

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université A/Mira de Béjaïa

Faculté des Sciences Exactes

Département d'Informatique



Mémoire de fin d'études

en vue de l'obtention du diplôme de master recherche en Informatique

Option : Réseaux et Systèmes Distribués

Thème

Le routage hiérarchique à basse consommation énergétique dans les réseaux de capteurs sans fil

Présenté par :

✓ *M^{lle}* BOUZIDA Sabrina.

✓ *M^{lle}* NADI Amel.

Proposé et encadré par :

✓ M. MIR Foudil.

Devant le jury composé de :

✓ Président : M. KHANOUCHE Med-Essaid.

✓ Examineur : M. LARBI Ali.

✓ Examineur : M^{me}. BOUTRID Samia.

Promotion 2012/2013

TABLE DES MATIÈRES

Table des Matières	III
Liste des figures	V
Introduction générale	1
1 Généralités sur les réseaux de capteurs sans fil	3
1.1 Introduction	3
1.2 Les réseaux sans fil	3
1.3 Les réseaux ad hoc	4
1.3.1 Définition	4
1.3.2 Caractéristiques et contraintes des réseaux ad hoc	5
1.3.3 Avantage	6
1.3.4 Inconvénients	6
1.4 Les réseaux de capteurs sans fil	6
1.4.1 Définition d'un capteur	6
1.4.2 Architecture d'un noeud de capteur	7
1.4.3 Types de capteurs	8
1.4.4 Définition d'un réseau de capteurs sans fil	9
1.4.5 Caractéristiques des réseaux de capteurs sans fil	9
1.4.6 Les domaines d'applications des réseaux de capteurs sans fil	10
1.4.7 La communication dans les réseaux de capteur sans fil	13
1.4.8 Les types de communications dans les réseaux de capteurs sans fil	14
1.4.9 Défis des RCSFs	15
1.4.10 Facteurs et contraintes de conception des RCSF	16
1.4.11 La différence entre les réseaux de capteurs et Ad hoc	17

1.4.12	Les problèmes liés aux réseaux de capteurs sans fil	18
1.5	Conclusion	18
2	Le routage dans les réseaux de capteurs sans fil	19
2.1	Introduction	19
2.2	Les défis de routage dans les réseaux de capteurs sans fil	19
2.2.1	Le déploiement des capteurs	19
2.2.2	Agrégation des données	20
2.2.3	Dynamacité des réseaux	20
2.2.4	Modèle de livraison de données	20
2.2.5	Hétérogénéité Noeuds/liens	21
2.2.6	Qualité de service (QoS)	21
2.3	Techniques de minimisation de la consommation d'énergie	21
2.4	Les approches de routage dans les réseaux de capteurs sans fil	23
2.4.1	Classification selon la structure du réseau	23
2.4.2	Classification selon les fonctions du protocole	31
2.4.3	Classification selon l'établissement de la route	35
2.4.4	Classification selon l'initiateur de la communication	37
2.5	Les métriques de mesure de l'efficacité des protocoles de routage dans les RCSFs	37
2.6	Tableau comparatif des protocoles de routage	39
2.7	Conclusion	40
3	Etat de l'art : les protocoles de routage hiérarchique dans les réseaux de capteurs sans fil	41
3.1	Introduction	41
3.2	Definition de la notion de clustering	41
3.3	Avantage de routage hiérarchique	43
3.4	Contraintes et facteurs de conception d'un protocole de routage hiérarchique . .	43
3.5	Taxonomie d'attribut de clustering	45
3.6	Les protocoles de routage hiérarchique	48
3.6.1	PEGASIS(Power-Efficient Gathering in Sensor Information Systems) . .	48
3.6.2	EEPSC(Energy Efficient Protocol with Static Clustering for wireless sensor network)	49
3.6.3	TEEN (Threshold sensitive Energy Efficient sensor Network protocol) .	50
3.6.4	APTEEN (Adaptive Threshold-sensitive Energy Efficient sensor Network protocol)	51
3.6.5	PEGASIS Hiérarchique	52
3.6.6	PEGASIS HM(Hop-Multiply)	53
3.6.7	DECSA(Distance-Energy Cluster Structure Algorithm) :	55

3.7	Tableau comparatif pour les protocoles de clustering	57
3.8	Conclusion	58
4	Contribution : DECSA-G (Distance-Energy Cluster Structure Algorithm based on Grid)	59
4.1	Introduction	59
4.2	Consommation d'énergie dans les réseaux de capteurs sans fil	60
4.2.1	Energie de capture	60
4.2.2	Energie de traitement	60
4.2.3	Energie de Communication	60
4.3	Modele de consommation d'énergie	60
4.4	Description de la proposition	61
4.4.1	Hypothèses	62
4.4.2	Le fonctionnement du protocole DECSA-G	62
4.5	Simulation et analyse des performances du protocole DECSA-G	70
4.5.1	Choix de l'environnement de developpement	70
4.5.2	Java	70
4.5.3	L'environnement de simulation :	71
4.5.4	Métriques considérées	71
4.5.5	Caracteristique du simulateur	72
4.5.6	Evaluation de performances	72
4.6	Conclusion	76
	Conclusion générale	77
	Bibliographie	78

TABLE DES FIGURES

1.1 Réseaux ad hoc[2].	4
1.2 Classification des réseaux [3].	5
1.3 Noeud de capteur [6].	7
1.4 Les composants d'un noeud de capteur.	8
1.5 Quelques capteurs existants [6].	8
1.6 Architecture d'un RCSF.	9
1.7 Domaine militaire.	10
1.8 Découverte de catastrophes naturelles.	11
1.9 Domaine agriculture.	11
1.10 Domaine médicale.	12
1.11 Couche des RCSFs[3].	13
2.1 Les techniques de conservation d'énergie[4].	22
2.2 Les différentes approches de routage dans les RCSFs.	23
2.3 Le modèle LEACH.	27
2.4 Protocole GAF [16].	31
2.5 Protocole SPIN[16].	34
3.1 Le protocole PEGASIS.	48
3.2 Le protocole EEPSC.	50
3.3 Le principe de PEGASIS Hiérarchique.	53
3.4 Algorithme de recherche du prochain saut.	54
3.5 Organigramme du PEGASIS HM.	54
3.6 Le protocole PEGASIS HM.	55
3.7 Le protocole DECSA.	57
4.1 Le modèle de consommation d'énergie.	61

4.2	Protocole DECSA-G : Formation des niveaux.	63
4.3	Protocole DECSA-G : Formation des clusters.	64
4.4	L'organigramme de phase d'organisation de groupes.	64
4.5	L'algorithme de phase d'organisation de groupes.	65
4.6	Election des clusters Heads.	67
4.7	Protocole DECSA-G : Election des clusters Heads.	67
4.8	L'algorithme de phase d'élection de CH et CH-p	68
4.9	Le fonctionnement de TDMA.	68
4.10	Protocol DECSA-G : Communication interne.	69
4.11	Protocole DECSA-G : Communication externe.	69
4.12	Interface du simulateur.	72
4.13	Moyenne d'énergie résiduelle.	73
4.14	Le premier capteur défaillant en fonction de nombre de noeuds.	74
4.15	La défaillance du 25% de noeuds de capteur en fonction de nombre de noeuds.	75
4.16	La défaillance du 50% de noeuds de capteur en fonction de nombre de noeuds.	75
4.17	La défaillance du dernier capteur en fonction de nombre de noeuds.	76

INTRODUCTION GÉNÉRALE

Grâce au progrès fait dans le domaine de la miniaturisation des systèmes de microélectromécanique MEMS (Miniaturization Electronic Mechanic Systems) et dans le marché des réseaux et des applications sans fil, s'est créée une nouvelle branche de réseaux mobile afin d'offrir des solutions économiquement intéressantes pour la surveillance à distance et le traitement des données dans des environnements complexes : les réseaux de capteurs sans fil (Wireless Sensor networks).

Les réseaux de capteurs sans fil (Wireless Sensor Networks ; WSN) sont considérés comme un type spécial de réseaux ad hoc, composés d'un grand nombre de capteurs matériellement petits, et placés généralement près des objets auxquels ils s'intéressent dans les environnements où ils sont déployés. Ces capteurs sont capables de récolter, traiter et acheminer les données environnementales de la région surveillée d'une manière autonome, vers des stations de collecte appelées noeuds puits ou stations de base.

Dans les réseaux de capteurs, la consommation d'énergie est une contrainte très cruciale puisque généralement les capteurs sont déployés dans des zones inaccessibles. Ainsi, il est difficile, voire même impossible, de remplacer les batteries après leurs épuisement. De ce fait, la consommation d'énergie au niveau des capteurs a une grande influence sur la durée de vie du réseau en entier. Il est donc impératif de mettre en place des protocoles de routage efficaces en énergie, et qui prennent en compte les contraintes imposées par ces capteurs. La majorité des travaux de recherche menés actuellement se concentrent principalement 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.

L'idée général de notre proposition est basée sur le clusering statique c'est à dire de fixer le nombre et la taille de clusters durant tout le processus du routage, ainsi de réaliser une

communication multi-saut afin d'acheminer les données vers la station de base.

Dans ce travail, nous nous intéressons à la couche réseau, où beaucoup de protocoles de routage ont été proposés pour conserver au maximum l'énergie du réseau. L'objectif de notre travail est de proposer un nouveau protocole de routages hiérarchique à basse consommation d'énergie, dans le but de résoudre le problème d'acheminement de données entre les noeuds du réseau avec la prise en compte de la contrainte énergétique.

Ce mémoire est organisé en quatre chapitres :

Le premier présente des généralités sur les réseaux de capteurs sans fil : leurs architectures, leurs caractéristiques, leurs domaines d'applications, leurs objectifs et les contraintes de conception d'un tel réseau.

Dans le deuxième chapitre, nous décrivons les défis du routage dans les réseaux de capteurs sans fil, quelques techniques de minimisation de la consommation d'énergie. Enfin, une classification et une comparaison de quelques protocoles de routage sont présentées.

Le troisième chapitre définit la notion de clustering dans les réseaux de capteurs sans fil, les avantages du routage hiérarchique, les contraintes et facteurs de conception d'un protocole de routage hiérarchique. Enfin, la présentation de quelques protocoles de routage hiérarchique et donner une comparaison entre eux.

Dans la première partie du quatrième chapitre, nous donnons une description détaillée du protocole que nous avons proposé. La deuxième partie est consacrée à réaliser une plateforme d'aide à la simulation des réseaux de capteurs sans fil et faire une analyse des résultats obtenus.

Enfin, notre mémoire s'achève par une conclusion générale résumant les points qui ont été abordés ainsi que des perspectives de recherches dans le domaine en question.

CHAPITRE 1

GÉNÉRALITÉS SUR LES RÉSEAUX DE CAPTEURS SANS FIL

1.1 Introduction

Les réseaux de capteurs forment une nouvelle génération de réseaux aux propriétés spécifiques [2]. ces réseaux sont constitués de plusieurs noeuds qui sont des micro- capteurs capables de récolter et de transmettre des données de manière autonome. Les noeuds capteurs composant le réseau possèdent généralement de faibles capacités de calcul, de mémoire et d'énergie. Ces contraintes matérielles ont influencé une grande partie des problématiques de recherche du domaine tel que le routage.

Dans ce chapitre, nous présentons tout d'abord brièvement les réseaux sans fil et Ad hoc. En outre, nous introduisons une description synthétique des réseaux de capteurs sans fil en présentant leurs évolutions, leurs architectures, leurs caractéristiques, et leurs domaines d'applications variés.

1.2 Les réseaux sans fil

Un réseau sans fil est comme son nom l'indique est un réseau dont lequel au moins deux périphériques (ordinateur, PDA, imprimantes, routeur) peuvent se communiquer sans liaison filaire. Les réseaux sans fil ont recours à des ondes radioélectroniques (radio et infrarouges) en lieu et place des câbles habituels. Il existe plusieurs technologies se distinguant d'une part par la fréquence d'émission utilisée ainsi que le débit et la portée des transmissions [1]. Les réseaux sans fil sont en plein développement du fait de la flexibilité de leur interface, qui permet à un

utilisateur de changer de place tout en restant connecté. Les communications entre équipements terminaux peuvent s'effectuer directement ou par le biais de stations de base, appelées points d'accès, ou AP (Access Point). Les communications entre points d'accès peuvent être hertziennes ou par câble. Les débits de ces réseaux se comptent en dizaines de mégabits par seconde [2].

1.3 Les réseaux ad hoc

1.3.1 Définition

Une autre grande catégorie des réseaux sans fil est appelée les réseaux ad-hoc, dans lesquels l'infrastructure n'est composée que des stations elles-mêmes. Ces dernières acceptent de jouer le rôle de routeur pour permettre le passage de l'information d'un terminal vers un autre, sans que ces terminaux soient reliés directement [2]. Contrairement aux apparences, les réseaux ad-hoc datent de plusieurs dizaines d'années. Ils visent à réaliser un environnement de communication qui se déploie sans autre infrastructure que les mobiles eux-mêmes. En d'autres termes, les mobiles peuvent jouer le rôle de passerelle pour permettre une communication d'un mobile à un autre. Deux mobiles trop éloignés l'un de l'autre pour communiquer directement peuvent trouver un mobile intermédiaire capable de jouer le rôle de relais [2].

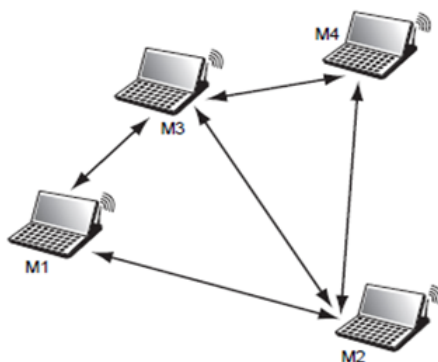


FIGURE 1.1 – Réseaux ad hoc[2].

La figure 1.2 résume, de façon hiérarchique, les types des réseaux avec leurs noms (les feuilles de la hiérarchie). Nous citons un exemple pour chaque type de réseau :

- **LAN** (Local Area Network) et **WAN** (Wired Area Network) : plusieurs équipements (ordinateurs, imprimantes, serveurs, etc.) connectés ensemble via des câbles.
- Ad-hoc filaire : deux ordinateurs connectés via un câble.
- **WLAN** (Wireless Local Area Network) : le réseau cellulaire (**GSM**) composé de téléphones mobiles sans fil et des stations fixes.

- **MANET** (Mobile Ad-hoc NETwork) : plusieurs équipements mobiles connectés via le radio.
- Ad-hoc non filaire : plusieurs équipements fixes connectés via le radio [3].

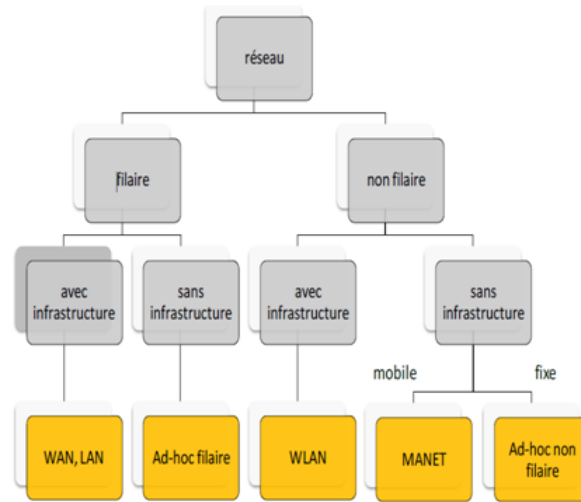


FIGURE 1.2 – Classification des réseaux [3].

1.3.2 Caractéristiques et contraintes des réseaux ad hoc

Les réseaux ad hoc présentent plusieurs avantages. Outre la mobilité qui constitue l'avantage principal, le coût d'installation et de déploiement peut également être un atout, puisqu'un peu d'électronique peut compenser un câblage onéreux. De par la nature du canal radio, un certain nombre de problèmes se posent qui ne trouvent pas d'équivalent dans le monde filaire. Les principales caractéristiques des réseaux ad hoc et les contraintes induites par ces propriétés sont comme suit :

- **La mobilité** : C'est l'atout principal de ce type de réseau mais ceci constitue, aussi, une contrainte majeure qu'il est nécessaire de la prendre en compte dans les protocoles utilisés.
- **Le multi saut (Multihopping)** : Un réseau ad hoc est un réseau multi saut, dans lequel, un chemin allant d'une source vers une destination traverse plusieurs autres noeuds.
- **L'auto-organisation** : Un réseau ad hoc doit déterminer de façon autonome ses propres paramètres de configuration : adressage des noeuds, routage des paquets, contrôle de la puissance d'émission, identification de la position d'un noeud, etc.
- **La conservation d'énergie** : La plupart des noeuds d'un réseau ad hoc (exp. Laptops, PDAs, Capteurs, etc.) possèdent une autonomie d'énergie limitée et de ce fait, les protocoles de communication (MAC, routage, etc.) utilisés doivent minimiser la consommation d'énergie au niveau de chaque noeud.

- **Les interférences** : Elles peuvent être liées à des transmissions simultanées ou à des noeuds cachés. L'environnement lui-même peut également produire des bruits parasites induits par certains équipements électriques ou électroniques (moteurs, etc.) qui interfèrent avec les communications.
- **L'atténuation radio** : Comme tout réseau sans fil, la portée est limitée et les obstacles rencontrés atténuent fortement le signal émis.
- **Le débit** : Les contraintes physiques liées aux réseaux ad hoc rendent ces derniers moins performants que les réseaux filaires.
- **La puissance de signal** : Non seulement elle est rapidement atténuée avec la distance, mais elle est également limitée par des réglementations très strictes. Un émetteur ne peut pas dépasser une certaine puissance à l'émission.
- **La sécurité** : Un réseau sans fil est potentiellement utilisable par toute personne se trouvant dans la zone de couverture radio des stations émettrices. Sans mesure de sécurité, il est assez facile d'écouter le trafic, de rejouer les transmissions, de manipuler les en-têtes des paquets ou de corrompre le signal émis (brouillage). Par conséquent, des mécanismes de sécurité et de confidentialité doivent être mis en oeuvre [4].

1.3.3 Avantage

Les avantages des réseaux ad-hoc sont leurs extensions très simples, leur couverture physique et leur coût, etc [2].

1.3.4 Inconvénients

Les réseaux ad-hoc posent de nombreux problèmes du fait de la mobilité de tous les équipements. Le principal d'entre eux est le routage nécessaire pour transférer les paquets d'un point à un autre point du réseau. L'un des objectifs du groupe MANET (Mobile Ad-hoc NETwork) est de proposer une solution à ce problème. Pour le moment, quatre grandes propositions ont vu le jour, deux de type réactif et deux de type proactif. Parmi les autres problèmes, nous retrouvons la sécurité, la qualité de service et la gestion de la mobilité en cours de communication [2].

1.4 Les réseaux de capteurs sans fil

1.4.1 Définition d'un capteur

Les capteurs sont des dispositifs de taille extrêmement réduite avec des ressources très limitées, autonomes, capables d'effectuer des mesures simples sur leurs environnements. Ces mesures peuvent tant concerner la température, la pression de la lumière ou le son et de les

transformer en données numériques ; afin de les communiquer par ondes radio à travers le réseau vers un ou plusieurs points de collecte appelés les stations de base (SB) [5].

1.4.2 Architecture d'un noeud de capteur



FIGURE 1.3 – Noeud de capteur [6].

Un noeud de capteur est composé essentiellement comme suit :

- **L'unité d'acquisition/ de captage (*Sensing unit*)** : composée d'un capteur qui obtient des mesures sur les paramètres environnementaux et d'un Convertisseur Analogique/Numérique appelé ADC (Analog to Digital Converter) qui convertit l'information relevée et la transmet à l'unité de traitement [6].
- **L'unité de traitement (*Processing unit*)** : composée d'un processeur et d'une mémoire intégrant un système d'exploitation spécifique. Cette unité possède deux interfaces, une interface pour l'unité d'acquisition et une interface pour l'unité de communication. Elle acquiert les informations en provenance de l'unité d'acquisition et les envoie à l'unité de communication. Cette unité est chargée aussi d'exécuter les protocoles de communications qui permettent de faire collaborer le capteur avec d'autres capteurs. Elle peut aussi analyser les données captées [6].
- **L'unité de transmission (*Transceiver unit*)** : unité responsable de toutes les émissions et réceptions de données via un support de communication radio. Elle peut être de type optique, ou de type radiofréquence [6].
- **L'unité d'énergie (*Power unit*)** : un capteur est muni d'une batterie pour alimenter tous ses composants. Cependant, à cause de sa taille réduite, la batterie dont il dispose est limitée et généralement irremplaçable. Pour cela, l'énergie est la ressource la plus précieuse

puisqu'elle influe directement sur la durée de vie des capteurs. Il existe des capteurs qui sont dotés d'autres composants additionnels comme le système de positionnement GPS (Global Positioning System) et un mobilisateur lui permettant le déplacement [6].

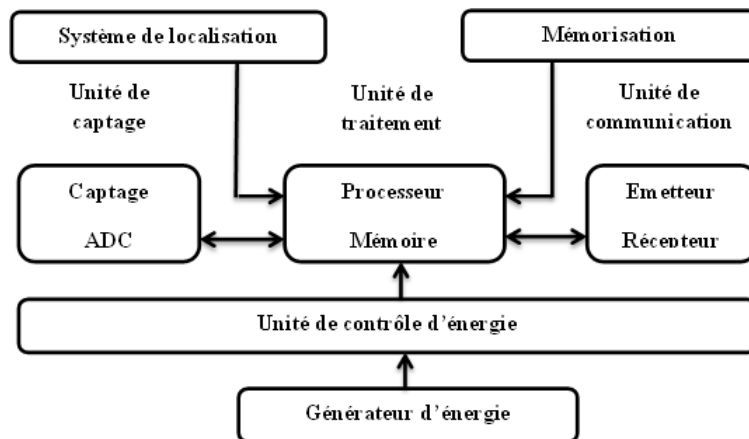


FIGURE 1.4 – Les composants d'un nœud de capteur.

1.4.3 Types de capteurs

Il existe actuellement un grand nombre de capteurs, avec des fonctionnalités diverses et variées. La plupart des capteurs dépendent de l'application pour lesquels ils ont été conçus (capteurs aquatiques, sous-terrain, etc). Il est plus intéressant de décrire les capteurs les plus utilisés et leur évolution au cours du temps. La figure 1.5 illustre quelques nouveaux types des capteurs existants sur le marché [6].

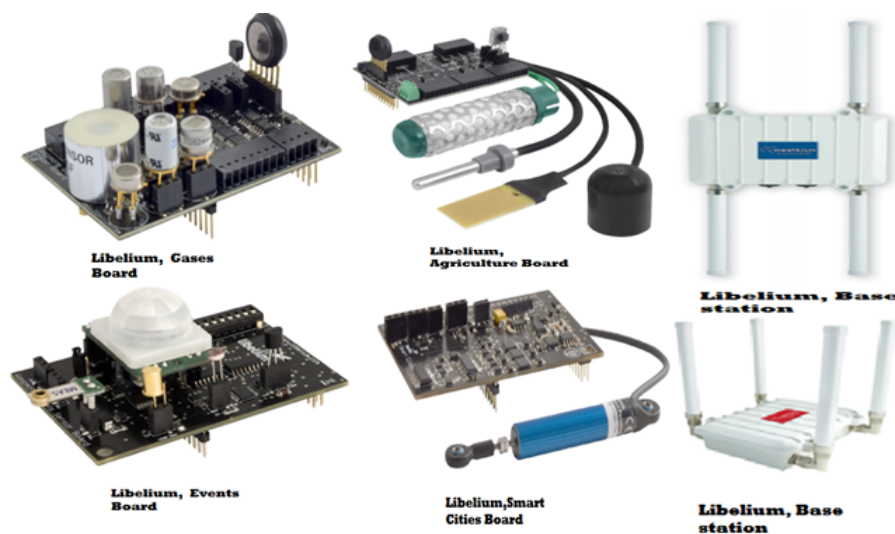


FIGURE 1.5 – Quelques capteurs existants [6].

1.4.4 Définition d'un réseau de capteurs sans fil

Un réseau de capteurs sans fil se définit par une/des station(s) de base(s), satellite, un (des) utilisateur(s), et un ensemble de capteurs connectés entre eux, où chaque capteur étant muni d'un émetteur-récepteur. Les réseaux de capteurs forment une nouvelle génération de réseaux aux propriétés spécifiques, qui n'entrent pas dans le cadre des architectures classiques [2]. Un exemple de réseaux de capteurs est fourni dans la Figure 1.4 : les capteurs sont déployés d'une manière aléatoire dans une zone d'intérêt, et une station de base, située à l'extrémité de cette zone, est chargée de récupérer les données collectées par les capteurs. Lorsqu'un capteur détecte un événement pertinent, un message d'alerte est envoyé à la station de base par le biais d'une communication entre les capteurs. Les données collectées sont traitées et analysées par des machines puissantes [3].

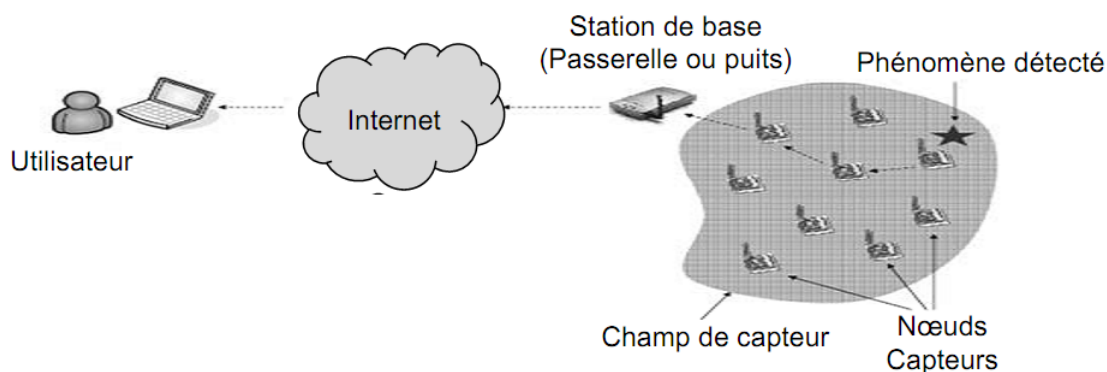


FIGURE 1.6 – Architecture d'un RCSF.

1.4.5 Caractéristiques des réseaux de capteurs sans fil

Un réseau de capteurs présente les caractéristiques suivantes : Réseaux Ad-hoc et réseaux de capteurs :

- **Absence d'infrastructure** : les réseaux Ad-hoc en général, et les réseaux de capteurs en particulier se distinguent des autres réseaux par la propriété d'absence d'infrastructure préexistante et de tout genre d'administration centralisée.
- **Taille importante** : un réseau de capteurs peut contenir des milliers de noeuds.
- **Interférences** : les liens radio ne sont pas isolés, deux transmissions simultanées sur une même fréquence, ou utilisant des fréquences proches, peuvent interférer.
- **Topologie dynamique** : les capteurs peuvent être attachés à des objets mobiles qui se déplacent d'une façon libre et arbitraire rendant ainsi la topologie du réseau fréquemment changeante.

- **Sécurité physique limitée** : les réseaux de capteurs sans fil sont plus touchés par le paramètre de sécurité que les réseaux filaires classiques. Cela se justifie par les contraintes et limitations physiques qui font que le contrôle des données transférées doit être minimisé.
- **Bande passante limitée** : une des caractéristiques primordiales des réseaux basés sur la communication sans fil est l'utilisation d'un médium de communication partagé. Ce partage fait que la bande passante réservée à un noeud est limitée.
- **Contrainte d'énergie, de stockage et de calcul** : la caractéristique la plus critique dans les réseaux de capteurs est la modestie de ses ressources énergétiques car chaque capteur du réseau possède de faibles ressources en termes d'énergie (batterie). Afin de prolonger la durée de vie du réseau, une minimisation des dépenses énergétiques est exigée chez chaque noeud. Ainsi, la capacité de stockage et la puissance de calcul sont limitées dans un capteur [3].

1.4.6 Les domaines d'applications des réseaux de capteurs sans fil

- **Applications militaires** : Le faible coût et le déploiement rapide sont des caractéristiques qui ont rendu les réseaux de capteurs efficaces pour les applications militaires. Plusieurs projets ont été lancés pour aider les unités militaires dans un champ de bataille et protéger les villes contre des attaques, telles que les menaces terroristes [5].



FIGURE 1.7 – Domaine militaire.

- **Découvertes de catastrophes naturelles** : Le contrôle des paramètres environnementaux par les réseaux de capteurs peut donner naissance à plusieurs applications. Par exemple, le déploiement des thermo-capteurs dans une forêt peut aider à détecter un éventuel début de feu et par suite faciliter la lutte contre les feux de forêt avant leur propagation. Le déploiement des capteurs chimiques dans les milieux urbains peut aider à détecter la pollution et analyser la qualité d'air. De même leur déploiement dans les sites industriels empêche les risques industriels tels que la fuite de produits toxiques (gaz, produits chimiques, éléments radioactifs, pétrole, etc.) [5].



FIGURE 1.8 – Découverte de catastrophes naturelles.

- **Détection d'intrusions** : En plaçant, à différents points stratégiques, des capteurs, on peut ainsi prévenir des cambriolages ou des passages de gibier sur une voie de chemin de fer (par exemple) sans avoir à recourir à de coûteux dispositifs de surveillance vidéo [1].
- **Applications métier** : On pourrait imaginer devoir stocker des denrées nécessitant un certain taux d'humidité et une certaine température (min ou max). Dans ces applications, le réseau doit pouvoir collecter ces différentes [1].
- **Agriculture** : Dans le domaine de l'agriculture, les capteurs peuvent être utilisés pour réagir convenablement aux changements climatiques par exemple le processus d'irrigation lors de la détection de zones sèches dans un champ agricole. Cette expérimentation a été réalisée par Intel Research Laboratory and Agriculture and Agri-Food Canada sur une vigne à British Columbia [5].



FIGURE 1.9 – Domaine agriculture.

- **Surveillance médicale** : Dans le domaine de la médecine, les réseaux de capteurs peuvent être utilisés pour assurer une surveillance permanente des organes vitaux de l'être humain grâce à des micro-capteurs qui pourront être avalés ou implantés sous la peau (surveillance de la glycémie, détection de cancers, etc). Ils peuvent aussi faciliter le diagnostic de quelques maladies en effectuant des mesures physiologiques telles que : la tension artérielle, battements du cŒur, etc. à l'aide des capteurs ayant chacun une tâche bien particulière. Les données physiologiques collectées par les capteurs peuvent être stockées pendant une longue durée pour le suivi d'un patient. D'autre part, ces

réseaux peuvent détecter des comportements anormaux (chute d'un lit, choc, cri, etc.) chez les personnes dépendantes (handicapées ou âgées) [5].



FIGURE 1.10 – Domaine médicale.

- **Applications commerciales** : Il est possible d'intégrer des capteurs au processus de stockage et de livraison dans le domaine commercial. Le réseau ainsi formé pourra être utilisé pour connaître la position, l'état et la direction d'un paquet. Il devient alors possible pour un client qui attend la réception d'un paquet, d'avoir un avis de livraison en temps réel et de connaître la localisation actuelle du paquet. Pour les entreprises manufacturières, les réseaux de capteurs permettront de suivre le procédé de production à partir des matières premières jusqu'au produit final livré. Grâce aux réseaux de capteurs, les entreprises pourraient offrir une meilleure qualité de service tout en réduisant leurs coûts [5].

1.4.7 La communication dans les réseaux de capteur sans fil

Comme tous les types de réseaux, les RCSFs utilisent une architecture de communication en couches, ce sont les cinq premières couches du modèle OSI ; la couche physique, la couche liaison de données, la couche réseau, la couche transport et la couche application. Chaque couche a son propre rôle et ses propres protocoles pour atteindre son objectif. La pile protocolaire est utilisée par le noeud puits et les autres noeuds du réseau.

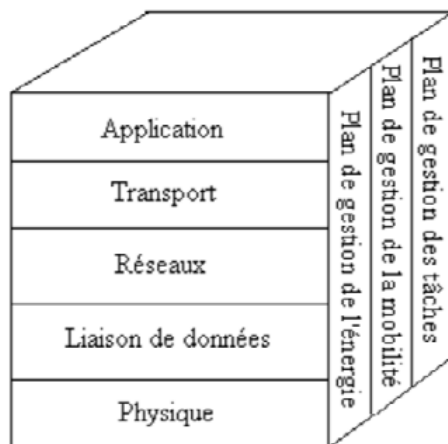


FIGURE 1.11 – Couche du RCSFs[3].

- **La couche physique** : Spécifions des caractéristiques matérielles, des fréquences Porteuses, etc.
- **La couche liaison de donnée** : spécifie comment les données sont expédiées entre deux noeuds /routeurs dans une distance d'un saut. Elle est responsable de multiplexage des données, du contrôle d'erreurs, de l'accès au media. Elle assure la liaison point à point et multipoints dans un réseau de communication.
- **La couche réseau** : Dans la couche réseau le but principal est de trouver une route et une transmission fiable des données, captée, des noeuds capteurs vers les puits " sink " en optimisant l'utilisation de l'énergie de capteurs.
- **La couche application** : cette couche assure l'interface avec les applications. Il s'agit donc du niveau le plus proche des utilisateurs, gérés directement par les logiciels.
- **Plan de gestion** : Les plans de gestion d'énergie, de mobilité et de tâche contrôlent l'énergie, le mouvement et la distribution de tâche au sein d'un noeud capteur. Ces plans aident les noeuds capteurs à coordonner la tâche de captage et minimiser la consommation d'énergie. ils sont donc nécessaires pour que les noeuds de capteurs puissent collaborer ensemble, acheminer les données dans un réseau mobile et partager les ressources entre eux en utilisant efficacement l'énergie disponible. Ainsi, le réseau peut prolonger sa durée de vie.

- **Plans de gestion d'énergie** : contrôle l'utilisation de la batterie. par exemple après réception d'un message, le capteur éteint son récepteur afin d'éviter la duplication des messages déjà reçus. En outre, si le niveau d'énergie devient bas, le noeud capteur diffuse à ses voisins une alerte les informant qu'il ne peut pas participer au potage. L'énergie restante est réservée au captage.
- **Plan de gestion de mobilité** : détecte et enregistre le mouvement du noeud capteur. Ainsi un retour arrière vers l'utilisateur est toujours maintenu et le noeud peut garder trace de ses noeuds voisins. en déterminant les voisins, noeuds capteurs peuvent balancer l'utilisation de leur énergie et la réalisation de tâche
- **Plan de gestion de tâches** : balance et ordonnance les différentes tâches de captage de données dans une région spécifique. Il n'est pas nécessaire que tous les noeuds de capteurs de cette région effectuent la tâche de captage au même temps ; certains exécutent cette tâche plus que d'autres selon leur niveau de batterie [5].

1.4.8 Les types de communications dans les réseaux de capteurs sans fil

Le but d'un réseau de capteur sans fil est la surveillance d'un environnement physique et la fourniture des informations capturées. Chaque noeud est équipé d'un ou plusieurs capteurs, par lesquels les données sont capturées et transportées à travers d'autre noeuds du réseau à la destination des données.

Dans les RCSFs, deux types de noeuds sont identifiés logiquement : les noeuds qui principalement transmettent leurs propres données capturées appelés noeuds capteurs, et les noeuds qui transmettent les messages aux autre noeuds appelés noeuds de relais ou station de base. Les données capturées sont acheminées depuis les noeuds source jusqu'aux noeuds destinataire à travers les noeuds intermédiaires, créant ainsi une topologie multi-sauts. Cette organisation logique donnera naissance à quatre types de communications qui sont défini comme suit :

- **La communication noeud capteur à un noeud capteur** : Ce type de communication directe est utilisé pour des opérations locales, par exemple pendant le processus de création de route.
- **La communication noeud capteur à un noeud intermédiaire** : Les données capturées sont transmises d'un noeud capteur à un noeud intermédiaire. Ce type de communication est souvent unicast (seul envois).
- **La communication noeud intermédiaire à un noeud capteur** : Les requêtes et la signalisation des messages, sont souvent multicast (plusieurs envois), elles sont diffusées

par les noeuds intermédiaires, pour atteindre un sous-ensemble de noeuds.

- **La communication noeud intermédiaire à un noeud intermédiaire** : La communication entre ces noeuds peut être dans la plupart du temps unicast.

1.4.9 Défis des RCSFs

Les recherches dans le domaine des réseaux de capteurs ont révélé plusieurs problématiques, parmi ces problématiques, nous citons[8] :

- **Routage** : Concevoir un protocole de routage performant en termes de minimisation de la consommation de l'énergie, du choix des routes optimales pour l'acheminement de l'information d'un capteur à la station de base et vice versa, de réduction du délai de délivrance des paquets...Ainsi le réseau doit passer à l'échelle sans que ses performances se dégradent.
- **Qualité de service** : Des protocoles au niveau de la couche MAC devraient être capables d'établir des priorités entre les flux, limiter les pertes de paquets vitaux pour la gestion du réseau, ou du moins en restreindre l'impact.
- **Diffusion de l'information** : les protocoles de diffusion conçus pour les réseaux de capteurs doivent tenir compte de leurs spécificités ainsi que de leurs contraintes intrinsèques imposées. Ainsi, pour concevoir un protocole efficace, il faudrait assurer une couverture maximale des capteurs composant le réseau (taux d'accessibilité supérieur 90
- **Sécurité** : pour les applications qui exigent un niveau de sécurité assez élevé telles que les applications militaires, des mécanismes d'authentification, de confidentialité, et d'intégrité doivent être mis en place au sein de leur communauté. Les algorithmes de cryptographie conçus pour les réseaux de capteurs doivent tenir compte des ressources limitées que présentent ces réseaux.
- **Couverture** : La couverture est un paramètre important dans les réseaux de capteurs sans fils dans la mesure où elle affecte le résultat de la tâche de perception effectuée par un réseau, En exploitant la redondance des capteurs issue de la forte densité du réseau, en utilisant l'ordonnancement d'activité afin d'avoir une surveillance totale d'une zone d'intérêt, on peut réduire la consommation d'énergie, et donc étendre, la durée de vie du réseau.

1.4.10 Facteurs et contraintes de conception des RCSF

La conception et la réalisation des réseaux de capteurs sans fil est influencée par plusieurs paramètres. Ces facteurs importants servent comme directives pour le développement des algorithmes et protocoles utilisés dans les réseaux de capteurs, ils sont considérés également comme métriques de comparaison de performances entre les différents travaux dans le domaine.

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

- **La scalabilité (le facteur d'échelle)** : Le nombre de noeuds déployés pour un projet peut atteindre le million. Un nombre aussi important de noeuds engendre beaucoup de transmissions inter nodales et nécessite que le puits "sink " soit équipé de beaucoup de mémoire pour stocker les informations reçues [2].
- **La consommation d'énergie** : Un capteur, de par sa taille, est limité en énergie. Dans la plupart des cas le remplacement de la batterie est impossible. Ce qui veut dire que la durée de vie d'un capteur dépend grandement de la durée de vie de la batterie. Dans un réseau de capteurs (multi-sauts) chaque noeud collecte des données et envoie/transmet des valeurs. Le dysfonctionnement de quelques noeuds nécessite un changement de la topologie du réseau et un re-routage des paquets. Toutes ces opérations sont gourmandes en énergie, c'est pour cette raison que les recherches actuelles se concentrent principalement sur les moyens de réduire cette consommation [2].
- **Topologie dynamique** : La topologie des réseaux de capteurs peut changer au cours du temps pour les raisons suivantes :
 - Les noeuds capteurs peuvent être déployés dans des environnements hostiles (champ de bataille par exemple), la défaillance d'un noeud capteur est, donc très probable.
 - Un noeud capteur peut devenir non opérationnel à cause de l'expiration de son énergie.
 - Dans certaines applications, les noeuds capteurs et les stations de base sont mobiles [3].
- **Durée de vie du réseau** : C'est l'intervalle de temps qui sépare l'instant de déploiement du réseau de l'instant où l'énergie du premier noeud s'épuise. Selon l'application, la durée de vie exigée pour un réseau peut varier entre quelques heures et plusieurs années [3].
- **Agrégation de données** : Dans les RCSFs, les données produites par les noeuds capteurs voisins sont très corrélées spatialement et temporellement. Ceci peut engendrer

la réception par la station de base d'informations redondantes. Réduire la quantité d'informations redondantes transmises par les capteurs permet de réduire la consommation d'énergie dans le réseau et ainsi d'améliorer sa durée de vie. L'une des techniques utilisée pour réduire la transmission d'informations redondantes est l'agrégation des données. Avec cette technique, les noeuds intermédiaires agrègent l'information reçue de plusieurs sources. Cette technique est connue aussi sous le nom de fusion de données [3].

- **La tolérance de fautes** : La défaillance ou le blocage des noeuds dans un réseau de capteurs peut être engendrés par plusieurs causes, notamment l'épuisement d'énergie, l'endommagement physique, ou les interférences liées à l'environnement. La propriété de tolérance aux pannes est définie par l'habilité du réseau à maintenir ses fonctionnalités sans interruptions provoquées par la panne des capteurs. Elle vise donc à minimiser l'influence de ces pannes sur la tâche globale du réseau [1].
- **Les contraintes matérielles** : La principale contrainte matérielle est la taille du capteur. Les autres contraintes sont que la consommation d'énergie doit être moindre pour que le réseau survive le plus longtemps possible, qu'il s'adapte aux différents environnements (fortes chaleurs, eau,...), qu'il soit autonome et très résistant vu qu'il est souvent déployé dans des environnements hostiles [2].

1.4.11 La différence entre les réseaux de capteurs et Ad hoc

Un réseau de capteurs sans fil appartient à la famille des réseaux ad hoc. Cependant, on peut noter quelques différences majeures :

- En général, le nombre de noeuds dans un réseau de capteurs est beaucoup plus élevé par rapport à celui d'un réseau ad hoc.
- Les noeuds capteurs sont déployés en grand nombre pour prévenir d'éventuelles défaillances.
- La topologie du réseau change fréquemment (ajout/retrait de noeuds capteurs, mobilité relative d'un noeud).
- La communication entre les noeuds capteurs repose en général sur la diffusion, alors que la plupart des réseaux ad hoc utilisent une communication point à point.
- Les noeuds capteurs ont des ressources limités (puissance de transmission, de calcul, autonomie d'énergie électrique, capacité mémoire).
- Les noeuds capteurs peuvent ne pas avoir des identificateurs globaux comme dans le protocole IP. Il s'agit ici d'éviter d'engendrer un overhead élevé dû à un grand nombre de noeuds [4].

1.4.12 Les problèmes liés aux réseaux de capteurs sans fil

- La miniaturisation des capteurs pose des problèmes de communication et de ressources d'énergie. Il faut que le capteur soit suffisamment intelligent pour rassembler l'information requise et l'émettre à bon escient
- Le processeur du capteur ne doit pas être utilisé trop intensivement afin de consommer le moins d'énergie possible. Il doit donc incorporer des éléments réactifs plutôt que cognitifs.
- Pour assurer un bon débit, la portée des émetteurs-récepteurs doit être nécessairement faible, de l'ordre d'une dizaine de mètres.
- La mise en place d'un réseau de capteurs pose donc des problèmes de routage, de contrôle des erreurs et de gestion de l'alimentation [1].

1.5 Conclusion

Dans ce chapitre, nous avons défini quelques concepts fondamentaux sur les réseaux sans fil, Ad-hoc et plus particulièrement les réseaux de capteurs sans fil qui présentent un intérêt considérable et une nouvelle étape dans l'évolution des technologies de l'information et de la communication. Cette nouvelle technologie suscite un intérêt croissant étant donnée la diversité de ces applications.

De ce fait, nous allons développer quelques techniques de conservation d'énergie dans le chapitre suivant.

CHAPITRE 2

LE ROUTAGE DANS LES RÉSEAUX DE CAPTEURS SANS FIL

2.1 Introduction

Le routage est une méthode d'acheminement des informations vers une destination donnée dans un réseau de connexion. Le développement des protocoles de routage spécifique aux réseaux de capteurs a attiré une grande attention des chercheurs dans le domaine. Mais les limites imposées par l'architecture matérielle des capteurs, en particulier celle de l'énergie. De ce fait, les protocoles de routage élaborés doivent assurer une consommation minimale d'énergie tout en maintenant le bon fonctionnement du réseau et sans dégrader ses performances.

Dans ce chapitre nous allons décrire quelques défis de routage dans les RCSFs, ensuite les techniques de minimisation de la consommation d'énergie. Par la suite, les différentes approches de routage et vers la fin, définir les métriques de mesure de l'efficacité des protocoles de routage dans les RCSFs.

2.2 Les défis de routage dans les réseaux de capteurs sans fil

2.2.1 Le déploiement des capteurs

Suivant le type d'application, le mode de déploiement des capteurs peut être déterministe ou aléatoire. Dans le déploiement déterministe, les capteurs sont placés manuellement et les données sont toujours acheminées via une route prédéterminée et fixe. Quand le déploiement

est aléatoire (auto-organiser), les noeuds capteurs sont dispersés aléatoirement, créant ainsi une infrastructure ad hoc. Dans ce cas, si la distribution qui en résulte n'est pas uniforme, le recours à une architecture de groupe optimale s'avère nécessaire afin de permettre une meilleure connectivité, ainsi que des opérations plus efficaces en consommation d'énergie [9].

2.2.2 Agrégation des données

L'agrégation de données est l'une des caractéristiques des réseaux de capteurs qui est une possibilité de réduire la quantité de données circulant dans le réseau, afin de conserver de l'énergie, en fusionnant les données par des noeuds particuliers du réseau.

L'agrégation exige la transmission des données et aussi les messages de contrôle, ce qui impose des contraintes sur l'architecture du réseau. Pendant l'agrégation, nous devons aussi prendre en considération quelques problèmes : Les erreurs dans les messages, les messages perdus, la redondance des données, la synchronisation entre les noeuds etc [10].

2.2.3 Dynamicité des réseaux

En général, les réseaux de capteurs sont supposés statiques. Dans ce cas, les protocoles de routage considèrent les composants de ces derniers comme stationnaire, comme il est le cas dans les réseaux de surveillance de forêts, généralement utilisés pour la prévention contre les incendies. Cependant, l'aspect dynamique de l'un des composants des réseaux de capteurs, est nécessaire dans plusieurs applications, comme la mobilité des capteurs ainsi que le noeud puits. Dans ce type de réseaux, le routage de données est une tâche complexe, car le facteur de stabilité de route choisit devient un facteur d'optimisation important, en plus de l'énergie consommée la bande passante disponible, etc. De plus, la surveillance des phénomènes dynamiques exige l'envoi de rapports périodiques, ce qui implique la génération du trafic considérable à router vers le noeud puits [11].

2.2.4 Modèle de livraison de données

Le modèle de renvoi des données captées constitue un facteur important qui doit être pris en compte lors de la conception d'un protocole de routage. Suivant l'application, ce modèle peut être continu, orienté événement, orienté requête, ou hybride.

Le modèle de renvoi continu est le mieux adapté aux applications qui nécessitent des rapports périodiques sur l'environnement surveillé. Pour cela, chaque noeud allume ses dispositifs de captage et de transmission d'une manière périodique, capte son environnement puis transmet les résultats dans des intervalles de temps réguliers.

Par contre le modèle de renvoi orienté événement ou requête, les noeuds doivent réagir d'une

manière immédiate à un changement brusque dans la valeur de l'attribut capté, ou à une requête générée par le noeud puits. Ce dernier modèle est convenable pour les applications où le temps de réponse des capteurs peut être critique. La combinaison de ces trois modèles donne naissance à un autre mode de transmission appelé hybride [10].

2.2.5 Hétérogénéité Noeuds/liens

Plusieurs études ont supposé qu'un réseau de capteurs est constitué de noeuds homogènes ayant les mêmes capacités en termes de calcul, de transmission et d'énergie disponible. Mais, selon l'application, les noeuds capteurs peuvent avoir des rôles différents.

L'existence d'un ensemble de capteurs hétérogènes soulève beaucoup de questions techniques liées au routage de données. En effet, certaines applications imposent un mélange de capteurs divers pour assurer différentes fonctionnalités du réseau. Ces capteurs spéciaux peuvent être déployés indépendamment comme ils peuvent inclure des fonctionnalités différentes; mêmes le captage et la délivrance des données peuvent être produits par ces capteurs à différents taux. Par exemple les algorithmes de routages hiérarchiques exigent qu'un cluster head doive posséder que les capteurs normaux, en termes d'énergie, bande passante et mémoire, car il est le seul responsable de l'agrégation des données captées et leurs transmissions vers la station de base [10].

2.2.6 Qualité de service (QoS)

Dans certaines applications exigent que les données soient délivrées aux destinataires durant certaines périodes, sinon elles sont inutiles, donc ce genre d'applications qui sont soumises à des contraintes temporelles, une latence limitée dans les délais de transmission doit être garantie. Comme la conservation d'énergie est une contrainte critique dans plusieurs applications de réseaux de capteurs, alors cette dernière doit être prise en compte en première priorité relativement à la qualité de service. Pour minimiser la dissipation d'énergie et prolonger la durée de vie de réseau [12].

2.3 Techniques de minimisation de la consommation d'énergie

Dans les réseaux ad hoc, la consommation de l'énergie a été considérée comme un facteur déterminant mais pas primordial car les ressources d'énergie peuvent être remplacées par l'utilisateur. Ces réseaux se focalisent plus sur la QoS (Quality of Service) que sur la consommation de l'énergie. Par contre, dans les réseaux de capteurs, la consommation d'énergie est très importante puisque généralement les capteurs sont déployés dans des zones inaccessibles.

Ainsi, il est difficile voire impossible de remplacer les batteries après leur épuisement. De ce fait, la consommation d'énergie au niveau des capteurs a une grande influence sur la durée de vie du réseau. Après la description des principales causes de consommation d'énergie dans les RCSF, nous présentons dans ce qui suit les différentes techniques utilisées pour minimiser cette consommation. Ces techniques sont appliquées soit au niveau de la couche liaison soit au niveau de la couche réseau. Le schéma suivant donne un aperçu global de ces mécanismes :

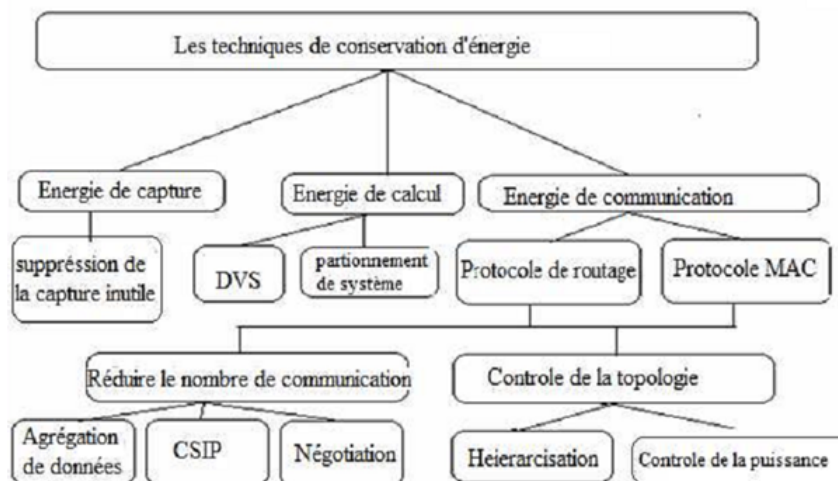


FIGURE 2.1 – Les techniques de conservation d'énergie[4].

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

- La seule solution apportée pour la minimisation de la consommation d'énergie au niveau de la capture consiste à réduire les fréquences et les durées de captures.
- L'énergie de calcul peut être optimisée en utilisant deux techniques :
 - L'approche DVS (Dynamique Voltage Scaling) [13] qui consiste à ajuster de manière adaptative la tension d'alimentation et la fréquence du microprocesseur pour économiser la puissance de calcul sans dégradation des performances.
 - L'approche de partitionnement de système qui consiste à transférer un calcul prohibitif en temps de calcul vers une station de base qui n'a pas de contraintes énergétiques et qui possède une grande capacité de calcul [14].
- La minimisation de la consommation d'énergie pendant la communication est étroitement liée aux protocoles développés pour la couche réseau et la sous-couche MAC. Ces protocoles se basent sur plusieurs techniques : l'agrégation de données, la négociation et à la technique CSIP (Collaborative Signal and Information Processing). Cette dernière technique est une discipline qui combine plusieurs domaines [15] : la communication et le calcul à basse puissance, le traitement de signal, les algorithmes distribués, la tolérance aux fautes, les systèmes adaptatifs et la théorie de fusion des capteurs et des décisions.

Ces techniques ont le but de réduire le nombre d'émission/ réception des messages.

2.4 Les approches de routage dans les réseaux de capteurs sans fil

Dans les réseaux de capteurs sans fil, les noeuds sont déployés d'une manière dense où chaque noeud peut avoir plusieurs dizaines de voisins. Pour permettre la communication dans le réseau déployé, des protocoles de routage basés sur la communication multi-sauts entre les noeuds capteurs et le noeud puits du réseau sont nécessaires.

L'objectif d'un protocole de routage est de trouver des chemins qui mènent vers la destination et qui optimise la métrique de consommation d'énergie. Plusieurs stratégies de routage ont été proposées pour les réseaux de capteurs sans fil. Leurs principes de fonctionnement diffèrent suivant la philosophie de la classe à laquelle ils appartiennent. Ces approches peuvent être classées suivant la structure du réseau, les fonctions des protocoles, l'établissement des routes et l'initiateur de la communication [16].

La figure suivante englobe les différentes approches de routage dans les RCSFs :

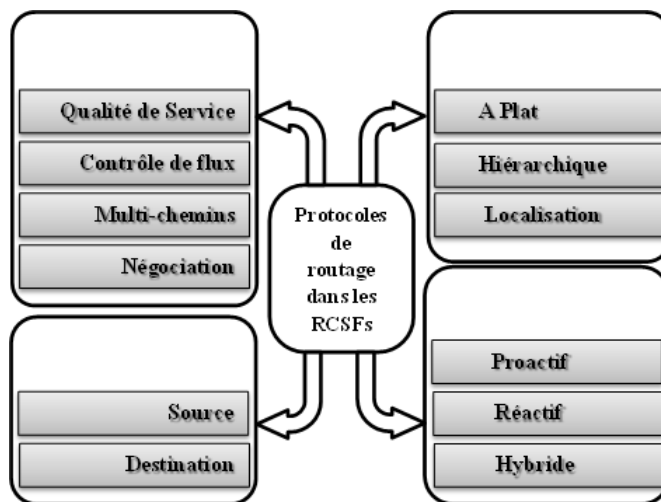


FIGURE 2.2 – Les différentes approches de routage dans les RCSFs.

2.4.1 Classification selon la structure du réseau

Les protocoles de routage basés sur la structure du réseau sont classés en trois catégories : protocoles à plat (Flat based routing), protocoles hiérarchiques (Hierarchic based routing) et

protocoles basés sur la localisation géographique (Location based routing).

1. Routage à plat

Dans ce type de protocole, tous les noeuds ont le même rôle, homogènes et communiquent entre eux sans aucun intermédiaire, et seul la station de base est chargée de la collecte de données issues des différents capteurs. Les topologies plates sont caractérisées par la simplicité des algorithmes utilisés pour le routage des données et la scalabilité du réseau, du fait que les noeuds ont besoin uniquement des informations sur leurs voisins directs afin de participer à la tâche de routage [16].

• Flooding

La technique du flooding est une technique classique qui peut être utilisée pour le routage dans les réseaux de capteurs. Dans cette approche, chaque noeud recevant un paquet de données le diffuse à tous ses voisins, jusqu'à ce que le nombre maximum de sauts pour ce paquet soit atteint [17].

Discussions

Le Flooding est une technique réactive qui nécessite pas une maintenance coûteuse de la topologie du réseau, ni des algorithmes complexes pour la découverte des routes, mais elle présente plusieurs inconvénients tels que :

- L'implosion : cette situation survient si des messages dupliqués sont envoyés au même noeud.
- Le Chevauchement : si deux noeuds observant la même région sont stimulés en même temps, leurs noeuds voisins recevront des messages dupliqués.
- Ignorance de ressources : cette technique de flooding ne prend pas en compte les ressources d'énergie disponibles.

Un protocole efficace en consommation d'énergie, cependant, doit prendre en compte à chaque instant la quantité d'énergie disponible.

• Gossiping

Le Gossiping est une amélioration de Flooding où un noeud recevant un paquet ne le diffuse pas à tous ses voisins, mais il le transmet à un seul, sélectionné aléatoirement. En effet, chaque noeud capteur dans le réseau sélectionne aléatoirement un noeud parmi ses voisins pour lui transmettre les données reçues, une fois que le noeud voisin reçoit ces données, il choisit un autre noeud d'une façon aléatoire pour lui envoyer ces données à son tour, jusqu'à ce que les données atteignent la station de base [17].

Discussions

L'approche Gossiping évite le problème d'implosion en ayant une seule copie du message au niveau de chaque noeud, elle prend beaucoup plus de temps pour propager les messages dans le réseau.

- **SGDF (Single Gossiping with Directional Flooding routing protocole)**

Propose par [18] pour résoudre les problèmes de Flooding. Il est constitué de deux étapes :

- L'étape 1(initialisation de la topologie de réseau) : Une fois les capteurs déployés aléatoirement, le sink diffuse un message " hello " à ses voisins. Ce message contient trois champs : l'adresse du sink, le nombre de sauts (initialisé à 1), et le seuil TH (Threshold)(nombre compris entre 0 et 1 pour décider dans la phase de routage s'il utilise le mode Flooding). Les capteurs voisins recevant ce message, sauvegardent l'adresse du sink et le nombre de sauts. Si un capteur n'a pas reçu le message, il garde le nombre de sauts comme gradient en l'incrémentant de 1, ensuite il va créer un nouveau message qui sera envoyé en diffusant à tous ses voisins. Si un capteur recevant un message avait un gradient, il compare le nombre de sauts du message avec son gradient et sauvegarde le plus petit. Cette étape se termine lorsque tous les noeuds ont reçu le message au moins une fois, tel qu'à la fin chaque capteur aura un gradient qui détermine le nombre de sauts pour atteindre le sink.

- L'étape 2 (le routage) : Après l'initialisation du réseau, vient l'étape de routage. la détection d'un évènement, le noeud initiateur diffuse un message de demande d'informations (adresse et gradient) à ses voisins, ainsi, il choisi aléatoirement parmi ceux dont le gradient est inférieur à lui pour être le prochain à recevoir le paquet. Les voisins non choisis (dont le gradient est inférieur à celui de la source) génèrent un nombre aléatoire. Si ce dernier est inférieur à celui généré dans la phase d'initialisation, alors ils diffusent le paquet en mode flooding directionnel. Les étapes précédentes sont reproduites jusqu'à ce que la station de base reçoive le paquet.

Discussions

Ce mode de routage évite le problème d'implosion en ayant une seule copie du message au niveau de chaque noeud. Aussi, il ne permet pas le bouclage infini qui est produit par le choix aléatoire des voisins (si le choix aléatoire est utilisé comme seul mode de routage). Ce protocole prend beaucoup de temps pour acheminer l'information de la source à la destination et la technique de choix des routeurs ne prend pas en compte le paramètre énergie, donc peut affaiblir certains capteurs plus vite.

2. Routage hiérarchique

Dans cette architecture, le réseau est partitionné en sous ensembles appelés clusters, afin de faciliter la gestion du réseau ainsi que le routage de données. Dans ce type de protocole, chaque cluster est constitué d'un noeud simple et d'un noeud leader (ou cluster head). Seul le cluster head communique avec d'autres capteurs ou avec la station de base. Tous les capteurs d'un cluster envoient les données au cluster head. Ensuite ce dernier s'en charge de les acheminées vers d'autre cluster head ou bien vers la station de base [16].

• LEACH(Low-Energy Adaptive Clustering Hierarchy)

LEACH est le premier protocole de routage hiérarchique avec une efficacité énergétique, qui a été proposé par **Heinzelman et al** en **2000** dans [19]. LEACH la technique de clustering qui dévise le réseau en deux niveaux : les clusters head (CHs) et les noeuds membres. Le protocole se déroule en périodes. Chaque période se compose de deux phases : construction et communication. La durée de la phase de communication est plus longue que celle de la phase de construction afin de minimiser l'overhead.

Le but de la première phase est la construction des clusters en choisissant les chefs (CHs) et en établissant la politique d'accès au média au sein de chaque groupe. Durant cette phase, chaque noeud n génère un nombre aléatoire entre 0 et 1. Si ce nombre est inférieur à un seuil $T(n)$, le noeud devient clusterhead. $T(n)$ est défini comme suit :

$$T(n) = \begin{cases} \frac{p}{1-p \times (r \bmod \frac{1}{p})}, & n \in G \\ 0 & \text{sinon} \end{cases} \quad (2.1)$$

Où :

- $T(n)$: valeur seuil.
- r : période donnée.
- p : est le pourcentage désiré de clusters head.
- G : est l'ensemble de noeuds qui n'ont pas été élu durant $\frac{1}{p}$ les dernières périodes précédentes.

Par la suite, chaque noeud qui s'est élu cluster head émet un message de notification à tous ces voisins. Les noeuds qui récoltent les message de notification, décident leur appartenance à un cluster. La décision est basé sur l'amplitude du signal reçu : le cluster head ayant le signal le plus fort est choisi (ie. le plus proche). En cas d'égalité un chef aléatoire est choisi. Chaque membre informe son chef de sa décision.

Pour la deuxième phase, les noeud de capteurs peuvent commencer à envoyer leurs données captées au cluster head pendant leur propre slots. Cela leur permet d'éteindre leur interface de communication en dehors de leurs slots réservés, afin d'économiser

leur énergie. Le cluster head agrège les données avant de les transmettre au collecteur (SB). Cette communication entre le cluster head et la station de base, se fait d'une manière directe. Après la phase de communication qui dure un certain temps, la phase de construction recommence.

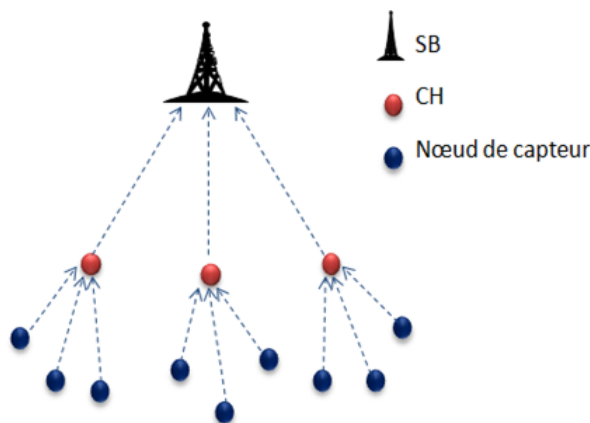


FIGURE 2.3 – Le modèle LEACH.

Discussions

LEACH est capable de préserver de l'énergie résiduelle et prolonge la durée de vie des capteurs. Mais comme il est le premier protocole de routage hiérarchique alors il pose quelques problèmes, donc il n'est pas souhaitable dans les larges réseaux, car si nous supposons que les clusters head sont très loin de la station de base alors la transmission directe consomme beaucoup d'énergie, de plus les clusters head sont sélectionnés aléatoirement et il pose aucune contrainte sur leur distribution ainsi que sur leur niveau d'énergie. D'où, les clusters head peuvent se concentrer dans un même endroit.

- **LEACH-C (LEACH Centralized)**

LEACH-C est une variante de LEACH, a été conçue pour répondre au problème de sélection aléatoire du CH dans LEACH. Elle a été proposée par les mêmes auteurs de LEACH dans [20]. Cette variante utilise une architecture centralisée pour choisir les clusters head, la phase d'état stable de LEACH-C est comme suit :

- Lors de la phase d'installation de LEACH-C, les noeuds informent la SB par l'envoi d'informations sur leur position par GPS et leur niveau d'énergie.
- La station de base désigne les clusters head en se basant sur la moyenne des niveaux d'énergie, les noeuds ayant un niveau d'énergie inférieur à la moyenne ne peuvent pas être des clusters head dans l'itération courante.
- Enfin, la station de base diffuse un message contenant les identificateurs des clusters head.
- Le fonctionnement des autres opérations est similaire à celui de LEACH.

Discussions

LEACH-C augmente considérablement le surcoût du réseau puisque tous les capteurs devront envoyer leurs informations de localisation à la station de base en même temps pendant chaque phase d'élection de clusters head. Plusieurs travaux présentés [21] dans la littérature ont prouvé qu'une telle architecture centralisée ne supporte pas le passage à l'échelle, étant plus particulièrement appropriée à des réseaux de petite taille.

- **HEED(Hybrid, Energy-Efficient Distributed clustering in Ad-hoc Sensor Networks)**

Les auteurs de [22] ont proposé un algorithme de clustering distribué appelé HEED pour les réseaux de capteurs sans fil. Contrairement aux techniques précédentes, HEED ne fait aucune restriction sur la distribution et la densité des noeuds. Il ne dépend pas de la topologie du réseau ni de sa taille, mais il suppose que les capteurs ont la possibilité de modifier leur puissance de transmission. HEED sélectionne les clusters heads selon un critère hybride regroupant l'énergie restante des noeuds et un second paramètre tel que le degré des noeuds. Il vise à réaliser une distribution uniforme des clusters heads dans le réseau et à générer des clusters équilibrés en taille. Un noeud est élu comme cluster head avec une probabilité CH_{prob} tel que :

$$CH_{prob} = C_{prob} * \frac{E_r}{E_{max}}$$

Où :

E_r est l'énergie résiduelle du noeud.

C_{prob} est le nombre optimal de clusters.

E_{max} est l'énergie globale dans le réseau.

Cependant, l'évolution de E_{max} présente une certaine difficulté, à cause de l'absence de toute commande centrale. Un autre problème réside dans la détermination du nombre optimal de clusters. De plus, HEED ne précise pas de protocole particulier à utiliser pour la communication entre les clusters head et le puits. l'intérieur du cluster, le problème ne se pose pas car la communication entre les membres du cluster et le cluster head est directe (un seul saut). D'autre part, avec HEED, la topologie en clusters ne réalise pas de cosommation minimale d'énergie dans les communication intra-cluster et les clusters générés ne sont pas équilibrés en taille.

Discussions

HEED prolonge la durée de vie du réseau en distribuant l'énergie de la communication et le nombre de CHs d'une façon uniforme et donc produire des clusters compact tout

en minimisant la charge des messages de contrôle.

L'étendue du réseau est posée comme un problème de ce protocole pour les réseaux à grande échelle, vu la stratégie adoptée pour la communication entre les CHs et la station de base qui se fait via un seul saut.

- **MHEED(Multi-hops Hybrid, Energy Efficient Distributed clustering)**

MHEED est un protocole de routage hiérarchique, il a été proposé dans [23] pour améliorer le protocole HEED. Il se base sur la notion de multi sauts pour résister au facteur d'échelle, et résoudre le problème de communication à un seul saut qui existe dans le protocole HEED.

Le protocole MHEED est constitué de plusieurs tours d'agglomération (clustering) successifs, où chaque tour comporte deux phases :

- **Phase d'agglomération (clustering)** : Faite selon le principe décrit dans HEED. la fin de cette phase, les chefs de grappe (cluster heads) sont élus et leurs grappes (clusters) sont formées.
- **Phase de communication** : La transmission de données se fait en deux étapes. Tout d'abord, les membres de grappes transmettent de manière périodique leurs données à leurs chefs de grappe (pendant les tranches de temps qui leur sont allouées). A la réception de ces données, chaque chef va agréger les données de ces membres pour minimiser le nombre de paquets à transmettre. Il envoie le paquet agrégé à la station de base. Cette transmission est faite, de relais en relais, par le biais des chefs de grappes relais prédéterminés comme suit :
 - ◊ Dans la première variante, appelée MHEEDB, chaque chef de grappe va garder un seul relais (le meilleur) parmi les chefs de grappes voisines.
 - ◊ Dans la seconde variante, appelée MHEEDP, chaque chef de grappe va garder plusieurs relais vers la station de base parmi les chefs de grappes voisines. A chaque transmission de données, il choisit l'un d'entre eux (d'une manière probabilisée) pour faire suivre ses données ou celles reçues d'autres chefs plus éloignés l'ayant choisit pour relais.

la fin de chaque tour d'agglomération, le processus d'agglomération est déclenché de nouveau. De nouveaux chefs de grappes sont élus et de nouveaux relais sont recherchés pour acheminer les données.

Discussions

Les grappes obtenues par la simulation de la phase d'agglomération, sont bien distribuées dans le réseau et sont formées d'une manière différente d'un tour à un autre, les noeuds avec la plus grande énergie résiduelle sont destinées à être chef de grappe (cluster heads).

L'énergie consommée pendant la phase d'agglomération du MHEED est inférieure à celle d'autre protocole d'agglomération tel que LEACH.

MHEEDP permet de réaliser un équilibrage de charge entre les différents CHs.

MHEED donne de meilleurs résultats en terme d'augmentation de la durée de vie du réseau et de délivrances des données.

Les CHs à la proximité de la station de base meurent rapidement, ce qui cause la perte de données dans le réseau.

3. Routage basé sur la localisation géographique

Dans ce type de routage, les noeuds sont adressés à l'aide de leurs positions. Les informations sur la position de chaque noeud dans le réseau est nécessaire pour estimer l'énergie consommée lors des différentes émissions. Comme les noeuds capteurs sont déployés dans une région d'une manière aléatoire, les informations de localisation de ces noeuds peuvent être utilisées dans le routage des données d'une manière efficace en terme d'énergie dans le but de maximiser la durée de vie du réseau [16].

- **GAF (Geographic Adaptive Fidelity)**

Proposé par (Xu et al ,en 2001). Geographic adaptive fidelity (GAF) est L'énergie-consciente basée sur l'emplacement algorithmme de routage conçu principalement pour les réseaux mobiles ad hoc, Mais peut être applicable aux réseaux de capteurs. ainsi GAF économise de l'énergie en désactivant les noeuds inutiles. Dans le réseau, Il forme une grille virtuelle pour la zone couverte. Chaque noeud utilise son GPS qui indique l'emplacement de s'associer à un point dans la grille virtuel. Les noeuds associés au même point sur Les grilles sont considérées comme équivalents en termes de coût de routage des paquets. Une telle équivalence est exploité en gardant certains noeuds situés dans une zone particulière de la grille en état de veille afin d'enregistrer L'énergie. GAF peut sensiblement augmenter la durrée de vie du Réseau comme le nombre de noeuds qui augmente.

Il existe trois états définis dans GAF. Ces états sont :

- Découverte :Pour déterminer les voisins dans la grille,
- Actif : Reffétant la participation dans le système de routage,
- Sommeil : quand La radio est désactivée.

L'état des transitions dans GAF est illustré sur la figure suivante [24].

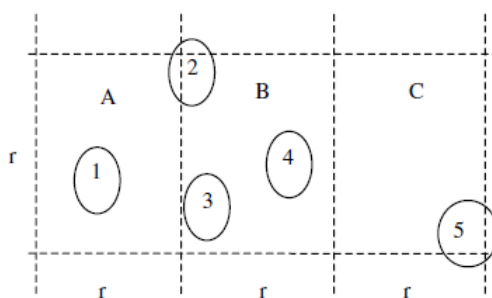


FIGURE 2.4 – Protocole GAF [16].

Discussions

Les résultats de simulation montrent que GAF effectue au moins ainsi qu'un normal protocole de routage ad hoc en termes de latence et de paquets perdus et augmente la durée de vie du réseau et economiser de l'énergie. Bien que GAF est basé sur l'emplacement, il peut également être considéré comme un protocole hiérarchique, où les clusters sont basés sur l'emplacement géographique. Pour chaque zone de la grille, représentant un noeud agit comme chef pour transmettre des données aux autres noeuds.

2.4.2 Classification selon les fonctions du protocole

Les protocoles de routage peuvent être classés selon leurs fonctionnalités en quatre catégories : routage basé sur la qualité de service (Quality of Service based routing), routage basé sur le flux de données dans le réseau (Network flow based routing), routage basé sur des multi-chemins (Multi-path based routing), et routage basé sur la négociation (Negociation based routing).

1. Routage basé sur la Qualité de Service (QoS)

Dans les protocoles de routage basés sur la QoS, le réseau doit assurer un équilibre la consommation d'énergie et la qualité de données. En particulier, le réseau doit satisfaire une certaine métrique de QoS. Ce type de protocole est utilisé dans les applications qu'ont des exigences temps réel (monitoring médical, application militaire, etc.) [16].

- **SAR (Sequential Assignment Routing)**

Proposé par (Sohabi et al, en 2000). Attribution séquentielle de routage (SAR) est le premier protocole pour les réseaux de capteurs qui inclut la notion de QoS dans ses décisions de routage. C'est une approche d'un Tableau-driven multi-chemin s'efforce de parvenir à l'efficacité énergétique et de la tolérance aux panne. Le protocole SAR

créé des arbres enracinés à un saut voisins de l'évier en prenant la métrique QoS, de la ressource énergétique sur chaque chemin et le niveau de priorité de chaque paquet en considération. En utilisant la création des arbres, Plusieurs chemins sont formées. L'un de ces chemins est sélectionné en fonction de la ressources énergétiques et QoS sur le chemin [16].

Discussions

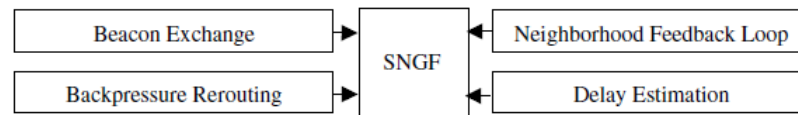
Les résultats montrent que SAR offre moins de puissance de consommation que le minimum de l'énergie métrique Algorithme, qui se concentre seulement sur la consommation d'énergie de chaque paquet sans tenir compte de sa priorité. SAR gère plusieurs chemins bien que cela garantit une tolérance aux pannes et une récupération facile. le protocole souffre de frais de maintenance des tableaux et d'états à chaque noeud de capteur en particulier lorsque le nombre de noeuds est énorme.

- **SPEED**

Proposé par (He et Tassiulas en 2000). Un protocole QoS du routage pour réseaux de capteurs que fournit en temps réel de bout en bout . Le protocole exige à chaque noeud de maintenir des informations sur ses voisins et utilise le transfert géographique pour trouver les chemins. En outre, SPEED s'efforce de garantir une certaine vitesse de chaque paquet dans le réseau afin que chaque application peut estimer le délai de bout en bout pour les paquets, en divisant la distance de l'évier par La vitesse du paquet avant de l'admission de la décision. En outre, SPEED peut fournir évitement de congestion lorsque le réseau est encombré. Le module de routage en vitesse est appelé apatrides Géographique non déterministe forwarding (SNFG) et fonctionne avec quatre autres modules au niveau de la couche réseau .

- Le mécanisme Gyrophare d'échange le recueille des informations sur les noeuds et leur emplacement.
- Delay estimation au niveau de chaque noeud est essentiellement faite par le calcul du temps écoulé lorsqu'un accusé de réception est reçu d'un voisin comme une réponse à une transmission de données.
- En examinant les valeurs de délai, SNGF sélectionne Le noeud, qui répond aux exigences de vitesse. Si un tel noeud ne peut pas être trouvé, le relais de ratio du noeud est cochée. Le quartier feedback Boucle est responsable de la fourniture du relais ratio qui est calculée en examinant la mlle. Les ratios des voisins d'un noeud (les noeuds qui Pourrait ne pas fournir la vitesse désirée) est alimentée au module SNGF. Si le relais ratio est inférieur à un numéro généré de façon aléatoire entre 0 et 1, alors le paquet est abandonné.

- Et enfin, le back pressurere routing Module est utilisé pour éviter les vides, lorsqu'un noeud ne parvient pas à trouver le prochain saut noeud, et à éliminer la congestion en envoyant des messages aux noeuds sources de sorte qu'ils vont poursuivre de nouvelles routes [16].



Discussions

SPEED fonctionne mieux en termes de délai de bout en bout . En outre, l'énergie total de transmission est inférieure à cause de la simplicité de l'algorithme de routage, c'est-à-dire le paquet du contrôle général est inférieure, et à la même répartition du trafic. Une telle procédure d'équilibrage de charge est obtenue grâce à la SNGF mécanisme de dispersion des paquets dans un relais, SPEED ne considère pas l'énergie système métrique supplémentaire dans son protocole de routage.

2. Protocole basé sur le multi-chemins

Les protocoles basé sur ce type de routage maintiennent plusieurs chemins afin d'augmenter les performances du réseau. Lorsque le chemin primaire est défaillant, les données vont être acheminées vers la destination via des chemins alternatifs.

3. Routage basé sur la négociation

Ces protocoles utilisent des descriptions de données afin d'éliminer les transmissions de données redondantes, par une négociation préalable entre la source et la destination. Cette procédure garantit que seules les informations utiles qui seront transmises.

- **SPIN (Sensor protocol for information via negotiation)**

Proposé par (Heinzelman et al, en 1999). SPIN est parmi les premiers travaux qui poursuivent un axée sur les données et mécanisme de routage. L'idée derrière SPIN est au nom des données à l'aide de descripteurs de niveau ou de méta-données. Avant la transmission, les métadonnées sont échangées entre capteurs via une donnée de mécanisme d'annonce, qui est l'élément clé de SPIN. à la réception de nouvelles données, chaque noeud annonce à ses voisins qui est le voisins intéressés, C'est-à-dire ceux qui n'ont pas les données, récupèrent les données en envoyant un message de demande. SPIN résout les problèmes classiques des inondations telles que les informations redondantes , Chevauchement des zones de détection et de ressources, ainsi la réalisation d'un lot de l'efficacité énergétique. Il n'y a aucune norme méta-format de données et il est supposé

à être spécifiques à une application, par exemple en utilisant un Application au niveau cadrage.

En SPIN il existe trois messages pour échanger des données entre les noeuds qui sont :

- ADV : message pour permettre au capteur d'Annoncer le méta-données, les méta-données peuvent décrire plusieurs aspects comme le type des données et la localisation de son origine.
- REQ : message à Demande des données spécifiques.
- DATA : message de données Transporter les données réelles .

Lorsqu' un noeud voulant émettre une donnée commence par envoyer un paquet ADV. Ce paquet ADV consiste d'une méta-données sur les données à émettre. Les noeuds qui reçoivent ce paquet vérifient si les données les intéressent. Si oui, ils répondent par un paquet REQ. Le noeud qui a initié la communication envoie alors un paquet DATA pour chaque réponse REQ reçue [16].

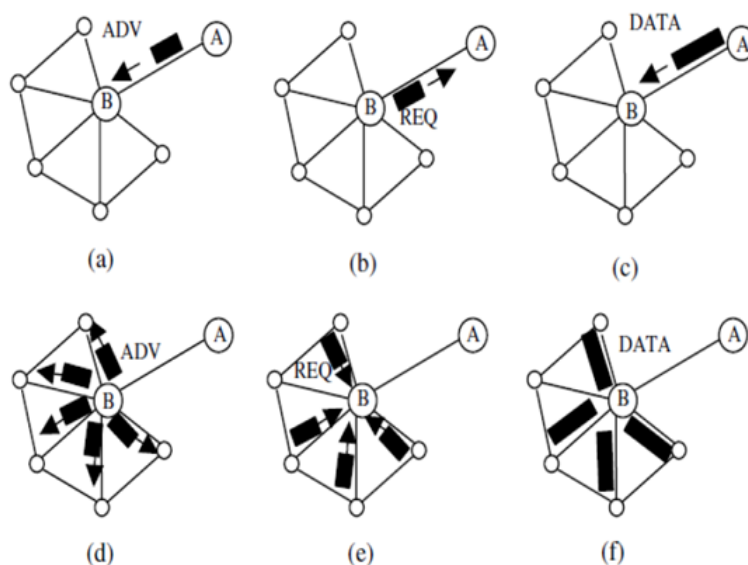


FIGURE 2.5 – Protocole SPIN[16].

Discussions

L'un des avantages de SPIN est que Les modification topologique sont localisées depuis chaque noeud, qui doit savoir son single-hop voisins. SPIN donne Un facteur de 3,5 [16] moins d'inondations en termes d'énergie et méta-données négociation presque la moitiés des données redondantes. Toutefois, le mécanisme d'annonce ne peut pas garantir la livraison des données, par exemple, si les noeuds qui sont intéressés par les données sont très loin du noeud source, et les noeuds entre la source et la destination ne sont pas intéressés par ces données, ces données ne seront pas être livrés à la destination du tout. Donc, SPIN n'est pas un bon choix pour les applications.

4. Routage basé sur le flux de données dans le réseau

Dans ce type de protocole, la phase d'établissement de route est modélisée et résolue comme un problème de demande de flux de données où le flot représente la route que les paquets empruntent, et le demande représente le taux auquel les paquets sont produits par les différents noeuds [16].

2.4.3 Classification selon l'établissement de la route

Suivant la manière de la création et la maintenance de routes lors de l'acheminement des données, les protocoles de routage peuvent être classés en trois catégories : les protocoles proactifs, les protocoles réactifs et les protocoles hybrides.

1. Protocole proactif

Les protocoles de routage proactifs essaient de maintenir les meilleurs chemins existants vers toutes les destinations possibles au niveau de chaque noeud du réseau. Pour ce faire, les noeuds du réseau maintiennent des tables de routage pour toutes les destinations indépendamment de l'utilité des routes.

- **DSDV (Destination Sequenced Distance Vector)**

Protocole DSDV est basé sur l'algorithme de routage Bellman-Ford . Il s'agit d'un protocole proactif. Les itinéraires entre les noeuds du réseau sont toujours en cours maintenus et mis à jour. Chaque noeud du réseau tient à jour une table de routage qui contient des Informations sur l'ancienne route, la distance la plus courte, ainsi que le premier noeud sur le plus court chemin vers les autres noeuds du RCSF . Afin d'empêcher la formation des boucles, un numéro de séquence qui est émis par le noeud de destination, les tags de chaque entrée dans le réseau. Le numéro de séquence pour chaque noeud est choisit au hasard et il est généralement le même nombre. Chaque noeud a pour mettre à jour périodiquement son numéro de séquence et dans la mise à jour normale, le numéro de séquence est augmenté de deux. Si un noeud découvre un chemin a expiré et veut envoyer une mise à jour à ce sujet à ses voisins, ce n'est qu'alors qu'il hausse le numéro de séquence du noeud déconnecté par un seul. Il existe deux catégories classées pour les mises à jour de routage : vidage complet mise à jour et mise à jour incrémentielle. Dans l'ancien, la table de routage complète est transmise par le noeud. Afin de réduire le potentiel du trafic, ce type de mise à jour doit être utilisé uniquement dans la situation où des sommes énormes topologique sont modifiées. Alors que, dans le cas de mises à jour incrémentielles, le noeud envoie uniquement les entrées qui ont été modifiés depuis la dernière mise à jour [25].

Discussions

Dans une évolution rapide, le réseau incrémentielle pourrait grandissent et deviennent de plus en plus gros qui affectent les performances du réseau. Donc les vidages complets sont préférées dans une modification rapide réseau .

2. Protocole réactif

Les protocoles de routage réactifs maintiennent des routes à la demande. Lorsque le réseau a besoin d'une route, une procédure de découverte est lancée. Une fois que la route n'est plus utilisée, elle sera immédiatement détruite ce qui permet une conservation d'énergie.

- **AODV (Ad-hoc On Demand Distance Vector)**

Proposé par (Charles et al , en 1999). AODV est une amélioration de l'algorithme DSDV. C'est un protocole de routage réactif qui utilise une approche au- demande pour trouver et établir des routes. AODV maintient les routes tant et aussi longtemps qu'ils sont requis par les noeuds sources, et il est considéré comme l'un des meilleurs protocoles de routage en termes de puissance du consommation et d'établir le chemin le plus court. Toutefois, il est principalement utilisé pour les réseaux ad hoc, Mais aujourd'hui, il est largement utilisé en RCSFs ainsi dans AODV, chaque noeud périodiquement diffuse des messages HELLO à ses voisins puis utilise ces voisins pour établir le routage et l'envoi de messages. Dans le cas où il y a un message qui veulent être envoyée à certains noeuds qui ne sont pas voisins pour le noeud source, le noeud source trouve un chemin vers la destination en envoyant un message de demande de routage (RREQ) à ses voisins . Bien que le trafic de contrôle les messages sont considérablement réduits en protocoles AODV en raison du fait que les noeuds ouvrent un processus de découverte de route seulement quand c'est nécessaire, les noeuds voisin répondent par un RREP [25].

Discussions

Avec AODV, s'il existe plusieurs routes possibles de la source vers la destination, AODV choisit la route la plus courte (la route où il y a un minimum de sauts). Si un tour de routage échoue, la source relance un nouveau RREQ avec un temps T plus important. Si plusieurs séries de Route Request échouent, alors aucune route ne peut être trouvée.

3. Les protocoles hybrides

Les protocoles hybrides combinent les deux idées des protocoles proactifs et réactifs. Ils utilisent un protocole proactif pour avoir connaissance du proche voisinage (par exemple

le voisinage à deux ou trois sauts). Au-delà de la zone de voisinage, le protocole hybride fait appel à un protocole réactif pour chercher des routes.

2.4.4 Classification selon l'initiateur de la communication

La communication dans un réseau de capteurs peut être initiée par les noeuds sources ou par les noeuds destinataires.

1. Communication lancée par la source

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

2. Communication lancée par la destination

Les protocoles où la communication est initiée par la destination, utilisent un modèle de livraison de donnée basé sur les requêtes. Les noeuds sources répondent aux requêtes envoyées par la destination.

2.5 Les métriques de mesure de l'efficacité des protocoles de routage dans les RCSFs

Cette section présente les métriques communes utilisées pour mesurer l'efficacité des protocoles de routage. Les premiers (le nombre de sauts " Hop-count ", le temps de traverser un saut " Per hop round trip time " et la différence en temps d'arrivée de deux paquets par saut " Per hop packet pair delay ") sont applicables à toutes les architectures des réseaux statiques sans fil, tandis que les cinq postérieures (la notion de coût " Cost awarness ", la notion de puissance " Power awarness ", la notion de coût-puissance " Power-Cost awarness ", le temps du premier noeud à mourir " Time for first node to die " et le temps du dernier noeud à mourir " Time for last node to die ") sont significatives seulement pour les environnements soumis à des contraintes d'énergie [26].

- **Le nombre de sauts** : C'est la métrique la plus typique qui est utilisée dans la gestion des réseaux. Elle représente le nombre des noeuds traversés par une transmission pendant le transfert des données depuis la source à la destination. L'inconvénient principal de

cette métrique est qu'elle ne mesure pas ou ne tient pas en compte la largeur de la bande passante disponible entre les noeuds. Elle ignore les longs chemins (qui ont un nombre élevé de sauts) malgré qu'il y a des chemins parmi eux qui peuvent avoir une passante plus large que la bande passante des courts chemins [26].

- **Le temps de traverser un saut** : Cette métrique mesure le temps d'aller-retour des requêtes envoyées aux noeuds voisins. Cette métrique peut être calculée en ayant un noeud qui va envoyer un paquet de requêtes avec une estampille " timestamped " à l'un de ses voisins chaque 500 ms. Quand le voisin reçoit le paquet, il le transmet de nouveau à l'expéditeur. En comparant le timestamped avec la durée du retour, la qualité du lien peut être évaluée. Naturellement, les résultats de ce test peuvent être altérés par le temps d'attente " queing delay " ou la charge sur les deux noeuds [26].
- **La différence en temps d'arrivée de deux paquets par saut** : Cette métrique est une amélioration de la métrique précédente, car elle réduit le temps d'attente qui peut modifier les résultats. Cette métrique peut-être calculée, en ayant un noeud examinateur qui va envoyer à l'un de ses voisins deux requêtes toutes les deux secondes, tel que la première requête doit être envoyé avant la deuxième. Le récepteur calculera la différence de temps entre la réception des deux paquets et fera un rapport à l'expéditeur, ce dernier va maintenir ces différences de temps [26].
- **La notion du coût "Cost awareness"** : représente une technique pour minimiser la consommation d'énergie dans le routage dans laquelle nous essayons de prolonger la durée de vie d'un noeud. Les choix des opérations de routage que le noeud fera sont une fonction relative à son énergie de batterie restante. Afin d'utiliser " Cost awareness " en tant qu'une métrique, on doit calculer la quantité d'énergie consommée pour chaque route imposée au réseau. Plus la consommation d'énergie est minimisée plus les tâches de routage peuvent être accomplies par le réseau/noeud avant qu'il soit défaillant [27].
- **La notion de puissance "Power Awareness"** : représente une technique pour minimiser la consommation d'énergie. Elle essaye de réduire au minimum l'énergie totale qui a été dépensée lors de l'envoi d'un message depuis sa source à sa destination [27]. Afin d'utiliser " Power awarness " en tant qu'une métrique, on doit attribuer un poids, basé sur la distance, sur chaque saut possible entre les noeuds du réseau.
- **La notion de coût-puissance** : Cette métrique est la combinaison des deux métriques précédentes. Elle vise à réduire au minimum l'énergie consommée dans tout le réseau et en même temps elle évite qu'un noeud ait une quantité d'énergie limitée [27].

- **Le temps du premier noeud à mourir** : cette métrique détermine le temps auquel le premier noeud épuise complètement son énergie [28], elle n'est pas concernée par la défaillance d'un noeud dû à des raisons techniques.
- **Le temps du dernier noeud à mourir** : C'est l'opposé exact de la métrique précédente, celle-ci enregistre le temps où le dernier noeud du réseau a consommé toute son énergie [?]. En d'autres termes, cette métrique mesure la durée de vie du réseau.

2.6 Tableau comparatif des protocoles de routage

Nous présentons ici la classification des protocoles de routage dans les réseaux de capteurs sans fil étudiés dans ce chapitre ; le tableau suivant représente une synthèse sur ces protocoles :

Protocole/type	À plat	Hierarchique	Localisation	QoS	Négociation	Proactif	Réactif
Flooding	oui	-	-	-	-	-	-
Gossiping	oui	-	-	-	-	-	-
SGDF	oui	-	-	-	-	-	-
LEACH	-	oui	-	-	-	-	-
LEACH-C	-	oui	-	-	-	-	-
HEED	-	oui	-	-	-	-	-
MHEED	-	oui	-	-	-	-	-
GAF	-	-	oui	-	-	-	-
SAR	-	-	-	oui	-	-	-
SPEED	-	-	-	oui	-	-	-
SPIN	-	-	-	-	oui	-	-
DSDV	-	-	-	-	-	oui	-
AODV	-	-	-	-	-	-	oui

TABLE 2.1 – Synthèse des protocoles de routage.

2.7 Conclusion

Dans ce chapitre, nous avons dressé un état de l'art sur les principales approches proposées pour résoudre le problème de routage dans les réseaux de capteurs sans fil. Ces approches prennent en considération les limites imposées par l'architecture matérielle des capteurs ; en particulier l'énergie. Nous avons aussi effectué une classification et une étude critique des protocoles de routage proposés dans le cadre des réseaux de capteurs sans fil.

CHAPITRE 3

ETAT DE L'ART : LES PROTOCOLES DE ROUTAGE HIÉRARCHIQUE DANS LES RÉSEAUX DE CAPTEURS SANS FIL

3.1 Introduction

Le routage hiérarchique est une stratégie qui vise à organiser le réseau d'une manière à réduire la dissipation énergétique afin de préserver cette ressource épuisable et difficilement rechargeable. Selon un processus de clustering, les noeuds du réseau sont groupés en clusters. Cela est établi par l'élection d'un chef pour le cluster, puis à déterminer les noeuds qui feront partie de ce dernier. Plusieurs techniques peuvent être utilisées pour le partitionnement du réseau en cluster. En effet, ces techniques permettent à classifier les différents protocoles proposés selon cette approche de routage.

Dans ce chapitre, nous présentons un état de l'art sur les protocoles de routage hiérarchiques proposés pour les réseaux de capteurs et cité leurs avantages. Puis nous présentons les challenges de conception d'un protocole de routage hiérarchique et nous terminerons par une étude détaillée de quelques protocoles hiérarchiques.

3.2 Définition de la notion de clustering

Dans des milieux hostiles et dangereux où l'intervention humaine est impraticable, il n'est pas toujours faisable d'organiser manuellement les noeuds en groupes. Pour cette raison, il y a eu plusieurs travaux de recherche sur des manières de créer des structures logiques

d'organisation. Le clustering est considéré comme un point clé dans la conception de ces réseaux.

Avec cette technique l'ensemble des capteurs est en général divisé en groupes (ou clusters), ou dans chaque groupe un chef (ou cluster head). Ce dernier communique avec les membres de son groupe et les clusters head des autres groupes. La tâche d'un cluster head ne consiste pas seulement à router les données, mais peut être complétée par une agrégation ou une fusion des données. De ce fait, des changements de rôle membre/clusterhead sont généralement appliqués pour répartir la charge de routage et d'agrégation des données entre les capteurs du réseau. De ce fait, l'utilisation du mécanisme de clustering dans les réseaux de capteurs peut contribuer d'une façon considérable à l'économie d'énergie, la réduction de la complexité des protocoles de routage ainsi qu'à une meilleure résistance au facteur d'échelle (scalability). En effet, le routage hiérarchique facilite l'agrégation de données, ce qui conduit à diminuer les redondances et les transmissions inutiles vers la station de base. D'autre part, le clustering permet aux noeuds d'effectuer des transmissions sur des courtes distances avec leurs clusters head, ce qui minimise la consommation d'énergie et optimise l'utilisation des ressources du médium de communication [29].

Nous présentons ci-dessous les principaux composants d'une architecture hiérarchique pour un réseau de capteurs :

- **Noeud capteur** : Le noeud capteur est le composant noyau du réseau. De plus, ces capteurs peuvent jouer des rôles multiples, tels que la sensation simple, stockage, conduite de données et traitement. Cela dépend de leurs capacités en termes de calculs et puissance énergétique sans oublier l'influence du mécanisme d'affectation des rôles adapté par le protocole de clustering [43].
- **Cluster** : le cluster est un ensemble de noeuds qui forme l'unité d'organisation d'un réseau de capteurs. La nature dense de ses réseaux exige de la décomposer en cellules ou clusters afin de simplifier les tâches de communication et répondre aux différentes contraintes [29].
- **Cluster head** : (CHs, leaders, chefs de groupe) se sont souvent nécessaire pour l'organisation des activités dans les clusters. Leurs tâches ne se limitent pas à l'agrégation de données, elle s'étend aussi à l'organisation de la communication intra-cluster et même inter cluster. Ces chefs peuvent être élus par les autres noeuds ou bien sont pré- assignés par le concepteur du réseau. Ils peuvent aussi être juste des noeuds ordinaires, ou bien des noeuds dotés de plus d'énergie [29].
- **Station de base** : la station de base se situe à un niveau supérieur de la hiérarchie d'un réseau de capteurs, elle permet de fournir une liaison entre le réseau et l'utilisateur final [29].

3.3 Avantage de routage hiérarchique

les protocoles de routage hiérarchique sont chargés généralement d'établir des clusters head et de définir la manière dont laquelle les noeuds décident quel cluster head à joindre.

Les buts principaux de routage hiérarchique sont [29] :

- Faciliter le partage de ressource et/ou la synchronisation au sein d'un cluster.
- Optimiser l'utilisation de la bande passante en limitant les interactions inter-cluster et évitant les échanges superflus des messages avec station de base.
- Réduire la taille des tables de routage stocké au niveau des noeuds en localisant le chemin, car la mise en place des routes se fait seulement au niveau des clusters head.
- Agrégation de données transmises au collecteur. Les données qui sont collectées à partir d'un ensemble de noeuds d'un cluster peuvent être fusionnées par un cluster head et par la suite envoyées vers le puits ce qui permet la réduction du nombre de paquets échanges.
- Stabiliser la topologie au niveau des noeuds et donc minimiser les frais de la maintenance de ce dernier. Un noeud ne s'inquiète que sur le choix de cluster head auquel il va être relié. Par conséquent il n'est pas affecté par les changements au niveau de ces chefs.
- Implémenter des mécanismes des clusterisation afin d'augmenter la durée de vie de la batterie, ce qui permet de prolonger la durée de vie du réseau.

3.4 Contraintes et facteurs de conception d'un protocole de routage hiérarchique

Dans cette partie, nous présentons un certain nombre de considérations qui sont indispensables pour la conception d'un protocole de routage hiérarchique pour les réseaux de capteurs. Comme les réseaux de capteurs imposent des fonctions spécifiques et des contraintes sur les ressources, leurs efficacités dépendent considérablement de la qualité de leurs protocoles. Afin de concevoir un protocole de routage efficace, les facteurs détaillés ci-dessous doivent être pris en compte.

- **Énergie limitée** : Les noeuds capteurs ont un stockage d'énergie limité et l'utilisation efficace de cette dernière est un défi à relever. Cette contrainte énergétique doit être prise en compte dans le processus de clustering afin de réduire la dissipation d'énergie entre les noeuds du réseau. Par ailleurs, ce défi énergétique a un impact direct sur la durée de vie du réseau en entier. Pour cela, grouper les noeuds dans les clusters devrait organiser la topologie du réseau et contribuer à réduire la consommation d'énergie de ses noeuds

afin d'augmenter sa durée de vie [29].

- **Formation de clusters** : l'efficacité d'un algorithme de clustering est évaluée en termes de nombre de clusters formés et de leurs stabilités en fonction de la mobilité des noeuds (la capacité de s'adopter au changement de la topologie du réseau). Il existe plusieurs méthodes de formation de clusters. Mais la plus répandue s'exécute sur trois [29].
 - ***Election de cluster head*** : appelé aussi set-up : chaque noeud prend la décision de devenir CH ou bien de se joindre à un CH selon sa connaissance locale de la topologie en utilisant une métrique spécifique ou une combinaison de métrique tels que le plus grand/petit ID dans son voisinage, le degré de connectivité, la puissance de transmission, l'énergie résiduelle, ou bien selon un poids qui représente une combinaison de quelques métriques [29].
 - ***Communication intra-cluster et inter-cluster*** : chaque cluster head se charge des communications à l'intérieur de son cluster et maintient les informations de routage lui permettant de joindre la station de base. Pour atteindre la station de base, les clusters head communiquent directement à cette dernière sinon une communication multi-saut est adoptée en utilisant d'autres clusters head comme noeud relais [43].
 - ***Maintenance des clusters*** : dans le but de s'adapter aux changements de la topologie du réseau, une mise à jour des clusters est dynamiquement réalisée pour reconnaître les nouveaux noeuds ajouter ou ceux disparus (les noeuds qui ont consommé leurs énergies). D'autre part, si le cluster head garde son statut le plus longtemps possible, même s'il ne possède pas par exemple le poids maximum dans son propre cluster alors il perdra son rôle une fois sa batterie sera épuisée [29].
 - ***Qualité de service(QoS)et les opérations temps réel*** : dans certaines applications, la donnée doit être délivrée rapidement après sa capture, sinon elle n'est plus utile et par conséquent, la latence définie pour la livraison des données est une autre condition pour les applications qui sont soumises sous des contraintes de temps. cependant, dans plusieurs applications, la conservation d'énergie, qui est directement liée à la durée de vie du réseau est considérée relativement plus important que la qualité des données envoyées. Pendant que l'énergie s'épuise, le réseau exige de réduire la qualité des résultats afin de réduire la diminution d'énergie dans les noeuds, et par conséquent augmenter la durée de vie du réseau [29].

- ***Synchronisation*** : l'une des limites des réseaux de capteurs sans fil est la limite d'énergie des noeuds, la transmission par tranche de temps en utilisant (TDMA) permet aux noeuds de programmer régulièrement des intervalles de sommeil afin de réduire au minimum la consommation d'énergie, de tels procédés exigent des mécanismes de synchronisation pour installer et maintenir le plan de transmission, ce qui aura des effets considérables sur la vie et l'ensemble des performances du réseau [29].
- ***Agrégation de données*** : une des caractéristiques majeures des réseaux de capteurs est la densité des noeuds. Cependant, il y a souvent de multiples capteurs qui recueillent la même information dans le réseau. L'agrégation sert à combiner les données qui proviennent de plusieurs noeuds en une information significative ce qui éliminera la redondance. Entre autres, l'utilisation des techniques d'agrégation de données dans les protocoles de clustering réduit la quantité des données transmises, minimise considérablement la communication entre les noeuds, résout le problème d'implosion dans les tables de routage et allège en conséquence la congestion [29].
- ***Mécanisme de réparation*** : En raison de la nature des réseaux de capteurs sans fil, ils sont souvent sujets à la mobilité des noeuds, à la mort d'un noeud ou à des interférences, toutes ces situations peuvent entraîner des échecs de liens. Lorsqu'on examine les méthodes de création de cluster, il est important de se pencher sur les mécanismes à mettre en place pour le rétablissement de lien et assuré une communication fiable.
- ***Capacité limitée*** : La taille physique des capteurs et la plus petite quantité d'énergie stockée dans un noeud limitent beaucoup la capacité de ces capteurs en termes de communication et puissance de traitement. Un bon protocole de clustering devrait se servir des ressources partagées entre les noeuds tout en considérant la limitation de leurs capacités [29].

3.5 Taxonomie d'attribut de clustering

Un ensemble d'attributs de clustering est potentiellement identifiable pour classer par catégorie et différencier les protocoles de clustering dédiés aux réseaux de capteurs sans fil.

1. Caractéristiques du clusters

souvent le processus de clustering s'efforce à parvenir à un certain nombre de caractéristiques pour les clusters générés. ces caractéristiques peuvent être liées à la structure interne du cluster ou à la façon dont il se rapporte avec d'autres clusters. ci-dessous les attributs les plus pertinents [29] :

- **Le nombre de clusters** : les plans de regroupement peuvent être classés en deux types : celles fixes et variables. Dans l'ancien système, l'ensemble des clusters head sont prédéterminés et le nombre de clusters est fixe. Toutefois, la sélection aléatoire des CHs par les capteurs déployés résulte en un nombre variable de clusters.
- **La topologie Intra-clusters** : la communication entre un noeud et son cluster head dans un cluster est directe. Néanmoins, la connection via multi-sauts entre les noeuds et leurs CH est parfois exigée, particulièrement quand la portée de la communication dans le cluster est limitée.
- **Connectivité inter-CHs** : une grande partie des travaux publiés sur les réseaux de capteurs présume que le CH peut atteindre directement la station de base. D'autre part, si le CH n'a pas la capacité de communication à longues distances, la connexion des CHs à la station de base doit être assurée par l'établissement de routes inter-CHs jusqu'à la station de base.

2. Caractéristiques d'un cluster head

- **Différence de capacités** : Basé sur l'uniformité de la cession de l'énergie pour les noeuds de capteurs, les systèmes de clustering dans les réseaux de capteurs peuvent être classés en ceux homogènes ou hétérogènes. Dans les systèmes homogènes, tous les noeuds de capteurs sont affectés aux ressources égale de l'énergie, de calcul et de communication et CH sont désignés selon un mode aléatoire ou d'autres critères. Cependant, les noeuds de capteurs sont affectés aux capacités inégales dans un environnement hétérogène, où les rôles du CH sont pré-affectés aux noeuds capteurs avec plus de capacités [10].
- **Mobilité** : lorsque le CH est mobile, les noeuds membres sont modifiés dynamiquement stationnaire tendent à avoir des groupes stables et à faciliter la gestion du réseau. Parfois, le CH se déplacer sur des distances limitées et à se repositionner afin d'avoir une meilleure performance pour le réseau [10].
- **Rôle** : Un CH ne peut tout simplement agir en tant que point de relais pour le trafic généré par les noeuds de capteurs dans son cluster ou effectuer agrégation / fusion des l'informations rassemblées. Parfoi, il peut agir comme un puits / BS qui prend des mesures sur la base des phénomènes ou des cibles détectées [10]. Il est utile de mentionner, parfois un CH agit dans plus d'un rôle.

3. Processus de clustering

Les caractéristiques de processus de clustering variaient de manière significative selon :

- **La manière du contrôle** : le regroupement des méthodes de routage dans les réseaux de capteurs peuvent être regroupés dans celles centralisées, distribuées et hybrides. Dans les méthodes centralisées, d'un évier ou CH exige une information globale du réseau ou du cluster pour contrôler le réseau ou le cluster. Dans les approches distribués, un noeud de capteur est capable de devenir un CH ou se joindre à un groupe formé de sa propre initiative sans information globale du réseau ou du cluster. Régimes hybrides sont composés d'approches centralisées et distribuées. Dans ce contexte, les approches distribués sont utilisées pour la coordination entre chs, et les manières centralisées sont effectuées pour le SHC pour construire des clusters individuels [29].
- **Nature d'exécution** : Compte tenu de la nature de l'exécution de la formation du groupe, les modes de clustering dans les réseaux de capteurs peuvent être classés en deux catégories : ceux probabilistes ou itératif. Lors d'un clustering probabiliste, une probabilité attribuée à tous les noeuds de capteurs est utilisée pour déterminer les rôles des noeuds de capteurs. En d'autres termes, chaque noeud capteur peut décider de façon indépendante sur ses propres rôles. Néanmoins, chaque noeud doit attendre un certain nombre d'itérations est atteint ou pour certains noeuds de décider de leurs rôles avant de prendre une décision de manière itérative de clustering [29].
- **Temps de convergence** : Compte tenu du temps de convergence, les méthodes de regroupement dans les RCSFs peuvent être regroupées en ceux de temps de convergence variable et constante. Le temps de convergence dépend du nombre de noeuds dans le réseau dans les algorithmes de convergence variables, qui accueillent bien à des réseaux à petite échelle. Après un nombre fixe d'itérations, les algorithmes de temps de convergence constants certainement convergent indépendamment de l'échelle des réseaux... [29].
- **Paramètres pour l'élection de CH** : Basé sur les paramètres utilisés pour l'élection de CH, les approches de clustering peuvent être classés comme déterministe, adaptative et aléatoires. Dans les schémas déterministes, les attributs particuliers des noeuds de capteurs sont envisagées, comme l'identifiant (ID), nombre de voisins dont ils disposent. Dans la classe d'adaptation, les CHs sont élus parmi les noeuds de capteurs déployés avec un poids plus élevé, ce qui inclut, comme l'énergie résiduelle, les coûts de communication, et etc. Et dans les modes aléatoires, principalement utilisé

dans les algorithmes de clustering sécurisés, les CHs sont élus de manière aléatoire sans tenir compte des autres paramètres tels que l'énergie résiduelle, les coûts de communication, etc [29].

- **Objectifs de clustering** : quelques objectifs ont été poursuivis pour la construction de cluster, telles que l'agrégation / fusion des données , l'équilibrage de charge, tolérance aux pannes, la garantie de la connectivité, l'extension de la durée de vie, la qualité de service, etc. Par conséquent, les méthodes clustering dans les réseaux de capteurs peuvent être classés dans les catégories ci-dessus en fonction de différents objectifs. Il est à noter que l'algorithme de clustering, de manière générale, a plus d'un objectif... [30].

3.6 Les protocoles de routage hiérarchique

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

Ce protocole a été proposé par **Lindsey** et al en 2002 dans [31]. Il vient pour améliorer le protocole de routage LEACH, dont l'idée principale est de former une chaîne entre les noeuds de sorte que chaque noeud envoie ses données à son voisin le plus proche et à tour de rôle utilisant la transmission point à point. Les données collectées sont transmises d'un noeud à un autre qui les agrège jusqu'à ce qu'elles arrivent à un noeud particulier qui les transmet à la station de base. De ce fait un seul chemin va-t-êtré formé du premier noeud vers la station de base.

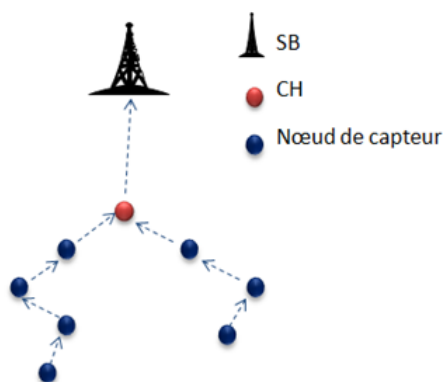


FIGURE 3.1 – Le protocole PEGASIS.

Discussions

Les résultats de simulation dans [31] ont montré que PEGASIS est plus performant par rapport à LEACH en terme de réduction de la consommation d'énergie, ce qui lui permet de prolonger de deux à trois fois la durée de vie du réseau de capteurs.

PEGASIS suppose que chaque noeud a la possibilité de transmettre les données capturées à la station de base directement ou à travers un noeud intermédiaire.

3.6.2 EEPSC(Energy Efficient Protocol with Static Clustering for wireless sensor network)

EEPSC est un protocole de routage hiérarchique basé sur le clustering statique qui a été proposé dans [32]. C'est une amélioration du protocole LEACH. Il se base sur le déploiement des noeuds, ainsi que la station de base se trouve à l'une des extrémités du réseau. Ce protocole partitionne le réseau en clusters statiques, élimine la surcharge de clustering dynamique et utilise des clusters head temporaires pour équilibrer la charge de l'énergie entre les noeuds qui ont de grandes capacités d'énergie dans le but de prolonger la durée de vie des réseaux de capteurs sans fil.

Le principe du protocole EEPSC se divise en trois phases :

- **Phase d'installation (sélection du clusters) :**

En premier lieu, la station de base diffuse $k-1$ messages différents avec différentes puissances d'émission afin d'avoir k clusters. Par la diffusion du message $k=1$, les noeuds qui reçoivent ce message seront dans le même cluster, ils mettent le groupe à **ID-K** et informent la station de base qu'ils sont les membres de cluster k par l'envoi du message "Join-REQ" en utilisant la méthode d'accès CSMA (Carrier Sense Multiple Access). De même pour la diffusion du message $k=k-1$. Après, les noeuds qui n'ont pas rejoint aucun clusters, collectent leur ID-cluster au message k et informent la station de base qu'ils sont les membres du dernier cluster. Par la suite, la station de base choisit aléatoirement un cluster head temporaire **CH-T** pour chaque cluster, et utilise le TDMA pour diviser le temps en tranches (time slot), chaque noeud du cluster aura une tranche de temps où il peut transmettre les données.

- **Phase d'élection de clusters head (CHs) :**

Après la formation des clusters, au début de chaque tour, chaque noeud envoie son énergie résiduelle au **CH-T** du même cluster durant sa tranche de temps. Par

la suite, le CH-T choisit le noeud qui a l'énergie résiduelle maximale comme cluster head (CH) pour le tour actuel afin de collecter et d'agrèger les données transmises à la station de base. De plus, il choisit le noeud qui a l'énergie résiduelle minimale comme cluster head temporaire pour le prochain tour en lui envoyant un message "round-start" incluant l'identifiant (ID) de cluster head du tour actuel.

- **Phase de communication :**

Dans cette phase, chaque noeud du cluster envoie ses données au cluster head CH durant sa tranche du temps. Ensuite, le CH collecte et agrège les données. Après, il les envoie directement (un saut) à la station de base.

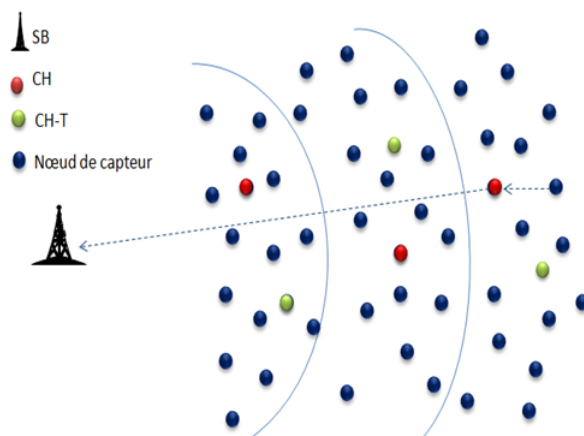


FIGURE 3.2 – Le protocole EEPSC.

Discussions

EEPSC offre une meilleure augmentation de la durée de vie du réseau par rapport à LEACH, ce qui augmente le nombre de paquets reçus par la station de base en utilisant le clustering statique et les clusters head temporaires.

Les résultats de simulation dans [32] ont montré que le nombre de noeuds restant en vie dans EEPSC est performant de 45% par rapport à LEACH (les noeuds meurent rapidement).

EEPSC utilise la communication directe (un seul saut) vers la station de base, ce qui pose problème dans les grands réseaux.

3.6.3 TEEN (Threshold sensitive Energy Efficient sensor Network protocol)

Proposé (A. Manjeshwar, D.P. Agrawal en 2001). Le protocole TEEN utilise la technique de clustering, il est similaire à celui de LEACH, mais TEEN est un protocole destiné

pour les RCSF réactifs, dans le processus d'établissement du cluster, le protocole effectue deux seuil (hard et soft). après la sélection du cluster Head les données sont transmises à travers le chemin de TDMA, deux paramètres de seuils doivent être diffusés, le seuil Hard (Hard Threshold) est le minimum de données transmises, et le seuil soft (soft Threshold) spécifie le changement de la portée des données captées. Lorsque le noeud écoute le médium continuellement, et lorsque la valeur captée du paramètre contrôlé dépasse le seuil Hard, le noeud transmet les données au cluster Head, et met l'information comme le nouveau seuil Hard et la stocke dans une valeur contrôlée SV (sensed values). Puis, seulement la valeur courante du paramètre contrôlé est supérieure au seuil Hard ou diffère du SV d'une quantité égale ou plus grande que la valeur du seuil soft. Le protocole réduit la quantité des données transférées par l'établissement du seuil soft et le seuil Hard significativement, lequel peut contrôler le nombre d'événement inattendu et les endroits chauds[33].

Discussions

L'inconvénient principal de ce protocole est que, si les seuils HT et ST ne sont pas reçus, les noeuds ne communiqueront jamais, et aucune donnée ne sera transmise à l'utilisateur, ainsi la station de base ne connaît pas les noeuds qui ont épuisé leur énergie. TEEN ne convient pas aux applications qui nécessitent des envois périodiques de données.

3.6.4 APTEEN (Adaptive Threshold-sensitive Energy Efficient sensor Network protocol)

Pour remédier aux limitations du protocole TEEN, les auteurs (A. Manjeshwar, D.P. Agrawal en 2002) ont proposé une extension de TEEN appelée APTEEN.

APTEEN est un protocole hybride qui change la périodicité et les valeurs seuils utilisées dans TEEN selon les besoins de l'utilisateur et le type d'application. Dans APTEEN, les cluster-heads transmettent à leurs membres les paramètres suivants :

- l'ensemble de paramètres physiques auxquels l'utilisateur est intéressé pour obtenir des informations (A),
- les seuils : seuil Hard HT et seuil Soft ST,
- un Schedule TDMA permettant d'assigner à chaque noeud un intervalle fini de temps appelé slot,
- un compteur de temps (CT) : c'est la période de temps maximum entre deux transmissions successives d'un noeud.

Dans APTEEN, les noeuds surveillent en continu l'environnement. Ainsi, les noeuds qui détectent une valeur d'un paramètre qui dépasse le seuil HT, transmettent leurs données. Une fois qu'un noeud détecte une valeur qui dépasse HT, il ne transmet les données au cluster head que si la valeur de ce paramètre change d'une quantité égale ou supérieure à ST. Si un noeud

ne transmet pas de données pendant une période de temps CT, il devrait faire une capture de données et les retransmettre [34].

Discussions

APTEEN offre une grande flexibilité qui permet à l'utilisateur de choisir l'intervalle de temps CT, et les valeurs seuils HT et ST pour que la consommation d'énergie soit contrôlée par la variation de ces paramètres. Cependant, APTEEN nécessite une complexité supplémentaire pour implémenter les fonctions de seuils et de périodes de temps CT. Ainsi, le surcoût et la complexité associés à la formation des clusters à plusieurs niveaux par TEEN et APTEEN sont assez élevés.

3.6.5 PEGASIS Hiérarchique

Dans [35], Z.ALIQUAT et al ont défini le protocole PEGASIS Hiérarchique qui est une amélioration du protocole PEGASIS. Dans ce protocole, chaque noeud transmet ses données à son proche voisin, ce dernier à son tour les transmet à son proche voisin jusqu'à atteindre le cluster head qui les agrège et les transmet par la suite directement à la SB.

L'algorithme se déroule en n périodes z . Chaque période est constituée d'une phase d'initialisation et d'une phase de transmission.

Au début de chaque période qui commence à l'instant t , les nouveaux CHs sont choisis selon :

- Calculer la probabilité $P_i(t)$ de devenir CH pour chaque noeud i , où la probabilité $P_i(t)$ est :

$$T(n) = \begin{cases} \frac{NbrdeCHdsir}{NbrdeCHquin'estpasencoretlusCH} : \text{silesnoeudshomogenesentermed'nergie.} \\ \frac{Energiersiduelle}{Energietotale} : \text{silesnoeudshetrogenesentermed'nergie.} \end{cases} \quad (3.1)$$

Chaque noeud i génère un nombre aléatoire entre 0 et 1. Si ce nombre est inférieur à $P_i(t)$, le noeud i deviendra CH durant cette période, la phase d'initialisation est composée de 3 sous phases : phase d'annonce, phase d'organisation des groupes et enfin de phase recherche du voisin proche.

Dans la phase de transmission, les clusters heads envoient directement les informations à la station de base.

La figure suivante illustre son principe général :

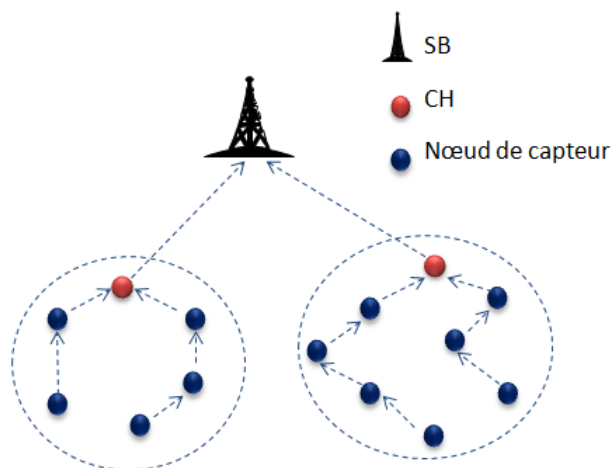


FIGURE 3.3 – Le principe de PEGASIS Hiérarchique.

Discussions

Le routage à un seul saut au niveau du cluster head (CH) présente un inconvénient très important de PEGASIS Hiérarchique dans les CHs épuisent rapidement leur énergie, car ils peuvent être très loin de la station de base.

3.6.6 PEGASIS HM(Hop-Multiply)

Ce protocole a été proposé dans [35]. C'est une extension de PEGASIS Hiérarchique dont le but est de réduire les délais de transmission des paquets vers la station de base. Dans le protocole PEGASIS Hiérarchique, l'organisation des noeuds appartenant à la même grappe (cluster) sous forme d'une chaîne permet d'améliorer et de réguler la dissipation d'énergie, ce qui permet de réduire la charge sur le cluster head. En effet, les noeuds communiquent uniquement avec leurs voisins et non pas directement avec le CH.

L'algorithme se déroule en "périodes". Chaque période est constitué d'une phase d'initialisation et d'une phase de transmission.

- **La phase d'initialisation** : Cette phase est composée de trois sous phases : d'annonce, d'organisation des groupes et enfin de recherche du voisin proche.

Au début de chaque période qui commence à l'instant t , les nouveaux CHs sont choisis selon :

- Chaque noeud i génère un nombre aléatoire entre 0 et 1. Si ce nombre est inférieur à

$P_i(t)$, le noeud deviendra CH durant cette période ; avec :

$$T(n) = \begin{cases} \frac{NbrdeCHdsir}{NbrdeCHquin'estpasencoretlusCH} : \text{silesnoeudshomognesentermed'nergie.} \\ \frac{Energiersiduelle}{Energietotale} : \text{silesnoeudshtrognesentermed'nergie.} \end{cases} \quad (3.2)$$

- **La phase de transmission** : Dans cette phase, chaque noeud transmet ses données à son proche voisin, ce dernier les transmet à son voisin jusqu'à atteindre le cluster head. Ensuite, le cluster head à son tour agrège les données reçues et les transmet par la suite au cluster head du niveau inférieur jusqu'à atteindre la SB.

```

Si (CH) alors
  Det_Niv ()
  Si (chID appartient au niveau2) alors
    Trouver nexthopID_ dans "niveau2_"
  Sinon
    Si (chID appartient au niveau1) alors
      Trouver nexthopID_ dans "niveau1_" ;
    Sinon Le "nexthopID_" est "bsID";
  Fin Si
Fin Si
SendDataNexthop ()
Si (nexthopID_ != bsID) alors
  Ajouter "chID" à sa liste
Fin Si
Fin Si

```

FIGURE 3.4 – Algorithme de recherche du prochain saut.

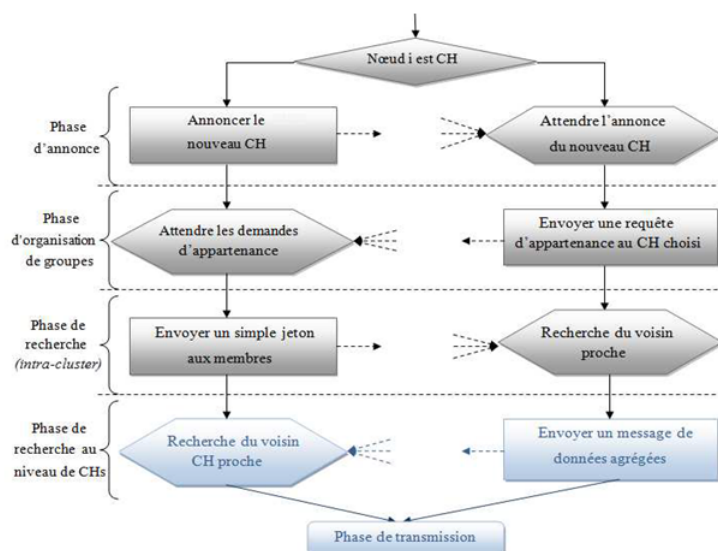


FIGURE 3.5 – Organigramme du PEGASIS HM.

Dans ce protocole un seul CH le plus proche qui communique avec la SB comme l'illustre la figure suivante :

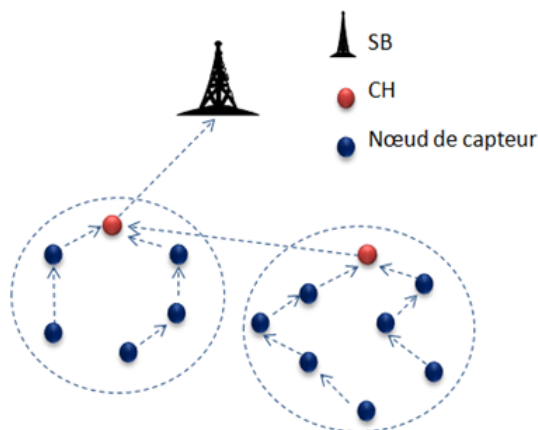


FIGURE 3.6 – Le protocole PEGASIS HM.

Discussions

Le protocole PEGASIS HM est une amélioration de PEGASIS Hiérarchique qui a apporté des avantages tels que la réduction de l'énergie moyenne consommée.

PEGASIS HM se base sur un routage multi-sauts, ce qui permet d'acheminer les paquets des clusters head les plus éloignés de la station de base. Il prolonge la durée de vie du réseau par rapport au protocole original PEGASIS Hiérarchique et ceci est dû aux communications multi-sauts entre les CHs et la station de base ainsi que l'agrégation des données aux niveaux CHs qui permet de réduire la quantité de paquets circulant dans le réseau.

La construction des chaînes dans PEGASIS HM présente un inconvénient de tomber sur les mêmes chemins lorsque le noeud de capteur choisit son voisin plus proche en terme de distance, ce qui mène à l'épuisement rapide des noeuds qui forment ces chemins.

3.6.7 DECSA (Distance-Energy Cluster Structure Algorithm) :

Cet algorithme a été proposé par **Zhu Yong et al** dans [36]. Il est basé sur la **distance** et l'**énergie résiduelle** des noeuds de capteurs, il est similaire à celui de LEACH. Le protocole DECSA est déroulé en périodes et chaque période est divisé en étape initialisation, et étape de travail.

- **Etape d'initialisation :**

La différence entre LEACH et DECSA est que DECSA utilise les paramètres de distance et l'énergie résiduelle pour choisir le cluster head.

Dans cette étape le processus de sélection du cluster head suit deux étapes : l'élection du cluster head (CH) , puis l'élection du station de Base Cluster Head (BCH) (càd le

cluster head qui communique avec la station de base).

1. Election du clusters head :

En premier lieu chaque noeud de capteur genère un nombre aléatoire entre 0 et 1 et il le compare avec le seuil T_n de LEACH ,si le nombre aléatoire est inferieur à T_n alors ce capteur deviendra le **Faux cluster head** dans cette période afin de former des clusters.

Après avoir selectionner le faux cluster head chaque noeud de même cluster y compris le faux cluster head vont calculer leur $K(i)$ pour selectionner le cluter head avec la formule de $k(i)$ qui est determiner comme suit :

$$K(i) = \frac{E_r(i)}{d(i)}$$

Où :

$K(i)$ est le seuil de l'élection de CH.

$E_r(i)$ est l'énergie résiduelle du capteur i .

$d(i)$ est la distance entre le capteur i et la station de base.

Puis chaque capteur de même cluster compare son $k(i)$ avec les autres , et celui qui a sa valeur de $k(i)$ maximal deviendra le cluster head pour la période courante.

2. Election de station de Base Cluster Head :

Après la sélection des CHs,on sélectionne parmi eux le BCH (station de Base Cluster Head). Chaque CH(i) calcule son TBCH(i).

$$\text{Avec : } TBCH(i) = \frac{E_r(i)}{E_0} + \frac{E_r(i)}{d(i)}.$$

où : $E_r(i)$ est l'énergie résiduelle du capteur i , E_0 est l'énergie initiale du capteur dans le réseau, et $d(i)$ est la distance entre le noeud i et la station de base.

Le BCH selectionné est celui qui a une valeur TBCH(i) maximale.

• Etape de travail (communication) :

Dans cette étape la SB diffuse les messages dans le réseau. Lors de la réception du message accordé avec la valeur de TBCH(i), le BCH sélectionne le maximum(TCBH) comme prochain saut , et le reste des sauts vont être sélectionner de la même façon jusqu'à ce que tout les CH seront connectés, dont le but de réduire la distance entre la station de base et les clusters head (CH) et de consommer moins d'énergie.

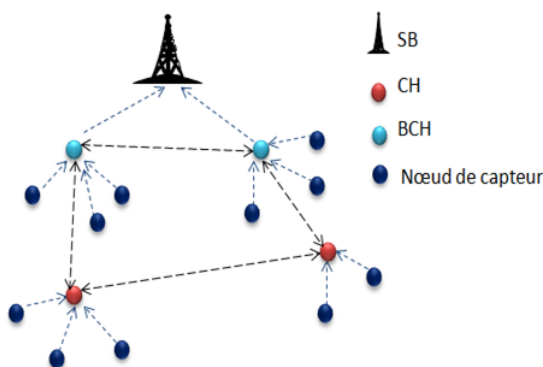


FIGURE 3.7 – Le protocole DECSA.

Discussions

D'après l'évaluation des protocoles LEACH et DECSA dans [20], nous constatons que DECSA est utilisé dans les vastes régions, car le nombre de noeuds dans la période utilisant DECSA est plus grand que celui de LEACH ; la durée de vie des noeuds dans DECSA est prolongée à 31% par rapport à LEACH. De plus, DECSA est plus performant que LEACH en termes d'énergie résiduelle et la distance.

Mais, DECSA utilise la sélection aléatoire des clusters head, ce qui du à avoir des périodes perdus ou de sélectionner des noeuds à peu d'énergie. En plus l'utilisation de TBCH dans la phase de communication crée un inconvénient càd vers les dernières périodes de réseau les BCHs seront très loin de la station de base, donc on peut avoir un envoi en arrière (sens contraire de la SB) vers les BCH ensuite vers la SB ce qui mène à l'épuisement rapide de l'énergie.

3.7 Tableau comparatif pour les protocoles de clustering

Après avoir cité quelques protocoles de routage hiérarchique, le tableau suivant illustre la comparaison entre les protocoles étudié précédemment suivant quelques critères, où la mobilité concerne la station de base qui est fixe pour tout les protocoles, nous considérons aussi l'aggrégation des données qui centralisé est dans les clusters, ainsi pour assurer les contraintes de qualité de service (QoS), et la Scalabilité :

Protocoles/Critères	Mobilité	Localisation	Agrégation de données	QoS	Scalabilité
PEGASIS	SB fixée	oui	centralisé	non	limité
EEPSC	SB fixée	oui	centralisé	non	bonne
TEEN	SB fixée	non	centralisé	non	limité
APTEEN	SB fixée	non	centralisé	non	limité
PEGASIS H	SB fixée	oui	centralisé	non	limité
PEGASIS HM	SB fixée	oui	centralisé	non	bonne
DECSA	SB fixée	oui	centralisé	non	limité

TABLE 3.1 – Synthèse des protocoles de routage hiérarchique.

3.8 Conclusion

Dans ce chapitre, nous avons présenté quelques protocoles de routage hiérarchique étudiés, ainsi la notion de taxonomie, et dans ce qui suit nous allons nous inspirer sur ces protocoles pour proposer un nouveau protocole de routage hiérarchique à basse consommation énergétique.

CHAPITRE 4

CONTRIBUTION : DECSA-G (DISTANCE-ENERGY CLUSTER STRUCTURE ALGORITHM BASED ON GRID)

4.1 Introduction

Les réseaux de capteurs sans fil deviennent de plus en plus répandus, ils sont utilisés dans divers domaines. Le développement d'une technique efficace permettant d'économiser la ressource énergétique est un objectif primordial pour les réseaux de capteurs sans fil.

Dans ce chapitre, nous proposons un nouveau protocole de routage hiérarchique à basse consommation d'énergie pour les RSCFs basé sur la technique de clustering statique, et qui porte une amélioration pour les protocoles EEPSC [32] et DECSA[36] en s'intéressant aux inconvénients de ces derniers. Pour le protocole DECSA la sélection aléatoire des clusters head peut tomber sur des noeuds à peu d'énergie, et pour le protocole EEPSC l'envoi direct à la station de base consomme beaucoup d'énergie surtout dans les zones plus étendues.

Dans notre proposition, nous éliminons la sélection aléatoire des clusters head, et pour le regroupement des noeuds nous gardons le même principe que EEPSC, et en améliorant avec un partitionnement horizontal. De plus, nous présentons la notion du multi-saut basé dans DECSA en améliorant dans la communication entre les clusters heads.

4.2 Consommation d'énergie dans les réseaux de capteurs sans fil

L'énergie totale consommée par un noeud capteur a pour origine trois fonctions principales : la capture, le traitement des données et la communication [37] [38].

4.2.1 Energie de capture

Elle est utilisée lors des opérations suivantes : (1) échantillonnage ; (2) traitement du signal ; (3) conversion analogique/numérique ; (4) activation de la sonde du capteur. En règle général, l'énergie de capture représente un faible pourcentage de l'énergie totale [39].

4.2.2 Energie de traitement

Cette énergie se décline en deux parties : l'énergie de commutation et l'énergie de fuite [39]. L'énergie de commutation est déterminée par la tension d'alimentation et la capacité totale commutée au niveau logiciel (en exécutant un logiciel). Si l'unité de calcul n'effectue aucun traitement, l'énergie consommée correspond à ce qu'on appelle l'énergie de fuite. L'énergie nécessaire aux traitements est très faible par rapport à celle exigée pour la communication des données [40] [41].

4.2.3 Energie de Communication

L'énergie de communication représente la plus grande proportion de l'énergie totale consommée au niveau d'un noeud. Elle est déterminée par la quantité des données à communiquer et la distance de transmission, ainsi que par les propriétés physiques du module radio. L'émission d'un signal est caractérisée par sa puissance. Quand la puissance d'émission est élevée, le signal aura une grande portée et l'énergie consommée sera plus élevée [42].

4.3 Modele de consommation d'énergie

Nous utilisons le modèle énergétique décrit dans [19]. Heinzelman et al proposent un modèle énergétique. Ainsi, les énergies nécessaires pour transmettre $\mathbf{ETx}(\mathbf{s}; \mathbf{d})$ et recevoir $\mathbf{ERx}(\mathbf{s}; \mathbf{d})$ des messages de \mathbf{s} bits à partir d'un noeud situé à une distance \mathbf{d} sont données par :

Pour l'émission de données :

$$\begin{aligned} \mathbf{ETx}(\mathbf{s}, \mathbf{d}) &= \mathbf{s} * \mathbf{Eelec} + (\mathbf{Efs} * \mathbf{s} * \mathbf{d}^2) \quad \text{si } \mathbf{d} < \mathbf{d}_0 \\ \mathbf{ETx}(\mathbf{s}, \mathbf{d}) &= \mathbf{s} * \mathbf{Eelec} + (\mathbf{Eamp} * \mathbf{s} * \mathbf{d}^4) \quad \text{si } \mathbf{d} \geq \mathbf{d}_0 \end{aligned}$$

Où $d_0 = \sqrt{\frac{Efs}{E_{amp}}}$ est égale en simulation à 87 m.

Pour la réception de données :

$$E_{Rx}(s) = (s * E_{elec})$$

Où **E_{amp}** représente l'énergie requise par bit pour l'amplificateur de transmission et **E_{elec}** c'est l'énergie électronique du module radio, la même pour l'émission et la réception. La consommation d'énergie dépend de la distance entre l'émetteur et le récepteur et des caractéristiques radios **E_{elec}** et **E_{amp}**, qui sont choisies lors de phase de conception matérielle. **d** est la distance entre l'émetteur et le récepteur.

Pour l'agrégation de données :

Pour l'énergie d'agrégation consommée par chaque cluster head dans le réseau par une constante **E_{da}** qui vaut toujours 5nj (nj : nano joule).

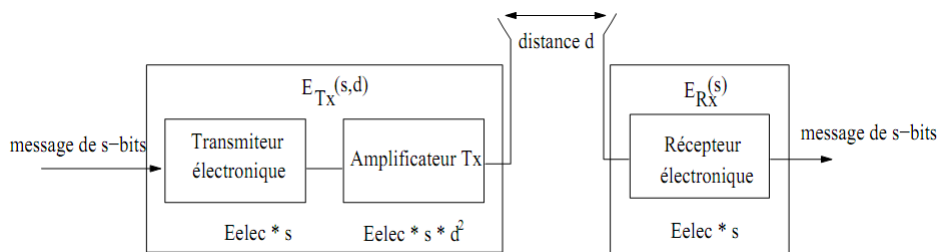


FIGURE 4.1 – Le modèle de consommation d'énergie.

4.4 Description de la proposition

Lors de l'étude du problème de routage, il devient de plus en plus complexe lorsque le nombre de nœuds augmente (Scalability).

Notre protocole DECSA-G se base sur le clustering statique. Il permet d'organiser le réseau en niveau afin d'assurer un routage multi-sauts entre les nœuds et effectue un balayage horizontal de 90 degré afin de créer des clusters de taille moyenne. Il élimine la surcharge de clustering dynamique (c'est à dire la surcharge des calculs lors de sélection des clusters head à chaque période), et utilise des clusters head prochain (c'est à dire les clusters head sélectionné pour la prochaine période) pour équilibrer la consommation de l'énergie entre les nœuds qui ont de grandes capacités d'énergie dans le but de prolonger la durée de vie des réseaux de capteurs

sans fil.

Le protocole EEPSC utilise l'élection aléatoire des clusters head c'est pour cela qu'on peut tomber sur des neuds à peu d'énergie. Pour la phase de communication nous inspirons du protocole DECSA. Ce dernier surcharge les clusters head à faire des calculs pour trouver le chemin vers la station de base, mais dans DECSA-G c'est à la station de base de prendre en charge les calculs afin d'économiser l'énergie des neuds de capteurs.

4.4.1 Hypothèses

Avant de passer à la description de la solution proposée, Nous souhaitons tout d'abord étayer certaines hypothèses. Nous avons considéré un ensemble de capteurs déployés dans une région carrée. Nous supposons que :

1. Les noeuds de capteurs sont homogènes et fixes.
2. Les noeuds sont distribués aléatoirement sur la zone de capture.
3. Tous les capteurs collectent les données et servent une unique station de base.
4. La station de base est vue comme ressource non limitée ni épuisable, elle se trouve à l'une des extrémités de la surface de déploiement.
5. La défaillance d'un capteur n'est causée que par l'épuisement de son énergie.
6. Chaque noeud peut atteindre la station de base (c'est à dire que chaque capteur peut envoyer des données à la station de base).
7. Le nombre de niveaux est fixé par la station de base.

4.4.2 Le fonctionnement du protocole DECSA-G

Notre protocole DECSA-G se déroule en trois phases , chaque phase est décrite comme suit :

- **Phase d'organisation de groupes :**

Cette première phase est exécutée une seule fois durant tout le processus de routage car DECSA-G est basé sur le clustring statique.

La phase d'organisation de groupes se réalise en deux étapes, la première consiste à former les niveaux et la deuxième permet la formation des clusters.

1. **Formation des Niveaux :**

Dans la première étape, la station de base diffuse $k-1$ messages différents avec différentes puissances de signal d'émission avec $k = \text{"numéro de niveau"}$ afin de former k niveaux. Par la diffusion du message $k=1$, les noeuds qui reçoivent ce mes-

sage seront dans le même cluster, ils mettent le groupe à **ID-K**. De même pour la diffusion du message **k=k-1**.

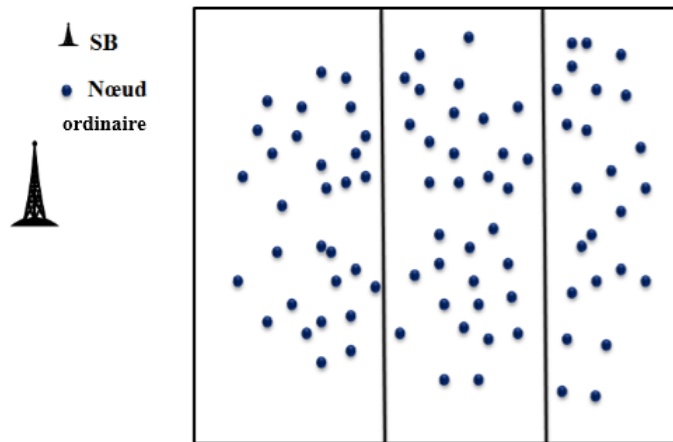


FIGURE 4.2 – Protocole DECSA-G : Formation des niveaux.

2. *Formation des Clusters :*

Dans cette étape, la station de base effectue un balayage avec un angle **90 degré** afin de former **2*k** clusters.

Après la formation des niveaux et des clusters, tous les noeuds informent la station de base qu'ils sont les membres de niveau **ID-niveau** et de cluster **ID-cluster** et envoient leur énergie initiale et l'énergie résiduelle. Chaque noeud i envoie le message "Join-REQ" contenant (**ID-niveau**, **ID-cluster**, $E_r(i)$) en utilisant la méthode d'accès CSMA (Carrier Sense Multiple Access) pour éviter les collisions. Par la suite, les noeuds qui n'ont pas rejoint aucun clusters informent la station de base qu'ils sont les membres du dernier niveau (ils appartiennent à l'un des deux derniers clusters) et envoient aussi le message "Join-REQ".

Le principe de cette phase est illustré sur la figure suivante :

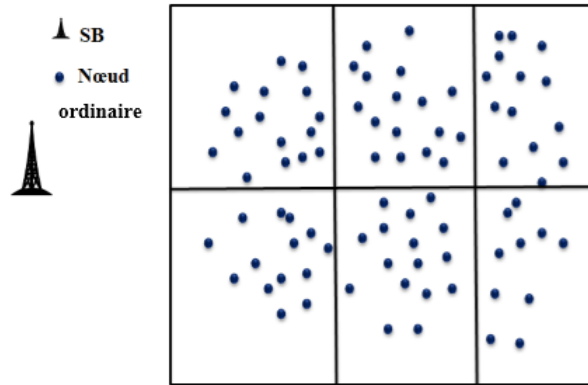


FIGURE 4.3 – Protocole DECSA-G : Formation des clusters.

À la fin de cette phase chaque nœud de capteur connaît son niveau et son cluster.

L'organigramme suivant présente le principe de cette phase :

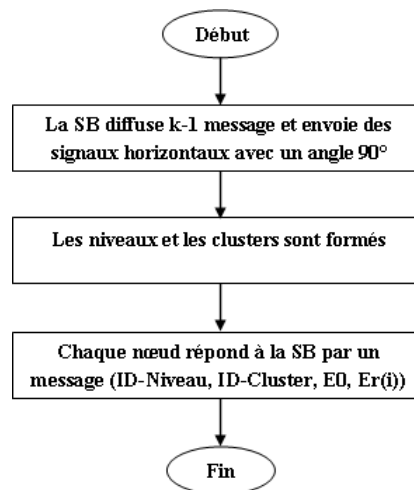


FIGURE 4.4 – L'organigramme de phase d'organisation de groupes.

l'algorithme suivant présente le principe de cette phase :

```

Fonction Phase_org_group()
Variables
    Capteur c1,c2;
    Entier id_i, id_cluster, id_niv, nb_noeud; //avec id_i est identité du noeud, id_cluster est l'identité du
                                                //cluster, id_niv est le n° identité du niveau ,nb_noeud : nombre de noeuds.
    Station_de_base SB;
    Message k=; //k= numéro de niveau
Debut
    diffuser (k-1,SB) //la fonction diffuser k-1 message par la station de base .
    diffuser2(sign, 90°) // la fonction diffuser2 c'ad que la station de base fait un balayage de 90° dans le réseau
    pour (i=0 à nb_noeuds ) faire
        Si (c1.k=c2.k) alors //si les noeuds ont reçus le même message
            appartenir (c1, nb_cluster, nb_niv); //les noeuds appartiennent au même cluster et même niveau
            finSi ;
        Envoyer (c1.id_i, id-cluster, id_niv); //envoyer un message d'annonce à la station de base.
    FinPour ;

Fin.
    
```

FIGURE 4.5 – L'algorithme de phase d'organisation de groupes.

Avantage :

Dans cette phase, le clustering statique a l'avantage d'éliminer la surcharge des calculs dans la première phase puisque elle s'exécute une seule fois durant tout le processus de routage , et le balayage effectué sert à avoir un nombre de clusters de taille moyenne.

• **Phase d'élection de clusters head (CHs) :**

Lorsque la station de base récupère les messages envoyés par les noeuds, elle calcule la distance qui la sépare avec chaque noeud dans le cluster en utilisant la puissance du signal.

- La station de base calcule la valeur $k(i)$ du capteur "i" pour la première période, la formule de calcul est :

$$k(i) = \frac{Er(i)}{d(i)}$$

Où :

$K(i)$: est le seuil de l'élection de CH.

$Er(i)$: est l'énergie résiduelle du capteur i.

$d(i)$: est la distance qui sépare le capteur "i" et la station de base.

- Ensuite, la station de base sélectionne un cluster head pour chaque cluster le noeud qui possède le maximum $k(i)$, c'est à dire le capteur qui possède une grande énergie résiduelle et plus proche de la station de base, puis elle diffuse un message pour chaque cluster contenant l'identité du cluster head (**ID-CH**) sélectionné.
- Après l'élection du cluster head, chaque noeud du cluster envoie son énergie résiduelle au cluster head, ce dernier à son tour sélectionne le noeud qui possède l'énergie maximale comme cluster head pour la prochaine période nous l'appellons ici cluster Head prochain (CH-p).
- Par la suite, afin de structurer le chemin de communication entre les clusters heads vers la station de base, cette dernière calcule le TBCH(i) (Threshold Base station Cluster Head) pour cluster head "i" comme suit :

$$TBCH(i) = \frac{Er(i)}{E0} + \frac{Er(i)}{d(i)}$$

Où :

TBCH(i) : est le seuil de l'élection du TBCH.

Er(i) : est l'énergie résiduelle du CH(i).

E0 : est l'énergie initial du CH.

d(i) : est la distance qui sépare chaque capteur de la station de base du même cluster.

- Après le calcul du TBCH(i), la station de base envoie des messages pour chaque cluster head, chaque message contient (ID-niveau, ID-cluster, ID-CH, Tab-TBCH) pour choisir le cluster head et informer les membres du cluster de leur cluster head.

Tab-TBCH c'est un tableau qui contient les valeurs de TBCH(i) de chaque cluster Head.

Ici le cluster qui a la valeur de TBCH(i) maximale est celui qui communique directement (un saut) avec la station de base.

La figure suivante illustre le principe de cette phase :

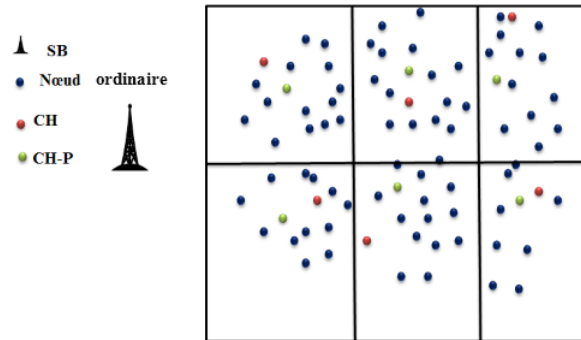


FIGURE 4.6 – Election des clusters Heads.

Voici l'organigramme de la deuxième phase :

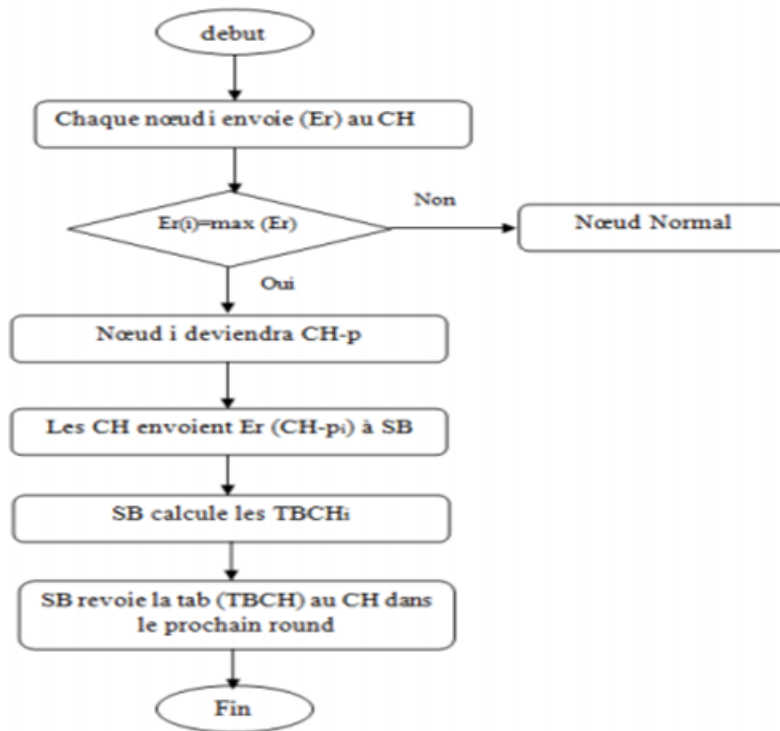


FIGURE 4.7 – Protocole DECSA-G : Election des clusters Heads.

l'algorithme suivant présente le principe d'élection de CH et CH-p :

```

Fonction election_CH_CH-P()
Debut
Ki=c1.Er/di // Er est l'énergie résiduel du capteur, di est la distance entre le nœud et la station de
//base. Le Ki est calculé par la station de base
Si (max(ki))
    c1.ch=vrai ;
finSi ;
tantque (c1.<> ch) faire
    envoyer(c1.Er, c1.ch) // envoyer l'énergie résiduelle à son cluster head
fintantque ;
Si (max(c1.Er))
    c1.ch_p=vrai ;
Finsi ;
Fin ;
    
```

FIGURE 4.8 – L'algorithme de phase d'élection de CH et CH-p .

Avantages :

Dans cette phase, nous avons éliminé la sélection aléatoire des clusters head, c'est à dire que nous avons choisit des clusters head en terme d'énergie et de la distance, et contrairement à DECSA nous avons surcharger la station de base par les calculs des TBCH au lieu de les faire calculer par les clusters heads.

• **Phase de communication :**

Cette phase assure la collecte et l'acheminement des données vers la sation de base. Dans notre protocole DECSA-G nous utilisons une communication multi-saut pour minimiser l'énergie consommée par les noeuds de capteurs et augmenter la durrée de vie du réseau. Afin que les noeuds transmettent leur données sans risque de collisions, nous utilisons l'ordonnancement TDMA (Time Division Multiple Access), qui alloue pour chaque noeud une tranche de temps pour émettre ses données.

La figure suivante montre le fonctionnement de TDMA :



FIGURE 4.9 – Le fonctionnement de TDMA.

À chaque période la phase de communication dans DECSA-G se divise en deux sous phases : la première concerne la communication interne, et la deuxième la communication externe.

1. *Communication interne (intra-cluster)* :

Dans cette sous phase, à la détection d'un évènement par un noeud de capteur dans chaque cluster. Ce noeud envoie ses données à son cluster head.

La figure suivante montre le principe de cette sous phase :

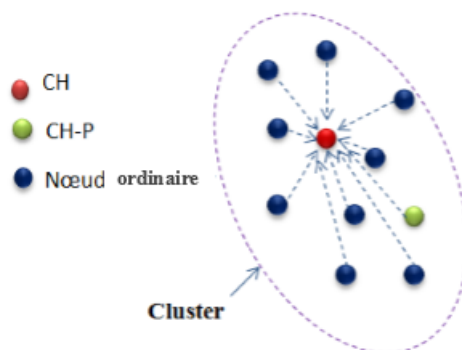


FIGURE 4.10 – Protocole DECSA-G : Communication interne.

2. *Communication externe (inter-cluster)* :

Le cluster head collecte et agrège les données reçues. Afin d'assurer l'acheminement des informations vers la station de base, les clusters head ont connu déjà leur clusters head voisins d'après la table de TBCH envoyé par la station de base, c'est à dire que chaque cluster head choisit son cluster head voisin celui qui a la valeur de TBCH juste supérieur à lui dans le niveau inférieur, ainsi de suite jusqu'à atteindre la station de base .

La figure suivante montre le principe de cette sous phase :

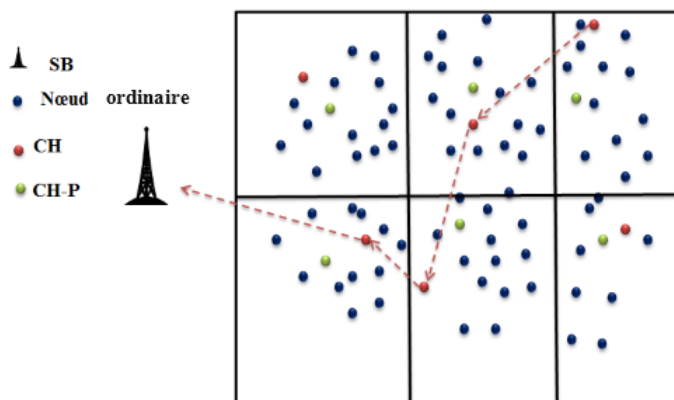


FIGURE 4.11 – Protocole DECSA-G : Communication externe.

Vers la fin de cette phase, chaque CH-p envoie son énergie résiduelle au cluster head actuel, ce dernier à son tour utilise la communication multi-saut pour le transmettre à la SB.

La station de base calcule les TBCH(i) des CH-P pour la prochaine période, et au début de chaque période suivante, la station de base envoie l'identité de CH pour informer les capteurs de chaque cluster du CH actuel, et elle envoie la table de TBCH pour les clusters head afin d'assurer la communication multi-saut.

Avantage :

Dans cette phase, nous avons éliminé la formation des chaînes dans les clusters pour ne pas tomber sur les mêmes chemins lorsque le noeud de capteur choisit son voisin le plus proche en terme de distance, ce qui mène à l'épuisement rapide des noeuds qui forment ces chemins, et nous avons utilisé la communication multi-saut pour éviter d'envoyer les données directement vers la station de base.

4.5 Simulation et analyse des performances du protocole DECSA-G

4.5.1 Choix de l'environnement de développement

Pour simuler les protocoles (EEPSC, DECSA et DECSA-G) nous tentons d'utiliser l'un des langages de programmation les plus populaires et répondu. Nous choisissons Java.

4.5.2 Java

Apparu fin 1995 début 1996 et développé par Sun Microsystems Java s'est très rapidement taillé une place importante en particulier dans le domaine de l'internet et des applications client-serveur. Destiné au départ à être un langage de programmation pouvant être intégré dans les appareils électroménagers, afin de pouvoir les contrôler de les rendre interactif et surtout de permettre une communication entre les appareils. La société Sun a eu l'idée de le recentrer sur les applications de l'internet et des réseaux. C'est un langage en évolution permanente Java 2 est la version stabilisée de java fondée sur la version initiale 1.1.2 du JDK (Java Développement Kit de Sun). Les objectifs de java sont d'être multi-plateformes et d'assurer la sécurité aussi bien pendant le développement que pendant l'utilisation d'un programme java. Il est en passe de détrôner le langage C++ dont il hérite partiellement la syntaxe mais non ses défauts. Comme C++ et Delphi, java est algorithmique et orienté objet. A ce titre il peut effectuer comme ses

compagnons, toutes les tâches d'un tel langage (bureautique, graphique, multimédias, bases de données, environnement de développement, etc). Son point fort qui le démarque des autres est sa portabilité due (en théorie) à ses bibliothèques de classes indépendantes de la plateforme, ce qui est le point essentiel de la programmation sur internet où plusieurs machines dissemblables sont interconnectées. La réalisation multi-plateformes dépend en fait du système d'exploitation et de sa capacité à posséder des outils de compilation et d'interprétation de la machine virtuelle Java. Actuellement ceci est totalement réalisé d'une manière correcte sur les plates-formes Windows et Solaris.

4.5.3 L'environnement de simulation :

Notre modèle de simulation est établi sur 100 à 500 noeuds , dispersés aléatoirement sur une surface de 100*100 m²; nous assumons que tous les noeuds ont une position fixe durant toute la période de simulation. Les paramètres de simulation sont résumés dans le Tableau 1.

Parametre	Valeur
Localisation de la station de base	(-50m,50m)
Nombre de noeuds	100 à 500
Nombre de clusters	8
rayon de communication	25m
Energie initiale	1J
Taille de paquet de données	4000bit
Taille de paquet de controle	32 bit
Durrée de simulation	1470 période
Durrées d'une période	5ms

TABLE 4.1 – Les paramètres de simulation.

4.5.4 Métriques considérées

1. La consommation d'énergie :

Le protocole DECSA et EEPSC visent à optimiser la consommation d'énergie afin de prolonger la durée de vie du réseau. De ce fait, la consommation d'énergie est un paramètre primordial lors de la phase de tests. Pour la comparaison des trois protocoles, nous nous intéressons à l'énergie résiduelle moyenne restante au niveau des noeuds.

2. La durée de vie du réseau : La durée de vie d'un RCSF est définie en utilisant trois métriques FND (First Node Dies), HNA (Half of the Nodes Alive), et LND (Last Node Dies) [35]. Dans notre travail, nous avons évalué la durée de vie d'un réseau de capteurs

dans un des deux contextes. Ces contextes suivent la métrique FND i.e. le temps écoulé jusqu'à ce que le premier capteur cesse de fonctionner, et la métrique LND i.e. le temps écoulé jusqu'à ce que le dernier capteur cesse de fonctionner.

4.5.5 Caractéristique du simulateur

Notre simulateur est caractérisé par une interface déviser en une zone de dessin à gauche, et un onglet de manipulation à droite, ce dernier comporte des boutons concernant la génération du nombre de noeuds (Generer) , un combobox pour le partitionnement du réseau (Nbr niveaux), un JTextField pour afficher le nombre de périodes courantes (Round), des boutons pour lancer et arrêter la simulation (Lancer et Arreter), et enfin les résultats à afficher durant le déroulement des trois protocoles (Resultats). la figure ci dessous montre les caractéristique global du simulateur :

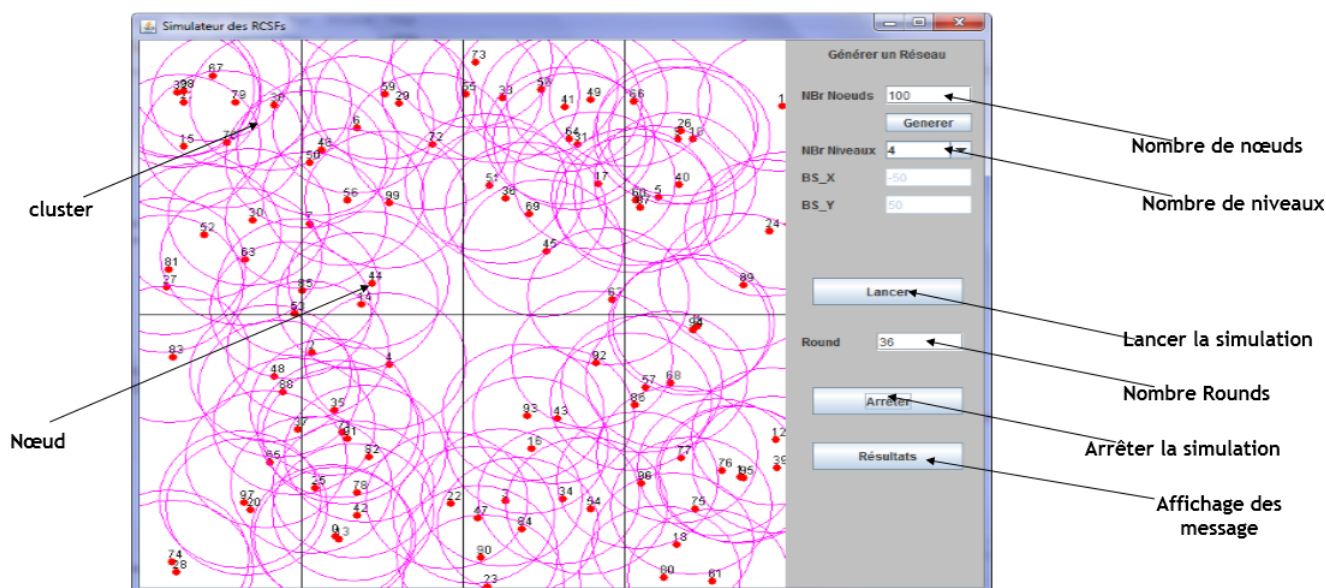


FIGURE 4.12 – Interface du simulateur.

4.5.6 Evaluation de performances

Dans ce qui suit nous présentons et analyser les résultats de simulations obtenus suivants les metriques discutées précédemment. Pour chaque tests, 5 exécutions independante sont réalisées : les résultat présentés sont la moyenne de toutes les exécutions.

Pour les simulations réalisées, le paramètre variable utilisé est le nombre de noeuds déployés sur la zone.

1. Moyenne d'énergie résiduelle

Dans la figure ci dessous avec le déploiement de 100 capteurs ,nous remarquons que le protocole DECSA-G surpasse le protocole DECSA avec un pourcentage moyen de 12% et de 16% par rapport à EEPSC.

En remarquant que le protocole EEPSC épuise toute son énergie après environs 400 périodes, et pour DECSA l'énergie du réseau est épuisée vers la période 660, par contre notre protocole se termine en période 1060. Ce gain en énergie est dû aux différentes techniques pour la formation des clusters statiques ainsi qu'aux celles utilisées pour la sélection des CH et CH-p. Ces techniques prennent en compte l'énergie résiduelle et la distance par rapport à la station de base.

De plus, les techniques de transmission utilisées entre les clusters head pour router les données à la station de base dans DECSA-G qui contrôle efficacement l'énergie de transmission en adoptant un routage multi-sauts entre les clusters head vers la station de base. Or le routage de données dans le protocole EEPSC se fait via un seul saut, et celui de DECSA malgré qu'il utilise le multi-saut, mais il ya un risque de revenir au derniers noeuds qui sont loin de la station de base et de faire l'envoi direct.

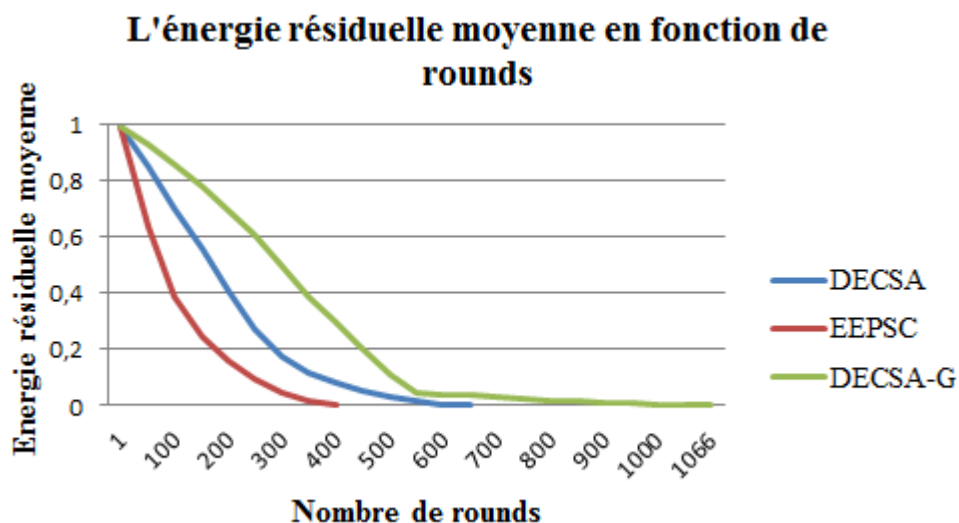


FIGURE 4.13 – Moyenne d'énergie résiduelle.

2. Durée de vie du réseau

Pour pouvoir étudier la durée de vie du réseau en fonction du nombre de noeuds déployés, nous avons suivi l'évolution de 100 à 500 noeuds dans le temps.

Il est clair, que la durée de vie offerte par le protocole DECSA-G surpasse celle de EEPSC et DECSA.

- **Le premier capteur mort en fonction de nombre de noeuds** : La figure ci-dessous, représente la période où le premier noeud meurt avec le déploiement de 100 à 500 noeuds de capteurs. En remarquant qu'à l'augmentation de nombre de noeuds déployés, la durée de vie du réseau est prolongée, ceci est dû aux techniques de sélection de clusters head, à l'augmentation du nombre de noeuds dans les protocoles DECSA et EEPSC surchargent les noeuds avec des calculs.

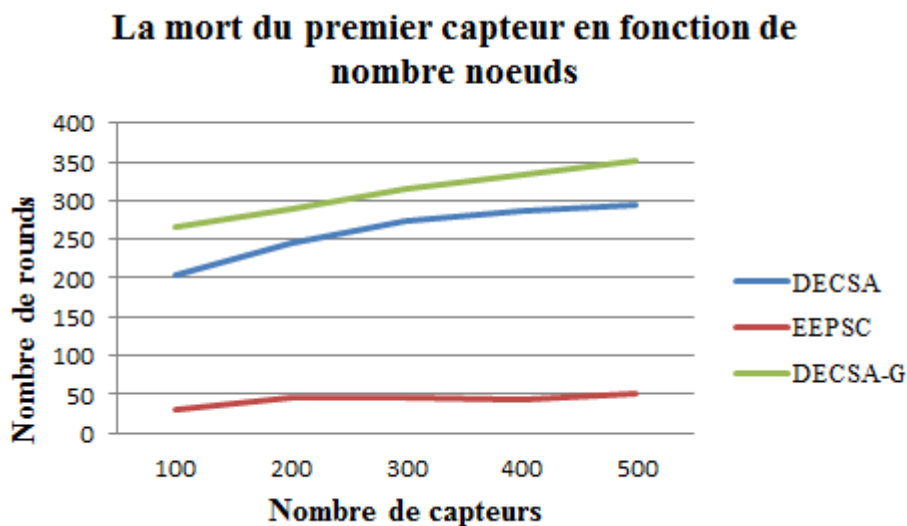


FIGURE 4.14 – Le premier capteur défaillant en fonction de nombre de noeuds.

- **La mort du 25% et 50% de noeuds de capteur** : Ces deux graphes représentent la défaillance du 25% et 50% des capteurs en fonction de nombre de noeuds qui varie entre 100 à 500. A l'augmentation de nombre de noeuds nous remarquons que notre protocole est plus performant par rapport à EEPSC et DECSA, et ceci dû aux mêmes avantages de notre protocole cités précédemment.

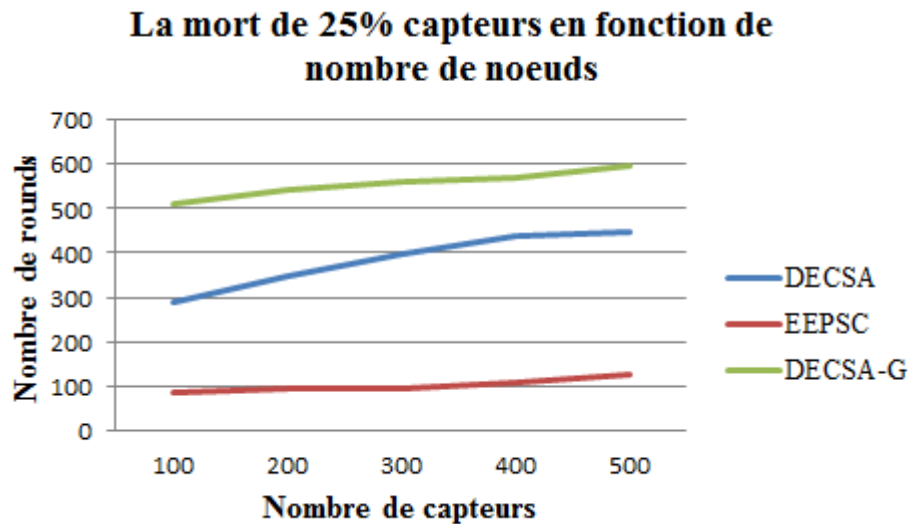


FIGURE 4.15 – La défaillance du 25% de noeuds de capteur en fonction de nombre de noeuds.

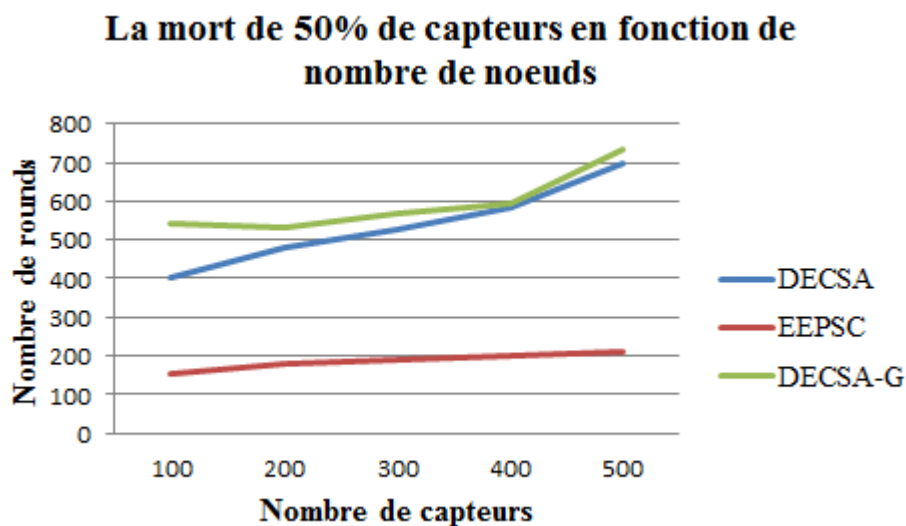


FIGURE 4.16 – La défaillance du 50% de noeuds de capteur en fonction de nombre de noeuds.

- Le dernier capteur mort en fonction de nombre de noeuds :** Le graphe ci-dessous, représente la mort du dernier capteur en fonction de nombre de noeuds qui varie entre 100 à 500 , où on peut remarquer que la durée de vie de notre protocole est prolongée jusqu'au 1850 périodes pour 500 noeuds, donc notre protocole DECSA-G est plus scalable que EEPSC et DECSA à cause de minimisation de nombre de clusters et l'utilisation de la technique multi-sauts ainsi que nous avons surchargé la station de base des calculs.

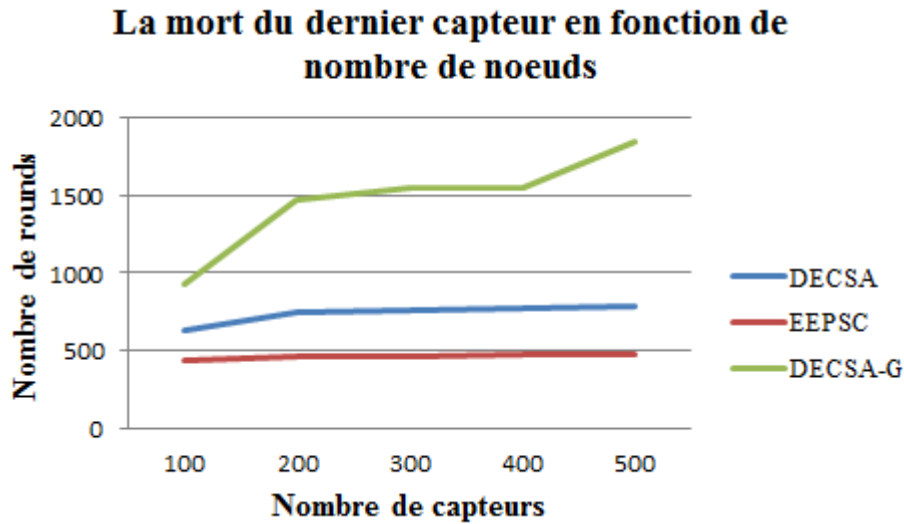


FIGURE 4.17 – La défaillance du dernier capteur en fonction de nombre de noeuds.

4.6 Conclusion

Dans ce chapitre, nous avons présenté un nouveau protocole de routage hiérarchique selon l'approche de clustering statique pour les réseaux de capteurs sans fil. Le protocole proposé adopte des mécanismes appropriés pour le routage à basse consommation d'énergie à travers l'introduction d'une structure en niveaux pour la topologie du réseau, ce qui offre une souplesse dans la communication multi sauts des données à la station de base. Par ailleurs, notre protocole élimine la sélection aléatoire de CHs et utilise une méthode exacte. Pour la phase de communication nous avons utilisé le tableau des valeurs de TBCH afin d'économiser l'énergie des neuds de capteurs et de prolonger la durée de vie du réseau.

CONCLUSION GÉNÉRALE

Les réseaux de capteurs sans fil constituent un axe très de recherche très fertile et peuvent être appliqués dans plusieurs domaines d'applications différentes. Ces dernières feront de cette technologie émergente une partie intégrale de nos vies actuelles et futures. L'énergie est la ressource la plus précieuse dans un réseau de capteurs, parce qu'elle influe directement sur la durée de vie des capteurs et du réseau en entier par conséquent.

Dans ce mémoire, nous nous sommes intéressées au problème du routage dans les réseaux de capteurs sans fil. Avant de proposer notre solution à ce problème, nous avons étudié plusieurs protocoles de routage appartenant à différentes catégories afin de s'inspirer de leurs techniques.

Nous avons présenté un nouveau protocole de routage hiérarchique selon l'approche de clustering statique, nommé DECSA-G. Ce protocole adopte des mécanismes appropriés pour le routage à basse consommation d'énergie à travers l'introduction d'une structure statique en niveaux avec un partitionnement uniforme en nombre de capteurs par cluster pour la topologie du réseau et cela pour éliminer la surcharge de clustering dynamique.

Par ailleurs, notre protocole utilise une méthode exacte pour l'élection des clusters head et des clusters head prochains pour équilibrer la charge entre les nIJudS qui ont beaucoup d'énergie. Pour la phase de communication nous avons utilisé un tableau qui contient des valeurs de TBCH afin d'assurer une communication multi-saut des données vers la station de base en passant par les CHs voisins dont le but est d'économiser l'énergie des nIJudS de capteurs et de prolonger la durée de vie du réseau.

Après la description de notre approche proposée, l'environnement de développement JAVA nous a permis de concevoir et de réaliser un simulateur pour les réseaux de capteurs sans fil avec des interfaces qui facilitent la tâche aux utilisateurs de ce type de réseau. La création du

simulateur nous a permis d'implémenter notre protocole de routage hiérarchique et les deux autres EEPSC et DECSA.

Nous avons démontré via les résultats de simulations que notre protocole présent de meilleures performances en termes de l'énergie moyenne restante et de la durée de vie par rapport aux protocoles DECSA et EEPSC.

Perspectives

Les réseaux de capteurs constituent un domaine de recherche très vaste. Ils ont de nombreuses perspectives d'application dans des domaines très variés. Il reste encore de nombreux problèmes à résoudre dans ce domaine afin de pouvoir les utiliser dans des conditions réelles.

Nous envisageons l'étude de notre protocole, dans le cas où la surface de deployment du RCSF est très grande, il y'aurait des clusters de grande taille, donc les noeuds qui sont loin de CH épuise rapidement leurs énergie. Pour cela, nous proposons de diviser la surface afin d'avoir des clusters moyen en taille. Et utiliser les noeuds capteurs hétérogènes au lieu des noeuds homogènes, traiter le problème de la mobilité soit de la station de base ou les noeuds de capteurs.

Aussi, développer notre simulateur conçu afin d'avoir un simulateur très puissant en ajoutant du parallélisme pour les exécutions des protocoles implémentés dans le but d'accélérer la simulation, ajouter d'autres types de radio, changer la position de la station de base pour trouver la bonne position qui donne de meilleures performances.

Par la suite, utiliser un SGBD(System de Gestion de Bases de Données) pour récupérer les données concernant les résultats de simulation afin de les interpréter par le service reporting.Enfin, application de notre approches dans différentes domaines.

BIBLIOGRAPHIE

- [1] DUNOD, "tous sur les réseaux sans fil", EYROLLES.
- [2] Guy Pujolle, "Les réseaux", édition 2008.
- [3] B.Kamal, "conception d'un protocole de routage hiérarchique pour les réseaux de capteurs", l'u.f.r des sciences et techniques de l'université de Franche-Comté, 2009.
- [4] Z.Rafik, "routage à basse consommation d'énergie dans les réseaux de capteurs sans fil", mémoire de magistère En Informatique Option Réseaux et Systèmes Distribués, 2006.
- [5] C.Yacine, "Réseaux de Capteurs sans fil", Version 1 SIT60, 2008.
- [6] <http://www.libelium.com/waspmote>
- [7] W.Quinghua, I.Balasingham, "Wireless Sensor Networks - An Introduction", wireless sensor networks. Application Centric Design, Geof V Merrett and Yen Kheng Tan (Ed), ISBN : 978-953-307-321-7, InTech, 2010.
- [8] L.Mohamed, "Diffusion et couverture basées sur le clustering dans les réseaux de capteurs : application à la domotique", Thèse de doctorat, Université A.B Tlemcen et Université de Franche-Comté U.F.R Sciences et Techniques École Doctorale SPIM, 2009.
- [9] k. Yang, "A study on power-friendly routing protocols for sensor networks", Technical Report. University of Essex Department of Electronic Systems Engineering.
- [10] A.kamel, M.Younis, "A routing protocol for wireless ad-hoc sensor networks : Multi-Path Source Routing Protocol (MPSR), ICN", 05 : 4th International Conference on networking (IEEE).
- [11] M.Sofiane, "Optimisation de la consommation d'énergie dans les réseaux de capteurs sans fil", Master : Recherche 2 en Informatique Université : IFSIC-Rennes 1 Laboratoire de recherche : DYONISOS-IRISA, 2008.
- [12] Kh.Lyes, B.Nadjib, "Les réseaux de capteurs : état de l'art", Université de Sciences Technologies Houari BOUMADIANE, Laboratoire de systèmes informatiques, Alger, 2004.

-
- [13] S.Ziane, A.Mellouk, "A swarm intelligent scheme for routing in mobile ad networks". Systems Communications, IEEE, Aug 2005.
- [14] Paolo Santi, "Topology Control in Wireless Ad Hoc and Sensor Networks". Hardcover, july 2005.
- [15] S.Kumar, D.Shepherd, F.Zhao, "Collaborative signal and information processing in micro-sensor networks". IEEE Signal Processing Magazine, March 2002.
- [16] A.Kemal, M.Younis, "A Survey on Routing Protocols for Wireless Sensor Networks ", Elsevier Ad Hoc Network Journal, Vol 3/3 pp. 325-349, 2005.
- [17] M Z.Adamu, L.Ang, K.Seng, "Classical and swarm intelligence based routing protocols for wireless sensor networks : A survey and comparison", Journal of Network and Computer Applications, 2012.
- [18] W.Yen, Ch.Chen, Ch.Yang, "Single Gossiping with Directinal Flooding routing protocole in wireless sensor networks". Département des techniques et sciences informatiques, Université de Tatung, 2008.
- [19] W.Heinzelman, A.Chandrakasan, H. Balakrishnan, "Energy-efficient communication protocol for wireless sensor networks", in :Proceeding of the Hawaii International Conference System Sciences, 2000.
- [20] W. Heinzelman, A. Chandrakasan, H. Balakrishnan, "Energy-efficient communication protocol for wireless sensor networks", An Application-Specific protocol achitecture for wireless microsensor networks, IEEE Transaction Wireless Communication, 2002.
- [21] W. Heinzelman et all, "Wireless Sensor Network Protocols", CRC Hall, 2005.
- [22] O.YOUNIS and S.FAHMY, "Heed : A Hybrid Energy Efficient DistributedClustering Approach for Ad-hoc Sensor Networks", IEEE Transactions on Mobile Computing, Mrch, 2004.
- [23] N.MEJRI, F.KAMOUN, "Algorithme de Routage Hiéa rchique MHEED à plusieurs sauts pour Les Grands Réseaux de Capteurs", 4th International Conference : Sciences of Electronic, Technologies of Information and Telecommunications, Tunisia, 2007.
- [24] H.Sofiane, "Amélioration de la Performance des Protocoles Routage et MAC pour une Meilleure QoS dans un MANET", Département d'Informatique.
- [25] S.Adel, El ashheb1, "Performance Evaluation of AODV and DSDV Routing Protocol in wireless sensor network Environment", College of civil aviation and meteorology, Libya.
- [26] J.P.R Draves, B.Zill , " Comparison of routing metrics for static multi-hop Wireless networks, Proceeding of ACM SIGCOM, 2004.
- [27] I. Stojmenovic, X. Lin, " power aware localized routing in wireless networks ", Power, Vol.12, No. 11, pp. 1122-1133,2004.

-
- [28] K. yang, "a study on power friendly routing protocols for sensor networks", technical report, university of essex departement of electronic systems engineering.
- [29] A. Abassi , M. Younis, " A survey on clustering algorithme for wireless sensor networks ", 2007.
- [30] X.Liu, "A Survey on Clustering Routing Protocols in Wireless Sensor Networks" ,School of Electronic and Information Engineering, South China University of Technology, 2012.
- [31] S.Lindsey, C.S.Raghavendra, "PEGASIS : Power-Efficient Gathering in Sensor Information Systems", IEEE Aerospace Conference Proceeding, 2002.
- [32] S Z.Amir, B.Abolhassani, A.Ali, B.Shirazi, and S B.Ali, "An Energy-Efficient Protocol with Static Clustering for Wireless Sensor Networks", Proceedings of World Academy of Science Engineering and Technology(PWASET), Iran Telecommunication Research Center, Iran, 2007.
- [33] A.Manjeshwar, D.P.Agrawal, "TEEN : a protocol for enhanced efficiency in wireless sensor networks", in : Proceedings of the 1st International Workshop on Parallel and Distributed Computing Issues in Wireless Networks and Mobile Computing, San Francisco, CA, April 2001.
- [34] A.Manjeshwar, D.P.Agrawal, "APTEEN : an hybrid protocol for efficient routing and comprehensive information retrieval in wireless sensor networks", in : Proceedings of the 2nd International Workshop on Parallel and Distributed Computing Issues in Wireless Networks and Mobile computing, Ft. Lauderdale, FL, April 2002.
- [35] A.Zibouda, A.Makhlouf, "Gestion Efficace du Budget Energétique dans le Protocole PEGASIS", Sciences of Electronics Technologies of Information and Telecommunications(SE-TIT), Tunisia, 2012.
- [36] Z.Yong, Q.Pei, "A Energy-Efficient Clustering Routing Algorithm Based on Distance and Residual Energy for Wireless Sensor Networks", Internatinal Workspace on Information and Electronics Engineering(IWIEE), 2012.
- [37] V.Raghunathan, C.Schurgers, S.Park, and M.B.Srivastava, "Energy-aware wireless micro-sensor". IEEE Signal Process. Mag, 19(2) :40-50, May 2002.
- [38] E.Shih and al, "Physical layer driven protocol and algorithm design for energy-efficient wireless". ACM/IEEE MOBICOM, pages 272-287, July 2001.
- [39] I.Mohammad, M.Imad, "Handbook of sensor Networks Compact Wireless and Wired Sesign Systems". Number 0-8493-1968-4. CRC PRESS LLS, USA, 2005.
- [40] F.Zhao and al. "Collaborative signal and information processing : an information directed approach". IEEE, 8 :1199-1209, 2003.
- [41] G.J.Pottie and W.J.Kaiser, "Wireless integrated network sensors". ACM, 43(5) :51-58, May 2000.

-
- [42] S.Paolo, "Topology Control in Wireless Ad Hoc and Sensor Networks". Hard-cover, july 2005.
- [43] H.Zhang, H.Shen, "Balancing energy consumption to maximize network in data-gathering sensor networks", IEEE Trans. Parallel Distributed Syst. 20 (10) (2009) 1526-1539.
- bibliography

Résumé

Les noeuds capteurs d'un Réseaux de Capteurs Sans Fil (RCSF) sont alimentés par des batteries épuisables à durée de vie limitée et non rechargeable. Comme le déploiement de ces RCSFs s'opère dans des zones généralement inaccessibles, parfois hostiles, et le remplacement de batteries épuisées n'est pas envisageable, les nœuds doivent donc assurer leur mission avec leur unique budget énergétique initial strict.

Cette contrainte fait de l'énergie la ressource la plus décisive et d'une importance capitale dans les RCSFs. Comme le clustering dans les réseaux de capteurs sans fil est un moyen efficace de structurer le réseau. Son but est d'identifier un sous-ensemble de noeuds dans le réseau et de lui attribuer un chef (un cluster-head). Celui-ci sera en charge de tâches spécifiques comme l'agrégation de données. L'exécution de ces tâches additionnelles entraîne une augmentation de la consommation énergétique et une diminution de la durée de vie du noeud. Dans ce mémoire, nous introduisons un nouveau protocole de routage hiérarchique à basse consommation énergétique, basé sur le clustering statique et par l'envoi multi-saut des données vers la station de base, les résultats obtenus au travers des simulations réalisées par un simulateur programmé en java ont montré que les performances réalisées par notre proposition sont louables et bien convaincantes.

Mot clés : Réseaux de capteurs, Protocole de routage hiérarchique, clustering statique, Énergie, multi-saut, DECSA-G.

Abstract

The sensor nodes of Wireless Sensor Networks (WSN) are powered by exhaustible finite-life and non-rechargeable batteries. As the deployment of these WSNs usually occurs in inaccessible areas, sometimes hostile, and replacement depleted batteries is not feasible, the nodes must carry out their duties with their unique initial strict energy budget.

This constraint makes the energy resource more decisive and crucial in WSNs. As clustering in networks wireless sensor is an effective way to structure the network. Its purpose is to identify a subset of nodes in the network and assign a leader (cluster-head). This will be in charge of specific data aggregation tasks. The performance of these tasks additional increases in energy consumption and a decrease in the life of the node. In this memory, we introduce a new routing protocol hierarchical in low power consumption, based on the static clustering and sending multi-hop data to the base station, the results obtained through simulations performed by a simulator programmed in Java showed that the performance achieved by our proposal is commendable and much convincing.

keyword :Energy, hierarchical routing protocol, sensor networks, static clustering, multi-hop.

