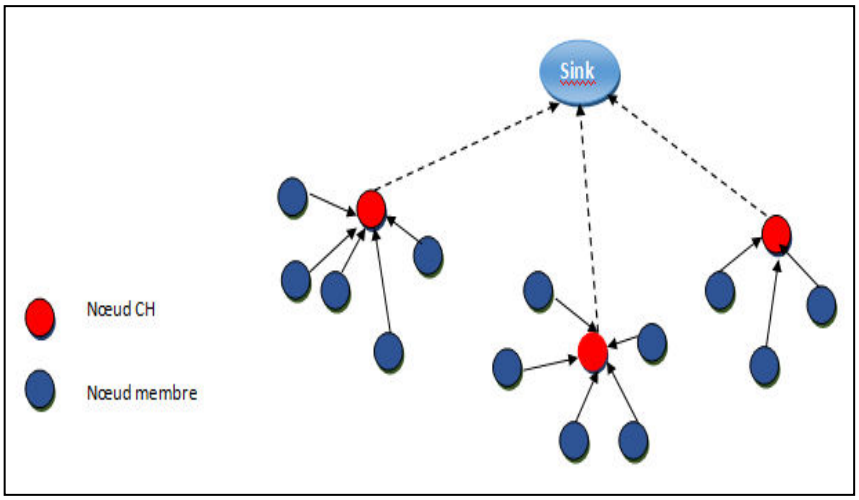

Figure 3.6 : Algorithme de routage LEACH

- **Avantages [54]**

✓ La charge d'énergie est répartir sur l'ensemble des nœuds.

✓ L'agrégation des données permet de réduire le taux de communications avec la *SB*.

✓ L'auto-organisation des clusters est possible.

- **Inconvénients [54]**

✓ LEACH ne résout pas le gaspillage de l'énergie entre les CHs et la SB.

✓ LEACH ne garantit pas l'obtention des clusters optimaux de même taille et qui sont distribués uniformément (augmentation de délai et de taux de perte).

✓ Les communications inutiles telles que la transmission des données inchangées et des messages *CH-ADV* aux nœuds distants.

✓ LEACH ne convient pas aux réseaux déployés dans de vastes régions.

✓ LEACH ne pose aucune contrainte sur la distribution et le niveau d'énergie des CHs.

✓ La rotation du rôle du *CH* génère une surconsommation d'énergie.

3.5.1.2.Low Energy Adaptive Clustering Hierarchy Centralized (LEACH-C)

LEACH-C est une variante de LEACH, a été conçue pour répondre au problème de sélection aléatoire du CH dans LEACH. Elle a été proposée par les mêmes auteurs de LEACH dans [54]. Cette variante utilise une architecteur centralisée pour choisir les CHs, la phase d'état stable de LEACH-C est comme suite :

✓ Lors de la phase d'installation de LEACH-C, les nœuds informent la SB par l'envoi d'information sur leur position par GPS et leur niveau d'énergie.

✓ La station de base désigne les CHs en se basent sur la moyenne des niveaux d'énergie, les nœuds ayant un niveau d'énergie inférieur à la moyenne ne peuvent pas être des CHs dans l'itération courante.

✓ Enfin, la station de base diffuse un message contenant les identificateurs des CHs .

✓ Le fonctionnement des autres opérations est similaire à celui de LEACH.

- **Avantage**

✓ LEACH-C permet une diminution remarquable de la consommation énergétique.

- **Inconvénient**

✓ La version centralisée n'est pas adaptée aux réseaux de grande dimension.

3.5.1.3.Fixed number of cluster Low Energy Adaptive Clustering Hierarchy (LEACH-F)

Le protocole LEACH-F a été proposé pour minimiser le cout de l'overhead lors de la formation des clusters. Ainsi, dans LEACH-F, les clusters sont formés une seule fois et le rôle de cluster head tourne parmi les nœuds membres du même cluster [116].

- **Avantages**

✓ Elimination des frais généraux de la re-clustering dans LEACH de base.

✓ Une fois que le nombre fixe de cluster est formé; ils sont maintenus tout au long du réseau.

- **Inconvénients**

✓ Ce protocole ne fournit aucune possibilité d'ajouter ou de retirer les nœuds une fois les clusters sont formés et les nœuds ne peuvent pas ajuster leur comportement sur le nœud mort.

3.5.1.4. Two level Low Energy Adaptive Clustering Hierarchy (TL-LEACH)

Contrairement au protocole LEACH ou les CHs envoient des données à la station de base directement en un seul saut, le protocole TL-LEACH fonctionne dans la hiérarchie à deux niveaux. Les données agrégées de chaque CH sont recueillies par un CH qui se trouve entre les CHs et la station de base, au lieu d'envoyer directement à la station de base. [117]

- **Avantages**
  - ✓ Ce protocole réduit l'énergie de transmission de données.
  - ✓ Les CHs meurent tôt par rapport à d'autres nœuds, loin de la station de base et TL-LEACH améliore l'efficacité énergétique en utilisant un CH comme nœud de relais entre les CHs de cluster.
- **Inconvénients**

- ✓ Frais généraux supplémentaires pour l'élection des CHs secondaires et la formation des clusters.

3.5.1.5. Improved Low Energy Adaptive Clustering Hierarchy (I-LEACH)

Naveen Kumar et Jasbir Kaur [113], ont proposés I-LEACH qui a surperformé le protocole LEACH en surmontant les deux lacunes existantes dans son homologue.

Dans I-LEACH, la première de toutes les probabilités est basée sur le critère de sélection des CHs qui est remplacé par le concept de l'énergie résiduelle. L'autre limitation a été résolue par la formation de clusters sur la base des coordonnées de l'axe des x des nœuds ; il a aidé à la distribution uniforme des CHs. Donc, les nœuds non-CH ne nécessiteront pas de transmettre leurs données sur de longues distances. Bien que I-LEACH a montré une amélioration substantielle par rapport à LEACH mais il n'a pas été autant efficace que par rapport à PEGASIS en termes de nombre de tours complétés pour une quantité donnée d'énergie.[112]

3.5.1.6.Cell Low Energy Adaptive Clustering Hierarchy (Cell-LEACH)

Dans Cell-LEACH, RCSF est divisé en plusieurs groupes ou chaque groupe est divisé à son tour en 7 sections appelées cellules. Plusieurs capteurs sont inclus dans chaque cellule à partir de laquelle un nœud de capteur est sélectionné comme Cell Head. Une fois formés, les groupes et les cellules ne peuvent plus être modifiés. Chaque nœud de la cellule envoie des données à Cell Head à l'heure désignée donnée par TDM. La fonction d'agrégation des données est effectuée par les Cell Head et les données traitées sont envoyées aux SB. [117]

- **Avantages :**
- ✓ Cell Head supprime les informations redondantes.
- ✓ Une agrégation de données est effectuée aux informations reçus de différentes sources.
- **Inconvénient :**
- ✓ CH et Cell HEAD sont déterminés d'une manière au hasard.

3.5.1.7.Multi-hop Low Energy Adaptive Clustering Hierarchy (MH-LEACH)

Dans le protocole classique LEACH, chaque nœud communique avec le CH et chaque CH communique avec la station de base et sans tenir compte du facteur de la distance et les sauts sont limités à deux sauts (du nœud au CH et du CH à la SB). Le LEACH Multi-Hop

(Multi-hop Low Energy Adaptive Clustering Hiérarchie) omet cet inconvénient de Leach et l'enrichit avec une nouvelle fonctionnalité de l'évolutivité [114].

Le protocole Multihop-LEACH [119][120] choisit un chemin optimal pour le transfert de données entre le nœud et la station de base. Il choisit essentiellement plusieurs CHs pour agir comme nœud de relais pour toute la communication.

Tout d'abord, la communication multi-hop est adoptée entre les CHs. Ensuite, selon la trajectoire optimale choisie, ces CHs transmettent les données au CH correspondant qui est le plus proche de la station de base. Enfin, ce CH envoie les données à SB.

Le setup phase de ce protocole est la même que celle du protocole LEACH, mais avec une petite amélioration du multi-saut.

3.5.1.8.Comparaison entre LEACH et ses variantes

Le tableau **3.1** montre la comparaison entre les protocoles étudiés précédemment suivant des points :

Tableau 3.1 : Comparaison entre LEACH et ses variantes [121]

protocole de routage en cluster	**Mobilité**	**Scalabilité**	**Organisation automatique**	**Nombre de sauts**	**Informations de localisation**
LEACH	SB fixe	Limité	Oui	Unique itinéraire	Non
LEACH-C	SB fixe	Bonne	Oui	Unique itinéraire	Oui
LEACH-F	SB fixe	Limité	Non	Unique itinéraire	Oui
TL-LEACH	SB fixe	Très bonne	Oui	Unique itinéraire	Oui
Multi-hop LEACH	SB fixe	Bonne	Oui	Multi chemins	Oui
I-LEACH	SB fixe	Très bonne	Oui	Unique itinéraire	Oui
Cell-LEACH	SB fixe	Très bonne	Oui	Multi chemins	Oui

3.5.2. L'approche basée sur la chaine

Dans cette approche, les nœuds vont être organisés de sorte qu'ils forment une chaîne de proches voisins connectant tous les nœuds et ensuite choisit une tête (Leader de la chaine) qui communique directement avec la SB. La formation de la chaine peut être effectuée soit d'une façon centralisée par la SB, ou d'une façon décentralisée par les nœuds capteurs eux-mêmes en employant un algorithme à vide (Greedy algorithme). Dans ce dernier cas, chaque nœud capteurs obtient tous d'abord toutes les informations sur l'emplacement de ses voisins avant de calculer localement la chaîne [97, 122].

3.5.2.1.PEGASIS [97]

- PEGASIS est un protocole basé sur la structure de la chaine.
- PEGASIS repose sur les mêmes hypothèses que LEACH, de plus il suppose que :

- Chaque nœud maintient les informations de localisation des nœuds voisins.
- Les nœuds sont capables de modifier leurs puissances de transmission.
- PEGASIS est destiné aux applications proactives.

-Comme LEACH, PEGASIS se déroule en rounds. Chaque round se compose de 3 phases :

➢ **Phase de formation d'une Chaine**

- Au début de chaque round, chaque nœud cherche le voisin le plus proche par l'envoi d'un signal atténué graduellement, jusqu'à ce qu'il soit reçu par un seul nœud.
- Appliquer 'Greedy algorithm' pour déterminer la tête de la chaine qui est généralement le plus éloigné nœud de la *SB*.
- La construction de la chaîne commence par la tête jusqu'à la *SB* en maintient dans chaque nœud la liaison entre son prédécesseur et son successeur dans la chaîne.

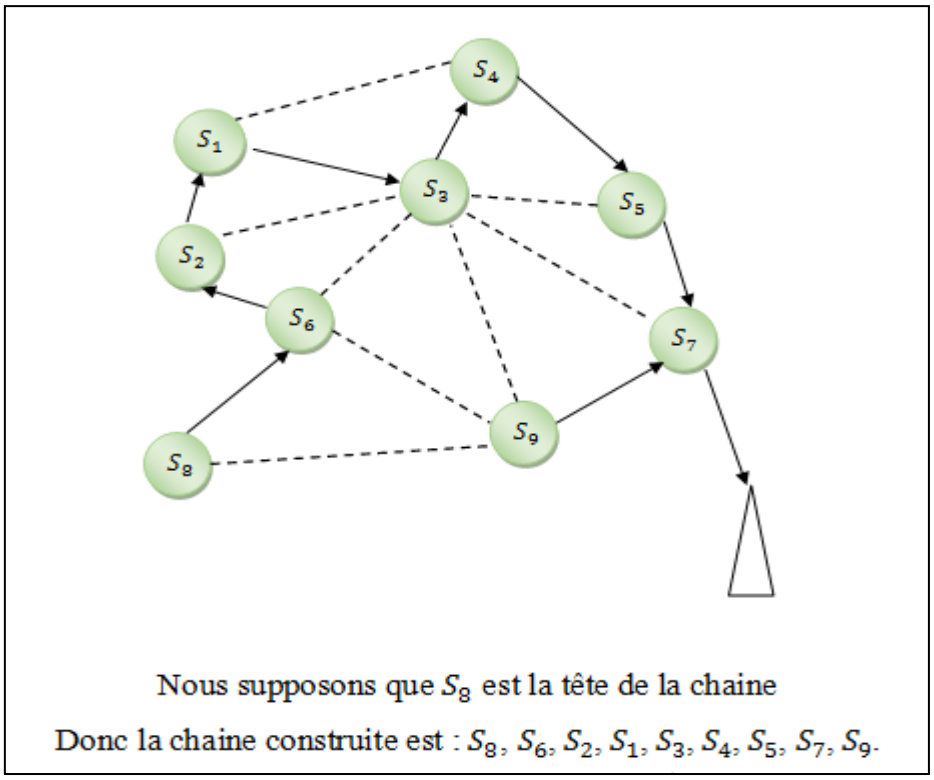

Figure 3.7 : Protocole PEGASIS : La construction de la chaîne

➢ **Itération pour sélection d'un nœud leader**

- Choisir un nœud leader (*NL*) qui prend en charge la communication avec la *SB*
- Une fois que tous les nœuds prennent le rôle d'un *NL*, un nouveau round de création d'une chaine commencera et ainsi de suite.

➢ **Phase de transmission des données**

- La *SB* envoie un jeton à la tête de la chaine pour initier la transmission des données
- Dès que la tête reçoit ce jeton il commence de transmettre leurs données à son successeur.
- Donc la transmission est réalisée de telle sorte que chaque nœud agrège les données de son prédécesseur avec les siennes et ce jusqu'à ce que toutes les données soient arrivées au *NL*.
- Dès lors, les données agrégées peuvent être transmises directement à la *SB*.

- **Avantages**

✓ La réduction du nombre d'émissions et de réceptions en utilisant l'agrégation de données.

✓ La minimisation de la dissipation de l'énergie en favorisant les communications locales et en réduisant la quantité de données transmise à la *SB*.

✓ La distribution de la charge causée par la transmission *NL-SB* sur l'ensemble des nœuds.

- **Inconvénients :**

✓ Introduction d'un overhead significatif, causée par l'ajustement dynamique de la topologie.

✓ La mise en vielle des nœuds est impossible pour ne pas rater le jeton.

✓ Les délais de transmission sont élevés pour les nœuds qui sont loin de la *SB*.

✓ Le risque de congestion au niveau d'un *NL* est fort.

✓ PEGASIS ne garantit pas la livraison des données à la *SB* car un *NL* peut tomber en panne.

✓ PEGASIS ne convient pas aux réseaux déployés dans de vastes régions.

3.5.2.2. Hierarchical Power-Efficient GAthering in Sensor Information Systems (H-PEGASIS) [88]

Ce protocole est une amélioration du protocole PEGASIS appelé Hierarchical-PEGASIS, vise à réduire les retards pris pour la transmission de données vers la station de base, et propose une solution au problème de la collecte de données en tenant compte de l'énergie.

Afin de résoudre certains problèmes rencontrés dans PEGASIS, en réduisant les retards la technique de communications simultanées de données est adoptée.

Pour éviter les collisions et les interférences dues aux signaux le long des nœuds du réseau, deux approches ont été utilisées :

– La première approche étant d'incorporer les codages du signal reçus et transmis, par exemple : CDMA (Code Division Multiple Access) ;
– La seconde approche étant de permettre juste aux nœuds qui sont spatialement séparés de transmettre les données en même temps.

Le protocole basé chaine et utilisant une approche de transmission CDMA, construit une chaîne de nœuds, qui forme un arbre hiérarchique, et chaque nœud sélectionné dans un niveau particulier transmet les données au nœud du niveau supérieur de la hiérarchie. Cette méthode permet d'assurer la transmission parallèle de données, et réduit les retards. Puisque l'arbre est équilibré, le retard sera en $O(logN)$ où N est le nombre de nœuds.

Cependant, l'approche n'utilisant pas le CDMA crée une hiérarchie à trois niveaux des nœuds, dans ce cas les effets d'interférence se trouvent réduits en planifiant avec attention les

transmissions simultanées. Un tel protocole basé chaîne a montré une amélioration de performances d'un facteur d'environ 60 comparé à la forme de base de PEGASIS.

3.5.2.3. Energy Balancing PEGASIS (EB-PEGASIS)

Dans [109], Les auteurs ont présenté un article qui est une extension du protocole H-PEGASIS dans lequel un nœud de capteur prend en compte la distance moyenne de la chaîne formée. Si la distance du nœud de capteur le plus proche à ses autres extrémités (le plus éloigné) est supérieur à la distance seuil, le nœud de capteur le plus proche est traité comme «nœud éloigné». Dans le cas où le nœud le plus proche rejoint le maillon de la chaîne, il apparaîtra comme une longue chaîne.

Dans cette situation, le "nœud distant" recherchera un proche nœud sur la chaîne établie. Avec ce schéma, le nouvel algorithme proposé, EB-PEGASIS, permet d'éviter efficacement la "chaîne longue".

3.6. Conclusion

Dans ce chapitre, nous avons effectué une étude sur la classification des protocoles du routage suivant un nombre de critères, nous avons présenté quelques exemples des protocoles de routage. Nous nous intéressons aux protocoles de routage hiérarchiques.

Chapitre 4

AED et Travaux liés

Sommaire

4.1. Introduction

4.2. Les métaheuristiques

4.3. L'algorithme à évolution différentielle

4.4. Travaux liés

4.5. Conclusion

4.1. Introduction

Dans ce chapitre, nous introduisons une étude détaillée sur l'AED, nous présentons la description de quelques travaux de recherche utilisant l'AED dans le but de la minimisation d'énergie pour les RCSFs.

4.2. Les métaheuristiques

Les "métaheuristiques" d'optimisation sont des algorithmes généraux d'optimisation applicables à une grande variété de problèmes. Elles sont apparues à partir des années 80, dans le but de résoudre au mieux des problèmes d'optimisation. Dans la pratique, trois types de problèmes d'optimisation sont souvent rencontrés : les problèmes combinatoires (discrets), les problèmes continus (à variables continues), et les problèmes mixtes [124]. L'exemple le plus répandu en optimisation combinatoire est celui du voyageur de commerce. En optimisation continue, un exemple simple est celui de la recherche des paramètres d'un modèle numérique pour approcher au mieux des données réelles [125]. Le dernier type est celui des problèmes mixtes, qui comportent à la fois des variables discrètes et des variables continues. Les métaheuristiques s'efforcent de résoudre tout type de problème d'optimisation. Elles sont caractérisées par leur caractère stochastique, ainsi que par leur origine discrète. Elles sont inspirées par des analogies avec la physique (recuit simulé, recuit microcanonique), avec la biologie (algorithmes évolutionnaires) ou encore l'éthologie (colonies de fourmis, essaims particulaires). Cependant, elles ont l'inconvénient d'avoir plusieurs paramètres à régler. Il est à souligner que les métaheuristiques se prêtent à toutes sortes d'extensions, notamment en optimisation mono objectif et multiobjectif. De nos jours les gestionnaires et les décideurs sont confrontés quotidiennement à des problèmes de complexité grandissante, qui surgissent dans des secteurs très divers. Le problème à résoudre peut souvent s'exprimer sous la forme générale d'un problème d'optimisation, dans lequel on définit une ou plusieurs fonctions objectif que l'on cherche à minimiser ou à maximiser par rapport à tous les paramètres concernés [126].

4.3. L'algorithme à évolution différentielle

4.3.1. Définition

L'algorithme à évolution différentielle (AED) (DE : Differential evolution) est l'un des algorithmes évolutionnaires (AE) qui a connu un grand développement ces dernières années, surtout dans l'industrie, et c'est l'un des

sujets actuels les plus brûlants dans la recherche en intelligence informatique [127, 128]. Il s'agit d'une métaheuristique stochastique d'optimisation des fonctions multidimensionnelles, conçue à l'origine pour les problèmes continus et sans contraintes. Ses extensions peuvent traiter les problèmes à variables mixtes [128] et gèrent les contraintes non linéaires. L'AED est inspirée par les algorithmes génétiques et les stratégies évolutionnistes, combinées avec une technique géométrique de recherche. Les algorithmes génétiques changent la structure des individus en utilisant la mutation et le croisement, alors que les stratégies évolutionnistes réalisent l'auto-adaptation par une manipulation géométrique des individus. Ces idées ont été mises en œuvre grâce à une opération de mutation sur des vecteurs [129] comparable à une sorte de « tetra-crossover » qui apporte une large perturbation.

4.3.2. Principe

L'idée de base de cet algorithme est l'utilisation d'une nouvelle stratégie de mutation. Elle consiste en l'ajout de la différence vectorielle pondérée entre deux individus à un troisième. Après l'application de cette mutation, un croisement entre l'individu mutant et l'individu correspondant de la population aura lieu. Finalement, une sélection gloutonne entre ces deux individus, le père et le fils, va décider lequel des deux individus survivra dans la prochaine génération. Dans sa version de base, il existe trois paramètres stratégiques de l'AED : la taille de la population *NP*, le taux de croisement *CR* et le facteur de différence *F* [129].

Pour chaque génération du processus d'évolution, chaque individu, qu'on appelle *individu cible*, rivalise avec un nouveau individu, appelé *individu essai*, pour survivre. Dans la prochaine génération le meilleur des deux, ayant la meilleure fitness, survivra. L'individu essai est créé en combinant l'individu cible avec un autre créé par mutation c'est *l'individu mutant* [129].

Formellement, les individus dans un AED sont représentés par des vecteurs *D*-dimensionnels de valeurs réelles x_i $i \in 1,\ldots, NP,$ où : *D* est le nombre des paramètres à optimiser et *NP* est la taille de la population. Selon la description de [129], un AED fonctionne comme décrit dans le pseudo code suivant :

Algorithme 4.1 AED

G←1
choisir *NP*, *F* et *CR*
P_G←initialiser la population aléatoirement
Tant que (le critère d'arrêt n'est pas atteint) **Faire**
  Pour (chaque individu $X_{i,G}$ de P_G) **Faire**
    choisir les parents auxiliaires $X_{r1,G}$, $X_{r2,G}$ et $X_{r3,G}$
    créer l'individu enfant $U_{i,G}$ en utilisant la mutation et le croisement différentiels
    appliquer la sélection : P_{G+1}←P_{G+1} [Meilleur($U_{i,G}$, $X_{i,G}$)
  Fin Pour
G←*G* + 1
Fin Tant que

Dans ce qui suit, on détaille les éléments d'un AED, notamment la mutation et le croisement différentiels, la sélection et le critère d'arrêt.

4.3.2.1. Mutation différentielle

Un mutant v_i correspondant à un individu cible x_i est créé à partir de trois autres individus choisis aléatoirement de la population selon l'équation 4.1. [131 ; 133 ; 134]

$$v_i = x_{i_1} + F.(x_{i_2} - x_{i_3}) \qquad \textit{(Equation 4.1)}$$

Où : $i_1, i_2, i_3 \in \{1, ..., NP\}$ sont des entiers mutuellement différents,
et $F > 0$ est un facteur constant réel positif servant à contrôler la variation différentielle $d_i = x_{i_2} - x_{i_3}$.

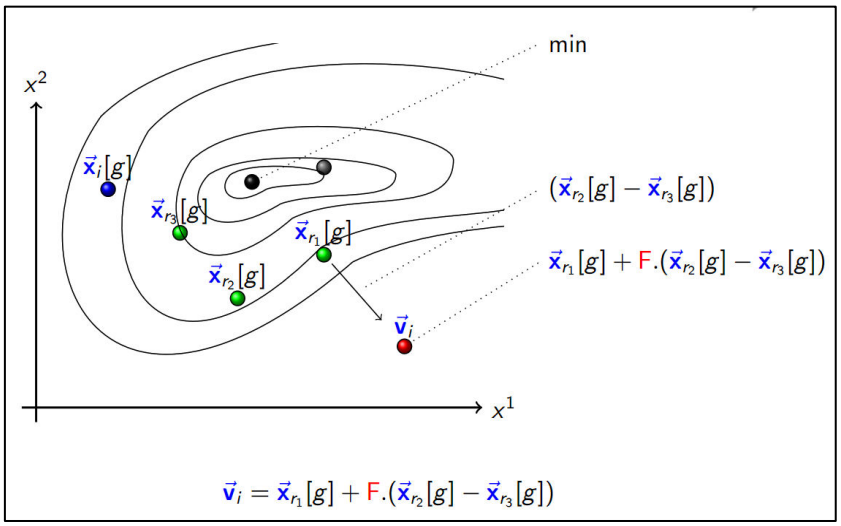

Figure 4.1 : Mutation différentielle [115]

4.3.2.2. Croisement différentiel

Le croisement dans le cadre d'un AED permet de générer un individu essai u_i à partir de la combinaison entre un mutant v_i et l'individu cible x_i. Ceci est fait selon le processus décrit par l'équation 4.2. [131 ; 133;134]

$$u_i = \begin{cases} v_i(j), si\ r_j(0,1) < CR \\ x_i(j), \qquad\qquad sinon \end{cases} \qquad \textit{(Equation 4.2)}$$

Où : $r_j(0,1)$ est un nombre aléatoire entre 0 et 1,

et $CR \in [0,\ 1]$ est le taux de croisement.

En plus de ce type de croisement appelé *croisement binomial*, il existe un autre type fréquemment utilisée dans des variantes plus récentes de l'AED que celle de Storn et Price, c'est le *croisement exponentiel*. Dans ce type de croisement, à partir d'une position choisie aléatoirement, L composants successifs de l'individu mutant, sont transmis à l'individu d'essai [133]. Ces deux types de croisement sont illustrés par la figure **4.2**, croisement binomial à gauche et croisement exponentiel à droite.

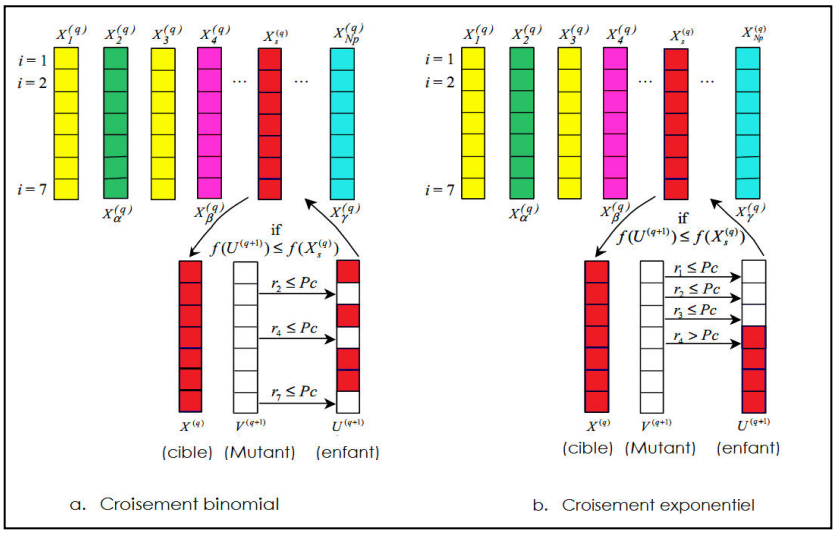

Figure 4.2: Croisement différentiel : binomial à gauche, exponentiel à droite [105]

4.3.2.3. Sélection

Comme on l'a déjà mentionné, la sélection permet de choisir un individu parmi l'individu cible x_i et l'individu essai u_i selon la qualité de leurs finesses comme indiqué dans l'équation 4.3. [131; 133 ;134]

$$x_i' = \begin{cases} u_i, si\ f(u_i) < f(x_i) \\ x_i, \qquad\qquad sinon \end{cases} \qquad \textit{(Equation 4.3)}$$

Où : $x_i^{'}$ est l'enfant remplaçant de x_i dans la prochaine génération.

4.3.2.4. Critère d'arrêt

Le critère d'arrêt d'un AED est généralement l'un des suivants.

1. Un nombre fixe d'itérations G_{max} avec une valeur largement suffisante à la résolution du problème en question.
2. L'absence d'une évolution remarquable dans la valeur de la fitness de la population pour un certain nombre d'itération successives.
3. L'obtention de la qualité requise de la fonction d'adaptation.

4.3.3. Exemple

Considérons une fonction à deux dimensions à minimiser:

$$f(x, y) = x^2 + y^2$$

Commençons avec une population à 5 solutions candidates initialisée aléatoirement dans l'intervalle (-100,100) :

$X_{1,0} = [85, -12]$

$X_{2,0} = [-14, -78]$

$X_{3,0} = [-63, -49]$

$X_{4,0} = [81, -18]$

$X_{5,0} = [96,19]$

Pour le premier vecteur X_1, sélectionner aléatoirement trois autres vecteurs différents, par exemple : X_2, X_4, X_5.

Le vecteur mutant est formé comme : $V_{1,0} = X_{2,0} + F(X_{4,0} - X_{5,0})$.

Fixons $F = 1$.

$$V_{1,0} = \begin{pmatrix} -14 \\ -78 \end{pmatrix} + 1 * \left\{ \begin{pmatrix} 81 \\ -18 \end{pmatrix} - \begin{pmatrix} 96 \\ 19 \end{pmatrix} \right\} = \begin{pmatrix} -29 \\ -115 \end{pmatrix}$$

Le vecteur d'essai $U_{1,0}$ est formé en échangeant les composants de $V_{1,0}$ avec le vecteur cible $X_{1,0}$.

Si nous fixons $CR = 0.9$.

Soit $rand(0,1) = 0.6, 0.6 < 0.9,\ U_{1,1,0} = V_{1,1,0} = -29$

Encore soit $rand(0,1) = 0.95, 0.95 > 0.9,$ donc $U_{1,2,0} = X_{1,2,0} = -12$

Le vecteur d'essai est :

$$U_{1,0} = \begin{pmatrix} -29 \\ -12 \end{pmatrix}$$

Finalement, Fitness du vecteur cible : $f(85, -12) = 7369$

Fitness du vecteur d'essai : $f(-29, -12) = 985$

$f(U_{1,0}) < f(X_{1,0})$, d'où le vecteur d'essai $U_{1,0}$ survivra pour la génération $G = 1$.

De même pour les autre candidates.

4.3.4. Variantes pour l'AED

L'algorithme de [129] décrit ci-dessus est connu sous le nom de DE/rand/1. Avec l'utilisation d'un croisement binomial, cette variante est appelée DE/rand/1/bin. De cette façon, les différentes variantes de l'AED sont nommées. La convention générale est DE/x/y/z, où : 'x' est le vecteur base à être perturbé, 'y' est le nombre de vecteurs de différences utilisés dans la perturbation de 'x', et 'z' est une abréviation du type de croisement utilisé (exp :exponentiel, bin : binomial).

Les variantes de l'AED diffèrent généralement entre eux par le type de mutation utilisé. En plus de la mutation de base de leur algorithme, [129] proposent quatre autres schémas de mutation présentés par les équations 4.4 à 4.7.

DE/best/1 : $\vec{V}_{i,G} = \vec{X}_{best,G} + F.(\vec{X}_{r_1^i,G} - \vec{X}_{r_2^i,G})$ *(Equation 4.4)*

DE/target-to-best/1 : $\vec{V}_{i,G} = \vec{X}_{best,G} + F.(\vec{X}_{best,G} - \vec{X}_{i,G}) + F.(\vec{X}_{r_1^i,G} - \vec{X}_{r_2^i,G})$ *(Equation 4.5)*

$$\text{DE/best/2} : \vec{V}_{i,G} = \vec{X}_{best,G} + F.(\vec{X}_{r_1^i,G} - \vec{X}_{r_2^i,G}) + F.(\vec{X}_{r_3^i,G} - \vec{X}_{r_4^i,G}) \quad \textit{(Equation 4.6)}$$

$$\text{DE/rand/2} : \vec{V}_{i,G} = \vec{X}_{r_1^i,G} + F.(\vec{X}_{r_2^i,G} - \vec{X}_{r_3^i,G}) + F.(\vec{X}_{r_4^i,G} - \vec{X}_{r_5^i,G}) \quad \textit{(Equation 4.7)}$$

Les indices r_1^i , r_2^i , r_3^i , r_4^i et r_5^i sont des entiers mutuellement exclusifs choisis aléatoirement de l'ensemble $\{1, ...,NP\}$ et ce sont tous différents de l'indice de base *i*. Ces indices sont générés aléatoirement pour chaque opération de mutation.

4.4. Travaux liés

Un certain nombre d'algorithmes de clustering basés sur l'AED ont été développé pour les RCSFs, nous allons présenter tels travaux dans cette section.

4.4.1. LEACH Protocol Improvement Based on the Differential Evolution Algorithm (DE_LEACH) [130]

Xiaofang Li et al. ont proposé une amélioration du protocole LEACH basée sur l'AED utilisant les fonctions de recherche simples et rapides d'AED pour optimiser la sélection multi-objectifs des CHs et empêcher les nœuds aveugles pour améliorer l'efficacité énergétique et la stabilité du système.

L'amélioration du protocole LEACH basée sur l'AED s'exécute en tours comme le protocole LEACH traditionnel. Chaque cycle est divisé en quatre phases: (1) le partitionnement des clusters initiales, (2) la collecte des informations d'état sur les nœuds à l'intérieur des clusters par les CHs auxiliaires, (3) l'optimisation et la sélection des CHs avec l'algorithme d'ED, et (4) formation des clusters optimisées. Parmi elles, la troisième est la clé des améliorations proposées dans le présent document et sera décrit en détail.

4.4.1.1. Partitionnement des clusters initiaux

Dans cette phase, l'algorithme de routage LEACH traditionnel est utilisé pour la partition des clusters initiaux pour les RCSFs.

4.4.1.2. Collecte d'informations d'état des nœuds à l'intérieur du cluster par les CHs auxiliaires

Les nœuds à l'intérieur du cluster envoient les informations d'état de leurs propres positions et de leur énergie aux CHs auxiliaires. Maintenant, la position $P(p_1, p_2, ..., p_n)$ et l'information d'énergie $E(e_1, e_2, ..., e_n)$ des nœuds voisins sont stockés dans les CHs auxiliaires, où n est le nombre de nœuds du le groupe, p_i est la valeur de localisation du nœud i et e_i est la valeur d'énergie du nœud i.

4.4.1.3. Optimisation de l'algorithme et détermination des CHs avec l'AED

L'algorithme de l'AED s'exécute fondamentalement de la même manière que d'autres algorithmes d'évolution, y compris les opérations de codage, de formation initiale d'essaims, de variation, de croisement et de sélection. Afin de l'adapter au domaine du problème, l'AED est modifié comme suit :

- Formation initiale d'essaims: Les essaims initiaux sont formés avec les groupes correspondants de séquences entières de numéros ID de nœuds voisins collectés par les CH auxiliaires.

Définir le vecteur de solution comme:

$$X_{i,G} = (x_1, x_2, \ldots, x_n) \quad i = 1, 2, \ldots, NP$$

Chaque vecteur de solution est un individu d'évolution. Chaque génération d'un essaim d'évolution est exprimée par G, i est la position d'un individu dans l'essaim, n est le nombre de nœuds voisins collectés par les CHs auxiliaires et NP est l'échelle de l'essaim (taille de la population).

L'essaim de première génération est créé aléatoirement. La règle consiste à trier les IDs des nœuds voisins collectés par les CHs auxiliaires dans un ordre croissant $ID_1 < ID_2 < \ldots < ID_n$, pour créer une relation correspondante entre $ID_i \leftrightarrow$i, i = 1,…, n et créer le essaim initial basé sur cette relation. Chaque élément de l'essaim est donné comme:

$$x_{ij,1} = 1 + round\,[rd \times (n-1)],\ i = 1, 2, \ldots, NP,\ j = 1, 2, \ldots, n$$

Où : rd est un nombre aléatoire compris entre (0,1), $round$ [] est l'entier le plus proche et n le nombre de nœuds voisins.

- Les opérations de la mutation et du croisement sont effectuées de même que l'AED traditionnel.

- Opération de sélection: pour calculer la valeur d'adaptation de chaque vecteur expérimental, il est nécessaire d'établir d'abord une fonction objective. Chaque vecteur est composé de n entiers compris entre 1∼n et correspondant aux numéros ID des nœuds voisins, respectivement. Le but de l'établissement de la fonction est de connecter une relation physique réelle à des nombres abstraits dans l'algorithme afin de juger de la qualité des vecteurs.

La détermination de la valeur d'adaptation des vecteurs d'expérience doit non seulement prendre en compte l'énergie des nœuds correspondants, mais également la répartition de l'énergie des nœuds environnants. Plus les nœuds voisins sont éloignés du nœud, plus l'énergie est importante; et vice versa. Sur la base de cette caractéristique, une fonction objective est définie comme suit:

$$f(i) = \frac{1}{n}\sum_{j=1}^{n}(\alpha e_j + \beta \bar{e}_j) \qquad \textit{(Equation 4.8)}$$

Où : $\alpha + \beta = 1$ et $\alpha \in [0,1]$, $\beta \in [0,1]$; e_j est l'énergie de nœud correspondant de l'élément *j* dans le vecteur expérimental; $\bar{e}_j$ est l'énergie moyenne des nœuds sauf le nœud *j*; α est le facteur énergétique d'influence de e_j; et β est le facteur d'énergie influent de $\bar{e}_j$. Ajuster α et β ajustera le taux de contribution de e_j et $\bar{e}_j$ à la valeur d'adaptation. Dans cet article, α = β = 0.5.

L'énergie équivalente moyenne $\bar{e}_j$ est définie comme suit:

$$\bar{e}_j = \frac{1}{n-1}\sum_{i=1, j\neq i}^{n} \tilde{e}_i \qquad \textit{(Equation 4.9)}$$

Où : $\tilde{e}_i$ est l'énergie équivalente du nœud *i*. Supposons que r_i soit la distance entre le nœud *i* et le nœud *j* et e_i l'énergie restante du nœud *i*. L'énergie restante du nœud *i* et sa distance au nœud *j* doivent être prises en compte ensemble pour la structure de *f* (e_i, r_i). Les conditions suivantes doivent être satisfaites: plus la distance entre le nœud *i* et le nœud *j* est grande, moins l'énergie équivalente du nœud *i* est faible, et inversement. La fonction reflète les caractéristiques des problèmes en simulation. Au total, la fonction cible est:

$$f(i) = \frac{1}{n}\sum_{j=1}^{n}[\alpha e_j + \beta \frac{1}{n-1}\sum_{i=1, j\neq i}^{n} e_j . \exp(-\frac{r_i}{l})] \qquad \textit{(Equation 4.10)}$$

Le vecteur d'expérience $U_{i,G+1}$ résultant de l'opération de croisement est comparé au vecteur objectif $X_{i,G}$ substitué dans la fonction objective du vecteur. Si la valeur de la fonction

objectif de $U_{i,G+1}$ est supérieure à celle de $X_{i,G}$, alors $X_{i,G}$ est remplacée par $U_{i,G+1}$. Sinon, $X_{i,G}$ seront conservés, ainsi l'essaim de la prochaine génération sera créé. À présent, un essaim a terminé une génération d'évolution. L'AED exécute de manière répétée les étapes de variation, de croisement et de sélection jusqu'à ce que la condition d'arrêt soit satisfaite. La condition d'arrêt est généralement l'évolution des générations, et l'élément avec la position dominante absolue dans les vecteurs d'expérience est simplement le nombre correspondant du CH final.

4.4.1.4. Formation des clusters optimisés

Au cours de cette phase, les CHs auxiliaires envoient les informations des CHs optimisées aux nœuds situés à l'intérieur du cluster. Les CHs optimisés collectent et intègrent les informations des nœuds situés à l'intérieur du cluster.

La formation des clusters optimisés prend en compte de manière exhaustive les informations d'état des nœuds voisins. Par conséquent, la perte d'énergie des nœuds à l'intérieur du cluster évite même les nœuds aveugles fréquents.

4.4.2. Differential Evolution based Clustering Algorithm (DECA) [132]

Pratyay Kuila, Prasanta K. Jana ont proposé un algorithme de clustering basé sur l'AED afin de prolonger la durée de vie du réseau en empêchant une mort plus rapide des CHs très chargés. Ils incorporent une phase d'amélioration locale à l'AED traditionnel pour une convergence plus rapide et de meilleures performances.

4.4.2.1. L'algorithme proposé

La configuration du réseau s'effectue en deux phases: amorçage et clustering. Pendant le processus d'amorçage, un ID unique est attribué à chaque nœud de capteur et à chaque passerelle (gateway). Ensuite, les nœuds de capteur diffusent leurs IDs à l'aide du protocole CSMA / CA. Par conséquent, les passerelles situées dans le rayon de communication de ces nœuds de capteur peuvent collecter les IDs de capteurs et finalement envoyer les informations de localisation à la SB. Ainsi, pour chaque nœud de capteur, le nombre de passerelles dans son rayon de communication peut être calculé par la SB. En phase de clustering, la SB exécute l'algorithme de clustering. Lorsque la mise en cluster est terminée, toutes les passerelles fournissent leurs IDs aux nœuds de capteurs membres par une communication à saut unique. Ensuite, les passerelles fournissent un ordonnancement TDMA à leurs nœuds de capteurs membres pour la communication intra-cluster.

a. Initialisation du vecteur de population

Le schéma proposé pour la représentation du vecteur est le suivant : les vecteurs sont représentés de telle sorte que chaque vecteur indique l'affectation complète de tous les nœuds de capteur aux passerelles. La représentation de l'$i$ème vecteur de la population de la Gème génération est notée :

$$X_{i,G} = [x_{1,i,G}, x_{2,i,G}, \ldots, x_{n,i,G}]$$

Où N est la dimension du vecteur et la valeur du $j$ième composant, i.e., $x_{j,i,G}$ mappe l'affectation du nœud de capteur s_j à une passerelle. Chaque composant est initialisé avec un nombre uniformément distribué généré aléatoirement *Rand(0,1]*. Chaque composant est initialisé avec un nombre uniformément distribué généré aléatoirement *Rand(0,1]*. Ainsi, le $j$ième composant de ce vecteur, i.e., $x_{j,i,G}$ = Rand (0, 1], 1 ≤ j ≤ N mappe une passerelle (disons g_k) à laquelle le nœud de capteur s_j est affecté. Le mappage se fait comme suit:

$$g_k = Index(ComCH(s_j), H) \qquad \textit{(Equation 4.11)}$$

Où $Index(ComCH(s_j), H)$ est une fonction d'indexage qui indexe le Hième passerelle de $ComCH(s_j)$ et $H = Plafond(x_{j,i,G} * |ComCH(s_j)|)$.

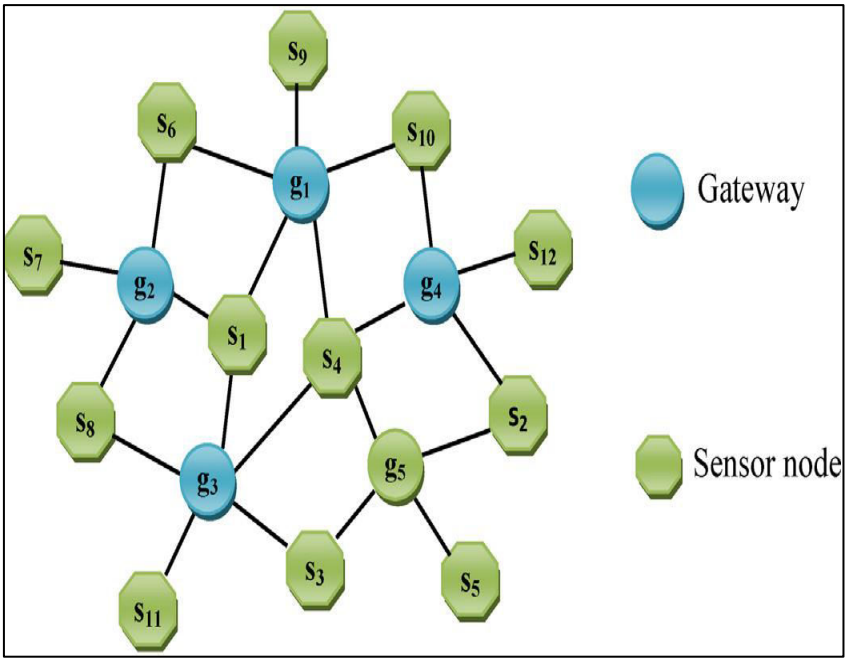

Figure 4.3 : Un réseau de capteur sans fil avec des passerelles

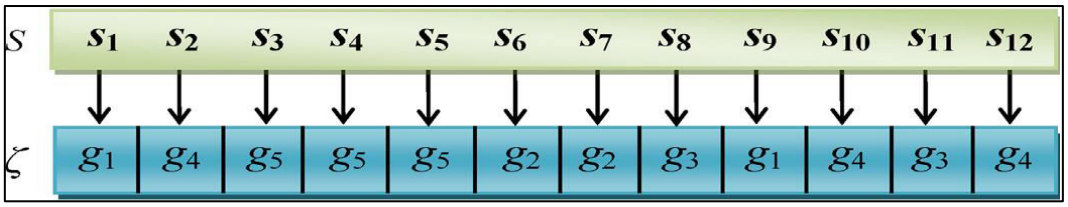

Figure 4.4 : Affectation finale

- **Illustration**

Considérons un RCSF avec 12 nœuds de capteur et 5 passerelles, c-à-d S = {s_1,s_2 ,…, s_{12} } et ζ = { g_1 , g_2 , g_3 , g_4 , g_5 } comme indiqué dans la Fig 4.1. Par conséquent, la dimension du vecteur est identique au nombre de nœuds de capteur, c'est-à-dire, D = 12.

Les arcs entre les nœuds de capteur et les passerelles indiquent que les passerelles se trouvent dans la zone de communication des nœuds de capteur. On peut observer sur la figure **4.1** que le nœud de capteur s_1 est connecté de trois passerelles. En d'autres termes,

ComCH (s_1) = {g_1 , g_2 , g_3 }.

Le tableau **4.1** représente l'affectation des nœuds de capteur à partir du vecteur de codage.

Tableau 4.1: Affectation des nœuds de capteur à partir du vecteur de codage

Sensor nodes	$ComCH(s_J)$	$\|ComCH(s_J)\|$	$x_{J,I,G}$	Ceiling ($x_{J,I,G} \times \|ComCH(s_J)\|$)	Assigned gateway
s_1	$\{g_3, g_1, g_2\}$	3	0.46	2	g_1
s_2	$\{g_4, g_5\}$	2	0.19	1	g_4
s_3	$\{g_5, g_3\}$	2	0.39	1	g_5
s_4	$\{g_1, g_4, g_5, g_3\}$	4	0.67	3	g_5
s_5	$\{g_5\}$	1	0.86	1	g_5
s_6	$\{g_1, g_2\}$	2	0.63	2	g_2
s_7	$\{g_2\}$	1	0.24	1	g_2
s_8	$\{g_3, g_2\}$	2	0.41	1	g_3
s_9	$\{g_1\}$	1	0.34	1	g_1
s_{10}	$\{g_1, g_4\}$	2	0.73	2	g_4
s_{11}	$\{g_3\}$	1	0.63	1	g_3
s_{12}	$\{g_4\}$	1	0.92	1	g_4

b. Fonction de fitness

La fonction de fitness est définie comme:

$$Fitness = \frac{1}{\sqrt{\frac{1}{m}\sum_{j=1}^{m}(\mu_L - L(j))^2}} X \frac{1}{\sqrt{\frac{1}{m}\sum_{j=1}^{m}(\mu_D - AvegClusDist(g_j))^2}} \quad \textit{(Equation 4.12)}$$

i.e.,

$$Fitness = \frac{m}{\sqrt{\sum_{j=1}^{m}(\mu_L - L(j))^2 X \sum_{j=1}^{m}(\mu_D - AvegClusDist(g_j))^2}} \quad \textit{(Equation 4.13)}$$

Où:

$$\mu_L = \frac{1}{m}\sum_{i=1}^{m} L(i) \qquad \text{et} \qquad \mu_D = \frac{1}{m}\sum_{i=1}^{m} AvegClusDist(g_i)$$

m: nombre des passerelles (CHs)

$L\ (i)$: la durée de vie de la passerelle g_j. Si g_j a une énergie résiduelle $E_{residual}(g_j)$ et consomme de l'énergie $E_{Gateway}(g_j)$ par tour, alors $L\ (i)$ peut être calculé comme suit:

$$L\ (i) = \frac{E_{residual}(g_j)}{E_{Gateway}(g_j)} \quad \textit{(Equation 4.14)}$$

$AvegClusDist(g_j)$: indique les distances moyennes de tous les nœuds de capteur membres à g_j. Par conséquent, ceci est calculé comme suit:

$$AvegClusDist(g_j) = \frac{1}{n_j}\sum_{i=1}^{n} dist(s_i, g_j) * \alpha_{i,j} \quad \textit{(Equation 4.15)}$$

Où : $\alpha_{i,j}$ est une variable booléenne telle que:

$$\alpha_{i,j} = \begin{cases} 1, & si\ le\ noeud\ s_i\ est\ affecté\ au\ CH\ g_j \\ 0, & sinon. \end{cases} \quad \textit{(Equation 4.16)}$$

c. Mutation et croisement

Les auteurs utilisent le schéma DE/best/1/bin pour les operations de mutation et de croisement.

d. Amélioration locale

L'étape d'amélioration locale est ajoutée pour améliorer la qualité du vecteur d'essai nouvellement généré, comme suit: Pour effectuer le processus d'amélioration locale, on sélectionne la passerelle avec une durée de vie minimale à partir du vecteur d'essai. Le but de la sélection de cette passerelle est d'augmenter sa durée de vie. Il est donc possible de prolonger la durée de vie du réseau en retardant le premier décès de la passerelle. Cela peut être possible si on peut réduire la charge de la passerelle en réaffectant un nœud de capteur membre de son cluster à un autre cluster avec une passerelle à durée de vie plus longue. Pour ce faire, un nœud de capteur est sélectionné de manière aléatoire dans le cluster de la passerelle à durée de vie minimale et est réaffecté à une autre passerelle qui en est plus proche que les passerelles restantes.

4.4.3. Hybrid differential evolution and simulated anneling (DESA) [135]

Dans cet article, les auteurs ont proposé un algorithme hybride de l'AED et de recuit simulé (DESA) pour le clustering et le choix des CHs. Comme les CHs sont généralement surchargées avec un nombre élevé de nœuds de capteurs, la mort des nœuds est généralement rapide en raison d'une élection incorrecte des CHs. Par conséquent, cet article visait à prolonger la durée de vie du réseau en empêchant la mort prématurée des CHs.

4.4.3.1. L'algorithme proposé

- L'ensemble des nœuds de capteurs est noté $S = \{s_1, s_2, \dots, s_n\}$
- Parmi l'ensemble des nœuds de capteurs, un petit pourcentage est selectionné comme CH
- Tous les nœuds sont affectés aux CHs en suivant la procédure décrite à la section **4.4.2**.

a. Initialisation

Les auteurs ont utilisé la méthode de point opposée (opposite point method) dans la phase d'initialisation.

b. Mutation

Un nombre aléatoire compris dans l'intervalle [0, 1] est choisi et comparé à une valeur seuil, dans ce cas 0,5. Si le nombre aléatoire choisi est inférieur au seuil, DE/rand/1 est choisi, sinon DE/current-to-best/1 est effectué.

c. Croisement

Une partie des vecteurs mutants est conservée telle quelle dans le DE classique avec la probabilité *Cr*. Les autres caractéristiques ne sont pas directement extraites du vecteur parent mais sont considérées comme un mélange, avec un ratio défini. Le taux de mélange Br détermine le taux selon lequel le mélange se produit.

$$u_{j,i,G} = Br * x_{j,i,G} + (1 - Br) * v_{j,i,G} \qquad \textit{(Equation 4.17)}$$

Où le taux de mélange Br est $Br = N\,(0.5, (1/2\pi))$ et $N = (\mu, \sigma)$ est la distribution normale avec moyenne μ et écart type σ.

d. Sélection

$$x_{i,G+1} = \begin{cases} u_{i,G} \; fitness(u_{i,G}) \geq fitness(x_{i,G}) \\ \begin{cases} u_{i,G} & choisir\ avec\ probabilité\ p_i \\ x_{i,G} & sinon \end{cases} \end{cases} \qquad \textit{(Equation 4.18)}$$

La probabilité est exprimée sous forme exponentielle de la différence entre la fonction de fitness et la valeur de fitness moyenne à chaque tour.

$$probability = \exp(-\frac{fitness_{u_i(t+1)} - fitness_{x_i(t)}}{average(fitness)}) \qquad \textit{(Equation 4.19)}$$

e. Fonction de fitness

La distance et les énergies des nœuds sont les facteurs pris en compte comme suit :

$$fitness = \varepsilon * f_1 + (1 - \varepsilon) * f_2 \qquad \textit{(Equation 4.20)}$$

$$f_1(i) = \frac{E(i)}{\sum_{k=1,k\neq i}^{m} E(k)} \qquad \textit{(Equation 4.21)}$$

$$f_2(i) = \frac{(m-1)}{\sum_{k=1,k\neq i}^{m} d(i,k)} \qquad \textit{(Equation 4.22)}$$

où ε est une constante définie par l'utilisateur, elle détermine la contribution de chacune des fonctions utilisées, f_1 est le ratio entre l'énergie du nœud actuel et l'énergie des nœuds du cluster et f_2 est la distance euclidienne totale des nœuds du cluster au nœud *i* et *d(i,k)* se réfère à la distance entre le nœud *i* et le nœud *k* et *m* est le nombre de nœuds du cluster particulier.

4.4.4. Switching-Differential Evolution(S-DE) for Cluster Head Election in Wireless Sensor Network (S-DE) [136]

Les auteurs ont proposé un algorithme d'optimisation S-DE (Switching Differential Evolution) dans lequel ils ont utilisé la technique de commutation avec l'AED pour le clustering et l'élection des CHs afin d'augmenter la durée de vie et réduire la complexité du réseau.

4.4.4.1. Commutation

La commutation est un moyen par lequel l'élection du CH est effectuée dans le réseau. Cette technique sélectionne le CH sur la base de l'énergie restante des nœuds entre des tours spécifiques. Dans le cluster, le nœud ayant l'énergie restante la plus élevée est sélectionné en tant que CH. Par exemple dans la figure **4.4**, l'algorithme d'optimisation sélectionne le nœud C en tant que CH dans le 1er cluster pour le tour 1.

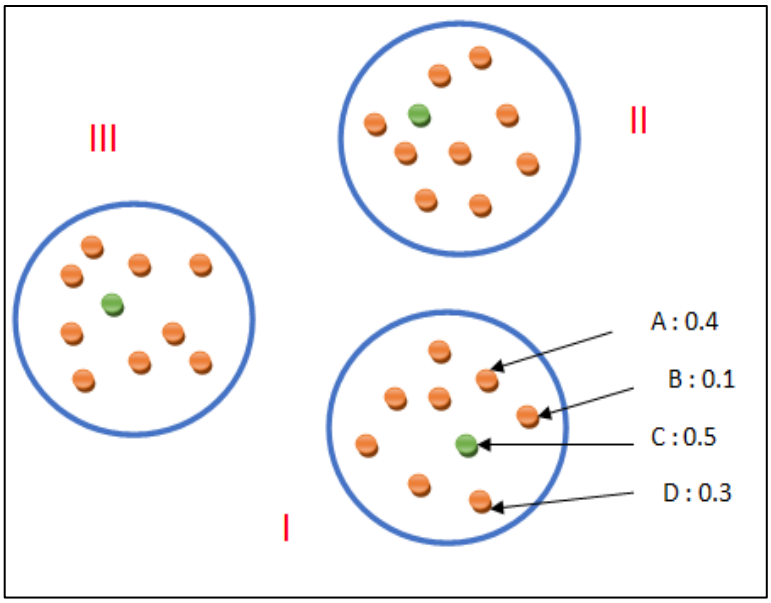

Figure 4.4 : Énergie restante au premier tour

4.4.4.2. Fonction de fitness

La durabilité d'un individu dépend fortement de la valeur de fitness. Dans cette étude, la fonction de fitness comprend les trois paramètres suivants:

- L'énergie résiduelle (RE)
- La distance des CHs à la SB (BSDist)
- La distance totale de communication intra-cluster (ICDist)

En tout, la fonction de fitness est définie comme:

$$Fitness = RE + \frac{ICDist}{N} + \frac{BSDist}{N} \qquad (Equation\ 4.23)$$

4.4.5. Comparaison entre les travaux

Dans ce qui suit nous effectuons une comparaison entre les différents travaux selon plusieurs paramètres.

4.4.1.5. Comparaison suivant les paramètres de l'AED

Tableau 4.2 : Comparaison suivant les paramètres de l'AED

Travaux / Paramètres	DE_LEACH [130]	DECA [132]	DESA [135]	S-DE [136]
Vecteur de solution	Composé des nœuds voisins des CHs	Composé des passerelles	Composé des nœuds CHs	/
Initialisation	Même que l'AED classique	Même que l'AED classique	Opposite point method	/
Mutation	DE/rand/1	DE/best/1	DE/rand/1, DE/current-to-best/1	/
Croisement	binomial	binomial	binomial, utilisant un taux de mélange	/
Sélection	Même que l'AED classique	Même que l'AED classique	Utilisant Simulated Annealing (SA)	/
Fonction de fitness	Énergie résiduelle, Distance inter-cluster	Durée de vie, Distance moyenne inter-cluster	Énergie résiduelle, Distance inter-cluster	Énergie résiduelle, Distance totale, intra-cluster, Distance entre CH et SB
Critère d'arrêt	Nombre d'itérations	Nombre d'itérations	/	/

4.4.1.6. Comparaison suivant autres paramètres

Tableau 4.3 : Comparaison suivant autres paramètres

Travaux / Paramètres	DE_LEACH [130]	DECA [132]	DESA [135]	S-DE [136]
Technique utilisée	AED, LEACH	/	AED, SA	AED, commutation
Modification par rapport à l'AED classique	/	Amélioration locale après la phase de croisement	/	/
Critère de sélection des CHs	aléatoire	utilise des passerelles comme CHs	aléatoire	Énergie restante
Nature du réseau	homogène	hétérogène	homogène	homogène
Outil utilize	simulateur NS2	MATLAB R2012b, langage C	MATLAB R2015a	MATLAB
Comparaison avec	LEACH, MTE, LEACH-C	GA, DE, EELBCA, LBC, GLBCA	LEACH, DE, HSA, modified HSA	GA

Dans l'algorithme DE_LEACH, la formation des clusters est effectuée avec le protocole LEACH et l'AED est utilisée seulement pour la détermination des CHs optimisés. Tandis que dans les deux travaux DECA et DESA, les auteurs, utilisent un schéma efficace du vecteur de solution pour un clustering complète. En se basant sur l'idée de ces deux derniers travaux, nous implémentons notre protocole.

4.5. Conclusion

Dans ce chapitre, nous présentons une description de quelques algorithmes du clustering pour les RCSFs. Nous nous sommes appuyés sur ses travaux pour la conception d'un nouveau protocole décrit dans le chapitre suivant.

Chapitre 5

Contribution

Sommaire

5.1. Introduction

5.2. Modèle du système

5.3. Description du protocole proposé DEBRC

5.4. Implémentation du protocole DEBRC

5.5. Conclusion

5.1. Introduction

Dans ce chapitre, nous présentons notre contribution résumé en : description du fonctionnement du protocole proposé, simulation et résultats pour démontrer l'efficacité énergétique de notre protocole.

5.2. Modèle du système

5.2.1. Modèle d'énergie

Le modèle utilisée est un modèle de radio simple discuté dans [31]. Lorsque la distance entre deux nœuds ou la SB est supérieure à une distance d_0, le modèle (*mp*) de propagation par trajets multiples (multipath fading channel model) est utilisé, et dans le cas inférieur, le modèle (*fs*) de l'espace libre de canal (free space channel model) est utilisé.

Les équations utilisées pour calculer les coûts de transmission et les coûts de réception pour un message de l bits et une distance d sont indiqué ci-dessous:

- Pour transmettre un message de l bits sur une distance de d mètres, l'émetteur consomme:

$$E_{TX}(l,d) = \begin{cases} l * E_{elec}(l,d) + \varepsilon_{fs} * d^2 & d < d_0 \\ l * E_{elec}(l,d) + \varepsilon_{mp} * d^4 & d > d_0 \end{cases} \qquad \textit{(Equation 5.1)}$$

- E_{elec} : L'énergie suffisante pour transmettre ou recevoir un seul bit
- ε_{fs} : Facteur de l'amplification correspond au modèle « free space channel »
- ε_{mp} : Facteur de l'amplification correspond au modèle « multipath fading channel »
- l : La taille d'un message.
- d_0 : La distance limite pour laquelle les facteurs d'amplification changent de valeur
- d : La distance entre l'émetteur et le récepteur

- Pour recevoir un message de L bits, le récepteur consomme :

$$E_{RX}(l) = l * E_{elec} \qquad \textit{(Equation 5.2)}$$

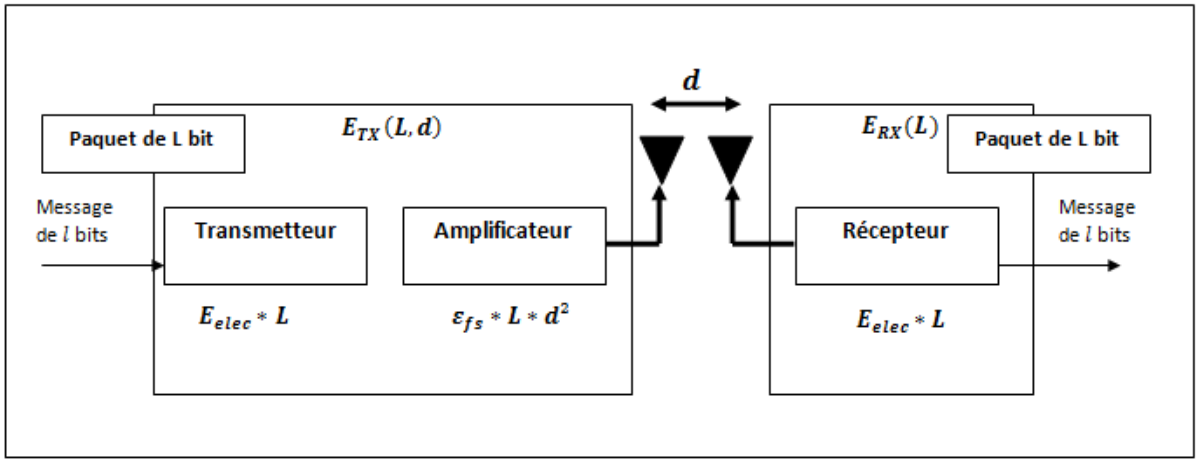

Figure 5.1. Le modèle de dissipation de l'énergie

5.2.1. Modèle du réseau :

Dans notre protocole, nous considérons:

- Un RCSF où tous les nœuds de capteurs sont déployés aléatoirement dans une espace a deux dimensions et une SB à une position fixe.
- Les nœuds de capteurs sont considérés stationnaires après le déploiement.
- Tous les nœuds de capteurs sont homogènes.
- Chaque nœud de capteur peut communiquer directement ou via d'autre nœud avec tous les nœuds du réseau.
- Un petit pourcentage (5 %) des nœuds est choisi comme CH.

5.3. Description du protocole proposé DEBRC

Notre protocole (**D**ifferential **E**volution **B**ased **C**lustering and **R**outing protocol) est un protocole de routage centralisé (exécuté par la SB). Initialement :

- Des IDs uniques sont attribués à tous les nœuds de capteurs.
- Tous les nœuds de capteurs envoient à la SB des informations de leurs IDs, leurs positions et leurs énergies initiales.

Le protocole s'exécute en plusieurs cycles, chaque cycle est divisé en trois phases : phase de clustering, phase de formation de la chaine et phase de transmission des données.

5.3.1. Phase de clustering

Dans cette phase, les CHs sont choisis aléatoirement et les clusters sont formés à l'aide de l'AED.

- L'ensemble des nœuds de capteurs est noté S = $\{s_1, s_2, \ldots, s_n\}$.
- L'ensemble des CHs est noté ζ = $\{CH_1, \ldots, CH_m\}$.

5.3.1.1. Initialisation

Chaque vecteur de la population indique l'affectation complète de tous les nœuds de capteurs aux CHs. La représentation de l'*i*ème vecteur de la population de la *G*ième génération est notée :

$$X_{i,G} = x_{1,i,G}, x_{2,i,G}, \ldots, x_{j,i,G}, \ldots, x_{N,i,G}$$

Où : N est la dimension du vecteur et le nombre des nœuds non-CH.

$x_{j,i,G}$ mappe l'affectation du nœud de capteur s_j à un CH. L'affectation est effectuée de la façon suivante :

$$CH_k = Plafond(x_{j,i,G} * Nb_{CH}) \qquad \textit{(Equation 5.3)}$$

Où: Nb_{CH} est le nombre des CHs.

5.3.1.2. Mutation

Nous utilisons les deux schémas : DE/rand/1 et DE/best/1 pour l'opération de la mutation.

5.3.1.3. Croisement

Pour le croisement, Nous utilisons les deux types (bin : binomial, exp :exponentiel).

5.3.1.4. Fonction objective

Pour la fonction de fitness, la distance est l'énergie sont les facteurs pris en compte comme suit :

$$Fitness(i) = \alpha * f_1 + (1 - \alpha) * f_2 \qquad \textit{(Equation 5.4)}$$

Où: i est la position du vecteur dans la population, α est une constante définie par l'utilisateur déterminant la contribution de chacune des fonctions utilisées. *(Equation 5.3)*

$$f_1 = \sum_{k=1}^{m} \frac{\sum_{j=1, j\neq k}^{n_c} E(j)}{E(k)} \qquad \textit{(Equation 5.5)}$$

Où f_1 est la somme des rapports de l'énergie de tous les nœuds du cluster particulier à l'énergie de CH (k). m est le nombre des CH, n_c est le nombre des nœuds du cluster particulier, $E(j)$ est l'énergie du nœud j, $E(k)$ est l'énergie du nœud k.

$$f_2 = \sum_{k=1}^{m} \sum_{j=1, j\neq k}^{n_c} d(j,k) \qquad \textit{(Equation 5.6)}$$

Où f_2 est la somme total des distances inter-cluster, $d(j,k)$ est la distance euclidienne de CH (k) et le nœud j.

5.3.1.5. Sélection

Dépendant de la fonction de fitness, l'individu ayant la minimum valeur de fitness est choisi pour la prochaine génération.

5.3.1.6. Critère d'arrêt

L'algorithme s'arrête quand la valeur de la fitness de la population entière est la même (les individus sont similaires).

A la fin de la phase de clustering, la SB envoie aux nœuds les IDs des CHs auquels ils appartiennent, et envoie aux CHs les IDs des nœuds de leur cluster.

La figure **5.2** représente l'organigramme du protocole DEBCR :

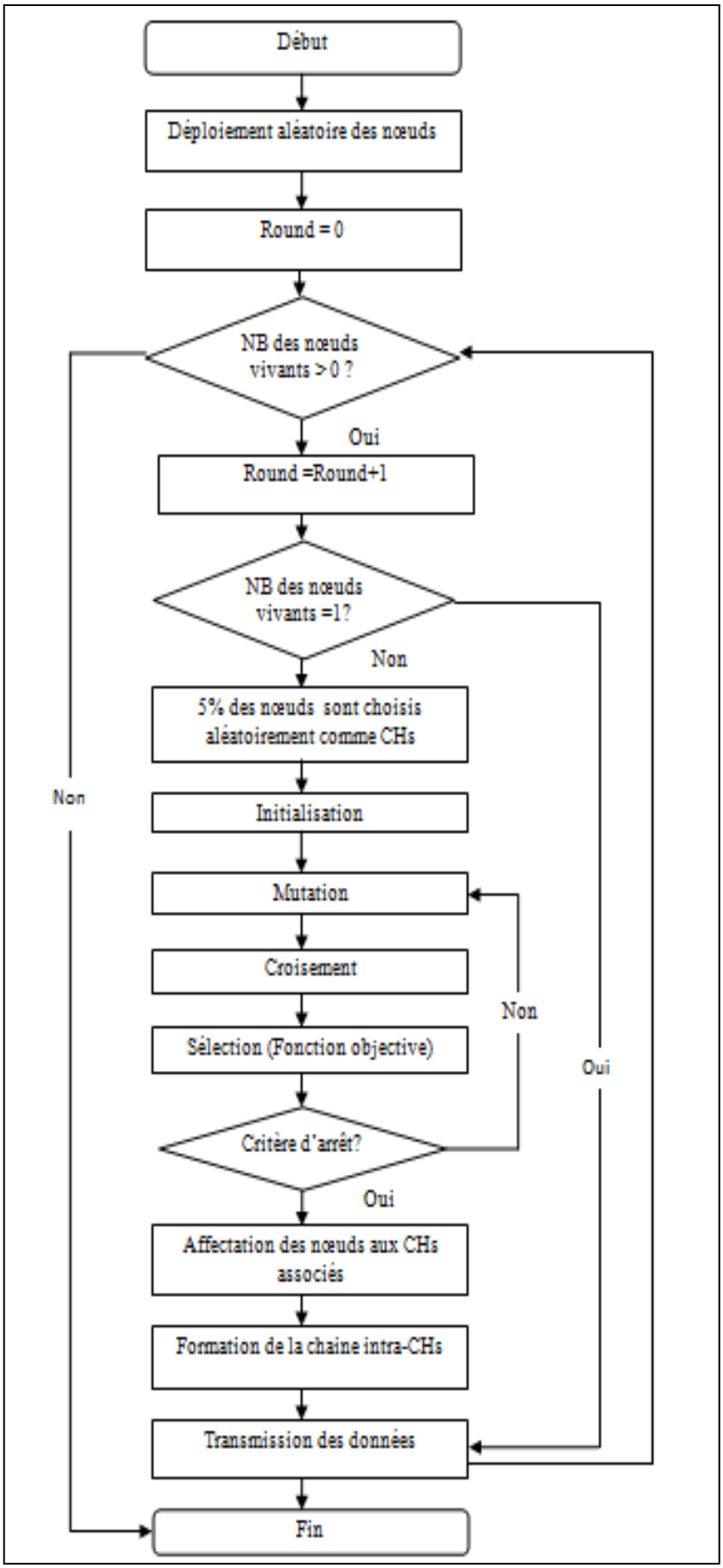

Figure 5.2 : Organigramme du protocole DEBCR

5.3.2. Phase de formation de la chaine

Dans cette phase, la SB va organiser les CHs de sorte qu'ils forment une chaîne qui se termine avec le CH le plus proche à la SB.

Parmi l'ensemble de toutes les chaines possibles, celle ayant l'énergie minimum de communication est choisie. Le coût d'une chaine est calculé par l'expression suivante:

$$Cout(Chaine_c) = \frac{\sum_{i=1}^{T_c-1} E_T\,(i)}{E_R}$$

Où T_c est la taille de la chaine c, $E_T(i)$ est l'énergie de transmission du nœud i et E_R est est l'énergie de reception.

5.3.3. Phase de transmission

Basé sur l'approche TDMA, Chaque CH établit et diffuse un ordonnancement affectant à chaque membre du groupe un temps durant lequel il peut communiquer les données captés au CH.

Une fois l'ordonnancement est connu par tous les nœuds du même cluster, chacun d'eux utilise le temps qui lui a été consacré pour transmettre au CH les données captées. Le CH applique, par la suite, un processus d'agrégation aux données reçues.

Puis, chacun des CHs envoie ses données à son voisin qui va, par la suite les agréger avec les siennes et ce jusqu'à ce que toutes les données soient arrivées à la SB.

A la fin de chaque cycle, tous les nœuds de capteurs envoient des informations de leurs énergies résiduelles à la SB. La figure **5.3** ci-dessous représente la phase de transmission:

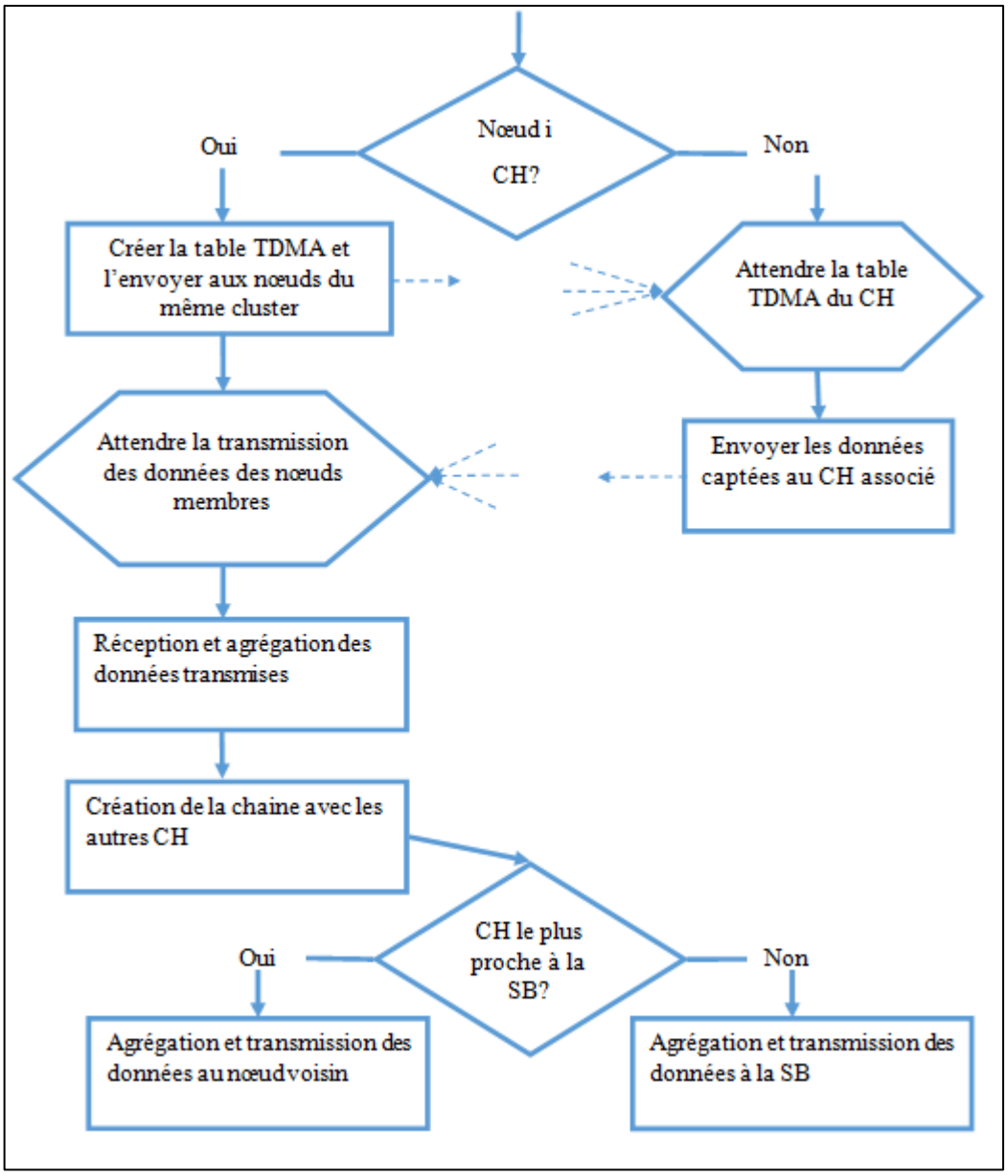

Figure 5.3 : Organigramme de la phase de transmission des données

5.4. Implémentation du protocole DEBRC

5.4.1. Choix du langage de programmation

Le langage de programmation utilisé pour l'implémentation est Matlab version 7.8.0 (R2009a).

MATLAB (« *matrix laboratory* ») est un langage de programmation de quatrième génération émulé par un environnement de développement du même nom ; il est utilisé à des fins de calcul numérique. Développé par la société The MathWorks, MATLAB permet de

manipuler des matrices, d'afficher des courbes et des données, de mettre en œuvre des algorithmes, de créer des interfaces utilisateurs, et peut s'interfacer avec d'autres langages comme le C, C++, Java, et Fortran. Les utilisateurs de MATLAB sont de milieux très différents (comme l'ingénierie, les sciences,…). Matlab peut s'utiliser seul ou bien avec des *toolboxes* (« boîte à outils »).

Le tableau **5.1** résume les caractéristiques de la machine utilisée.

Tableau 5.1 : caractéristiques matérielles

Hardware	Caractéristique
Processeur	Intel (R) Core(TM) i3-4005U CPU @ 1.70GHZ 1.70 GHz
Mémoire (RAM)	4.00 GO
Système d'exploitation	Microsoft Windows7 64 bits

5.4.2. Description de l'interface de simulation

Le simulateur est constitué de 3 parties principales : une partie de paramètres de simulation à gauche, une partie de zone de simulation au milieu et une partie de résultats de simulation à droite.

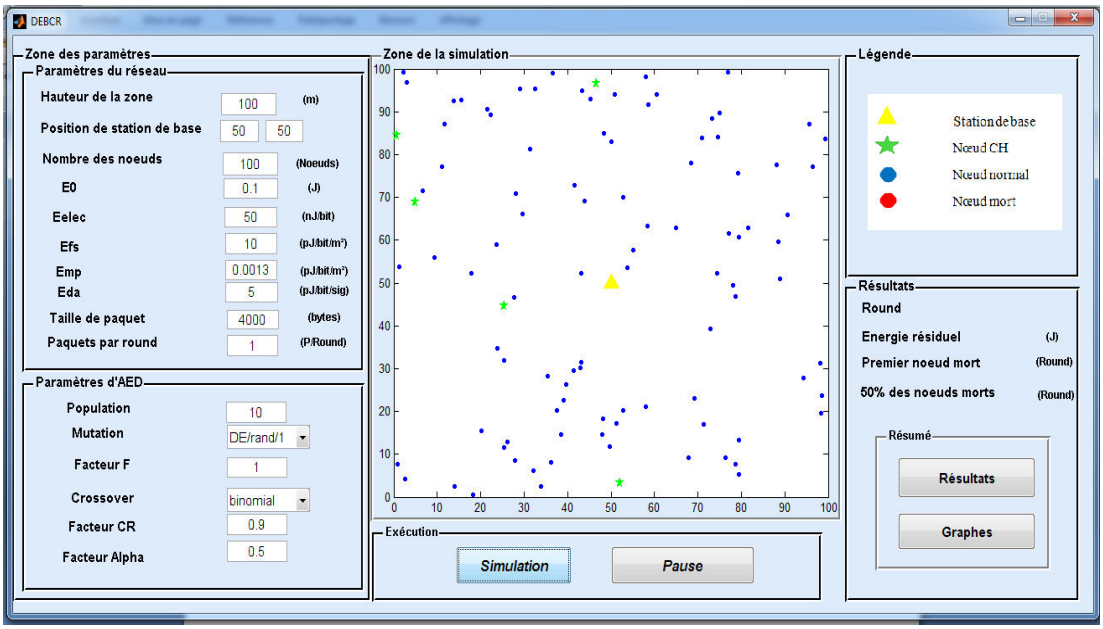

Figure 5.4 : L'interface « Simulateur DEBCR »

a. Partie de paramètres de simulation

Cette interface permet à l'utilisateur de saisir tous les paramètres de simulation, elle est divisé en deux partie : la première partie est réservée aux paramètres du réseau qui sont : les dimensions de la zone de simulation, la position de SB, le nombre des nœuds à distribuer dans la zone, les paramètres de modèle d'énergies choisi dans cet étude (l'énergie initiale Eo, Eelec, Efs, Emp, Eda), la taille de paquet et le nombre de paquets à transmettre par une seule itération.

La deuxième partie est réservée aux paramètres d'algorithme de DEBCR qui sont : la taille de population, le type de mutation, le type de croisement, les facteurs de mutation et de croisement, et enfin, le facteur de la fonction de fitness.

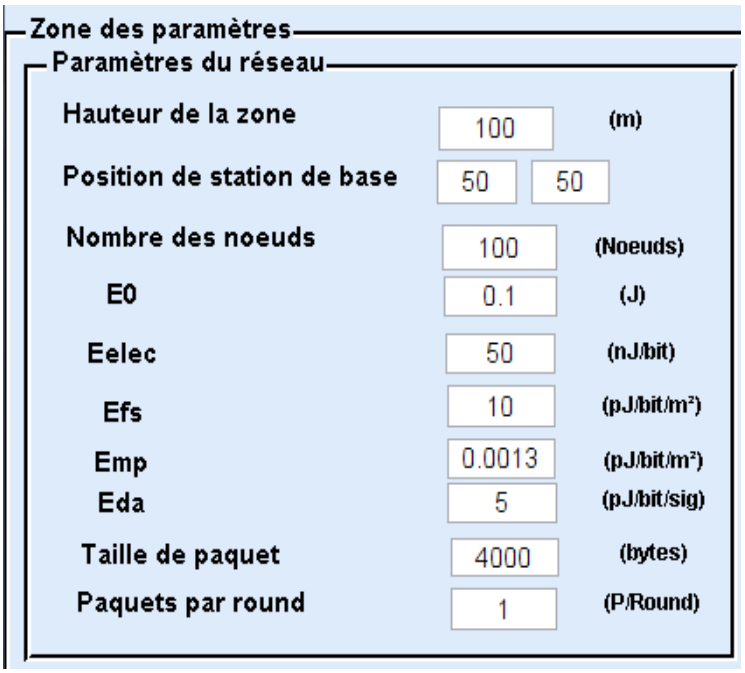

Figure 5.5 : Paramètres du réseau

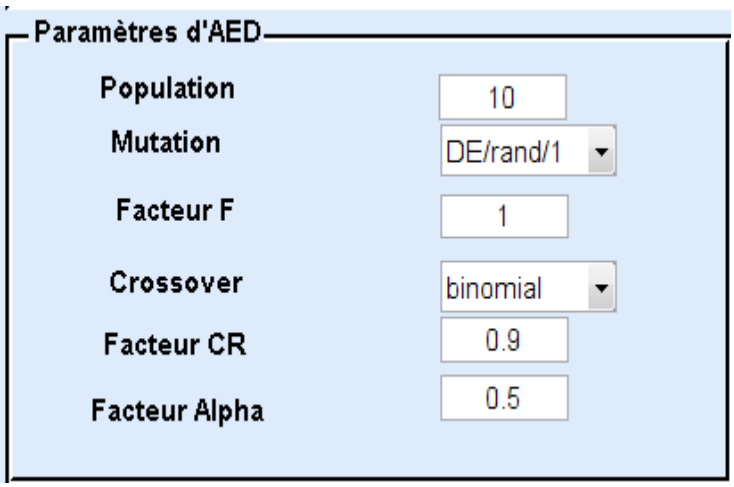

Figure 5.6 : Paramètres de l'AED

b. Partie de zone de simulation

Cette zone contient 3 parties :

- La première partie représente la zone de simulation.

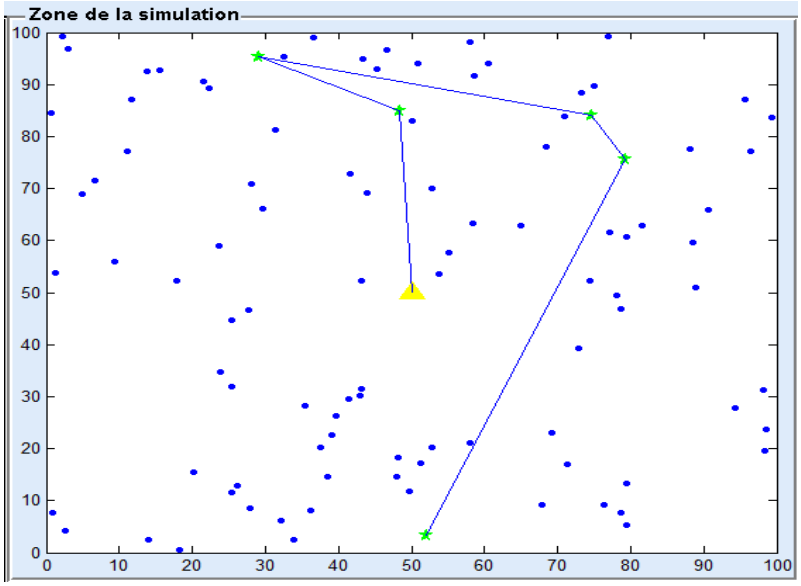

Figure 5.7 : Zone de simulation

- La deuxième partie représente la légende.

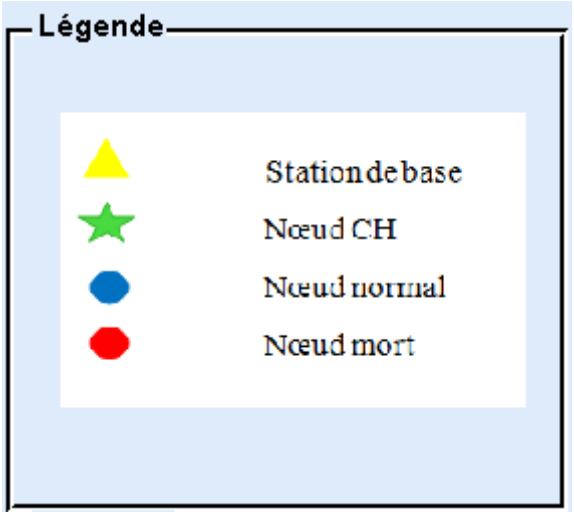

Figure 5.8 : Légende

- La troisième partie se compose d'un bouton "Simulation" pour lancer la simulation et un bouton "Pause " pour mettre en pause ou reprend la simulation.

Figure 5.9 : Boutons d'exécution

c. Partie de résultats de simulation

Cette partie se compose de :

- Résultats courants de la simulation : l'itération courante (Round), l'énergie résiduelle, Premier nœud Mort et 50 % des nœuds Morts.
- Un bouton "Résultats" qui sert à résumer les résultats sous forme d'un fichier excel.
- Un bouton "Graphes" qui sert à résumer les résultats sous forme des graphes.

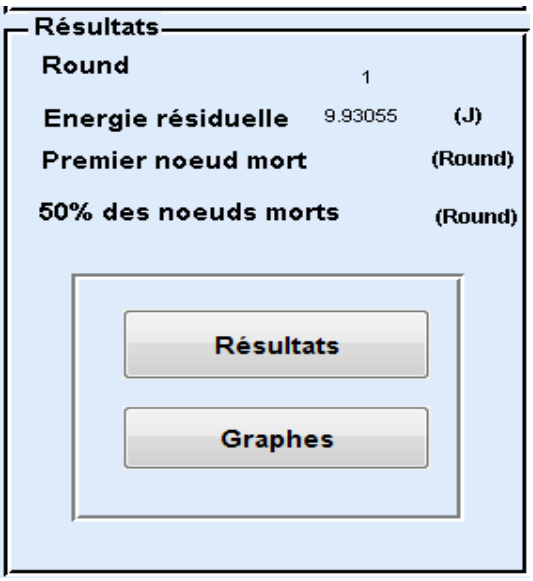

Figure 5.10 : Zone des résultats

5.4.3. Résultats et discussion

Afin de mieux évaluer la performance de notre travail, nous avons implémenté notre protocole en utilisant 4 méthodes de l'AED. Les méthodes sont illustrées dans le tableau suivant:

Tableau 5.2: Méthodes utilisées de l'AED

Méthode de l'AED utilisé	*Type de mutation*	*Type de croisement*
DE-1	Rand/1	Binomial
DE-2	Rand/1	Exponentiel
DE-3	Best/1	Binomial
DE-4	Best/1	Exponentiel

5.4.3.1. Les paramètres de simulation

Le tableau **5.3** représente les paramètres utilisés dans la simulation et leurs contextes

Tableau 5.3: Paramètres de simulation

Paramètre	***Contexte***
Zone de deployment	100m x 100m
Nombre des nœuds capteurs	100 nœuds
Position de la SB	50 x 50
Déploiement des nœuds	Aléatoire
Taille d'un paquet de donnée	4000 Bytes
Eo	0.1 J
Eelec	50 nJ/bit
Efs	10 pJ/bit/m²
Emp	0.0013 pJ/bit/m²
Eda	5 nJ/bit/sig
Taille de population	10
Facteur de mutation (F)	1
Facteur de croisement (CR)	0.9
Facteur de fonction de fitness (α)	0.5

5.4.3.2. Métriques d'évaluation

Afin d'évaluer notre protocole, nous avons considéré les métriques suivants:

a. ***La durée de vie du réseau:*** Dans notre protocole, nous avons utilisé la durée de vie du réseau en termes de nombre des cycles depuis le lancement du protocole jusqu'à le cycle ou:

- le premier nœud mort

- 50 % des nœuds morts
- le dernier nœud mort

b. Nombre des nœuds morts et nœuds vivants dans chaque cycle

c. Energie résiduelle

5.4.3.3. Résultats

Le tableau présenté ci-dessous résume la moyenne des résultats d'un nombre de simulations (5 fois) effectué sur les méthodes utilisées dans notre protocole :

Tableau 5.4: Résultats des simulations

	Premier nœud	50 % des nœuds	Dernier nœud
DE-1	69,6	86	120,2
DE- 2	65,6	85	118,8
DE-3	65,8	79,2	112,6
DE-4	68	82,8	118,6

5.4.3.4. Analyse des résultats

Dans ce qui suit, nous présentons et analysons les résultats des simulations en termes de: Durée de vie du réseau, Nombre des nœuds morts, Nombre des nœuds vivants, énergie résiduelle par tour.

a. Durée de vie du réseau:

La figure **5.11** représente la comparaison entre les méthodes en termes de la durée de vie du réseau (Premier nœud mort, 50 % des nœuds morts et dernier nœud mort).

Nous remarquons que, pour DE-1, le premier nœud meurt au tour 69ème, 50 % des nœuds sont morts au tour 86^{ème}et le dernier nœud meurt au tour 120ème.

Pour DE-2, le premier nœud meurt au tour 65ème, 50 % des nœuds sont morts au tour 85^{ème}et le dernier nœud meurt au tour 119ème.

Pour DE-2, le premier nœud meurt au tour 66ème, 50 % des nœuds sont morts au tour 79^{ème}et le dernier nœud meurt au tour 113ème.

Pour DE-2, le premier nœud meurt au tour 68ème, 50 % des nœuds sont morts au tour 83^{ème}et le dernier nœud meurt au tour 119ème.

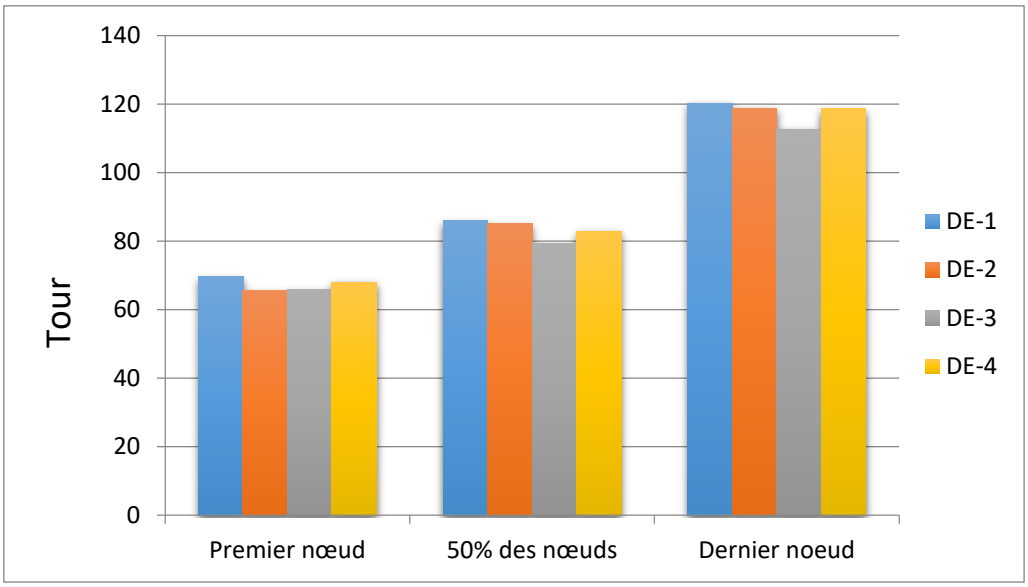

Figure 5.11: Comparaison en termes de la durée de vie

b. Nombre des nœuds morts et nœuds vivants

La figure **5.12** représente la comparaison entre les méthodes en termes de nombre des nœuds morts par tour. Constatons que le taux du mort des nœuds pour DE-1 et DE-2 est inférieur par rapport à la méthode DE-4 et cette dernière est mieux que la méthode DE-3 en termes de taux du mort des nœuds.

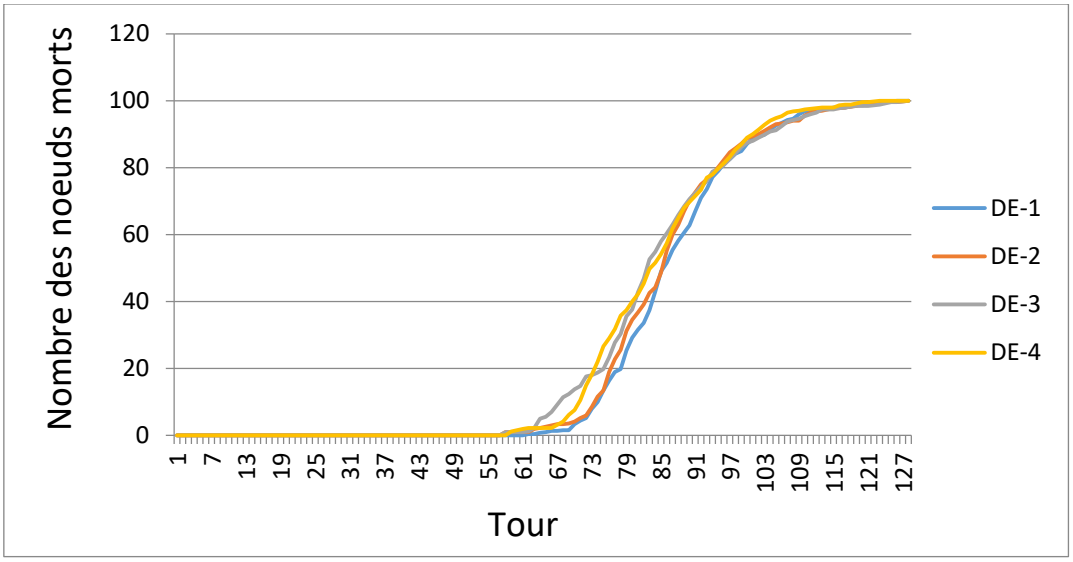

Figure 5.12: Comparaison en termes de nombre des nœuds morts par tour

La figure **5.13** représente la comparaison entre les méthodes en termes de nombre des nœuds vivants par tour.

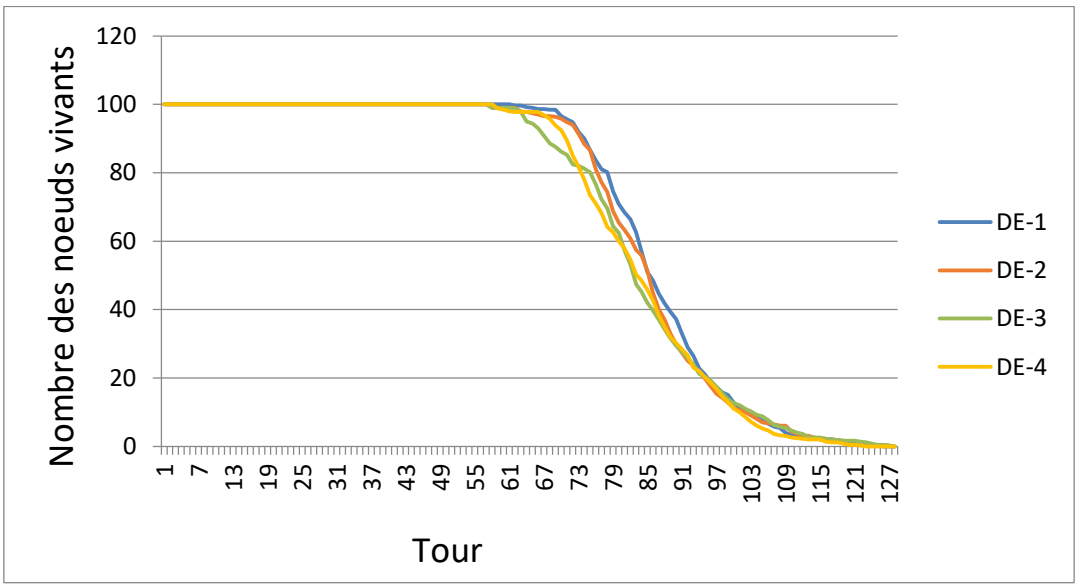

Figure 5.13: Comparaison en termes de nombre des nœuds vivants par tour

c. Energie résiduelle

La figure **5.14** représente la comparaison entre les méthodes en termes de l'énergie résiduelle par tour. Nous constatons que les résultats sont très proches pour toutes les méthodes en termes du taux de consommation d'énergie. Notons que la méthode DE-1 est mieux que les autres méthodes.

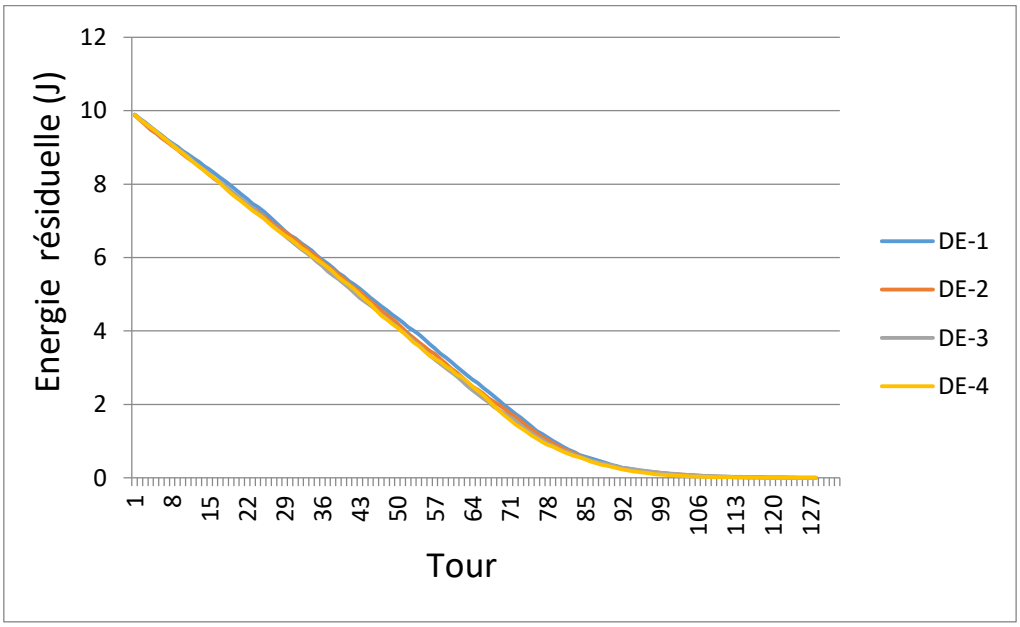

Figure 5.14: Comparaison en termes de l'énergie résiduelle

5.5. Conclusion

Dans ce chapitre, nous avons implémenté un protocole de routage basé sur l'AED en utilisant 4 méthodes, l'analyse des résultats de simulation nous a permis de découvrir que certaines d'entre elles sont mieux que les autres.

Conclusion générale et perspectives

Dans les dernières années, le domaine des réseaux de capteurs sans fil connaissait un grand intérêt grâce à ses diverses utilisations.

Cependant, les RCSFs ont posé plusieurs problèmes qu'il est nécessaire de les résoudre afin d'exploiter de façon mieux ses performances.

La conservation d'énergie est le plus grand défi dans les RCSFs. Beaucoup de recherches sont mises en œuvre dans le but d'améliorer la durée de vie du réseau.

Dans notre contribution, nous avons bénéficié les différentes variantes de l'AED pour implémenter un protocole de routage avec 4 méthodes afin d'augmenter les performances énergétique du réseau. La comparaison entre les méthodes nous a aidés à découvrir les méthodes les plus performantes parmi elles.

Enfin, dans une perspective d'amélioration des performances de notre protocole, nous allons essayer d'implémenter un protocole prenant en compte de:

- Améliorer les méthodes implémentées.
- Utiliser d'autres variantes de l'AED.
- Créer une fonction objective plus efficace.
- Hybrider l'AED avec une autre métaheuristique.

Table des matières

Résumé I

Remerciements III

Dédicaces IV

Table des matières VI

Liste des figures XIII

Liste des tableaux XV

Abréviations XVI

Introduction générale 1

Chapitre 1 : Réseaux de capteurs sans fil: Etat de l'art

1.1. Introduction 4

1.2. Capteur 4

1.2.1. Définitions 4

1.2.2. Architecture d'un capteur 5

1.2.2.1. Architecture matérielle 5

1.2.2.2. Architecture logicielle (Systèmes d'exploitation) 7

1.2.3. Classification des capteurs 10

1.2.3.1. Les capteurs Actifs 10

1.2.3.2. Les capteurs passive 11

1.2.4 Types des capteurs 13

1.2.4.1. Mica2Dot 15

1.2.4.2. MicaZ 15

1.2.4.3. Tmote Sky 16

1.3. Réseaux de capteurs sans fil 18

1.3.1. Définition 18

1.3.2. Architecture d'un RCSF 18

1.3.3. Caractéristiques d'un RCSF 19

1.3.4. Contraintes de conception des RCSF 20

1.3.4.1. La tolérance aux fautes 20

1.3.4.2. Le facteur d'échelle 21

1.3.4.3. Les coûts de production 21

1.3.4.4. L'environnement 21

1.3.4.5. La topologie de réseau 21

1.3.4.6. Les contraintes matérielles 21

1.3.4.7. Les médias de transmission 21

1.3.4.8. La consommation d'énergie 21

1.4. Pile protocolaire 21

1.4.1. Rôle des couches 23

1.4.2. Plans de gestion 24

1.5. Applications des RCSF 25

1.5.1. Découverte de catastrophes naturelles 25

1.5.2. Détection d'intrusions 25

1.5.3. Surveillance de l'environnement 26

1.5.4. Surveillance de la santé 26

1.5.5. Agriculture 27

1.5.6. Contrôle du trafic routier 28

1.6. Conclusion 29

Chapitre 2 : Techniques de conservation d'énergie dans les RCSFs

2.1. Introduction 31

2.2. Définition de durée de vie de réseaux 31

2.2.1. Durée de vie basée sur le nombre de nœuds vivants 31

2.2.2. Durée de vie basée sur la connectivité 32

2.2.3. Durée de vie basée sur la couverture 33

2.2.4. Durée de vie basée sur des les exigences de QoS 33

2.3. Consommation d'énergie dans les RCSF 33

2.3.1. Energie de capture 34

2.3.2. Energie de traitement 34

2.3.3. Energie de communication 34

2.4. Facteurs intervenants dans la consommation d'énergie 34

2.4.1. Etat du module radio 34

2.4.2. Accès au medium de transmission(MAC) 35

2.4.2.1. Retransmission 35

2.4.2.2. Ecoute active 35

2.4.2.3. Surécoute 35

2.4.2.4. Surcharge 36

2.4.4.5. Surémission 36

2.4.4.6. Taille des paquets 36

2.4.4.7. Modèle de propagation radio 37

2.5. Méthodes de conservation d'énergie dans les RCSF 37

2.5.1. Techniques du Duty-cycling 37

2.5.1.1. Protocole de contrôle de topologie 38

2.5.1.2. Protocoles de gestion d'énergie 38

a. Protocoles sleep/wakeup 38

b. Protocoles du niveau MAC 39

2.5.2. Approches orientées donnée 42

2.5.2.1. Technique de Réduire les données 42

2.5.2.2. Acquisition de données efficace en énergie 43

2.5.3. Mobilité 44

2.6. Comparaison des techniques de conservation énergétique 45

2.7. Conclusion 47

Chapitre 3 : Le routage dans les RCSFs

3.1. Introduction 49

3.2. Définition de routage 49

3.3. Classification des protocoles de routage 49

3.3.1. Classification selon la topologie du réseau 50

3.3.1.1. Les protocoles de routage à plat 50

3.3.1.2. Les protocoles hiérarchiques 51

3.3.2. Classification selon le fonctionnement du protocole 51

3.3.2.1. Routage basé sur la QoS 51

3.3.2.2. Routage basé sur la négociation 52

3.3.2.3. Routage basé sur les requêtes 52

3.3.2.4. Routage basé sur les multi-chemins 52

3.3.3. Classification selon le paradigme de communication 52

3.3.3.1. Centré-nœuds (Node-centric) 53

3.3.3.2. Centré-données (Data-centric) 53

3.3.3.3. Basé-localisation (location-based) 53

3.3.4. Classification selon l'établissement de la route 53

3.3.4.1. Protocoles proactifs 53

3.3.4.2. Protocoles réactifs 54

3.3.4.3. Protocoles hybrides 54

3.3.5. Classification selon l'initiateur de communication 54

3.3.5.1. Communication lancée par la source 55

3.3.5.2. Communication lancée par la destination..... 55

3.4. Exemples des protocoles de routage 55

3.4.1. Low Energy Adaptive Clustering Hierarchy (LEACH) 55

3.4.2. Power-Efficient GAthering in Sensor Information Systems (PEGASIS) 56

3.4.3. Threshold sensitive Energy Efficient sensor Network protocol (TEEN et APTEEN)..... 56

3.4.4. Le protocole de routage (MECN) 57

3.4.5. Sensor Protocol for information via negociation (SPIN) 57

3.4.6. Le protocole de routage (GAF) 58

3.4.7. La diffusion dirigée 58

3.5. Les protocoles hiérarchiques 59

3.5.1. L'approche basée sur le cluster 59

3.5.1.1. LEACH..... 61

3.5.1.2. Low Energy Adaptive Clustering Hierarchy Centralized (LEACH-C) ... 63

3.5.1.3. Fixed number of LEACH (LEACH-F) 64

3.5.1.4. Two level Low Energy Adaptive Clustering Hierarchy (TL-LEACH).... 64

3.5.1.5. Improved Low Energy Adaptive Clustering Hierarchy (I-LEACH)..... 65

3.5.1.6. Cell Low Energy Adaptive Clustering Hierarchy (Cell-LEACH) 65

3.5.1.7. Multi-hop Low Energy Adaptive Clustering Hierarchy (MH-LEACH).. 65

3.5.1.8. Comparaison entre LEACH et ses variantes 66

3.5.2. L'approche basée sur la chaine 67

3.5.2.1. PEGASIS 67

3.5.2.2. H-PEGASIS (Hierarchical PEGASIS) 69

3.2.2.3.Energy Balancing PEGASIS (EB-PEGASIS) 70

3.6. Conclusion 70

Chapitre 4: AED et travaux liés

4.1. Introduction 72

4.2. Métaheuristiques..... 72

4.3. L'algorithme à évolution différentielle 72

4.3.1. Définition 72

4.3.2. Principe 73

4.3.2.1. Mutation différentielle..... 74

4.3.2.2. Croisement différentiel..... 75

4.3.2.3. Sélection 75

4.3.2.4. Critère d'arrêt 76

4.3.3. Exemple 77

4.3.4. Variantes pour l'AED 77

4.4. Travaux liés 78

4.4.1. LEACH Protocol Improvement Based on the Differential Evolution Algorithm (LEACH_DE) 78

4.4.2. Differential Evolution based Clustering Algorithm (DECA) 81

4.4.3. Hybrid differential evolution and simulated anneling (DESA) 85

4.4.4. Switching-Differential Evolution(S-DE) for Cluster Head Election in Wireless Sensor Network 86

4.4.5. Comparaison entre les travaux 87
4.4.5.1. Comparaison suivant les paramètres de l'AED 88
4.4.5.2. Comparaison suivant autres paramètres 89
4.5. Conclusion 89

Chapitre 5 : Contribution

5.1. Introduction 91
5.2. Modèle du système 91
5.2.1. Modèle d'énergie 91
5.2.1. Modèle du réseau 92
5.3. Description du protocole proposé DEBRC 92
5.3.1. Phase de clustering 92
5.3.2. Phase de transmission 96
5.4. Implémentation du protocole DEBRC 97
5.4.1. Choix du langage de programmation 98
5.4.2. Description de l'interface de simulation 98
5.4.3. Résultats et discussion 101
5.4.3.1. Les paramètres de simulation 102
5.4.3.2. Métriques d'évaluation 102
5.4.3.3. Résultats 103
5.4.3.4. Analyse des résultats 103
5.5. Conclusion 106
Conclusion générale et perspectives 107
Bibliographique 108

Liste des figures

Figure 1.1 : Schéma interne d'un Capteur sans fil. 4

Figure 1.2 : Architecture d'un nœud de capteur. 5

Figure1.3 : Principe de fonctionnement d'un dispositif RFID 12

Figure 1.4: Principe de fonctionnement d'un capteur SAW 13

Figure 1.5 : Quelques exemples de capteurs. 14

Figure 1.6 : Capteur Mica2Dot 15

Figure 1.7 : Capteur MicaZ 15

Figure 1.8 : Capteur Tmote Sky. 16

Figure 1.9 : Réseau de capteurs sans fil 18

Figure 1.10: Architecture générale d'un réseau de capteurs sans fil..... 19

Figure 1.11 : La pile protocolaire des RCSF 23

Figure 1 .12 : Découverte de catastrophes naturelles par des RCSFs 25

Figure 1.13 : Application des capteurs dans la détection d'intrusions 26

Figure 1.14 : Application des RCSFs dans la surveillance de la santé. 27

Figure 1.15 : Application des RCSFs dans l'Agriculture 28

Figure 1.16 : Application des RCSFs pour la Contrôle du trafic routier 28

Figure 2.1: La surécoute dans une transmission 36

Figure 2.2 : Les méthodes de conservation d'énergie 37

Figure 2.3 : les techniques utilisées pour minimiser la consommation d'énergie 46

Figure 3.1: Classification des protocoles de routage 50

Figure 3.2 : Routage plat 51

Figure 3.3 : Le routage hiérarchique 59

Figure 3.4 : Organigramme de l'algorithme de formation de clusters distribué pour LEACH. 61

Figure 3.5 : Chronologie montrant le fonctionnement de LEACH. 61

Figure 3.6 : Algorithme de routage LEACH 63

Figure 3.7 : Protocole PEGASIS : La construction de la chaîne 68

Figure 4.1 : Mutation différentielle 74

Figure 4.2: Croisement différentiel : binomial à gauche, exponentiel à droite 75

Figure 4.3 : Un réseau de capteur sans fil avec des passerelles 83

Figure 4.4 : Affectation finale 87

Figure 5.1 : Le modèle de dissipation de l'énergie 92

Figure 5.2 : Organigramme du protocole DEBCR 95

Figure 5.3 : Organigramme de la phase de transmission des données 97

Figure 5.4 : L'interface « Simulateur DEBCR » 98

Figure 5.5 : Paramètres du réseau 99

Figure 5.6 : Paramètres de l'AED 99

Figure 5.7 : Zone de simulation 100

Figure 5.8 : Légende 100

Figure 5.9 : Boutons d'exécution. 100

Figure 5.10 : Zone des résultats 101

Figure 5.11: Comparaison en termes de la durée de vie 104

Figure 5.12: Comparaison en termes de nombre des noeuds morts par tour 105

Figure 5.13: Comparaison en termes de nombre des noeuds vivants pour chaque tour 105

Figure 5.14 : Comparaison en termes de l'énergie résiduel 105

Liste des tableaux

Tableau 1.1 : Comparaison entre les caractéristiques de quelques systèmes d'exploitation ... 7

Tableau 1.2. : Comparaison entre les caractéristiques de quelques capteurs 17

Tableaux 2.1: Comparaison entre les techniques d'accès au médium dans les RCSFs 41

Tableau 2.2 *:* Comparaison entre quelque approches de conservation d'énergie dans les RCSFs. 41

Tableau 3.1 : Comparaison entre LEACH et ses variantes. 66

Tableau 4.1: Affectation des nœuds de capteur à partir du vecteur de codage 83

Tableau 4.2 : Comparaison suivant les paramètres de l'AED 88

Tableau 4.3 : Comparaison suivant autres paramètres de l'AED 89

Tableau 5.1 : caractéristiques matérielles 98

Tableau 5.2 : Méthodes utilisées de l 'AED 101

Tableau 5.3: Paramètres de simulation 102

Tableau 5.4: Résultats des simulations 103