

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE
SCIENTIFIQUE
UNIVERSITE MOULOU D MAMMERI DE TIZI-OUZOU
FACULTE DE GENIE ELECTRIQUE ET INFORMATIQUE
DEPARTEMENT DE L'INFORMATIQUE



Mémoire de fin d'études en Master
académique



Domaine : Mathématique et informatique

Filière : Informatique.

Spécialité : Systèmes informatiques.

Réalisé par :

Melle : MADIOU Nabila

Melle : MERBOUTI Kahina

Thème

*Approches du clustering dans les réseaux de
capteurs sans fil*

Mémoire soutenu publiquement le 28/09/2017 devant le jury composé de :

Présidente : M^{elle} S. ILTACHE

Examineur : M^r N. AMIROUCHE

Promoteur : M^r S. TALBI

2016/2017

Résumé

De nos jours les réseaux de capteurs sans fil (RCSF) occupent de plus en plus une place importante dans le quotidien, donnant accès à des domaines de recherches attractifs et présentant des défis très intéressants tel que la consommation d'énergie. Les nœuds formant ces réseaux captent des informations sur l'environnement, les traitent et les communiquent à l'utilisateur final. Une parmi les solutions proposées consiste à améliorer l'efficacité énergétique des protocoles de communication et augmenter la durée de vie du réseau est le regroupement des nœuds en clusters et la sélection des leaders de ces derniers. Notre travail de recherche a pour objectif d'optimiser les communications dans les réseaux de capteurs sans fil homogènes en utilisant l'approche de clustering. Pour cela, nous avons proposé, en se basant sur les protocoles LEACH et EUCA, un nouveau protocole de routage hiérarchique Free Collision based Energy and Time Clustering Algorithm (FC-ETC). Cet algorithme utilise une fonction d'élection de cluster head en se basant sur une fonction permettant à ces derniers de se situer uniformément à travers le champ de captage tout en réduisant les communications et le risque de collision des notifications d'élection.

Abstract

Nowadays, wireless sensor networks (WSNs) are increasingly taking a dominant position in everyday life, giving access to attractive research domains with major challenges such as energy consumption. The nodes forming these networks capture information about the environment, process it and communicate it to the end user. One of the proposed solutions to improve the energy efficiency of communication protocols and increase the lifetime of the network is the clustering of nodes in clusters and selection of leaders. Our research aims to optimize communications in homogeneous wireless sensor networks using the clustering approach. In order to do this, we proposed, based on the LEACH and EUCA protocols, a new Free Collision based Energy and Time Clustering Algorithm (FC-ETC) hierarchical routing protocol that introduces, at the level of election of leaders, a mathematical model that eliminates collisions and elects the node has high energy capacity.

Remerciements

Nous présentons nos remerciements les plus sincères à notre promoteur Mr TALBI Said, qui nous a encadrés, pour ses conseils et son soutien.

Nous présentons nos gratitudes aux membres du jury qui ont bien voulu examiner et évaluer notre travail et qui nous font l'honneur de participer à la soutenance.

Nous tenons tout particulièrement à remercier :

Les responsables de la faculté des sciences.

Monsieur DAHOU Brahim ,Doctorant à l'université « USTHB » et membre du centre de recherche millitaire d' alger.

Monsieur DIBOUNE Abdelhani ,Doctorant à l'université « USTHB » d'alger.

Monsieur HADJI Baghdad ,Doctorant à l'université « USTHB » d'alger.

pour leur soutient et encouragement .

Nos remerciements vont aussi à tous ceux qui ont contribué de près ou de loin à la réalisation de ce travail.

Table des matières

Résumé	i
Abstract	ii
Remerciements	iv
Table des matières	viii
Liste des figures	ix
Liste des tableaux	x
1 Introduction Générale	17
1.1 Problématique	19
1.2 Objectif	20
2 Réseaux de Capteur sans Fil	21
2.1 Introduction	22
2.2 Définition d'un capteur sans fil	22
2.2.1 Architecture d'un capteur sans fil	22
2.2.2 Caractéristiques d'un capteur sans fil	23
2.3 Réseau de capteur sans fil	23
2.3.1 Architecture d'un réseau de capteur sans fils	23
2.3.2 Caractéristiques d'un réseau de capteur sans fils	25
2.3.3 Classification des réseaux de capteurs sans fils	25
2.3.4 Comparaison entre réseaux de capteurs et réseaux ad hoc	26
2.3.5 Facteurs et contraintes de conception d'un RCSF	27
2.3.6 Domaines d'application	29
2.4 Communication dans les RCSFs	33
2.4.1 Pile protocolaire	33
2.4.2 Standards de communication dans les RCSFs	35
2.5 Conclusion	37
3 Routage dans les Réseaux de Capteurs sans Fil	38
3.1 Introduction	39

3.2	Métriques de routage	39
3.3	Classification des protocoles de routage	41
3.3.1	Selon les paradigmes de communication	42
3.3.2	Selon le mode de fonctionnement du protocole	43
3.3.3	Selon le mode d'établissement des chemins	44
3.3.4	Selon la topologie du réseau	45
3.4	Routage plat	47
3.4.1	Directed Diffusion (DD)	47
3.4.2	Gradient-Based Routing (GBR)	48
3.4.3	Energy Aware Routing (EAR)	49
3.4.4	SAR (Seqential Assignment Routing)	50
3.4.5	SPIN (Sensor Protocols for Information via Negotiation)	51
3.5	Routage hiérarchique	52
3.5.1	MECN (Minimum Energy Communication Network) :	53
3.5.2	LEACH (Low Energy Adaptive Clustering Hierarchy) :	53
3.5.3	TEEN (Threshold sensitive Energy Efficient sensor Network protocol)	57
3.5.4	APTEEN (Adaptive Periodic TEEN)	57
3.5.5	HEED (Hybrid, Energy-Efficient Distributed Clustering)	58
3.5.6	UCA (Unéqual Clustering Algorithm)	58
3.5.7	PEGASIS (Power-Efficient Gathering in Sensor Information Systems)	59
3.5.8	HEERP (Hierarchical Energy Efficient Routing Protocol) :	60
3.5.9	DEEC (Distributed Energy Efficient Clustering Algorithm for Hetero- geneous Wireless Sensor Networks)	61
3.5.10	DEEC-LCH (DEEC- linear cluster handling)	61
3.5.11	EUCA (Enhanced Unequal Clustering Algorithm) :	62
3.6	<i>Synthèse</i>	63
3.7	Conclusion	63
4	Free Collision based Energy and Time Clustering Algorithm	64
4.1	Motivation	65
4.2	Considérations	65
4.2.1	Modèle du réseau	65
4.2.2	Modèle énergétique	65
4.3	Fonctionnement :	67
4.3.1	Initialisation	68
4.3.2	Calcul du timer	68
4.3.3	Algorithme général	68
4.4	Analyse théorique	69
4.4.1	Choix des paramètres de la fonction objective	69

4.4.2	Discussion des valeurs du timer	77
4.5	Analyse expérimentale	78
4.5.1	Environnement de la simulation TOSSIM	78
4.5.1.1	Le système d'exploitation TinyOs	78
4.5.1.2	Le Langage NesC :	80
4.5.1.3	Les outils de simulation :	80
4.5.2	Paramètre de simulation	81
4.5.3	Implémentation	81
4.5.4	Métrique de performance	85
4.6	Conclusion	85
5	Conclusion générale	86
	Bibliographie	88

Table des figures

2.1	Architecture d'un capteur sans fil	22
2.2	Architecture d'un réseau sans fil	24
2.3	Collecte des données à la demande	24
2.4	Collecte des données suite à un événement	25
2.5	Application des RCSFs dans le domaine militaire	30
2.6	Application des RCSFs dans le domaine médical	30
2.7	Application des RCSFs dans le domaine domestique	31
2.8	Application des RCSFs dans le domaine environnement et agricole	32
2.9	Application des RCSFs dans le domaine de surveillance	33
2.10	Pile protocolaire	33
2.11	La topologie du standard IEEE 802.15.1	36
2.12	La topologie du standard IEEE 802.15.4	37
3.1	Classification des protocoles de routages	42
3.2	Topologie plate	46
3.3	Le Clustering	47
3.4	Les phases de communication du protocole DD	48
3.5	phase d'initialisation du EAR	50
3.6	Fonctionnement du protocole SAR	51
3.7	Fonctionnement du protocole SPIN	52
3.8	Le protocole LEACH	54
3.9	Les opérations de la phase d'initialisation	55
3.10	Les phases d'un cycle du protocole LEACH	56
3.11	Les protocoles TEEN et APTEEN	58
3.12	Le protocole UCA	59
3.13	Le protocole PEGASIS	60
4.1	Modèle de consommation d'énergie	66
4.2	L'organigramme du fonctionnement du protocole FC-ETC	67
4.3	Graphe représentant la variation du timer selon les valeurs de λ	77
4.4	Graphe représentant la variation du timer selon l'énergie	78

Liste des tableaux

3.1	Comparaison des protocoles hiérarchique	63
4.1	Tableau de variation du timer T2	70
4.2	Tableau de variation du timer selon le scénario 1	70
4.3	Tableau de variation du timer selon le scénario 2	71
4.4	Tableau de variation du timer selon le scénario 3	72
4.5	Tableau de variation du timer selon le scénario 4	72
4.6	Tableau de variation du timer selon le scénario 5	73
4.7	Tableau de variation du timer selon le scénario 6	73
4.8	Tableau de variation du timer selon le scénario 7	74
4.9	Tableau des variation du timer pour le 1er cas	74
4.10	Tableau de variation du timer 2ème cas	75
4.11	Tableau de variation du timer 3ème cas	76
4.12	Tableau de variation du timer A=55	76
4.13	Paramètre de simulation	81

1 Introduction Générale

Avant propos Depuis leur création, les réseaux de communication sans fil ont connus un succès croissant au sein des communautés scientifiques et industrielles. Le média hertzien utilisé (ondes radio) offre en effet des propriétés uniques, qui peuvent être résumées en trois points : la facilité du déploiement, l'ubiquité de l'information et le coût réduit d'installation. Au cours d'évolution de cette technologie, le paradigme sans fil a vu naître diverses architectures dérivées, telles que : les réseaux cellulaires, les réseaux locaux sans fils et autres.

Durant cette dernière décennie, et grâce aux progrès faits, il est apparu un nouveau type de réseaux ad-hoc, qui sont les réseaux de capteurs sans fil (RCSF) ou bien Wireless Sensor Network (WSN). Ce type de réseaux est le résultat d'une fusion de deux pôles informatique moderne : les systèmes embarqués et les communications sans fil. Les RCSFs sont des réseaux sans infrastructure fixe, constitués d'un grand nombre de nœuds capteurs qui peuvent être déployés aléatoirement d'une façon rapide dans des zones sensibles et/ou difficilement accessibles pour surveiller un phénomène donné. Les nœuds capteurs appelées aussi "motes".

Le but général d'un réseau de capteur sans fil est le plus souvent de surveiller une zone, en prenant régulièrement des mesures environnementaux (température, pression, humidité,...) par les nœuds capteurs et de faire remonter les informations vers certains nœuds intermédiaires du réseau, capables de relayer l'information à grande échelle vers un centre de contrôle distant (station de base).

Les RCSF sont souvent considérés comme étant les successeurs des réseaux ad hoc. En effet, ils partagent avec les MANET (Mobile Ad hoc NETWORKS) plusieurs propriétés en commun, telles que l'absence d'infrastructure et les communications sans fil. Mais l'une des différences clés entre les deux architectures est le domaine d'application. Contrairement aux réseaux MANET, les RCSF ont su attirer un nombre croissant d'industriels, vu leur réalisme et leur apport concret. En effet, le besoin d'un suivi continu d'un environnement donné est assez courant dans diverses activités de la société. Les processus industriels, les applications militaires, le monitoring d'habitat, ainsi que l'agriculture de précision ne sont que quelques exemples d'une variété d'applications possibles. Grâce à ce potentiel riche en applications, les RCSFs ont attiré de grandes firmes à travers le monde, telles qu'IBM, Sun, Intel et Philips.

Les RCSFs présentent de nombreuses contraintes qui motivent de nombreux chercheurs, et qui résident principalement dans le fait que leurs ressources sont limitées en termes de communication, de calculs et d'énergie. En particulier, la contrainte liée à l'énergie est considérée comme un problème fondamental. En effet, tous les nœuds capteurs ont besoin d'énergie (batterie) pour fonctionner. De ce fait, la durée de vie globale du réseau RCSF dépend complètement de celle de la batterie[52].

Dans les RCSFs, une large partie de l'énergie est consommée lorsque les communications sont établies [11]. Par conséquent, les communications directes entre les nœuds et la Station

de Base (SB) ainsi que les transmissions fréquentes et de longue distance doivent être réduites au maximum afin de minimiser l'énergie consommée et prolonger la durée de vie du réseau[73].

L'organisation hiérarchique du réseau en clusters est l'une des solutions adoptée parmi plusieurs. Chaque cluster est géré par un nœud chef appelé "cluster-head". Dans cette organisation, les nœuds d'un cluster envoient leurs données directement au cluster-head correspondant (communication intra-cluster). Le nœud chef collecte les données des autres nœuds, les agrège pour les transmettre directement à la station de base (SB) ou bien d'autres cluster heads (communication inter-cluster).

Lors de la communication multi sauts les leaders les plus proches de la station de base épuisent rapidement leurs énergie car ils sont trop sollicités par le relais des données des autres cluster-heads, dans le cas contraire c'est les leaders les plus éloignés de la station qui épuiseront leurs énergie plus rapidement que ceux qui sont proches.

1.1 Problématique

A nos jours, plusieurs algorithmes de clustering sont proposés où l'élection des cluster heads (CHs) se fait selon une métrique spécifique ou bien une combinaison de métriques : l'identifiant, degré, k-densité, la mobilité des nœuds. Le principe consiste en premier lieu, de choisir aléatoirement et à tour de rôle les clusters-heads pour que la consommation d'énergie soit équilibrée à travers tout le réseau. Une fois que les Clusters Heads sont élus, la prochaine étape consiste à former les clusters de sorte que chaque nœud du RCSF recevra des messages d'avertissement de tous les leaders, et selon le Rapport Signal sur Bruit des messages reçus. Chaque nœud choisira son cluster-head qui présente un fort signal.

Le principal problème du protocole LEACH est la sélection aléatoire des clusters heads qui ne prend pas en compte l'énergie résiduelle des nœuds et la formation des clusters qui suit. En effet, tous les CHs peuvent être tous localisés dans une petite région de la zone globale du réseau d'où les nœuds peuvent captés les mêmes données (redondance), par contre le reste de la zone ne sera pas couverte par les CHs. Et cela donc permet d'avoir des nœuds qui seront isolés et nécessiteront plus d'énergie pour faire communiquer leurs données captées. De ce fait, la mauvaise répartition des headers, de plus les CHs élus de faible énergie peuvent engendrer le dysfonctionnement du réseau.

D'autre part le protocole EUCA, propose un processus de clustering basé sur un timer calculé en fonction de l'énergie du capteur, et offre une meilleure répartition des CHs. Le problème majeur de ce protocole est l'apparition des collisions dans le cas d'égalité de quantité d'énergie des capteurs appartenant au même rayon

1.2 Objectif

Afin d'optimiser les communication dans les reseaux réseaux de capteurs sans fil homogène toute en éliminant les collisions ,et d'avoir une meilleure répartition des cluster-heads sur toute la zone de couverture à tout instant, en se basant sur les principes de LEACH et EUCA, deux questions s'imposent :

1. Comment peut-on définir les CHs?
2. Comment procéder à la formation des clusters à moindre coût énergétique?

Nous répondrons à toutes ces questions dans ce présent travail.

Ce projet est composé de cinq chapitre :

Après l'introduction générale, dans le deuxième chapitre nous allons décrire les réseaux de capteurs sans fils, en évoquant leur caractéristiques, leurs architectures, leurs domaines d'applications ainsi que les contraintes qui peuvent influencer sur leur conception. Le troisième chapitre sera consacré au routage des réseaux de capteurs sans fil, nous allons aborder les métriques de routage, la classification des protocoles de routage, aussi des exemples seront décrit tout en mettant l'accent sur les protocoles LEACH, et EUCA, nous terminerons ce chapitre par une description de la problématique de l'élection des CH et des collisions qui peuvent se produire durant l'élection. Dans le quatrième chapitre, nous allons présenter notre solution au problème de collision et d'élection de CH, en décrivant le fonctionnement de l'algorithme proposé le "Free Collision based Energy an Timer Clustering Algorithm", ainsi nous allons faire une analyse théorique pour expliquer pas à pas notre solution, et nous terminerons par une analyse expérimentale. En fin nous allons clôturer notre mémoire par une conclusion.

2 Réseaux de Capteur sans Fil

2.1 Introduction

Les réseaux de capteurs sans fil ont connu récemment un grand essor suite aux avancées technologiques dans les domaines de l'informatique et la microélectronique, qui permet le développement des microcomposants (appelé micro capteur) qui intègrent dans un seul circuit, des dispositifs de captage et de communication sans fil. Ainsi ces micros capteurs peuvent être déployés dans des endroits à conditions rudes et accès difficile pour recueillir et transmettre les données de manière autonome

2.2 Définition d'un capteur sans fil

Un capteur sans fil est petit dispositif électronique autonome [45], mis dans un environnement afin de détecter un événement réel (température, humidité, mouvement...), puis les transformer en données binaires exploitable par un système informatique [69].

2.2.1 Architecture d'un capteur sans fil

Un capteur sans fil est principalement composé de quatre unités de base : unité d'acquisition, une unité de traitement, une unité de communication et une unité d'énergie[6].

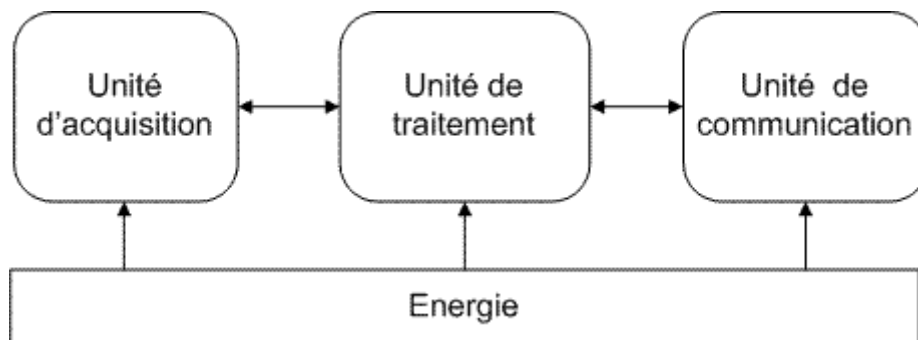


FIGURE 2.1 – Architecture d'un capteur sans fil

Unité d'acquisition : Est composée de deux sous unités un capteur et un convertisseur analogique numérique (CAN). Le capteur permet de fournir au CAN des signaux analogiques basés sur le phénomène observé. Le CAN convertit ces signaux en signal numérique compréhensible par l'unité de traitement [23].

Unité de traitement : Est composée d'un processeur et d'une petite mémoire, elle dispose aussi de deux interfaces une avec l'unité d'acquisition et l'autre avec l'unité de communication. Les informations sont acquises de l'unité d'acquisition est envoyées à l'unité de

communication[6]. Cette unité est chargée de l'exécution des protocoles de communication permettant ainsi au capteur de collaborer avec les autres capteurs du réseau[23].

Unité de communication : Cette unité est responsable de toutes les transmissions et réceptions des informations, elle est composée d'un émetteur /récepteur qui permet la communication entre les nœuds du réseau [70].

Unité d'énergie : C'est l'élément essentiel dans l'architecture des capteurs, cette unité alimente toute les autres unités du capteur, elle correspond généralement à une batterie ou une pile de ressources limitées[69].

2.2.2 Caractéristiques d'un capteur sans fil

Les capteurs sans fil peuvent varier considérablement, en ce qui concerne la taille, leur coût, la puissance de traitement, les protocoles et les systèmes d'exploitation. Les capteurs sont équipés d'émetteurs-récepteurs de radiofréquence de courte portée pour faciliter la recherche et la récupération des données dans le réseau. Les processeurs utilisés dans ces capteurs peuvent varier d'ultra faible puissance à des processeurs 32bits, de même pour l'espace mémoire qui peut varier de quelques kilooctets à l'ordre de mégaoctets[34].

2.3 Réseau de capteur sans fil

Les réseaux de capteur sans fil (RCSF) ou en anglais Wireless Sensor Network (WSN), sont considérés comme un type spécial des réseaux Ad Hoc, ils sont constitués d'un grand nombre de petits nœuds de détection (capteurs) qui jouent à la fois le rôle d'hôte et de routeur.

Les nœuds capteurs sont déployés de façon aléatoire sur une zone géographique afin de collecter les données sur leur environnement puis les transmettre à un ou plusieurs puits. La communication sans fil, la topologie dynamique ainsi que l'absence de contrôle centralisé rend les RCSF plus vulnérables et fragile par rapport aux réseaux filaires ou sans fil traditionnel[39, 33, 48, 29, 20].

2.3.1 Architecture d'un réseau de capteur sans fils

Un RCSF est composé d'un grand nombre de capteur dispersés en densité dans un terrain d'intérêt appelé zone de captage, le nombre de capteurs déployés est déterminé selon la taille de cette zone, la distance entre deux nœuds est en fonction de la puissance de communication.

Les capteurs collectent les données puis les transmettent via un routage multi-sauts a un point de collecte appelé puits (en anglais sink) ou station de base, ce dernier transmet ces données à l'utilisateur final[45, 70].

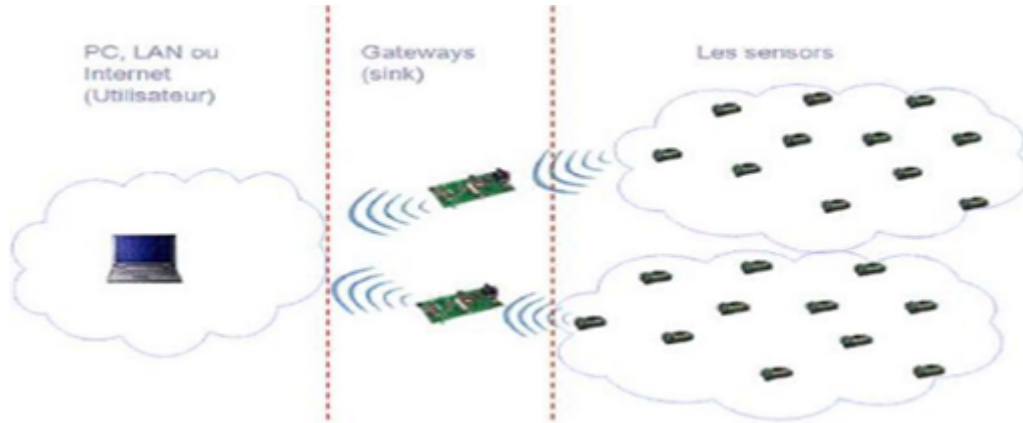


FIGURE 2.2 – Architecture d'un réseau sans fil

La collecte des données peut se faire de deux façons :

- *A la demande* : Si à un moment T en veux savoir l'état de de la zone de couverture, le puits émet un broadcast à toute la zone pour que les capteurs envoient leur dernier relevé vers le puits.

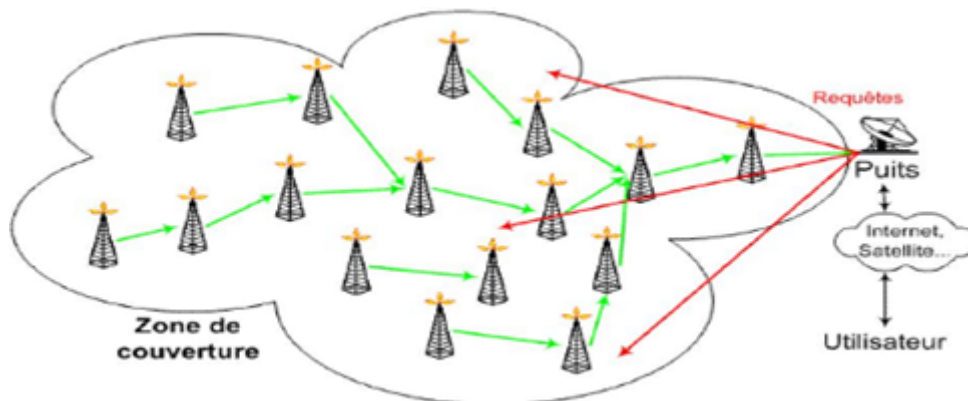


FIGURE 2.3 – Collecte des données à la demande

- *Suite à un évènement* : Si un évènement se produit dans un point de la zone de couverture (changement brusque de la température), les capteurs sis à proximité remontent les données relevées au puits.

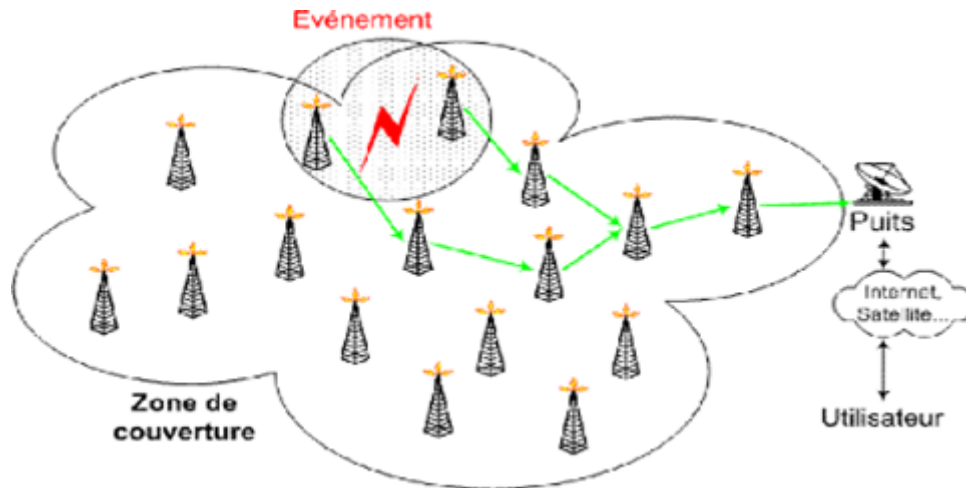


FIGURE 2.4 – Collecte des données suite à un événement

2.3.2 Caractéristiques d'un réseau de capteur sans fils

Les RCSFs sont caractérisés par :

La topologie dynamique due aux changements fréquents, ajout et suppression des capteurs.

- Absence d'infrastructure
- Faible débit
- Faible consommation d'énergie
- Les capteurs sont limités en taille de mémoire, en capacité de calcul et de traitement
- Le nombre de capteur dans un RCSF est plus élevé que dans un réseau traditionnel.
- Absence de sécurité

2.3.3 Classification des réseaux de capteurs sans fils

Les réseaux de capteur sans fil peuvent être classés selon quatre critères suivant :

Selon le mode d'acquisition et de livraison de données au puits : On distingue le modèle continu (time driven), le modèle événementiel (event driven) et le modèle requête (query driven) [81] :

- *Le modèle continu* : L'acquisition et la transmission des données se fait de façon périodique, la période peut être plus au moins longue, elle peut aller de quelques secondes jusqu'à quelques heures voir quelque jours.
- *Le modèle événementiel* : Les capteurs envoient les données lorsqu'un événement spécifique se produit [68].
- *Le modèle requête* : Dans ce cas-là, les capteurs envoient les informations suite à une demande explicite de la station de base [68].

Selon la distance entre les nœuds capteur et le puits : Dans cette classification on distingue les réseaux multi-sauts (multi-hop WSN) et les réseaux à un seul saut (single-hop WSN)[81].

- *Les réseaux multi-sauts :* Comme le puits ne se trouve pas à la portée du nœud capteur, l'envoi de données collecté nécessite le passage par des nœuds intermédiaires.
- *Les réseaux à un seul saut :* Les nœuds capteurs sont des voisins immédiats du puits, les données collectées sont alors envoyées de façon directe sans passer par des nœuds intermédiaires.

Selon le modèle de mobilité dans le réseau : Cette classification consiste en une combinaison entre la mobilité des nœuds capteurs et celle du puits, on peut distinguer deux grandes catégories de réseaux, les réseaux statiques (static networks) et les réseaux dynamiques ou mobiles (mobile networks)[81].

- *Les réseaux statiques :* Les nœuds capteurs ainsi que le puits sont fixes.
- *Les réseaux dynamiques :* On peut avoir dans un réseau des nœuds capteurs mobile et un puits fixe.

Selon les capacités des nœuds du réseau : On distingue les réseaux homogènes, et les réseaux hétérogènes :

- *Les réseaux homogènes :* Dans ce type de réseau les nœuds capteurs, le ou les puits ainsi que la passerelle ont la même capacité de calcul, de stockage et d'énergie.
- *Les réseaux hétérogènes :* En plus des nœuds capteurs ordinaires du réseau on trouve d'autres nœuds capteurs sophistiqués, ce qui rend le réseau plus performant.

2.3.4 Comparaison entre réseaux de capteurs et réseaux ad hoc

On comparant les RCSFs avec les réseaux Ad Hoc on a pu relever quelques points en communs ainsi que leurs différences[3, 81] :

- Tous les deux sont des réseaux sans fil, ce qui fait que la portée de communication est limitée.
- Les deux réseaux sont sans infrastructure, ils sont donc auto configurables.
- Les communications dans les deux réseaux sont vulnérables aux interférences car ils travaillent avec une bande de fréquence non propriétaire.
- Dans les deux réseaux les entités sont alimentées par des batteries.
- Dans les RCSFs les nœuds capteurs interagissent avec la nature ou l'environnement tandis que dans les réseaux Ad Hoc classiques les nœuds sont utilisés directement par l'être humain comme les PDA, portable. . .
- Le nombre de nœuds déployé dans les RCSFs peut être beaucoup plus élevé que dans les réseaux Ad Hoc.

- Les nœuds des RCSFs risquent de tomber en panne plus souvent que ceux des réseaux Ad Hoc, car ils sont déployés dans des endroits hostiles.
- Le changement de la topologie d'un RCSF est fréquent, suite aux pannes ou à la mobilité des nœuds capteurs.
- Le modèle de communication dans les RCSFs est « many-to-one » c'est-à-dire que les données sont collectées par plusieurs nœuds puis envoyées vers la station de base, par contre dans les réseaux Ad hoc les données circulent entre tous les nœuds et dans tous les sens ce modèle est appelé « any-to-any ».
- Les communications broadcast sont principalement utilisées dans les RCSFs alors que la plupart des réseaux Ad Hoc sont basés sur les communications point à point.
- Les nœuds dans un réseau Ad Hoc sont plus puissants et possèdent plus des capacités plus importantes que ceux des RCSFs qui sont caractérisés par une puissance, capacités de calcul et de mémoire limitées.
- A cause de leur nombre élevé, Les nœuds capteurs peuvent ne pas avoir d'identification globale (ID), alors que dans les réseaux ad hoc il y'a toujours la notion d'ID.
- Les RCSFs sont qualifié comme des réseaux à basse consommation et à bas débit, ce n'est pas le cas pour les réseaux Ad Hoc car leurs nœuds sont alimentés par des batteries facilement rechargeables.
- Dans un RCSF, les nœuds collaborent pour remplir un objectif, alors que dans les réseaux Ad Hoc chaque nœud a son propre objectif.
- Dans RCSF l'énergie est un facteur déterminant, et dans les réseaux ad hoc le débit est majeur.

2.3.5 Facteurs et contraintes de conception d'un RCSF

Les principaux facteurs et contraintes qui peuvent influencer la conception des RCSFs sont :

Tolérance aux pannes : Le principe de la tolérance aux pannes est de pouvoir maintenir le fonctionnement du réseau, en cas de défaillance d'un ou plusieurs nœuds capteurs. Cette défaillance peut être causée par la perte d'énergie, dommage physique, etc. On dit que la tolérance aux pannes est meilleure lorsque le nombre de nœuds en panne est grand et que le réseau maintien toujours son bon fonctionnement[3].

Passage à l'échelle : Dans un réseau de capteurs sans fil le nombre de nœuds capteur est variable, il peut aller de quelques entités à une dizaine de milliers, le réseau doit pouvoir fonctionner avec cette grande densité qui engendre beaucoup de communication inter-nodale aussi les puits doivent être doté d'une grande capacité mémoire pour stocker les informations reçues.

Système d'exploitation : Les systèmes d'exploitation pour capteurs en réseau sont spécifiquement conçus pour optimiser l'usage des ressources matérielles limitées dont ils disposent : une mémoire avec une petite capacité, un processeur de faible vitesse de traitement et une ressource énergétique très limitée. De nombreux systèmes d'exploitation spécialisés existent, parmi lesquels : TinyOS, ContikiOS, MantisOS, Nano-RK, LiteOS[3].

Sécurité limitée : Les RCSFs sont exposés aux attaques qui ciblent les données transmises, à cause de l'absence d'infrastructure et de la topologie dynamique ainsi que la communication sans fil. La limite des capteurs en ressources, ne permet pas d'appliquer les techniques conventionnelles pour faire face aux attaques.

Coût de production : Dans un réseau de capteur sans fil le nombre de capteurs est très important, ainsi l'évaluation globale du réseau dépend du coût d'un seul capteur. L'utilisation de cette nouvelle technologie ne serait pas rentable si le coût du réseau est supérieur à celui d'un réseau de capteur classique[3].

Environnement : Les RCSFs sont utilisés dans des endroits hostiles, les nœuds capteurs sont donc déployés dans des zones telles que des volcans en pleine activité, des champs biologiquement ou chimiquement souillés etc. Ils doivent être conçus d'une manière à pouvoir fonctionner sans surveillance dans de telles conditions[82].

Topologie : Dans les réseaux de capteurs sans fil un grand nombre de nœuds est déployé, ce qui nécessite une maintenance de la topologie, elle consiste en trois phases :

- Déploiement
- Post-déploiement (les capteurs peuvent bouger, ne plus fonctionner)
- Redéploiement de nœuds additionnels

Agrégation des données : Dans les RCSFs, les valeurs relevées par les nœuds capteurs sont très corrélées, ce qui engendre la réception de données redondantes par la station de base. Les capteurs ont de forte contrainte d'énergie, il est donc préférable de réduire la quantité des informations redondantes transmises pour économiser l'énergie. Une des techniques utilisées pour la réduction d'informations transmises est l'agrégation des données[70].

— *Le principe de cette technique est :*

Chaque nœud attend de recevoir la valeur de ces enfants avant d'envoyer quoi que ce soit. Il agrège, soit au fur à mesure soit tout d'un coup, ces valeurs entre elles et agrège aussi sa propre valeur, selon une fonction bien définie (comme par exemple le maximum), une fois terminé, ils transmettent le résultat à son parent et ainsi de suite.

Média de transition : Les nœuds capteur sont reliés de manière sans fil, le média de transmission doit être standardisé, afin de permettre des opérations sur les réseaux dans le monde. Le Bluetooth, l'infrarouge et les communications zigbee sont les plus utilisés.

Connectivité : On dit qu'un réseau est connecté si chaque nœud a la possibilité de communiquer avec n'importe quel nœud du réseau[3]. La connectivité est affectée par la topologie dynamique du réseau qui est causée par la mobilité, la défaillance des nœuds etc, ce qui engendre la perte des liens, les nœuds isolés, le partitionnement du réseau[49]. De ce fait il est utile d'envisager une connectivité plus puissante comme la k-connectivité. Dans un réseau k-connecté, il existe k chemins disjoints entre deux nœuds, ce qui permet de maintenir la connectivité même si k-1 nœuds tombent en panne.

Consommation d'énergie : Les nœuds capteurs sont équipés d'une batterie à énergie limitée, dont la recharge est coûteuse et parfois impossible. L'énergie doit être économisée pour prolonger la durée de vie de l'ensemble du réseau.[57] A cet effet, des périodes de veille sont prévues durant l'inactivité des capteurs actuels afin de préserver leur énergie[69].

Contraintes matérielles : La taille réduite des capteurs permet le déploiement facile du réseau d'une part, elle est aussi la source des contraintes matérielles des RCSFs, parmi ces contraintes : La puissance limitée des batteries alimentant les nœuds capteurs, pour cela la consommation d'énergie doit être moindre afin de prolonger la vie du réseau. Une faible capacité de calcul car les processeurs des nœuds capteurs sont différents de ceux d'une machine classique, ils utilisent plutôt des microcontrôleurs de faibles fréquences. Les nœuds capteurs doivent aussi être autonomes et résistants puisqu'ils sont déployés dans des environnements hostiles[3, 66, 81].

2.3.6 Domaines d'application

La communication sans fil, le faible coût, la miniaturisation ainsi que les différents types de capteurs existant on permet au RCSF de conquérir plusieurs domaines d'application parmi eux nous citons :

Applications militaires : Comme dans le cas de plusieurs technologies a été le moteur initial pour le développement des RCSFs, DNS (Distributed Sensor Network) était l'un des premiers projets ayant utilisés ces réseaux dans les années 80. Les RCSFs sont caractérisés par l'auto-configuration, le déploiement rapide et la tolérance aux pannes, ce qui fait d'eux un outil appréciable dans un tel domaine[66, 67]. Déployés dans des endroits stratégiques ou difficiles d'accès, les RCSFs peuvent être utilisés pour[45] :

- Contrôler des équipements et des munitions
- Analyser et surveiller un champ de bataille

- Détecter et poursuivre des objets d'ennemis
- Protéger les éléments sensibles comme les bases nucléaires
- Détecter et reconnaître des attaques nucléaires, biologiques et chimiques.



FIGURE 2.5 – Application des RCSFs dans le domaine militaire

Applications médicales : Les RCSFs sont de plus en plus utilisés dans le domaine de la médecine, ils peuvent assurer la surveillance permanente des organes vitaux grâce à des micro-capteurs qui peuvent être avalés ou implantés sous la peau, faciliter le diagnostic de quelques maladies en relevant des mesures physiologiques telles que la tension artérielle, le battement du cœur etc. Ces données peuvent être stockées pendant une longue durée pour le suivi d'un patient. Aussi ils peuvent minimiser la possibilité de prescrire de mauvais médicaments aux patients à qui on intègre des capteurs qui servent à identifier leurs allergies et les médicaments qui leur seront nécessaires [53, 9, 40].

De plus les RCSFs contribuent à la résolution de certains problèmes sociaux, tels que le gardiennage des malades chroniques des personnes âgées et des personnes handicapées mentales et physiques [10].

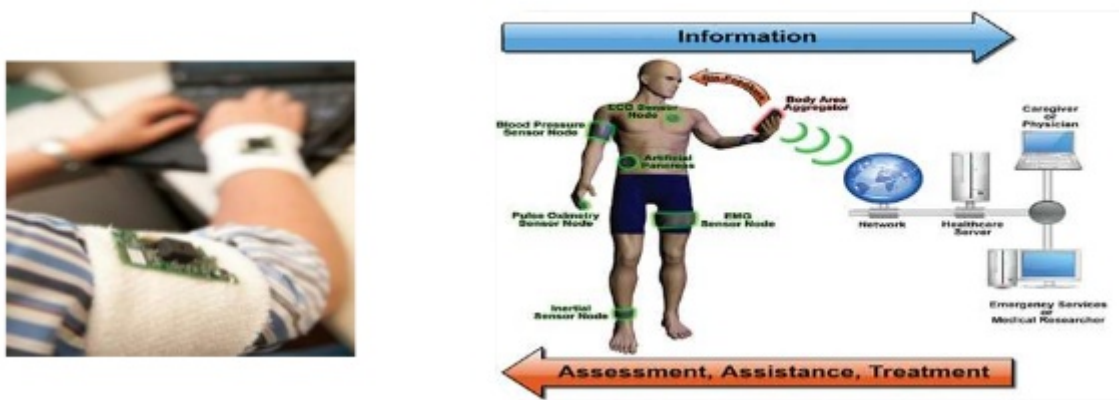


FIGURE 2.6 – Application des RCSFs dans le domaine médical

Applications domestiques : Les capteurs peuvent être embarqués dans des appareils tels que les fours, les réfrigérateurs etc. L'utilisateur a la possibilité de contrôler les appareils

domestique localement ou à distance grâce à l'interaction de ces capteurs externe entre eux et avec un réseau externe via internet, ce qui va faciliter les activités domestiques quotidiennes comme l'automatisation de l'activation/extinction de la lumière, mise en marche de la télévision ou le chauffage.

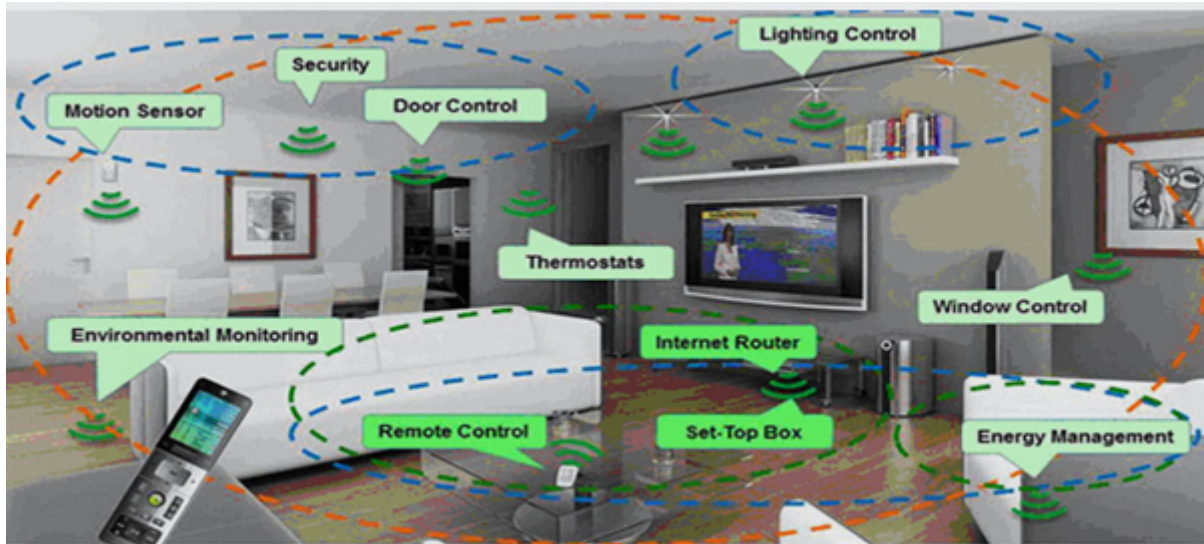


FIGURE 2.7 – Application des RCSFs dans le domaine domestique

Applications environnementales et agricoles : L'homme a toujours cherché à observer, comprendre et connaître son environnement qui constitue un axe important pour son devenir. De ce fait, l'utilisation des RCSFs pour le contrôle des paramètres environnementaux peut donner naissance à plusieurs application tel que[45, 83] :

- La détection et la prévention contre les désastres comme les inondations, la pollution et notamment les feux de forêts en déployant, dans ce cas, des thermo-capteurs capables d'évaluer les risques d'un éventuel début d'incendie grâce à des relevés périodiques de température ;
- Dans l'agriculture, les contrôles du climat et de l'état de la terre permettent d'augmenter la précision et l'efficacité des activités agricoles telles que l'irrigation et l'utilisation d'engrais ;
- L'étude et l'observation de la faune et de la flore dans les grandes forêts inaccessibles ou dans les profondeurs sous-marines où la présence de l'homme est quasi impossible ;
- L'exploration et la surveillance des environnements hostiles tels que les volcans ou les régions toxiques.



FIGURE 2.8 – Application des RCSFs dans le domaine environnement et agricole

Applications de surveillance : Ces dernières années des recherches ont été lancées afin de développer une technologie qui s'applique à l'utilisation des capteurs pour la surveillance des structures telles que les bâtiments, les ponts..., son objectif est de développer une structure intelligente qui sera capable de se diagnostiquer automatiquement, et d'établir des priorités pour la réparation en cas de problèmes. Cette technologie est très utile pour les zones sismiques, bien que les tremblements doux ne provoquent pas des dommages visibles, mais ils peuvent provoquer des fissures cachées qui pourraient éventuellement échouer lors d'un tremblement de terre d'une magnitude élevée.

En outre, le véritable état d'une structure après un tremblement doux ne peut apparaître sans mesure précise, les capteurs développés par les ingénieurs sont promoteurs à cet égard.



FIGURE 2.9 – Application des RCSFs dans le domaine de surveillance

2.4 Communication dans les RCSFs

dans cette section nous allons décrire la pile protocolaire et les différents standards de communication dans les réseaux de capteurs sans fil.

2.4.1 Pile protocolaire

Le rôle de la pile protocolaire consiste à standardiser la communication entre les participants afin que différents constructeurs puissent mettre au point des produits (logiciels ou matériels) compatibles[49]. Cette pile comprend cinq couches ; la couche physique, la couche liaison de données, la couche réseau, la couche transport et la couche application, ainsi que trois niveaux (plans) de gestion (énergie, mobilité, tâche) qui dépendent aux besoins des RCSFs.

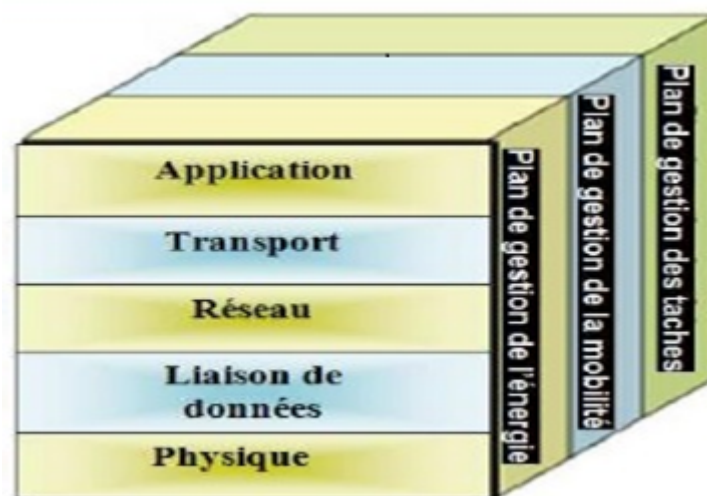


FIGURE 2.10 – Pile protocolaire

Couche Physique : Elle gère le support de communication qui assure l'acheminement des données entre les capteurs, elle est donc responsable de[90] :

- La sélection de la fréquence
- La génération de la porteuse
- La détection du signal
- La modulation et le cryptage des données.

Couche liaison de données : Elle est chargée d'assurer l'accès au média physique, de détecter et corriger les erreurs survenues au niveau de la couche physique, aussi elle établit une connexion fiable (point à point ou point à multipoints) selon la connexion du réseau[2, 81].

La couche réseau : Elle est responsable du routage des données fournies par la couche transport, elle assure ainsi l'acheminement des données via le réseau en établissant les routes entre les nœuds capteurs et le puits et sélectionnant le meilleur entre eux en terme d'énergie, de délais de transmission, de débit etc [2, 49].

La couche transport : Cette couche assure la vérification du bon acheminement des données ainsi que la qualité de transmission. Si le RCSF est prévu d'être accessible à partir d'autres réseaux externes comme Internet, les protocoles de la couche transport seront nécessaires. Etant donné que le protocole de transport universel TCP (Transmission Control Protocol) n'est pas compatible avec les environnements des RCSFs, une interface TCP Splitting est nécessaire pour permettre l'interaction de ces derniers avec les réseaux externes.

D'autre part un protocole proche d'UDP appelé UDP Like (User Datagram Protocol Like) est utilisé comme protocole de transport pour la communication entre les nœuds capteurs et les puits, à cause de la limitation de mémoire, aussi les UDP possèdent une taille réduite ce qui permet d'économiser la bande passante et l'énergie de transmission[2, 43].

La couche application : Elle constitue l'ensemble des applications implémentées sur un RCSF, ces applications doivent fournir des mécanismes qui permettent à l'utilisateur d'interagir avec le réseau à travers des interfaces, Elles doivent aussi rendre transparents le matériel et les logiciels utilisés dans les couches inférieures[29].

Plan de gestion de l'énergie : Les nœuds capteurs sont dotés d'une source énergétique limitée, généralement non rechargeable. Ainsi sa durée de vie dépend de sa consommation d'énergie, c'est pour cela il faut minimiser la consommation d'énergie pour prolonger la durée de vie du nœud. Des fonctions sont intégrées à ce niveau pour la gestion de la consommation d'énergie par le capteur, comme par exemple la mise en veille après la réception d'un message d'un nœud voisin, pour éviter la réception des messages dupliqués. Aussi un capteur

atteignant un niveau faible d'énergie, peut diffuser un messages aux autres capteurs pour ne pas participer aux tâches de routage, et conserver son énergie pour la capture[29].

Plan de gestion de la mobilité : Pour permettre aux nœuds capteurs de garder continuellement un chemin vers le nœud puits et de maintenir une image récente des nœuds voisins, le niveau de la gestion de la mobilité détecte et enregistre tous leurs mouvements[29].

Plan de gestion des tâches : Les nœuds capteurs d'un réseau ne doivent pas obligatoirement travailler avec le même rythme, cela dépend essentiellement de la nature du capteur, son niveau d'énergie et sa région de déploiement, en tenant compte de ces considérations, le niveau de gestion des tâches assure l'équilibrage et la distribution des tâches sur les différents nœuds du réseau afin d'assurer un travail coopératif et efficace en matière de consommation d'énergie, et par conséquent, prolonger la durée de vie du réseau[81].

2.4.2 Standards de communication dans les RCSFs

La standardisation est une solution au problème d'interopérabilité dans les nouvelles technologies, permettant ainsi de créer une base commune de travail pour tout intervenant dans le domaine, afin de pouvoir produire du matériel et du logiciel compatibles, parmi ces standards nous citons :

IEEE 802.15.1 : Prend charge la norme Bluetooth qui est un standard de communication sans fil, conçu comme un réseau WAP (Wireless Personal Area Network) avec une application qui permet de relier des périphériques tels que les imprimantes, les scanners, les souris, les claviers etc, avec un ordinateur. Cette technologie a été déjà utilisée dans les RCSFs comme un moyen de communication, deux topologies sont définies dans cette norme, piconet et scatternet[31, 21, 65, 49].

Un piconet est formé d'un nœud qui sert de maître et jusqu'à sept esclaves actifs, tous ces nœuds sont synchronisés en utilisant l'horloge du maître, les esclaves ne communiquent qu'avec leur maître en mode point à point, les transmissions du maître peuvent être point à point ou point à multipoint[21, 36].

Un scatternet est un groupe de piconets opérationnels qui se chevauchent dans le temps et l'espace, un nœud peut participer à plusieurs piconets au même temps, ce qui donne la possibilité de circuler l'information au-delà de la zone de couverture du piconet. Dans un scatternet un nœud peut être esclave dans plusieurs piconets mais maître dans un seul d'entre eux[36].

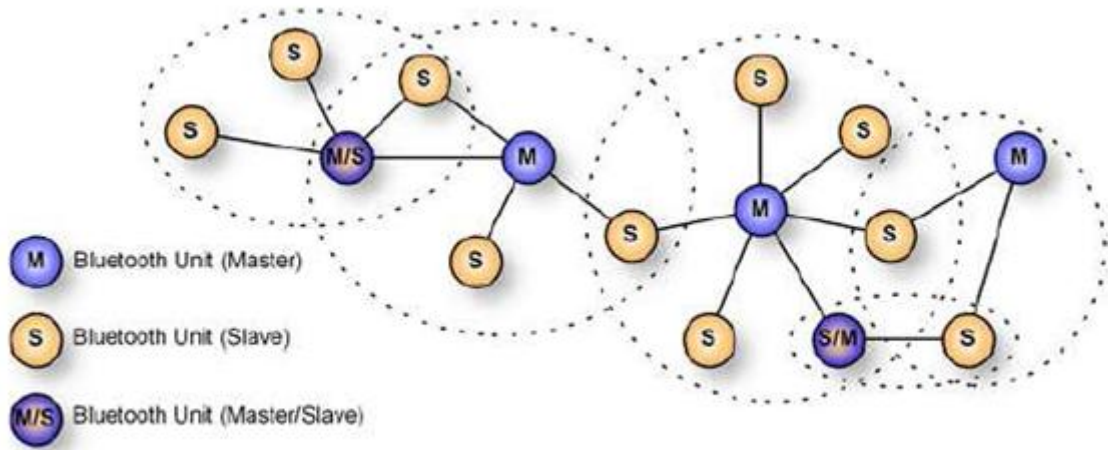


FIGURE 2.11 – La topologie du standard IEEE 802.15.1

IEEE 802.15.3 : Définit la norme UWB (Ultra-Wide Band), elle met en œuvre une technologie de communication sans fil de courte distance très spéciale, caractérisée par sa faible consommation énergie et son haut débit. Elle a l'avantage et la mobilité de la communication sans fil qui assure l'interconnexion à une grande vitesse entre les dispositifs numériques, sa particularité étant de transmettre des successions d'impulsions très courtes[21, 25].

IEEE 802.15.4 : Protocole de communication destiné aux réseaux sans fil de la famille des LRWPAN (Low Rate Wireless personal Area Network) caractérisés par leur faible consommation d'énergie, leur faible portée et d'un faible débit des nœuds utilisant ce protocole. Etant donné que le principal but n'est pas d'atteindre un débit élevé mais de maximiser la durée de vie du réseau.

Le standard IEEE 802.15.4 supporte différentes topologies, paire à paire (peer to peer), étoile (star) et la topologie arbre de cellules "Cluster tree" qui combine les deux topologies précédentes. Dans cette norme on distingue deux type de nœuds ; les nœuds avec une charge complète (Full Function Device (FFD)) et les nœuds avec une charge réduite (Reduced Function Device (RFD)). Le FFD peut communiquer avec des FFD et des RFD, et peut ainsi jouer le rôle de coordinateur du réseau, de routeur (coordinateur) ou de simple capteur. Le RFD est un nœud final, il ne peut communiquer qu'avec des FFD (coordinateurs) et ne peut participer au routage dans le réseau. Ils ne sont ainsi que des nœuds terminaux dans le réseau avec des ressources moindre en comparaison avec des FFD. Dans cette norme, la topologie en étoile met l'accent sur la durée de vie des batteries puisque chaque RFD est relié directement au coordonnateur. Par contre la topologie paire à paire s'intéresse à la fiabilité et à la scalabilité puisque tous les nœuds sont des FFDs et peuvent donc être reliés ensemble[21, 4, 19, 32].

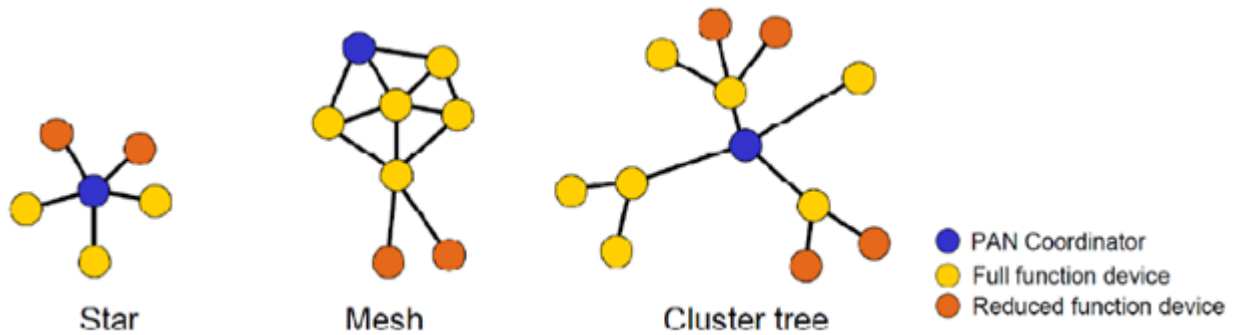


FIGURE 2.12 – La topologie du standard IEEE 802.15.4

Zigbee Alliance : Zigbee est le nom d’une suite de protocole de hauts niveaux basée sur le standard IEEE 802.15.4 pour les réseaux personnels sans fil (WPAN : Wireless Personal Area Network), elle est moins connue que le Bluetooth. En 2002, une association d’entreprises a constitué Zigbee Alliance, dans le but de développer des produits de contrôle fiables avec un coût réduit, et une faible consommation d’énergie. Ces produits doivent pouvoir être gérés par un réseau sans fil en utilisant une norme standard globale. Trois classes de composants sont définies par Zigbee[3] :

- Coordinateur Zigbee (Zigbee Coordinator ; ZC) : il est unique dans le réseau, il est responsable de la formation du réseau et peut aussi faciliter le routage des messages.
- Routeur Zigbee (Zigbee Router ; ZR) : Les ZR participent au routage et peuvent également capter ou déclencher une application
- Composant d’extrémité Zigbee (End Device Zigbee ; ZED) : Les ZED exécutent seulement les applications et ne participent pas au routage des messages, chaque ZED est tenu de faire un rapport à un ZR ou à un ZC.

2.5 Conclusion

Dans ce chapitre nous avons présenté une description générale portant sur les réseaux de capteur sans fil. Nous avons décrit les caractéristiques et l’architecture d’un capteur, on a aussi mentionné les caractéristiques de ces réseaux par la suite on a présenté leurs architectures ainsi que quelques domaines d’applications, ce qui nous a permis de relever quelques facteurs et contraintes qui faussent la route à leur développement, tels que la consommation d’énergie réduite, la scalabilité.

Le choix d’un protocole et d’un algorithme adéquat pour le routage d’information dans les réseaux de capteurs s’impose, ce qui va être étudié et présenté dans le prochain chapitre.

3 Routage dans les Réseaux de Capteurs sans Fil

3.1 Introduction

Le routage est le processus par lequel des chemins sont sélectionnés dans un réseau pour acheminer les paquets de données d'un nœud émetteur jusqu'à la station de base à travers le réseau. En effet, le rôle des techniques de routage consiste à assurer un bon acheminement des données pour prolonger la durée de vie du réseau qui est en dépendance directe avec la consommation d'énergie.

Les caractéristiques et les contraintes tel que : Tolérance aux pannes, consommation d'énergie, limitations de capacités des nœuds, scalabilité, connectivité, modes de transmission de données, hétérogénéité, imposées par les RCSFs exigent de développer de nouveaux protocoles de routage spécifiques qui diffèrent de ceux déployés dans les réseaux traditionnels, ou bien d'améliorer ceux utilisés dans les réseaux ad-hoc.

Le fonctionnement d'un protocole de routage peut être divisé en trois étapes : la découverte des nœuds voisins, le calcul de la métrique et le calcul des itinéraires optimums.

3.2 Métriques de routage

Une métrique de routage est une valeur attribuée à un lien ou à une route dans un réseau. Elle permet de déterminer si la route est de bonne qualité ou non. Elles sont généralement appelées coûts ou poids calculées selon un ou plusieurs paramètres tels que : le nombre de saut, le taux de perte, le taux d'interférence, le délais, la longueur de chemin (distance), l'énergie, etc. Généralement plus la métrique est petite (optimale), plus le protocole de routage considère que le chemin emprunté pour atteindre SB à travers ce nœud intermédiaire est optimal [85]. De plus, les métriques peuvent être considérées seules ou combinées (hybrides) selon les besoins du protocole[55]. Le fait de combiner plusieurs paramètres augmente les probabilités d'avoir des métriques précises qui correspondent à la qualité du lien ou de la route[85].

Une métrique doit pouvoir quantifier de manière précise l'état du réseau pour que le protocole de routage puisse choisir les routes les plus adéquates en tout temps. La récupération de ces informations peut se faire avec différentes méthodes :

a. Les méthodes actives : Consiste à injecter périodiquement des paquets de contrôle dans le réseau et récupérer les informations utiles telles que le taux de perte, le délai, la puissance de réception etc. Prenant comme exemple les métriques : Expected Transmission Count, Expected Transmission Time

Les avantages : L'avantage principal est que les mesures sont relativement précises

Les inconvénients :

- Les paquets de contrôle sont différents du trafic réel (taille des paquets, débit) et ne reflètent pas toujours la réalité.
- L'envoi périodique de paquets consomment des ressources notamment la bande passante qui peut créer une surcharge du réseau (overhead).

b. Les méthodes passives : Utilisent le trafic courant pour effectuer ces mesures.

Les avantages : Ces méthodes évitent la surcharge inutile du réseau.

Les inconvénients : Les mesures ne peuvent se faire qu'en présence d'un trafic suffisant, ce qui ne peut pas être garanti à l'avance. L'hétérogénéité du trafic peut aussi fausser les mesures.

c. Les méthodes hybrides : Utilisent en même temps les méthodes passives et actives. Si le trafic est suffisant les mesures sont faites grâce à une méthode passive, sinon on utilise les paquets de contrôle pour faire les mesures. Ces méthodes regroupent les avantages des deux méthodes précédentes[38].

Dans un réseau RCSF plusieurs paramètres peuvent influencer sur le routage. Il est donc important de prendre en considération le maximum de métriques possible. Dans cette partie, nous allons présenter quelques contributions sur les métriques de routages :

La consommation énergétique : L'idée est de calculer l'énergie disponible (résiduelle) (ED) pour chaque nœud du réseau et l'énergie nécessaire (EN) pour les transmissions des paquets entre une paire de nœuds. Les routes entre les nœuds et le puits sont établies et chacune d'elles est caractérisée par la somme des ED des nœuds qui la constituent et par la somme des EN des liaisons qui la construisent. La consommation d'énergie suit plusieurs approches dont on peut citer :

- *Par considération de puissance* : La route choisie est celle caractérisée par la somme des ED la plus élevée.
- *Par considération du coût* : La route choisie est celle caractérisée par la plus petite somme des EN.
- *Par considération de puissance et du coût* : Cette métrique est la combinaison des deux métriques précédentes. La route choisie est celle caractérisée par la plus petite somme des EN et la plus grande somme des ED.

Nombre de saut : Le principe consiste à choisir le plus court chemin en termes de nombre de sauts lors d'une transmission d'un paquet de données du nœud source vers la (SB) dans un réseau. Chaque chemin a un coût de 1 au minimum. le nombre de sauts(le

nombre de nœuds intermédiaires) est calculé durant la découverte des routes, la métrique nombre de sauts ne nécessite pas beaucoup de paquet de contrôle[85].

Perte de paquets : L'idée est de calculer le ratio des paquets perdus et des paquets émis transitant dans le réseau. Autrement dit, on calcule le nombre de paquets perdus sur le nombre de paquets transmis lors d'une transmission. Dans le cas où le taux de perte de paquets est élevé, il est nécessaire de mettre en place des mécanismes qui permettent de minimiser les collisions[27].

Délai de bout-en-bout : Le délai de bout-en-bout, en anglais EED (End-to-End Delay) est le temps moyen nécessaire pour qu'un paquet de données soit acheminé à partir de la source vers la destination [HNA06]. Cette technique est parmi les métriques les plus connues dans les réseaux sans fil. Le but est de minimiser le temps de propagation des paquets de données échangés pendant le routage[27].

La durée de vie : La durée de vie d'un Rcsf est liée directement à la consommation énergétique du réseau. Plus la consommation d'énergie diminue, plus la durée de vie du réseau augmente.

L'application doit s'exécuter au niveau des nœuds de capteurs aussi longtemps que possible. Par conséquent, les protocoles visant cette préoccupation, tentent d'équilibrer la consommation d'énergie de manière égale entre les nœuds en prenant en compte leurs niveaux d'énergie résiduelle[28]. Cependant, la métrique utilisée pour déterminer la durée de vie du réseau dépend aussi de l'application. La plupart des protocoles considèrent que la durée de vie du réseau est la période de temps jusqu'à ce que le premier nœud meure.

3.3 Classification des protocoles de routage

Plusieurs travaux de recherches ont été effectués dans le domaine des réseaux de capteurs sans fil et qui ont abouti à une conception d'une multitude de protocoles de routage dédiés à ces réseaux. Vu leurs multiplicité, il semble important de les classifiés selon un certain nombre de critères comme l'illustre la figure ci-dessous :

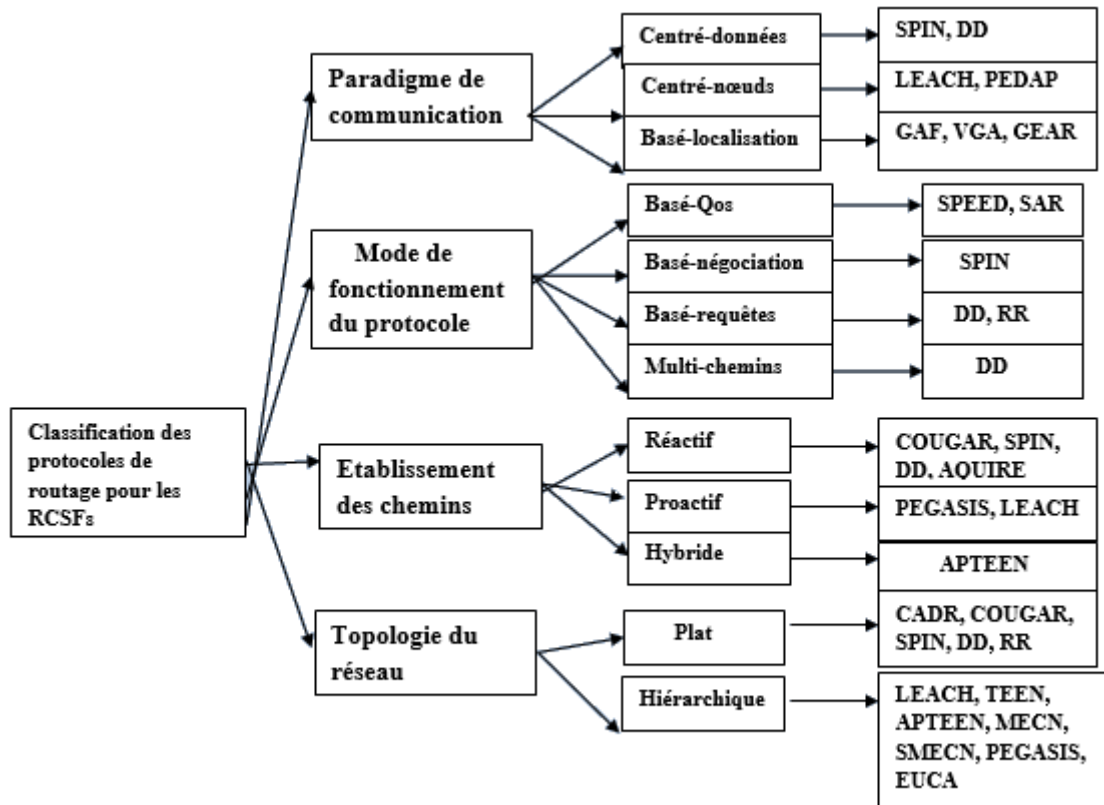


FIGURE 3.1 – Classification des protocoles de routages

3.3.1 Selon les paradigmes de communication

Le paradigme de communication détermine comment les nœuds du réseau sont interrogés. Dans les RCSFs, il existe trois sortes de paradigmes de communication[16] :

a. Centré-données : Dans les RCSFs, le principe de ce paradigme est généralement de donner l'importance aux données disponibles au niveau de nœud capteur plus que le nœud lui-même[82]. Le routage se fait en fonction des données associées aux capteurs car l'identification (IP, MAC) des nœuds est difficile vue leur nombre élevé. De plus, l'absence des identificateurs et le déploiement aléatoire des nœuds rendent la sélection difficile d'un groupe de nœuds afin d'être interrogés. Une des solutions proposées pour y remédier est l'utilisation des protocoles centré-données qui visent aussi l'utilisation de l'agrégation de données afin d'éviter le gaspillage d'énergie. Parmi ces protocoles on cite : DD, SPIN[44].

b. Centré-nœuds (node-centric) : Le principe de ce paradigme est basé sur l'utilisation de l'identification individuelle des nœuds participants à la communication dans le réseau. Or cette technique est très gourmande en terme de ressources ce qui rend ce paradigme peu utilisé dans les RCSFs mais reste à ne pas l'écarter car certaines applications nécessitent l'interrogation individuelle des nœuds. Par contre ce type de paradigme est utilisé beaucoup

plus dans les réseaux conventionnels et réseau Ad-hoc. Comme exemple de protocole on cite : PEDDA

c. Basé-localisation (position-centric) : Le principe de communication de ce paradigme est basé sur la localisation des nœuds dans le réseau et le routage s'effectue grâce à des techniques géographiques afin d'acheminer les données d'une sous-région à une autre. Avant d'envoyer ses données à un nœud destinataire, le nœud émetteur utilise une technique pour déterminer sa localisation géographique dont le degré de précision dépend de l'application en question. Vu que la localisation par le système GPS est trop coûteuse pour les RCSFs, plusieurs méthodes de positionnement des nœuds capteurs ont été développées, comme la triangulation. Parmi les protocoles position-centric on peut citer : GAF (Geographic Adaptive Fidelity)[86] , GEAR (Geographic and Energy Aware Routing)[87, 50].

3.3.2 Selon le mode de fonctionnement du protocole

Le mode de fonctionnement des protocoles de routage définit la manière avec laquelle les données sont acheminées dans le réseau. Selon ce critère on distingue quatre classes (catégories) :

a. Basé-Qos : Dans cette approche, le protocole doit satisfaire certain exigence de la qualité de service(QOS) comme le niveau de fiabilité, le délai de réponse, largeur de bande passante, etc, en équilibrant la consommation énergétique. Cette classe de protocoles est très recommandées pour les applications à contraintes temps réel telle que applications de surveillance d'aviation, centrale nucléaire, militaires, etc, où les données doivent être acheminées en temps réel vers le nœud puits et tout retard peut causé des dégâts catastrophiques sur la vie humain [45, 81]. Les protocoles SAR(Sequential Assignment Routing) et SPEED, sont parmi les premiers protocoles de routage pour les RCSFs qui ont introduit la notion de la QOS dans les décisions de routage[72].

b. Basé-Négociation : Afin d'éliminer les inondations dans les RCSFs , des nouveaux protocoles basés sur la négociation ont été développés. Avant transmission, le nœud émetteur négocie ses données en envoyant des messages appelés métadonnées (descripteurs de données) vers ses voisins qui vérifient à leur niveau l'existence ou non des données en question. Cette technique assure que seuls les nœuds intéressés qui vont recevoir les données complètes, ce mécanisme est exécuté au niveau de chaque nœud recevant ces données .De cette manière, le problème de redondance des données est éliminé. SPIN et SAR sont des protocoles de cette catégorie[46, 81].

c. Basé-Requête : Dans cette approche de routage, le nœud puits débute l'interrogation du système par diffusion des requêtes vers ses voisins. Ces requêtes sont exprimés soit par

un schéma valeur-attribut ou bien par un langage spécifique comme SQL (Structured Query Language) [81]. La réponse des nœuds disposant la donnée requise doit être acheminée selon le chemin inverse de la requête vers l'émetteur de cette dernière. Directed Diffusion(DD) et Rumor Routing (RR) [BRA02] sont des exemples de protocoles de cette approche [15].

d. Multi-Chemin : Dans cette approche, les protocoles de routage construisent une multiplicité de chemins complètement indépendants afin d'augmenter la performance et la fiabilité du réseau. Dans le cas de défaillance du chemin primaire choisi, les protocoles permet de trouver des chemins alternatifs (chemins de secours) entre la source et la destination. Ce qui permet d'assurer une meilleure tolérance aux pannes, malgré cet avantage ces protocoles requièrent plus de ressources en termes d'énergie et de messages de contrôles. Parmi les protocoles basé sur cette approche on cite : DD, SPIN, EAR[45, 58].

3.3.3 Selon le mode d'établissement des chemins

Selon la manière de créer et de maintenir les routes pendant le routage des données, on peut classifie les protocoles en trois classes suivantes :

a. Proactifs Le principe de base est d'établir à l'avance des tables de routage au niveau de chaque nœud, et de les maintenir périodiquement. En effet, tous les nœuds capteurs échangent, périodiquement leurs tables de routage avec leurs voisins afin de garder une vision permanente sur l'état de tous les liens disponibles utilisés ou non, et sur l'état de la topologie du réseau. Grace à cette dynamicité des tables qu'un protocole peut déterminer le chemin optimal depuis n'importe quel nœud vers n'importe quelle destination[41].

L'avantage principal de cette technique est que le délai de réponse est plus court. Par contre, si le réseau est dense les tables deviennent volumineuses et difficile à gérer d'où l'utilisation constante de la bande passante. Les protocoles proactifs sont conseillés dans la communication à courte portée.

Pour un fonctionnement optimal de ce type de protocoles, un compromis entre l'échange des informations de routage et la prise en compte de l'évolution de la topologie du réseau doit être trouvé.[cour madam chamek] OLSR, DSDV (Destination Sequenced Distance Vector routing), FSR sont des exemples de cette catégorie de protocoles[2, 5].

b. Réactifs Les protocoles de routage réactifs ne maintiennent une route que si elle est utilisée. Lorsqu'un nœud source a besoin de transmettre des données vers un nœud destination, il doit au préalable déterminer une route, pour cela, des paquets de contrôle sont diffusés sur le réseau. Les protocoles réactifs ne conservent que les routes qui ont une utilité. Par conséquent, la taille des tables de routage est moins importante que pour les protocoles

proactifs. Le chemin trouvé est maintenu dans la table de routage jusqu'à ce que la destination soit inaccessible à partir du nœud source ou bien que la source n'aura plus besoin de cette route.

Pour certaines applications nécessitant un minimum de réactivité, l'étape de recherche de route peut être problématique. Ce délai n'est seulement occasionné au début de l'échange d'information ou lorsqu'une route est rompue (lors du déplacement d'un nœud par exemple). Citons comme exemple DSR (Dynamic Source Routing)[41].

c. Hybrides Les protocoles de routage hybrides combinent les avantages des protocoles proactifs et réactifs. Lorsqu'il faut traverser un grand nombre de nœuds, les protocoles réactifs deviennent plus intéressants au niveau de la consommation en bande passante, et les protocoles réactifs sont plus performants dans des réseaux ayant un faible nombre de nœuds. En effet, ils connaissent à tout moment au moins une topologie partielle du réseau, les protocoles hybrides vont donc tirer avantage de ces deux protocoles. Un nœud va utiliser, dans son proche entourage, un algorithme de routage proactif. Ainsi, chaque nœud a une connaissance globale de son voisinage. Puis à l'extérieur de son entourage immédiat, il va utiliser un algorithme de routage réactif. ZRP (Zone Routing Protocol) est un exemple de cette catégorie.

3.3.4 Selon la topologie du réseau

La topologie détermine l'organisation des nœuds capteurs au sein du réseau, elle joue un rôle important dans le fonctionnement des protocoles de routage. On distingue deux catégories : plat et hiérarchique[2].

a. La topologie plate Un réseau de capteurs sans fil plat est un réseau homogène, où tous les nœuds sont identiques en termes de batterie et de complexité du matériel, à l'exception du nœud puits qui joue le rôle d'une passerelle et qui est responsable de la transmission des informations collectées à l'utilisateur final. La densité de capteurs suit le service et le type de capteurs, ainsi une communication multi-saut peut être nécessaire. En présence d'un très grand nombre de nœuds capteurs, la scalabilité devient critique, le routage et le contrôle d'accès au médium (MAC) doivent gérer et organiser les nœuds d'une manière très efficace en termes d'énergie[68].

Le réseau plat adopte des algorithmes et protocoles plus simples[14] qui permettent une importante tolérance aux pannes et une grande possibilité de construire des routes au cours du temps. Vu que la collecte des données uniquement à la charge du nœud puits, les nœuds voisins de celui-ci vont être sollicités plus que les autres pour le routage. De ce fait, leurs batteries vont rapidement être épuisées [7], un autre inconvénient est que le passage à l'échelle est considérablement faible. Dans le cas où les nœuds opèrent de la même façon et d'une

manière distribuée, l'emploi d'un grand nombre de messages de contrôle est nécessaire afin d'aboutir à un bon fonctionnement du protocole [76].

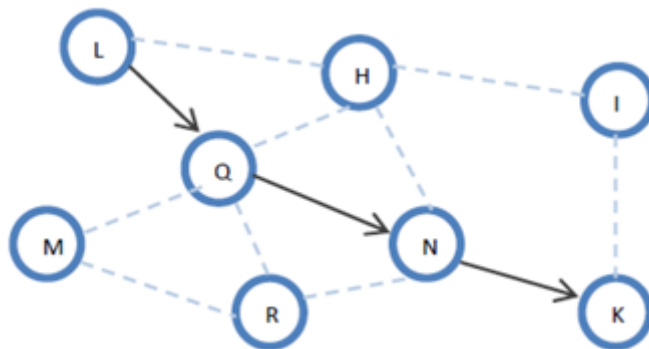


FIGURE 3.2 – Topologie plate

b. La topologie hiérarchique Une architecture hiérarchique était proposée dans le but de réduire la consommation d'énergie et la complexité de la plus part des nœuds capteur[68]. Le principe est de décomposer les nœuds du réseau en plusieurs niveaux de responsabilité et d'associer des rôles différents aux nœuds, dont certains sont élus pour exécuter des fonctions particulières et plus coûteuses en termes d'énergie tel que l'agrégation de données[2, 82]. Ceci en créant une infrastructure qui décharge la majorité des nœuds simples à faible coût d'énergie, qui se charge uniquement du captage[68]. L'une des méthodes les plus employées est « Le Clustering », elle consiste à regrouper les nœuds en groupes appelés « clusters » et chaque cluster est représenté par un nœud particulier appelé cluster-head . Ainsi, définir des schémas de communication à l'intérieur des clusters (communication intra-cluster) et entre les clusters (communication inter-cluster). Un nœud est élu comme chef selon une métrique spécifique ou bien une combinaison de métriques telles que : l'identifiant, le degré, l'énergie, etc. Ce dernier se charge de la coordination entre les différents nœuds membres de son cluster pour agréger leurs données collectées et les transmettre vers la SB. Chaque nœud membre d'un cluster garde des informations sur son cluster et sur d'autres clusters, ce qui minimise considérablement la taille des tables de routage et le nombre de messages échangés dans le réseau.[6].

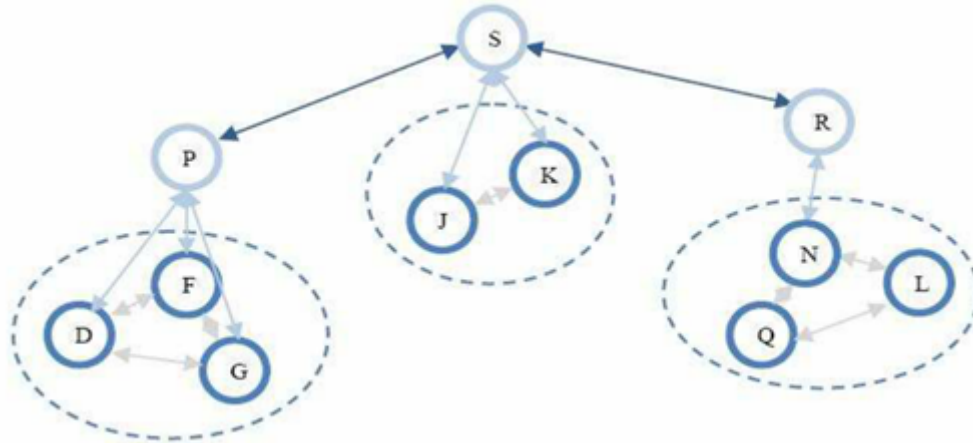


FIGURE 3.3 – Le Clustering

Si le nœud D génère des paquets de données qui veulent atteindre le nœud N, il faut passer par les Cluster-Head (passerelles) : P, S, R.

Cette topologie présente beaucoup d'avantages tels que la scalabilité et l'agrégation des données. Cependant, son inconvénient majeur est la surcharge des leaders qui induit un déséquilibre de la consommation d'énergie, pour remédier à ce problème deux solutions sont proposées. L'une consiste à ce que les cluster-heads soient des capteurs spécifiques avec plus de ressources en termes d'énergie et puissance de traitement, l'autre consiste à sélectionner dynamiquement les têtes et ainsi garantir un équilibre de la consommation d'énergie et augmenter la tolérance aux pannes[59].

Dans les sections suivantes, nous allons présenter quelques protocoles de ces deux structures afin de mieux voir le fonctionnement des réseaux plat et hiérarchiques.

3.4 Routage plat

Nous présentons quelques protocoles qui sont fondés sur la structure plate :

3.4.1 Directed Diffusion (DD)

[rapport final] Ce protocole de routage est basé sur une approche centrée-donnée, où les nœuds intermédiaires peuvent agréger des données et les envoyer vers le nœud puits en utilisant plusieurs chemins [56]. Au départ, le nœud puits commence à diffuser périodiquement un message d'intérêt à tous ses voisins, et qui à leur tour le diffuse jusqu'à atteindre la source.

Ce message spécifie les données par lesquelles l'utilisateur est intéressé comme la sous-région de captage, débit de réponse, durée de l'intérêt etc. chaque nœud recevant l'intérêt, maintient un cache pour garder traces des intérêts reçus et les champs de gradients qui lui

sont associés. Sachant qu'un gradient est un lien de réponse des voisins recevant l'intérêt. Dès que le gradient est expiré l'intérêt est supprimé du cache. Cette technique permet d'établir plusieurs chemins entre le puits et la source.

Quand la source capte un événement, cherche dans son cache l'intérêt correspond aux données captées s'il existe, envoie la donnée captée à tous ses voisins à travers ses gradients. Sinon la données sera écartée, le même mécanisme sera exécuté au niveau de tous les nœuds capteurs jusqu'au puits. Ce dernier choisit l'un de ses voisin (selon certain critères) et lui envoie un message de renforcement positif -une règle qui consiste à choisir le premier voisin émetteur de la donnée correspondante à l'intérêt et un chemin faible latence est établis entre la source et le puits- et ce nœud intermédiaires va renforcer son gradient avec l'émetteur et diffuse a son tour, le message de renforcement à ses voisins et ainsi de suite. La figure ci-dessous illustre les phases de communication du DD[64].

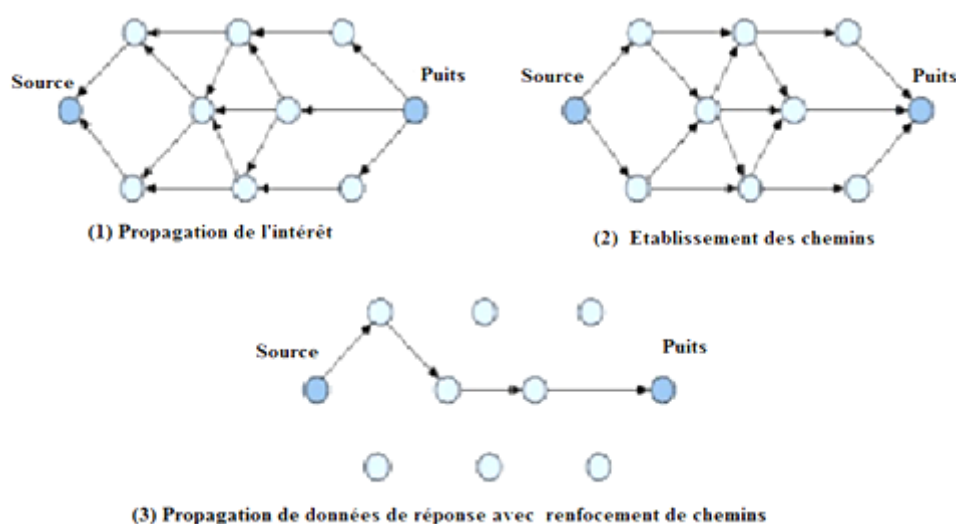


FIGURE 3.4 – Les phases de communication du protocole DD

3.4.2 Gradient-Based Routing (GBR)

Routage basé sur gradient est une version légèrement modifiée de Directed Diffusion. L'idée est de garder le nombre de sauts lorsque l'intérêt est diffusé à travers le réseau. Par conséquent, chaque nœud peut découvrir le nombre minimum de sauts sur le puits, appelé hauteur du nœud. La différence entre la hauteur d'un nœud et celle de son voisin est considérée comme le dégradé sur ce lien. Un paquet est transmis sur un lien avec le plus grand dégradé. Les auteurs visent à utiliser des techniques auxiliaires telles que l'agrégation des données et la circulation se répandant avec GBR afin d'équilibrer uniformément le trafic sur le réseau. Les nœuds servant de relais pour plusieurs chemins, peuvent créer une entité de combinaison de données pour l'agrégation, pour cela trois techniques de diffusion différentes ont été présentées :

- *Schéma stochastique* : lorsqu'il y a deux ou plus de sauts suivants avec le même dégradé, le nœud choisit l'un d'eux au hasard.
- *Schéma énergétique* : lorsque l'énergie d'un nœud descend en dessous d'un certain seuil, augmente sa hauteur afin que d'autres capteurs soient découragés d'envoyer des données à ce nœud.
- *Schéma basé sur les flux* : l'idée est de détourner de nouveaux flux loin des nœuds qui sont fait actuellement partie de la voie d'autres flux.

Les techniques utilisées pour l'équilibrage de la charge du trafic et la fusion des données sont également applicables à d'autres protocoles de routage pour des performances améliorées[13].

3.4.3 Energy Aware Routing (EAR)

Une solution hybride pour la tolérance aux pannes est proposée dans le protocole EAR. Pour son concept préventif, EAR offre une meilleure conservation d'énergie et définit plusieurs chemins de routage afin de garantir une fiabilité du transport et d'augmenter la durée de vie du réseau. En outre, un mécanisme de recouvrement de pannes est implémenté. Le protocole EAR supporte des réseaux de capteurs à collecteurs multiples (plusieurs puits). Chaque nœud capteur génère un paquet RPT (Report) contenant des informations pour les intérêts (requêtes) et préférences de l'utilisateur. Ces paquets peuvent être envoyés vers n'importe quel collecteur. Cependant, pour chaque nœud intermédiaire le protocole de routage choisit le meilleur chemin qui réduit la consommation d'énergie et la latence. Le fonctionnement du protocole EAR passe par trois phases suivantes :

a. Phase d'initialisation : Cette phase permet la construction de l'arbre de routage contenant tous les chemins possibles pour la dissémination des données. Chaque nœud puits diffuse un message ADV (advertisement) demandant des paquets RPT. Seuls les nœuds voisins du puits reçoivent ce message puis enregistrent le chemin dans leur table de routage, sans le propagé. Les autres nœuds cherchant un chemin vers le puits envoient une demande RREQ (Route Request). Les nœuds ayant déjà une route stockée dans leurs tables, envoient un paquet RREP (Route Reply) à leurs voisins concernés par la demande. Le processus d'initialisation se termine quand chaque nœud reçoit une réponse RREP suite à sa requête RREQ, puis enregistre le chemin dans sa table de routage.

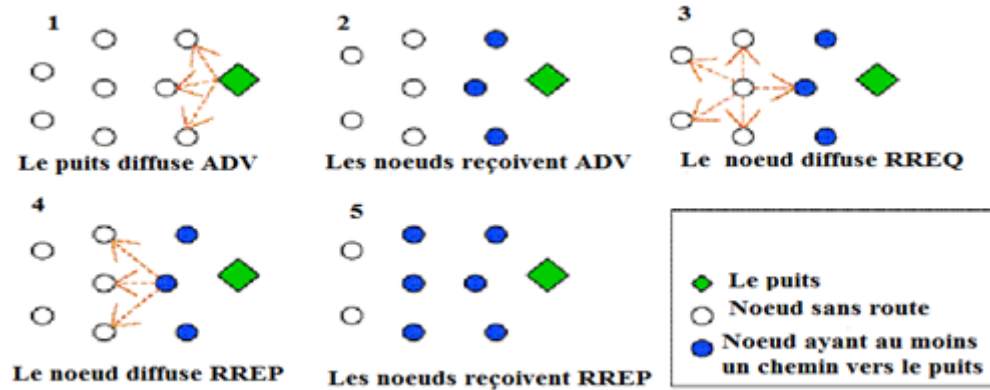


FIGURE 3.5 – phase d’initialisation du EAR

b. Phase de Gestion des routes Les micro-capteurs, ne peuvent garder tous les chemins possibles dans leurs tables de routage. Pour cela, sauf les meilleurs chemins sont gardés. Le protocole EAR définit donc deux métriques pour la sélection. La première métrique est le nombre de sauts dans une route, ce qui permet de choisir le chemin le plus court sans prendre en considération la qualité des liens ; dans ce cas, le plus court chemin n’assure pas forcément la fiabilité de transmission. D’où on utilise le concept de « routes bannies ». En effet, si un chemin échoue à transmettre plusieurs paquets consécutifs, il sera mis dans une « liste noire » l’écartant ainsi d’une future utilisation. La deuxième métrique, appelée Score de route, est définie selon une fonction, dans le but de sélectionner les meilleurs chemins en termes de fiabilité de liens et de conservation d’énergie. Ceci garantit une bonne tolérance aux pannes en évitant la sélection des mauvaises routes.

C. Phase de dissémination de données : Les capteurs commencent à générer des paquets RPT, et le routage des données utilise la métrique « score de route » pour définir le meilleur chemin à emprunter. En cas de panne d’un ou plusieurs nœuds, un mécanisme de recouvrement de route est exécuté, afin d’élire un second chemin fiable pour transmettre les données. Par ailleurs, au moment de sa période d’inactivité chaque capteur est mis en veille afin d’épargner d’avantage la perte d’énergie et augmenter ainsi la durée de vie de tout le réseau[59].

3.4.4 SAR (Sequential Assignment Routing)

SAR ou routage d’affectation séquentielle. SAR crée des multiples arbres à partir de la racine où les racines sont les voisins de la station de base. Ensuite, l’arbre est généré vers l’extérieur dans le réseau en évitant les nœuds à faible débit et à long délai. A la fin de la procédure, les nœuds appartenant à plusieurs arbres au même temps, choisissent un arbre pour envoyer les données à la SB à travers une racine.

Le protocole SAR est un protocole de routage multivoies qui prend la décision de routage à base de trois facteurs : ressources énergétiques, QoS sur chaque chemin et niveau de priorité de paquets. Parmi les chemins complets, le meilleur chemin choisit pour l'envoi des données, fournit un meilleur délai et un débit élevé. Deux paramètres sont attribués à chaque nœud de capteur sur chaque chemin à travers celui-ci :

- Les ressources énergétiques disponibles sur le chemin.
- Qualité additive de métrique de service[63].

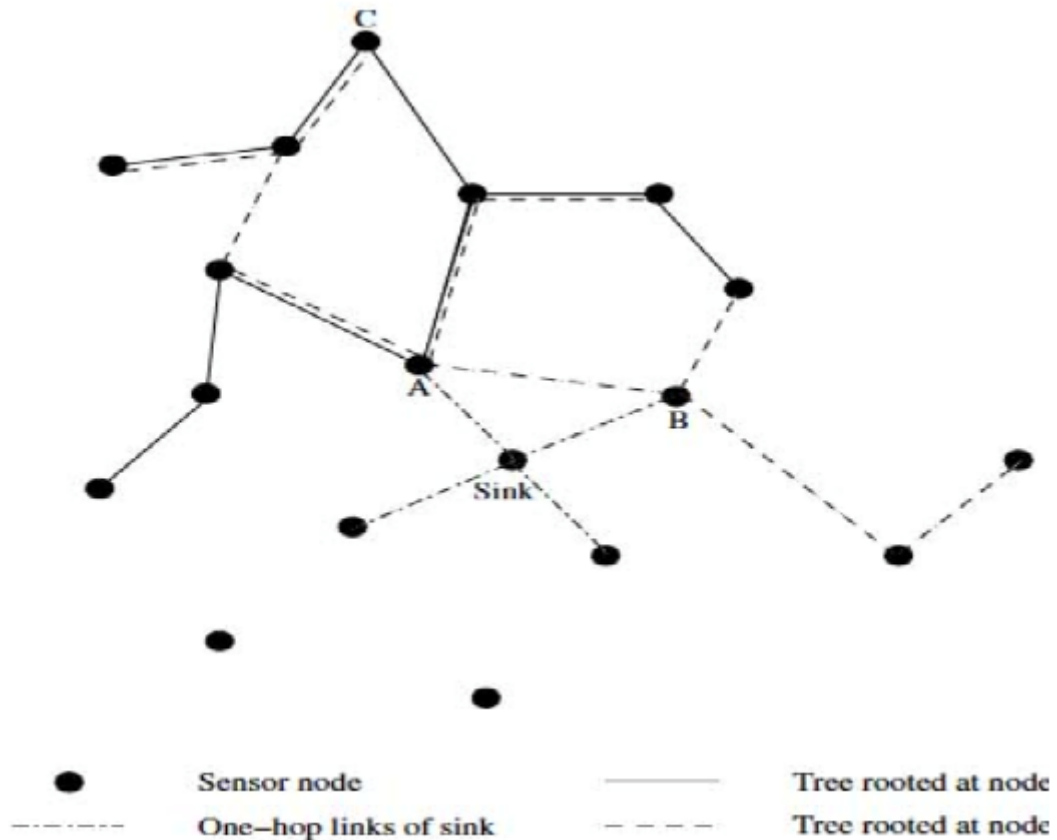


FIGURE 3.6 – Fonctionnement du protocole SAR

3.4.5 SPIN (Sensor Protocols for Information via Negotiation)

SPIN est un protocole data-centric basé sur la négociation. Il a été développé afin de surmonter les inconvénients des inondations de l'information dans les RCSFs qui épuise rapidement la durée de vie des batteries. Pour cela, SPIN utilise des métadonnées avant d'envoyer le message complet aux nœuds voisins. Donc seuls les nœuds intéressés par ce message qui vont le recevoir. Pour assurer cette communication SPIN utilise trois types de messages à savoir : ADV(DVesrtise), REQ (REQuest), DATA.

Le nœud envoie d'abord la métadonnée en utilisant un message ADV vers tous les nœuds voisins. A la réception de ce dernier le nœud vérifie l'existence d'un intérêt correspondant à la donnée en question en consultant son cache. S'il est intéressé il répond l'émetteur par un

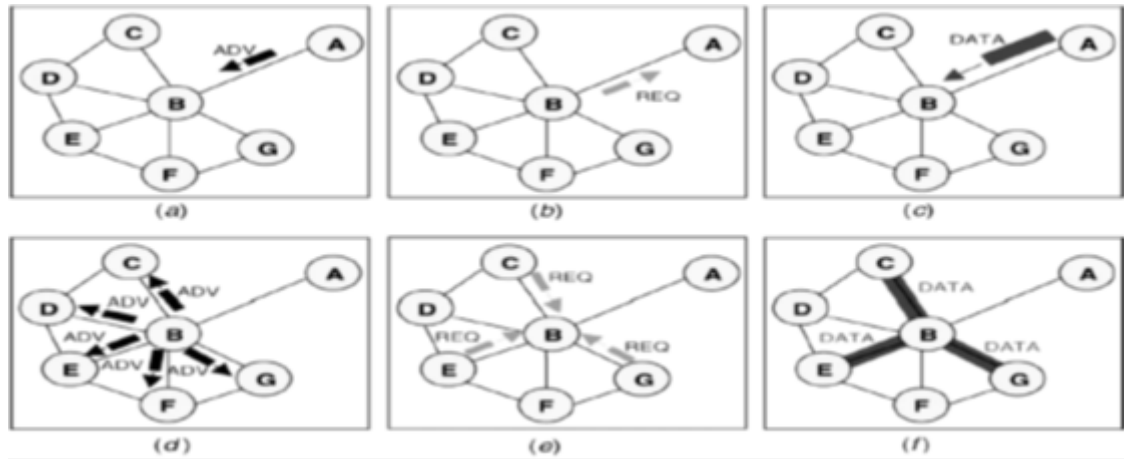


FIGURE 3.7 – Fonctionnement du protocole SPIN

message REQ (accord de réception de la donnée). A son tour, l'émetteur lui envoie le message DATA (message complet). Le même mécanisme est exécuté pour chaque transmission.[42, 47].

Remarque : Dans les architectures plates, la majorité des protocoles de routage conçus pour les réseaux de capteurs de petite ou moyenne taille avec une faible mobilité de nœuds fournissent de bonnes performances. Cependant, lorsque le nombre de nœuds augmente ou que les nœuds sont mobiles, le trafic de contrôle prédomine les communications réelles. Cela conduit à une augmentation de la latence et à une explosion des tables de routage. Afin de pallier à ces limites, l'architecture hiérarchique est considérée comme l'une des solutions communément efficaces pour le routage dans les réseaux de capteurs.

3.5 Routage hiérarchique

Le routage hiérarchique est considéré comme étant l'approche la plus favorable en terme d'efficacité énergétique, deux grandes approches sont dérivées de ce type de protocoles à savoir :

Chaîne-based approach (approche chaînée) qui consiste à former des chaînes à partir du nœud plus éloigné et acheminer les données du nœud en nœud et choisir un seul nœud de cette chaîne qui va les transmettre vers la station de base.

Cluster-based approach (approche à cluster). qui consiste à former des clusters et à élire des chefs qui assurent l'agregation et l'acheminement des données collectées vers la station de base.

Voici quelques exemples de protocoles illustrant ces deux approches :

3.5.1 MECN (Minimum Energy Communication Network) :

Minimum Energy Communication Network [75] est un protocole de routage qui cherche à établir et à entretenir une énergie minimale pour les réseaux sans fil en utilisant des GPS de faible puissance. MECN utilise une station de base comme destination de l'information, ce qui est toujours le cas pour les réseaux de capteurs.

MECN identifie une région de relais pour chaque nœud. La région de relais se compose de nœuds dans une zone périphérique où la transmission à travers ces nœuds est plus économe en énergie que la transmission directe. L'idée principale de MECN est de trouver un sous-réseau qui a moins de nœuds et qui nécessite moins d'énergie pour la transmission entre deux nœuds quelconques. Cela est effectué en utilisant une recherche localisée pour chaque nœud en prenant en considération sa région de relais[37].

3.5.2 LEACH (Low Energy Adaptive Clustering Hierarchy) :

La hiérarchie de regroupement d'adaptation à faible énergie ou LEACH, est considéré comme étant le premier protocole de routage hiérarchique avec efficacité énergétique basé sur le clustering, proposé pour les RCSFs. Il est aussi l'un des algorithmes de routage hiérarchique le plus populaire pour les réseaux de capteurs. L'idée est de former des clusters de nœuds, dont chacun constitue, selon la responsabilité, deux catégories de capteurs à savoir : le CH et les nœuds membres . puis utiliser des clusters-head comme passerelles pour atteindre la destination en optimisant la consommation d'énergie suivant un algorithme qui utilise la rotation randomisée (round) des CHs pour distribuer d'une manière équitable la charge d'énergie entre les nœuds du réseau. Un nœud décide quel cluster rejoindre en se basant sur la puissance des signaux reçus, S'il reçoit deux signaux de même puissance, alors il rejoint aléatoirement un des deux.

Les nœuds membres captent et transmettent des données à leur chef qui, à son tour, effectue des fonctions sur ces données collectées (agrégation et compression), puis les transmet à la station de base selon une communication unicast (à un seul saut) [79].

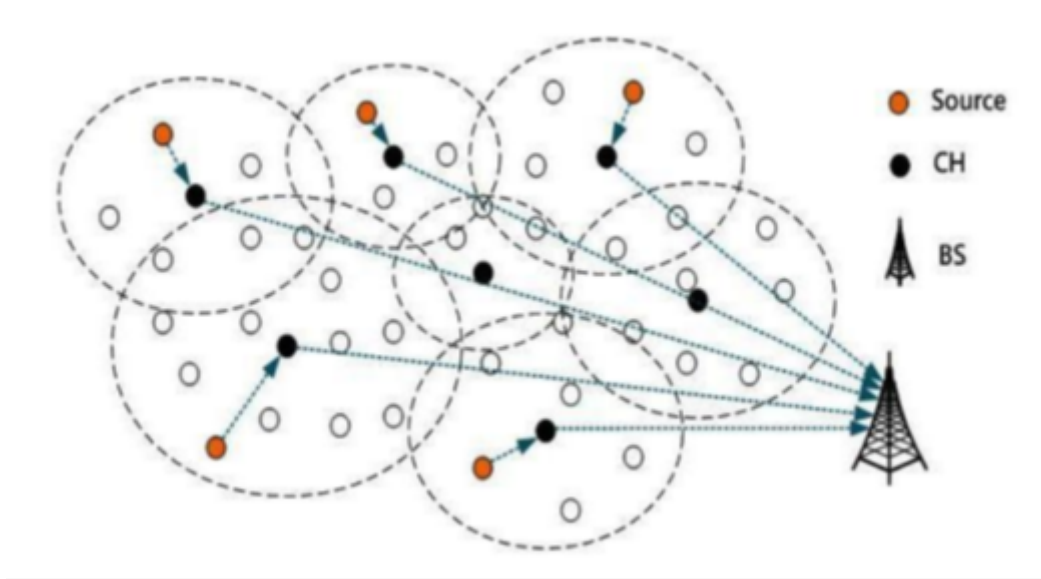


FIGURE 3.8 – Le protocole LEACH

L'algorithme se déroule en rounds (tours) qui ont presque le même cycle (intervalle) de temps déterminé au préalable. Où chaque cycle commence par une phase d'initialisation suivie d'une phase de transmission :

a. Phase d'initialisation : Cette phase est composée de trois sous-phases qui sont : annonce, organisation des groupes, ordonnancement.

— *Sous-phase d'annonce :*

Avant de lancer cette phase, on désire avoir un certain nombre de CH. Ce nombre est fixe et il est inchangé durant tous les rounds. Le pourcentage optimal du nombre de CH désirés devrait être de 5% à 15% du nombre total de nœuds [17]. Si ce pourcentage n'est pas respecté, cela mènera à une grande dissipation d'énergie dans le réseau[17].

Dès que le puits annonce un nouveau tour, les nœuds prennent décision de devenir ou pas CH selon une certaine probabilité .Chaque nœud i choisit un nombre aléatoire entre 0 et 1. Si le nombre est inférieur à un seuil $T(n)$ le nœud devient un CH pour le cycle actuel. Le seuil est définit comme suit [79] :

$$T(n) = \begin{cases} \frac{P}{1-p*(r \bmod \frac{1}{p})} & \text{si } n \in G \\ 0 & \text{sinon} \end{cases} \quad (3.1)$$

P : est le pourcentage souhaité de CHs c'est-à-dire choisi comme $p = 0,05$ pour une condition Optimale

r : est le tour courant (round)

G : est l'ensemble des nœuds qui n'ont pas été CHs lors des $(1/p)$ tours précédents.

— *Sous-phase d'organisation des groupes :*

Chaque nœud qui sera désigné comme CH pour le tour courant, diffuse un message d'initialisation vers le reste des nœuds qui contient l'ID (identifiant) du nœud et d'un entête, utilisant le protocole MAC (Media Access Control) CSMA (Carrier Sense Multiple Access). Les autres nœuds rejoignent le CH le plus proche. Après le choix du CH à rejoindre, ce dernier doit être informé de l'ensemble des nœuds composant son groupe. Chaque nœud non-CH transmet une requête vers le CH choisi en utilisant également un protocole CSMA MAC. Ce message ou requête se compose de l'ID du nœud, de l'ID du CH et d'un entête[79].

— *Sous-phase d'ordonnancement* :

Une fois que les groupes sont formés, chaque CH coordonne les transmissions des données au sein de son groupe. Il crée un ordonnanceur (scheduler) TDMA et assigne à chaque nœud membre un slot de temps durant lequel il peut transmettre ses données. L'ensemble des slots assignés aux nœuds d'un groupe est appelé frame. La durée de chaque frame diffère selon le nombre de membres du groupe. Pour éviter les interférences des communications, chaque CH choisit aléatoirement un code dans une liste de codes de propagation CDMA. Il le transmet par la suite à ses membres afin de l'utiliser pour leurs transmissions[61].

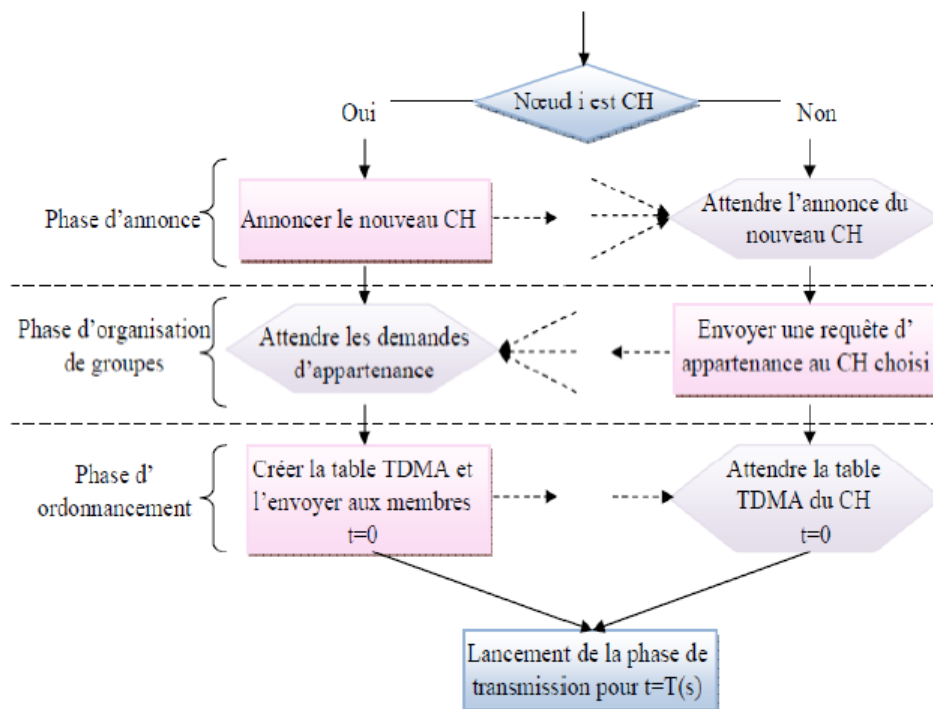


FIGURE 3.9 – Les opérations de la phase d'initialisation

Phase de communication (transmission) : Cette phase est plus longue que la phase précédente, et permet la collecte de données captées. En utilisant l'ordonnanceur TDMA, les membres émettent leurs données captées pendant leurs propres slots. Cela leur permet d'éteindre leurs interfaces de communication en dehors de leurs slots afin d'économiser leurs

énergie. Ces données sont ensuite agrégées par les CHs qui les fusionnent et les compressent, et envoient le résultat final au nœud puits.

Après un certain temps prédéterminé, le réseau va passer à un nouveau round. Ce processus est répété jusqu'à ce que tous les nœuds du réseau soient élus comme CH, une seule fois, tout au long des rounds précédents. Dans ce cas, le round est réinitialisé à 0[84].

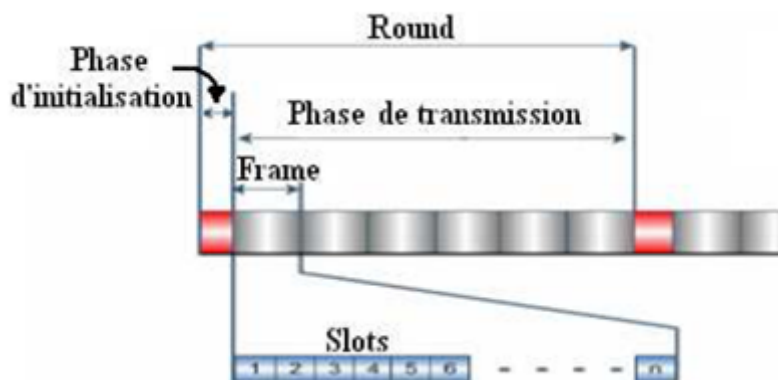


FIGURE 3.10 – Les phases d'un cycle du protocole LEACH

Inconvénients du protocole LEACH On peut cerner quelques inconvénients du protocole LEACH, qui sont les suivants[2] :

- On pourra ne pas avoir des CH durant un round si les nombres aléatoires générés par tous les nœuds du réseau sont supérieurs à la probabilité $T(n)$
- Les nœuds les plus éloignés du CH meurent rapidement par rapport aux plus proches.
- L'utilisation d'une communication à un seul saut au lieu d'une communication multi-sauts diminue l'énergie des nœuds.
- Le protocole LEACH ne peut pas être appliqué à des applications temps-réel du fait qu'il résulte en une longue latence.
- La rotation des CH permet de ne pas épuiser les batteries. Cependant, cette méthode n'est pas efficace pour de grandes structures de réseaux à cause de la surcharge d'annonces engendrées par le changement des CH, et qui réduit le gain d'énergie initial.
- Il n'est pas évident que les CH soient uniformément distribués. Donc, il est possible que les CH puissent être concentrés dans une partie du réseau. Par conséquent, certains nœuds n'auront pas des CH dans leurs voisinages.
- Le protocole LEACH n'est pas sécurisé. Aucun mécanisme de sécurité n'est intégré dans ce protocole. Ainsi, il est très vulnérable même aux simples attaques. Donc, un attaquant peut facilement monopoliser le réseau et induit à son dysfonctionnement.

3.5.3 TEEN (Threshold sensitive Energy Efficient sensor Network protocol)

TEEN ou protocole à Efficacité énergétique sensible au seuil. Conçu pour répondre aux changements inattendus des valeurs détectées tel que la température. La réactivité est urgente pour les applications temps réel. L'architecture du réseau de capteur dépend d'un groupement hiérarchique où les nœuds proches forment des clusters. Le cluster Head diffuse deux seuils vers les nœuds, qui sont le seuil dur et le seuil doux des valeurs détectés.

Le seuil dur c'est la valeur minimale possible détectée qui permet d'activer un nœud capteur pour pouvoir, par la suite, transmettre des données au CH, Ce qui réduit considérablement le nombre de transmissions.

Le seuil doux c'est le degré de changement de la valeur détectée. Un nœud envoie des données uniquement lorsque la valeur détectée est supérieure ou égale au seuil doux. En conséquence, le seuil doux réduira encore le nombre de transmissions s'il y a peu (ou pas) de changement de la valeur détecté. On peut adapter à la fois les valeurs des seuils dur et doux afin de contrôler le nombre de transmissions.

TEEN n'est pas efficace pour les applications qui nécessitent des rapports réguliers car l'utilisateur ne peut recevoir aucune donnée du tout si les seuils ne sont pas atteints[37].

3.5.4 APTEEN (Adaptive Periodic TEEN)

Une nouvelle approche de TEEN qui se base sur la collection périodique des données et la réaction aux événements temps réels. La structure est semblable à TEEN. Dans APTEEN, les Cluster-Head diffuse des paramètres comme la période de comptage, les attributs, et les seuils à tous les nœuds. La performance d'APTEEN se situe entre LEACH et TEEN en termes de durée de vie du réseau et de consommation d'énergie. Il fait une amélioration par rapport à TEEN car il utilise un rapport régulier pour les événements chronologiques. Le principal inconvénient est la surcharge et la complexité de la formation des clusters de ces deux algorithmes[35].

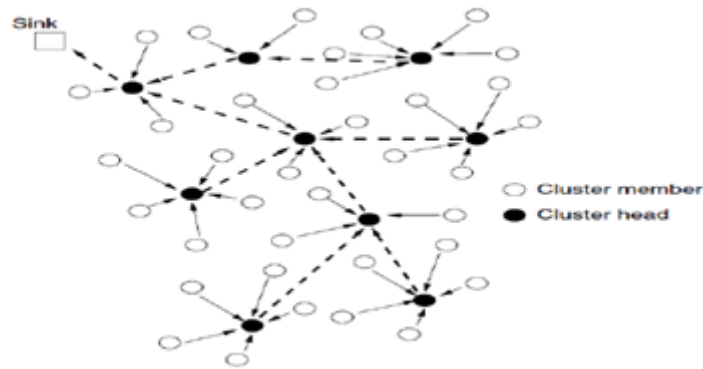


FIGURE 3.11 – Les protocoles TEEN et APTEEN

3.5.5 HEED (Hybrid, Energy-Efficient Distributed Clustering)

HEED est une version améliorée de protocole de routage LEACH, qui considère l'énergie résiduelle et le degré ou la densité du nœud (nombre de nœuds voisins) en tant que métrique pour la sélection des CHs afin d'obtenir un équilibrage de puissance. Il opère dans Multi-hop, en utilisant une puissance d'émission adaptative dans la communication inter-Clustering.

Dans HEED, l'algorithme proposé sélectionne périodiquement CHs Selon une combinaison de deux paramètres de clustering :

Le paramètre principal est l'énergie résiduelle (ER) de chaque nœud capteur est utilisée pour calculer la probabilité de devenir un CH, et le paramètre secondaire est la communication intra-cluster, le coût en fonction de la densité du cluster ou du degré de nœud.

HEED améliore la durée de vie du réseau mieux que LEACH, car ce dernier sélectionne au hasard les CHs, ce qui peut entraîner une mort plus rapide de certains nœuds. Par contre, Les CHs finaux sélectionnés dans HEED sont bien répartis à travers le réseau et les coûts de la communication sont minimisés.

Cependant, La sélection de cluster n'est qu'une méthodes qui conviennent pour prolonger la durée de vie du réseau plutôt que pour tous les besoins des RCSFs[8, 52].

3.5.6 UCA (Unéqual Clustering Algorithm)

Dans cette approche, les CHs coopèrent entre eux pour transmettre leurs données à la SB (communication multi-sauts), les clusters qui se rapprochent de la SB sont chargés d'un trafic de relais plus lourd et ont tendance à mourir tôt, en laissant les zones du réseau découvertes et en provoquant le partitionnement du réseau. Pour atténuer le problème des points chauds, un algorithme de clustering inégalé (UCA) a été proposé. Il permet un regroupement des nœuds en clusters de tailles inégales et la sélection périodique des CHs à base de la concurrence.

Les CHs plus proches de la SB gèrent des clusters de tailles plus petites (consomme moins d'énergie durant la communication intra-cluster). Tantqu'on s'éloigne de la SB les CHs gèrent des clusters de tailles, plus en plus, grandes par rapport à ceux qui se rapprochent de la SB, ils peuvent ainsi conserver une certaine énergie pour la communication inter-cluster.

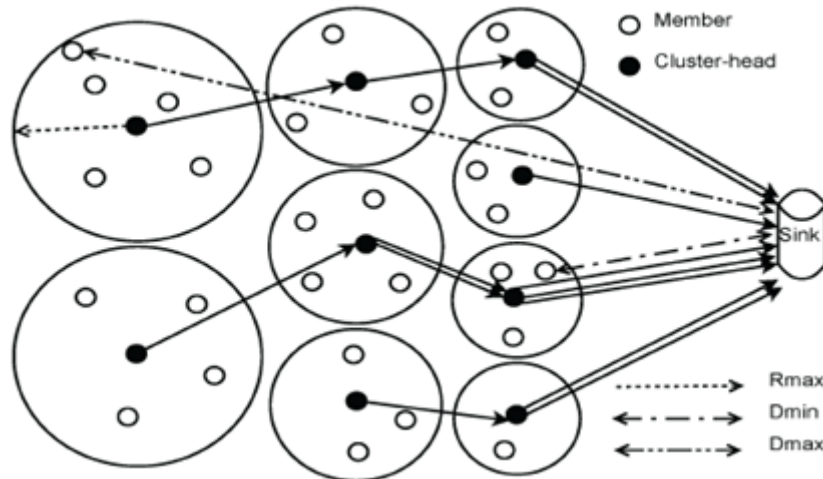


FIGURE 3.12 – Le protocole UCA

Pendant la phase d'élection des clusters, les nœuds échangent leurs énergie résiduelle en utilisant un rayon de concurrence appropriée. Un nœud qui a une grande quantité d'énergie décide pour devenir le cluster head pour le cycle actuel. Ce mécanisme évite d'élire plus d'un cluster-head sur la même région de compétition. Toutefois, le nombre d'échanges de messages n'est pas négligeable. En effet, la phase électorale des chefs de cluster consiste de deux diffusions locales. Dans la première diffusion, chaque nœud, localement diffuse son enchère (c.-à-d., la concurrence des nœuds pour devenir cluster-head). Dans la deuxième diffusion, les nœuds qui sont élus partagent leurs statuts en utilisant la gamme de rayon de concurrence correspondante.

UCA atténue le problème du point chaud (surcharge du trafic au niveau des CHs plus proches de SB), permet une meilleure répartition des CHs sur le champ du réseau, et donc une amélioration évidente de la durée de vie du réseau. Or, la sélection périodique des CHs d'une manière concurrente avec des tailles inégales, induit à une densité des CHs dans les zones les plus proches de la station de base[24].

3.5.7 PEGASIS (Power-Efficient Gathering in Sensor Information Systems)

PEGASIS est une extension du protocole LEACH, qui forme des chaînes à partir de nœuds de capteurs, de sorte que chaque nœud transmet et reçoit d'un voisin et un seul nœud est

choisi parmi cette chaîne pour transmettre à la station de base. Contrairement à LEACH, les données sont collectées, transférées de nœud en nœud, agrégées, et finalement transmises à la SB, au lieu de les transmettre directement aux Cluster-head.

La création de la chaîne est exécutée d'une manière gourmande. Contrairement à LEACH, PEGASIS reste éloigné de la formation des clusters. Supposant, qu' à la création des chaînes, que tous les capteurs possèdent des informations globales sur le réseau, notamment l'emplacement des capteurs, lorsqu'un capteur échoue ou meurt en raison de la faible puissance énergétique [63], la chaîne est construite en utilisant la même approche gourmande en évitant le capteur défaillant. A chaque tour, un nœud capteur est sélectionné de la chaîne pour transférer les données agrégées à la SB, ce qui diminue la consommation d'énergie par cycle par rapport à LEACH. PEGASIS permet d'augmenter la durée de vie[63] du réseau deux fois plus que LEACH. Un tel gain de performance est obtenu par l'élimination de la charge causée par la formation dynamique des clusters dans LEACH, et par l'utilisation de l'agrégation des données.

PEGASIS a toujours besoin d'une adaptation de topologie dynamique, car un nœud de capteur doit connaître l'état de l'énergie de ses voisins pour savoir où acheminer ses données.

Une telle adaptation de topologie peut introduire une charge importante principalement pour les réseaux hautement utilisés.

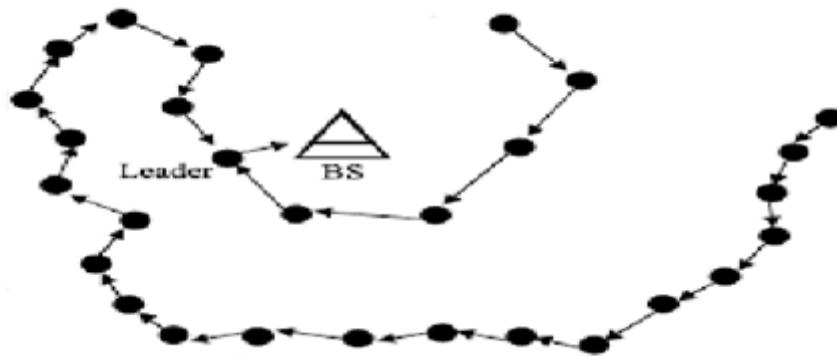


FIGURE 3.13 – Le protocole PEGASIS

3.5.8 HEERP (Hierarchical Energy Efficient Routing Protocol) :

HEERP ou protocole de routage hiérarchique efficace en énergie, établit une nouvelle approche centralisée à la formation de l'hierarchie. HEERP inclut la hiérarchie du réseau, la construction des tables voisines, et la transmission des données. Le puits commence la formation de la hiérarchie en diffusant le paquet LCREQ, les nœuds choisissent ces paquets à partir des nœuds dont le nombre de sauts est moindre. Par conséquent, l'inondation des paquets est évitée jusqu'à la construction temporelle du réseau. Dans la phase de transmission,

chaque nœud transmet des données à ses nœuds parents. HEERP utilise moins d'énergie, et améliore aussi le délai de réponse par rapport au protocole LEACH [54].

3.5.9 DEEC (Distributed Energy Efficient Clustering Algorithm for Heterogeneous Wireless Sensor Networks)

Un algorithme de cluster multi-niveaux distribué avancé pour les RCSFs hétérogènes. Dans DEEC, les Cluster-Head sont choisis selon une probabilité qui dépend du rapport entre l'énergie résiduelle de chaque nœud et l'énergie moyenne du réseau. L'approche d'être un Cluster-Head pour les nœuds est différente selon leur énergie initiale et résiduelle.

Les auteurs ont supposé que tous les nœuds du réseau de capteurs sont dotés d'une quantité d'énergie différente. Deux niveaux de nœuds hétérogènes sont examinés dans l'algorithme et, après cela, on obtient un résultat commun pour une hétérogénéité à plusieurs niveaux. Pour éviter que chaque nœud nécessite de connaître l'information globale des réseaux, DEEC s'approche de la valeur idéale de la durée de vie du réseau, qui est utilisée pour évaluer l'énergie de référence que chaque nœud devrait dépenser pendant le routage d'un paquet. Le calcul du cluster est variable dans cette approche conduisant à des clusters inégaux[35].

3.5.10 DEEC-LCH (DEEC- linear cluster handling)

Ce protocole, utilise la technique de gestion linéaire de clusters, où l'ensemble du réseau est divisé en différentes sous-régions égales. Pour une collecte de données efficace, on place trois SBs statiques, c'est-à-dire deux SB placées dans les deux extrémités, et une au centre du champ. Les opérations de protocole sont divisées en trois phases différentes à savoir :

- Phase de publicité
- Phase de configuration de cluster
- Phase de transmission de données.

Dans ce protocole, la sélection des CHs est basée sur le pourcentage recommandé de CH pour le réseau et le nombre de fois où le nœud a été CH jusqu'à présent. Un protocole de routage proactif et le regroupement efficace en énergie distribuée avec gestion de cluster linéaire (DEEC-LCH) [4] est implémenté dans ce réseau. En comparant DEEC-LCH avec DEEC, la durée de vie des réseaux de cette approche proposée est augmentée, car ces nœuds conservent leur énergie en raison de la formation de la sous-région et du placement de puits multiples[51].

3.5.11 EUCA (Enhanced Unequal Clustering Algorithm) :

EUCA est une amélioration d'UCA. Cette solution réduit le trafic de contrôle lors d'un processus de clustering en éliminant la concurrence des nœuds pour devenir cluster-head et permet une meilleure réactivité à l'événement et conserve plus d'énergie. Le puits lance le premier tour par diffusion d'un message « NEW-ROUND-MSG » qui contient Les paramètres d_{min} (la distance entre le puits et le nœud le plus proche) et d_{max} (la distance entre le puits et le nœud le plus éloigné). Chaque nœud i recevant ce message, déclenche la décrémentation de son Timer $T(i)$:

$$T(i) = A * T_{maxDelay} \quad (3.2)$$

Où : $T_{maxDelay}$: est le délai maximum tolérable à partir duquel le timer se décrémente.

A : est un nombre aléatoire appartenant à l'intervalle $[0, 1]$.

Si le nœud i reçoit un message de notification « CH-NOTIFY-MSG » avant l'expiration de son timer $T(i)$ alors le nœud éteint son timer $T(i)$ et attend un éventuel message de notification , sinon le nœud i élit lui-même comme cluster-Head et notifie ses voisins avec « CH-NOTIFY-MSG ». Chaque nœud i non élu choisit son cluster-Head selon la puissance du signal reçu en envoyant un message « JOIN-CLUSTER-MSG » vers le plus proche CH, sinon le nœud i attend de recevoir les messages « JOIN-CLUSTER-MSG ». La transmission de données est similaire à LEACH.

Pour les tours qui suivent, EUCA effectue la même chose Sauf que le timer $T(i)$ pour un nœud i est met en place pour être inversement proportionnel à l'énergie résiduelle :

$$T(i) = T_{maxDelay} * (1 - E_i/E_0) \quad (3.3)$$

Où :

E_i : est l'énergie résiduelle d'un nœud i

E_0 : est L'énergie initiale des nœuds déployés.

Cette approche permet une meilleure distribution des clusters sur le réseau, et une densité des CHs dans des zones qui se rapprochent de la SB. Par conséquent, à un moment donné, lors de l'élection des CHs on peut avoir des nœuds dans une même portée radio qui ont la même énergie résiduelle (même Timer $T(i)$). Ce qui induit à des collisions des données transmit[62].

3.6 Synthèse

	Livraison	Mobilité	Négociation	Agrégation	Echel	Multi chemin
LEACH	CH	SB fixe	Non	Oui	Bonne	Non
TEEN/ APTEEN	Seuil actif	SB fixe	Non	Oui	Bonne	Non
PEGASIS	Chaine	SB fixe	Non	Non	Bonne	Non
DEEC/DEECLCH	CH	SB fixe	Non	Oui	Bonne	Non
HEERP	Chaine	SB fixe	Non	Non	Bonne	Non
MECN	CH	Non	Non	Non	Faible	Non
HEED	CH	SB fixe	Non	Oui	Bonne	Non
UCA	CH	SB fixe	Non	Oui	Bonne	Non
EUCA	CH	SB fixe	Non	Oui	Bonne	Non

TABLE 3.1 – Comparaison des protocoles hiérarchique

3.7 Conclusion

Le routage dans les réseaux de capteurs sans fil est un nouveau domaine de recherche, avec un ensemble de résultats limités mais en croissance rapide. Toute conception d'un protocole pour les RCSFs doit prendre en considération les caractéristiques de ce réseau et des exigences des applications pour lesquelles ces réseaux sont conçus. Le défi principal de cette conception est l'efficacité énergétique due aux ressources limitées des capteurs, la consommation d'énergie des nœuds est dominée par la transmission et la réception des données. Ces protocoles devraient être plus efficace que possible afin d'augmenter la durée de vie du réseau.

Dans ce chapitre, plusieurs protocoles de routage sont proposés pour les RCSFs, dont quelque uns basés sur la hiérarchie sont discutés selon la topologie du réseau. Les techniques hiérarchiques ont un avantage particulier qui est l'évolutivité et l'efficacité de communication. Le routage hiérarchique maintient la consommation d'énergie des nœuds capteurs et effectue l'agrégation des données, ce qui permet de diminuer le nombre de messages transmis vers la station de base. Parmi ces protocoles, on distingue LEACH qui est particulièrement intéressant pour les RCSF. En effet, il constitue un standard sur lequel est basée la conception de plusieurs protocoles de routage tel que UCA et EUCA. De nombreux problèmes et défis existent encore qui doivent être résolus.

Le chapitre suivant sera consacré pour une contribution afin d'améliorer le protocole EUCA.

4 Free Collision based Energy and Time Clustering Algorithm

4.1 Motivation

Le problème majeur des réseaux de capteur sans fil ; est l'énergie, vu que les capteurs sont dotés d'une batterie non chargeable et qu'ils sont déployés dans des zones à accès difficile, de ce fait plusieurs protocoles de routages ont été conçus pour ce type de réseau, permettant ainsi la réduction du nombre de messages transmis et une bonne organisation du réseau (le clustering), ce qui a réduit la consommation d'énergie, assurant ainsi une longue vie du réseau.

Par ailleurs le protocole de routage hiérarchique EUCA étudié dans le chapitre précédent vise à élire le nœud ayant un niveau énergétique plus élevé comme CH, le problème s'est posé lors d'égalité des niveaux énergétiques des nœuds appartenant au même rayon de transmission, où des collisions peuvent se produire. Pour y remédier un nouveau protocole de routage le « *Free Collision based Energy and Time Clustering Algorithm* », est proposé permettant de réduire au maximum les collisions et d'élire le nœud de niveau énergétique élevé comme CH.

4.2 Considérations

Avant de décrire le fonctionnement du protocole proposé nous allons en premier lieu présenter, le modèle du réseau ainsi que le modèle énergétique pris en compte lors de la réalisation de notre protocole.

4.2.1 Modèle du réseau

Pour la réalisation de notre protocole, nous avons considéré que le réseau un ensemble de capteur déployé dans zone, ainsi le réseau se présente selon les hypothèses suivantes :

- Les nœuds capteurs sont fixes.
- Les nœuds capteurs sont distribués aléatoirement dans le réseau
- Le réseau est homogène ; tous les nœuds capteurs ont la même quantité d'énergie initiale, la même capacité de stockage et de traitement sauf la station de base.
- Tous les nœuds capteurs collectent les données et servent une unique station de base.
- Les données collectées sont agrégées par les CHs puis acheminées à la station de base via un routage multi-saut.
- Les nœuds capteurs sont laissés sans surveillance, il est donc impossible de les recharger.

4.2.2 Modèle énergétique

Un nœud un capteur consomme son énergie pour réaliser essentiellement trois opérations ; l'acquisition, la communication et le traitement des données.

L'acquisition : L'énergie consommée pour effectuer cette tâche n'est pas très importante.

Le traitement : l'énergie de cette opération se divise en deux parties :

L'énergie de commutation : elle est déterminée par la tension de l'alimentation et la capacité totale commutée en exécutant un logiciel.

L'énergie de fuite : c'est l'énergie consommée quand l'unité de calcul n'effectue aucun traitement.

La communication : c'est l'opération la plus coûteuse en termes d'énergie, elle comprend l'énergie de transmission et celle de réception; Cette énergie est déterminée par la quantité de données à communiquer, la distance de transmission, et les propriétés physiques du module radio. L'émission d'un signal est caractérisée par sa puissance, quand la puissance d'émission est élevée, le signal aura une grande portée et l'énergie consommée sera plus élevée.

Pour notre travail nous allons utiliser le modèle radio de consommation d'énergie proposé par Heinzelman[78] comme l'illustre la figure ci-dessous; ainsi l'énergie consommée pour émettre un message de l bits à un récepteur à une distance de d mètres est :

$$E_{TX} = l * E_{elec} + l * \epsilon_{mp} * d^\alpha \quad (4.1)$$

Et pour recevoir un message l'énergie consommé par le récepteur est :

$$E_{Rx}(l) = E_{Rx-elec} = l * E_{elec} \quad (4.2)$$

Où :

E_{elec} correspond à l'énergie de transmission/réception électronique.

l c'est la taille du message.

d indique la distance entre l'émetteur et le récepteur.

ϵ_{mp} représente le facteur d'amplification.

d_0 c'est la distance limite pour laquelle les facteurs de transmission changent de valeur.

α prend la valeur de 2 si $d < d_0$ et la valeur de 4 sinon

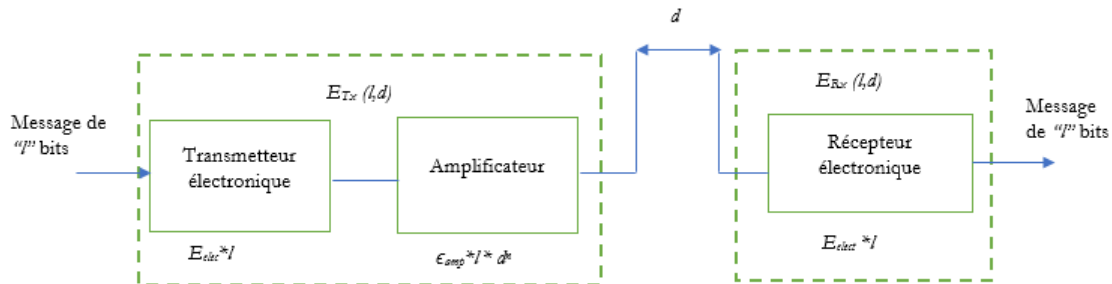


FIGURE 4.1 – Modèle de consommation d'énergie

4.3 Fonctionnement :

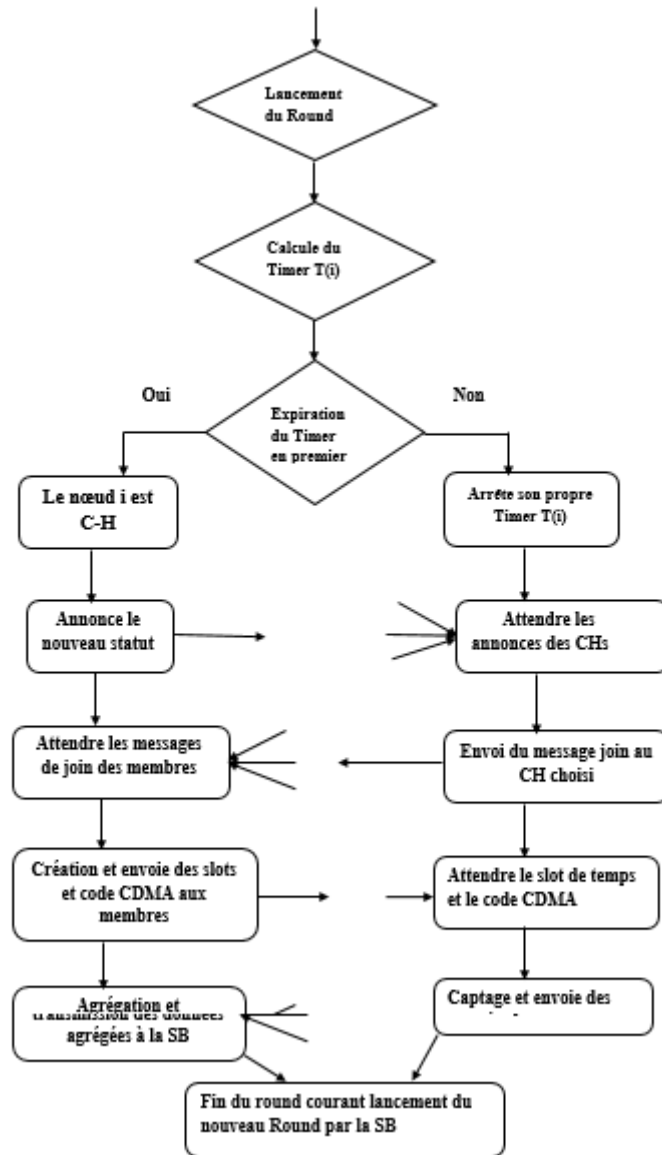


FIGURE 4.2 – L’organigramme du fonctionnement du protocole FC-ETC

Notre algorithme est basé sur une architecture hiérarchique dont le réseau est composé de clusters de nœud, où chacun constitue deux catégories de nœud, selon la responsabilité à savoir ; les nœuds membres et les CH qui servent de passerelles pour atteindre la SB, pour distribuer la charge d’énergie entre les nœuds de manière équitable et prolonger la durée de vie du réseau, les chefs sont élus à base de leurs énergies résiduelles selon une fonction randomisée. Les nœuds membres choisissent à rejoindre un CH en fonction de la force du signal reçu, en cas d’égalité le choix est aléatoire. Ensuite ils captent et transmettent leurs données à leur chef qui à son tour effectue des fonctions sur ces dernières et il l’est transmet à la SB selon une communication multi sauts.

4.3.1 Initialisation

Dans cette étape la station de base initialise le round en envoyant un message « NEW_ROUND_MSG » invitant les noeuds du réseau à participer au processus, ce message contiendra le nombre de CHs désiré sachant qu'il représente 5 à 15% de la totalité des noeuds capteurs du réseau.

4.3.2 Calcul du timer

A la réception du message d'initialisation « NEW_ROUND_MSG », annonçant le début du premier round ; chaque noeud i lance son timer après l'avoir calculer selon son énergie et λ un nombre aléatoire entre 0 et 1 tiré au début de chaque round. Chaque noeud n'ayant pas reçu de message d'annonce « ANNONCE_MSG » avant l'expiration de son timer, s'auto-élite comme CH, ensuite il informe ses voisins non CH ; autrement le noeud arrête son timer, et attend d'éventuel messages d'annonce. Le non CH choisit de rejoindre le leader ayant le plus fort signal, en cas d'égalité le choix est aléatoire.

Le timer est calculé de la même façon pour tout les rounds, pas de distinction entre le premier round et les autres, et pour tout les pourcentages selon la formule suivante, que nous allons expliquer en détail plus loin la façon dont nous l'avons obtenu :

$$T(i) = [A * (1 - \frac{E_i}{E_0}) - (\frac{E_i}{E_0 * \exp(\lambda)})] + 1 \quad (4.3)$$

E_0 représente l'énergie initiale d'un noeud.

E_i représente l'énergie résiduelle d'un noeud i .

λ représente un nombre aléatoire entre 0 et 1, tiré par chaque noeud au début de chaque round.

A représente un entier dont la valeur sera déduite dans l'analyse théorique.

4.3.3 Algorithme général

Dans ce qui suit, nous allons représenter l'algorithme qui décrit les étapes de la formation des clusters pour notre protocole :

Algorithme 4.1 Algorithme FC-ETC

```

1 : Diffusion de la SB du « NEW_ROUND_MSG »
2 : Chaque nœud  $i$  recevant le « NEW_ROUND_MSG » calcule son timer selon son énergie
3 : Si le nœud  $i$  reçoit un « ANNONCE_MSG » avant l'expiration de son timer alors
4 : Le nœud arrête son timer  $T(i)$  et attend un éventuel message « ANNONCE_MSG »
5 : Sinon
6 : Le nœud  $i$  s'autoélit et diffuse le message « ANNONCE_MSG »
7 : Fsi
8 : Si le nœud  $i$  n'est pas élu alors
9 : Le nœud  $i$  rejoint un cluster-head ayant le plus fort signal et envoie le « JOIN_MSG »
10 : Sinon
11 : Le nœud  $i$  attend la réception des message « JOIN_MSG » des autres nœuds
12 : Fsi

```

4.4 Analyse théorique

Dans cette section nous allons expliquer étape par étape comment nous sommes arrivé à obtenir les deux formules proposées pour le calcul du timer, ainsi que une breve discussion sur les valeurs du timer obtenue.

4.4.1 Choix des paramètres de la fonction objective

Le but de notre travail est d'éliminer les collisions pendant l'élection ; il faut donc avoir des valeurs du timer différentes pour la même quantité d'énergie, aussi le timer doit être inversement proportionnel à l'énergie pour élire le nœud ayant un niveau énergétique élevé comme CH. Pour cela nous avons gardé la formule proposée dans le protocole EUCA

$$T(i) = T_{max} * \left(1 - \frac{E_i}{E_0}\right) \quad (4.4)$$

Tout en changeant la T_{max} par un paramètre A dont la valeur sera déduite, selon la valeur maximum du timer estimée à 56 secondes, et en ajoutant un temps aléatoire $T2$ compris entre 0 et 1, qui dépend du rapport de l'énergie et de la valeur du nombre aléatoire λ . A cet effet nous avons dressé des tableaux, en prenant des valeurs de λ d'un pas de 0.05 et celle de taux d'énergie d'un pas de 10% afin de vérifier la fiabilité des scénarios proposés.

Nous allons prendre pour cette formule :

A paramètre dont la valeur sera déduite

$$T1 = A * \left(1 - \frac{E_i}{E_0}\right) \quad (4.5)$$

$$T2 = \frac{E_i}{E_0} * \frac{1}{\exp(\lambda)} \quad (4.6)$$

	valeurs de λ																		
Taux E λ	0,05	0,1	0,15	0,2	0,25	0,3	0,35	0,4	0,45	0,5	0,55	0,6	0,65	0,7	0,75	0,8	0,85	0,9	0,95
10	0,09512	0,09048	0,08607	0,08187	0,07788	0,07408	0,07047	0,06703	0,06376	0,06065	0,05769	0,05488	0,0522	0,04966	0,04724	0,04493	0,04274	0,04066	0,03867
20	0,19025	0,18097	0,17214	0,16375	0,15576	0,14816	0,14094	0,13406	0,12753	0,12131	0,11539	0,10976	0,10441	0,09932	0,09447	0,08987	0,08548	0,08131	0,07735
30	0,28537	0,27145	0,25821	0,24562	0,23364	0,22225	0,21141	0,2011	0,19129	0,18196	0,17308	0,16464	0,15661	0,14898	0,14171	0,1348	0,12822	0,12197	0,11602
40	0,38049	0,36193	0,34428	0,32749	0,31152	0,29633	0,28188	0,26813	0,25505	0,24261	0,23078	0,21952	0,20882	0,19863	0,18895	0,17973	0,17097	0,16263	0,1547
50	0,47561	0,45242	0,43035	0,40937	0,3894	0,37041	0,35234	0,33516	0,31881	0,30327	0,28847	0,27441	0,26102	0,24829	0,23618	0,22466	0,21371	0,20328	0,19337
60	0,57074	0,5429	0,51642	0,49124	0,46728	0,44449	0,42281	0,40219	0,38258	0,36392	0,34617	0,32929	0,31323	0,29795	0,28342	0,2696	0,25645	0,24394	0,23204
70	0,66586	0,63339	0,6025	0,57311	0,54516	0,51857	0,49328	0,46922	0,44634	0,42457	0,40386	0,38417	0,36543	0,34761	0,33066	0,31453	0,29919	0,2846	0,27072
80	0,76098	0,72387	0,68857	0,65498	0,62304	0,59265	0,56375	0,53626	0,5101	0,48522	0,46156	0,43905	0,41764	0,39727	0,37789	0,35946	0,34193	0,32526	0,30939
90	0,85611	0,81435	0,77464	0,73686	0,70092	0,66674	0,63422	0,60329	0,57387	0,54588	0,51925	0,49393	0,46984	0,44693	0,42513	0,4044	0,38467	0,36591	0,34807
100	0,95123	0,90484	0,86071	0,81873	0,7788	0,74082	0,70469	0,67032	0,63763	0,60653	0,57695	0,54881	0,52205	0,49659	0,47237	0,44933	0,42741	0,40657	0,38674

TABLE 4.1 – Tableau de variation du timer T2

La variation des valeurs du timer T2 sont représentées dans le tableau suivant :

D'après le tableau ci-dessus, nous remarquons que les valeurs du timer sont proportionnelles par rapport à l'énergie et décroissantes par rapport à λ . Aussi pour la même quantité d'énergie, quelque soit la valeur de λ

les valeurs du timer sont distincts.

Scénario 1 : $T(i) = T1 + T2$ avec $A = 1$ Les valeurs du timer que nous avons eu sont représentés dans le tableau ci-dessous comme suit :

	valeurs de λ																		
Taux E	0,05	0,1	0,15	0,2	0,25	0,3	0,35	0,4	0,45	0,5	0,55	0,6	0,65	0,7	0,75	0,8	0,85	0,9	0,95
0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
10	0,99512	0,99048374	0,98607	0,98187	0,97788	0,97408	0,97047	0,96703	0,96376	0,96065	0,957695	0,95488	0,9522	0,94966	0,94724	0,94493	0,94274	0,94066	0,93867
20	0,99025	0,98096748	0,97214	0,96375	0,95576	0,94816	0,94094	0,93406	0,92753	0,92131	0,91539	0,90976	0,90441	0,89932	0,89447	0,88987	0,88548	0,88131	0,87735
30	0,98537	0,97145123	0,95821	0,94562	0,93364	0,92225	0,91141	0,9011	0,89129	0,88196	0,873085	0,86464	0,85661	0,84898	0,84171	0,8348	0,82822	0,82197	0,81602
40	0,98049	0,96193497	0,94428	0,92749	0,91152	0,89633	0,88188	0,86813	0,85505	0,84261	0,83078	0,81952	0,80882	0,79863	0,78895	0,77973	0,77097	0,76263	0,7547
50	0,97561	0,95241871	0,93035	0,90937	0,8894	0,87041	0,85234	0,83516	0,81881	0,80327	0,788475	0,77441	0,76102	0,74829	0,73618	0,72466	0,71371	0,70328	0,69337
60	0,97074	0,94290245	0,91642	0,89124	0,86728	0,84449	0,82281	0,80219	0,78258	0,76392	0,74617	0,72929	0,71323	0,69795	0,68342	0,6696	0,65645	0,64394	0,63204
70	0,96586	0,93338619	0,9025	0,87311	0,84516	0,81857	0,79328	0,76922	0,74634	0,72457	0,703865	0,68417	0,66543	0,64761	0,63066	0,61453	0,59919	0,5846	0,57072
80	0,96098	0,92386993	0,88857	0,85498	0,82304	0,79265	0,76375	0,73626	0,7101	0,68522	0,66156	0,63905	0,61764	0,59727	0,57789	0,55946	0,54193	0,52526	0,50939
90	0,95611	0,91435368	0,87464	0,83686	0,80092	0,76674	0,73422	0,70329	0,67387	0,64588	0,619255	0,59393	0,56984	0,54693	0,52513	0,5044	0,48467	0,46591	0,44807
100	0,95123	0,90483742	0,86071	0,81873	0,7788	0,74082	0,70469	0,67032	0,63763	0,60653	0,57695	0,54881	0,52205	0,49659	0,47237	0,44933	0,42741	0,40657	0,38674

TABLE 4.2 – Tableau de variation du timer selon le scénario 1

En additionnant $T1$ et $T2$, nous remarquons les collisions sont éliminées, mais pour ce qui est de l'élection du CH ; nous constatons qu'à l'exception de certains cas, les nœuds ayant un

faible taux d'énergie résiduelle peuvent être élu devant ceux qui en ont le taux le plus élevé, prenant à titre d'exemple le nœud ayant un taux de 90% avec $\lambda = 0.05$ nous remarquons que son timer prend une valeur élevée par rapport aux autres taux la partie en jaune montre les cas où ce nœud peut être élu.

Scénario 2 : $T(i) = T1 - T2$ avec $A = 1$ Pour ce scénario nous avons eu les résultats suivants :

Taux de Ei	valeurs de λ																		
	0,05	0,1	0,15	0,2	0,25	0,3	0,35	0,4	0,45	0,5	0,55	0,6	0,65	0,7	0,75	0,8	0,85	0,9	0,95
0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
10	0,80488	0,80951626	0,81393	0,81813	0,82212	0,82592	0,82953	0,83297	0,83624	0,83935	0,842305	0,84512	0,8478	0,85034	0,85276	0,85507	0,85726	0,85934	0,86133
20	0,60975	0,61903252	0,62786	0,63625	0,64424	0,65184	0,65906	0,66594	0,67247	0,67869	0,68461	0,69024	0,69559	0,70068	0,70553	0,71013	0,71452	0,71869	0,72265
30	0,41463	0,42854877	0,44179	0,45438	0,46636	0,47775	0,48859	0,4989	0,50871	0,51804	0,526915	0,53536	0,54339	0,55102	0,55829	0,5652	0,57178	0,57803	0,58398
40	0,21951	0,23806503	0,25572	0,27251	0,28848	0,30367	0,31812	0,33187	0,34495	0,35739	0,36922	0,38048	0,39118	0,40137	0,41105	0,42027	0,42903	0,43737	0,4453
50	0,02439	0,04758129	0,06965	0,09063	0,1106	0,12959	0,14766	0,16484	0,18119	0,19673	0,211525	0,22559	0,23898	0,25171	0,26382	0,27534	0,28629	0,29672	0,30663
60	-0,1707	-0,14290245	-0,1164	-0,0912	-0,0673	-0,0445	-0,0228	-0,0022	0,01742	0,03608	0,05383	0,07071	0,08677	0,10205	0,11658	0,1304	0,14355	0,15606	0,16796
70	-0,3659	-0,33338619	-0,3025	-0,2731	-0,2452	-0,2186	-0,1933	-0,1692	-0,1463	-0,1246	-0,10386	-0,0842	-0,0654	-0,0476	-0,0307	-0,0145	0,00081	0,0154	0,02928
80	-0,561	-0,52386993	-0,4886	-0,455	-0,423	-0,3927	-0,3638	-0,3363	-0,3101	-0,2852	-0,26156	-0,239	-0,2176	-0,1973	-0,1779	-0,1595	-0,1419	-0,1253	-0,1094
90	-0,7561	-0,71435368	-0,6746	-0,6369	-0,6009	-0,5667	-0,5342	-0,5033	-0,4739	-0,4459	-0,41925	-0,3939	-0,3698	-0,3469	-0,3251	-0,3044	-0,2847	-0,2659	-0,2481
100	-0,9512	-0,90483742	-0,8607	-0,8187	-0,7788	-0,7408	-0,7047	-0,6703	-0,6376	-0,6065	-0,57695	-0,5488	-0,522	-0,4966	-0,4724	-0,4493	-0,4274	-0,4066	-0,3867

TABLE 4.3 – Tableau de variation du timer selon le scénario 2

D'après les valeurs du timer, nous constatons que les collisions sont éliminées, et pour ce qui est de l'élection du CH, la partie en vert détermine les cas où la formule est vérifiée, pour un taux allant de 0 à 30% la formule est valable pour n'importe quelle valeur de λ , mais lorsque le pourcentage atteint les 60% des valeurs négatives apparaissent.

Scénario 3 : Nous remarquons dans le tableau du scénario précédent que la plus petite valeur négative est approximativement -1 . Afin d'éliminer les valeurs négatives nous avons opté pour l'ajout de 1, nous aurons donc la formule $T(i) = (T1 - T2) + 1$ qui a engendré le tableau de variation suivant :

	valeurs de λ																	
Taux de Ei	0,05	0,1	0,15	0,2	0,25	0,3	0,35	0,4	0,45	0,5	0,55	0,6	0,65	0,7	0,75	0,8	0,85	0,9
0	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
10	1,80488	1,80951626	1,81393	1,81813	1,82212	1,82592	1,82953	1,83297	1,83624	1,83935	1,842305	1,84512	1,8478	1,85034	1,85276	1,85507	1,85726	1,85934
20	1,60975	1,61903252	1,62786	1,63625	1,64424	1,65184	1,65906	1,66594	1,67247	1,67869	1,68461	1,69024	1,69559	1,70068	1,70553	1,71013	1,71452	1,71869
30	1,41463	1,42854877	1,44179	1,45438	1,46636	1,47775	1,48859	1,4989	1,50871	1,51804	1,526915	1,53536	1,54339	1,55102	1,55829	1,5652	1,57178	1,57803
40	1,21951	1,23806503	1,25572	1,27251	1,28848	1,30367	1,31812	1,33187	1,34495	1,35739	1,36922	1,38048	1,39118	1,40137	1,41105	1,42027	1,42903	1,43737
50	1,02439	1,04758129	1,06965	1,09063	1,1106	1,12959	1,14766	1,16484	1,18119	1,19673	1,211525	1,22559	1,23898	1,25171	1,26382	1,27534	1,28629	1,29672
60	0,82926	0,85709755	0,88358	0,90876	0,93272	0,95551	0,97719	0,99781	1,01742	1,03608	1,05383	1,07071	1,08677	1,10205	1,11658	1,1304	1,14355	1,15606
70	0,63414	0,66661381	0,6975	0,72689	0,75484	0,78143	0,80672	0,83078	0,85366	0,87543	0,896135	0,91583	0,93457	0,95239	0,96934	0,98547	1,00081	1,0154
80	0,43902	0,47613007	0,51143	0,54502	0,57696	0,60735	0,63625	0,66374	0,6899	0,71478	0,73844	0,76095	0,78236	0,80273	0,82211	0,84054	0,85807	0,87474
90	0,24389	0,28564632	0,32536	0,36314	0,39908	0,43326	0,46578	0,49671	0,52613	0,55412	0,580745	0,60607	0,63016	0,65307	0,67487	0,6956	0,71533	0,73409
100	0,04877	0,09516258	0,13929	0,18127	0,2212	0,25918	0,29531	0,32968	0,36237	0,39347	0,42305	0,45119	0,47795	0,50341	0,52763	0,55067	0,57259	0,59343

TABLE 4.4 – Tableau de variation du timer selon le scénario 3

A part la disparition des valeurs négatives, le même constat pour les collisions et un petit changement pour l'élection du CH. De ce fait, et pour que les valeurs du timer dépendent plus du taux d'énergie résiduelle la valeur de $T1$ doit être plus important sur $T2$, pour cela et pour les scénarios suivants nous allons varier la valeur de A jusqu'à l'obtention des résultats satisfaisants.

Scénario 4 : $A = 2$ Avec cette valeur nous avons eu le tableau de variation suivant :

	valeurs de λ																		
Taux de Ei	0,05	0,1	0,15	0,2	0,25	0,3	0,35	0,4	0,45	0,5	0,55	0,6	0,65	0,7	0,75	0,8	0,85	0,9	0,95
0	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
10	2,70488	2,70951626	2,71393	2,71813	2,72212	2,72592	2,72953	2,73297	2,73624	2,73935	2,742305	2,74512	2,7478	2,75034	2,75276	2,75507	2,75726	2,75934	2,76133
20	2,40975	2,41903252	2,42786	2,43625	2,44424	2,45184	2,45906	2,46594	2,47247	2,47869	2,48461	2,49024	2,49559	2,50068	2,50553	2,51013	2,51452	2,51869	2,52265
30	2,11463	2,12854877	2,14179	2,15438	2,16636	2,17775	2,18859	2,1989	2,20871	2,21804	2,226915	2,23536	2,24339	2,25102	2,25829	2,2652	2,27178	2,27803	2,28398
40	1,81951	1,83806503	1,85572	1,87251	1,88848	1,90367	1,91812	1,93187	1,94495	1,95739	1,96922	1,98048	1,99118	2,00137	2,01105	2,02027	2,02903	2,03737	2,0453
50	1,52439	1,54758129	1,56965	1,59063	1,6106	1,62959	1,64766	1,66484	1,68119	1,69673	1,711525	1,72559	1,73898	1,75171	1,76382	1,77534	1,78629	1,79672	1,80663
60	1,22926	1,25709755	1,28358	1,30876	1,33272	1,35551	1,37719	1,39781	1,41742	1,43608	1,45383	1,47071	1,48677	1,50205	1,51658	1,5304	1,54355	1,55606	1,56796
70	0,93414	0,96661381	0,9975	1,02689	1,05484	1,08143	1,10672	1,13078	1,15366	1,17543	1,196135	1,21583	1,23457	1,25239	1,26934	1,28547	1,30081	1,3154	1,32928
80	0,63902	0,67613007	0,71143	0,74502	0,77696	0,80735	0,83625	0,86374	0,8899	0,91478	0,93844	0,96095	0,98236	1,00273	1,02211	1,04054	1,05807	1,07474	1,09061
90	0,34389	0,38564632	0,42536	0,46314	0,49908	0,53326	0,56578	0,59671	0,62613	0,65412	0,680745	0,70607	0,73016	0,75307	0,77487	0,7956	0,81533	0,83409	0,85193
100	0,04877	0,09516258	0,13929	0,18127	0,2212	0,25918	0,29531	0,32968	0,36237	0,39347	0,42305	0,45119	0,47795	0,50341	0,52763	0,55067	0,57259	0,59343	0,61326

TABLE 4.5 – Tableau de variation du timer selon le scénario 4

Nous remarquons que l'intervalle de pourcentage pour laquelle cette formule est vérifiée pour toute valeur de λ s'est élargie, de 0 à 50% pour ce qui est du reste la formule est vérifiée que pour certains cas par exemple pour un taux de 60% elle est vérifiée que pour λ allant de

0.05 à 0.75, tandis que pour 99% elle est vérifiée que lorsque la valeur de λ est entre 0.05 et 0.35.

Scénario 5 : $A = 3$ Pour ce scénario nous avons eu le tableau de variation suivant :

Taux de E _i	valeurs de λ																		
	0,05	0,1	0,15	0,2	0,25	0,3	0,35	0,4	0,45	0,5	0,55	0,6	0,65	0,7	0,75	0,8	0,85	0,9	0,95
0	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
10	3,60488	3,60951626	3,61393	3,61813	3,62212	3,62592	3,62953	3,63297	3,63624	3,63935	3,642305	3,64512	3,6478	3,65034	3,65276	3,65507	3,65726	3,65934	3,66133
20	3,20975	3,21903252	3,22786	3,23625	3,24424	3,25184	3,25906	3,26594	3,27247	3,27869	3,28461	3,29024	3,29559	3,30068	3,30553	3,31013	3,31452	3,31869	3,32265
30	2,81463	2,82854877	2,84179	2,85438	2,86636	2,87775	2,88859	2,8989	2,90871	2,91804	2,926915	2,93536	2,94339	2,95102	2,95829	2,9652	2,97178	2,97803	2,98398
40	2,41951	2,43806503	2,45572	2,47251	2,48848	2,50367	2,51812	2,53187	2,54495	2,55739	2,56922	2,58048	2,59118	2,60137	2,61105	2,62027	2,62903	2,63737	2,6453
50	2,02439	2,04758129	2,06965	2,09063	2,1106	2,12959	2,14766	2,16484	2,18119	2,19673	2,211525	2,22559	2,23898	2,25171	2,26382	2,27534	2,28629	2,29672	2,30663
60	1,62926	1,65709755	1,68358	1,70876	1,73272	1,75551	1,77719	1,79781	1,81742	1,83608	1,85383	1,87071	1,88677	1,90205	1,91658	1,9304	1,94355	1,95606	1,96736
70	1,23414	1,26661381	1,2975	1,32689	1,35484	1,38143	1,40672	1,43078	1,45366	1,47543	1,496135	1,51583	1,53457	1,55239	1,56934	1,58547	1,60081	1,6154	1,62928
80	0,83902	0,87613007	0,91143	0,94502	0,97696	1,00735	1,03625	1,06374	1,0899	1,11478	1,13844	1,16095	1,18236	1,20273	1,22211	1,24054	1,25807	1,27474	1,29061
90	0,44389	0,48564632	0,52536	0,56314	0,59908	0,63326	0,66578	0,69671	0,72613	0,75412	0,780745	0,80607	0,83016	0,85307	0,87487	0,8956	0,91533	0,93409	0,95193
100	0,04877	0,09516258	0,13929	0,18127	0,2212	0,25918	0,29531	0,32968	0,36237	0,39347	0,42305	0,45119	0,47795	0,50341	0,52763	0,55067	0,57259	0,59343	0,61326

TABLE 4.6 – Tableau de variation du timer selon le scénario 5

Scénario 6 : $A = 4$ Le tableau ci-dessous illustre la variation du timer :

Taux de E _i	valeurs de λ																		
	0,05	0,1	0,15	0,2	0,25	0,3	0,35	0,4	0,45	0,5	0,55	0,6	0,65	0,7	0,75	0,8	0,85	0,9	0,95
0	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
10	4,50488	4,50951626	4,51393	4,51813	4,52212	4,52592	4,52953	4,53297	4,53624	4,53935	4,542305	4,54512	4,5478	4,55034	4,55276	4,55507	4,55726	4,55934	4,56133
20	4,00975	4,01903252	4,02786	4,03625	4,04424	4,05184	4,05906	4,06594	4,07247	4,07869	4,08461	4,09024	4,09559	4,10068	4,10553	4,11013	4,11452	4,11869	4,12265
30	3,51463	3,52854877	3,54179	3,55438	3,56636	3,57775	3,58859	3,5989	3,60871	3,61804	3,626915	3,63536	3,64339	3,65102	3,65829	3,6652	3,67178	3,67803	3,68398
40	3,01951	3,03806503	3,05572	3,07251	3,08848	3,10367	3,11812	3,13187	3,14495	3,15739	3,16922	3,18048	3,19118	3,20137	3,21105	3,22027	3,22903	3,23737	3,2453
50	2,52439	2,54758129	2,56965	2,59063	2,6106	2,62959	2,64766	2,66484	2,68119	2,69673	2,711525	2,72559	2,73898	2,75171	2,76382	2,77534	2,78629	2,79672	2,80663
60	2,02926	2,05709755	2,08358	2,10876	2,13272	2,15551	2,17719	2,19781	2,21742	2,23608	2,25383	2,27071	2,28677	2,30205	2,31658	2,3304	2,34355	2,35606	2,36796
70	1,53414	1,56661381	1,5975	1,62689	1,65484	1,68143	1,70672	1,73078	1,75366	1,77543	1,796135	1,81583	1,83457	1,85239	1,86934	1,88547	1,90081	1,9154	1,92928
80	1,03902	1,07613007	1,11143	1,14502	1,17696	1,20735	1,23625	1,26374	1,2899	1,31478	1,33844	1,36095	1,38236	1,40273	1,42211	1,44054	1,45807	1,47474	1,49061
90	0,54389	0,58564632	0,62536	0,66314	0,69908	0,73326	0,76578	0,79671	0,82613	0,85412	0,880745	0,90607	0,93016	0,95307	0,97487	0,9956	1,01533	1,03409	1,05193
100	0,04877	0,09516258	0,13929	0,18127	0,2212	0,25918	0,29531	0,32968	0,36237	0,39347	0,42305	0,45119	0,47795	0,50341	0,52763	0,55067	0,57259	0,59343	0,61326

TABLE 4.7 – Tableau de variation du timer selon le scénario 6

Nous remarquons pour les scénarios 5 et 6 qu'en augmentant la valeur de A , l'intervalle du taux d'énergie résiduelle pour laquelle la formule est vérifiée s'élargit de plus en plus (la partie verte sur les tableaux).

Scénario 7 $A = 5$ Avec la valeur 5 nous avons eu le tableau suivant :

	valeurs de λ																		
Taux de E _i	0,05	0,1	0,15	0,2	0,25	0,3	0,35	0,4	0,45	0,5	0,55	0,6	0,65	0,7	0,75	0,8	0,85	0,9	0,95
0	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6
10	5,40488	5,40951626	5,41393	5,41813	5,42212	5,42592	5,42953	5,43297	5,43624	5,43935	5,442305	5,44512	5,4478	5,45034	5,45276	5,45507	5,45726	5,45934	5,46133
20	4,80375	4,81903252	4,82786	4,83625	4,84424	4,85184	4,85906	4,86594	4,87247	4,87869	4,88461	4,89024	4,89559	4,90068	4,90553	4,91013	4,91452	4,91869	4,92265
30	4,21463	4,22854877	4,24179	4,25438	4,26636	4,27775	4,28859	4,2989	4,30871	4,31804	4,326915	4,33536	4,34339	4,35102	4,35829	4,3652	4,37178	4,37803	4,38398
40	3,61951	3,63806503	3,65572	3,67251	3,68848	3,70367	3,71812	3,73187	3,74495	3,75739	3,76922	3,78048	3,79118	3,80137	3,81105	3,82027	3,82903	3,83737	3,8453
50	3,02439	3,04758129	3,06965	3,09063	3,1106	3,12959	3,14766	3,16484	3,18119	3,19673	3,211525	3,22559	3,23898	3,25171	3,26382	3,27534	3,28629	3,29672	3,30663
60	2,42326	2,45709755	2,48358	2,50876	2,53272	2,55551	2,57719	2,59781	2,61742	2,63608	2,65383	2,67071	2,68677	2,70205	2,71658	2,7304	2,74355	2,75606	2,76796
70	1,83414	1,86661381	1,8975	1,92689	1,95484	1,98143	2,00672	2,03078	2,05366	2,07543	2,096135	2,11583	2,13457	2,15239	2,16934	2,18547	2,20081	2,2154	2,22928
80	1,23902	1,27613007	1,31143	1,34502	1,37696	1,40735	1,43625	1,46374	1,4899	1,51478	1,53844	1,56095	1,58236	1,60273	1,62211	1,64054	1,65807	1,67474	1,69061
90	0,64389	0,68564632	0,72536	0,76314	0,79908	0,83326	0,86578	0,89671	0,92613	0,95412	0,980745	1,00607	1,03016	1,05307	1,07487	1,0956	1,11533	1,13409	1,15193
100	0,04877	0,09516258	0,13929	0,18127	0,2212	0,25918	0,29531	0,32968	0,36237	0,39347	0,42305	0,45119	0,47795	0,50341	0,52763	0,55067	0,57259	0,59343	0,61326

TABLE 4.8 – Tableau de variation du timer selon le scénario 7

Pour cette valeur nous remarquons que la formule est vérifiée pour toutes valeur de λ , étant donné que pour $A = 5$ la formule est vérifiée cela n'exclut pas qu'il peut avoir des valeurs pour laquelle la formule posera des problèmes, pour cela nous avons décidé de diminuer progressivement le pas de taux d'énergie et faire une analyse pour chaque tableau de valeurs. En prenant un pas de diminution de 3% nous avons eu les résultats suivants :

Premier Cas : $pas = 7$:

	valeurs de λ																		
Taux de E _i	0,05	0,1	0,15	0,2	0,25	0,3	0,35	0,4	0,45	0,5	0,55	0,6	0,65	0,7	0,75	0,8	0,85	0,9	0,95
0	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6
7	5,58341	5,58666138	5,58975	5,59269	5,59548	5,59814	5,60067	5,60308	5,60537	5,60754	5,60961	5,61158	5,61346	5,61524	5,61693	5,61855	5,62008	5,62154	5,62293
14	5,16683	5,17332276	5,1795	5,18538	5,19097	5,19629	5,20134	5,20616	5,21073	5,21509	5,21923	5,22317	5,22691	5,23048	5,23387	5,23709	5,24016	5,24308	5,24586
21	4,75024	4,75938414	4,76925	4,77807	4,78645	4,79443	4,80202	4,80923	4,8161	4,82263	4,82884	4,83475	4,84037	4,84572	4,8508	4,85564	4,86024	4,86462	4,86878
28	4,33366	4,34664552	4,359	4,37076	4,38194	4,39257	4,40269	4,41231	4,42146	4,43017	4,43845	4,44633	4,45383	4,46096	4,46774	4,47419	4,48032	4,48616	4,49171
35	3,91707	3,9333069	3,94875	3,96344	3,97742	3,99071	4,00336	4,01539	4,02683	4,03771	4,04807	4,05792	4,06728	4,0762	4,08467	4,09273	4,1004	4,1077	4,11464
42	3,50048	3,51936828	3,5385	3,55613	3,5729	3,58886	3,60403	3,61847	3,6322	3,64526	3,65768	3,6695	3,68074	3,69143	3,70161	3,71128	3,72049	3,72924	3,73757
49	3,0833	3,10662367	3,12825	3,14882	3,16833	3,187	3,2047	3,22154	3,23756	3,2528	3,26729	3,28108	3,2942	3,30667	3,31854	3,32983	3,34057	3,35078	3,3605
56	2,66731	2,69323105	2,718	2,74151	2,76387	2,78514	2,80537	2,82462	2,84293	2,86034	2,87691	2,89267	2,90765	2,92191	2,93547	2,94838	2,96065	2,97232	2,98343
63	2,25073	2,27935243	2,30775	2,3342	2,35936	2,38328	2,40605	2,4277	2,44829	2,46789	2,48652	2,50425	2,52111	2,53715	2,55241	2,56692	2,58073	2,59386	2,60635
70	1,83414	1,86661381	1,8975	1,92689	1,95484	1,98143	2,00672	2,03078	2,05366	2,07543	2,09614	2,11583	2,13457	2,15239	2,16934	2,18547	2,20081	2,2154	2,22928
77	1,41755	1,45327518	1,48725	1,51958	1,55032	1,57957	1,60733	1,63385	1,65903	1,68297	1,70575	1,72742	1,74802	1,76763	1,78628	1,80402	1,82089	1,83694	1,85221
84	1,00097	1,03939657	1,07701	1,11227	1,14581	1,17771	1,20806	1,23693	1,26439	1,29051	1,31536	1,339	1,36148	1,38287	1,40321	1,42256	1,44097	1,45848	1,47514
91	0,58438	0,62653795	0,66676	0,70496	0,74129	0,77586	0,80873	0,84001	0,86976	0,89806	0,92498	0,95058	0,97494	0,99811	1,02015	1,04111	1,06105	1,08002	1,09807
98	0,1678	0,21325933	0,25651	0,29764	0,33678	0,374	0,40941	0,44309	0,47512	0,5056	0,53459	0,56216	0,5884	0,61335	0,63708	0,65966	0,68113	0,70156	0,72099
100	0,04877	0,09516258	0,13929	0,18127	0,2212	0,25918	0,29531	0,32968	0,36237	0,39347	0,42305	0,45119	0,47795	0,50341	0,52763	0,55067	0,57259	0,59343	0,61326

TABLE 4.9 – Tableau des variation du timer pour le 1er cas

Deuxième cas : $pas = 4$

Taux de Et	valeurs de λ																		
	0,05	0,1	0,15	0,2	0,25	0,3	0,35	0,4	0,45	0,5	0,55	0,6	0,65	0,7	0,75	0,8	0,85	0,9	0,95
4	5,76195082	5,7638065	5,76557168	5,76725077	5,76884797	5,77036727	5,77181248	5,7731872	5,77449487	5,77573877	5,77692201	5,77804753	5,77911817	5,78013659	5,78110534	5,78202684	5,7829034	5,78373721	5,78453036
8	5,52390165	5,52761201	5,53114336	5,53450154	5,53769594	5,54073454	5,54362495	5,5463744	5,54898975	5,55147755	5,55384402	5,55609507	5,55823634	5,56027318	5,56221068	5,56405368	5,56580161	5,56747443	5,56906072
12	5,28585247	5,29141951	5,29671504	5,30175231	5,30654391	5,31110181	5,31543743	5,31956159	5,32348462	5,32721632	5,33076402	5,3341426	5,33735451	5,34040976	5,34331601	5,34608052	5,34871021	5,35121164	5,35359108
16	5,04780329	5,05222601	5,05622872	5,06000308	5,06359187	5,06704698	5,0703987	5,07364991	5,07680249	5,07987975	5,08295509	5,08603803	5,08911904	5,09220818	5,09520435	5,09819737	5,10118737	5,10417431	5,10715814
20	4,80975412	4,81903252	4,8278584	4,8362359	4,84423984	4,85183636	4,85906238	4,86595999	4,87247437	4,87869387	4,88461004	4,89023767	4,89559084	4,90068294	4,90552689	4,91013421	4,91451701	4,91868607	4,9226518
24	4,57170494	4,58283902	4,59343009	4,60359462	4,613308781	4,622220363	4,63087486	4,63912319	4,64696924	4,65443264	4,66153205	4,66828521	4,67470901	4,68081952	4,68663203	4,69216105	4,69742042	4,70242328	4,70718215
28	4,33365576	4,34664552	4,35900177	4,37075529	4,38193578	4,3925709	4,40268733	4,41231029	4,42146412	4,43017142	4,43845405	4,44633274	4,45382718	4,46095611	4,46773737	4,47418789	4,48032382	4,4861605	4,49171251
32	4,09560658	4,11045203	4,12457345	4,13800616	4,15078275	4,16293817	4,17449981	4,18549759	4,19595999	4,20591019	4,21537606	4,22436028	4,23294525	4,2410927	4,2488427	4,25621473	4,26322722	4,26999771	4,27624287
36	3,85755741	3,87425853	3,89041513	3,90525693	3,91962172	3,93339054	3,94631229	3,95868478	3,97045387	3,98164896	3,99229907	4,00242781	4,01206352	4,02122929	4,02994904	4,03824157	4,04613062	4,05363492	4,06077323
40	3,61950823	3,63806503	3,65571681	3,6725077	3,68847969	3,703647271	3,71812476	3,73187198	3,74494874	3,75737774	3,76922008	3,78047535	3,79118169	3,80136588	3,81105338	3,82026341	3,82903403	3,83737214	3,84530359
44	3,38145905	3,40187154	3,42128849	3,43975847	3,45732766	3,47402998	3,48993724	3,50505918	3,51944361	3,5332651	3,54642008	3,55892288	3,57079986	3,58195027	3,59238572	3,60229526	3,61169374	3,6205935	3,62893295
48	3,14340988	3,16567804	3,18686017	3,20700924	3,22617562	3,24440725	3,26174972	3,27824628	3,29393849	3,30886528	3,32306409	3,33657041	3,34941803	3,36163905	3,37326405	3,3843221	3,3948403	3,40484656	3,41436431
52	2,9053607	2,92948454	2,95243185	2,97426001	2,99502359	3,01477453	3,03356219	3,05143358	3,06843336	3,08460406	3,0999861	3,11461798	3,1285362	3,14177564	3,15436929	3,16634894	3,17774424	3,18858378	3,19889467
56	2,66731152	2,69322105	2,71800353	2,74151078	2,76387156	2,7851418	2,80537467	2,82462077	2,84292824	2,86034283	2,87690311	2,89266548	2,90765437	2,92191223	2,93547473	2,94833758	2,96054764	2,97220299	2,98342503
60	2,42926235	2,45709755	2,48357521	2,50876155	2,53271953	2,55550907	2,57718745	2,59780797	2,61742311	2,6360816	2,65383011	2,67071302	2,68677253	2,70204882	2,71658007	2,73040262	2,74355104	2,7560582	2,76795539
64	2,19121317	2,22090405	2,2691469	2,27610232	2,3015675	2,32587624	2,34899962	2,37099517	2,39191798	2,41182038	2,43075212	2,44876055	2,4658907	2,48218541	2,49768541	2,51242946	2,52645444	2,53979542	2,55248574
68	1,95316399	1,98471056	2,01471858	2,04326309	2,07041547	2,09624361	2,1208121	2,14412237	2,16641286	2,18755915	2,20767413	2,22680809	2,24500687	2,26232199	2,27879074	2,2944563	2,30935795	2,32359263	2,33710161
72	1,71511481	1,74851706	1,78029026	1,81051386	1,83926344	1,86661038	1,89262458	1,91736957	1,94090773	1,96329793	1,98469614	2,00485562	2,02412704	2,04258558	2,05989608	2,07648315	2,09226125	2,10726984	2,12154646
76	1,47706564	1,51232356	1,54586194	1,57776463	1,6081114	1,63697815	1,66443705	1,69055677	1,71540026	1,7390367	1,76151814	1,78290316	1,80324521	1,82259517	1,84001042	1,85580999	1,87516465	1,89100706	1,90607682
80	1,23901646	1,27612007	1,31143262	1,34501954	1,37695937	1,40734542	1,43624952	1,46374396	1,48999748	1,51477547	1,53844015	1,56095069	1,58236328	1,60273176	1,62210676	1,64053683	1,65806605	1,67474427	1,69060718
84	1,00096728	1,02993657	1,0770052	1,11226617	1,14580734	1,17771269	1,208062	1,23692116	1,26439235	1,29051425	1,31536216	1,33899823	1,36148155	1,38286834	1,4032121	1,42256367	1,44097146	1,45848149	1,47512754
88	0,76291811	0,80374307	0,84257693	0,87951694	0,91465551	0,94807997	0,97987448	1,0101636	1,03898723	1,06625302	1,09202417	1,11704576	1,14059972	1,16200492	1,18134743	1,20459051	1,2237486	1,2422187	1,25967679
92	0,52486893	0,56754958	0,60818866	0,64676771	0,68350328	0,71844724	0,75168696	0,78330556	0,8133821	0,84199179	0,86920617	0,89509329	1,3677036	0,94314152	0,96542277	0,98661735	1,00677826	1,02595591	1,04419826
96	0,28681975	0,32135606	0,37372034	0,41401840	0,45235125	0,48881451	0,52349942	0,55649276	0,58787697	0,61773057	0,64612818	0,67314003	0,698932605	0,72227811	0,74652311	0,76864419	0,78968167	0,80964913	0,82872862
100	0,04877058	0,09516258	0,13929202	0,18126925	0,22119922	0,25920776	0,29531191	0,32964795	0,36237185	0,39346934	0,42305019	0,45118836	0,47795422	0,5034147	0,52763345	0,55067104	0,57258507	0,59343034	0,61325898

TABLE 4.10 – Tableau de variation du timer 2ème cas

Nous remarquons que des valeurs ne vérifiant pas la formule apparaissent avec un pas de 7%, pour des niveaux énergétiques élevés ; 5 cas pour un taux allant de 91% à 98%, mais elles sont minimales, tandis qu'avec un pas de 4% elles deviennent de plus en plus importantes car l'intervalle des taux énergétique s'élargit d'un taux compris entre 40% et 96% on enregistre 98 cas. A cet effet nous pouvons conclure que plus nous diminuons le pas plus le nombre de cas où l'élection des nœuds ayant un taux d'énergie moins élevé augmente, il atteint son maximum lorsque l'énergie s'approche des 100%. Ce qui nous mène à dire que pour pouvoir optimiser notre modèle il faut diminuer d'avantage le pas jusqu'à 1% et l'essayer pour l'intervalle des taux les plus élevés c'est-à-dire de 90% à 100%, tout en changeant la valeur de A .

Troisième cas $pas = 1$

Taux de E _i	valeurs de λ																		
	0,05	0,1	0,15	0,2	0,25	0,3	0,35	0,4	0,45	0,5	0,55	0,6	0,65	0,7	0,75	0,8	0,85	0,9	0,95
89	0,70341	0,7446947	0,78397	0,82133	0,85687	0,89067	0,92283	0,95342	0,98251	1,01019	1,036515	1,06156	1,08538	1,10804	1,12959	1,1501	1,1696	1,18815	1,2058
90	0,64389	0,68564632	0,72536	0,76314	0,79908	0,83326	0,86578	0,89671	0,92613	0,95412	0,980745	1,00607	1,03016	1,05307	1,07487	1,0956	1,11533	1,13409	1,15193
91	0,58438	0,62659795	0,66676	0,70496	0,74129	0,77586	0,80873	0,84001	0,86976	0,89806	0,924976	0,95058	0,97494	0,99811	1,02015	1,04111	1,06105	1,08002	1,09807
92	0,52487	0,56754958	0,60815	0,64677	0,6835	0,71845	0,75169	0,78331	0,81338	0,84199	0,869206	0,89509	0,91972	0,94314	0,96542	0,98662	1,00678	1,02596	1,0442
93	0,46536	0,5085012	0,54954	0,58858	0,62572	0,66104	0,69464	0,7266	0,75701	0,78593	0,813437	0,83961	0,8645	0,88818	0,9107	0,93212	0,9525	0,97189	0,99033
94	0,40584	0,44945283	0,49093	0,53039	0,56793	0,60363	0,63759	0,6699	0,70063	0,72986	0,757667	0,78412	0,80928	0,83321	0,85598	0,87763	0,89823	0,91782	0,93646
95	0,34633	0,39040445	0,43233	0,47221	0,51014	0,54622	0,58055	0,6132	0,64425	0,6738	0,701898	0,72863	0,75406	0,77824	0,80125	0,82314	0,84396	0,86376	0,8826
96	0,28682	0,33135608	0,37372	0,41402	0,45235	0,48881	0,5235	0,55649	0,58788	0,61773	0,646128	0,67314	0,69884	0,72328	0,74653	0,76864	0,78968	0,80969	0,82873
97	0,22731	0,2723077	0,31511	0,35583	0,39456	0,43141	0,46645	0,49979	0,5315	0,56167	0,590359	0,61765	0,64362	0,66831	0,6918	0,71415	0,73541	0,75563	0,77486
98	0,1678	0,21325933	0,25651	0,29764	0,33678	0,374	0,40941	0,44309	0,47512	0,5056	0,534589	0,56216	0,5884	0,61335	0,63708	0,65966	0,68113	0,70156	0,72099
99	0,10828	0,15421096	0,1979	0,23946	0,27899	0,31659	0,35236	0,38638	0,41875	0,44953	0,47882	0,50668	0,53317	0,55838	0,58236	0,60516	0,62686	0,6475	0,66713
100	0,04877	0,09516258	0,13929	0,18127	0,2212	0,25918	0,29531	0,32968	0,36237	0,39347	0,42305	0,45119	0,47795	0,50341	0,52763	0,55067	0,57259	0,59343	0,61326

TABLE 4.11 – Tableau de variation du timer 3ème cas

D'après le tableau ci-dessus nous remarquons que la formule est vérifiée que pour 89% mais pas pour les autres 11 taux, on suppose que si on multiplie A par 11 on aura une vérification totale ainsi $A = 55$, on a eu le tableau de variation du timer suivant :

Taux de E _i	valeurs de λ																		
	0,05	0,1	0,15	0,2	0,25	0,3	0,35	0,4	0,45	0,5	0,55	0,6	0,65	0,7	0,75	0,8	0,85	0,9	0,95
90	6,14389	6,18564632	6,22536	6,26314	6,29908	6,33326	6,36578	6,39671	6,42613	6,45412	6,480745	6,50607	6,53016	6,55307	6,57487	6,5956	6,61533	6,63409	6,65193
91	5,53438	5,57659795	5,61676	5,65496	5,69129	5,72586	5,75873	5,79001	5,81976	5,84806	5,874976	5,90058	5,92494	5,94811	5,97015	5,99111	6,01105	6,03002	6,04807
92	4,92487	4,96754958	5,00815	5,04677	5,0835	5,11845	5,15169	5,18331	5,21338	5,24199	5,269206	5,29509	5,31972	5,34314	5,36542	5,38662	5,40678	5,42596	5,4442
93	4,31536	4,3585012	4,39954	4,43858	4,47572	4,51104	4,54464	4,5766	4,60701	4,63593	4,663437	4,68961	4,7145	4,73818	4,7607	4,78212	4,8025	4,82189	4,84033
94	3,70584	3,74945283	3,79093	3,83039	3,86793	3,90363	3,93759	3,9699	4,00063	4,02986	4,057667	4,08412	4,10928	4,13321	4,15598	4,17763	4,19823	4,21782	4,23646
95	3,09633	3,14040445	3,18233	3,22221	3,26014	3,29622	3,33055	3,3632	3,39425	3,4238	3,451898	3,47863	3,50406	3,52824	3,55125	3,57314	3,59396	3,61376	3,6326
96	2,48682	2,53135608	2,57372	2,61402	2,65235	2,68881	2,7235	2,75649	2,78788	2,81773	2,846128	2,87314	2,89884	2,92328	2,94653	2,96864	2,98968	3,00969	3,02873
97	1,87731	1,9223077	1,96511	2,00583	2,04456	2,08141	2,11645	2,14979	2,1815	2,21167	2,240359	2,26765	2,29362	2,31831	2,3418	2,36415	2,38541	2,40563	2,42486
98	1,2678	1,31325933	1,35651	1,39764	1,43678	1,474	1,50941	1,54309	1,57512	1,6056	1,634589	1,66216	1,6884	1,71335	1,73708	1,75966	1,78113	1,80156	1,82099
99	0,65828	0,70421096	0,7479	0,78946	0,82899	0,86659	0,90236	0,93638	0,96875	0,99953	1,02882	1,05668	1,08317	1,10838	1,13236	1,15516	1,17686	1,1975	1,21713
100	0,04877	0,09516258	0,13929	0,18127	0,2212	0,25918	0,29531	0,32968	0,36237	0,39347	0,42305	0,45119	0,47795	0,50341	0,52763	0,55067	0,57259	0,59343	0,61326

TABLE 4.12 – Tableau de variation du timer A=55

Avec le $A = 55$ toutes les valeurs du timer vérifient la formule pour toute valeur de λ , nous constatons que la plus grande valeur que peut prendre un timer pour un taux d'énergie quelconque est plus petite que la plus petite valeur que peut prendre le taux énergétique strictement inférieur, ainsi la formule de calcul du timer est la suivante

$$T(i) = [55 * (1 - \frac{E_i}{E_0}) - (\frac{E_i}{E_0 * \exp(\lambda)})] + 1 \quad (4.7)$$

4.4.2 Discussion des valeurs du timer

Afin de faire une petite analyse sur les valeurs du timer, nous avons extrait deux graphes du dernier tableau que nous avons eu pour avoir la formule finale, le premier représente la variation du timer selon les valeurs de λ , et le deuxième représente la variation des valeur du timer selon l'énergie.

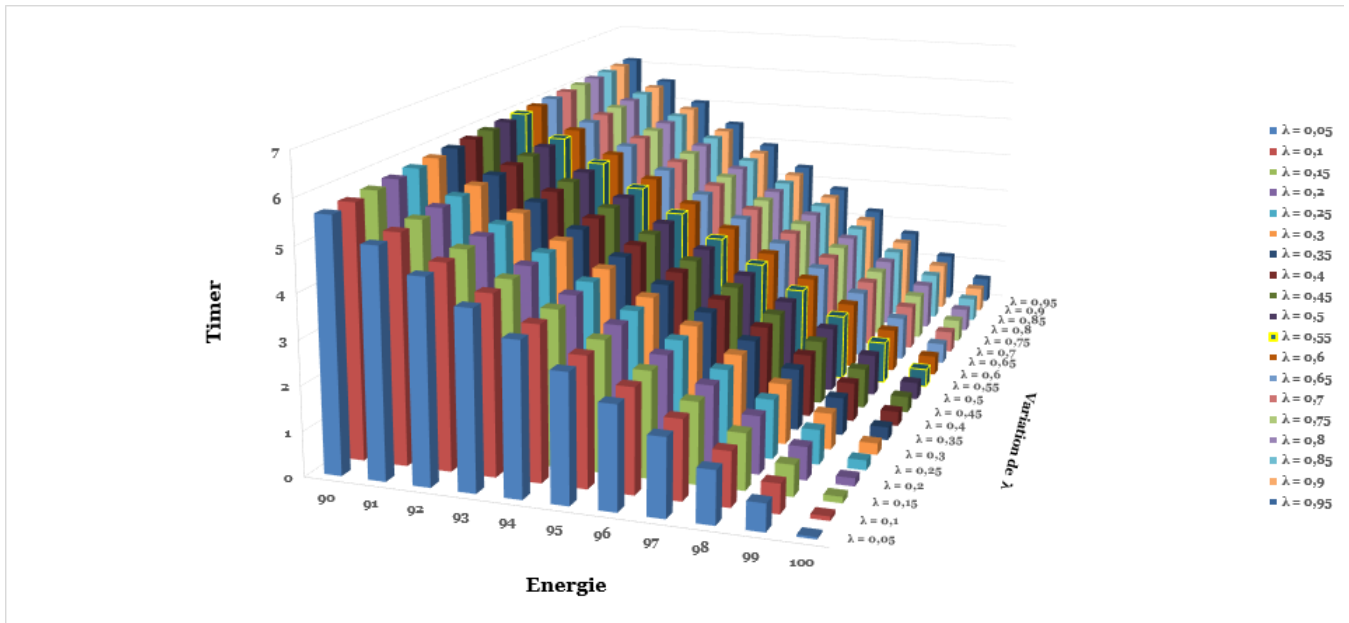


FIGURE 4.3 – Graphe représentant la variation du timer selon les valeurs de λ

Nous constatons dans le dernier tableau dressé de la section précédente, que les valeurs sont toutes différentes pour le même niveau énergétique, comme nous pouvons le voir sur la graphe ci-dessus, ce qui explique que les collisions des messages durant l'élection sont éliminées, nous pouvons aussi constater que les valeurs du timer varient selon la variation des valeurs de λ , c'est à dire que lorsque λ tend vers sa valeurs maximal le timer aussi tend vers sa valeur maximale, même constat pour les valeurs minimales, nous pouvons aussi remarquer dans la graphe que les lignes ne se croisent jamais, et cela signifie qu'à aucun moment un nœud ayant un taux énergétique bas peut avoir une valeur du timer moins élevé qu'un nœud ayant un niveau énergétique élevé.

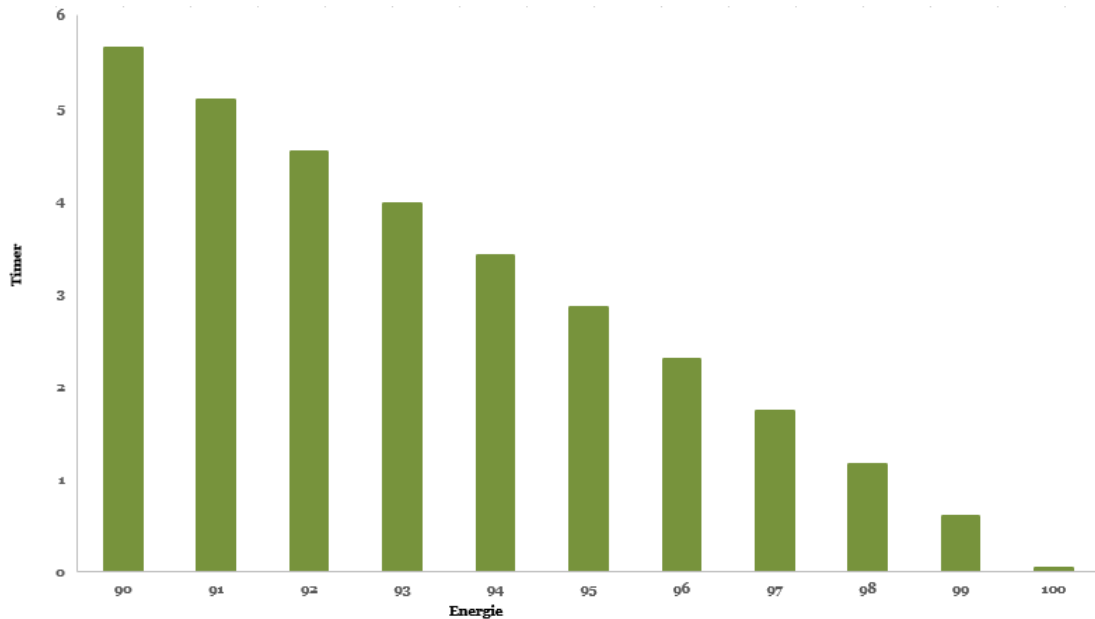


FIGURE 4.4 – Graphe représentant la variation du timer selon l'énergie

Pour le deuxième graphe, nous remarquons que la valeur du timer est inversement proportionnelle à l'énergie, ainsi le timer prend sa valeur minimale quand l'énergie atteint son maximum, autrement dit lorsque le taux d'énergie est à 100% la valeur du timer tend vers le zéro, ainsi nous pouvons conclure que les valeurs du timer varient entre 0 et 56 secondes.

4.5 Analyse expérimentale

Dans ce qui suit on va décrire l'environnement prévu pour la simulation TOSSIM, les paramètres des simulation, l'implémentation, et en fin les métriques de performances.

4.5.1 Environnement de la simulation TOSSIM

L'environnement qu'on a prévu pour la simulation va être décrit en détail dans cette section.

4.5.1.1 Le système d'exploitation TinyOs

Présentation : TinyOS est un système d'exploitation open source conçu pour les réseaux des capteurs et développé par l'université BERKELEY de Californie à la fin des années quatre-vingt-dix. Ce système d'exploitation est programmé en NesC; langage orienté composant proche du langage C. Il respecte une architecture basée sur une association de composants réduisant ainsi la taille du code nécessaire à sa mise en place. De plus TinyOS dispose d'une bibliothèque de composants particulièrement complète, on y retrouve des protocoles réseaux, des pilotes de capteurs, et des outils d'acquisition de données. L'ensemble de ces

composants peut être utilisé tel quel ou bien il peut être adapté à une application bien précise[23, 69, 30].

Allocation de la mémoire : Dans les réseaux de capteur sans fil le problème de la mémoire est récurrent, il est donc important de bien la gérer. TinyOS dispose d'une très faible mémoire de 300 à 400 octets, en plus de cela 4ko supplémentaire répartie comme suit :

- Pile : sert de mémoire temporaire au fonctionnement du système notamment pour l'empilement et le dépilement des variables locales.
- Les variables globales : réservant un espace mémoire pour le stockage de valeurs accessible depuis des applications différentes.
- La mémoire libre : pour le reste du stockage temporaire.

La gestion de la mémoire possède de plus quelques propriétés. Ainsi, il n'y a pas d'allocation dynamique de mémoire et pas de pointeurs de fonctions. Bien sûr cela simplifie grandement l'implémentation. Par ailleurs, il n'existe pas de mécanisme de protection de la mémoire sous TinyOS ce qui rend le système particulièrement vulnérable[12, 69].

Propriété : TinyOS s'adapte aux systèmes à faible ressources, de ce fait on peut distinguer quatre grandes propriétés :

- *Événementiel* : Le fonctionnement de TinyOS s'appuie sur la gestion des événements. Ainsi le capteur n'est actif que lorsqu'un événement se produit comme par exemple la réception d'un message, dans le cas contraire il en veille[69]. Ce fonctionnement événementiel (event driven) s'oppose au fonctionnement dit temporel (time driven), où les actions du système sont gérées par une horloge donnée[23].
- *Non préemptif* : cette propriété précise si le système d'exploitation permet l'interruption d'une tâche en cours ou non. TinyOS ne gère pas ce mécanisme de préemption entre les tâches, mais donne la priorité aux interruptions matérielles. Ainsi, les tâches entre elles ne s'interrompent pas mais une interruption peut stopper l'exécution d'une tâche.
- *Pas de temps réel* : Lorsqu'un système est dit « temps réel » celui-ci gère des niveaux de priorité dans ses tâches, permettant de respecter des échéances données par son environnement. Dans le cas d'un système strict, aucune échéance ne tolère de dépassement contrairement à un système temps réel. TinyOS se situe au-delà de ce second type, car il n'est pas prévu pour avoir un fonctionnement temps réel.
- *Consommation d'énergie* : TinyOS a été conçu pour préserver au maximum l'énergie du capteur. Ainsi, lorsqu'aucune tâche n'est pas active, il se met automatiquement en veille.

4.5.1.2 Le Langage NesC :

Le langage NesC (network embedded system C)[1], est un langage orienté composant conçu pour les systèmes embarqués distribués en particulier les RCSFs. Une application est un ensemble de composants, ayant un but précis. Chaque composant correspond à un élément matériel (Leds, ADC. . .); et peut être réutilisé dans différentes applications, un composant est constitué de trois partie essentielles[69, 2, 26, 18] :

- Les interfaces : Spécifient un ensemble de fonctions à mettre en application par le fournisseur des interfaces (commandes) et par l'utilisateur des interfaces (événements). Ces fonctions sont précédées par des mots-clés respectifs "command" ou "event". L'utilisation des mots clés "use" et "provide" au début d'un composant permet de savoir respectivement si celui-ci fait appel à une fonction de l'interface ou redéfinit son code. De plus, tous les composants possèdent l'interface StdControl car sa tâche est l'initialisation, le démarrage et l'arrêt des composants.
- Les modules : Contiennent le code de l'application, en mettant en œuvre une ou plusieurs interfaces.
- Les configurations : définissent les composants et/ou les interfaces utilisés par l'application déployée sur le capteur, ainsi que la description des liaisons entre ces composants.

4.5.1.3 Les outils de simulation :

TinyOS offre les outils de simulation ; Tinyviz, TOSSIM et Power-TOSSIM [18] :

TOSSIM : C'est un simulateur de TinyOs, permettant la simulation du comportement des capteurs au sein d'un RCSF comme l'envoi et /ou la réception des messages via des ondes radio. Pour une meilleure visualisation et compréhension de l'activité du réseau, TOSSIM est souvent utilisé avec l'interface graphique Tinyviz[26, 23, 18].

Power- TOSSIM : Conçu par l'université de Harvard, le simulateur Power-TOSSIM, permet de faire des simulations de la même manière que TOSSIM en prenant en considération la consommation énergétique, ainsi le nœud n'ayant plus d'énergie s'arrête de fonctionner ce qui nous permet d'exécuter la simulation jusqu'à la mort du réseau [2, 18].

Tinyviz : Est une application graphique Java, qui donne un aperçu de notre réseau de capteurs à tout instant, ainsi que des divers messages qu'ils émettent. Il permet de déterminer un délai entre chaque itération des capteurs afin de permettre une analyse pas à pas du bon déroulement des actions. Il possède aussi des options afin de pouvoir simuler la consommation de l'énergie[23, 18].

4.5.2 Paramètre de simulation

Pour lancer les simulations, certains paramètres sont présentés par le tableau ci-dessous :

FC-ETC _ROUND_LENGTH	100 secondes (100 000 millisecondes)
FC-ETC _ANNONCE_LENGTH	56 secondes (56 000 millisecondes)
FC-ETC _ORGANISATION_LENGTH	40 secondes (40 000 millisecondes)
FC-ETC _SLOT_LENGTH	1 seconde (1000 millisecondes)
K	5% (le nombre des CHs élus parmi la totalité des nœuds pour chaque round)
N	100 (nombre total des nœuds du réseau)
Energie initiale	1500 joules
La taille du paquet de contrôle	29 octet (paquet de transmission de TinyOS)

TABLE 4.13 – Paramètre de simulation

4.5.3 Implémentation

Dans ce qui suit nous allons décrire les structures de données ainsi que les principaux commandes et événements nécessaires pour l'implémentation du protocole.

Structure de données : Les paquets dans TinyOS sont envoyés dans une structure appelés TOS_Msg, les différentes structures du paquet selon le rang du nœud sont :

— *La Station de Base :*

```
typedef struct PUIITS
{
  uint16_t ID; // l'identificateur de chaque nœud
  uint8_t round; // le round courant uint8_t
  Depth; // la profondeur du nœud dans le réseau
}PUIITS;
```

— *Le Cluster-head :*

```
typedef struct CLUSTER_HEAD
{
  uint16_t ID_MEMBRE; //l'identificateur de chaque nœud
  uint16_t ID_CH; //l'identificateur du CH dont lequel appartiendra le nœud membre
  uint8_t donnee_aggreger; //la donnée agrégée à envoyer au nœud PUIITS
  uint16_t FREQ; //La fréquence avec laquelle les membre d'un membre Cluster envoi
  uint16_t SLOT_ATTRIBUER; //le slot attribuer À chaque membre
  uint8_t NBR_MBR; // le nombre des membre utilisé pour la connectivité
}CLUSTER_HEAD;
```

— *Le Noeud membre*

```

typedef struct MEMBRE
{
  uint16_t ID_MEMBRE; // l'identificateur d'un nœud qui correspond
  uint16_t ID_CH; // l'identificateur du CH
  uint8_t temp; // valeur captée
  uint8_t req; // req=1 le nœud membre prévient le CH qu'il fait partie de son groupe,
  req= 2 le nœud envoie la valeur captée
}MEMBRE;

```

Événements et Tâches Les principaux événements et tâches utilisés pour notre implémentation sont :

Réception du round

```

event TOS_MsgPtr Round_ReceiveMsg.receive(TOS_MsgPtr pmsg)
{
  ptr = (struct PUIITS *) (pmsg->data);
  /******calcul d'énergie*****/
  len=sizeof (TOS_MsgPtr);
  rec_dt=(float)len;
  update_my_energy(RX_TYPE);
  /******/
  dbg(DBG_USR2, "réception du nœud %i d'un nouveau round %i du noeud %i\n",TOS_LOCAL_ADDR,
  >round,ptr->ID);
  depth=ptr->Depth+1;
  depth_recu=TRUE;
  dbg(DBG_USR2, "Ma profondeur est %i\n",depth);
  dbg(DBG_USR2, "le neoud %i dont j'ai reçu le msg est de profondeur %i\n",ptr->ID,ptr-
  >Depth);
  if( r_CH<ptr->round)
  {
    Depth_CH_Init=0xff; // à chaque nouveau round on initialise la profondeur initiale pour
    la comparer à celle des cluster-head et prend la plus petite
    round_CH=FALSE;
    r_CH=ptr->round;
    recu_r=FALSE;
    if (r_CH%N/K)==0) //le round doit être réinitialisé à zéro car il a atteint N/K round
    donc tous les CH doivent revenir à faux
    {
      is_CH=FALSE;
      Membre=0;
      Membre_Reel=0;

```

```

    temper_moy_Agr=0;
}
}
if ( (!recu_r) && (!is_CH) &&(TOS_LOCAL_ADDRESS != BASE_STATION_ADDRESS))
{
    float randNo = (float)call Random.rand()/100 000;
    recu_r=TRUE;
    call ReqRelay_Timer.start(TIMER_ONE_SHOT,(call Random.rand())%800+200); //
    declenchement du timer pour la requette du relais
    dbg(DBG_USR2,"j'ai reçu le msg du neoud %i \n",ptr->ID);
    dbg(DBG_USR2,"j'ai reçu le début du round=%i\n",ptr->round);
    T=(5*(1-ER/ENI)-(ER/ENI*exp(randNo)))+1; // calcul du timer
    call Annonce_Timer.start(TIMER_ONE_SHOT,(uint16_t)T); // declenchement du ti-
mer
}
}
return pmsg;
}

```

Expiration de l' Annonce timer event result_t Annonce_Timer.fired()

```

{
//allumer les leds pendantl'envoi du message d'annonce
call Leds.redToggle();
call Leds.greenToggle();
call Leds.yellowToggle();
is_CH=TRUE;
round_CH=TRUE;
dbg(DBG_USR2,"je suis le noeuds %i j'annonce mon statut \n ",TOS_LOCAL_ADDRESS);
ptr = (struct PUIITS*)(buffer.data);
ptr->round=r_CH;
ptr->ID=TOS_LOCAL_ADDRESS;

call ANNOUNCE_SendMsg.send(TOS_BCAST_ADDR,sizeof(struct PUIITS),&buffer);

return SUCCESS; }

```

Tâche d'initialisation

```

static void init()
{

```

```

if (TOS_LOCAL_ADDRESS == BASE_STATION_ADDRESS) {
r=-1;
K=(uint16_t)N*5/100; //le pourcentage de CH par round est de 0.05
if (K==0 ){K++;}
depth_recu=TRUE;
depth=0;
dbg(DBG_USR2, "je suis la station de base \n");
dbg(DBG_USR2, " j'initialise le round 0 zéro\n");
dbg(DBG_USR2, "le nombre le CH décidera pour chaque round est : %i\n",K);
}
}

```

Tâche Calcul d'énergie

```

static void update_my_energy(int type) {
if (type == RX_TYPE)
{
ER=(ER-((rec_dt/3600*10000)* RADIO_RX));// calcule du l'énergie résiduelle apres
reception
}
else
{
if (type == TX_TYPE) {
ER=(ER-((send_dt/3600*10000)* RADIO_TX_FF)); // calcule du l'énergie résiduelle
après transmission
}
}
}
}

```

Commande Commande d'initialisation

```

command result_t StdControl.init()
{
init();
call Random.init();
return SUCCESS;
}

```

4.5.4 Métrique de performance

Nous nous sommes intéressés à la consommation d'énergie des nœuds puisqu'elle constitue un paramètre primordial pour la détermination de la durée de vie d'un RCSF. Ainsi que les collisions lors d'élection des nœuds ayant même niveau d'énergie comme CHs. Pour cela, nous analysons donc l'impact du modèle d'élection des CHs intégrés dans le protocole FC-ETC sur l'énergie consommée et le nombre de collisions, toute en gardant le calcul du temporisateur inversement proportionnel à l'énergie résiduelle.

4.6 Conclusion

Dans ce chapitre nous avons présenté un nouveau protocole de routage le « Free Collision based Energy and Time Clustering Algorithm » qu'on a proposé comme solution aux insuffisances des protocoles étudiés dans le chapitre précédents, nous avons expliqué son fonctionnement, le modèle du réseau et énergétique, on a aussi dressé des tableaux et des graphes afin de montrer l'apport de ce protocole en termes de présence de collisions et l'élection des CH, nous avons aussi évoqué l'environnement prévu pour la simulation, et par faute de temps nous n'avons pas pu arriver à terme de notre travail, mais on estime que les résultats obtenus dans l'analyse théorique refléteront de près la réalité.

5 Conclusion générale

Au cours de notre projet de fin d'études on a eu l'occasion d'approfondir nos connaissances dans le domaine des réseaux de capteurs sans fil, la flexibilité, la tolérance aux pannes, et le coût, réduit, etc ; sont les caractéristiques qui leur ont permis d'étendre d'avantage les domaines d'applications.

D'un autre côté, la contrainte majeure de ce type de réseaux est la consommation d'énergie au niveau des capteurs qui a une grande influence sur la durée de vie du réseau en entier, de ce fait la majorité des travaux de recherches menés actuellement se basent essentiellement sur les moyens de réduire au minimum l'énergie consommée dans la communication de données de sorte à maximiser la durée de vie du réseau.

Dans ce mémoire, nous avons proposé, en se basant principalement sur les protocoles LEACH et EUCA, un nouveau protocole de routage hiérarchique « Free Collision based Energy and Time Clustering Algorithm », avec une approche de clustering et une nouvelle formule de calcul du timer, visant à améliorer l'élection des nœuds autant que leader en évitant les collisions afin d'optimiser les communications dans les réseaux de capteurs sans fil.

Pour tester la solution proposée, le protocole a été implémenté partiellement, sur le simulateur TOSSIM, en effet le facteur de temps n'était pas en notre faveur, pour cela nous avons fait une analyse théorique approfondie qui a donné des résultats satisfaisants.

En perspective, nous proposons de continuer notre implémentation, afin d'obtenir des résultats, et des éventuels changements

Bibliographie

- [1] A. Badaoui, M.A. Mostefaoui, « Acquisition des données à distance dans les réseaux de capteur sans fil, Mémoire de master en informatique », Université Abou Bekr Belkaid Tlemcen, Algérie, 2013.
- [2] A. Berrachedi, A.diarbakirli, « Sécurisation du protocole de routage hiérarchique LEACH dans les réseaux de capteurs sans fil », Mémoire d'ingénieur en informatique, Ecole nationale Supérieure d'Informatique, Alger, 2009
- [3] A. Boudries, «Maintenance de la Connectivité dans les Réseaux Ad hoc sans fil »,thèse de doctorat en sciences option réseaux et système distribués, université Ferhat Abbas de Setif 1, Algérie,2014.
- [4] A. Delye, V. Gauthier, M. Marot, and M. Becker. "Etat de l'art sur les réseaux de capteurs". Rapport de Recherche INT N-05001RST GET-INT, UMR5157 SAMOVAR, Institut National des Télécommunications, Evry, France, 2005.
- [5] A. Hamzi, « Plateforme pour l'aide à la conception et à la simulation des réseaux sans fil », Mémoire de magister en informatique, Institut National de formation en Informatique, 2007
- [6] A. Jorio, « Le Clustering basé sur la Classification Spectrale pour l'Optimisation D'Energie dans les Réseaux de Capteurs Sans Fil Homogènes », Thèse de doctorat en informatique et télécommunication, Université MOHAMMED V, Rabat, Maroc, 2015.
- [7] A. Kherrab, I. Mehdad, « Sécurisation du protocole de routage AORP dans les réseaux de capteurs sans fil », Mémoire d'ingénieur, Ecole nationale Supérieur d'Informatique, 2010.
- [8] A. Manjeshwar, Dharma P. Agrawal “APTEEN : A Hybrid Protocol for Efficient Routing and Comprehensive Information Retrieval in Wireless Sensor Networks” Proceedings of the International Parallel and Distributed Processing Symposium (IPDPS.02) 2002 IEEE.
- [9] A.Milenkovic et al, « Wireless sensor network for personal health monitoring : Issues and an implementation », Elsevier, 2006.
- [10] A-B. Garcia-Hernando, J-F. Martinez-Ortega, J-M. Lopez- Navarro, A. Prayati, and L. Redondo-López, « Problem Solving for Wireless Sensor Networks », Series : Computer Communications and Networks, ISBN-10 : 1848002025, ISBN-13 : 978-1848002029, Springer, 2008.

-
- [11] C. Intanagonwiwat, R. Govindan, and D. Estrin. «Directed Diffusion : a scalable and robust communication paradigm for sensor networks», In MOBICOM, pp 56–67, 2000.
- [12] C. Laborde, F. Delgado, « Réseaux de capteur sans fil : Configuration et test de connectivité IP », Master de technologie de l’Internet 1ère année, Université de pau et des pays de l’Adour, France.
- [13] C. Schurgers, M.B. Srivastava, « Energy efficient routing in wireless sensor networks, in the MILCOM Proceedings on Communications for Network-Centric Operations : Creating the Information Force », McLean, VA, 2001.
- [14] D. Benchaira, A Bencheikh, «Sécurité de la dissémination de données dans un réseau de capteurs sans fil : cas du protocole Tiny diffusion », Mémoire d’ingénieur, Ecole nationale Supérieure d’Informatique ESI, Alger, Algérie, 2009.
- [15] D. Braginsky, D. Estrin, «Rumor Routing Algorithm for Sensor Networks. in the Proceedings of the First Workshop on Sensor Networks and Applications (WSNA) », Atlanta, GA, 2002.
- [16] D. Niculescu, «Communication paradigms for networks», Communications Magazine, IEEE 43, N° 3, 116–122, 2005.
- [17] Dj. E Boubiche, «Protocole de routage pour les réseaux de capteurs sans fil», Mémoire de magistère, Université de l’Hadj Lakhdar, Batna, Algérie, 2008.
- [18] E. Damou, L. Mounier, « Simulation d’un réseau de capteur sans fil avec TinyOS », VERIMAG.
- [19] E. H. Callaway. « Wireless sensor networks : Architectures and protocols », Crc press Edition, 2004.
- [20] Equipe de Get 2005 Capt’Ad-hoc. « Sensor networks : State of the art ». Technical Report, Telecom Paris, ENST Br, INT, INRIA, 2006.
- [21] E.R. Albu, « Architecture de communication pour les réseaux d’instrumentation sans fil »,Thèse de doctorat, Université Toulouse, France, 2011.
- [22] E. Y. Song, K. Lee, Understanding IEEE 1451 - Networked smart transducer interface standard, IEEE Instrumentation & Measurement Magazine, Vol.11, No. 2, pp.11-17, 2008.
- [23] F. Abdelfatah, « développement d’une bobliothèque de capteurs »,rapport de Master en informatique, Université de Montpellier 2,France,2008.
- [24] Guihai Chen, Chengfa Li, Mao Ye , Jie Wu, « An unequal cluster-based routing protocol in wireless sensor networks », Wireless Netw 2009 , DOI 10.1007/s11276-007-0035-8.
- [25] G. Pujolle, Les Réseaux, 5ème edition, 2006, ISBN : 2-212-11987-9.
- [26] H. Alastrista, S. Aliaga, K. Gouaich, J. Mathieu, « Implémentation de protocole sur une plateforme de réseaux de capteur sans fil », TER master 1 informatique, Université de Montpellier II, France, 2008.

-
- [27] H. Hadjammar, N. Doufene, « Routage dans les réseaux de capteurs : optimisation du protocole Directed Diffusion », Projet de fin d'étude, Institut National de formation en Informatique INI, Algérie, 2006.
- [28] H. Zhang, H. Shen, "Balancing energy consumption to maximize network lifetime in data-gathering sensor networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 20, N° 10, pp 1526-1539, Oct. 2009.
- [29] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. I. Cayirci. « A Survey on sensor networks ». *IEEE Communications Magazine*, Vol. 40, No 8, pp 102-116, August 2002.
- [30] I. Mansour, « Contribution à la sécurité des communications des réseaux de capteurs sans fil », Thèse de doctorat en informatique, Université Blaise Pascal Clermont II, France, 2013.
- [31] Institute of Electrical and Electronics Engineers, IEEE Std 802.15.1-2005, Part 15.1 : Wireless medium Access control (MAC) and physical layer (PHY) specifications for Wireless personal area networks (WPANs), 4 June 2005.
- [32] Institute of Electrical and Electronics Engineers, IEEE Std 802.15.4-2006, Part 15.4 : Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs), 8 September 2006.
- [33] I.S. AlShawi, Y. Lianshan, P. Wei, L. Bin L., « Lifetime Enhancement in Wireless Sensor Networks Using Fuzzy Approach and A-Star Algorithm », *Sensors Journal, IEEE* , vol 12, N°10, pp 3010,3018, Oct. 2012.
- [34] J. A. Stankovic, « Wireless Sensor Networks », Charlottesville, Virginia 22904, Université de Virginie, USA, 2006.
- [35] J. Kaur and Varsha Sahni, « Survey on Hierarchical Cluster Routing Protocols of WSN », *International Journal of Computer Applications* (0975 – 8887) Volume 130 – No.17, Shahpur, Jalandhar, India , November 2015.
- [36] : J. Shyan Lee, Yu-Wei Su, and Cung-Chou Shen, « A Comparative Study of Wireless Protocols : Bluetooth, UWB, ZigBee, and Wi-Fi. In Proceeding » the 33rd Annual Conference of the IEEE Industrial Electronics Society (IECON), Taipei, Taiwan, pp. 46-51 , Novembre 2007.
- [37] K. Akkaya, and M. Younis, "A Survey on Routing Protocols for Wireless Sensor Networks". *Journal of Ad Hoc Networks*, Vol. 3, No. 3, May 2005, pp. 325-349.
- [38] Kim, K.-H. and Shin, K.G. (2006) On accurate measurement of link quality in multi-hop wireless mesh networks. *Proceedings of the 12th annual international conference on Mobile computing and networking*. ACM, Los Angeles, CA, USA, 38-49.
- [39] K. Heurtefeux, F. Maraninchi et F. Valois, « AreaCast : une communication par zone dans les réseaux de capteurs sans fil ».

-
- [40] K.Lorincz, B. Chen, G. Werner Challen, A.R. Chowdhury, S. Patel, P. Bonato, and M. Welsh, Mercury : A wearable sensor network Platform for High-fidelity motion analysis. In Proceedings of the 7th ACM Conférence on Embedded Networked Sensor Systems, SenSys '09, pages 183, 196, New York, NY, USA, 2009. ACM.
- [41] L.Chamek, « Réseaux Ad-Hoc Mobile :Mobile Ad-Hoc Network »,cours ,Université Mouloud Mammeri de Tizi-Ouzou, Algérie,2015.
- [42] L. Jing, F. Liu, « Energy Saving Routing Algorithm Based on SPIN Protocol in WSN », 978-1-61284-881-5/11 @2011 IEEE.
- [43] L.Khelladi, N. Badache, « Les réseaux de capteurs : état de l'art », Rapport de recherche, Université des Sciences et de la technologie Houari Boumediene de BabEzzouar, Algérie, 2004.
- [44] L. Khelladi, Nadjib Badache, « Improving Directed Diffusion With Power-Aware Topology Control For Adaptation to High Density », LOCALGOS'08 workshop, in conjunction with The 4th IEEE/ACM International Conference on Distributed Computing In Sensor SystemS (DCOSS 2008), Algeria, 2008.
- [45] M. Abed, M.Bacha, « Description des comportements d'un réseau de capteurs sans fils à l'aide de SMA », Mémoire d'ingénieur en informatique, Ecole National Supérieur D'informatique d'Alger, Algérie, 2012.
- [46] M.A .Benmouffok, A.R. Ouaz, «Conception d'un protocole de routage hiérarchique pour les réseaux de capteurs sans fil», Mémoire d'ingénieur, Ecole nationale Supérieure d'Informatique ESI, Algérie, 2011.
- [47] M.A. Matin, M.M. Islam, « Overview of Wireless Sensor Network », cours,Institut Teknologi Brunei, Brunei Darussalam and North South University, Dhaka, Bangladesh,2012.
- [48] M. E. Diouri, « Réseau de capteurs Sans fil : Routage et Sécurité », Mémoire d'ingénieur en Informatique, Université de Lyon, France, 2010.
- [49] M.L. Messai, « Sécurité dans les réseaux de capteurs sans fil », Mémoire de Magistère en Informatique, Université Abderrahmane Mira de Bejaia, Algérie, 2008.
- [50] M. Oularbi, S. Kassab, « HAORP (Hierarchical Adaptive Optimized Routing Protocol), Elaboration d'un protocole de routage efficace en énergie pour réseaux de capteurs sans fil », Mémoire d'ingénieur, Ecole nationale Supérieur d'Informatique, 2010.
- [51] M. Sajid, K. Khan, U. Qasim, Z.A. Khaan,S. Tariq, N. Javaid, "A New Linear Cluster Hnadling(LCH)Technique Toward's Energy efficiency in Linear WSNs" 2015 IEEE 29 International Conference on Advanced Information Networking and Applications.
- [52] O. Younis, S. Fahmy. « Heed : a hybrid, energy efficient, distributed clustering approach for ad hoc sensor networks ». IEEE Transactions on Mobile Computing, 3(4) :366, 379, 2004.

- [53] P. Johnson, D. Andrews, « Remote continuous monitoring in the home », *Journal of Télémedecine and Télécare*, vol.2, N° 2, pp 107-113, June 1996.
- [54] P. Khurana, Inderdeep Aulakh Punjab University Chandigarh, India “Wireless Sensor Network Routing Protocols : A Survey” *International Journal of Computer Applications* Volume 75– No.15, August 2013.
- [55] P. Spathis, S. Fdida, Y. Barouni, « Modèle générique pour le routage orienté contenu », Document scientifique, hal-00260342, Laboratoire d’informatique de Paris 6, Université Pierre et Marie Curie, Mars 2008.
- [56] Publié dans : *Avances in Wireless and Optical Communications (RTUWO)*, 2016..Date de la conférence : 3-4 novembre 2016..Date ajoutée à IEEE Xplore : 19 janvier 2017..Renseignements sur l’ISBN : Numéro d’accès INSPEC : 16604419..DOI : 10.1109 / RTUWO.2016.7821862..Editeur : IEEE.
- [57] R.Kacimi, « Techniques de conservation d’énergie pour les réseaux de capteurs sans fil »,Thèse doctorat,Universiré Toulouse , France, 2009.
- [58] R.Jurdak , « Wireless Ad-Hoc and Sensor Networks :A Cross-Layer Design perspective », Univesity College Dublin, Ireland, 2007.
- [59] R. Shah and J. Rabaey, « Energy Aware Routing for Low Energy Ad Hoc Sensor Networks », the Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC), Orlando, FL, March 2002.
- [60] S. Bandyopadhyay, E. J. Coyle. « An energy efficient hierarchical clustering algorithm for wireless sensor networks ». In *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications*, volume 3, pages 1713{1723. IEEE, 2003.
- [61] S.J. Mujumdar, « Prioritized Geographical Routing In Sensor Networks », Thesis Master of science in Electrical Engineering, Vanderbilt University, USA, 2004.
- [62] S.Talbi, L. Zaouche, « Enhanced Unequal Clustering Algorithm for Wireless Sensor Networks », 978-1-5090-0478-2/15 ©2015 IEEE.
- [63] S. K. Singh , M P Singh , and D K Singh Routing Protocols in Wireless Sensor Networks – A Survey” *International Journal of Computer Science & Engineering Survey (IJCSES)* Vol.1, No.2, November 2010.
- [64] S. A-A. Brahami, « Evaluation du Protocole Directed Diffusion dans un réseau de capteurs sans fil », Mémoire Master en Informatique, Université Abou Bekr Belkaid de Tlemcen, Algérie, 2014.
- [65] « Spécification of the Bluetooth System, Core, Version 1.1 », Bluetooth SIG, www.Bluetooth.com.
- [66] S. Bouguer, « Etude et simulation comparative entre les réseaux de capteurs sans fil traditionnels et les réseaux de capteurs véhiculaires », Mémoire d’ingénieur en Télécommunication, Université Abou Bekr Belkaid de Tlemcen, Algérie, 2012.

- [67] S. Sentilles, « Architecture logicielle pour capteurs sans fil en réseau », Mémoire de master, Université de Pau et des Pays de l' Adour, Juin 2006.
- [68] S.Kabou, · A. Belgourari, « Etat de l'art sur les réseaux de capteurs sans fil »,Mémoire de Licence en Informatique, Université Bechar, Algérie, 2010.
- [69] S. Maarouf, S. Ouadah , « Implémentation et évaluation des schémas de routage sur une plateforme réelle de réseaux de capteurs sans fil », Mémoire de Master en informatique, université Abou Bekr Belkaid de Tlemcen, Algérie, 2014.
- [70] S. Moad, « Optimisation de la consommation d'énergie dans les réseaux de capteurs sans fil », Mémoire de Master en informatique, Institut de Formation Supérieur en Informatique et Communication-Rennes 1, France, 2008.
- [71] « The First European TinyOS Technology Exchange (ETTX 2009) », 10-févr-2009. [En ligne]. Disponible sur : <https://sites.google.com/site/ettx2009/>. [Consulté le : 31-mars-2013].
- [72] T. He, J.A. Stankovic, C. Lu, and T. Abdelzaher, « SPEED : a stateless protocol for real-time communication in sensor networks»,International Conference on Distributed Computing Systems, 2003.
- [73] J.N. Al-Karaki and A.E. Kamali. Routing techniques in wireless sensor networks : A survey. *Wireless Communications, IEEE*, 11 no :6 :6 28, Dec 2004.
- [74] « Université de Berkeley ». [En ligne]. Disponible sur : <http://www.berkeley.edu/>. [Consulté le : 31-mars-2013].
- [75] V. Rodoplu, T. H. Ming, « Minimum energy mobile wireless networks ». *IEEE Journal of Selected Areas in Communications*, Vol. 17, No 8, 1999, pp 1333-1344.
- [76] W. Bechkit, « Un nouveau protocole de routage avec conservation d'énergie dans les réseaux de capteurs sans fil », Mémoire d'ingénieur, Ecole nationale Supérieure d'Informatique ESI D d'Alger, Algérie, 2009.
- [77] W. B. Heinzelman, A.P. Chandrakasan, and H. Balakrishnan. « Energy-efficient communication protocol for wireless microsensor networks. In Proceedings » the 33rd Hawaii International Conference on System Sciences (HICSS-33), January 2000.
- [78] W.B. Heinzelman, A.P. Chandrakasan, and H. Balakrishnan. « An application specific protocol architecture for wireless microsensor networks ». *IEEE Transactions on Wireless Communications*, 1, N° 4 :660, 670, October 2002.
- [79] W-B. Heinzelman, A.P. Chandrakasan, H. Balakrishnan, Energy-Efficient Communication Protocol for Wireless Microsensor Networks, International Conférence on Systems Science, vol.8, pp. 8020, January 2000.
- [80] W.Heinzelman, «Application-Specific Protocol Architectures for Wireless Network», *IEEE Transactions on Wireless Communications*, Massachusetts Institute of Technology, June 2000.

- [81] Y.Yacine, « Minimisation d'énergie dans un réseau de capteurs », Mémoire Magistère en Informatique, Université Mouloud Mammeri de Tizi-Ouzou, Algérie, 2012.
- [82] Y. Challal, « Réseaux de capteurs sans fils », Version 1, <http://creativecommons.org/licenses/by-nc/2.0/fr/>, 17/11/2008.
- [83] Y. Benabbassi, « Application de la redondance pur la surveillance par réseau de capteurs sans fil : cas réseau des capteurs image sans fil », Thèse de doctorat en informatique, Université d'Oran, Algérie, 2014.
- [84] Y. Romdhane, « Evaluation des performances des protocoles S-MAC et Directed Diffusion dans les réseaux de capteurs », Projet de fin d'études, Ecole Supérieure des Communications de Tunis (Sup'Com), Tunisie, 2007.
- [85] Y. Sarr, « Métriques de routage dans les réseaux maillés sans fil », Mémoire de Maîtrise en sciences (M.Sc.) d'Informatique, Université Montréal, Canada, 2011.
- [86] Y. Xu, J. Heidemann, D. Estrin, «Geography-informed Energy Conservation for Ad-hoc Routing », In Proceedings of the Seventh Annual ACM/IEEE International Conference on Mobile Computing and Networking, Pages : 70-84. 2001.
- [87] Y. Yu, D. Estrin, and R. Govindan, «Geographical and Energy-Aware Routing : A Recursive Data Dissemination Protocol for Wireless Sensor Networks», UCLA Computer Science Department Technical Report, UCLA-CSD TR-01-0023, May 2001.
- [88] : Z. Yong, Qing Pei. « A energy efficient clustering routing algorithm based on distance and residual energy for wireless sensor networks ». Procedia Engineering, International Workshop on Information and Electronics Engineering.
- [89] <http://compilers.cs.ucla.edu/avrrora>, site d'informations sur le simulateur Avrora, Novembre 2005.
- [90] <http://www.prism.uvsq.fr/~mogue/M2IR/Biblio/Sensor/Sensor%20%20Net/french%20doc/>.