

République Algérienne Démocratique et Populaire  
Ministère de L'enseignement Supérieur et de la Recherche Scientifique

Université Abderrahmane Mira de Béjaia

Faculté des Sciences Exactes

Département d'Informatique

## Mémoire de Master Recherche

En Informatique

Option

Réseaux et Systèmes Distribués

## Thème

---

# Routage hiérarchique multi-sauts dans les réseaux de capteurs sans fil à basse consommation d'énergie

---

*Réalisé par :*

BOURIDAH Nawel

SIOUANI Amazigh

Soutenu le 26 Juin 2016 devant le jury composé de :

<i>Président :</i>	M. LARBI Ali	M.A.A, Université A.Mira de Béjaia
<i>Examineur :</i>	M. ABBACHE Bournane	M.A.A, Université A.Mira de Béjaia
<i>Examineur :</i>	M. KHANOUCHE Med-Essaid	M.A.A, Université A.Mira de Béjaia
<i>Encadreur :</i>	M. Mir Foudil	M.A.A, Université A.Mira de Béjaia

Année universitaire 2015/2016

# Table des matières

Table des figures	iv
Liste des tableaux	vi
Liste des abréviations	vii
Introduction générale	1
<b>1 Généralités sur les réseaux de capteurs sans fil</b>	<b>3</b>
1.1 Introduction . . . . .	3
1.2 Capteurs . . . . .	4
1.2.1 Définition d'un capteur . . . . .	4
1.2.2 Architecture d'un capteur . . . . .	4
1.2.3 Types de capteurs . . . . .	6
1.3 Réseaux de capteurs sans fil . . . . .	7
1.3.1 Description des réseaux de capteurs sans fil . . . . .	7
1.3.2 Architecture de communication d'un Réseau de capteurs sans fil . . . . .	7
1.3.3 Communications dans les Réseaux de capteurs sans fil . . . . .	8
1.3.4 Domaines d'applications des Réseaux de capteurs sans fil . . . . .	11
1.3.5 Caractéristiques des réseaux de capteurs sans fil . . . . .	13
1.3.6 Facteurs et contraintes des réseaux de capteurs sans fil . . . . .	14
1.4 Conclusion . . . . .	17
<b>2 Etat de l'art des protocoles de routages dans les RCSFs</b>	<b>18</b>

2.1	Introduction . . . . .	18
2.2	Facteurs de conception des protocoles de routage dans les Réseaux de capteurs sans fil	19
2.3	Approches de routage dans les réseaux de capteurs sans fil . . . . .	20
2.4	Classification des protocoles de routage . . . . .	21
2.4.1	Selon la structure du réseau . . . . .	21
2.4.2	Selon le mode de fonctionnement du protocole . . . . .	44
2.4.3	Selon l'établissement de routes . . . . .	52
2.4.4	Selon l'initiateur de communication . . . . .	53
2.5	Conclusion . . . . .	54
<b>3</b>	<b>Notre contribution : Routage avec efficacité énergétique basé sur une arborescence selon les nœuds frontières de la portée dans les R2E-ANFP)</b>	<b>55</b>
3.1	Introduction . . . . .	55
3.2	Description de la proposition . . . . .	56
3.3	Cas particuliers de notre protocole . . . . .	63
3.4	Conclusion . . . . .	65
<b>4</b>	<b>Evaluation des performances</b>	<b>67</b>
4.1	Introduction . . . . .	67
4.2	Outils et langages de développement . . . . .	67
4.2.1	JAVA . . . . .	68
4.2.2	NetBeans . . . . .	68
4.3	Environnement de simulation . . . . .	69
4.4	Modèle de consommation énergétique . . . . .	74
4.4.1	Energie consommée durant chaque phase de notre protocole . . . . .	76
4.4.2	Paramètres utilisés pour la simulation . . . . .	78
4.5	Résultats de simulations . . . . .	78
4.5.1	Simulation comparative . . . . .	78
4.5.2	Scalabilité . . . . .	79
4.6	Conclusion . . . . .	91

Conclusion générale et Perspectives

92

Bibliographie

94

# Table des figures

1.1	Architecture d'un capteur [3] . . . . .	5
1.2	Architecture de communication d'un RCSF [3] . . . . .	8
1.3	Schéma de la pile protocolaire [19] . . . . .	9
1.4	Application militaires [14] . . . . .	11
1.5	Applications environnementales [14] . . . . .	12
1.6	Applications dans le domaine médical [14] . . . . .	12
2.1	Diagramme de classification des protocoles de routage [3] . . . . .	21
2.2	Illustration de construction de la chaine dans PEGASIS [2] . . . . .	29
2.3	Illustration de la hiérarchie du clustering du protocole TEEN [2] . . . . .	31
2.4	Architecture du réseau pour VGA [15] . . . . .	35
2.5	Construction d'une chaine possible dans PEDAP et PEDAP-PA [12] . . . . .	36
2.6	Un nœud source $B$ et nœud Sink $S$ [18] . . . . .	37
2.7	Phase de communication du protocole CBCCP [21] . . . . .	40
2.8	Illustration du protocole GAF [2] . . . . .	44
2.9	Illustration du protocole SPIN [14] . . . . .	47
2.10	Illustration du fonctionnement du protocole DD [24] . . . . .	50
2.11	Le plan de requête dans le nœud leader dans le protocole COUGAR [2] . . . . .	51
3.1	Organigramme de la phase d'élection du CH . . . . .	58
3.2	Phase d'élection du CH . . . . .	59
3.3	Organigramme de la phase de formation de clusters . . . . .	60
3.4	Phase de formation de Cluster . . . . .	61

3.5	Organigramme de la phase de communication de données . . . . .	62
3.6	Phase de communication de données . . . . .	63
3.7	Cas du noeud isolé . . . . .	64
3.8	Noeud isolé rejoint le cluster le plus proche . . . . .	64
3.9	Cas de la zone morte . . . . .	65
4.1	Fenêtre principale (accueil) . . . . .	69
4.2	Fenêtre "arbre de routage" . . . . .	70
4.3	Arbre de routage lors de l'exécution de notre protocole . . . . .	71
4.4	Fenêtre "Statistiques du protocole" . . . . .	72
4.5	Fenêtre "Simulation comparative" . . . . .	73
4.6	Fenêtre "Scalabilité" . . . . .	73
4.7	Fenêtre "Simulation *100 . . . . .	74
4.8	Modèle de consommation d'énergie [10] . . . . .	75
4.9	Graphe des résultats de simulation représentant "nombre de noeuds actifs dans le réseau en fonction des rounds" . . . . .	79
4.10	Tableau des résultats pour le premier noeud qui meurt . . . . .	81
4.11	Graphe du premier noeud qui meurt dans le réseau . . . . .	81
4.12	Tableau des résultats pour un pourcentage 25% de mortalité des noeuds . . . . .	83
4.13	Graphe "Pourcentage de mortalité des noeuds (25%)" . . . . .	83
4.14	Tableau des résultats pour pourcentage 50% de la mortalité des noeuds . . . . .	85
4.15	Graphe "Pourcentage de mortalité des noeuds (50%)" . . . . .	85
4.16	Tableau des résultats pour un pourcentage de 75% de la mortalité des noeuds . . . . .	86
4.17	Graphe "Pourcentage de mortalité des noeuds (75%)" . . . . .	86
4.18	Tableau des résultats du dernier noeud qui meurt . . . . .	87
4.19	Graphe "Pourcentage de mortalité de noeuds (100%)" . . . . .	88
4.20	Tableau de la consommation moyenne d'énergie par round . . . . .	89
4.21	Graphe de consommation moyenne d'énergie en fonction du nombre de noeuds dans le réseau par round . . . . .	89

# Liste des tableaux

1.1	Comparaison de quelques types de capteurs [16] . . . . .	6
2.1	Tableau comparatif des protocoles de routage de la topologie plat dans les RCSFs [18]	23
2.2	Tableau comparatif des protocoles de routage de la topologie hiérarchique dans les RCSFs [12, 21] . . . . .	42
2.3	Tableau comparatif des protocoles de routage de la topologie géographique dans les RCSFs [2] . . . . .	44
2.4	Tableau comparatif des protocoles de routage selon le mode de fonctionnement du protocole dans les RCSFs [18, 20] . . . . .	52
4.1	Paramètres de simulations . . . . .	78

# Liste des abréviations

<b>ACQUIRE</b>	<i>ACTive QUery forwarding In sensoR nEtworks</i>
<b>ADC</b>	<i>Analog/Digital Converter</i>
<b>ADV</b>	<i>ADVertisement</i>
<b>APTEEN</b>	<i>TheAdaptive Threshold sensitive Energy Efficien sensor Network</i>
<b>BCDCP</b>	<i>Base – Station Controlled Dynamic Clustering Protocol</i>
<b>BDD</b>	<i>Base de Données Distribuées</i>
<b>CBCCP</b>	<i>Chain Based Cluster Cooperative Protocol</i>
<b>CDMA</b>	<i>Code Division Multiple Access</i>
<b>CH</b>	<i>Cluster Head</i>
<b>CSMA</b>	<i>Carrier Sense Multiple Access</i>
<b>DD</b>	<i>Directed Diffusion</i>
<b>E-HEED</b>	<i>Extented – Hybrid Energy Efficient Distributed</i>
<b>GAF</b>	<i>Geographic Adaptive Fidelity</i>
<b>GBDD</b>	<i>Grid Based Data Dissemination</i>
<b>GBR</b>	<i>Gradient Based Routing</i>
<b>GEAR</b>	<i>Geographic and Energy – Aware Routing</i>
<b>Ghz</b>	<i>Giga hertz</i>
<b>GPS</b>	<i>Global Positioning System</i>
<b>HEED</b>	<i>Hybrid Energy Efficient Distributed</i>
<b>H-PEGASIS</b>	<i>Hierarchical Power – Efficient GAthering in Sensor Information Systems</i>
<b>HS</b>	<i>Hors Service</i>
<b>Ht</b>	<i>Hard Threshold</i>

<b>HTML</b>	<i>Hyper Text Markup Language</i>
<b>ID</b>	<i>IDentifiant</i>
<b>IDE</b>	<i>Integrated Development Environment</i>
<b>IP</b>	<i>Internet Protocol</i>
<b>ISM</b>	<i>Industrial Scientific Medical</i>
<b>J</b>	<i>Joule</i>
<b>Kbps</b>	<i>Kilobyte per second</i>
<b>Ko</b>	<i>Kilo octets</i>
<b>LA</b>	<i>Local Aggregator</i>
<b>LEACH</b>	<i>Low – Energy Adaptative Clustering Hierarchy</i>
<b>LEACH-C</b>	<i>Low – Energy Adaptative Clustering Hierarchy – Centralized</i>
<b>MA</b>	<i>Master Aggregator</i>
<b>MAC</b>	<i>Media Access Control</i>
<b>MANET</b>	<i>Mobile Ad hoc NETWORK</i>
<b>mAh</b>	<i>milliAmpère – heure</i>
<b>MECN</b>	<i>Minimum Energy Communication Network</i>
<b>MEMS</b>	<i>Micro Electro Mechanical Systems</i>
<b>MERR</b>	<i>Minimum Energy Relay Routing</i>
<b>MH-LEACH</b>	<i>Multi Hop Low – Energy Adaptative Clustering Hierarchy</i>
<b>Mhz</b>	<i>Méga hertz</i>
<b>MST</b>	<i>the Minimum Spanning Trees</i>
<b>PC</b>	<i>Personal Computer</i>
<b>PEDAP</b>	<i>Power Efficient Data gathering and Aggregation Protocol</i>
<b>PEDAP-PA</b>	<i>Power Efficient Data gathering and Aggregation Protocol – Power Aware</i>
<b>PEGASIS</b>	<i>Power – Efficient GATHERing in Sensor Information Systems</i>
<b>QOS</b>	<i>Quality Of Service</i>
<b>RAM</b>	<i>Random Access Memory</i>

<b>RCSF</b>	<i>Réseaux de Capteurs Sans Fil</i>
<b>REQ</b>	<i>REQuest</i>
<b>ROM</b>	<i>Read Only Memory</i>
<b>RR</b>	<i>Rumor Routing</i>
<b>RSSI</b>	<i>Received Signal Strength Indicator</i>
<b>SAR</b>	<i>Sequential Assignment Routing</i>
<b>SB</b>	<i>Station de Base</i>
<b>SHPER</b>	<i>Scaling Hierarchical Power Efficient Routing</i>
<b>SMECN</b>	<i>Small Minimum Energy Communication Network</i>
<b>SMP</b>	<i>Sensor Management Protocol</i>
<b>SPIN</b>	<i>Sensors Protocols for Information via Negotiation</i>
<b>SPIN-BC</b>	<i>SPIN for Broadcast Networks</i>
<b>SPIN-EC</b>	<i>SPIN with Energy Conservation</i>
<b>SPIN-PP</b>	<i>SPIN for Point to Point Communication</i>
<b>SPIN-RL</b>	<i>SPIN with Reliability</i>
<b>St</b>	<i>Soft Threshold</i>
<b>TADAP</b>	<i>Task Assignment and Data Advertisement Protocol</i>
<b>TDMA</b>	<i>Time Division Multiple Access</i>
<b>TEEN</b>	<i>Threshold sensitive Energy Efficient sensor Network</i>
<b>TTDD</b>	<i>Two – Tier Data Dissemination</i>
<b>UDP</b>	<i>User Datagram Protocol</i>
<b>UDP-Like</b>	<i>User Datagram Protocol – Like</i>
<b>USB</b>	<i>Universal Serial Bus</i>
<b>VGA</b>	<i>Virtual Grid Architecture Routing</i>
<b>WLAN</b>	<i>Wide Local Area Network</i>
<b>WSN</b>	<i>Wireless Sensor Network</i>

*\*\*\* Remerciements \*\*\**

Avant toutes choses, nous tenons à remercier Dieu le tout Puissant qui nous a donné la patience et la persévérance ainsi que le courage afin d'achever notre travail.

Nous exprimons nos plus sincères remerciements et gratitude à notre promoteur M. MIR Foudil qui a toujours été présent pour nous avec ses conseils et ses orientations qui nous ont guidés jusqu'à la finalisation de notre travail de recherche.

Nous adressons également nos remerciements, aux membres du jury qui ont accepté d'examiner et de juger ce modeste travail.

Nos sincères remerciements sont également exprimés à toutes les personnes qui ont contribué à la réalisation de ce mémoire, sans oublier tous nos amis du département Informatique.

*Mlle. BOURIDAH Nawel*

*M. SIOUANI Amazigh*

\*\*\* *Dédicaces* \*\*\*

Je dédie ce modeste travail :

Á mes très chers parents. Aucune dédicace n'est assez éloquente pour vous exprimer ma gratitude pour les sacrifices que vous avez consentis depuis 22ans afin de m'offrir tout ce dont j'avais besoin pour m'épanouir. La chance que j'ai de vous avoir est inestimable et j'espère que ce travail vous  
rendra fiers ;

Á mon très cher frère "Y". Merci d'être la pour moi, de me soutenir et surtout pour me faire rire ;

Á ma très chère Grand-Mère. Pour tes encouragements et pour ta présence à mes cotés ;

Á mes tantes, et mes oncles ;

Á mes cousins et cousines ;

Á tout mes amis.

*Mlle. BOURIDAH Nawel*

\*\*\* *Dédicaces* \*\*\*

Je dédie ce modeste travail :

Á celui qui garnis mes chemins avec force et lumière... mon très cher père ;

Á la plus belle perle du monde... ma tendre mère ;

Á mes sœurs, en leurs souhaitant toute le sucée... tout le bonheur ;

Á toute ma famille pour tout l'amour et le respect qui m'ont toujours accordé ;

Á ma binôme pour la partenaire agréable qu'elle était et qui le restera toujours ;

Á tout mes amis, pour la sincérité si merveilleuse... inoubliables, en leurs souhaitant tout le sucées

... tout le bonheur ;

Á toute personne qui m'a aidé a franchir un horizon dans ma vie... .

*M. SIOUANI Amazigh*

# Introduction Générale

La nécessité d'obtenir des informations, a tout moment sur l'environnement qui nous entoure, a conduit l'être humain à améliorer les moyens de communications et d'informations.

Les avancées technologiques lors de ces dernières décennies réalisés dans les domaines de la microélectronique et des technologies de la communication sans fil ont permis de concevoir et de fabriquer des composants miniaturisés, autonome et fiable tels que les capteurs. En effet, l'apparition des réseaux de capteurs sans fil a élargi considérablement les horizons d'utilisation des équipements de collecte, de traitement et de transmission de données.

D'autre part, les réseaux de capteurs sans fil (RCSF), appelé aussi en anglais (Wireless Sensor Network) qui représente un cas particulier des réseaux Ad hoc, ont vu le jour en combinant un nombre important de capteurs qui sont déployés sur une zone géographique vaste afin de collecter des informations sur des événements bien définis, et de les acheminer vers un nœud particulier de traitement, appelé nœud puits (sink). Ces capteurs sont utilisés dans plusieurs domaines à des fins aussi bien d'information, de contrôle ou pour la surveillance, ces domaines peuvent être (médical, militaire, environnementales, etc.).

Cependant, dans un RCSF un capteur est un équipement de taille très réduite, cet équipement est limité en terme de mémoire, de calcul ainsi que l'énergie, ces caractéristiques particulières impliquent de nouveaux défis lors de la conception d'un RCSF. Plusieurs contraintes ont été identifiées incluant la topologie dynamique, l'énergie limitée, etc.

Dans ce mémoire, nous allons nous intéresser aux problèmes relatifs au routage de données dans

les RCSFs. En effet, le routage de données collectées dans un environnement est une caractéristique primordiale pour le besoin du traitement des informations, et plus précisément le routage hiérarchique de données. Ce type de routage est très important car il est considéré comme un outil qui offre plus de performances en termes de consommation d'énergie par rapport aux autres types de routages, à savoir, le routage à topologie plate et le routage à topologie géographique [4].

Plusieurs solutions de routage qui représentent de nombreux protocoles de routage ont été proposées pour les réseaux de capteurs sans fil en littérature, chacun de ces protocoles présente une technique différente afin de minimiser la consommation d'énergie sur l'ensemble du réseau. Néanmoins, des améliorations restent à apporter pour obtenir de meilleurs performances.

Notre travail, consiste en une première phase d'étudier et d'analyser les protocoles de routage existants dans la littérature puis de dégager les avantages ainsi que les inconvénients de chacun des protocoles. Deuxièmement, l'objectif étant de concevoir un nouveau protocole hiérarchique dont le but est de garantir de meilleures performances en termes de réduction dans la consommation d'énergie afin d'assurer la longévité du réseau entier.

Les démarches citées ci-dessus nous ont amenés à organiser ce présent mémoire en quatre chapitres. Après cette introduction générale nous décrivons, dans le premier chapitre les généralités sur les réseaux de capteurs sans fil, le second chapitre est consacré à l'état de l'art sur les protocoles de routages étudiés où nous allons présenter et critiquer chaque protocole en présentant les avantages et inconvénients, le troisième chapitre est dédié à notre contribution dans les RCSFs, où nous proposerons notre protocole de routages hiérarchique et ses grandes phases, nous détaillerons par la suite le fonctionnement de notre protocole, le quatrième chapitre sera consacré à l'évaluation des performances où nous allons effectués des simulations de notre protocole afin d'évaluer ses performances. Enfin nous finalisons notre travail avec une conclusion générale qui synthétise tout le travail qui a été effectué.

# Chapitre 1

## Généralités sur les réseaux de capteurs sans fil

---

### 1.1 Introduction

Les progrès récents dans les technologies des systèmes micro-électro-mécanique (MEMS : Micro-Electro-Mechanical Systems) [3], et le domaine de l'électronique et les communications sans fil ont donné naissance à des composants peu coûteux, multifonctionnels et de basse puissance. Ces composants sont de petites tailles capables de collecter et de transmettre des données environnementales telles que (la température, humidité, etc.). Ces composants sont appelés des capteurs, ils ont la capacité de s'auto organiser pour former un réseau de capteurs sans fil.

Un réseau de capteurs sans fil (RCSF) ou WSN est un type particulier des réseaux Ad hoc MANET (Mobile Ad hoc Network). Notons que les RCSFs se composent d'un grand nombre de capteurs qui permettent le contrôle de l'environnement à distance, ils peuvent couvrir le réseau en se déployant soit de manière aléatoire ou de manière déterministe pour être utilisés dans diverses applications (militaires, médicales, etc.). Ces nœuds capteurs sont limités en termes de ressources énergétiques et sont exposés à des pannes accidentelles ou intentionnelles dans le réseau.

Chaque capteur est composé de plusieurs unités qui leurs permettent de communiquer entre eux, cette communication est établit selon une certaine topologie du réseau afin d'acheminer les informations

détectées jusqu'à la station de base, cette dernière est située à de grandes distances de la zone de déploiement des capteurs ce qui pose problème parmi tant d'autres. Dans cette optique, plusieurs solutions ont été proposées dans la littérature, qui vise à minimiser la consommation d'énergie afin de prolonger la durée de vie de tout le réseau.

## 1.2 Capteurs

### 1.2.1 Définition d'un capteur

Les capteurs sont des dispositifs électroniques utilisés dans les réseaux de capteurs sans fil, Ces dispositifs sont de taille extrêmement réduite avec des ressources très limitées, ils sont autonomes, les capteurs déployés sont capables de surveiller et de mesurer des valeurs physiques environnementales (température, lumière, pression, etc.) et de les communiquer à un centre de contrôle via une station de base. La détection de données environnementales est obtenue par la collaboration d'un grand nombre de capteurs qui sont utilisés pour la détection, le traitement de données, ils sont généralement de faibles puissance et sont capables de communiquer à de petites distances [3].

### 1.2.2 Architecture d'un capteur

L'architecture typique d'un capteur est représentée sur la figure 1.1, cette architecture comporte quatre (4) unités de base pour le fonctionnement du capteur : Une unité de détection, une unité de traitement, une unité d'émission/réception ainsi qu'une unité de puissance, chacune de ces unités sera détaillé dans ce qui suit [3] :

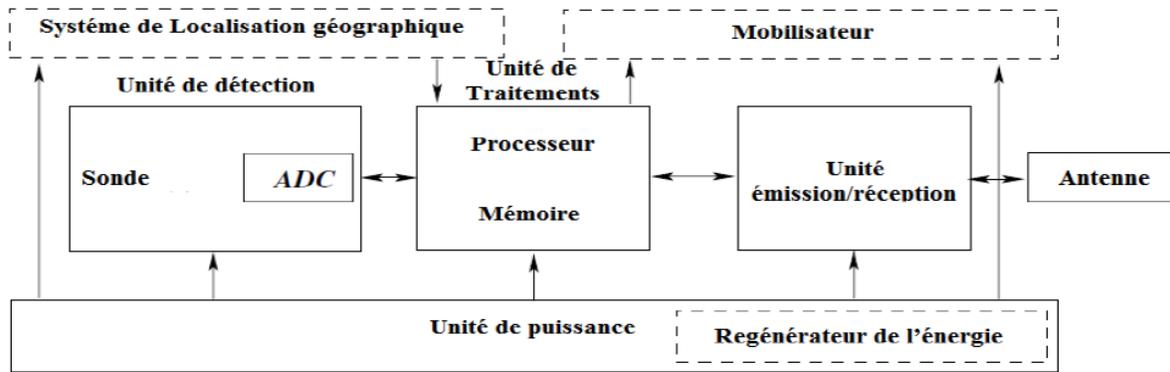


FIGURE 1.1 – Architecture d'un capteur [3]

- **L'unité de détection :**

C'est la composante principale du capteur, cette unité est généralement composée de deux sous-unités : une sonde et un convertisseur analogique-numérique (ADCs : Analog/ Digital Converter). Les capteurs obtiennent des mesures numériques sur les paramètres environnementaux et la sonde les transforme en signaux analogiques. Les ADCs convertissent ces signaux analogiques en signaux numériques puis sont introduits dans l'unité de traitement.

- **L'unité de traitement :**

Cette unité est composée de deux interfaces : une interface avec l'unité d'acquisition et une autre avec le module de transmission. Elle contrôle les procédures permettant à un nœud d'exécuter les opérations de détection et de collaborer avec les autres nœuds par la communication sans fil.

- **L'unité de communication (émission/réception) :**

Elle est responsable de toutes les communications (émission/réception) via un support de communication radio qui relie le nœud aux autres nœuds du réseau. Essentiellement, le RCSF est relié en réseau grâce à cette unité.

- **L'unité de puissance :**

Un des composants les plus importants d'un capteur est l'unité de puissance, Chaque composant dans le capteur est alimenté par l'unité de puissance, on en cite les unités ci-dessus (unité de détection, unité de traitement et unité de communication).

En plus des quatre unités, citées ci-dessus, un capteur peut également contenir suivant son domaine d'application, des éléments supplémentaires, à savoir un système de localisation géographique, un générateur de l'énergie et un mobilisateur.

### 1.2.3 Types de capteurs

Les types de capteurs varient selon l'application envisagée, le tableau 1.1 présente une comparaison entre certains types de capteurs les plus populaires conçus ces dernières années en termes de Type de microcontrôleur, RAM / ROM, Bande passante, Portée maximale, etc.

Nom / Image du capteur	Mica2 (Berkeley) 	Telos (Intel) 	Imote2 (Crossbow) 
Type de microcontrôleur	7.37 MHz, 8 bits	8 MHz, 8 bits	13- 416 MHz, 16 bits
RAM	4 Ko	2 Ko	256 Ko
ROM	512 Ko	256 Ko	32 Mo
Bande passante	38.4 Kbps	250 Kbps	
Capacité de la batterie	2xAA 5700mAh	Coin Cell 1000mAh	3xAAA 3750mAh
Portée maximale	150 – 300 m	100 m	
Radio	Chipcon CC1000 6 MHz 38.4 Kbauds	Chipcon CC2420 2.4 GHz 250 Kbps IEEE802.15.4	
Système d'exploitation	TinyOs		
Connexion au PC	Carte de programmation	Port série	Interface USB

TABLE 1.1 – Comparaison de quelques types de capteurs [16]

## 1.3 Réseaux de capteurs sans fil

### 1.3.1 Description des réseaux de capteurs sans fil

Un réseau de capteurs sans fil (RCSF), est un type particulier des réseaux ad hoc, il est composé de centaines ou de milliers d'éléments nommés capteurs placés de manière plus au moins aléatoire. Dans ce type de réseau chaque nœud est capable de surveiller son environnement et de réagir en cas de besoin en envoyant l'information détectée à un ou plusieurs point de collecte à l'aide d'une connexion sans fil.

Un RCSF est constitué essentiellement de : plusieurs nœuds capteurs, un nœud puits (Sink) et un centre de traitement des données [3].

### 1.3.2 Architecture de communication d'un Réseau de capteurs sans fil

Les nœuds capteurs sont habituellement dispersés dans une zone de capture. Chacun de ces nœuds a la possibilité de collecter les données et de les router vers une ou plusieurs stations de base (sink). Ce dernier est un point de collecte de données capturées. Il peut communiquer les données collectées à l'utilisateur final à travers un réseau de communication, éventuellement l'Internet. L'utilisateur peut à son tour utiliser la station de base comme passerelle, afin de transmettre ses requêtes au réseau [3]. Cette architecture est illustrée dans la figure 1.2.

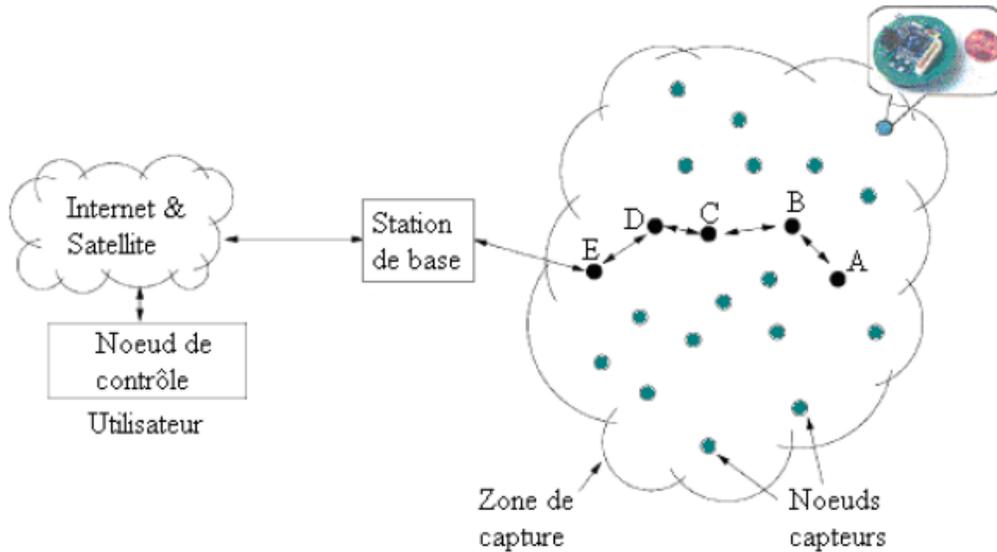


FIGURE 1.2 – Architecture de communication d'un RCSF [3]

### 1.3.3 Communications dans les Réseaux de capteurs sans fil

Dans les RCSFs, les nœuds sont déployés dans un environnement sans infrastructure, en n'ayant aucune information sur la topologie globale même locale du réseau. Pour cela, les nœuds capteurs doivent graduellement établir l'infrastructure de communication durant une phase d'initialisation. Cette infrastructure doit leur permettre de répondre aux requêtes venant des sites distants, d'interagir avec l'environnement physique, réagir aux données captées et transmettre ces données via une communication multi-sauts [3].

#### 1.3.3.1 Types de communications

Il existe deux types de communications dans les réseaux de capteurs sans fil, qui sont cités ci-dessous :

##### 1. Communication nœud capteur à un nœud capteur

Ce type de communication est utilisé localement par les nœuds capteurs, par exemple soit pendant le processus de clustering ou pour l'établissement des chemins (routes).

##### 2. Communication nœud capteur à un nœud intermédiaire

Dans ce cas, les données détectées sont transmises à partir d'un nœud capteur à un nœud intermédiaire, ce type de communication est souvent appelée communication Unicast.

### 1.3.3.2 Pile protocolaire

Dans le but d'un établissement efficace d'un RCSF, une architecture en couches est adoptée afin d'améliorer la robustesse du réseau. Une pile protocolaire est donc utilisée par les nœuds du réseau, elle comporte cinq (5) couches. Citons la couche application, la couche transport, la couche réseau, la couche liaison de données et la couche physique.

De plus cette pile possède 3 niveaux ou plans (Un plan de gestion d'énergie, un plan de gestion de mobilité et un plan de gestion des tâches) [19].

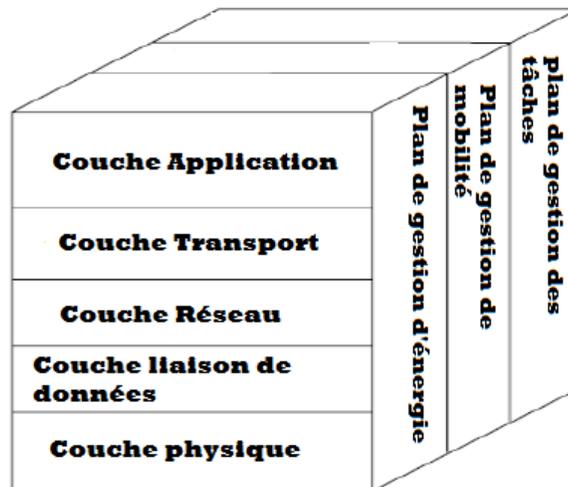


FIGURE 1.3 – Schéma de la pile protocolaire [19]

#### 1. La couche application :

Elle assure l'interface avec les applications. Il s'agit donc de la couche la plus proche des utilisateurs, gérée directement par les logiciels. Parmi les protocoles d'application, nous citons : SMP (Sensor Management Protocol) et TADAP (Task Assignment and Data Advertisement Protocol).

#### 2. La couche transport :

Elle vérifie le bon acheminement des données et la qualité de la transmission. Dans les RCSFs, la fiabilité de transmission n'est pas majeure. Ainsi, les erreurs et les pertes sont tolérées. Par conséquent, un protocole de transport proche du protocole UDP et appelé UDP-Like (User Datagram Protocol Like) est utilisé.

**3. La couche réseau :**

Elle s'occupe du routage de données fournies par la couche transport. Elle établit les routes entre les nœuds capteurs et le nœud puits et sélectionne le meilleur chemin en termes d'énergie, délai de transmission, débit, etc.

**4. La couche liaison de données :**

Elle est responsable d'assurer une communication fiable par des techniques de détection, et la correction d'erreurs intervenues sur la couche physique et l'accès canal. De plus, elle établit une communication saut-par-saut entre les nœuds. C'est-à-dire, elle détermine les liens de communication entre eux dans une distance d'un seul saut.

**5. La couche physique :**

Cette couche satisfait les besoins de modulation, de la transmission et de la réception des données dans le média physique tout en choisissant les bonnes fréquences.

**6. Un plan de gestion d'énergie :**

La durée de vie du nœud est indépendante de celle de la batterie, le plan de gestion d'énergie doit maximiser de l'énergie par exemple après réception d'un message le nœud doit se mettre en sommeil pour éviter la duplication de message reçus.

**7. Un plan de gestion de mobilité :**

La mobilité est l'un des caractéristiques d'un capteur, pour se localiser le plan de gestion de mobilité aide le capteur à enregistrer sa position qu'il soit mobile ou en mouvement dans le réseau.

**8. Un plan de gestion des tâches :**

Il assure l'équilibrage de la distribution des tâches sur les différents nœuds du réseau, afin d'assurer un travail efficace en matière de consommation d'énergie.

Ces plans de gestion sont nécessaires pour que les capteurs puissent fonctionner ensemble d'une manière « power-efficient », et de router les données dans un réseau mobile de capteurs et partager leurs ressources ; Sans eux, chaque capteur fonctionnera d'une façon isolée. Du point de vue du réseau entier, il est plus efficace si les nœuds peuvent collaborer les uns avec les autres, ainsi la durée de vie du réseau peut être prolongée [19].

### 1.3.4 Domaines d'applications des Réseaux de capteurs sans fil

Le concept de capteurs connectés via un support sans fil permet la réalisation de plusieurs types d'applications. Ces dernières peuvent être subdivisées en plusieurs domaines, entre autres le domaine militaire, environnement, santé, maisons ainsi que d'autres applications commerciales.

#### 1.3.4.1 Applications militaires

Dans ce domaine, les capteurs sont utilisés pour [3] :

- Contrôle des forces, équipements et munitions ;
- Reconnaissance et surveillance du champ de bataille ;
- Reconnaissance des forces opposées et le terrain ;
- Evaluation et estimation des dommages des combats ;
- Détection et reconnaissance des attaques nucléaires, biologiques et chimiques.



FIGURE 1.4 – Application militaires [14]

#### 1.3.4.2 Applications dans l'environnement

Diverses, sont les applications d'environnement qui peuvent bénéficier de la technologie des réseaux de capteurs sans fil, on peut citer par exemple [3] :

- Suivie des mouvements d'oiseaux, des petits animaux et des insectes dans leurs environnements ;
- Détection des incendies dans les forêts ;
- Détection des inondations ;

- Agriculture.



FIGURE 1.5 – Applications environnementales [14]

### 1.3.4.3 Applications dans le domaine médical

En ce qui concerne le domaine médical, les réseaux de capteurs peuvent être très utiles où les capteurs sont utilisés pour [3] :

- Télésurveillance des informations physiologiques humaines ;
- Suivi et la surveillance des patients et des médecins au sein de l'hôpital ;
- Administration des médicaments.

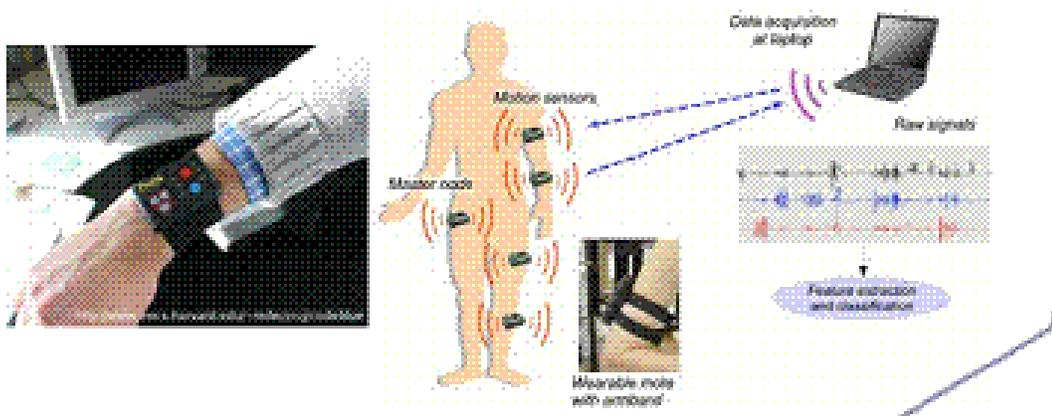


FIGURE 1.6 – Applications dans le domaine médical [14]

#### 1.3.4.4 Applications dans les maisons

Dans ce domaine, les capteurs sont utilisés pour [3] :

- Automatisation de la maison (manipulation de l'électroménager) ;
- Environnement intelligent.

#### 1.3.4.5 Autres applications commerciales

Les capteurs sont utilisés pour [3] :

- Contrôle de l'environnement dans les bâtiments d'administration ;
- Les musées interactifs ;
- Détection et surveillance des vols de voitures ;
- Surveillance et dépistage des véhicules.

### 1.3.5 Caractéristiques des réseaux de capteurs sans fil

Le routage dans les réseaux de capteurs sans fil est caractérisé par plusieurs critères qui les distinguent des réseaux ad hoc, on en cite [3] :

- Le nombre des nœuds dans le réseau de capteurs est nettement plus grand que dans les réseaux ad hoc ;
- Les nœuds capteurs sont déployés d'une manière dense dans le réseau et ils sont plus exposés aux pannes ;
- La topologie du réseau de capteurs change fréquemment ;
- Les nœuds capteurs sont caractérisés par des ressources plus limitées (ressource d'énergie, puissance de calcul et mémoire) ;
- Les réseaux de capteurs utilisent principalement les communications broadcast (diffusion) alors que la plupart des réseaux ad hoc sont basés sur les communications point-à-point ;
- L'impossibilité d'établir un système d'adressage mondial pour le déploiement d'un grand nombre de nœuds capteurs, puisqu'ils ne possèdent aucune identification (ID) globale tel que les adresses IP dans les réseaux ad hoc ;

La haute densité de déploiement des nœuds capteurs dans les réseaux de capteurs favorise l'utilisation des communications multi-saut qui consomment moins d'énergie que les communications à un

seul saut. Ces communications peuvent faire face à certains problèmes liés à la propagation du signal fréquemment rencontrés dans les transmissions sans fil à longues distances.

### 1.3.6 Facteurs et contraintes des réseaux de capteurs sans fil

La conception et la réalisation des réseaux de capteurs sans fil sont influencées par plusieurs paramètres, ces facteurs importants servent comme directives pour le développement des algorithmes et des protocoles utilisés dans les RCFS.

- **Tolérance aux pannes [3] :**

La défaillance ou le blocage des nœuds dans un réseau de capteurs peut être engendrés par plusieurs causes, notamment l'épuisement d'énergie, l'endommagement physique, ou les interférences liées à l'environnement.

La propriété de tolérance aux pannes est définie par l'habilité du réseau à maintenir ses fonctionnalités sans interruptions provoquées par la panne des capteurs. Elle vise donc à minimiser l'influence de ces pannes sur la tâche globale du réseau.

Cette propriété  $R(t)$  est modélisée dans [3] par une distribution de poisson où  $R(t)$  donne la probabilité de ne pas avoir une panne pour un nœud capteur pendant l'intervalle de temps  $[0, t]$ .

$$R(t) \equiv \exp(-\lambda_k t), \quad (1.1)$$

Où :

- $\lambda_k$  est le taux de pannes du nœud capteur  $k$  ;
- $t$  est la période de temps.

Les protocoles conçus pour les réseaux de capteurs doivent atteindre le niveau de tolérance aux pannes requis par le réseau, cela dépend essentiellement de l'environnement de déploiement du réseau, des caractéristiques des micro-capteurs, etc.

En effet, si le réseau de capteurs est destiné aux environnements avec un faible degré d'interférences, tel que ceux utilisés dans les bâtiments pour surveiller le taux d'humidité et le degré de température, les protocoles utilisés ne doivent pas cibler une grande tolérance aux pannes, car dans ce type de réseau, il n'existe pas une grande interférence avec l'environnement, et ses nœuds ne sont pas exposés au risque d'endommagement.

Par contre, si le réseau est destiné aux applications militaires telle que la surveillance et le contrôle d'un champs de bataille, le niveau de tolérance aux pannes visé par les protocoles employés doit être très élevé, car les nœuds sont exposés à un grand risque d'endommagement par des actions hostiles, et les informations captées sont très critiques.

Par conséquent, le niveau de tolérance aux pannes requis dépend de l'application du réseau de capteurs conçu, et les schémas de conception doivent prendre en charge ce paramètre ;

- **Scalabilité (scalability) :**

La surveillance d'un phénomène peut nécessiter le déploiement d'un nombre de nœuds qui est de l'ordre de plusieurs milliers de capteurs. Suivant l'application, ce nombre peut encore augmenter jusqu'à des millions de capteurs, les nouveaux schémas doivent pouvoir garantir un bon fonctionnement avec ce nombre de capteurs élevés, ils doivent aussi exploiter la nature fortement dense des réseaux de capteurs. Dans ce cas la topologie du réseau doit s'adapter au nombre de nœuds déployés [3] ;

- **Support de communication :**

Dans un réseau de capteurs, les nœuds sont liés via un moyen de communication sans fil, et ceci, en utilisant des fréquences radio. Cependant, il faut s'assurer de la disponibilité du moyen de transmission choisi dans l'environnement de captage, afin de permettre au réseau d'accomplir la totalité de ses tâches.

Pour les liens de communications via les fréquences radio, les bandes ISM (Industrial scientific Medical bands) peuvent être utilisées dans les systèmes téléphoniques sans fil et les réseaux WLAN, ces bandes de fréquence sont employées pour assurer des communications libres de charge dans le domaine industriel, scientifique ou médical, et ceci dans la plupart des pays du monde [3] ;

- **Consommation d'énergie :**

Comme les nœuds capteurs sont des composants micro-électroniques, ils ne peuvent être équipés que par des sources limitées d'énergie ( $<0.5$  Ampère-heure, 1.2 V). De plus, dans certaines applications, ces nœuds ne peuvent pas être dotés de mécanismes de rechargement d'énergie, par conséquent, la durée de vie d'un nœud capteur dépend fortement de la durée de vie de la batterie associée.

Sachant que les réseaux de capteurs sont basés sur la communication multi-sauts, chaque nœud joue à la fois un rôle d'initiateur de données et de routeur également, le mal fonctionnement d'un certain nombre de nœuds entraîne un changement significatif sur la topologie globale du réseau, et peut nécessiter un routage de paquets différent et une réorganisation totale du réseau. Sachant que la tâche principale d'un nœud capteur est de détecter des événements, effectuer un traitement rapide de données, et la transmission de données. C'est pour cela que le facteur de consommation d'énergie est d'une importance primordiale dans les réseaux de capteurs.

La majorité des travaux de recherche menés actuellement se concentrent sur ce problème afin de concevoir des algorithmes et protocoles spécifiques à ce genre de réseau qui permettent de consommer le minimum d'énergie [3];

- **Environnement de déploiement :**

Les caractéristiques de déploiement aléatoire, fonctionnement autonome, et fréquence élevé de pannes rendent la maintenance de la topologie d'un réseau de capteurs une tâche complexe. En effet, plusieurs centaines de capteurs sont déployés d'une manière dense dans le réseau, ceci exige une bonne gestion de la maintenance de la topologie du réseau déployé [3];

- **Coût de production :**

Le coût de production d'un seul capteur est très important pour l'évaluation du coût global du réseau, si ce dernier est supérieur à celui nécessaire pour le déploiement des capteurs, l'utilisation de cette nouvelle technologie ne serait pas financièrement justifié puisque le réseau de capteurs n'est pas équilibré en coût, car chaque capteur a un coût, ce dernier a une influence sur le coût de tout le réseau qui consiste en plusieurs capteurs [3];

- **Contraintes matérielles :**

Un nœud capteur contient quatre composants de base : l'unité de détection, l'unité de traitement, l'unité de communication, et l'unité de puissance. Il peut contenir également, suivant son domaine d'application, des éléments supplémentaires tels qu'un système de localisation géographique, ou bien un système générateurs d'énergie.

Comme nous l'avons invoqué précédemment, un nœud capteur peut contenir d'autres unités dépendantes de l'application du réseau. En effet, la plupart des opérations de captage et des algorithmes de routage dans les réseaux de capteurs sans fil requièrent la connaissance de la

localisation des nœuds avec une grande précision, car ces nœuds sont déployés d'une manière aléatoire et fonctionne d'une façon autonome, ceci rend l'intégration d'une unité, consacré au système de localisation, très commune dans un nœud capteur.

Toutes ces unités peuvent exiger leur intégration dans un boîtier de taille minimale inférieure à un centimètre cube, et avec un poids très léger qui permet aux nœuds de rester suspendu dans l'air, si l'application l'exige [3].

## 1.4 Conclusion

Les réseaux de capteurs sans fil présentent un intérêt considérable, et nécessitent de prendre une place très appréciée au sein de la communauté de la recherche vu leurs déploiement et leurs applications qui se développent chaque jour pour élargir leurs horizons, ils présentent une nouvelle étape dans l'évolution des technologies de l'information et de la communication.

Dans ce premier chapitre nous avons décrit les RCSFs et nous avons présenté leurs architectures de communications, ainsi que leurs caractéristiques et leur différente applications.

Cependant, nous avons remarqué que plusieurs facteurs et contraintes compliquent la gestion de ce type de réseau, cette mise au point nous a permis de déduire que les protocoles de routage jouent un rôle déterminant et crucial dans la conception des RCSFs, cela nous a mené à faire une étude sur les principaux protocoles de routages présentés dans la littérature dans le chapitre qui suit.

## Chapitre 2

# Etat de l'art des protocoles de routages dans les RCSFs

---

### 2.1 Introduction

Le routage dans les réseaux de capteurs sans fil, est une méthode qui consiste à acheminer les données dans le réseau depuis un nœud source vers un nœud destination appelé nœud puits (Sink), ou encore (Station de Base). Dont le but étant d'assurer la connectivité de tout le réseau, où chaque nœud qui compose le réseau peut participer au routage de données directement à la SB, et dans le cas où un nœud n'est pas capable de transmettre les données de manière directe, il est possible d'utiliser le routage multi-sauts en utilisant des nœuds intermédiaires.

Cependant, les réseaux de capteurs sans fil sont fortement influencés par certaines caractéristiques, comme la limitation de la ressource énergétique disponible au niveau de chaque nœud, la densité des nœuds, ainsi que la topologie qu'ils forment dans le réseau. Actuellement, diverses recherches sur le domaine des RCSFs visent à assurer le prolongement de la durée de vie du réseau, à travers la conception de protocoles de routage qui permettent de satisfaire cet objectif.

La consommation d'énergie représente un défi majeur pour le routage de données. De ce fait, le développement de nouveaux protocoles de routage dans les RCSFs est essentiel. Ces protocoles de routage doivent permettre l'établissement de routes pour le routage de données dans le réseau d'un nœud à

un autre jusqu'à arriver à l'utilisateur final, tout en optimisant la consommation d'énergie de manière efficace.

## 2.2 Facteurs de conception des protocoles de routage dans les Réseaux de capteurs sans fil

Les protocoles de routage dans les RCSFs sont responsables de découvrir et de maintenir une efficacité énergétique dans les réseaux, afin de rendre les communications fiables et efficaces.

En raison des limitations concernant ce type de réseaux, l'objectif principal de la conception du protocole de routage étant de prolonger la durée de vie du réseau tout en gardant les capteurs en vie le plus longtemps possible. Ce problème résulte dans le fait de maintenir le réseau connecté pendant une longue période de temps.

La conception de protocoles de routage dans les RCSFs est affectée par de nombreux facteurs. De ce fait, une communication efficace peut être obtenue dans les RCSFs en les surmontant, ces facteurs sont présentés ci-dessous [22] :

- Déploiement des nœuds ;
- Consommation d'énergie ;
- Qualité de service (QOS "Quality Of Service") ;
- Tolérance aux pannes ;
- Agrégation de données ;
- Scalabilité "Scalability" ;
- Couverture du réseau ;
- Nature des nœuds ;
- Modèles de livraison de données.

## 2.3 Approches de routage dans les réseaux de capteurs sans fil

Dans un réseau de capteurs sans fil, les nœuds sont densément déployés dans un champ de capture proche ou à l'intérieur du phénomène perçu. Afin de permettre la communication dans le réseau, des protocoles de routage basés sur la communication multi-sauts sont nécessaires entre les nœuds et le nœud puits (sink) du réseau. Le principe du fonctionnement de chaque protocole diffère suivant la philosophie de l'approche à laquelle il appartient [22]. Ces approches peuvent être distinguées suivant :

- La structure du réseau ;
- Le mode de fonctionnement du protocole ;
- L'établissement de routes ;
- Initiateur de communications.

La figure 2.1 ci-dessous montre la classification des différentes approches de routage dans les RCSFs selon les critères cités précédemment.

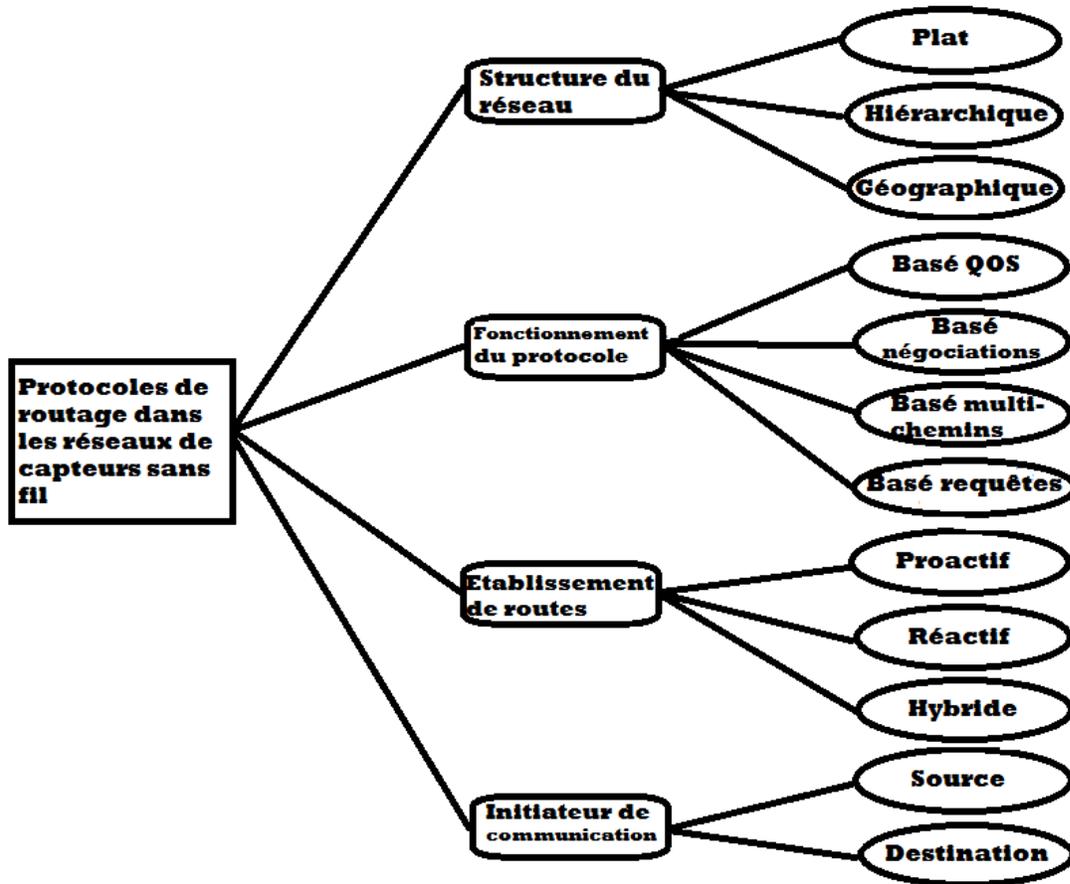


FIGURE 2.1 – Diagramme de classification des protocoles de routage [3]

## 2.4 Classification des protocoles de routage

### 2.4.1 Selon la structure du réseau

La structure du réseau peut être classée en fonction de l'uniformité des nœuds. Dans certains réseaux de capteurs, les nœuds peuvent être déployés de manière uniforme et possèdent le même rôle. Alors que dans d'autres réseaux de capteurs, il est possible de faire la distinction entre les nœuds déployés dans le réseau.

Les principaux facteurs des protocoles de routage qui appartiennent à cette catégorie, est la manière dont les nœuds sont reliés entre eux, ainsi que la manière dont les données sont routées jusqu'à la station de base (ou nœud puits) [18].

Ainsi, les protocoles de routage appartenant à cette catégorie peuvent être classés comme suit :

- Protocoles plat (Flat protocols);
- Protocoles hiérarchiques (Hierarchical protocols/ Clustering-based protocols);
- Protocoles géographiques (Location-based protocols).

## 1. Topologie plat

Dans ce type de topologie, les nœuds jouent le même rôle et communiquent entre eux pour router les données jusqu'au nœud puits (Station de Base). Les routages plats sont caractérisés par : la simplicité des protocoles de routage, un faible coût de maintiens de l'infrastructure entre les nœuds qui communiquent, ainsi qu'une faible scalabilité du au fonctionnement identique des nœuds car lorsque le nombre de nœuds dans le réseau est grand, il est impossible d'affecter à chaque nœud un identifiant unique [18].

Parmi les protocoles appartenant à cette classe, présentés dans la littérature on en cite :

- **Flooding et Gossiping** [18]

Flooding et gossiping sont deux mécanismes classiques pour le routage de données dans les réseaux de capteurs sans avoir besoin d'aucun des algorithmes de routage et de maintenance de topologie.

La technique d'inondation (Flooding), permet à chaque nœud recevant une donnée de la diffuser à tous les nœuds voisins dans le réseau jusqu'à ce que le nombre de sauts maximum pour cette donnée est atteint ou que la donnée soit arrivée à destination qui représente la station de base, par conséquent avec cette technique il y aura une inondation totale du réseau.

**Discussion :** Flooding est un protocole réactif qui n'exige pas une maintenance dans la topologie du réseau ni des algorithmes complexes pour la découvertes des routes (chemins), mais souffre de certains problèmes :

- L'implosion :** c'est la duplication dans l'envoi du même message vers le même nœud, où chaque nœud reçoit autant de fois la même donnée que le nombre de ses voisins ;
- Overlap :** Lorsque les nœuds partagent la même région dans le réseau, et observent un même phénomène dans cette région, donc les mêmes informations seront envoyés plusieurs fois aux nœuds du réseau (redondance de données) ;

- iii. **Indisponibilité de ressources énergétiques** : les nœuds utilisent leurs ressources énergétiques aveuglément sans tenir compte qu'ils sont limités dans le temps.

La technique Flooding possède deux caractéristiques qui découlent du fait que toutes les routes sont testées :

- Tant qu'il y a une route du nœud source vers le nœud destination, la livraison des paquets de données est garantie ;
- Une copie du paquet de données arrivera par la route la plus rapide possible.

D'autre part, le mécanisme **Gossiping** propose une légère amélioration de **Flooding** où un nœud qui reçoit une donnée ne la diffuse pas à tous ses voisins, mais il la transmet à un seul nœud voisin sélectionné aléatoirement jusqu'à ce que la donnée arrive à la station de base. Malgré que cette technique permet de résoudre le problème d'implosion, en ayant une seule copie du message au niveau de chaque nœud. Cependant, l'inconvénient avec **Gossiping** est qu'il prend beaucoup de temps pour propager les données dans tout le réseau.

#### • Tableau comparatif des protocoles la topologie plat

Protocole de routage	Scalabilité	Mobilité	Métrique de routes	Robustesse
<b>Gossiping</b>	Bon	Bon	Aléatoire	Bon
<b>Flooding</b>	Limité	Faible	Chemin plus court	Bon

TABLE 2.1 – Tableau comparatif des protocoles de routage de la topologie plat dans les RCSFs [18]

## 2. Topologie Hiérarchique

Dans une topologie hiérarchique, les nœuds ont des rôles différents. En effet, certains nœuds sont utilisés pour effectuer des fonctions particulières tels que, l'agrégation de données. L'une des méthodes les plus employées dans cette topologie est le Clustering.

Le Clustering consiste à diviser le réseau en groupes appelés clusters, un cluster est constitué d'un chef appelé Cluster Head (CH) et de ses nœuds membres. Suivant l'application, les nœuds membres peuvent être des voisins directs ou indirects du CH.

Après le déploiement des nœuds sur une zone du réseau pour la détection des événements, ils commencent par découvrir leurs nœuds voisins afin de construire une topologie de communica-

tion. Par la suite, pour l'établissement de la communication, les nœuds sont chargés de collecter des données, de les acheminer vers les CHs qui les agrègent et les envoie vers un nœud particulier appelé nœud puits (Sink) ou Station de Base. Ce dernier analyse ces données et les transmet à son tour à l'utilisateur final soit par internet ou bien satellite.

Cette topologie présente beaucoup d'avantages, tels que l'agrégation de données collectées ainsi qu'un grande scalabilité ce qui permet de prolonger la durée de vie du réseau. Le principal inconvénient de cette topologie est la surcharge des CHs, induisant ainsi à un déséquilibre de la consommation d'énergie sur le réseau [22].

Parmi les protocoles appartenant à cette classe :

- **LEACH (Low-energy Adaptative Clustering Hierarchy) [12]**

Le protocole LEACH, est le plus populaire des protocoles de routage hiérarchique, il a été proposé par *Heinzelman et Al.* Il est basé sur l'algorithme de clustering distribué, qui permet de diviser le réseau en un ensemble de cluster où dans chaque cluster on y trouve un Cluster-Heads (CHs) et des nœuds membres qui appartiennent à chaque cluster, dans chaque cluster le protocole se déroule en plusieurs rounds, et chaque round est composé de deux phases :

- i. **La phase Set up :**

Dans cette phase, la formation des clusters est établit et des nœuds CHs sont élus. Au départ, chaque nœud va choisir un nombre aléatoire ( $nb$ ), qui doit être borné entre 0 et 1, ( $0 < nb < 1$ ), ce nombre sera par la suite comparé au seuil  $T(n)$  qui est représenté par la formule suivante :

$$T(n) = \begin{cases} \frac{p}{1-p(r \bmod \frac{1}{p})} & \text{si } (n \in G), \\ 0 & \text{sinon,} \end{cases}$$

Où :

- $p$  : représente le pourcentage des CHs sur le réseau (généralement 5%) ;
- $G$  : ensemble des nœuds qui n'étaient pas élus CH dans les  $(1/p)$  rounds précédents ;
- $r$  : le numéro du round en cours.

Dans le cas où ( $nb < T(n)$ ), alors le nœud sera élu CH, sinon il va rejoindre le CH le plus proche de son voisinage.

Une fois le Cluster Head élu, ce dernier informe les autres nœuds en envoyant un message ADV en utilisant le protocole CSMA de la couche MAC, ADV représente un message de petite taille qui comporte l'identité (id) du nœud, les autres nœuds déterminent dans quel cluster ils appartiennent dans une itération en choisissant le CH, pour se faire chaque nœud se base sur la puissance du signal reçu du message ADV à partir des CHs. Après que le nœud ait choisit son cluster, chaque nœud envoie un message Join Request (Join-Req) au Cluster Head de son cluster afin de rejoindre le cluster en utilisant aussi le protocole CSMA, ce message contient l'identité du nœud ainsi que l'identité du CH, à la fin de cette première phase tous les clusters sont formés ce qui permet de passer à la deuxième phase.

ii. **La phase Steady-State :**

Dans cette phase, les données détectées par les nœuds des clusters seront transmises aux CHs durant des slots de temps. Cette transmission est synchronisée par l'ordonnanceur TDMA, ensuite les CHs effectuent l'opération d'agrégation et le résultat sera ainsi envoyé à la station de base.

**Discussion :** LEACH est complètement distribué, ne nécessite aucune connaissance globale du réseau. Il permet de réduire la dissipation (perte) d'énergie dans le réseau, sachant que l'élection des CHs est probabiliste il ya des chances de sélectionner un nœud avec une faible énergie donc il peut mourir rapidement, ce qui provoque un dysfonctionnement.

LEACH utilise le scheduler TDMA afin d'empêcher les collisions entre les CHs du réseau, il utilise aussi une communication à un seul saut où chaque nœud transmet directement au CH puis ce dernier transmet à la station de base, par conséquent il ne peut pas être appliqué dans de vastes réseaux.

• **LEACH-C (Low-energy Adaptative Clustering Hierarchy-Centralized) [23]**

LEACH-C est une version améliorée du protocole de routage hiérarchique LEACH, où les clusters sont formés d'une manière centralisée par la station de base. LEACH-C utilise presque les mêmes étapes que ceux de LEACH avec une méthodologie structurale différente, en gardant la même phase de communication (Steady-State) et une phase (Set up) légèrement différente.

Durant la phase Set up la station de base reçoit depuis chaque nœud les informations concernant leurs localisations, et leur niveau d'énergie. Après la collection de ses informations, elle calcule l'énergie moyenne des nœuds, et n'importe quel nœud ayant une énergie inférieure à cette moyenne il ne peut pas être élu comme CH. Dès que le nombre optimal des Clusters Heads optimal et les clusters correspondants sont trouvés, la station de base transmet cette information à tous les autres nœuds du réseau.

**Discussion :** LEACH-C diminue considérablement la dissipation d'énergie. La station de base utilise l'ensemble des informations qu'elle collecte dans le réseau pour la création des clusters ce qui nécessite moins d'énergie pour la transmission de données.

Comme cité dans l'article [23], dans le protocole de routage LEACH-C le nombre de Clusters Heads reste fixe pour chaque round, alors que dans LEACH le nombre change à chaque tour en raison du manque de coordination entre les nœuds.

- **MH-LEACH (Multi-Hop Low-energy Adaptative Clustering Hierarchy) [13]**

Le protocole MH-LEACH représente une extension du protocole de routage hiérarchique LEACH, l'objectif de ce protocole est d'envoyer les données à la station de base en utilisant le routage multi-sauts.

Comme dans le protocole LEACH, Multi Hop-LEACH utilise le même mécanisme pour l'élection de CHs. Pour la collecte de données, il ya deux types de communications qui sont autorisées :

- i. Communications inter-cluster, où le réseau est divisé en clusters, les CHs de chaque cluster reçoivent des données des nœuds membres du cluster, ils effectuent l'agrégation de données et envoient les informations finales à la station de base par l'intermédiaire d'autres nœuds ;
- ii. Communications intra-cluster, où les nœuds membres du cluster envoient leurs données aux autres membres du cluster pour atteindre les CHs.

Ce protocole fonctionne par rounds comme LEACH, et sélectionne le chemin avec le minimum de sauts entre les CHs et la SB. Il est important de souligner que lors de la construction de routes entre les CHs et la SB, les clusters sont formés et les CHs de chaque cluster sont sélectionnés.

L'objectif de MH-LEACH étant de trouver plusieurs chemins possibles pour les CHs afin de transmettre leurs données agrégées jusqu'à la station de base dans le but d'économiser de l'énergie. Pour ce faire, il repose sur la puissance du signal reçu, à savoir RSSI (Received Signal Strength Indicator) [6, 11, 17] que nous allons expliquer dans la section "Discussion" ci-dessous, le nœud ayant le plus grand RSSI est celui qui est le plus proche.

Pour la création de routes entre les CHs du réseau, le protocole MH-LEACH propose deux étapes :

i. Etape (1) :

Après élection des CHs comme dans le protocole LEACH, ces derniers diffusent des messages de contrôle à travers le réseau, dans le but que chaque CH du réseau va construire une table de routage contenant la puissance du signal reçu RSSI de chaque CH ainsi que la puissance du signal de la SB, cette table sera utilisée pour former les CHs les plus proches.

ii. Etape (2) :

Après cela, chaque CH envoie à la SB ses routes initiales obtenues à partir des tables de routage. Malgré que ces tables de routage indiquent les nœuds CHs les plus proches pour envoyer les données, elles peuvent contenir de mauvaises routes, pour résoudre ces problèmes, la SB va vérifier et corriger ses tables de routage, en vérifiant si un CH va faire partie de la route d'un autre, et si un CH est plus proche qu'un autre par rapport à elle.

Après avoir examiné la table de routage de chaque CH, la station de base renvoie aux Cluster Heads les tables correctes, c'est-à-dire les routes corrigées pour chaque CH. Ainsi, le processus de calcul des routes se termine, où chaque CH ne transmet pas directement à la station de base, mais pour un autre CH qui est plus proche à elle. Après cela, et la phase de la collecte et de transfert de données peut commencer.

**Discussion :** L'objectif principal du protocole MH-LEACH est l'utilisation des communications multi-sauts entre les CHs jusqu'à ce que les données atteignent la station de base. Grâce à cette caractéristique, ce protocole permet de diminuer la consommation d'énergie des nœuds du réseau et de prolonger la durée de vie du réseau, étant donné que plus les

distances pour la transmission de données sont courtes plus la consommation d'énergie est faible.

D'après les simulations établis sur les deux protocoles LEACH et MH-LEACH dans l'article ([13]), des économies en termes de consommation d'énergies et le prolongement de durée de vie du réseau en termes de nombre de nœuds morts dans le réseau ont été augmenté dans MH-LEACH par rapport au protocole LEACH [13].

i. **Received Signal Strength Indicator (RSSI)** [6, 11, 17] : Plusieurs techniques sont développées pour les estimations des distances entre les nœuds voisins. Parmi lesquelles nous trouvons la méthode de la force du signal reçu (RSSI).

Dans les RCSFs, chaque capteur est équipé d'une radio. De ce fait, un nœud à l'écoute d'une transmission radio doit être en mesure d'utiliser la force du signal reçu pour calculer sa distance de l'émetteur. RSSI permet l'estimation des distances entre les nœuds dans le réseau de capteurs puisque tous les nœuds possèdent leur propre radios.

Cependant, en pratique les distances calculées par RSSI contiennent des erreurs de l'ordre de quelques mètres en raison, par exemple, des obstacles physiques tels que les murs, meubles, etc.

- **PEGASIS (Power-Efficient Gathering in Sensor Information Systems)** [2]

PEGASIS est une version améliorée de LEACH, il a été proposé par Lindsey et Raghavendra. Dont l'idée principale est de former une chaîne entre les nœuds de telle sorte que chaque nœud reçoive de et communique à un voisin proche, les données collectées sont transmises d'un nœud à un autre jusqu'à ce qu'elle arrivent à un nœud particulier qui agrège les données reçues et les transmet à la station de base. La construction de la chaîne est réalisée de la manière suivante. Comme on le voit sur figure 2.2 :

Le nœud  $C_0$  passe ses données au nœud  $C_1$ . Le nœud  $C_1$  à son tour agrège les données du nœud  $C_0$  avec ses propres données et transmet ensuite au nœud leader. Après que le nœud  $C_2$  transmet le jeton au nœud  $C_4$ , le nœud  $C_4$  transmet ses données au nœud  $C_3$ . Le nœud  $C_3$  agrège les données du nœud  $C_4$  avec sa propre et transmet ensuite au chef. Le nœud  $C_2$  attend de recevoir les données des deux voisins, puis agrège ses données avec les données de ses voisins. Enfin, le nœud  $C_2$  transmet un message à la station de base.

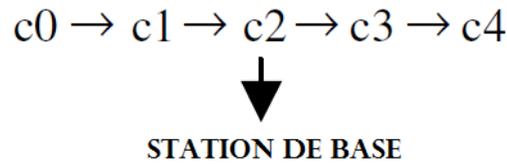


FIGURE 2.2 – Illustration de construction de la chaîne dans PEGASIS [2]

Dans PEGASIS, les nœuds qui font la transmission de données à la station de base sont choisis tour à tour en utilisant la politique "Round-Robin" (c'est-à-dire Tourniquet), Afin de minimiser l'énergie dépensée par un nœud durant chaque round.

**Discussion :** Contrairement au protocole LEACH, PEGASIS utilise un routage multi-sauts ce qui minimise la distance entre les nœuds en formant des chaînes en sélectionnant un seul nœud pour la transmission vers la station de base au lieu d'utiliser plusieurs nœuds (Les nœuds CHs dans le cas du protocole LEACH), par conséquent PEGASIS provoque des retards dans la transmission des données entre les nœuds de la chaîne ce qui fait que le nœud qui est sensé transmettre à la station de base reste bloqué.

Le protocole PEGASIS, permet de réduire l'overhead causé par la formation dynamique des clusters vu dans LEACH et permet aussi de réduire le nombre de transmissions en utilisant l'agrégation de données.

Cependant, dans PEGASIS il nécessite que chaque nœud connaisse les informations globales du réseau, pour cela le réseau ne peut être scalable, et un tel protocole ne convient pas pour les réseaux de capteurs lorsque cette connaissance globale n'est pas facile à obtenir.

- **H-PEGASIS (Hierarchical Power-Efficient Gathering in Sensor Information Systems) [2]**

Ce protocole est une amélioration du protocole PEGASIS appelé Hierarchical-PEGASIS, H-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 chaîne 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  $O(\log N)$  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.

**Discussion :** Dans [2], l'avantage du protocole H-PEGASIS est qu'il permet d'éviter la surcharge (overhead) des clusters dans le réseau par rapport au protocole LEACH, mais qu'il nécessite un ajustement dans la topologie lorsqu'elle est dynamique. Par exemple, chaque nœud doit être informé de l'état de son voisin pour qu'il sache où acheminer ces données. Un tel ajustement de la topologie peut introduire une surcharge en particulier pour les réseaux très utilisés.

- **BCDCP (Base-Station Controlled Dynamic Clustering Protocol) [18]**

C'est un protocole de routage hiérarchique, l'idée principale de BCDCP est de former des clusters équilibrés en taille, c'est-à-dire le nombre de nœuds qui forment le cluster.

De ce fait, avant la construction de routes pour le routage de données, la station de base en émettant un message de contrôle reçoit les données des nœuds, qui concernent leurs niveaux d'énergies. Puis, cette dernière calcule la moyenne des énergies reçues, en se basant sur cette moyenne des énergies, elle sélectionne les nœuds ayant un niveau d'énergie supérieure à la

moyenne calculés seront élus Clusters Heads.

Pour former les clusters, chaque CH fait en sorte d'avoir le même nombre de nœuds membres de son cluster que les autres CHs. Tout ceci est effectué, dans le but de :

- Eviter la surcharge des Cluster Heads dans le réseau ;
- Assurer le positionnement uniforme des Cluster Heads à travers tout le réseau ;
- Utiliser les communications CH à CH afin de transmettre les données à la station de base.

**Discussion :** Dans [18], d'après les résultats de simulation effectués sur le protocole BCDP, ont montré qu'il permet de réduire la consommation d'énergie et de prolonger la durée de vie du réseau par rapport aux protocoles cités précédemment LEACH, LEACH-C et PEGASIS.

- **TEEN (Threshold sensitive Energy Efficient sensor Network) [2]**

Le protocole TEEN a été proposé par Manjeshwar and Agrawal, c'est un protocole Hiérarchique conçu pour être sensible aux changements soudains des attributs détectés tels que : la température.

Ce protocole est basé sur une approche hiérarchique avec l'utilisation d'un mécanisme centrée-données.

L'architecture du réseau de capteurs est basée sur un groupement hiérarchique où les nœuds les plus proches, c'est-à-dire côte à côte, forment des clusters et ce processus va sur plusieurs niveaux jusqu'à ce que la station de base soit atteinte. Comme représenté dans la figure 2.3 ci-dessous.

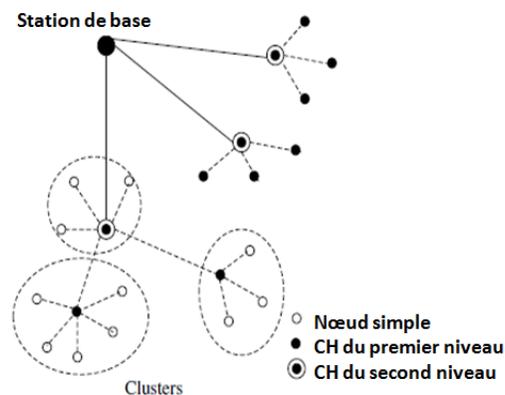


FIGURE 2.3 – Illustration de la hiérarchie du clustering du protocole TEEN [2]

TEEN emploie la même stratégie pour l'étape de formation des clusters que celle du protocole LEACH, ce qui diffère est dans la phase de transmission de données où durant cette phase, chaque nœud Cluster Head envoie deux seuils : le seuil Hard ( $Ht$  : *Hard Threshold*) et le seuil soft ( $St$  : *Soft Threshold*) aux nœuds de son cluster. Le seuil Hard ( $Ht$ ) qui représente la valeur du paramètre surveillé (contrôlé) et le seuil soft ( $St$ ) qui représente une petite variation de la valeur du paramètre contrôlé.

Lorsque la valeur captée pour la première fois dans le réseau est supérieur au seuil ( $Ht$ ) alors elle est stockée dans un variable appelée ( $SV$ ), cette dernière sera transmise par le CH au nœud concerné. Si la valeur captée dépasse la valeur stockée ( $SV$ ) par un paramètre ( $St$ ) ainsi le nœud décide de transmettre et stocke cette nouvelle valeur dans son cache pour faire des comparaisons ultérieurs.

**Discussion :** L'efficacité énergétique des TEEN dépend du nombre de transmissions. Cela signifie que, s'il y a beaucoup de transmissions de données dans le réseau, la quantité d'énergie consommée par l'ensemble du réseau est élevée et vice-versa.

Dans TEEN l'utilisation du seuil ( $St$ ) diminue le nombre de transmissions s'il y a un petit ou pas de changement de la valeur du paramètre surveillé, ce qui permet de réduire la consommation d'énergie. Cependant TEEN souffre de certains problèmes, tels que, si les seuils ne sont pas atteints alors aucune transmission ne sera établie par les nœuds, et aucune donnée ne sera transmise à l'utilisateur.

- **APTEEN (The Adaptive Threshold sensitive Energy Efficient sensor Network protocol) [2]**

Manjeshwar and Agrawal ont proposé ce protocole hiérarchique, APTEEN est une extension du protocole TEEN, il permet de détecter les données de manière périodique.

L'architecture du réseau est identique à celle du protocole hiérarchique TEEN. Lorsque la station de base forme des Clusters, les Clusters Head transmettent les attributs, les valeurs des seuils Hard ( $Ht$ ) et Soft ( $St$ ) à tous les nœuds. Les Clusters Head utilisent également l'agrégation de données dans le but d'économiser de l'énergie.

APTEEN supporte trois types de requêtes différentes :

- Historique : pour analyser les données du passé ;

- Temps réel : pour avoir une vue instantanée du réseau ;
- Persistante : pour surveiller un évènement sur une période de temps.

**Discussion :** L'avantage du protocole APTEEN comparé au protocole TEEN, est que les nœuds consomment moins d'énergie, ce qui prolonge la durée de vie du réseau. Des simulations sur TEEN et APTEEN ont montré qu'il surpasse en performance le protocole de routage hiérarchique LEACH. et que les performances d'APTEEN se situent entre LEACH et TEEN en matière de dissipation d'énergie et de durée de vie du réseau.

Cependant, APTEEN nécessite une complexité supplémentaire pour implémenter les fonctions de seuils, ce qui provoque de longs retards dans les délais de transmission de données dans le réseau.

- **HEED (Hybrid Energy Efficient Distributed) [1]**

HEED est un protocole de routage qui utilise une approche distribuée, la sélection des Clusters Head se fait sur la base de l'énergie disponible et du coût de la communication, à la différence du protocole LEACH, il n'utilise pas la sélection aléatoire.

HEED possède trois caractéristiques principales :

- Une meilleure distribution des CHs sur le réseau, où la probabilité que deux nœuds à portée deviennent des CHs est faible ;
- La consommation d'énergie n'est pas supposée être uniforme pour tous les nœuds ;
- La probabilité d'élire un CH peut être ajusté pour assurer la connectivité inter-CH, avec un rayon de transmission prédéfini.

Le protocole HEED, se compose de trois phases :

- La phase d'initialisation :

Dans cette phase, chaque nœud définit une probabilité de devenir un Cluster Head (CHprob), définit comme suit :

$$CHprob = Cprob * \frac{E_{residuelle}}{E_{max}}, \quad (2.1)$$

Où :

- Eresiduelle : Représente l'énergie restante du nœud ;

- Emax : Représente l'énergie initiale du nœud ;
- Cprob : Représente le nombre optimal des clusters.

ii. La phase répétition :

Pendant cette phase, chaque nœud passe par plusieurs itérations jusqu'à ce qu'il trouve le CH avec qui il peut transmettre en utilisant une puissance de transmission minimum (coût). Si ce nœud n'est pas entendu par aucun CH, il choisit lui-même d'être un CH et diffuse un message afin de l'annoncer à ses nœuds voisins en les informant du changement de son statut. Enfin, chaque nœud double sa valeur CHprob et va à la prochaine itération de cette phase.

iii. La phase finale :

Dans cette phase, chaque nœud prend une décision finale sur son statut. Soit il choisit le CH avec un coût minimal ou alors il se prononce comme CH.

**Discussion :** HEED assure la scalabilité et une grande efficacité énergétique grâce à la communication multi-sauts. Cependant, la topologie en clusters ne réalise pas la consommation minimale d'énergie dans les communications intra-cluster car la consommation d'énergie n'étant pas uniforme sur tous les nœuds, les clusters générés ne sont pas tellement équilibrés en taille, et il souffre d'*Overhead* (surcharge) due à la multiplicité de rounds.

Huang et Wu ont fait une extraction du protocole (HEED) appelé(E-HEED) [1], en abandonnant pas les nœuds qui ne sont pas écoutés par aucun CH, ces nœuds sont appelés (nœuds orphelins), en devenant eux même des CHs. Cette version du protocole HEED a opté pour juste ré-exécuter l'algorithme HEED pour ces nœuds orphelins, ce qui permet de diminuer le nombre de CHs, ce qui devrait réduire la taille de l'arbre de routage utilisée pour une communication inter-CH (CH à CH) et limite les retards dans la collection de données qui transitent dans le réseau.

- **VGA (Virtual Grid Architecture Routing) [15]**

Ce protocole hiérarchique est divisé, en deux phases :

i. Phase de Clustering :

Durant cette phase, tous les nœuds du réseau sont disposés dans une certaine topologie, la plupart du temps la topologie est fixe car les nœuds sont généralement fixes dans les

applications. A l'intérieur de chaque cluster un Cluster-Head est présent, ce dernier est utilisé pour l'agrégation de données, le CH dans VGA est appelé agrégateur local (LA : Local Aggregator). Un sous-ensemble de ces agrégateurs locaux (LA) est sélectionné pour effectuer l'agrégation globale ou l'agrégation dans le cluster et ses membres sont connus en tant que maître agrégateur (MA : Master Aggregator).

ii. Phase d'agrégation de données :

Dans cette phase, certaines approches heuristiques sont utilisées afin de pouvoir obtenir une solution facile, équitable, efficace et optimale.

Un exemple de l'architecture du réseau pour le protocole de routage hiérarchique VGA, est représenté sur la figure 2.4 ci dessous.

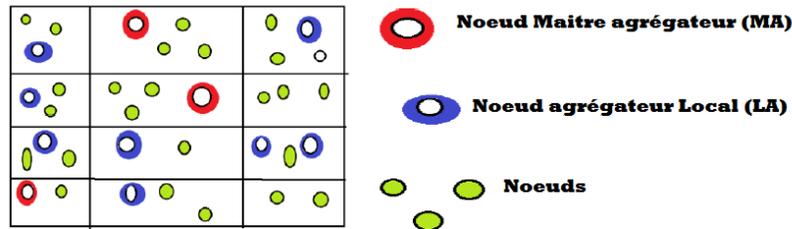


FIGURE 2.4 – Architecture du réseau pour VGA [15]

**Discussion :** Le principal avantage de ce protocole est qu'il peut atteindre l'efficacité énergétique et une maximisation de la durée de vie du réseau, mais le problème de la détermination de l'ensemble (MA "Maître Agrégateurs") à partir des (LA "Agrégateurs Local") est un problème NP-difficile.

- **PEDAP-PA (Power Efficient Data gathering and Aggregation Protocol-Power Aware) [12]**

Tan et Körpeoglu ont proposé PEDAP et PEDAP-PA. Les deux protocoles sont des protocoles de routages hiérarchiques centralisés basé-chaines. L'objectif principal de PEDAP est également de prolonger la durée de vie du réseau.

Le mécanisme des protocoles PEDAP et PEDAP-PA est similaire à celui du protocole hiérarchique cité précédemment PEGASIS, la différence réside dans le processus de formation de la chaîne. Au lieu de former la chaîne de nœuds à l'aide de l'interaction entre les nœuds locaux, alors que les 2 protocoles de routage PEDAP et PEDAP-PA appliquent le calcul des

liens dans tout le réseau avant la construction des arbres de recouvrement minimal (MST) appelé en anglais (the Minimum Spanning Trees). Comme le montre la figure 2.5 ci-dessous un exemple d'une chaîne possible est construite par PEDAP et PEDAP-PA.

Le MST est le meilleur arbre qui donne les chemins les plus courts pour interconnecter tous les nœuds dans le réseau.

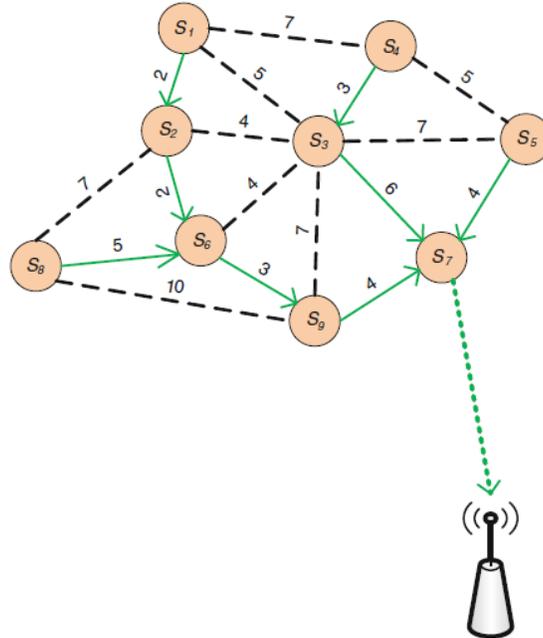


FIGURE 2.5 – Construction d'une chaîne possible dans PEDAP et PEDAP-PA [12]

**Discussion :** D'après les résultats d'une simulation faite entre les protocoles de routage hiérarchiques LEACH, PEGASIS et PEDAP-PA dans [12], en terme de durée de vie du réseau, montre que le protocole de routage PEDAP-PA garantie une bonne qualité de réseau pour une longue période par rapport aux protocoles LEACH et PEGASIS, vu le nombre de nœuds qui restent vivants, l'inconvénient de ce protocole est que la gestion de la topologie se fait au niveau de la SB. Comme évoqué précédemment, l'algorithme centralisé limite la capacité de l'auto-organisation des RCSFs, ce qui limite la scalabilité du réseau.

- **TTDD (Two-Tier Data Dissemination) [18]**

Le protocole de routage hiérarchique TTDD suppose que les nœuds capteurs sont fixes à leurs emplacements courant et les nœuds puits (sink) sont autorisés à modifier leur emplacement dynamiquement. Au moment où un événement est détecté par les capteurs voisins, alors l'un

d'entre eux devient la source qui va générer des rapports de données.

Après que la grille virtuelle soit construite par le nœud source, il se choisit comme un point de cette grille. Il envoie un message d'annonce de données à ses quatre différents points qui lui sont adjacents. Le message ne s'arrête que lorsqu'il atteint le plus proche nœud du point de passage. Ce processus se poursuit jusqu'à ce que le message atteigne la limite du réseau. Sur la figure 2.6 , un exemple est présenté pour la construction de la grille. Dans ce cas, la source B et un sink (nœud puits) S et un champ de capteur à deux dimensions sont considérées. La source B divise le champ en une grille de cellules. Chaque cellule est un carré de dimension  $\alpha$ . La source B est lui-même un point de la grille. Il propage des annonces de données pour atteindre tous les autres points de la grille.

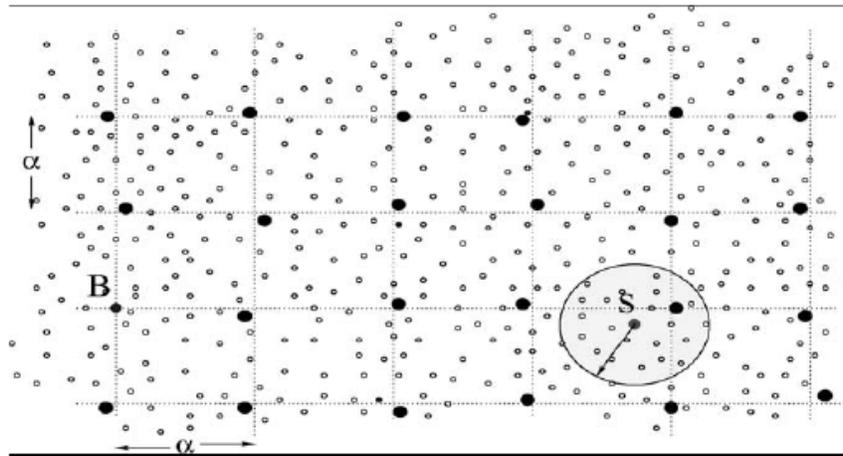


FIGURE 2.6 – Un nœud source B et nœud Sink S [18]

**Discussion :** Le principal avantage de ce protocole est qu'il résout le problème de mobilité du nœud puits (sink) dans un vaste réseau de capteurs sans fil. Cependant il enregistre une grande latence et une faible efficacité énergétique.

Le principal inconvénient est que chaque nœud source construit une structure de grille virtuelle de points pour fournir des données aux nœuds puits mobiles.

- **GBDD (Grid Based Data Dissemination) [18]**

Contrairement au protocole de routage hiérarchique TTDD cité précédemment, où c'est le nœud source qui commence la construction de la grille. Dans GBDD, c'est le nœud puits

(Sink) qui est intéressé par envoyer ou recevoir des données est celui qui commence le processus de construction de la grille.

Les positions (coordonnées  $(x, y)$ ) de ce nœud puits représentent le point de départ pour la formation des cellules qui composent la grille, où la taille des cellules est déterminée par le double de la portée radio du nœud capteur. On en distingue deux paramètres, le "RH" et "RL" qui désignent les portées de chaque nœud pour une puissance radio respectivement élevée et faible. Les cellules de la grille sont des carrés de cotés de taille égale à  $(a)$ .

**Discussion :** D'après [18], les résultats de simulations ont montré qu'il conserve environ 43% d'énergie comparé au protocole TTDD, ce qui prolonge la durée de vie du réseau.

- **SHPER (Scaling Hierarchical Power Efficient Routing) [4]**

SHPER a été proposé par Kandris et al. Ce protocole suppose que les nœuds sont déployés de manière aléatoire sur le réseau, et sont fixes, et suppose aussi que la station de base est fixe est située très loin de la zone de déploiement.

SHEPR est un protocole de routage hiérarchique, qui consiste en deux (2) phases :

- i. Phase d'initialisation :

Durant cette première phase, la station de base envoie un message de contrôle aux nœuds du réseau afin de connaître leurs positions et distances par rapport à elle, ainsi que leurs niveau d'énergies. Après avoir reçu les réponses des nœuds, la station de base va élire les nœuds CHs en sélectionnant aléatoirement les nœuds possédant un niveau d'énergie qui dépasse le seuil fixé par cette dernière.

Afin d'informer les nœuds du réseau des CHs élus, la station de base envoie un message contenant les identifiants des CHs élus et la valeur du seuil d'énergie fixée.

- ii. Phase Steady-State : Après élection des CHs dans la phase précédente, ces derniers vont définir des chemins à efficacité énergétique pour router les données jusqu'à la station de base.

**Discussion :** Dans [4], les résultats de simulation ont montré qu'il est plus performant que le protocole cité précédemment TEEN en termes de consommation énergétiques.

L'avantage dans le protocole SHEPR est qu'il prend en considération l'énergie résiduelle des nœuds qui composent le réseau, ce qui permet à la station de base d'assurer l'équilibrage de

charge en termes d'énergie. Cependant, SHEPR ne supporte pas la mobilité des nœuds.

- **CBCCP (Chain Based Cluster Cooperative Protocol) [21]**

Les auteurs dans [21] ont proposé ce protocole de routage hiérarchique. Après le déploiement aléatoire des nœuds dans le réseau ce protocole commence son traitement en divisant la surface de déploiement en dix (10) sous zones (appelés clusters). Dans ce qui suit, la surface de déploiement est supposée de dimension ( $200m * 200m$ ).

De ce fait, chaque cluster est de dimension ( $200m * 20m$ ) et dans chaque cluster une élection de Cluster Head (CH) et de nœuds coordinateurs de cluster (CCOs) est effectuée. Chaque cluster est composé d'un seul nœud CH, et d'un nombre variant de nœuds CCOs cela dépend du niveau de cluster auxquels ils sont situés.

Le cluster de dimension ( $0 - 200m * 0 - 20m$ ) représente le cluster du premier niveau, où le nœud CH envoie ses données collectées de son cluster au cluster de dimension ( $0 - 200m * 20 - 40m$ ) qui représente le cluster du second niveau. Ce dernier est composé d'un CH et d'un CCO, où ce dernier en recevant les données envoyées par le CH du cluster de premier niveau les envoient à son tour au nœud CCO du prochain cluster qui est de dimension ( $0 - 200m * 40 - 60m$ ). Ce processus continue jusqu'à ce que toutes les données arrivent à la station de base.

CBCCP se compose essentiellement de deux phases suivantes :

- i. Phase d'élection des nœuds CHs et CCOs :

Dans chaque cluster, les nœuds connaissent leurs positions dans le réseau, et sont immobiles, où chaque nœud envoie un message de contrôle à tout le réseau contenant le niveau d'énergie, les nœuds CHs sont élus aléatoirement par rapport au plus grand niveau d'énergie. Chaque cluster contient un seul nœud CH, au total il y aura dix (10) CHs (10 clusters). Puis, les nœuds CCOs seront élus aléatoirement et de la même manière que les CHs, c'est-à-dire par rapport aux plus hauts niveaux d'énergie. Cependant, leurs nombres diffèrent d'un cluster à un autre, en d'autres termes un cluster de niveau  $n$  se compose d'un CH et de  $n - 1$  CCOs.

Après élection des nœuds CHs et CCOs, ces derniers informent les nœuds membres du cluster en diffusant un message de contrôle.

La réélection des nœuds CHs et CCOs sera effectuée, lorsque le niveau d'énergie de ces derniers aura atteint un seuil (Si (le niveau d'énergie de CH/CCO)  $<$  20% (énergie de départ de CH/CCO) alors (réélection CH/CCO)).

ii. Phase de transmission de données :

Après la phase d'élection de nœuds CHs et CCOs, la transmission de données est basée sur le concept de construction des chaînes entre les nœuds (CCOs/CHs) et entre les nœuds CCOs de clusters de niveaux différents, ainsi que l'utilisation du multi-saut. Les nœuds CHs et CCOs du dernier niveau, sont responsables de communiquer l'ensemble de données à la station de base

Comme cité précédemment, les dimensions des clusters restent fixes. Comme le montre la figure 2.7 le nœud CH du premier niveau transmet les données récoltées des nœuds membres de son cluster, les agrège et les envoient au nœuds CCO du cluster de niveau au dessus de lui. Sachant que les nœuds CHs de chaque cluster, sont responsables de collecter les données de leurs clusters, d'effectuer l'agrégation de données et de transmettre les données au nœuds CCOs du cluster de niveau au dessus d'eux. Alors que, les nœuds CCOs reçoivent les données du nœud CH du cluster de niveau en dessous d'eux, les agrègent et les envoient aux nœuds CCOs du cluster de niveau au dessus d'eux.

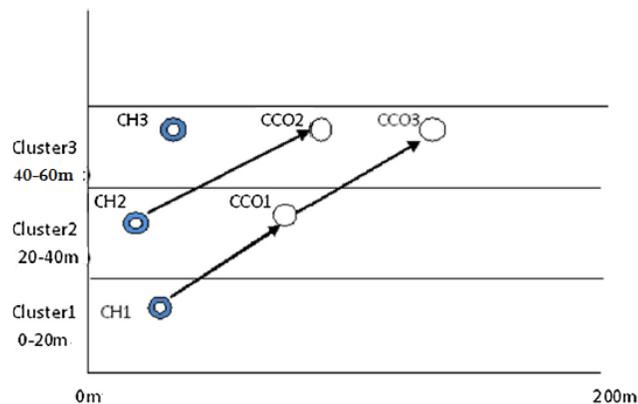


FIGURE 2.7 – Phase de communication du protocole CBCCP [21]

**Discussion :** Dans ce protocole, l'avantage de l'immobilité des nœuds déployés dans le réseau, est que le clustering n'est pas affecté par le changement de positions, sachant que

dans le cas où les nœuds seraient mobiles, ils pourraient dépenser leurs énergies rapidement, ce qui affecterait la durée de vie de tout le réseau.

Dans [21], le protocole CBCCP a été comparé avec le protocole LEACH (vu précédemment). Les résultats de simulation ont montré que, le protocole CBCCP est plus performant que le protocole LEACH en termes de taux de mortalité des nœuds, car dans le protocole CBCCP les nœuds restent vivants bien plus longtemps que ceux dans LEACH. Et cela grâce à plusieurs facteurs tels que [21] :

- Dans le protocole LEACH, l'élection des nœuds Clusters Heads est basée sur une formule bien définie (vu précédemment). Alors que, dans le protocole CBCCP l'élection des nœuds Clusters Heads et des nœuds Coordinateurs de clusters (CCOs) dépend du nombre de clusters ;
- Dans le protocole LEACH, la transmission de données s'effectue en un seul saut. Par contre, en ce qui concerne le protocole CBCCP la transmission de données s'effectue en utilisant le multi-saut.

Les résultats de simulation, ont montré aussi que le protocole CBCCP est aussi plus performant que le protocole LEACH, en termes de consommation énergétique, il permet l'équilibrage de charge car les nœuds consomment un taux équilibré d'énergie tout au long des rounds.

Ce qui permet de prolonger la durée de vie des nœuds, ainsi maximiser la durée de vie du réseau.

- **Tableau comparatif des protocoles la topologie hiérarchique**

Protocole de routage	Efficacité énergétique	Scalabilité	Mobilité	Consommation énergie
LEACH	Très faible	Très faible	SB fixe	Elevée
LEACH-C	Bon	Bon	SB fixe	Elevée
PEGASIS	Faible	Très faible	SB fixe	—
BCDCP	—	Limitée	Non	—
TEEN	Très élevée	Faible	SB fixe	—
APTEEN	Bon	Bon	SB fixe	Elevée
HEED	Mediocre	Mediocre	SB fixe	Bon
VGA	Bon	Bon	Non	Faible
PEDAP-PA	Meilleur	—	—	—
TTDD	Très faible	Elevée	Non	—
GBDD	—	Bon	Limitée	—
SHPER	—	Bon	SB fixe	—
CBCCP	Très bonne	Oui	SB fixe	—

TABLE 2.2 – Tableau comparatif des protocoles de routage de la topologie hiérarchique dans les RCSFs [12, 21]

### 3. Topologie géographique (ou basé-localisation)

Dans ce type de topologie de réseau, les nœuds sont répartis dans le réseau, ils sont surtout connus par leurs positions géographiques dans la zone de déploiement. Ils sont situés principalement au moyen de GPS.

Les protocoles de routage basé-Localisation utilisent les positions des nœuds afin de transmettre les données reçues vers les régions souhaitées du réseau, plutôt que l'ensemble du réseau.

Dans ce type de topologie, les protocoles de routage permettent aux nœuds de trouver un chemin à partir du nœud source vers le nœud destination, en utilisant la distance entre les nœuds voisins par le biais de la puissance du signal reçue à partir de ces nœuds, et les coordonnées sont calculées par l'échange d'informations entre les nœuds voisins.

Cette topologie géographique, permet de minimiser la consommation d'énergie des nœuds. Cependant, la scalabilité est limitée dans le cas où les nœuds sont mobiles, aussi un nœud du réseau doit connaître et apprendre d'avantages sur les positions des autres nœuds [18].

Parmi les protocoles appartenant à cette classe, présentés dans [2], on en cite :

- (a) Geographic Adaptive Fidelity (GAF) ;

- (b) Geographic and Energy-Aware Routing (GEAR);
- (c) Minimum Energy Relay Routing (MERR).

- **GAF (Geographic Adaptive Fidelity)**

GAF est un protocole de routage basé sur la localisation, dont l'objectif est la mise en veille des nœuds inutiles du réseau sans pour autant affecter le niveau de fidélité de l'opération de routage de données, le niveau de fidélité concerne la connectivité persistante des nœuds capteurs communiquant entre eux. Initialement il a été conçu pour les réseaux Ad hoc, néanmoins il peut très bien s'appliquer aux réseaux de capteurs.

Le principe de GAF est de former une grille virtuelle à travers le champ de captage en divisant la zone du réseau en zones fixes, où chaque nœud est affecté à une zone en utilisant ses coordonnées GPS pour se constituer comme un point de la grille virtuelle. De ce fait les nœuds se trouvant dans la même zone de la grille sont considérés équivalents en terme de coûts liés au routage de données, cette équivalence est assurée en utilisant un seul nœud à la fois et de mettre en veille les autres nœuds associés au même point de la grille afin d'optimiser la quantité d'énergie consommée. Ainsi GAF permet de prolonger considérablement la durée de vie du réseau.

Dans GAF, le nœud peut se retrouver dans l'un des trois cas possible :

- (a) Mode découvert, afin de déterminer ses voisins de zone ;
- (b) Mode actif, si le nœud participe dans le routage de données ;
- (c) Mode endormi, si le nœud ne participe pas au routage de données.

La figure (2.8) présentée ci-dessous illustre l'exemple de la grille dans GAF, où le nœud 1 peut atteindre les nœuds 2,3 et 4. Ces derniers peuvent atteindre le nœud 5, donc les nœuds 2, 3 et 4 sont équivalents ainsi deux d'entre eux peuvent être mis en veille.

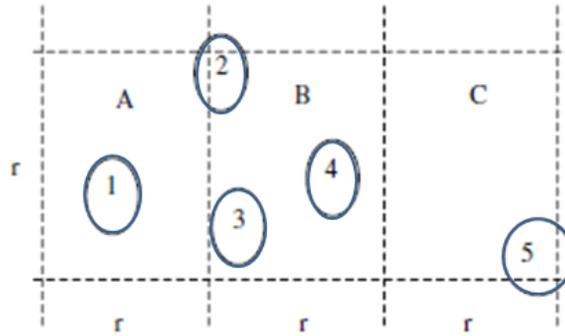


FIGURE 2.8 – Illustration du protocole GAF [2]

**Discussion :** Ce protocole est basé sur la mise en veille des nœuds inutiles, l'inconvénient est qu'il utilise du matériel extra pour trouver l'emplacement des nœuds. Cependant, l'avantage de ce protocole est qu'il économise beaucoup d'énergie grâce au grand nombre de nœuds en sommeil (mis en veille).

GAF est considéré aussi comme protocole de routage hiérarchique vu que le nœud qui est destiné à envoyer les données agit comme un Cluster Head sans qu'il utilise l'agrégation de données.

- **Tableau comparatif des protocoles de la topologie géographique**

Protocole de routage	Consommation d'énergie	Scalabilité	Mobilité	Agrégation de données
GEAR	Limitée	Limitée	Limitée	Non
GAF	Limitée	Elevée	Limitée	Non
MERR	—	Limitée	Faible	Non

TABLE 2.3 – Tableau comparatif des protocoles de routage de la topologie géographique dans les RCSFs [2]

### 2.4.2 Selon le mode de fonctionnement du protocole

Ce mode de fonctionnement définit la manière avec laquelle les données sont propagées dans le réseau, selon ce critère, les techniques de routage de cette classe peuvent être regroupées en quatre catégories :

- Routage basé sur la Qualité de service (QOS "Quality Of Service");
- Routage basé sur les négociations;
- Routage basé sur les multi-chemins;
- Routage basé sur les requêtes.

### 1. Routage basé sur la qualité de service (QOS)

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.) [12].

Le protocole que nous allons présenter dans cette catégorie de routage basé sur la qualité de service (QOS), est le protocole : Sequential Assignment Routing (SAR).

- **SAR (Sequential Assignment Routing) [19]**

Ce protocole est l'un des premiers protocoles dans les RCSFs à avoir introduit la notion de qualité de service QOS (Quality Of Service) dans les décisions de routage. SAR est une approche multi-chemins qui s'efforce de réaliser l'efficacité énergétique et la tolérance au pannes.

Pour cela, le routage dans SAR dépend de trois facteurs :

- Ressources énergétiques;
- Métriques QOS sur chaque chemin;
- Niveau de priorité de chaque paquet.

L'idée du protocole SAR, est de créer des arbres, en utilisant ces arbres, des routes multiples sont formées entre les nœuds et le nœud puits (Sink) pilotée par une table de routage. Une de ses routes est choisie selon les critères énergétiques et les métriques QOS.

En cas de rupture de route, le rétablissement est possible en proposant automatiquement une restauration à partir de la table de routage entre les nœuds ascendants et descendants dans l'arbre.

**Discussion :** La simulation du protocole SAR montre qu'il consomme moins d'énergie, et permet d'emprunter plusieurs routes grâce à la formation de l'arbre. SAR assure la tolérance

aux pannes et la couverture du réseau, il permet de calculer périodiquement les routes à emprunter en cas de défaillance d'un nœud ou de changement de topologie. Malgré ça il souffre du problème d'*overhead* qui est élevé dans la maintenance des tables des nœuds, particulièrement quand le réseau est constitué d'un grand nombre de nœuds.

## 2. Routage basé sur les négociations

Dans cette classe de routage, lorsque le même phénomène est détecté sur le réseau, les nœuds inondent le réseau par les mêmes paquets de données. Ce problème de redondance de données peut être résolu en utilisant des protocoles de routage basés sur la négociation.

En effet, cette classe de protocole utilise des descripteurs de haut niveau appelés aussi des métadonnées, c'est-à-dire, avant toute transmission les nœuds négocient entre eux leurs données en échangeant les métadonnées, ces paquets permettent de vérifier si les nœuds voisins disposent déjà de la donnée à transmettre ou pas. Ces protocoles de routage permettent de garantir que seules les données utiles seront transmises sur le réseau et élimine la redondance de données [12].

Parmi les protocoles appartenant à cette catégorie de routage :

- (a) Sensors Protocols for Information via Negotiation (SPIN) ;
- (b) SPIN for Point to Point communication (SPIN-PP) [18] ;
- (c) SPIN with Energy Conservation (SPIN-EC) [18] ;
- (d) SPIN for Broadcast Networks (SPIN-BC) [18] ;
- (e) SPIN with Reliability (SPIN-RL) [18].

- **SPIN (Sensors Protocols for Information via Negotiation) [14]**

Heinzelman et al, ont proposé ce protocole basé sur les négociations SPIN. Ce dernier est l'un des premiers protocoles utilisant le routage centré-données (Data-centric)<sup>1</sup> développés pour les RCSFs, il repose sur deux idées fondamentales :

- Premièrement, pour fonctionner efficacement et conserver de l'énergie, les capteurs ont besoin de communiquer les uns avec les autres sur les données qu'ils ont déjà, et des données dont ils ont besoin d'obtenir ;

---

1. Dans le routage centré-données, le nœud puits envoie des requêtes à certaines régions et attends les données correspondantes aux requêtes provenant des nœuds situés dans les régions sélectionnés [2]

- Deuxièmement, les nœuds dans le réseau doivent surveiller et s'adapter aux changements dans leurs propres ressources énergétiques afin de prolonger la durée de vie du réseau.

L'idée principale du protocole SPIN est de nommer les données à l'aide de descripteurs de haut niveau ou des métadonnées. Il utilise les négociations métadonnées afin de réduire les transmissions redondantes dans le réseau. Avant la transmission, ces métadonnées sont échangées entre les nœuds par un mécanisme de publicité, ou bien une annonce, qui est la caractéristique principale du protocole SPIN.

Le protocole SPIN utilise essentiellement trois types de messages (ADV, REQ, et DATA).

Tel que :

- **ADV** : Message qui contient une description des métadonnées ;
- **REQ** : Message permettant de demander des données spécifiques ;
- **DATA** : Message qui contient les données.

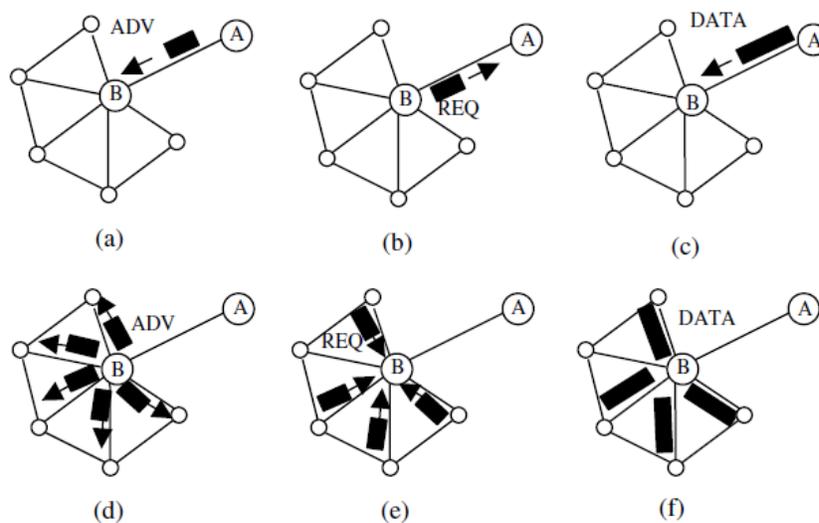


FIGURE 2.9 – Illustration du protocole SPIN [14]

Les communications dans SPIN comme le montre la figure 2.9, se font comme suit :

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

- En recevant un message REQ, le nœud ayant initié la communication transmet à l'intéressé la donnée sous forme d'un message DATA.

SPIN offre une évolutivité dans un sens que chaque nœud doit connaître que ses voisins ceux avec qui il communique, de sorte que, tout changement dans la topologie serait local.

**Discussion :** Ce protocole résout le problème d'implosion rencontré dans la technique Flooding, en utilisant les négociations et l'adaptation des ressources, l'avantage est que lorsque la topologie du réseau change, elle est détectée car chaque nœud connaît son voisin à un seul saut, les problèmes dont souffre SPIN est que son mécanisme de livraison ne garantie pas toujours la livraison de données car si les nœuds qui sont intéressés par cette donnée sont situés loin géographiquement et que les nœuds intermédiaires entre le nœud source et le nœud puits ne sont pas intéressés par cette données donc la livraison ne sera pas établie, Ce protocole provoque une surcharge (overhead) du réseau à cause de l'envoi constant des publicités de données.

### 3. Routage basé sur les Multi-Chemins

Dans cette classe de routage, les protocoles de routage permettent l'utilisation de chemins multiples plutôt qu'un chemin unique pour le routage de données jusqu'à destination afin d'augmenter les performances du réseau.

La fiabilité de protocoles de routage basé sur les multi-chemins peut être mesurée par la capacité à trouver des chemins alternatifs entre la source et la destination en cas de défaillance du premier chemin. Pour cette raison certains protocoles de routage construisent plusieurs chemins indépendants, c'est-à-dire, ces chemins ne partagent d'un nombre réduit (voir nul) de nœuds.

Malgré leur grande tolérance aux pannes, ces protocoles requièrent plus de ressources énergétiques et plus de message de contrôle pour le routage de données [12].

Les protocoles de routage que nous allons présenter, appartenant à cette catégorie de routage basé sur les Multi-Chemins, sont :

- (a) Directed Diffusion (DD) ;
- (b) Rumor Routing (RR) ;

(c) Gradient Based Routing (GBR).

- **DD (Directed Diffusion) [24]**

Ce protocole de routage basé sur les multi-chemins fait partie de la catégorie des protocoles Centré-Données (Data-centric), permettant d'utiliser plusieurs chemins pour le routage d'informations. Le fonctionnement de DD se compose principalement des opérations suivantes :

- La propagation d'intérêts ;
- La propagation de données ;
- Le renforcement de routes.

La station de base (SB) envoie des messages en diffusant un intérêt à tous les nœuds, qui représente une description des tâches affectées à tous les nœuds du réseau (figure 2.10 (a)). Ces descripteurs englobent une paire (attribut-valeur), qui décrivent les tâches requises par la SB.

Durant le processus de diffusion d'intérêts, des gradients à partir du nœud source de données vers la SB seront déterminés pour créer des routes pour le transfert des données capturées (figure 2.10 (b)) l'une de ses routes est par la suite renforcée pour être utilisée.

Lorsque les données seront disponibles au niveau du nœud source, il les envoie à la SB suivant la route renforcée, elle représente la route ayant le gradient fort pour l'intérêt diffusé, comme illustré dans la figure 2.10 (c).

Pour la réduction des coûts de communications, les données peuvent être agrégées par les nœuds intermédiaires qui se trouvent entre la station de base et le nœud intéressé par l'intérêt, Dans le but qu'il y aura un arbre d'agrégation qui permet l'acheminement des données de la source jusqu'à la destination (SB).

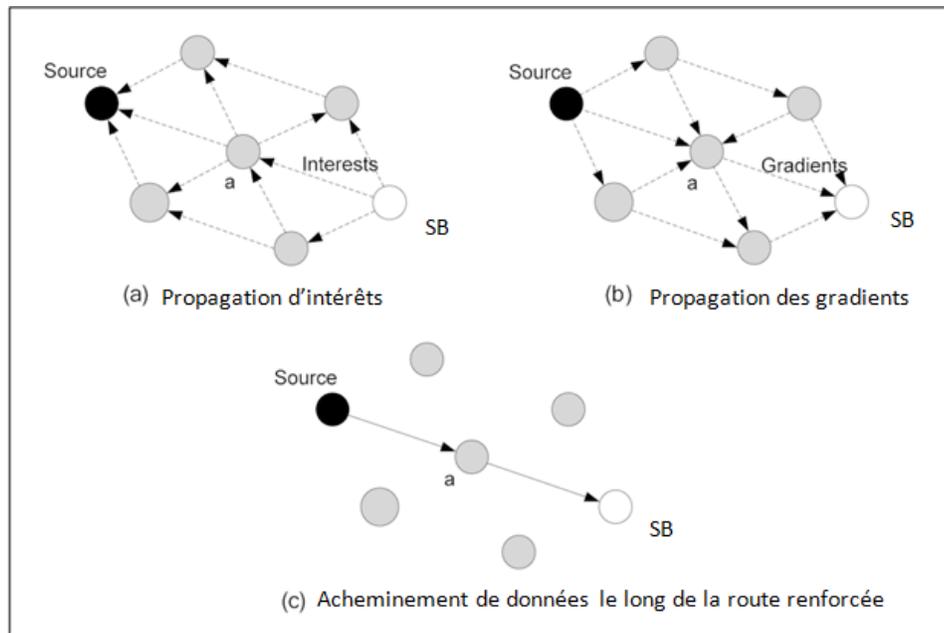


FIGURE 2.10 – Illustration du fonctionnement du protocole DD [24]

**Discussion :** Le protocole de routage Directed Diffusion est Centré-Données, où toutes les communications sont voisin à voisin sans avoir besoin du mécanisme d'adressage des nœuds, il améliore la robustesse du réseau face à des échecs de routes provoqués par le changement fréquent de topologie.

Bien que ce protocole permette d'obtenir une certaine économie d'énergie, il a aussi des problèmes. Par exemple, pour mettre en œuvre l'agrégation des données, il utilise la technique de synchronisation de temps, ce qui est difficile à réaliser dans un réseau de capteurs.

#### 4. 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'envoi 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 [18].

Parmi les protocoles de routage appartenant à cette catégorie dans [2] :

- (a) COUGAR;
- (b) ACTive QUery forwarding In sensoR nEtworks (ACQUIRE).

- **COUGAR**

COUGAR fait partie de la catégorie des protocoles Centré-Données (Data-centric), ce protocole propose une architecture qui considère le réseau comme un immense système de Base de Données Distribuées (BDD), où les nœuds choisissent un nœud appelé "nœud leader" pour effectuer les opérations d'agrégations et de transmissions de données détectées vers le nœud puits (Sink). L'architecture est illustrée dans la figure 2.11 ci dessous.

Pour pouvoir interroger les nœuds du réseau. COUGAR fournit une interface au nœud puits, cette interface lui permet de générer des requêtes afin d'interroger les nœuds, ces requêtes spécifient les informations nécessaires sur les données ou bien les données produites par les nœuds.

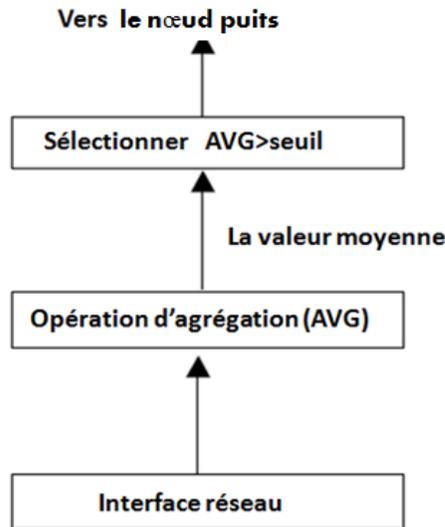


FIGURE 2.11 – Le plan de requête dans le nœud leader dans le protocole COUGAR [2]

**Discussion :** Bien que cette architecture permet de garantir une grande efficacité en consommation d'énergie. Cependant COUGAR souffre de certains problèmes, comme :

- L'ajout d'une couche additionnelle consacrée aux traitements des requêtes au niveau de chaque nœud introduit un *overhead* supplémentaire en terme de consommation d'énergie

et de stockage ;

- Les opérations de traitement intermédiaire telle que l'agrégation nécessitent une certaine synchronisation au niveau du nœud leader qui doit attendre l'arrivée de tous les paquets à partir des nœuds sources afin d'effectuer le traitement et envoyer le résultat au nœud puits ;
- Les nœuds leaders doivent être maintenus de façon dynamique pour éviter leurs pannes.

- **Tableau comparatif des protocoles de routage selon le mode de fonctionnement du protocole**

Protocole de routage	Agrégation de données	Scalabilité	Mobilité	Consommation d'énergie
SAR	Oui	Limitée	Non	Elevée
SPIN	Oui	Limitée	—	Limitée
SPIN-PP	—	Elevée	Oui	—
SPIN-EC	—	Elevée	Oui	—
SPIN-BC	—	Elevée	Oui	—
DD	Oui	Elevée	Limitée	Limitée
RR	Oui	Elevée	Faible	Faible
GBR	Oui	Limitée	—	Faible
COUGAR	Oui	Limitée	—	Limitée
ACQUIRE	Oui	Limitée	—	Faible

TABLE 2.4 – Tableau comparatif des protocoles de routage selon le mode de fonctionnement du protocole dans les RCSFs [18, 20]

### 2.4.3 Selon l'établissement de routes

Traditionnellement, les protocoles de routage se divisent en trois familles : les protocoles proactifs, qui anticipent la demande d'acheminement des données, les protocoles réactifs qui réagissent à une demande d'acheminement, et enfin la famille des protocoles hybrides qui est une combinaison des propriétés des protocoles proactifs et réactifs.

- **Protocoles proactifs :**

Ces protocoles de routage essaient de maintenir les meilleurs chemins existants vers toutes les destinations possibles au niveau de chaque nœud du réseau. Les routes sont sauvegardées même si elles ne sont pas utilisées. Chaque nœud du réseau maintient une table de routage pour toutes les destinations indépendamment de l'utilité des routes. Les protocoles proactifs sont adaptés aux applications qui nécessitent un prélèvement périodique des données. Et par conséquent, les capteurs peuvent se mettre en veille pendant les périodes d'inactivité, et n'enclencher leur dispositif de capture qu'à des instants particuliers [12].

- **Protocoles réactifs :**

Ces protocoles (dits aussi, les protocoles de routage à la demande) créent et maintiennent des routes selon les besoins. Lorsque le réseau a besoin d'une route, une procédure de découverte de routes est lancée. Ce type de protocoles est pratique pour des applications à temps réel où les capteurs doivent réagir immédiatement à des changements soudains des valeurs captées. En effet, un prélèvement périodique des données aurait été inadapté pour ce type de scénarios [12].

- **Protocoles hybrides :**

Ces protocoles combinent les deux idées des protocoles proactifs et réactifs. Ils utilisent un protocole proactif pour le routage des données dans le voisinage, c'est à dire entre les nœuds voisins, (par exemple le voisinage à deux ou à trois sauts), ainsi, ils disposent de routes immédiatement dans le voisinage. Au-delà de la zone du voisinage, le protocole hybride fait appel à un protocole réactif pour chercher des routes [12].

#### 2.4.4 Selon l'initiateur de communication

La communication dans les réseaux de capteurs est initiée soit par les nœuds sources ou par le biais de la station de base

1. **Communication initiée par le nœud source**

Dans ce type de communication, c'est les nœuds qui détectent et collectent les données dans le réseau et les transmettent à la station de base, tout en essayant d'assurer une efficacité énergétique afin de pouvoir prolonger la durée de vie de tout le réseau.

2. **Communication initiée par le nœud destination**

Dans ce type de communication, le nœud destination représente la station de base c'est elle qui gère les transmissions de données dans le réseau, en envoyant des requêtes aux nœuds du réseau, ces derniers répondent aux requêtes envoyés par la station de base. Autrement dit, la station de base décrit aux nœuds ce qu'ils doivent transmettre comme données. Afin de réduire les transmissions inutiles, mais les requêtes de grandes tailles pourraient engendrer un affaiblissement du niveau d'énergie de la station de base.

## 2.5 Conclusion

Dans ce chapitre nous avons présenté l'état de l'art des protocoles de routage proposés dans la littérature, en présentant les différents facteurs de conception des protocoles de routage dans les RCSFs, puis nous avons fournis une classification suivant les différentes approches adoptées qui nous a permis de voir les différents protocoles de routage ainsi que leurs différents avantages et inconvénients. L'étude de ces protocoles va nous permettre dans le chapitre qui suit de concevoir un nouveau protocole de routage hiérarchique pour les RCSFs et qui a pour but de minimiser la consommation d'énergie ce qui permet d'augmenter la durée de vie du réseau.

## Chapitre 3

# Notre contribution : Routage avec efficacité énergétique basé sur une arborescence selon les nœuds frontières de la portée dans les RCSFs (R2E-ANFP)

---

### 3.1 Introduction

Dans ce chapitre, nous allons présenter notre contribution dans le domaine des RCSFs.

Dans les protocoles de routage hiérarchiques des réseaux de capteurs sans fil, le réseau est découpé en groupes appelés Clusters, les nœuds membres du cluster ne transmettent pas les données directement à la station de base, mais d'abord ils les transmettent à leurs nœuds Cluster Head respectifs, ensuite ce dernier les agrège et envoie le résultat directement à la station de base distante ou en utilisant le routage multi-sauts. Tout ceci permet d'affaiblir les nœuds membres du cluster ainsi que les nœuds CHs.

L'objectif de notre travail, est de concevoir un protocole de routage hiérarchique qui permet d'avoir une faible consommation d'énergie à travers tout le réseau, tout en garantissant une durée de vie plus longue du réseau.

Nous commençons par décrire notre proposition en décrivant toutes les phases qui la constituent, et les nouveautés apportées par ce dernier, puis nous présentons un organigramme qui représente une architecture de fonctionnement de notre proposition ainsi que son concept de base.

## 3.2 Description de la proposition

R2E-ANFP (Routage avec efficacité énergétique basé sur une arborescence selon les nœuds frontières de la portée dans les RCSFs), est un protocole de routage centralisé, où l'élection du Cluster Head se fait au niveau de la station de base, lorsque les clusters sont formés ses derniers se composent principalement de trois types de nœuds capteurs (Cluster Head, Sous Cluster Head, nœuds capteurs) que nous définirons par la suite.

Afin de minimiser l'énergie consommée par les nœuds du réseau le protocole s'appuie sur le routage multi-sauts effectués entre le CH et les différents sous CH qui composent le réseau.

**Définition 3.2.1.** Les Cluster Heads (CHs) et les sous Cluster Heads (sous CHs) sont les représentants des clusters. En plus de ça, les tâches qui leurs sont affectées sont l'agrégation de données collectées, ainsi que l'organisation du routage dans le réseau en utilisant l'approche multi-sauts.

Dans notre protocole, il existe une légère différence entre le CH et les sous CHs, le premier Cluster Head est élu par la station de base et c'est lui qui communique toutes les données à la station de base, alors que les sous-CHs sont élus soit par le premier CH ou bien par sous CH de niveau inférieur, et ces derniers sont situés aux frontières du rayon de déploiement dans la phase de formation des clusters que l'on verra par la suite.

Notre protocole de routage se compose principalement de trois phases :

- **Phase d'élection du CH :**

Les nœuds sont répartis de manière aléatoire dans le réseau. A chaque nouveau round, la station de base (SB) effectue une diffusion d'un message de contrôle à un rayon  $r$ , afin que ce message atteigne 30% de la largeur de la surface de déploiement, voir figure 3.2 :

Les nœuds de la zone délimitée par le rayon  $r$ , appelée Zone1, répondent à la SB. Cette dernière va générer une table contenant les identifiants des nœuds, la puissance du signal reçue ainsi que

l'énergie associée à chaque nœud appartenant à cette Zone1. Nous distinguons deux cas 3.1 :

- (a) Si c'est le premier round, le Cluster Head (CH) qui sera élu par la SB est le nœud ayant le signal le plus fort, autrement dit, le nœud le plus proche de la SB, cette dernière informe le nœud élu CH en envoyant un message de contrôle.
- (b) Si c'est les rounds qui suivent le premier round, le CH qui sera élu dans le round courant par la SB est le nœud ayant le niveau d'énergie le plus élevée, La SB va informer ce nœud qu'il est élu CH en envoyant un message de contrôle.

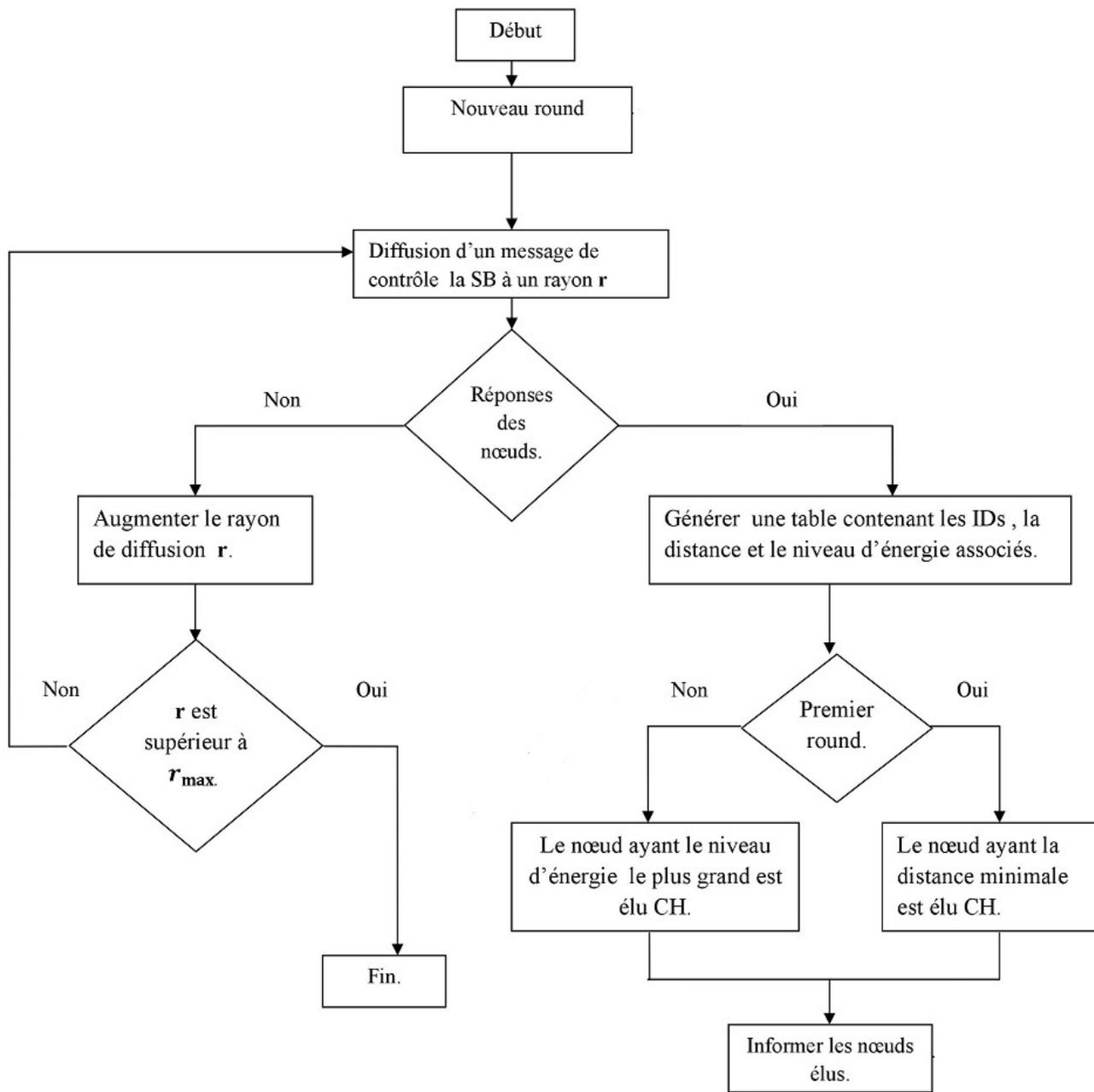


FIGURE 3.1 – Organigramme de la phase d’élection du CH

Dans le cas, où la SB ne reçoit aucune réponse de la Zone1 elle doit augmenter son rayon de diffusion  $r$  au fur et à mesure qu’elle reçoit des réponses des nœuds, puis refait les mêmes étapes jusqu’à ce qu’il y ait aucun nœud dans le réseau.

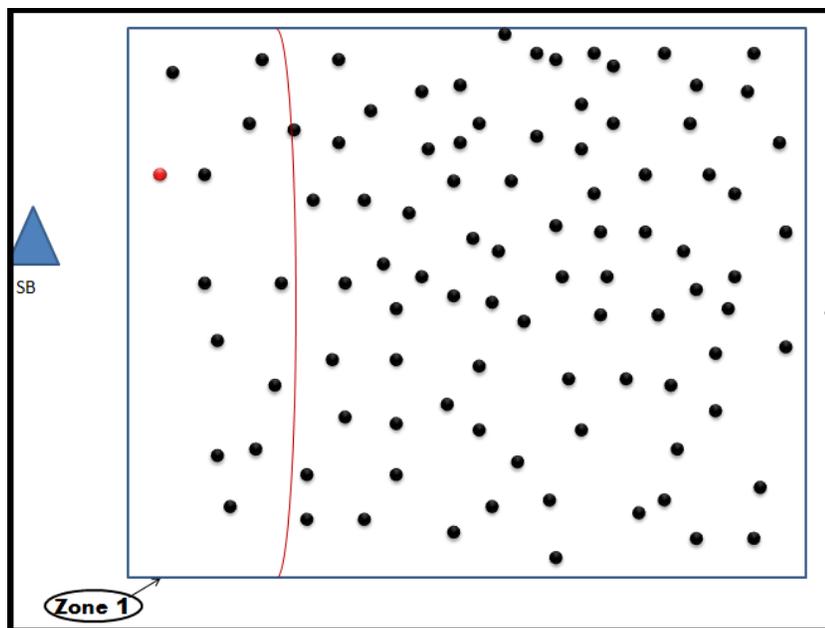


FIGURE 3.2 – Phase d'élection du CH

- **Phase de formation de Cluster :**

Après l'élection du CH dans la Zone1, ce dernier va pouvoir former son cluster et élire les sous CHs de premier niveau en effectuant une émission d'un paquet de contrôle à un rayon  $r$  de distance égale à  $25m$ , ainsi les nœuds ayant reçue le message répondent en envoyant leurs IDs. Lorsque le CH reçoit ses réponses, il construit une table contenant les IDs des nœuds ainsi que la puissance du signal reçue pour chaque nœud. Les nœuds qui se trouvent à une distance inférieur à 60% de  $r$  sont considérés comme étant des nœuds normaux (ordinaires), or pour les nœuds qui se trouvent à une distance supérieur sont élus sous CHs par le CH. A la fin de cette phase le CH émet un message contenant les statues de chaque nœud du cluster (voir organigramme 3.3).

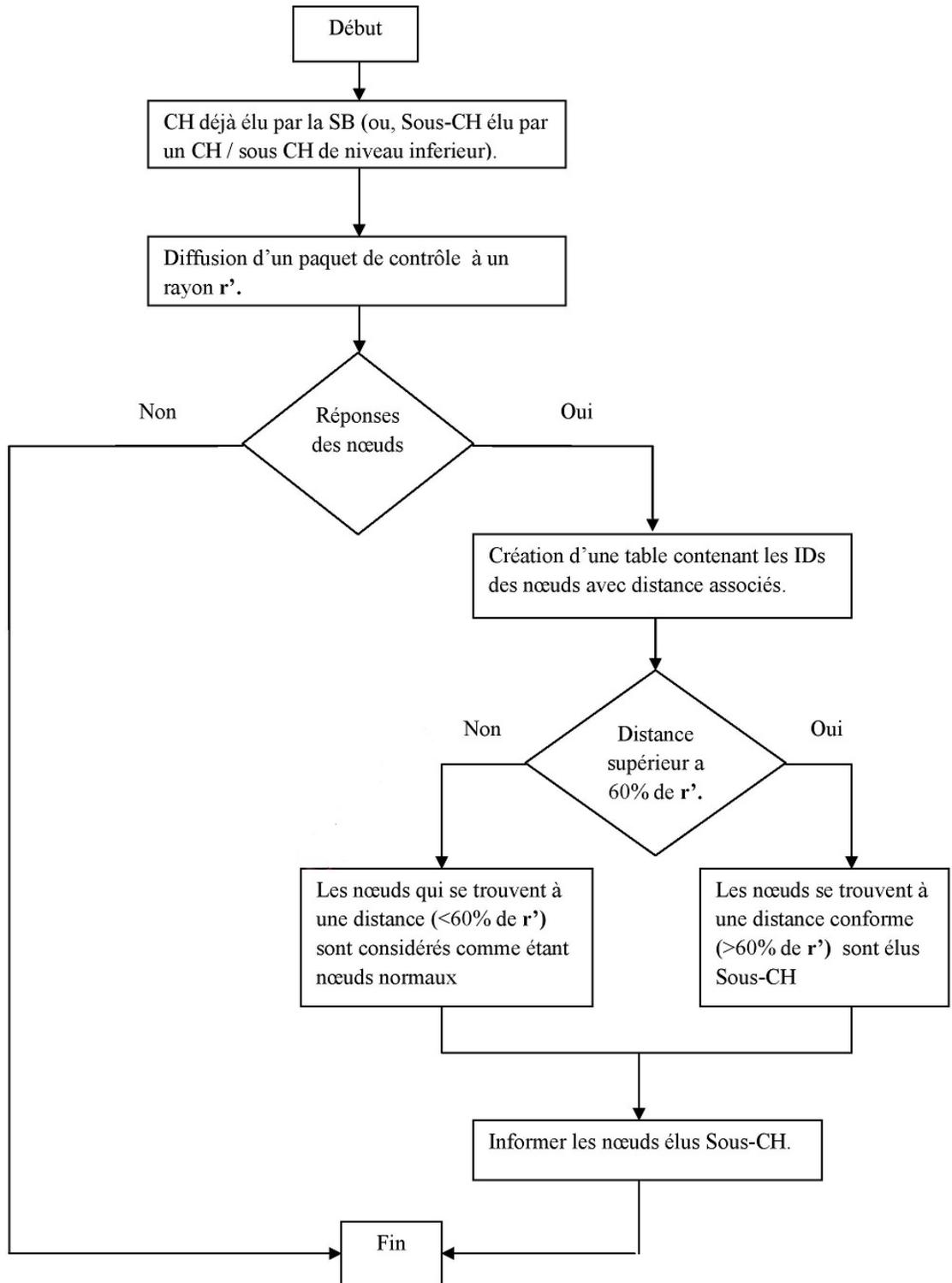


FIGURE 3.3 – Organigramme de la phase de formation de clusters

Les sous CHs vont refaire exactement la même procédure exécutée par le CH afin de former leurs propres clusters. Ainsi de suite, jusqu'à structurer tout le réseau. Comme le montre la figure 3.4

ci-dessous, où le nœud rouge représente le nœud CH, les nœuds vert représentent les nœuds Sous CHs du premier niveau, les nœuds violets sont les nœuds Sous CH du second niveau, les nœuds jaunes sont les Sous CHs du troisième niveau, les nœuds oranges représentent les Sous CH du quatrième niveau, et enfin les nœuds roses représentent les Sous CHs du cinquième niveau (le dernier niveau dans ce cas).

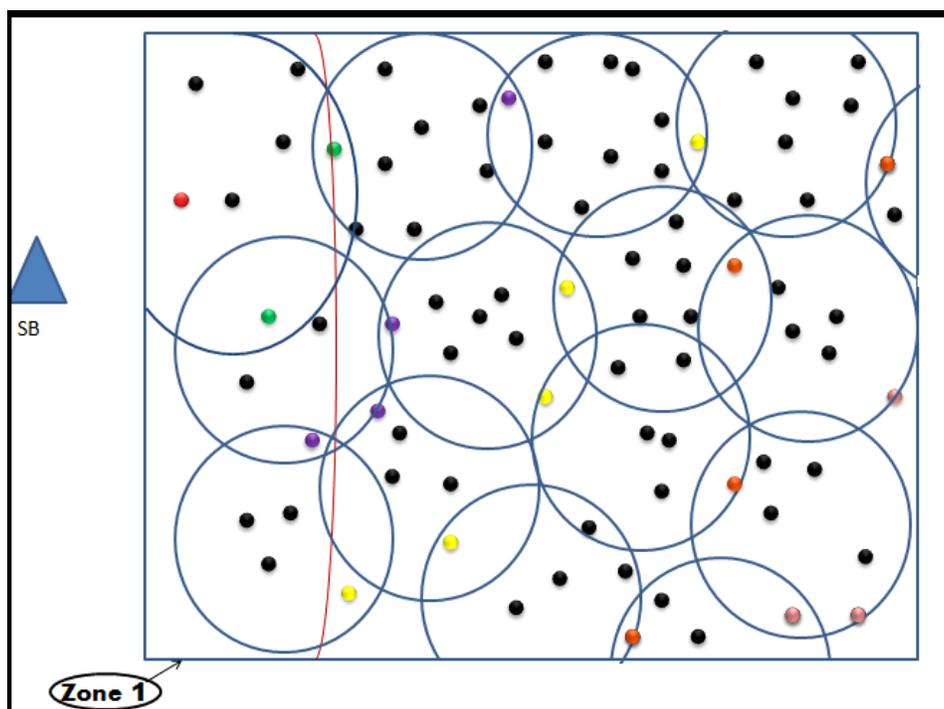


FIGURE 3.4 – Phase de formation de Cluster

- **Phase de communication de données :**

Durant cette phase, nous avons deux types de communications inter-cluster et intra-cluster 3.5 :

(a) Communication inter-cluster, voir figure 3.6(a) :

Chaque nœud normal ayant collecté des données sur le réseau, les envoient soit au CH ou au Sous CH de son cluster ;

(b) Communication intra-cluster, voir figure 3.6(b) :

Lorsque le nœud Sous CH reçoit les données collectées dans son cluster, il effectue également l'agrégation de données et envoie ce résultat d'agrégation au sous CH de niveau inférieur.

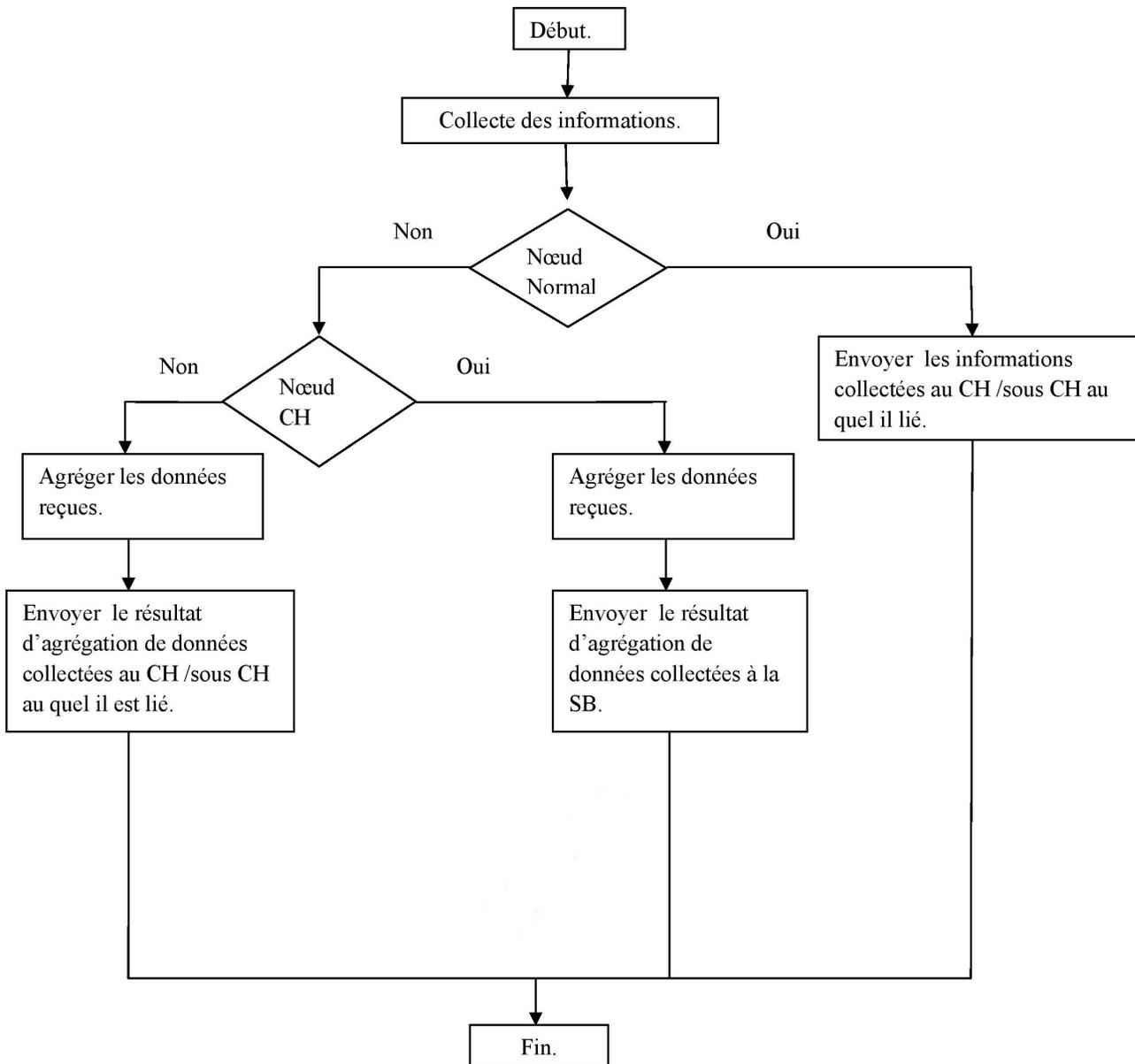


FIGURE 3.5 – Organigramme de la phase de communication de données

au début nous avons des communications inter-cluster, qui ont pour but de rassembler les données locales des différents clusters. Après ses données circulent entre les différents clusters selon un arbre de routage tracé jusqu'à atteindre le nœud CH, et lorsque ce dernier reçoit les données, il effectue l'opération d'agrégation sur ses données, et envoie le résultat à la station de base voir figure 3.6 ;

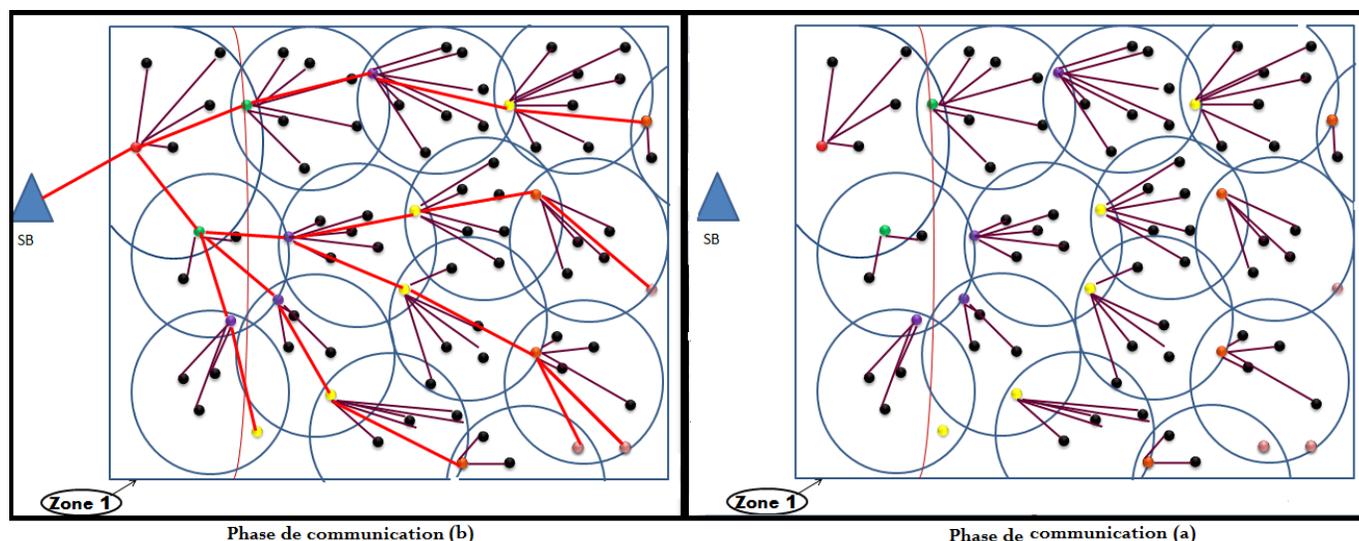


FIGURE 3.6 – Phase de communication de données

A la fin de cette phase, qui représente la fin d'un round et le début d'un autre.

### 3.3 Cas particuliers de notre protocole

Lors de l'exécution de notre protocole, nous pouvons remarquer les cas particuliers suivants :

#### 1. Cas de nœuds isolés :

A chaque début de round, la station de base émet un message de contrôle à travers tout le réseau, afin de prévenir les nœuds que c'est le début de round.

Lors de la phase de formation de clusters, nous pouvons avoir des nœuds qui n'appartiennent à aucun cluster, que ça soit un cluster formé par les nœuds (CH ou Sous CHs) voir figure 3.7.

Dans ce cas, et après un temps d'attente les nœuds n'ayant pas reçu de messages de contrôle de la part de sous CHs ou du CH, sont considérés comme nœuds isolés et décident de joindre le cluster le plus proche formé par les nœuds (CH ou Sous CHs), comme le montre la figure 3.8.

Pour ce faire, les nœuds isolés vont émettre un message de contrôle afin de joindre le cluster le plus proche et deviendront des nœuds membres du CH ou Sous CH le plus proche.

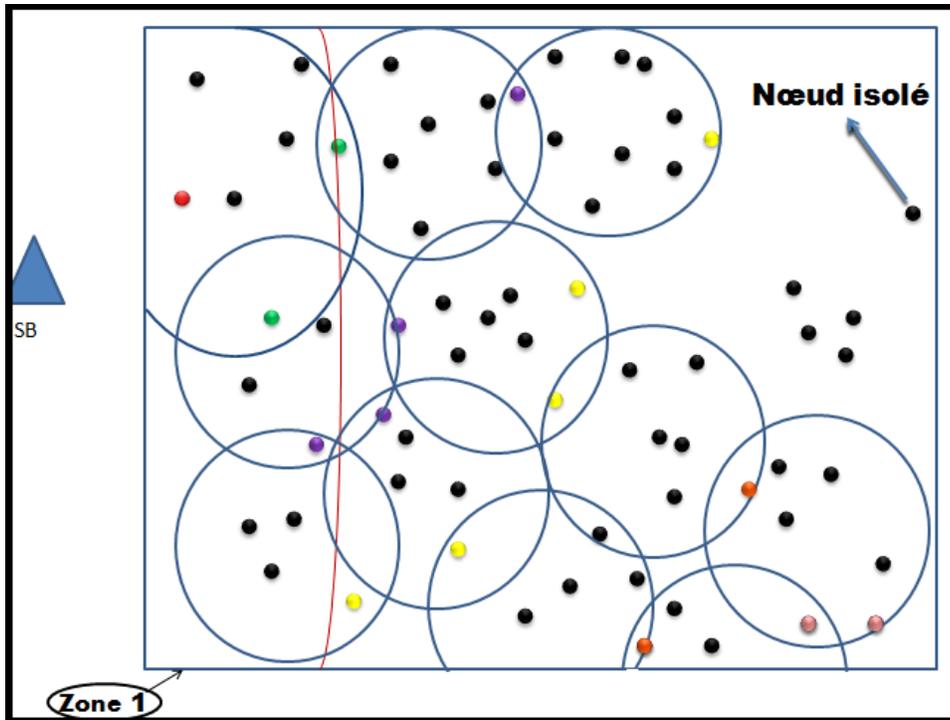


FIGURE 3.7 – Cas du noeud isolé

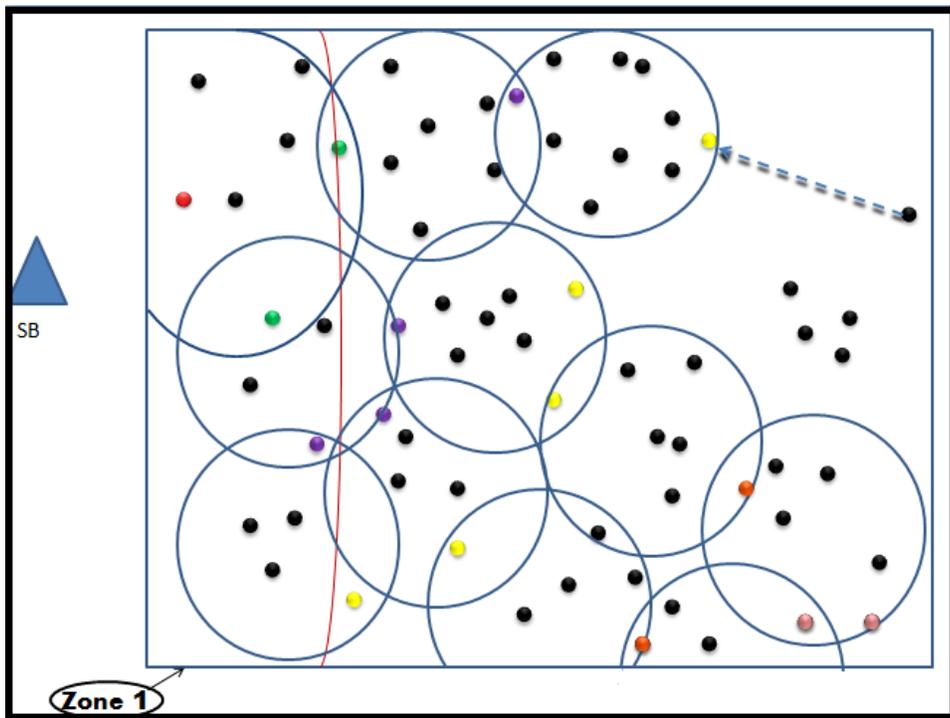


FIGURE 3.8 – Noeud isolé rejoint le cluster le plus proche

2. Cas de la mort des nœuds de la zone(1) :

Au début de chaque round, la station de base émet un message de contrôle dans la zone(1) délimitée par son rayon de diffusion, cette zone sera considérée comme zone morte dans le cas où, la station de base ne reçoit aucune réponse à son message de contrôle. Ce la veut dire, qu'aucun nœud n'est actif dans cette zone(1).

Dans ce cas, la station de base double son rayon de diffusion pour créer la zone(1) et commencer l'élection du nœud Cluster Head comme le montre la figure 3.9. Par la suite, les phases de notre protocole seront exécutées.

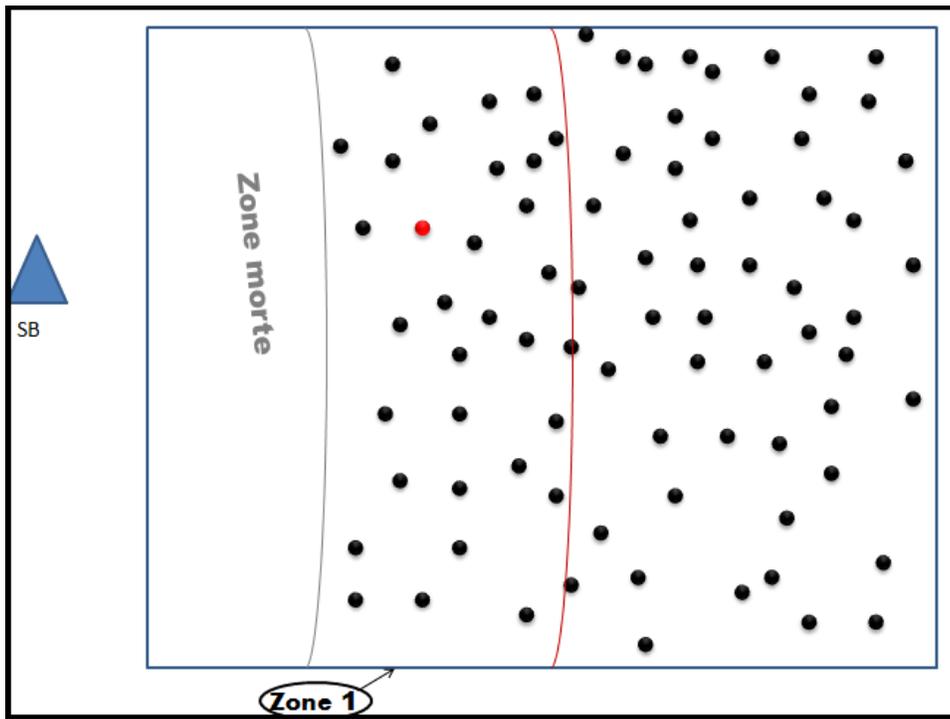


FIGURE 3.9 – Cas de la zone morte

### 3.4 Conclusion

Dans ce chapitre, nous avons présenté le protocole R2E-ANFP dans les réseaux de capteurs sans fil en le décrivant. Ce protocole permet l'utilisation du routage multi-sauts et l'agrégation de données, ainsi que l'utilisation de trois types de nœuds dans le réseau, et définit pour chacun des types un rôle dans le routage de données, et cela afin d'économiser l'énergie des nœuds, et prolonger la durée de vie du réseau.

Nous avons aussi schématisé les phases qui constituent R2E-ANFP sous forme d'organigramme afin de plus le détailler.

Notre objectif est de prouver les performances du protocole R2E-ANFP, au niveau énergétique et durée de vie du réseau. De ce fait, dans le chapitre qui suit nous allons implémenter et simuler ce dernier et conclure avec une comparaison par rapport à trois autres protocoles de routage hiérarchiques.

# Chapitre 4

## Evaluation des performances

---

### 4.1 Introduction

Dans le chapitre précédent, nous avons décrit notre protocole et détailler toutes les phases qui le constituent.

Dans ce chapitre, nous allons présenter les résultats de simulations du protocole R2E-ANFP obtenus grâce au simulateur développé, ces résultats constituent l'étape la plus importante de notre travail, puisqu'ils nous permettent d'évaluer ses performances en termes d'économie d'énergie et de durée de vie du réseau.

De ce fait, les résultats fournis par la simulation sont comparés aux trois autres protocoles LEACH, MH-LEACH et CBCCP en termes de durée de vie du réseau et consommation d'énergie.

### 4.2 Outils et langages de développement

Nous avons développé notre simulateur en utilisant l'outil de développement Netbeans, ainsi que le langage de programmation JAVA.

### 4.2.1 JAVA

JAVA est un langage de programmation informatique orienté objet, développé par Sun Microsystems, il fut présenté officiellement en 1995. Le langage JAVA reprend en grande partie la syntaxe du langage C++, très utilisé par les informaticiens. Il permet de développer des applications, des sites web, des jeux, etc [8].

On peut faire de nombreux types de programmes avec JAVA, comme par exemple [7] :

- Des applications sous formes de fenêtres ou de console ;
- Des applets, qui sont des programmes JAVA incorporés à des pages web ;
- Des applications pour appareils mobiles ;
- Des sites web dynamiques ;
- etc.

JAVA présente plusieurs avantages pour son utilisation, tels que [8] :

- Simple, car le code utilisé est proche de celui du C++ il permet de faciliter la compréhension du système ;
- Fiable, puisque le compilateur Java détecte de nombreux problèmes qui, dans d'autres langages, ne sont visibles qu'au moment de l'exécution ;
- Portable, les logiciels écrits dans ce langage sont facilement portables sur plusieurs systèmes d'exploitation tels que, UNIX, Windows, Mac OS, etc.

### 4.2.2 NetBeans

Netbeans est un environnement de développement intégré (en anglais : Integrated Development Environment (IDE)) open source racheté et développé par Sun Microsystems. C'est un outil qui permet aux programmeurs d'écrire, de compiler des programmes. En plus de JAVA, Netbeans permet l'utilisation de plusieurs autres langages de programmation, comme (C, C++, HTML, Javascript, etc.)

L'avantage d'utiliser Netbeans est qu'il peut fonctionner sous Windows, Linux, MACOS X, Solaris, etc. Et son utilisation représente un gain de temps en développement grâce à sa boîte à outils [5].

### 4.3 Environnement de simulation

Afin d'effectuer nos simulations, nous avons utilisé :

- Le langage de programmation (JAVA) ;
- L'outil de développement Netbeans version (8.1).

Notre simulateur est composé de plusieurs fenêtres que nous allons détailler :

- **Fenêtre principale** (voir figure 4.1) : elle permet d'introduire les variables de simulation telles que (nombre de nœuds, surface de déploiement, énergie initiale des nœuds), et nous oriente vers d'autres fenêtres spécifiques selon l'expérimentation voulu.

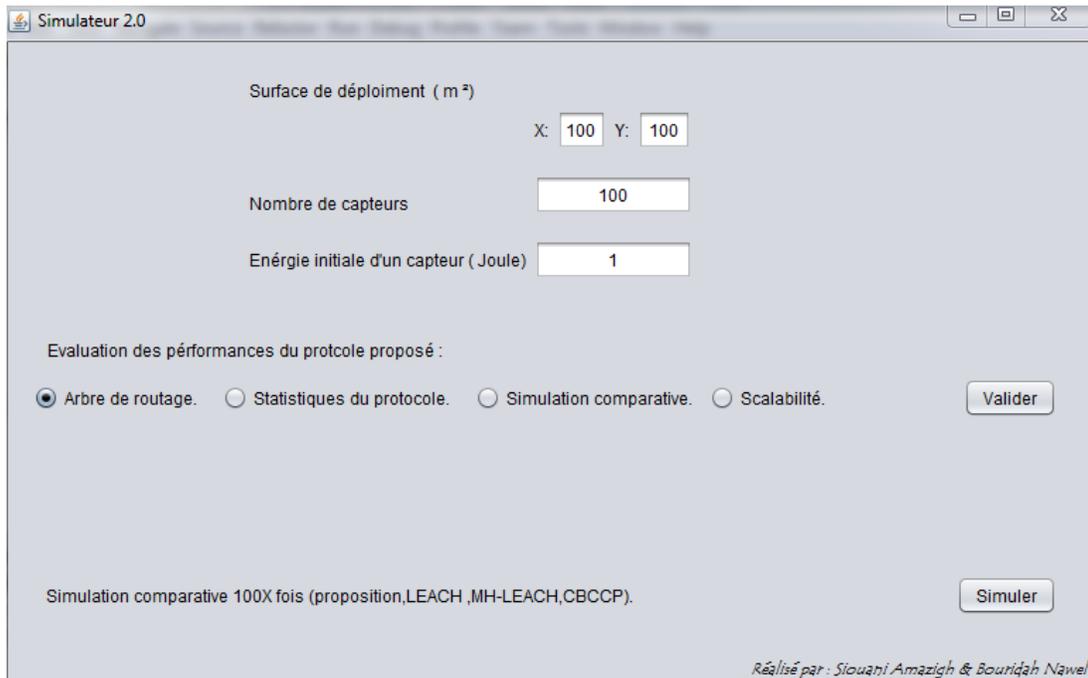


FIGURE 4.1 – Fenêtre principale (accueil)

- En choisissant "*arbre de routage*" de la fenêtre principale et en validant, nous obtenons :
  - **Fenêtre "arbre de routage"** (voir figure 4.2) :  
notre modèle de simulation, est établi sur 100 nœuds déployés aléatoirement sur une surface de déploiement carré de dimensions  $(100 * 100)m^2$ .  
Cette surface de déploiement, est considérée comme étant le réseau pour l'évaluation de notre protocole de routage, où :

- (a) Le round représente la durée d'exécution de notre protocole depuis la première phase (élection des CH/ sous CHs) jusqu'à la dernière phase (communication de données), c'est-à-dire jusqu'à ce que les données collectées arrivent à la station de base ;
- (b) La durée de simulation est en fonction de rounds ;
- (c) Les nœuds sont fixe ;
- (d) La station de base (SB) est immobile, et elle est située loin du réseau et a comme coordonnées (50, 175).

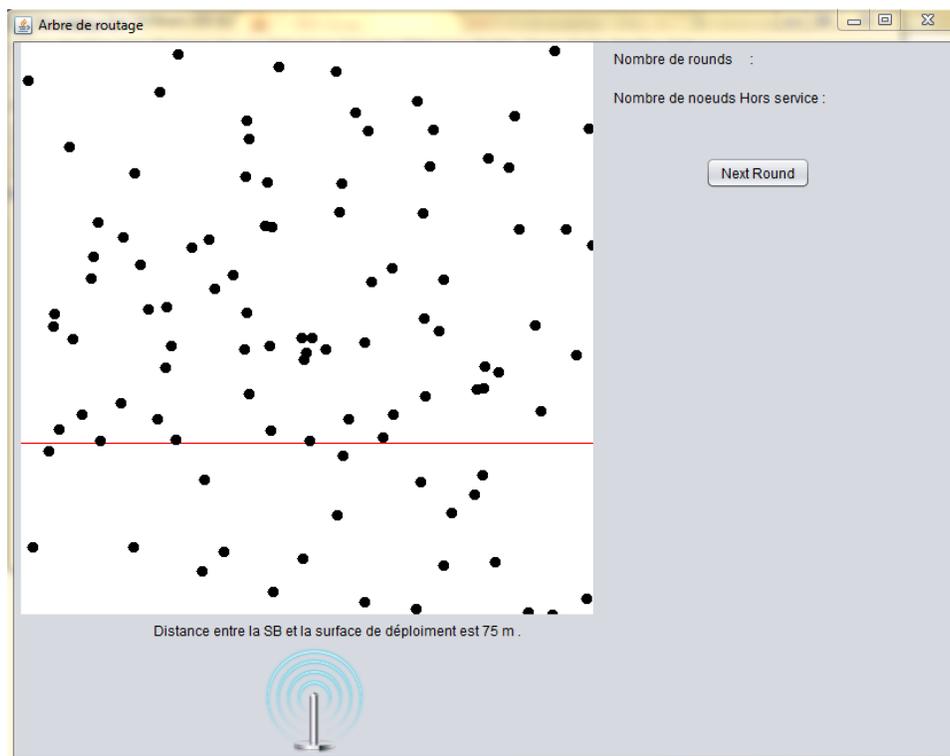


FIGURE 4.2 – Fenêtre "arbre de routage"

En cliquant sur le bouton (Next round) de la fenêtre "arbre de routage", notre protocole de routage sera exécuté dans les différentes phases qui le compose, alors l'arbre de routage du round actuel sera affiché (voir figure 4.3).

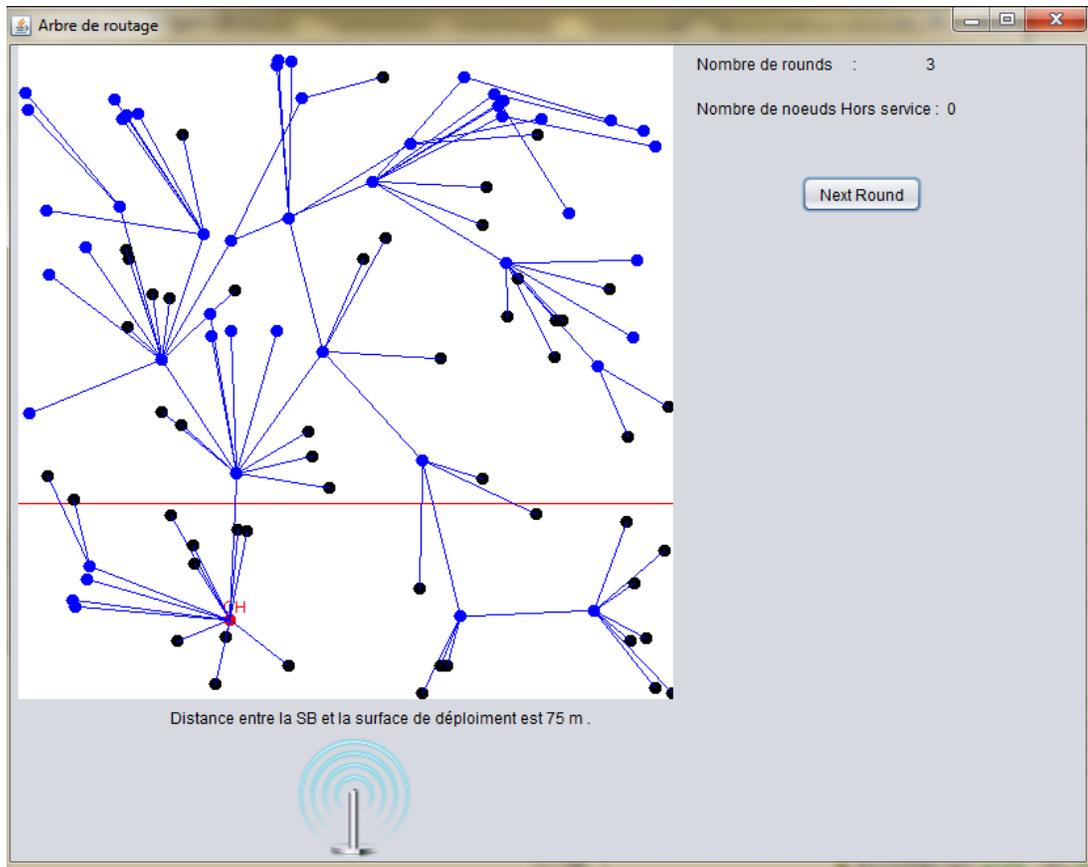


FIGURE 4.3 – Arbre de routage lors de l'exécution de notre protocole

- En choisissant "*Statistiques du protocole*" de la fenêtre principale et en validant, nous obtenons :

- Fenêtre "Statistiques du protocole" (voir figure 4.4) :

Dans cette fenêtre, nous avons un tableau composé de trois (3) colonnes, qui représentent (Round, le nombre de nœuds Hors Service (HS), l'énergie moyenne consommée).

En cliquant sur le bouton (Graphe) de cette fenêtre, notre protocole sera exécuté du début jusqu'à la fin. Ainsi, des résultats seront affichés, qui concernent le nombre de nœuds hors service depuis le début jusqu'à la fin de l'exécution, et l'énergie moyenne consommée par les nœuds pour chaque round.

Puis, les résultats affichés dans le tableau, seront envoyés à un fichier Excel afin de les schématiser en un graphe englobant les résultats affichés.

The screenshot shows a window titled "Statistiques du protocole" with a sub-header "Historique d'exécution". It contains a table with three columns: "Round", "Nbre Noeud HS", and "Énergie moyenne consom...". The table lists data for rounds 1 through 20. The "Nbre Noeud HS" column consistently shows 0. The "Énergie moyenne consom..." column shows values that fluctuate between approximately 419,000 and 446,000. Below the table is a button labeled "Graphe".

Round	Nbre Noeud HS	Énergie moyenne consom...
1	0	432280.0120204687
2	0	418616.3153408766
3	0	419557.1991522312
4	0	419319.02758204937
5	0	419045.452760458
6	0	432138.0949970484
7	0	1097981.4700695276
8	0	422416.6703219414
9	0	422744.8906955719
10	0	432899.66538238525
11	0	436016.4226202965
12	0	432065.9938031435
13	0	418877.4339455366
14	0	432455.4955393076
15	0	421916.98012828827
16	0	421573.5990062952
17	0	932670.9240201712
18	0	418671.1528788805
19	0	419980.93422818184
20	0	446476.8600207567

FIGURE 4.4 – Fenêtre "Statistiques du protocole"

- En choisissant "*Simulation comparative*" de la fenêtre principale et en validant, nous obtenons :
  - **Fenêtre "Simulation comparative"** (voir figure 4.5) :

Cette fenêtre représente l'exécution de quatre (4) protocoles de routage hiérarchiques (R2E-ANFP/ LEACH/ MH-LEACH/ CBCCP), où chaque tableau de la fenêtre représente l'exécution de chaque protocole. Chacun des quatre (4) tableaux est composé de deux (2) colonnes qui représentent (round et le nombre de nœuds actifs dans le réseau).

En cliquant sur le bouton (Graphe), les quatre (4) protocoles seront exécutés en même temps pour un même nombre de nœuds (100noeuds), l'exécution des protocoles sera effectué du début jusqu'à la fin (nombre de nœuds actifs = 0).

Chaque tableau contient les résultats d'exécution, où pour chaque round il lui est associé le nombre de nœuds actifs correspondant. Ce qui nous permettra d'établir une comparaison entre ces quatre protocoles en termes de durée de vie du réseau. A la fin, les résultats affichés sont envoyés dans un fichier "Excel", où un graphe comparatif sera schématisé.

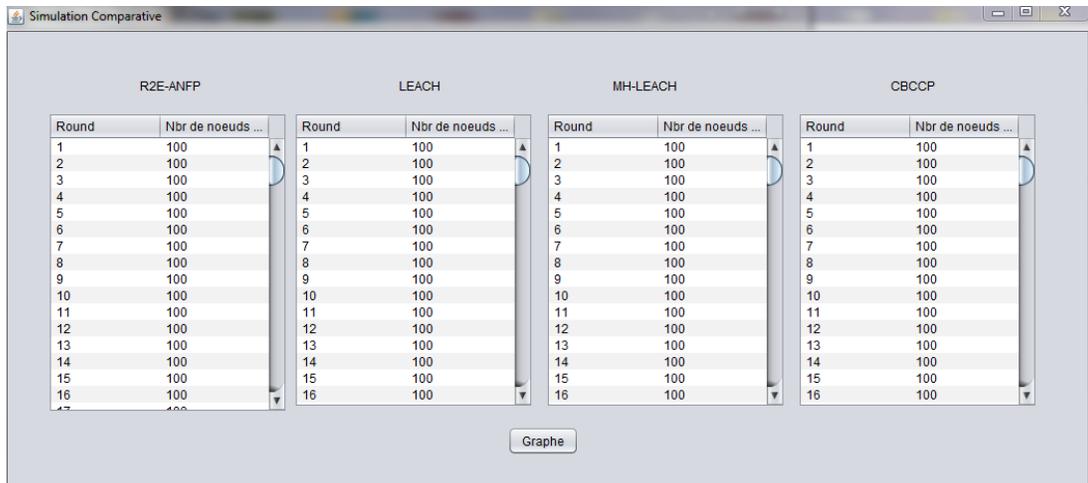


FIGURE 4.5 – Fenêtre "Simulation comparative"

- En choisissant "Scalabilité" de la fenêtre principale et en validant, nous obtenons :
  - Fenêtre "Scalabilité" (voir figure 4.6) :

Cette fenêtre est composée d'un tableau, qui permet d'afficher pour un nombre de noeuds voulu (en introduisant le nombre de noeuds et en cliquant sur (OK)), le taux de mortalité à savoir pour chaque pourcentage (1%, 25%, 50%, 75%, 100%) le round en question, ainsi que l'affichage de l'énergie moyenne consommée. Tout cela en exécutant notre protocole de routage du début jusqu'à la fin.

Ces résultats seront envoyés dans un fichier "Excel" afin de les schématiser sous forme d'un graphe en cliquant sur le bouton (Graphe) de cette fenêtre.

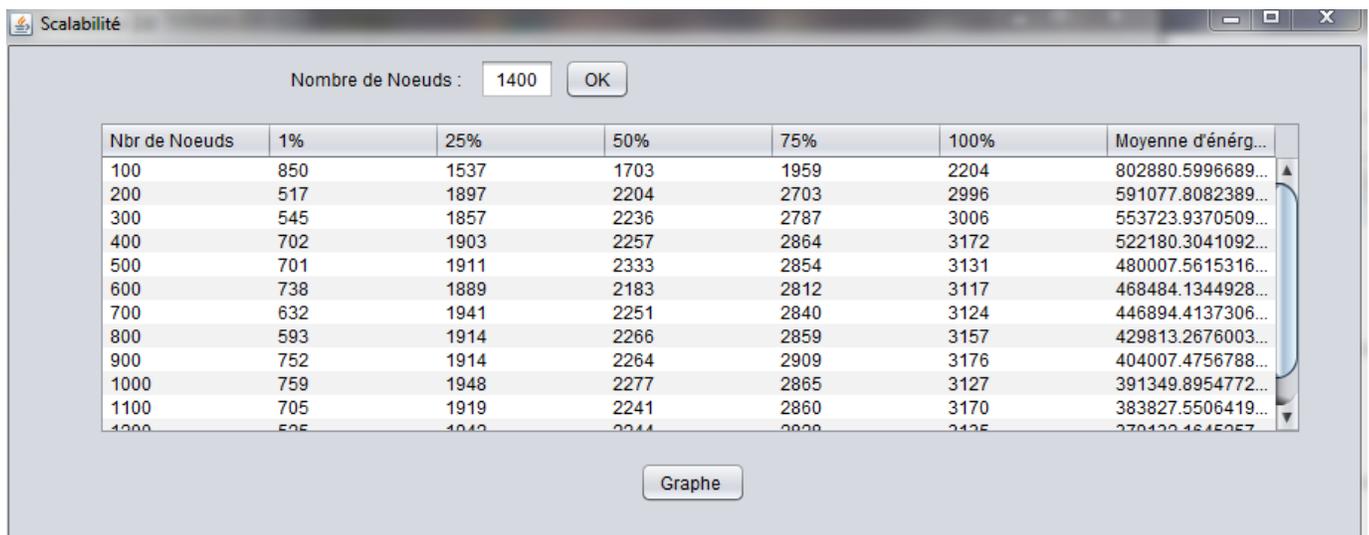


FIGURE 4.6 – Fenêtre "Scalabilité"

- En cliquant sur le bouton "Simuler" de la fenêtre principale et en validant, nous obtenons :
  - Fenêtre "Simulation \*100" (voir figure 4.7) :

Cette fenêtre contient cinq (5) tableaux, où quatre d'entre eux sont consacrés pour l'affichage des résultats des quatre protocoles de routage cités précédemment (R2E-ANFP/ LEACH/ MH-LEACH/ CBCCP). Ces derniers sont exécutés 100 fois du début jusqu'à la fin, pour chaque tableau sera affiché le nombre d'exécutions et le taux de mortalité en pourcentage (1%, 25%, 50%, 75%, 100%), où pour chaque pourcentage les rounds associés.

Le cinquième tableau représente la moyenne des 100 exécutions effectués auparavant, pour les mêmes pourcentages (1%, 25%, 50%, 75%, 100%) associés à chaque protocole.

Les résultats du cinquième tableau (moyenne des 100 exécutions) seront envoyés dans un fichier Excel afin d'obtenir un graphe englobant les résultats d'exécutions.

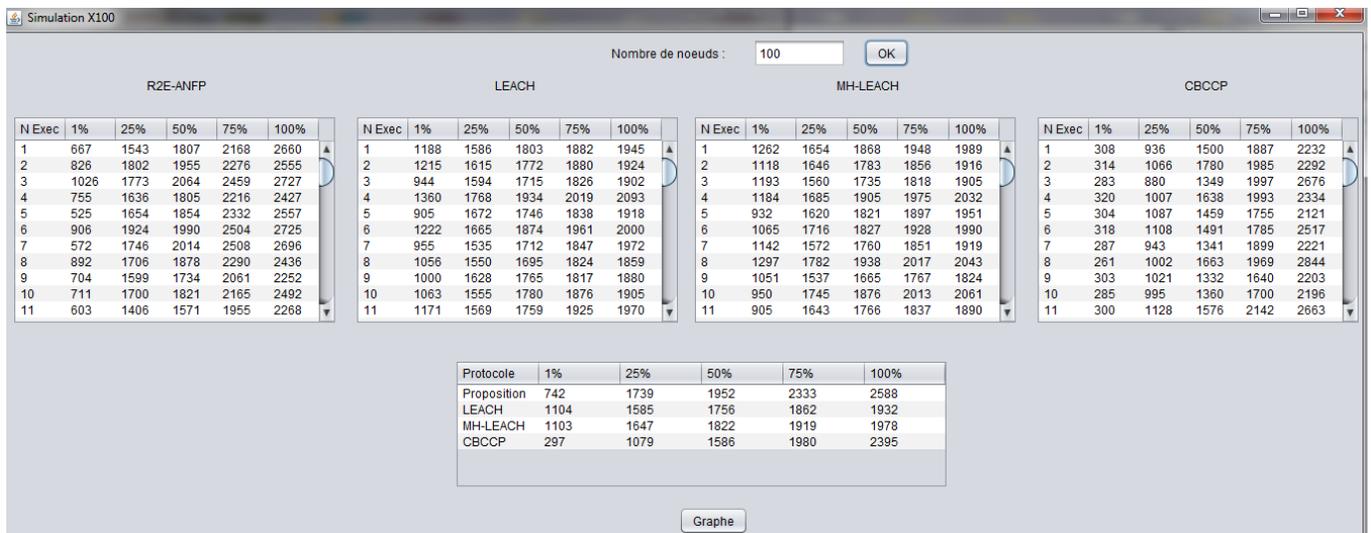


FIGURE 4.7 – Fenêtre "Simulation \*100"

## 4.4 Modèle de consommation énergétique

W.R.Heinzelman a proposé un modèle de radio de consommation d'énergie [10], qui est le premier modèle radio utilisé pour le calcul de la consommation d'énergie dans les RCSFs voir figure 4.8.

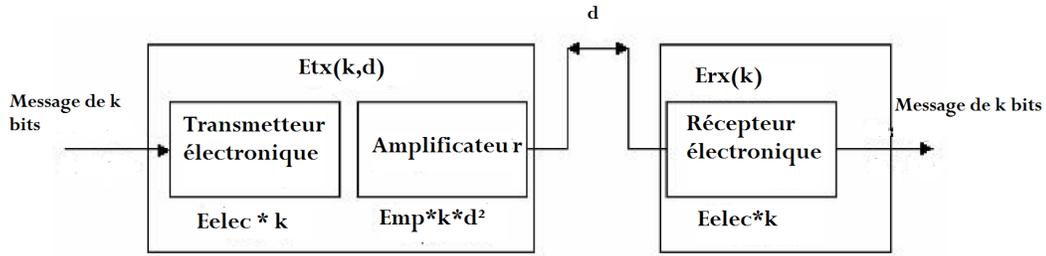


FIGURE 4.8 – Modèle de consommation d'énergie [10]

Lors de la communication de données, chaque nœud consomme de l'énergie selon l'opération qu'il effectue soit une réception ou une émission de messages de  $k$  bits à une distance  $d$ . Ainsi, les énergies nécessaires pour transmettre ( $E_{Tx}$ ) et pour recevoir ( $E_{Rx}$ ), sont données comme suit :

1. Afin d'émettre un message de  $k$  bits à une distance de  $d$  mètres, l'énergie consommée par l'émetteur est donnée par la formule suivante :

$$E_{Tx}(k, d) = \begin{cases} k * E_{elec} + k * \epsilon_{fs} * d^2 & \text{si } (d \leq d_0), \\ k * E_{elec} + k * \epsilon_{mp} * d^4 & \text{si } (d \geq d_0), \end{cases}$$

Où :

- $d$ , représente la distance entre l'émetteur et le récepteur ;
- $k$ , représente la taille des messages en bits ;
- $E_{elec}$ , représente l'énergie consommée pour émettre ou recevoir un message de  $1bit$  ;
- $\epsilon_{fs}$  et  $\epsilon_{mp}$ , représentent les énergies d'amplification utilisée pour transmettre à de courtes distances ( $\leq d_0$ ) et à de longues distances ( $\geq d_0$ ) respectivement ;
- $d_0$ , représente la distance seuil auquelle un message de  $k$  bits peut être envoyé, elle est calculée en utilisant la formule suivante [9] :

$$d_0 = \sqrt{\epsilon_{fs} / \epsilon_{mp}} \quad (4.1)$$

2. Afin de recevoir un message de  $k$  bits, l'énergie consommée par le récepteur est donnée par la

formule suivante :

$$E_{Rx}(k, d) = k * E_{elec}$$

#### 4.4.1 Energie consommée durant chaque phase de notre protocole

Vu que les tâches et opérations effectuées par chaque nœud diffèrent d'une phase à une autre, par conséquent l'énergie consommée va être aussi différente selon la phase dans laquelle le nœud du réseau se retrouve.

Nous allons expliquer ci-dessous pour chaque phase qui constitue notre protocole, comment l'énergie sera consommée par les nœuds en utilisant le modèle de consommation énergétique :

- **Phase d'élection du CH** : Durant cette phase le nœud se voit effectuer une réception d'un paquet de contrôle en provenance de la station de base, au quel il répondra par un paquet contenant son identifiant (ID) et son niveau d'énergie.

Après que la station de base ait élu le CH, ce dernier reçoit un paquet contenant son ID que la SB aurait transmis, dans le but d'informer le nœud concerné qu'il est élu CH.

Suite à ça les nœuds concernés consomment une énergie totale  $E_{phase1}$  à la fin de cette première phase :

$$E_{phase1} = 2 * E_{Rx}(k) + E_{Tx}(k, d).$$

- **Phase de formation du cluster** : Le nœud CH diffuse un message de contrôle dans un rayon  $r'$  définie, afin de créer son cluster et élire un ou plusieurs nœuds Sous CHs selon les conditions mentionnées dans le chapitre précédent.

Au cours de cette phase le nœud CH effectue les opérations suivantes :

- Emission d'un paquet de contrôle à une distance  $r'$  ;
- Réception de  $n$  réponses (  $n$  correspond au nombre de nœuds présents à un rayon  $r'$  ) ;
- Emission d'un paquet de contrôle contenant les identifiants (IDs) des nœuds élus Sous-CH.

Au cours de cette phase les nœuds présents dans le rayon de diffusion  $r'$  effectuent les opérations suivantes :

- Réception d'un paquet de contrôle depuis le nœud CH ;
- Emission d'un message d'appartenance au cluster vers le CH.

Nous rappelons qu'un Sous-CH effectue les mêmes opérations après son élection.

Suite a ça l'énergie total consommée  $E_{phase2}$  dans cette phase est :

$$E_{phase2} = 2 * E_{Tx}(k, d) + n * E_{Rx}(k).$$

- **Phase de communication de données** : La consommation d'énergie durant cette phase, diffère selon le type du nœud et les opérations à effectuer pour chaque type de nœud, comme suit :

(a) Nœud normal :

Il transmet les données collectées au nœud CH / Sous CH au quel il est lié dans le cluster formé. Suivant cette formule :

$$E_{phase3Noeud} = E_{Tx}(k, d).$$

(b) Nœud CH / Sous CH :

Suivant ce type de nœuds, il effectue deux opérations durant cette phase :

- Il reçoit les données des nœuds membre de son cluster  $n * E_{Rx}(k)$  ;
- Il transmet les données agrégées au supérieur hiérarchique au quel il est lié  $E_{Tx}(k, d)$ .

Donc, au total l'énergie consommée durant cette phase pour les nœuds CH/ Sous CH est comme suit :

$$E_{phase3CH} = n * E_{Rx}(k) + E_{Tx}(k, d),$$

Où :

$n$  : représente le nombre de nœuds membres du cluster.

A la fin de cette phase l'énergie totale consommée par les nœuds qui composent le cluster, est  $E_{phase3}$  donnée par la formule suivante :

$$E_{phase3} = E_{phase3Noeud} + E_{phase3CH}.$$

#### 4.4.2 Paramètres utilisés pour la simulation

Le tableau 4.1 suivant, résume les paramètres que nous avons utilisé pour la simulation de notre protocole :

Surface du réseau	$(100 * 100)m^2$
Position de la SB	(50, 175)
Energie initiale des noeuds	1J
$E_{elec}$	50nJ/bit
$\epsilon_{fs}$	100pJ/bit/m <sup>2</sup>
$\epsilon_{mp}$	0.0013pJ/bit/m <sup>4</sup>
$d_0$	$\sqrt{\epsilon_{fs}/\epsilon_{mp}} = 87m$
Taille des paquets de données	2000bits
Taille des messages de contrôle	200bits
Rayon de diffusion de la SB	30% de la largeur de la surface du réseau= 30m
Rayon de diffusion de (CH/Sous CH)	25% de la largeur de la surface du réseau= 25m

TABLE 4.1 – Paramètres de simulations

### 4.5 Résultats de simulations

Dans cette section, nous allons détailler chaque résultat en fonction des simulations effectuées sur le protocole R2E-ANFP, ainsi que trois (3) autres protocoles de routage. Ces résultats représentent la moyenne de 100 exécutions, en utilisant les paramètres du tableau 4.1, nos résultats englobent :

1. Energie moyenne consommée ;
2. Durée de vie du réseau en fonction du nombre de nœuds actifs ;
3. Scalabilité de notre protocole ;
4. Taux de mortalité.

#### 4.5.1 Simulation comparative

Nous avons effectué une simulation du protocole R2E-ANFP et nous l'avons comparé à trois (3) protocoles LEACH [12], MH-LEACH [13] et CBCCP [21].

Les résultats de simulations (voir le graphe de la figure (4.9)), nous montre que dans :

1. LEACH : Le premier nœud meurt au round 1160, et le dernier au round 1890, alors la durée de mortalité de ce protocole est égale à 730 rounds ;
2. MH-LEACH : Le premier nœud meurt au round 1400, et le dernier au round 2069, alors la durée de mortalité de ce protocole est égale à 669 rounds ;
3. CBCCP : Le premier nœud meurt au round 350, et le dernier au round 2070, alors la durée de mortalité de ce protocole est égale à 1720 rounds ;
4. R2E-ANFP : Le premier nœud meurt au round 850, et le dernier au round 2610, alors la durée de mortalité de ce protocole est égale à 1760 rounds.

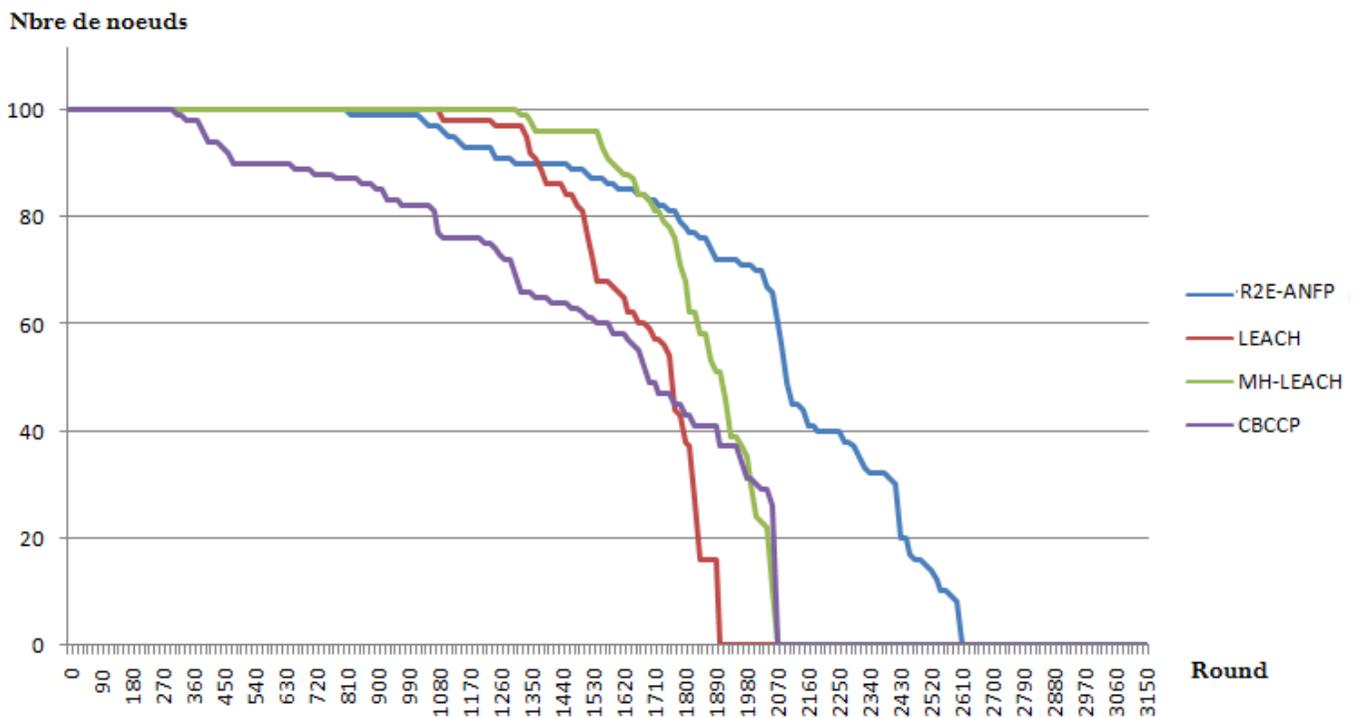


FIGURE 4.9 – Graphe des résultats de simulation représentant "nombre de nœuds actifs dans le réseau en fonction des rounds"

Durant cette simulation, nous avons constaté que le protocole R2E-ANFP a une durée de mortalité élevée par rapport à LEACH, MH-LEACH et CBCCP.

#### 4.5.2 Scalabilité

Afin de vérifier la scalabilité du protocole R2E-ANFP, nous avons effectué une simulation en fonction du nombre de nœuds déployés, et nous avons affiché dans le tableau de la fenêtre "Scalabilité"

(figure 4.6) les pourcentages de mortalité ainsi que l'énergie moyenne consommée.

Les résultats obtenus sont comparés aux performances de LEACH, MH-LEACH, et CBCCP.

1. **Pourcentage de mortalité de nœuds (1%) :**

Nous avons englobé les rounds dans lesquels le premier nœud meurt 4.10, et le tableau obtenu est traduit sous forme de graphe (figure 4.11).

• **Discussion :**

- Le protocole "CBCCP" : Nous remarquons que sa courbe évolue en dessous de notre protocole, ce qui permet de dire que les nœuds commencent à mourir dans ce protocole bien avant le notre. Cependant, la courbe reste constante malgré que le nombre de nœuds augmente dans le réseau ;
- Les protocoles "LEACH" et "MH-LEACH" : On constate que les deux courbes respectifs des deux protocoles sont presque similaires et évoluent de la même façon, elles se situent au dessus de notre protocole et du protocole CBCCP, donc le pourcentage de nœuds hors service est atteint bien après notre protocole ;
- Le protocole "R2E-ANFP" : La courbe a évolué d'une manière presque constante avec de petites variations en termes de rounds pour 1% de mortalité des nœuds du réseau, et cela malgré que le nombre de nœuds augmente, contrairement aux protocoles précédents LEACH et MH-LEACH où il ya une augmentation jusqu'à atteindre un seuil après diminue d'une façon progressive.

<i>Nombre de nœuds</i>	<i>R2E-ANFP</i>	<i>LEACH</i>	<i>MH-LEACH</i>	<i>CBCCP</i>
100	834	1113	885	322
200	711	1756	1873	325
300	622	1782	1944	279
400	725	1642	1900	278
500	556	1587	1784	274
600	764	1481	1621	249
700	680	1406	1538	204
800	657	1308	1445	212
900	690	1195	1346	190
1000	665	1156	1281	201
1100	695	1110	1203	146
1200	559	1051	1146	169
1300	607	998	1080	182
1400	586	953	1034	155
1500	477	901	984	141

FIGURE 4.10 – Tableau des résultats pour le premier nœud qui meurt

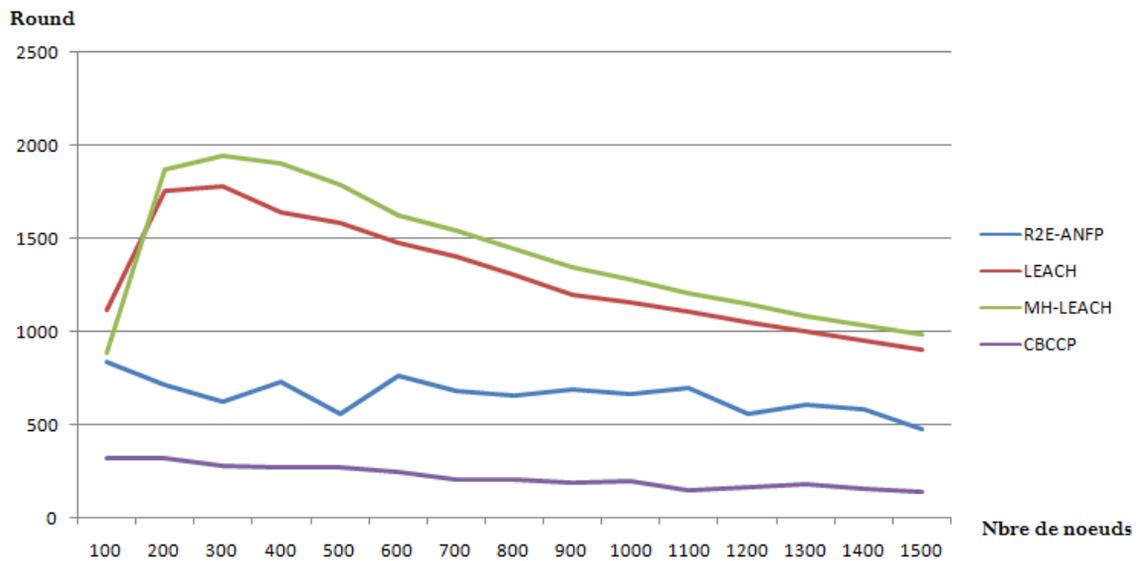


FIGURE 4.11 – Graphe du premier nœud qui meurt dans le réseau

D'après les résultats de simulations, dans R2E-ANFP le premier nœud meurt avant les protocoles LEACH et MH-LEACH, et cela est dû aux messages de contrôles générés pour l'exécution

de la phase d'élection du CH et des nœuds Sous CHs.

Au sein de R2E-ANFP, le nombre de clusters générés est approximativement le même entre un round et un autres (et cela s'applique même pour un nombre de nœuds qui augmente), ce qui revient à dire que le nombre de messages de contrôles générées durant chaque round est approximativement le même, ce qui engendre une consommation d'énergie régulière. Contrairement, à LEACH et MH-LEACH ou la procédure d'élection des CHs est probabiliste (le nombre de CHs élus diffère d'un round à un autre) ce qui à engendré une variation dans le nombre de messages de contrôles générées et une variation dans la consommation d'énergie.

## 2. Pourcentage de mortalité de nœuds (25%) :

Pour un pourcentage de mortalité égale à (25%), nous avons les résultats du tableau 4.12 représentés sous forme de graphe (figure 4.13).

### • Discussion :

- Le protocole "CBCCP" : La courbe de ce protocole évolue de manière constante pour un pourcentage de mortalité égal à 25%, mais elle est située en dessous de R2E-ANFP, LEACH et MH-LEACH ;
- Les protocoles "LEACH" et "MH-LEACH" : Comme le montre le graphe leurs courbes augmentent jusqu'à un seuil puis diminuent à partir d'un nombre de nœuds supérieur à 200 ;
- Le protocole "R2E-ANFP" : La courbe monte progressivement avec quelques variations, puis se stabilise.

Nombre de nœuds	R2E- ANFP	LEACH	MH-LEACH	CBCCP
100	1755	1646	1485	977
200	1892	2034	2262	800
300	1859	1982	2188	835
400	1906	1858	2047	809
500	1945	1728	1911	826
600	1983	1626	1771	824
700	1919	1514	1645	826
800	1926	1425	1536	851
900	1919	1336	1432	885
1000	1927	1259	1346	912
1100	1911	1189	1268	887
1200	1950	1131	1197	874
1300	1925	1074	1135	870
1400	1923	1025	1079	875
1500	1961	976	1026	860

FIGURE 4.12 – Tableau des résultats pour un pourcentage 25% de mortalité des nœuds

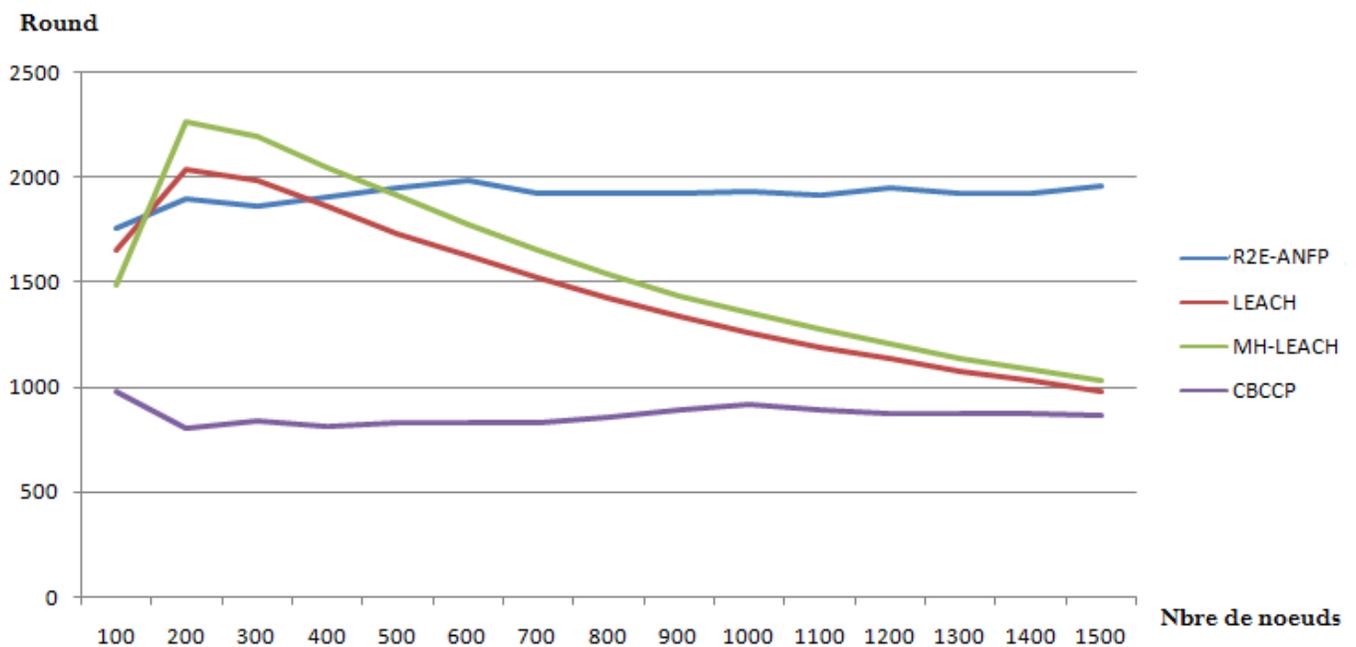


FIGURE 4.13 – Graphe "Pourcentage de mortalité des nœuds (25%)"

Dans R2E-ANFP, le nombre de clusters formés ne diffèrent pas vraiment d'un nombre de

nœuds à un autre, donc même si que le nombre de nœuds augmente il n'y aura pas une grande augmentation du nombre de nœuds dans les clusters, ainsi la transmission de données s'effectue à une distance ( $< 25m$ ). Alors que dans les protocoles LEACH et MH-LEACH l'élection des CHs est probabiliste ce qui engendre dans la majorité des cas des CHs éloignés par rapport au reste du réseau, la transmission de donnée des nœuds vers le CH des clusters s'effectuera malgré cet éloignement, ce qui engendre une grande consommation d'énergie d'où la progression dans la descente des courbes.

Dans le protocole CBCCP, les zones formées sont fixes (10 zones) et ont un nombre égale de nœuds, donc lorsque le nombre de nœuds dans le réseau est doublé, triplé, etc, le nombre de nœuds dans les zones l'est aussi. Sachant qu'il ya un seul CH pour chaque zone, alors les réceptions des données effectuées par le CH augmente avec le nombre de nœuds, il en résulte une grande consommation d'énergie.

### 3. Pourcentage de mortalité de nœuds (50%) :

Pour un pourcentage de mortalité égale à (50%), nous avons les résultats du tableau 4.14 représentés sous forme de graphe (figure 4.15).

#### • Discussion :

- Le protocole "CBCCP" : La courbe varie de manière constante, à un certain nombre de nœuds il dépasse les deux protocoles de routage LEACH et MH-LEACH ;
- Les deux protocoles "LEACH" et "MH-LEACH" : On constate la même évolution des deux courbes que celles du pourcentage de mortalité égale à (25%) ;
- Le protocole "R2E-ANFP" : La courbe monte, puis se stabilise.

Nombre de nœuds	R2E- ANFP	LEACH	MH-LEACH	CBCCP
100	1868	1781	1697	1378
200	2075	2183	2330	1302
300	2194	2151	2232	1252
400	2272	1997	2082	1168
500	2248	1860	1947	1192
600	2248	1742	1805	1196
700	2211	1611	1678	1250
800	2243	1520	1561	1297
900	2281	1421	1459	1321
1000	2287	1330	1371	1360
1100	2262	1254	1291	1381
1200	2253	1186	1215	1302
1300	2251	1129	1152	1348
1400	2316	1072	1094	1363
1500	2267	1018	1040	1404

FIGURE 4.14 – Tableau des résultats pour pourcentage 50% de la mortalité des nœuds

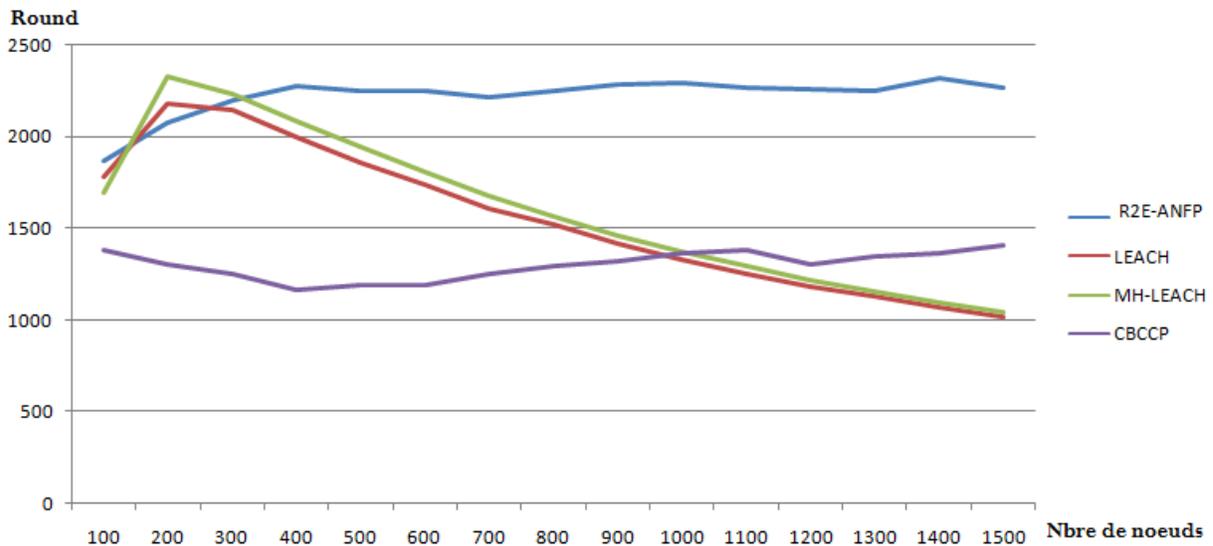


FIGURE 4.15 – Graphe "Pourcentage de mortalité des nœuds (50%)"

Dans R2E-ANFP, nous avons tout les nœuds de la zone(1) qui sont morts, ce qui pousse la procédure d'élection du CH a augmenter le rayon de la zone(1) pour atteindre de nouveaux nœuds. Au final, ce qui diffère dans l'exécution du protocole dans ce pourcentage par rapport au précédent (25%) est que la distance entre le CH élu et la station de base a augmenté.

#### 4. Pourcentage de mortalité de nœuds (75%) :

Pour un pourcentage de mortalité égale à (75%), nous avons les résultats du tableau 4.16 représentés sous forme de graphe (figure 4.17).

<i>Nombre de nœuds</i>	<i>R2E- ANFP</i>	<i>LEACH</i>	<i>MH-LEACH</i>	<i>CBCCP</i>
<b>100</b>	2084	1867	1773	1900
<b>200</b>	2728	2305	2370	1602
<b>300</b>	2707	2261	2265	1563
<b>400</b>	2799	2094	2131	1562
<b>500</b>	2834	1950	1988	1520
<b>600</b>	2892	1828	1836	1535
<b>700</b>	2894	1692	1712	1592
<b>800</b>	2887	1592	1589	1602
<b>900</b>	2854	1485	1492	1688
<b>1000</b>	2833	1392	1398	1690
<b>1100</b>	2916	1308	1315	1709
<b>1200</b>	2861	1239	1241	1614
<b>1300</b>	2891	1173	1175	1669
<b>1400</b>	2890	1115	1114	1708
<b>1500</b>	2868	1060	1060	1734

FIGURE 4.16 – Tableau des résultats pour un pourcentage de 75% de la mortalité des nœuds

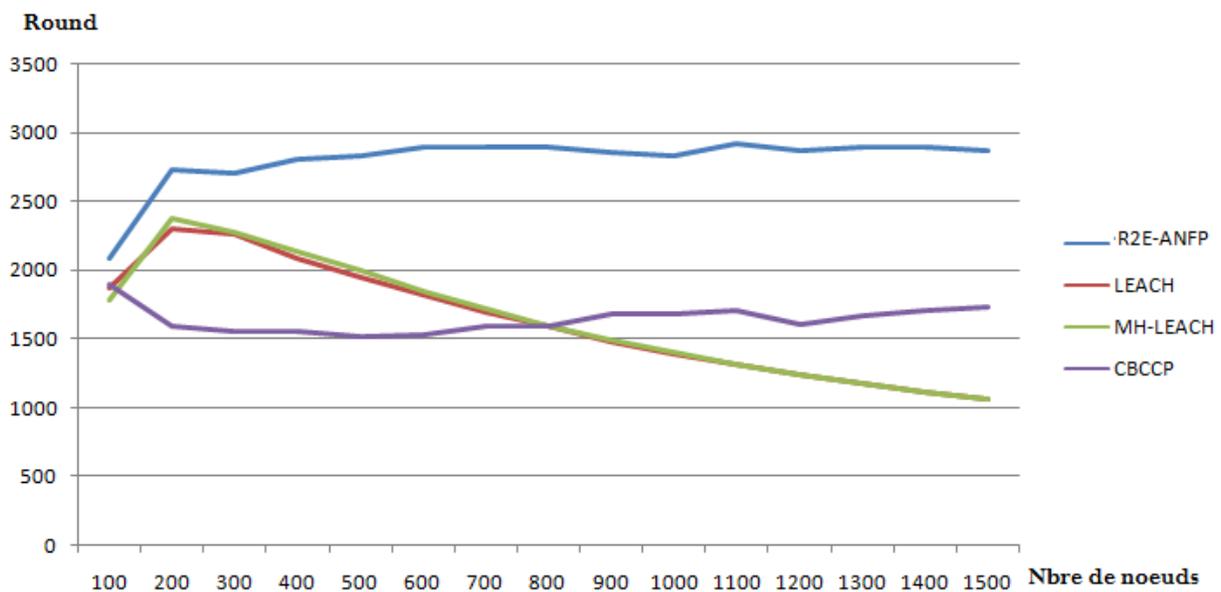


FIGURE 4.17 – Graphe "Pourcentage de mortalité des nœuds (75%)"

- **Discussion** : R2E-ANFP est le plus performant par rapport aux autres protocoles à ce stade de la simulation, comme le graphe de la figure 4.17 le reflète.

Et cela, est du à la zone(1) qui avance au fur et a mesure que les nœuds meurt dans le réseau pour élire le CH, et aussi à la procédure de gestion des nœuds isolés qui permet une liaison entre les nœuds du réseau malgré les écarts de distances qui peuvent survenir.

#### 5. Pourcentage de mortalité de nœuds (100%) :

Nous avons englobé les rounds dans lesquels le dernier nœud meurt 4.18, et le tableau obtenu est traduit sous forme de graphe (figure 4.19).

<i>Nombre de nœuds</i>	<i>R2E- ANFP</i>	<i>LEACH</i>	<i>MH-LEACH</i>	<i>CBCCP</i>
<b>100</b>	2451	1920	1832	2285
<b>200</b>	3039	2398	2413	2061
<b>300</b>	2972	2333	2313	1857
<b>400</b>	3057	2223	2196	1859
<b>500</b>	3129	2035	2056	1887
<b>600</b>	3161	1971	1939	1813
<b>700</b>	3148	1791	1797	1897
<b>800</b>	3201	1691	1678	1964
<b>900</b>	3120	1598	1566	2058
<b>1000</b>	3127	1512	1477	2092
<b>1100</b>	3160	1436	1418	2082
<b>1200</b>	3194	1349	1309	1991
<b>1300</b>	3192	1285	1267	2008
<b>1400</b>	3201	1221	1178	2035
<b>1500</b>	3163	1175	1133	2032

FIGURE 4.18 – Tableau des résultats du dernier nœud qui meurt

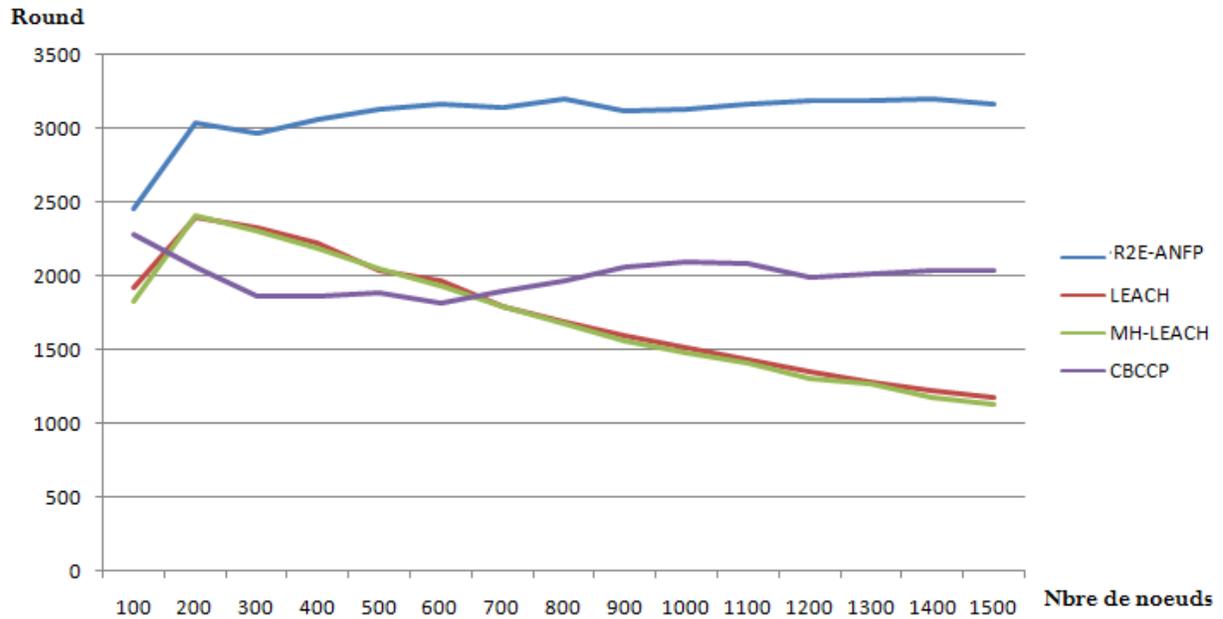


FIGURE 4.19 – Graphe "Pourcentage de mortalité de nœuds (100%)"

- Discussion :** La courbe de notre protocole est supérieure à celle des autres protocoles, ce qui prouve que notre protocole est plus performant que les autres, malgré l'augmentation du nombre de nœuds dans le réseau. Dans ce cas, la station de base aurait balayé tout le réseau et le dernier nœud à mourir est le nœud le plus éloigné de la station de base.

Afin d'appuyer les résultats obtenus dans les graphes précédents nous avons généré un graphe comparatif des consommations d'énergie moyenne des quatre protocoles en fonction du nombre de nœuds qui augmente. Comme le montre le graphe de la figure(4.21).

Nombre de nœuds	R2E- ANFP	LEACH	MH-LEACH	CBCCP
100	407996,736	520833,333	545851,528	437636,761
200	329055,61	417014,178	414421,881	485201,359
300	336473,755	428632,662	432338,954	538502,962
400	327118,09	449842,555	455373,406	537923,615
500	319590,924	491400,491	486381,323	529941,706
600	316355,584	507356,672	515729,758	551571,98
700	317662,008	558347,292	556483,027	527148,129
800	312402,374	591366,056	595947,557	509164,969
900	320512,821	625782,228	638569,604	485908,649
1000	319795,331	661375,661	677048,07	478011,472
1100	316455,696	696378,83	705218,618	480307,397
1200	313087,038	741289,844	763941,94	502260,171
1300	313283,208	778210,117	789265,983	498007,968
1400	312402,374	819000,819	848896,435	491400,491
1500	316155,549	851063,83	882612,533	492125,984

FIGURE 4.20 – Tableau de la consommation moyenne d'énergie par round

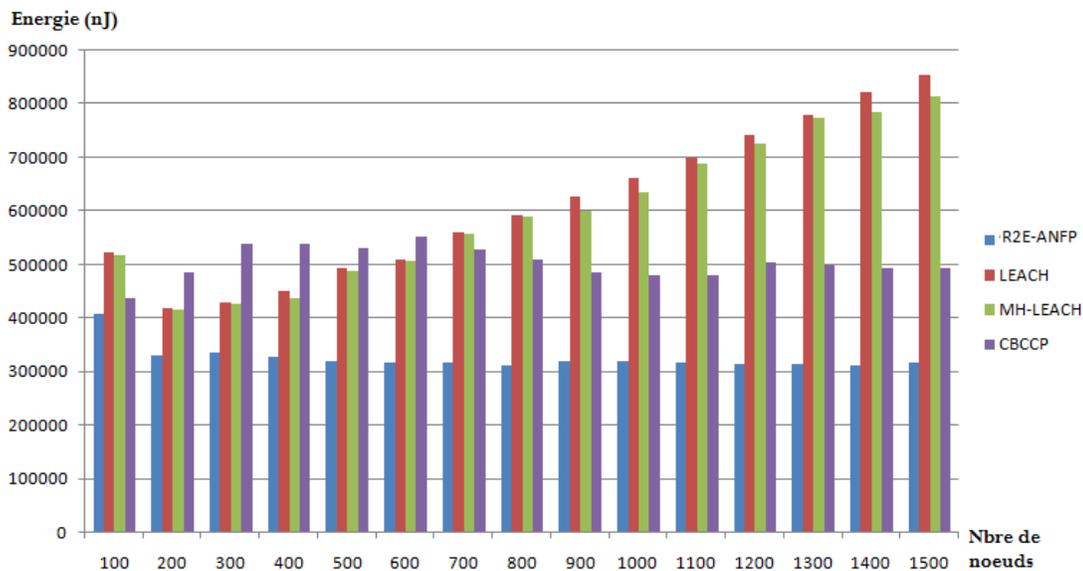


FIGURE 4.21 – Graphe de consommation moyenne d'énergie en fonction du nombre de nœuds dans le réseau par round

- Discussion :

(a) Protocole "LEACH" :

Comme le graphe le montre, la consommation d'énergie du protocole LEACH augmente au

fur et a mesure que le nombre de nœuds dans le réseau augmente, cela est du à l'élection probabiliste des nœuds CHs qui engendre une variation dans le nombre de CHs dans le réseau d'un round à un autre, et dans la majorité des cas les CHs élus se trouvent à des distances éloignées des nœuds membres de clusters. Cela à pour conséquences, une augmentation considérable des messages de données reçus par les CHs et une émission à de longues distances de ses messages pour les nœuds membres d'où une grande consommation d'énergie de part et d'autres, et cela justifie les résultats obtenus dans le graphe (4.21) ;

(b) Protocole "MH-LEACH" :

Tout comme LEACH, la consommation d'énergie des nœuds augmente avec l'augmentation des nœuds du réseau. Cependant, elle est légèrement plus faible que celle de ce dernier et cela est due à l'utilisation d'un routage multi-sauts entre les nœuds CHs, ce qui réduit la distance d'émission des CHs (un CH émet le message agrégé à un autre CH plus proche que lui de la station de base) ;

(c) Protocole "CBCCP" :

La consommation d'énergie de ce protocole comme le montre le graphe est plus faible que les deux protocoles précédents (LEACH et MH-LEACH), et dans la majorité des cas constante, cela est du à la division du réseau en 10 zones et à la gestion des émissions utilisée dans ce protocole (un CH envoie à un CCO de la zone supérieur, et les CCOs communiquent avec les CCOs des zones supérieurs) ce qui diminue considérablement la consommation d'énergie. Mais cependant, la consommation d'énergie dans ce protocole reste considérablement élevée car avec l'augmentation des nœuds dans le réseau, le nombre des nœuds dans les zones augmente aussi et il ya qu'un seul nœud CH dans chaque zone pour gérer le flux de données qui s'y trouve ;

(d) Le protocole "R2E-ANFP" :

Comme le graphe le montre, la consommation d'énergie reste presque constante malgré que le nombre de nœuds augmente dans le réseau et elle est largement inférieure aux autres protocoles.

Car premièrement, lorsque le nombre de nœuds dans le réseau augmente, le nombre de nœuds Sous CHs élus est augmenté, ce qui génère une légère hausse dans le nombre de

clusters formés (Sans autant avoir une grande différence dans le nombre de clusters formés entre un réseau et un autre "100, 200, 300, 400, etc."). Cela revient à dire que le nombre des nœuds membres dans les clusters diminue, ce qui permet de diminuer le nombre de données reçues par les Sous CHs, en conséquence la charge de la consommation d'énergie est divisée au sein des clusters.

Deuxièmement, le rayon de diffusion fixé au départ permet aux nœuds une émission à une petite distance ( $\leq 25m$ ) ce qui est un gain dans la consommation d'énergie dues aux émissions.

Après étude des graphes, nous constatons que même si que le nombre de nœuds augmente dans le réseau la consommation d'énergie est équilibrée d'un réseau à un autre, ce qui prolonge la durée de vie du réseau pour l'exécution de notre protocole et assure une constance dans les résultats malgré la variation du nombre de nœuds. Au final, nous sommes dans la mesure de dire que notre protocole assure la scalabilité et cela d'après les résultats fournis dans la simulation effectuée.

## 4.6 Conclusion

Au cours de ce chapitre nous avons présenté les outils et langages utilisés pour le développement du simulateur ainsi que les différentes fenêtres de l'environnement de simulation, nous avons également définie les paramètres de simulation, ensuite simuler notre protocole (R2E-ANFP) en se basent sur ces derniers, et recenser les différents résultats obtenues (Durée de vie du réseau, Scalabilité, Consommation d'énergie...) dans des tableaux que nous avons traduit sous forme de graphes descriptifs.

D'après les résultats de simulations fournis précédemment, nous avons constaté que notre protocole est plus performant que les trois (3) autres protocoles (LEACH/ MH-LEACH / CBCCP) en termes de consommation moyenne d'énergie, car les nœuds du réseau consomment moins d'énergie vu que les nœuds meurent plus longtemps, ce qui prouve que notre protocole assure une longue durée de vie du réseau.

Aussi, nous pouvons dire que la consommation énergétique est équilibrée lors de l'exécution de notre protocole. Enfin, notre protocole est scalable, en d'autre termes, même si que le nombre de nœuds augmente la consommation d'énergie reste constante, d'où une constance dans les résultats.

# Conclusion Générale

L'émergence des réseaux de capteurs sans fil ne cessent d'accroître, puisqu'ils représentent un axe riche et innovant pour les concepteurs de ce type de réseau. Vu leurs nécessités et leurs utilisations dans diverses domaines, nous les retrouvons notamment dans le domaine militaire, médical, environnementale, etc.

A l'heure actuelle les RCSFs sont littéralement omniprésents, grâce à leurs utilisations dans le but d'obtenir des informations sur l'environnement qui nous entoure. Les RCSFs sont composés d'un grand nombre de nœuds capteurs de petites tailles, autonomes, avec des ressources de calcul et d'énergie limitées, ils collaborent entre eux afin de collecter les données et les envoyer à l'utilisateur final.

Cependant, les RCSFs sont affectés par la contrainte d'énergie qui influence négativement sur la durée de vie du réseau. Toutefois, l'objectif principal des concepteurs de réseaux est de minimiser la consommation énergétique au travers de la conception et le développement de protocoles de routage visant à assurer le prolongement de la durée de vie de tout le réseau. De ce fait, le type de routage qui a fait ses preuves, et qui s'est avéré une solution pour la problématique d'énergie est le routage hiérarchique, où il permet d'organiser le réseau en clusters et chaque cluster est gouverné par un chef nommé nœud Cluster Head (CH). Cette architecture permet la communication de données collectées jusqu'à la station de base avec un minimum de perte d'énergie, tout en maximisant la durée de vie du réseau.

Dans ce mémoire, nous avons commencé par présenter les généralités qui englobent le domaine des RCSFs, puis nous avons présenté l'état de l'art des protocoles de routage présentés dans la litté-

ture, selon les approches et les techniques de routage utilisées afin de les différencier, en établissant pour chaque protocole présenté une critique en présentant ses différents avantages et inconvénients. Ensuite, nous avons présenté notre contribution dans les RCSFs, qui est la proposition d'un protocole de routage hiérarchique à basse consommation énergétique.

Notre protocole de routage permet l'utilisation de trois (3) types de nœuds (nœud CH, nœud Sous CH, nœud normaux). Il est composé principalement de trois (3) phases (phase d'élection du CH, phase de formation de clusters et phase de communication de données). En plus, de l'agrégation de données qui est effectuée par les nœuds CH et Sous CHs, notre protocole permet d'utiliser le routage multi-sauts jusqu'à ce que les données arrivent à la station de base, dans le but de minimiser la consommation des nœuds qui composent le réseau.

Enfin, nous sommes passé à l'évaluation des performances de notre protocole en utilisant le simulateur que nous avons créé, dans lequel nous avons constaté que le notre proposition est plus performante que les protocoles LEACH, MH-LEACH et CBCCP en termes de consommation d'énergie et la durée de vie du réseau selon les résultats de simulations fournis.

Comme perspectives de notre travail, nous voudrions appliquer le protocole R2E-ANFP dans un environnement mobile, et étudier la possibilité de la mobilité des nœuds capteurs et la station de base, aussi nous voudrions l'exécuter sur des simulateurs tels que Jsn, NS2, OPNET, etc. Nous pouvons aussi le tester dans un environnement réel des réseaux de capteurs sans fil.

# Bibliographie

- [1] A.Ahmed Abbasi and M. Younis. A survey on clustering algorithms for wireless sensor networks. *Computer Communications* 30 , pp. 2826-2841, (2007).
- [2] K. Akkaya and M. Younis. A survey on routing protocols for wireless sensor networks. *Ad Hoc Networks* 3 , pp. 325-349, (2005).
- [3] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless sensor networks : a survey. *Computer Networks* 38, pp. 393-422, (2002).
- [4] M. Alishah, G. Abbas, A.B. Dogar, and Z. Halim. Scaling hierarchical clustering and energy aware routing for sensor networks. *Springer, Complex Adapt Syst Model*, pp. 1-23, (2015).
- [5] J. Doudoux. [http ://www.jmdoudoux.fr/java/dej/chap-outils.htm](http://www.jmdoudoux.fr/java/dej/chap-outils.htm), Consulté le 28 mai 2016.
- [6] S. Hara, T. Ishimoto, M. Kitano, and T. Tsujioka. A common coordinates/heading direction generation method for a robot swarm with only rssi-based ranging. *EURASIP Journal on Advances in Signal Processing*, pp. 1-11, (2009).
- [7] C. Herby. [https ://openclassrooms.com/courses/apprenez-a-programmer-en-java](https://openclassrooms.com/courses/apprenez-a-programmer-en-java), Consulté le 2 juin 2016.
- [8] C. Hortsman and G. Cornell. *Au coeur de JAVA2 notions fondamentales, volume (1)*. Campus Press, (2004).
- [9] V. Katiyar and al. Improvement in leach protocol for large-scale wireless sensor networks. *IEEE, PROCEEDINGS OF ICETEECT*, pp. 1070-1075, (2011).
- [10] N. Kumar and J. Kaur. Improved leach protocol for wireless sensor networks. *IEEE* , pp. 1-5, (2011).

- 
- [11] A. Makhoul. *Réseaux de capteurs : localisation, couverture et fusion de données*. PhD thesis, Thèse de doctorat en Informatique, Université de Franche-Compte, (2008).
- [12] Z. Manap, B. Mohd Ali, C. Kyun Ng, N. K. Noordin, and A. Sali. A review on hierarchical routing protocols for wireless sensor networks. *Springers Wireless Pers Commun*, pp. 1077-1104, (2013).
- [13] J.H.B. Neto and A. da Silva Rego. Mh-leach : A distributed algorithm for multi-hop communication in wireless sensor networks. *The Thirteenth International Conference on Networks*, pp. 55-61, (2014).
- [14] L.I. Niar. Analyse graphique pour la surveillance dans un réseau de capteurs sans fil (rcsf) simulateur : Omnet++. Master's thesis, Mémoire de magister en Informatique, Université d'Oran, (2012).
- [15] Nivedita and al. A survey on power efficient hierarchical routing protocols in wsns. *International Journal of Advanced Research in Computer Science and Software Engineering* 3 (11) , pp. 637-646, (2013).
- [16] H. Noureddine. Conception et implémentation d'un système de surveillance des feux de forêts basé sur les réseaux de capteurs sans fil. Master's thesis, Mémoire de magister en Informatique, Université d'Oran, (2012).
- [17] K. Oguchi, S. Maruta, and D. Hanawa. Human positioning estimation method using received signal strength indicator (rssi) in a wireless sensor network. *Procedia Computer Science* 34, pp. 126-132, (2014).
- [18] N.A. Pantazis, S.N. Nikolidakis, and D.D. Vergados. Energy efficient routing protocols in wireless sensor networks : A survey. *IEEE COMMUNICATIONS SURVEYS and TUTORIALS* 15 (2), pp. 551-591, (2013).
- [19] M. Patil and R.C. Biradar. A survey on routing protocols in wireless sensor networks. *IEEE* , pp. 86-91, (2012).
- [20] G.H. Raghunandan and B.N. Lakshmi. A comparative analysis of routing techniques for wireless sensor networks. *IEEE Proceedings of the National Conference on Innovations in Emerging Technology*, pp. 17-22, (2011).
-

- [21] S. Rani, J. Malhotrab, and R. Talwar. Energy efficient chain based cooperative routing protocol for wsn. *Applied Soft Computing* 35, pp. 386-397, (2015).
- [22] S.Pal Singh and S.C.Sharma. A survey on cluster based routing protocols in wireless sensor networks. *Procedia Computer Science* 45 , pp. 687-695, (2015).
- [23] S.Varshney, C. Kumar, and A. Swaroop. A comparative study of hierarchical routing protocols in wireless sensor networks. *IEEE 2nd International Conference on Computing for Sustainable Global Development*, pp. 1018-1023, (2015).
- [24] L. Zhaohua, L. Jianfeng, and J. Guiquan. Survey of routing protocols in wireless sensor networks based on applications. *IEEE 2nd International Conference on Industrial and Information Systems*, pp. 381-385, (2010).

## RÉSUMÉ

Le développement qu'a connu ces dernières années les technologies a permis aux chercheurs et concepteurs des réseaux informatiques d'élaborer un nouveau type de réseau appelé "Réseaux de Capteurs Sans Fil (RCSFs)". Ces derniers représente un cas particulier des réseaux Ad hoc, ils sont composés d'un grand nombre de nœuds capteurs déployés dans le réseau de manière aléatoire ou déterministe, dont le but étant de recueillir des informations dans les milieu hostiles ou inaccessible et les transmettre à un nœud particulier appelé nœud puits (Sink) ou station de base, cette transmission doit s'effectuer avec un minimum de perte d'énergie afin de garantir une longue durée de vie du réseau. En effet, le problème majeur de ce type de réseau est la ressource énergétique qui est limitée, car cette dernière est fortement utilisée pour accomplir la communication de données. Dans ce mémoire nous avons pour but de proposer un nouveau protocole de routage hiérarchique (basé clustering) à basse consommation énergétique pour les RCSFs, ce type de routage permet de découper le réseau en clusters où chaque cluster est géré par un nœud particulier appelé (Cluster Head "CH") qui est utilisé pour l'agrégation de données collectées au sein de son cluster et de les transmettre à la station de base.

Le protocole R2E-ANFP, où chaque cluster est géré par deux types de nœuds (CH et sous CH), le premier type est élu par la station de base selon sa distance par rapport à cette dernière ou le niveau d'énergie, alors que le second type est élu par soit le CH ou le sous CH de niveau inférieur. La communication de données s'effectue en utilisant le multi-saut, les résultats de simulation ont montré que notre protocole est plus performant que LEACH (2000), MH-LEACH (2014) et CBCCP (2015) en termes de durée de vie du réseau et consommation d'énergie.

**Mots clés :** Réseaux de Capteurs Sans Fil, multi-saut, Clustering, agrégation, Routage hiérarchique, consommation énergétique, LEACH/MHLEACH/CBCCP, R2E-ANFP.

## ABSTRACT

The development has known in recent years technology has allowed researchers and designers of computer networks to develop a new type of network called "Wireless Sensor Networks (WSNs)". They represent a special case of Ad hoc networks, they are composed of a large number of sensors nodes that are deployed in a network in random or deterministic way, whose goal is to collect information in hostile or inaccessible areas and forward it to a particular node called sink node or base station, this transmission must be carried out with minimal loss of energy to ensure a long life time of the network. Indeed, the major problem with this type of network is the energy resource that is limited because it is heavily used to perform data communications. In this document we aim to propose a new hierarchical routing protocol (based clustering) low energy consumption for WSNs, this type of routing can cut the network into clusters, where each cluster is managed by a particular node called (Cluster Head "CH") which is used for aggregating data collected within its cluster and to transmit them to the base station.

The protocol R2E-ANFP, where each cluster is managed by two types of nodes (CH and sous CH), the first type is elected by the base station according to their distance from it or the energy level, while the second type is elected by either the CH or Sous CH on lower level. Data communication is performed using the multi-hop, the simulation results show that our protocol outperforms LEACH (2000), MH-LEACH (2014) and CBCCP (2015) in terms of life time of the network and energy consumption.

**Key words :** Wireless Sensor Networks, Multi-hop, Clustering, Aggregation, Hierarchical routing, Energy consumption, LEACH/MHLEACH/CBCCP, R2E-ANFP.