

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieure et de la Recherche Scientifique
Université Ahmed Draia - Adrar
Faculté des Sciences et de la Technologie
Département des Mathématiques et Informatique



Mémoire de fin d'étude, en vue de l'obtention du diplôme de Master en informatique
Option : Réseaux et systèmes intelligent (RSI).

Thème

Une approche bio-inspiré basé pour la
conception d'un protocole de routage à base de
clustering dans les RCFs

Préparés par
DLIM / Rekia et TOUATI / Meryem

Encadré par
Mr. Demri Mohamed

Année Universitaire 2016/2017

Résumé

Concevoir et développer des protocoles de routage dans le cadre des applications des Réseaux de Capteurs Sans Fil (RCSF) présente un apport considérable pour améliorer les performances des réseaux et prolonger leur durée de vie. L'objectif de ce travail de recherche est de proposer et implémenter une méthodologie de routage hiérarchique, basée sur un mécanisme de classification du réseau en un ensemble de clusters constitués de nœuds capteurs qui ont un nœud chef (Clusters-Heads). Ces derniers sont sélectionnés selon un processus sélectif basé sur une méthode d'optimisation bio-inspiré appelé **PSO** (Particle Swarm Optimization). Pour tester l'efficacité et montrer les performances et les améliorations apportées par notre protocole proposé (**APSOC**), nous avons réalisé une étude comparative avec le célèbre protocole de routage **LEACH**. En termes de consommation énergétique, **APSOC** apporte des améliorations considérables.

Mots-clés: APSOC, LEACH, Protocole, Routage, Clustering, RCSF, PSO, Energie.

Abstract

Design and develop of routing protocols in the context of applications to the Wireless networks Sensors Networks (WSN) is a significant contribution to improve the performance of networks and extend their lifetime. The objective of this research work is to propose and implement a hierarchical routing methodology, based on a mechanism of classification of the network into a set of clusters made up of sensors that have a node cluster-Heads. These clusters are selected according to a selective process based on a bio-inspired optimization method called PSO (Particle Swarm Optimization). To test the effectiveness and show the performance and the improvements made by our proposed Protocol (APSOC), we have made a comparative study with the famous LEACH routing protocol. In terms of energy consumption, APSOC brings considerable improvements.

Keywords: APSOC, LEACH, Protocol, Routing, Clustering, WSN, PSO, Energy.

Dédicaces

*A mes parents ;
A mes frères et mes sœurs ;
A toute ma famille DLIM ;
A mes cousines et mon binôme Meryem ;
A mes collègues TOUDJI-A et DLIMI-A
A mon encadreur Mr. Demri Mohamed ;
A mes amies et tous ceux que me sont cher.*

Dédicaces

*A mes parents ;
A mes frères et mes sœurs ;
A toute ma famille TOUATI ;
A toute la famille BENBA ;
A mes cousines et mon binôme Rekia ;
A mes collègues TOUDJI-A et DLIMI-A
A mon encadreur Mr. Demri Mohamed ;
A mes amies et tous ceux que me sont cher.*

Remerciement

Tout d'abord Nous remercions Dieu le tout puissant qui nous donne le courage et la force de mener à bien ce travail.

On' a tant aussi à remercier très vivement et à exprimer notre gratitude :

- Tout d'abord à notre encadreuse Mr. DEMRI Mohammed pour ses conseils, et sa grande contribution dans la réalisation de notre travail et surtout pour sa grande patience et gentillesse.

- Aux membres du département d'informatique pour leur amabilité et leur soutien.

- A tout le corps des enseignants de l'Université d'Adrar.

Table des matières

Résumé	i
Dédicace.....	ii
Dédicace.....	iii
Remerciement	iv
Sommaire	v
Liste des tableaux	viii
Liste des figures	ix
Liste des abréviations	x
Introduction générale.....	1
1 I Chapitre 01 :Introduction aux réseaux de capture sans fil.....	3
1.1 Introduction.....	4
1.2 Les capteurs	4
1.2.1 Définition	4
1.2.2 Architecture.....	5
1.3 Réseau Ad-hoc	7
1.4 Les réseaux de capture sans fil.....	8
1.4.1 L'architecture de communication des réseaux de capture sans fil.....	8
1.4.2 Caractéristique requises des réseaux de capture sans fil	9
1.4.3 Mécanismes utilisés pour atteindre les caractéristiques des réseaux de capteurs sans fil ...	10
1.4.4 Collection des informations	11
1.5 Pile protocolaire	11
1.5.1 Rôle de chaque couche.....	12
1.6 Topologies des réseaux de capteurs	14
1.6.1 Topologie en étoile.....	14
1.6.2 Topologie en toile (Mesh Network).....	14
1.6.3 topologie hybride	15
1.7 Domain d'application.....	15
1.7.1 Applications militaires	15
1.7.2 Applications environnementales	16
1.7.3 Applications médicales	16
1.8 Conclusion	17
2 II.Chapitre 02 : Les protocoles de routage dans les RCSFs.....	18
2.1 Introduction.....	18
2.2 Contraintes de routage dans les réseaux de capteurs sans fil	18

2.3	Classification des protocoles de routage dans les RCSF	18
2.3.1	Protocole de routage multi-chemin	19
2.3.2	Protocole de routage basé sur la négociation des données	19
2.3.3	Protocole de routage basé sur les interrogations	19
2.3.4	Protocoles de routage basé sur la QOS	19
2.3.5	Les protocoles de routage plat (flat based-routing).....	19
2.4	Protocole de routage hiérarchique.....	20
II.4.1	Protocole LEACH (Low-Energy Adaptive Clustering Hierarchy)	21
II.4.2	Le protocole PEGASIS	23
II.4.3	TEEN et APTEEN (Adaptive Threshold-sensitive Energy Efficient sensor Network protocol)	23
II.4.4	CTLMN (Clustering Technique for Large multi hop Mobile wireless Networks)	24
II.4.5	HEED (Hybrid, Eenergy-Efficient, Distributed approach)	24
2.5	Protocoles de routage non hiérarchiques.....	25
2.5.1	DSDV (Destination Sequenced Distance Vector).....	26
2.5.2	.GSR (Global State Routing)	26
2.5.3	FSR (Fisheye State Routing).....	27
2.5.4	AODV (Ad-hoc On Demand Distance Vector)	28
2.5.5	DSR (Dynamic Source Routing).....	29
2.5.6	SPIN (Sensor Protocols for Information via Negotiation).....	29
2.5.7	OLSR (optimized link State ROUTING).....	30
2.5.8	GPSR (Greedy preimeter Statless Routing).....	31
2.6	Conclusion	31
3	.Chapitre 03 Optimisation par essaim des particules	31
3.1	.Introduction.....	31
3.2	Optimisation Par Essaims de Particule :	31
3.3	UN PEU D'HISTOIRE	31
3.4	Topologie du voisinage :.....	32
3.5	L'ALGORITHME DE PSO.....	32
3.5.1	Vitesse maximale et coefficient de constriction.....	35
3.5.2	Critères d'arrêt	35
3.6	Conclusion	36
4	Chapitre 04 : Implémentation et discussion	37
4.1	Introduction.....	37
4.2	Description détaillée de l'algorithme proposé	38

4.2.1	Formation des CHs.....	38
4.2.2	Algorithme de APSOC.....	39
4.2.3	Fonction objective.....	39
4.3	A propos de l'environnement MATLAB	42
4.4	Les étapes d'exécution de l'application	42
4.4.1	Description et paramètres de simulation.....	42
4.4.2	Paramètres de simulation	44
4.5	Simulation et résultat	46
4.6	Conclusion	49
	Conclusion générale	50
	Références	51



Liste des tableaux

Tableau IV.1 : Les paramètres de simulation.....	46
Tableau IV.2 : Les paramètres de l'algorithme APSOC.....	46

Liste des figures

Figure 1.1 : Quelques exemples des capteurs.....	6
Figure 1.2 : L'architecture d'un nœud capteur.....	7
Figure 1.3 : Réseau en mode ad hoc.....	8
Figure 1.4 : Architecture de communication d'un RCSF.....	9
Figure 1.5 : Modèle en couches du réseau de capteurs sans fil.....	13
Figure 1.6 : Topologies des réseaux de capteurs.....	15
Figure 1.7 : Applications des réseaux de capteurs.....	15
Figure 1.8 : Réseau de capteurs militaire.....	16
Figure 1.9 : Ensemble de capteurs dans un corps humain.....	16
Figure 2.1 : Classification des Protocoles de routage pour les RCSF.....	20
Figure 2.2 : Routage plat.....	21
Figure 2.3 : Topologie hiérarchique.....	22
Figure 2.4 : Formation des clusters dans LEACH.....	22
Figure 2.5 : Formation des chaînes dans PEGASIS.....	23
Figure 2.6 : Technique "œil de poisson" dans le protocole FSR.....	27
Figure 2.7 : Fonctionnement de la procédure de demande de route dans AODV.....	28
Figure 2.8 : Fonctionnement du protocole SPIN.....	30
Figure 3.1 : différents types de topologie pour un essaim de particules.....	32
Figure 4.1 : L'organigramme de la méthode APSOC.....	41
Figure 4.2 : L'interface principale de simulation.....	43
Figure IV.3 : L'interface d'affichage.....	44
Figure IV.4 : Modèle de consommation d'énergie pour.....	45
Figure IV.5 : la consommation d'énergie après chaque rond.....	47
Figure IV.6 : Nombre des nœuds vivants.....	47
Figure IV.7 : Données envoyées de CHs vers station de base.....	48





Liste des figures

Figure 1.1 : Quelques exemples des capteurs.....	6
Figure 1.2 : L'architecture d'un nœud capteur.....	7
Figure 1.3 : Réseau en mode ad hoc.....	8
Figure 1.4 : Architecture de communication d'un RCSF.....	9
Figure 1.5 : Modèle en couches du réseau de capteurs sans fil.....	13
Figure 1.6 : Topologies des réseaux de capteurs.....	15
Figure 1.7 : Applications des réseaux de capteurs.....	15
Figure 1.8 : Réseau de capteurs militaire.....	16
Figure 1.9 : Ensemble de capteurs dans un corps humain.....	16
Figure 2.1 : Classification des Protocoles de routage pour les RCSF.....	20
Figure 2.2 : Routage plat.....	21
Figure 2.3 : Topologie hiérarchique.....	22
Figure 2.4 : Formation des clusters dans LEACH.....	22
Figure 2.5 : Formation des chaînes dans PEGASIS.....	23
Figure 2.6 : Technique "œil de poisson" dans le protocole FSR.....	27
Figure 2.7 : Fonctionnement de la procédure de demande de route dans AODV.....	28
Figure 2.8 : Fonctionnement du protocole SPIN.....	30
Figure 3.1 : Différents types de topologie pour un essaim de particules.....	32
Figure 4.1 : L'organigramme de la méthode APSOC.....	41
Figure 4.2 : L'interface principale de simulation.....	43
Figure 4.3 : L'interface d'affichage.....	44
Figure 4.4 : Modèle de consommation d'énergie pour.....	45
Figure 4.5 : la consommation d'énergie après chaque rond.....	47
Figure 4.6 : Nombre des nœuds vivants.....	47
Figure 4.7 : Données envoyées de CHs vers station de base.....	48





Les progrès technologiques dans le domaine de l'électronique et le domaine des communications sans fil ont donné naissance à la création d'équipements à faible coût, appelés nœuds capteurs ou "motes". Ces derniers peuvent communiquer via des ondes hertziennes (la radio) et collaborer entre eux pour former un réseau de capteurs sans fil (RCSF).

Ce nouveau type de réseaux présente une grande amélioration comparé aux capteurs classiques qui sont généralement positionnés loin du phénomène surveillé. La position des nœuds utilisés n'est pas obligatoirement conçue au préalable, ce qui permet leur déploiement aléatoire dans les terrains inaccessibles ou pendant les opérations de secours aux cas de désastres.

Les réseaux de capteurs sont constitués de plusieurs capteurs minuscules ou nœuds ayant une caractéristique essentielle résidant dans l'absence d'infrastructure fixe et ayant une topologie changeante due à la mobilité des capteurs et pose le problème de l'épuisement de leurs batteries. Ces capteurs transmettent régulièrement les données au nœud central où les traitements sont accomplis et les données sont fusionnées jusqu'à l'aboutissement à la station de base.

Les principaux problèmes dans les réseaux de capteurs sans fil ou les WSNs "Wireless Sensor Networks" sont le protocole de routage, l'énergie consommée par les nœuds, la sécurité, l'agrégation de données, la mobilité imprévisible des nœuds, etc. Ces capteurs sont parfois déployés dans des zones hostiles, Il est donc nécessaire d'avoir une stratégie efficace qui prend en considération l'énergie du réseau pour Augmenter son durée de vie en réduisant la perte d'énergie.

Les progrès récents dans les réseaux de capteurs sans fil ont conduit au développement de nombreux protocoles spécifiquement conçus pour les réseaux de capteurs où la présentation de l'énergie est un facteur essentiel.

Ces problèmes sont relatifs au routage de données sur les RCSFs et plus précisément le routage hiérarchique de données, et pour répondre à cette problématique, nous proposons un nouveau protocole de routage (APSOC).

On a essayé d'adapter un nouveau protocole de routage basé sur un algorithme bio-inspirée qui simule le comportement des essaims afin de sélectionner les CHs et de regrouper des clusters dans les réseaux de capteurs sans fils.

Notre travail est organisé en quatre chapitres selon un plan méthodologique suivant :

Dans le premier chapitre, nous présenterons les réseaux de capteurs sans fil leurs architectures de communication et leurs applications. Nous discuterons également les principaux facteurs et contraintes qui influencent la conception des réseaux de capteurs sans fil.

Dans le deuxième chapitre, nous présenterons un état de l'art sur un ensemble des protocoles des routages hiérarchiques et non hiérarchiques.

Le troisième chapitre, nous présente L'algorithme de PSO (Particle Swarm Optimisation) et son principe de fonctionnement.

Dans le quatrième nous présenterons notre protocole de routage hiérarchique pour les RCSFs (APSOC). Ce dernier sera implémenté et évalué par simulation et selon plusieurs critères. Ensuite, les résultats obtenus seront analysés et discutés .

Les abréviations

RCSF: Réseau de capture sans fil.

WSN: Wireless Sensor Networks.

CH: Cluster Head.

ADC: Analog Digital Converter.

MANET: Mobile Ad hoc NETwork.

PDR: Paquet Delivery Ratio.

LEACH: Low Energy Adaptive Clustering Hierarchical.

PEGASIS: Power-Efficient GATHERing in Sensor Information Systems.

TEEN et APTEEN: Adaptive Threshold-sensitive Energy Efficient sensor Network protocol.

CTLMN: Clustering Technique for Large multi hop Mobile wireless Networks.

HEED: Hybrid, Eenergy-Efficient, Distributed approach.

DSDV: Destination Sequenced Distance Vector.

GSR: Global State Routing.

DBF: Distributed Bellman-Ford.

FSR: Fisheye State Routing.

AODV: Ad-hoc On Demand Distance Vector.

DSR: Dynamic Source Routing.

SPIN: Sensor Protocols for Information via Negotiation.

OLSR: optimized link State ROUTING.

MPR: MultiPoint Relaying.

GPSR: Greedy perimeter Stateless Routing.

PSO: particule swarm optimization.

APSOC: Adaptive Particle Swarm Optimization for clustering.

QOS: Qualite of service.

Introduction

Les progrès technologiques dans le domaine de l'électronique et le domaine des communications sans fil ont donné naissance à la création d'équipements à faible coût, appelés nœuds capteurs ou "motes". Ces derniers peuvent communiquer via des ondes hertziennes (la radio) et collaborer entre eux pour former un réseau de capteurs sans fil (RCF).

Ce nouveau type de réseaux présente une grande amélioration comparé aux capteurs classiques qui sont généralement positionnés loin du phénomène surveillé. La position des nœuds utilisés n'est pas obligatoirement conçue au préalable, ce qui permet leur déploiement aléatoire dans les terrains inaccessibles ou pendant les opérations de secours aux cas de désastres.

Les réseaux de capteurs sont constitués de plusieurs capteurs minuscules ou nœuds ayant une caractéristique essentielle résidant dans l'absence d'infrastructure fixe et ayant une topologie changeante due à la mobilité des capteurs et pose le problème de l'épuisement de leurs batteries. Ces capteurs transmettent régulièrement les données au nœud central où les traitements sont accomplis et les données sont fusionnées jusqu'à l'aboutissement à la station de base.

Les principaux problèmes dans les réseaux de capteurs sans fil ou les WSNs "Wireless Sensor Networks" sont le protocole de routage, l'énergie consommée par le nœud, la sécurité, l'agrégation de données, la mobilité imprévisible des nœuds, etc. Ces capteurs sont parfois déployés dans des zones hostiles, Il est donc nécessaire d'avoir une stratégie efficace qui prend en considération l'énergie du réseau pour Augmenter sa durée de vie en réduisant la perte d'énergie.

Les progrès récents dans les réseaux de capteurs sans fil ont conduit au développement de nombreux protocoles spécifiquement conçus pour les réseaux de capteurs où la présentation de l'énergie est un facteur essentiel.

Ces problèmes sont relatifs au routage de données sur les RCFs et plus précisément le routage hiérarchique de données, et pour répondre à cette problématique, nous proposons un nouveau protocole de routage (APSOC).

On essayera d'adapter un nouveau protocole de routage basé sur un algorithme bio-inspirée qui simule le comportement des essaims a fin de sélectionner les CHs et de regrouper des clusters dans les réseaux de capteurs sans fils.

Notre travail est organisé en quatre chapitre selon un plan méthodologique suivant :

Dans le premier chapitre, nous présenterons les réseaux de capteurs sans fil leurs architectures de communication et leurs applications. Nous discuterons également les principaux facteurs et contraintes qui influencent la conception des réseaux de capteurs sans fil.

Dans le deuxième chapitre, nous présenterons quelques solutions proposées pour conserver la consommation d'énergie, un état de l'art sur un ensemble des protocoles des routages hiérarchiques et non hiérarchiques.

Le troisième chapitre, nous présenterons l'algorithme de PSO (particle swarm optimisation) et son principe de fonctionnement. Et dans le quatrième on présente notre protocole de routage hiérarchique pour les RCSFs (APSOC). Ce dernier était évalué selon plusieurs facteurs. Ensuite, nous détaillerons le simulateur que nous avons conçu pour la validation de notre protocole et nous discuterons les résultats obtenus.



CHAPITRE 01

INTRODUCTION AUX

RESEAUX DE CAPTURE

SANS FIL



I.1 Introduction

Les progrès technologiques dans le domaine de l'électronique et le domaine des communications sans fil ont donné naissance à la création d'équipements miniaturisés à faible coût, appelés nœuds capteurs. Ces derniers peuvent communiquer via des ondes hertziennes (la radio) et collaborer entre eux pour former un réseau de capteurs.

Les réseaux de capteurs sans fil (RCSF)- Wireless Sensor Networks (WSN) - sont considérés comme un type spécial de réseaux ad hoc. Les nœuds de ce type de réseau consistent en un grand nombre de micro-capteurs capables de récolter et de transmettre des données environnementales d'une manière autonome. La position de ces nœuds n'est pas obligatoirement prédéterminée. Ils sont dispersés aléatoirement à travers une zone géographique appelée champ de captage, qui définit le terrain d'intérêt pour le phénomène capté. Les données captées sont acheminées à un nœud considéré comme un "point de collecte", appelé nœud puits (ou sink). Ce dernier peut être connecté à l'utilisateur du réseau via Internet ou un satellite. Ainsi, l'utilisateur peut adresser des requêtes aux autres nœuds du réseau, précisant le type de données requises et récolter les données environnementales captées par le biais du nœud puits.

I.1 Les capteurs

I.1.1 Définition

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

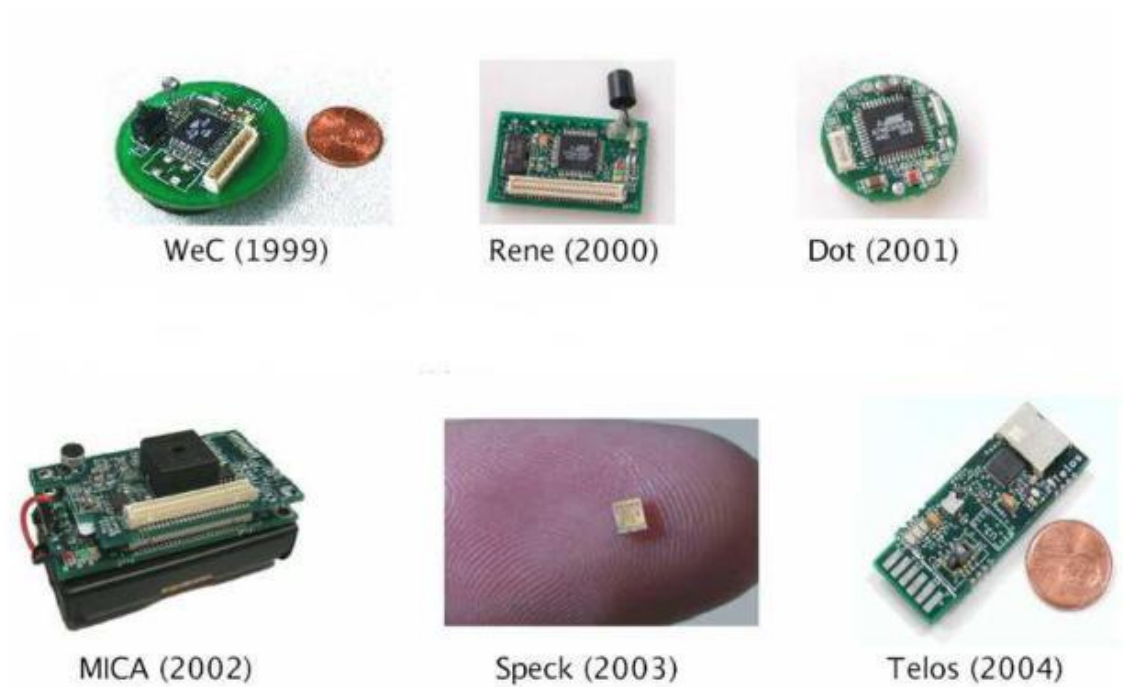


Figure 1.1 : Quelques exemples de capteurs.

I.1.2 Architecture

Un capteur est composé de quatre unités de base [1] :

1. **L'unité d'acquisition** : Elle est généralement composée de deux sous-unités qui sont les capteurs et les convertisseurs analogique-numérique ADCs (Analog Digital Converter). Les capteurs obtiennent des mesures sur les paramètres environnementaux et les transforment en signaux analogiques. Les ADCs convertissent ces signaux analogiques en signaux numériques.
2. **L'unité de traitement** : Elle est composée de deux interfaces qui sont une interface avec l'unité d'acquisition et une autre avec le module de transmission. Elle contrôle les procédures permettant au nœud de collaborer avec les autres nœuds pour réaliser les tâches d'acquisition et stocker les données collectées.
3. **Un module de communication (Transceiver)** : Il est composé d'un émetteur/récepteur permettant la communication entre les différents nœuds du réseau via un support de communication radio.

4. **Batterie** : Elle alimente les unités que nous avons citées et elle n'est généralement ni rechargeable ni remplaçable. La capacité d'énergie limitée au niveau des capteurs représente la contrainte principale lors de conception de protocole pour les réseaux de capteurs.

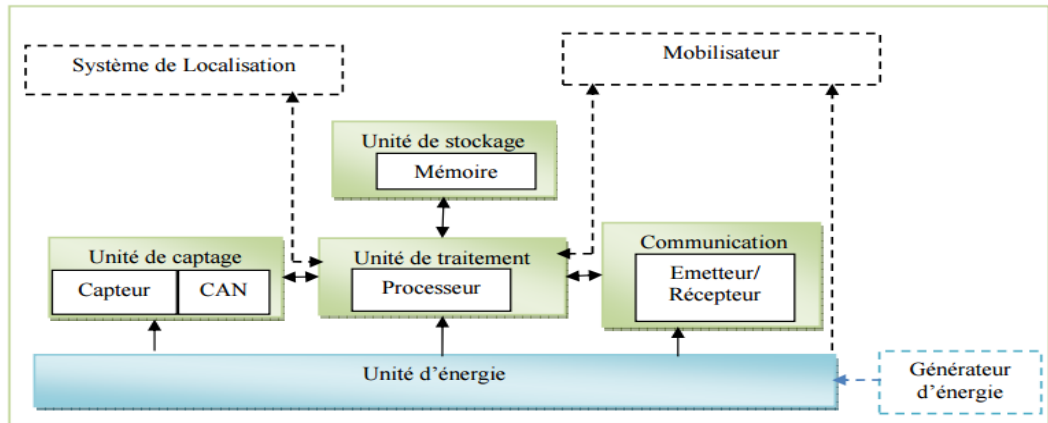


Figure 1.2 : L'architecture d'un nœud capteur

I.1.3 Classification

Les capteurs ont plusieurs modes de classification :

- **Capteurs passifs** : Ils n'ont pas besoin d'apport d'énergie extérieure pour fonctionner (exemple : thermistance, potentiomètre, thermomètre à mercure...). Ce sont des capteurs modélisables par une impédance. Une variation du phénomène physique étudié (mesuré) engendre une variation de, l'impédance.
- **Capteurs actifs** : Ils sont constitués d'un ou d'un ensemble de transducteurs alimentés (exemple : chronomètre mécanique, jauge d'extensomètre appelée aussi jauge de contrainte, gyromètre...). Ce sont des capteurs que l'on pourrait modéliser par des générateurs comme les systèmes photovoltaïques et électromagnétiques. Ainsi ils génèrent soit un courant, soit une tension en fonction de l'intensité du phénomène physique mesuré.

Les capteurs peuvent aussi faire l'objet d'une classification par type de sortie :

- **Capteurs analogiques** : Le signal des capteurs numériques peut être du type : sortie tension, sortie courant, règle graduée ... etc. (Quelques capteurs analogiques typiques : Capteur à jauge de contrainte, LVDT...etc.).

- **Capteurs numériques** : Le signal des capteurs numériques peuvent être du type Train d'impulsions avec un nombre précis d'impulsions ou avec une fréquence précise, Code numérique binaire, Bus de terrain ... etc. [1].

I.2 Réseau Ad-hoc

Un réseau ad hoc, connu aussi sous le nom de MANET (Mobile Ad hoc NETwork), est un réseau dont la topologie ne bénéficie d'aucune infrastructure fixe préexistante ou administration topologie ne bénéficie d'aucune infrastructure fixe préexistante ou administration centralisée.

Un réseau ad hoc est un réseau sans fil auto-configurable. Lorsque deux machines ou plus se retrouvent dans le même secteur géographique, elles doivent se reconnaître pour pouvoir s'échanger des données. Le réseau doit se configurer automatiquement ; périodiquement ou à la demande pour assurer la liaison entre ces machines [2].

Dans un réseau ad-hoc les hôtes mobiles doivent former, d'une manière ad-hoc, une sorte d'architecture globale qui peut être utilisées comme infrastructure du système. Les applications des réseaux ad-hoc sont nombreuses, on cite l'exemple classique de leur application dans le domaine militaire et les autres applications de tactique comme les opérations de secours et les missions d'exploration.

Du fait que le rayon de propagation des transmissions des hôtes soit limité, et afin que le réseau ad-hoc reste connecté, (c'est à dire tout unité mobile peut atteindre toutes autre), il se peut qu'un hôte mobile se trouve dans l'obligation de demander de l'aide à un autre hôte pour pouvoir communiquer avec son correspondant. Il se peut donc que l'hôte destination soit hors de la portée de communication de l'hôte source, ce qui nécessite l'emploi d'un routage interne par des nœuds intermédiaires afin de faire acheminer les paquets de messages à la bonne destination.

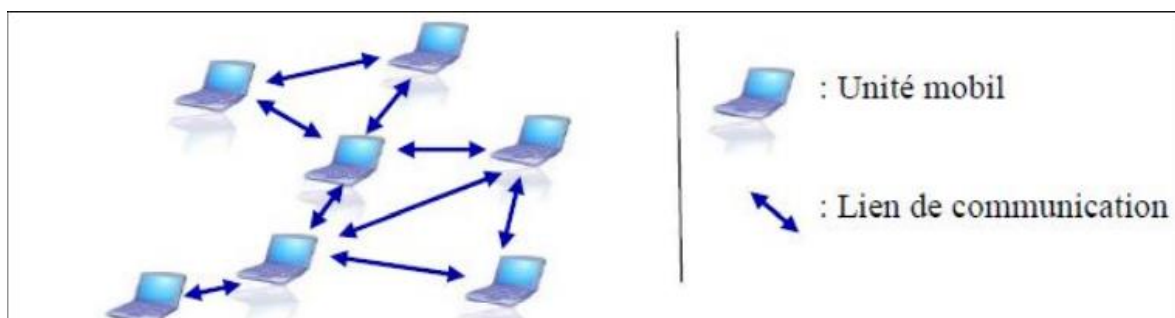


Figure1.3 : Réseau en mode ad hoc.

I.3 Les réseaux de capture sans fil

Les réseaux de capteurs sans fil ou WSN (Wireless Sensor Networks) constituent une catégorie particulière de réseaux ad hoc. Ces derniers sont conçus pour répondre à des problématiques de communications où l'homme est souvent un acteur principal (accès à un réseau global comme Internet, téléphonie, télécommande. . .). Les WSN's offrent des moyens de communication très souvent spontanés entre objets autonomes, généralement sans aucune intervention humaine. Il existe des différences considérables entre les réseaux de capteurs et les réseaux ad hoc, donc dans la plus part des cas on ne pourra pas utiliser les mêmes protocoles

Un Réseau de Capteurs Sans Fil (RCSF) est un ensemble de dispositifs très petits, nommés nœuds capteurs, variant de quelques dizaines d'éléments à plusieurs milliers. Dans ces réseaux, chaque nœud est capable de surveiller son environnement et de réagir en cas de besoin en envoyant l'information collectée à un ou plusieurs points de collecte, à l'aide d'une connexion sans fil [3].

I.3.1 L'architecture de communication des réseaux de capture sans fil

Les nœuds capteurs sont habituellement dispersés dans une zone de capture. Chacun de ces nœuds a la possibilité de collecter les données et de les router vers une ou plusieurs stations de base (sink nodes). Ce dernier est un point de collecte de données capturées. Il peut communiquer les données collectées à l'utilisateur final à travers un réseau de communication, éventuellement l'Internet. L'utilisateur peut à son tour utiliser la station de base comme passerelle, afin de transmettre ses requêtes au réseau. Cette architecture est illustrée dans la figure [1.3] [3].

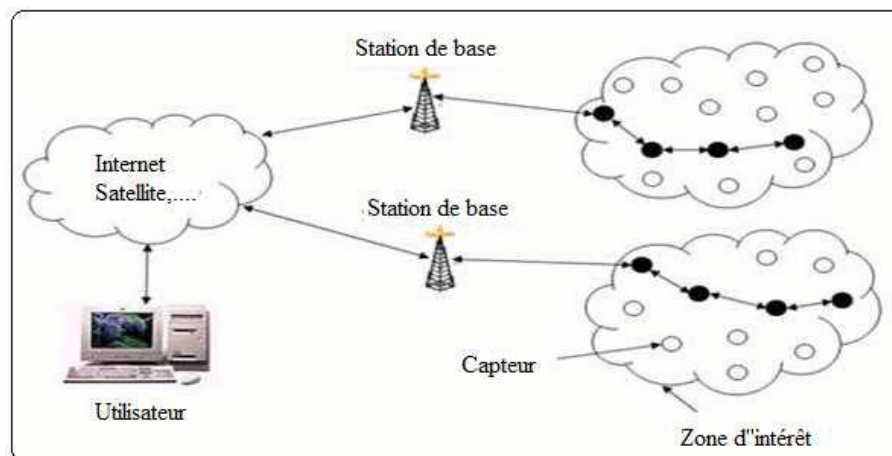


Figure 1.4 : Architecture de communication d'un RCSF.

✍ **Les types d'architecture des RCSFs :**

▪ **Les réseaux de captures sans fil plats**

Est un réseau homogène, où tous les nœuds sont identiques en termes de batterie et de complexité du matériel et ont le même rôle, excepté le nœud puits qui joue le rôle d'une passerelle et qui est responsable de la transmission de l'information collectée à l'utilisateur final. Selon le service et le type de capteurs, une densité de capteurs élevée (plusieurs nœuds capteurs/m²) ainsi qu'une communication multi-sauts peut être nécessaire pour l'architecture plate. Ce type de solution permet une grande tolérance aux pannes, cependant, elle souffre d'une faible scalabilité. En effet, si tous les nœuds opèrent de la même façon et d'une manière distribuée, on aura un grand nombre de messages de contrôle nécessaires pour le bon fonctionnement du réseau.

▪ **Les réseaux de captures sans fil hiérarchiques :**

Ont été introduites en divisant les nœuds en plusieurs niveaux de responsabilité. L'une des méthodes la plus employée est le Clustering, avec laquelle le réseau est partitionné en groupes appelés clusters. Un cluster est constitué d'un chef (Cluster Head) et de ses membres. Suivant l'application, les membres peuvent être des voisins directs ou indirects du Cluster Head donc il est plus facile d'intégrer un mécanisme d'agrégation au système.

I.3.2 Caractéristique requises des réseaux de capture sans fil

Un ensemble de caractéristiques sont importantes pour l'accomplissement des tâches assignées aux applications, Les plus importants sont [4] :

- ✍ **Le Type de service** : on s'attend à ce que le RCSF (réseaux de capteurs sans fil) offre à l'utilisateur, des informations significatives sur l'objet d'intérêt.
- ✍ **La Qualité de service (QoS)** : C'est une métrique de la qualité de service qui va être offerte par un RCSF à ses utilisateurs/applications. Le niveau de QoS est défini par un ensemble d'attributs comme le temps d'attente, la largeur de bande, et la perte de paquets qu'on relie directement avec le type

de service du réseau. Dans QoS pour les RCSFs, la quantité et la qualité d'information extraites à partir des puits deviennent appropriées.

- ✍ **Tolérance aux fautes** : il est important que le RCSF soit capable de traiter l'échec des nœuds capteurs. Une manière reconnue de satisfaire cette contrainte est de réaliser un déploiement redondant des nœuds capteurs.
- ✍ **La durée de vie** : C'est la durée pendant laquelle le réseau reste opérationnel. On s'attend à ce que le RCSF puisse fonctionner au moins pendant le temps requis pour accomplir la tâche donnée. Néanmoins, la définition de la durée de vie dépend de l'application du RCSF et elle est en relation directe avec le fonctionnement efficace du réseau.
- ✍ **Scalabilité** : Cette caractéristique traduit la capacité de maintenir la performance indépendamment de la taille du réseau. Comme un grand nombre de nœuds de capteur peuvent être employés dans les applications de RCSF, les architectures et les protocoles doivent fournir le support approprié pour maintenir efficacement les services fournis par le réseau.
- ✍ **Maintenance** : les changements dans l'environnement du réseau, par exemple, l'apparition de nœuds de capteurs avec des batteries épuisées, exigent une solution permettant l'adaptation et le maintien des services du RCSF.
- ✍ **Programmation flexible** : C'est la capacité des nœuds de capteur à modifier les options de traitement des données acquises et à effectuer des changements et des ajustements de leurs tâches.

I.3.3 Mécanismes utilisés pour atteindre les caractéristiques des réseaux de capteurs sans fil

Afin de réaliser les caractéristiques précédentes, des nouveaux mécanismes de communications, d'architectures et de protocoles doivent être développés. Les mécanismes typiques considérés dans les RCSFs sont [3] :

- ✍ **Communication multi-sauts** : Pour les communications à longue distance, l'utilisation des nœuds intermédiaires pour transmettre les paquets, peut induire une réduction de l'énergie exigée.
- ✍ **Utilisation efficace d'énergie** : C'est là un mécanisme clé pour offrir et soutenir sur une durée longue l'opérationnalité du réseau.

- ✍ **Auto-configuration** : Ce mécanisme est employé dans différents aspects des RCSF. Le nœud capteur devrait être capable d'adapter ses paramètres de service pour tenir compte des défaillances des autres nœuds, des obstacles et de l'ajout de nœuds au réseau.
- ✍ **Collaboration et traitement dans le réseau** : Selon l'application, il est parfois exigé qu'un groupe de nœuds de capteurs interagissent afin de détecter un événement ou faire un traitement plus complet d'informations. La considération des approches des offres de traitement dans le réseau comme l'agrégation de données, qui réduisent la quantité de données transmises et par conséquent améliore l'efficacité énergétique du réseau.

I.3.4 Collection des informations

- **A la demande** : Lorsque l'on souhaite connaître l'état de la zone d'intérêt à l'instant t , le nœud puits émet une requête en broadcast vers tous les nœuds déployés dans la zone d'intérêt pour que ces derniers remontent leur dernier relevé vers le nœud puits. Les informations sont alors acheminées vers le nœud puits par le biais d'une communication multi-sauts.
- **A suite à un événement** : Un événement se produit en un point de la zone d'intérêt (changement brusque de température, détection d'un mouvement...), les capteurs situés à proximité de ce point cible remontent alors les informations relevées et les acheminent jusqu'au nœud puits¹.

I.4 Pile protocolaire

Comme tous les types de réseaux, les RCSF utilisent une architecture de communication en couches, ce sont les cinq premières couches du modèle OSI. Chaque couche a son propre rôle et ses propres protocoles pour atteindre son objectif [4].

¹<http://igm.univ-mlv.fr/~dr/XPOSE2006/Bunel/Presentation.html#unreseau>, visité le 05/02/2017

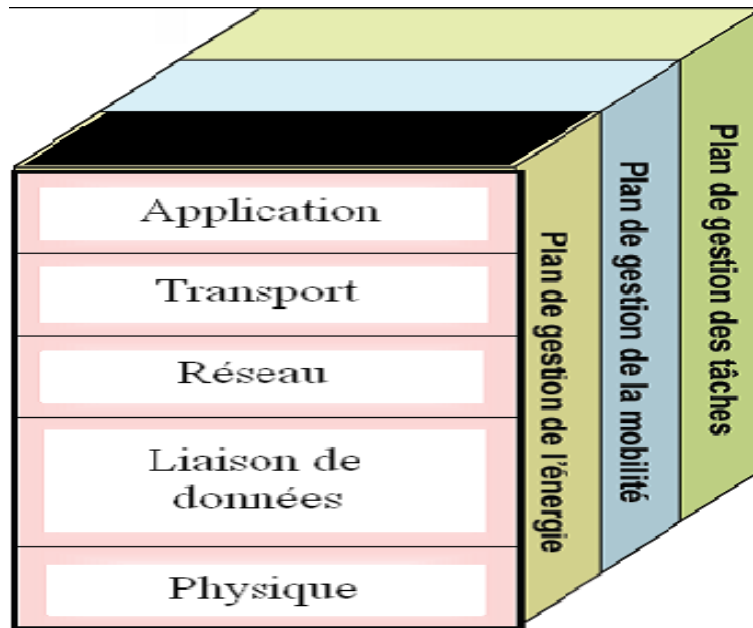


Figure 1.5 : Modèle en couches du réseau de capteurs sans fil.

I.4.1 Rôle de chaque couche

- **Couche physique :** Matériels pour envoyer et recevoir les données.
- **Couche liaison de données :** Gestion des liaisons entre les nœuds et les stations de base, contrôle d'erreurs. Spécifie comment les données sont expédiées entre deux nœuds/routeurs dans une distance d'un saut. Elle est responsable du multiplexage des données, du contrôle d'erreurs, de l'accès au media,... Elle assure la liaison point à point et multi-point dans un réseau de communication.
- **Couche réseau :** Routage et transmission des données. Dans la couche réseau le but principal est de trouver une route et une transmission fiable des données, captées, des nœuds capteurs vers le puits "sink" en optimisant l'utilisation de l'énergie des capteurs. Ce routage diffère de celui réseaux de transmission ad hoc sans fils par les caractéristiques suivantes :
 - Il n'est pas possible d'établir un système d'adressage global pour le grand nombre de nœuds.
 - Les applications des réseaux de capteurs exigent l'écoulement des données mesurées de sources multiples à un puits particulier.
 - Les multiples capteurs peuvent produire de mêmes données à proximité d'un phénomène (redondance).

- Les noeuds capteur exigent ainsi une gestion soigneuse des ressources. En raison de ces différences, plusieurs nouveaux algorithmes ont été proposés pour le problème de routage dans les réseaux de capteurs.
- **Couche transport** : Transport des données, contrôle de flux. Cette couche est chargée du transport des données, de leur découpage en paquets, du contrôle de flux, de la conservation de l'ordre des paquets et de la gestion des éventuelles erreurs de transmission.
- **Couche application** : Interface pour les applications au haut niveau. Cette couche assure l'interface avec les applications. Il s'agit donc du niveau le plus proche des utilisateurs, géré directement par les logiciels.
- **Plan de gestion** :

En outre, les plans de gestion de l'énergie, de la mobilité et des tâches surveillent la puissance, le mouvement et la distribution des tâches, respectivement, entre les nœuds capteurs. Ces plans de gestion sont nécessaires, de sorte que les nœuds capteurs puissent fonctionner ensemble d'une manière efficace pour préserver l'énergie, router des données dans un réseau de capteurs mobile et partager les ressources entre les nœuds capteurs. Du point de vue global, il est plus efficace d'utiliser des nœuds capteurs pouvant collaborer entre eux. La durée de vie du réseau peut être ainsi prolongée.

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

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

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

I.6 Topologies des réseaux de capteurs

Un réseau de capteurs sans fil est composé d'un ensemble de nœuds capteurs et des Gateway qui s'occupent de collecter les données des capteurs et de les transmettre à l'utilisateur via l'internet ou le satellite, il existe plusieurs topologies pour les réseaux de capteurs [4].

I.6.1 Topologie en étoile

La topologie en étoile est un système uni-saut. Tous les nœuds envoient et reçoivent seulement des données avec la station de base. Cette topologie est simple et elle demande une faible consommation d'énergie, mais la station de base est vulnérable et la distance entre les nœuds et la station est limitée.

Avantage : simplicité et faible consommation d'énergie des nœuds, moindre latence de communication entre les nœuds et la station de base.

Inconvénient : la station de base est vulnérable, car tout le réseau est géré par un seul nœud [4].

I.6.2 Topologie en toile (Mesh Network)

La topologie en toile est un système multi-saut. La communication entre les noeuds et la station de base est possible. Chaque noeud a plusieurs chemins pour envoyer les données. Cette topologie a plus de possibilités de passer à l'échelle du réseau, avec redondance et tolérance aux fautes, mais elle demande une consommation d'énergie plus importante.

Avantage : Possibilité de passer à l'échelle du réseau, avec redondance et tolérance aux fautes.

Inconvénient : Une consommation d'énergie plus importante est induite par la communication multi-sauts. Une latence est créée par le passage des messages des noeuds par plusieurs autres avant d'arriver à la station de base.

I.6.3 topologie hybride

La topologie hybride est un mélange des deux topologies ci-dessus. Les stations de base forment une topologie en toile et les nœuds autour d'elles sont en topologie étoile. Elle assure la minimisation d'énergie dans les réseaux de capteurs [4].

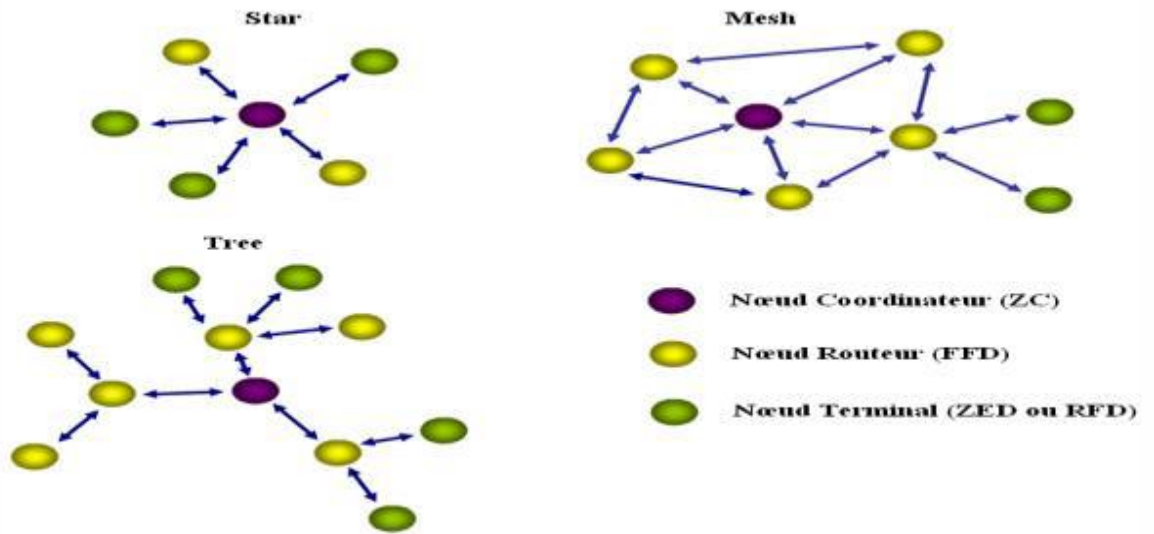


Figure 1.6 : Topologies des réseaux de capteurs.

I.7 Domain d’application



I.7.1 Applications

Figure 1.7 : Applications des réseaux de capteurs.

Comme dans le cas de plusieurs technologies, le domaine militaire a été un moteur initial pour le développement des réseaux de capteurs. Le déploiement rapide, le coût réduit, l'auto-organisation et la tolérance aux pannes des réseaux de capteurs sont des caractéristiques qui rendent ce type de réseaux un outil appréciable dans un tel domaine [3].



Figure 1.8 : Réseau de capteurs militaire.

I.7.2 Applications environnementales

Des thermo-capteurs dispersés à partir d'un avion sur une forêt peuvent signaler un éventuel début d'incendie dans le champ de captage, ce qui permettra une meilleure efficacité pour la lutte contre les feux de forêt [3].

I.7.3 Applications médicales

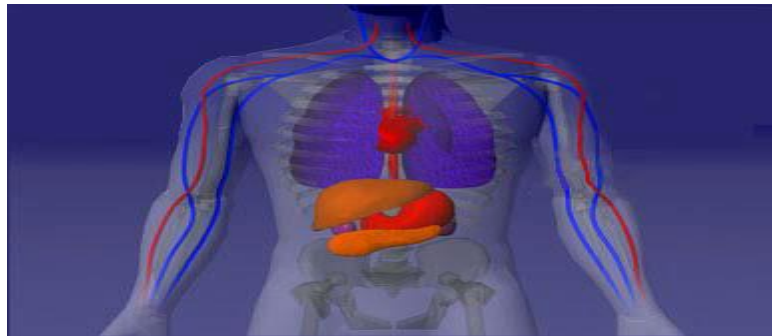


Figure 1.9 : Ensemble de capteurs dans un corps humain.

On pourrait imaginer que dans le futur, la surveillance des fonctions vitales de l'être humain serait possible grâce à des micro-capteurs qui pourront être avalés ou implantés sous la peau.

Actuellement, des micro-caméras qui peuvent être avalées existent. Elles sont capables, sans avoir recours à la chirurgie, de transmettre des images de l'intérieur d'un corps humain avec une autonomie de 24 heures. Les auteurs d'une récente étude, présentent

des capteurs qui fonctionnent à l'intérieur du corps humain pour traiter certains types de maladies [3].

I.8 Conclusion

Nous avons essayé à travers ce chapitre de mettre le point sur l'architecture des RCSFs, ainsi que leurs principaux domaines d'application et leur facteur et défi de conception. Cette mise au point nous a permis de déduire que les protocoles de routage jouent un rôle déterminant et crucial dans la conception des RCSFs. Cela nous a mené à faire une étude des principaux protocoles de routage proposés dans le chapitre qui suit.



CHAPITRE 02

PROTOCOLES DE ROUTAGE

DANS LES RÉSEAUX DE

CAPTURES SANS FIL

Etat de l'art



II.1 Introduction

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

Dans ce chapitre, nous présenterons un état de l'art sur les principaux protocoles de routage dans les RCSF. Cela nous permettra de mieux analyser le principe de fonctionnement de chacun d'eux.

II.2 Contraintes de routage dans les réseaux de capteurs sans fil

Le routage dans les réseaux de capteurs diffère de celui des réseaux Ad Hoc dans les points suivants :

- Il n'est pas possible d'établir un système d'adressage global pour le grand nombre de nœuds.
- les applications des réseaux de capteurs exigent l'écoulement de données mesurées depuis des sources multiples vers la destination finale « *Sink* ».
- les différents capteurs peuvent générer produire les mêmes données à proximité d'un phénomène (problème de la redondance des données).
- les nœuds capteurs exigent ainsi une gestion soignée des ressources. En raison de ces différences, de nouveaux protocoles de routage ont été proposés dans les réseaux de capteurs [4].

II.3 Classification des protocoles de routage dans les RCSF

Les protocoles de routage dans les réseaux peuvent être classés selon deux concepts :

- ✓ La structure de réseau
- ✓ Le mode opératoire de protocole.

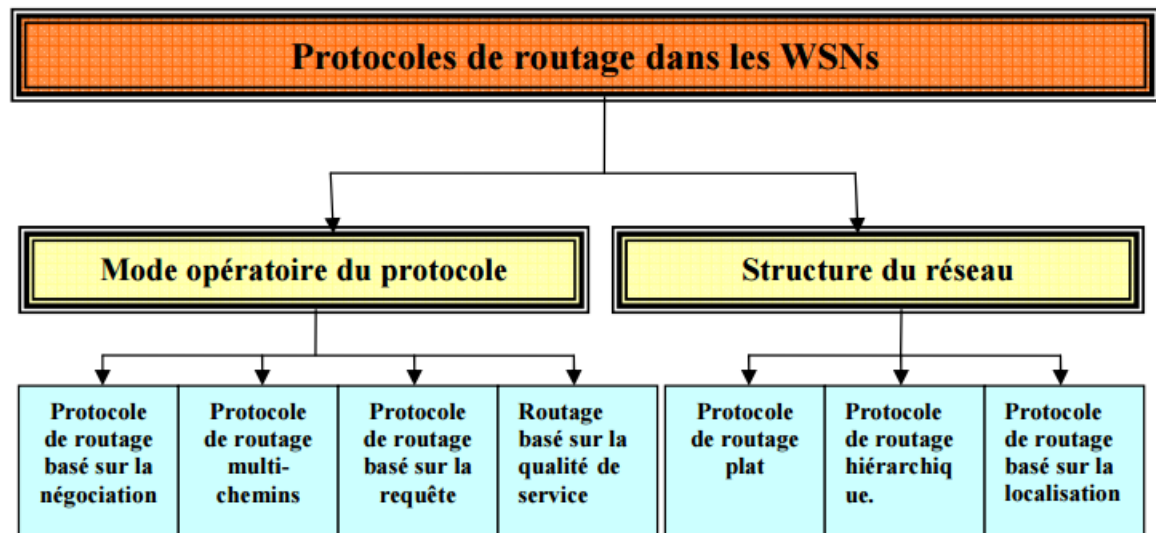


Figure 2.1 : Classification des Protocoles de routage pour les RCSF.

II.3.1 Protocole de routage multi-chemin

Il se base sur l'adoption de plus qu'un chemin menant vers la destination, et ce, pour avoir des chemins de secours si jamais le chemin principal serait rompu.

II.3.2 Protocole de routage basé sur la négociation des données

En détectant le même phénomène, les nœuds capteurs inondent le réseau par les mêmes paquets de données. Ce problème de redondance peut être résolu en employant des protocoles de routage basés sur la négociation. En effet, avant de transmettre, les nœuds capteurs négocient entre eux leurs données en échangeant des paquets de signalisation spéciales, appelés *META-DATA*. Ces paquets permettent de vérifier si les nœuds voisins disposent des mêmes données à transmettre. Cette procédure garantit que seules les informations utiles seront transmises et élimine la redondance des données [4].

II.3.3 Protocole de routage basé sur les interrogations

La collecte des informations sur l'état de l'environnement est initiée par des interrogations envoyées par le nœud « Sink ».

II.3.4 Protocoles de routage basé sur la QOS

Ce type de protocoles tend à satisfaire certaines métriques, pendant la transmission des données vers la destination finale. Parmi ces métriques, nous citons : le délai de bout en bout, la gigue, PDR (*Paquet Delivery Ratio*), énergie consommée.

II.3.5 Les protocoles de routage plat (flat based-routing)

Ces protocoles considèrent que tous les nœuds sont identiques, c'est à dire ont les mêmes fonctions à exécuter sauf le nœud de contrôle (*sink*) qui est chargé de collecter toutes les informations issues des différents nœuds capteurs pour les transmettre vers l'utilisateur final. La décision d'un nœud de router des paquets vers un autre dépendra de sa position et pourra être remise en cause au cours du temps [4].

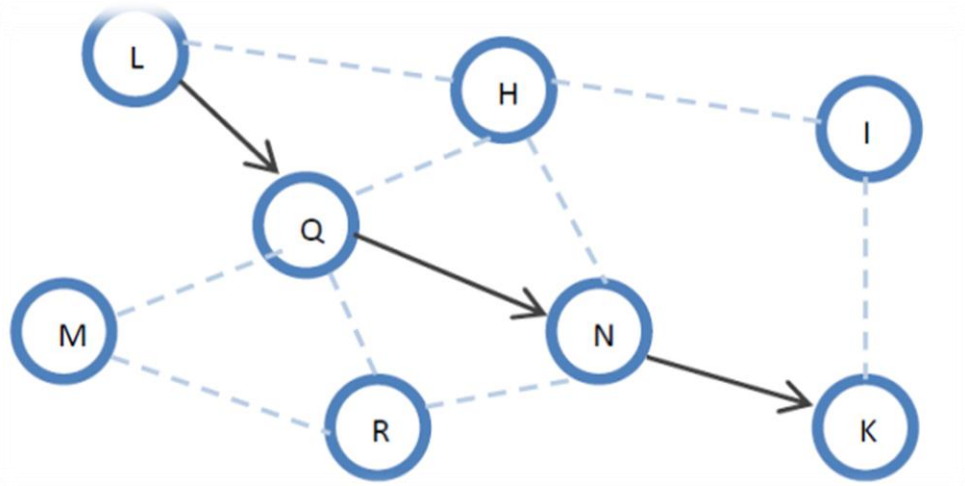


Figure 2.2 : Routage plat

II.4 Protocole de routage hiérarchique

L'objectif principal du routage hiérarchique est de maintenir efficacement la consommation d'énergie de nœuds de capteurs en les impliquant dans la communication multi-hop au sein d'un cluster et en effectuant l'agrégation et la fusion des données afin de diminuer le nombre de messages transmis à la destination.

La formation de clusters est généralement fondée sur la réserve d'énergie des capteurs et sur les capteurs qui sont à proximité de cluster-head. LEACH (Low Energy Adaptive Clustering Hierarchical) [23] est l'une de premières approches de routage pour les réseaux de capteurs. L'idée proposée par LEACH a été une inspiration pour de nombreux protocoles de routage hiérarchique, bien que certains protocoles aient été développés de manière indépendante [13].

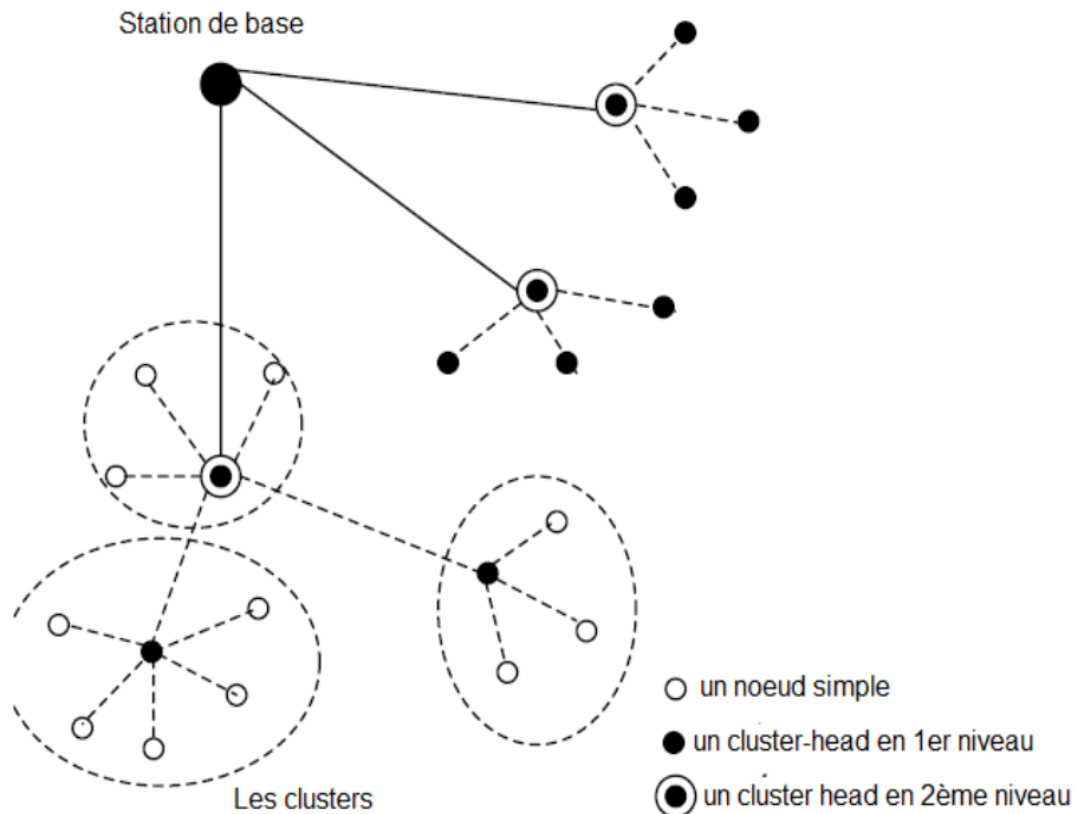


Figure 2.3 : Topologie hiérarchique

II.4.1 Protocole LEACH (Low-Energy Adaptive Clustering Hierarchy)

Est un protocole autoorganisateur basé sur le clustering adaptatif, qui utilise la rotation randomisée des têtes de groupe pour distribuer équitablement la charge d'énergie entre les nœuds capteurs dans le réseau. Il est considéré comme étant l'une des premières approches de routage hiérarchique basées sur le clustering. LEACH est fondé sur deux hypothèses de base

- ✍ la station de base est fixe et est placée loin des capteurs,
- ✍ Tous les nœuds du réseau sont homogènes et limités en énergie.

L'idée derrière LEACH est de former des clusters des nœuds capteurs selon la force reçue du signal et d'utiliser les têtes locales de groupe (cluster head) comme des routeurs pour conduire des données à la station de base. Les dispositifs principaux de LEACH sont :

- ✍ **Coordination et contrôle localisés** pour l'initialisation et le traitement de groupe.
- ✍ **Rotation randomisée de la groupe** (effectuée par "les stations de base" ou "les têtes de cluster").

✍ **Compression locale (agrégation)** Les nœuds CH compressent les données arrivant des nœuds appartenant à leurs groupes respectives, et envoient un paquet d'agrégation à la station de base afin de réduire la quantité d'information qui doit être transmise à la station de base.

Dans LEACH, le traitement est séparé dans des cycles de longueur constante, où chaque cycle commence par une phase d'initialisation suivie d'une phase de transmission. La durée d'un cycle est déterminée. Dans la première phase, les groupes sont organisés et les CHs sont sélectionnés. Cette élection est basée sur le pourcentage désiré de CHs et le nombre d'itérations au cours duquel un nœud a pris le rôle de CH. Ainsi, un nœud n prend une valeur aléatoire entre 0 et 1. Si cette valeur est inférieure au seuil $T(n)$, le nœud se déclare CH.

$$T(n) = \begin{cases} \frac{P}{1 - P * (r \bmod \frac{1}{P})} & \text{Si } n \in G \\ 0 & \text{Si non} \end{cases} \quad (2.1)$$

Avec :

- ✦ **P** : pourcentage désiré de CHs.
- ✦ **r** : itération actuelle.
- ✦ **G** : ensemble des nœuds qui ont été sélectionnés comme CH durant les dernières $(1/P)$ itérations.

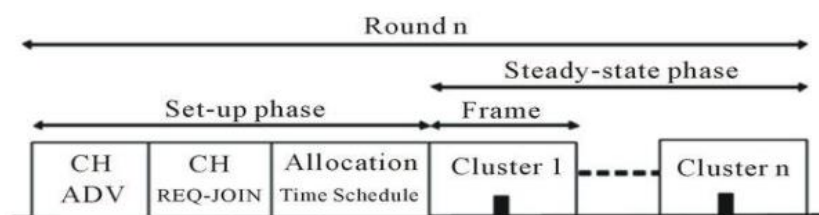


Figure 2.4 : Formation des clusters dans LEACH.

Chaque CH élu émet un message de signalisation au reste des nœuds dans le réseau, et qui présentent les nouveaux leaders. Tous les nœuds non leaders, et après avoir reçu ce message, décident de la groupe à laquelle ils veulent appartenir. Cette décision est basée sur la force du signal du message. Les nœuds non leader informent les leaders appropriés qu'ils seront un membre de la groupe [11].

II.4.2 Le protocole PEGASIS

Power-Efficient Gathering in Sensor Information Systems (PEGASIS) [8] est une version améliorée du protocole LEACH. PEGASIS forme des chaînes plutôt que des clusters de nœuds de capteurs afin que chaque nœud transmette et reçoive uniquement des données d'un voisin. Un seul nœud est sélectionné à partir de cette chaîne pour transmettre à la station de base.

L'idée de PEGASIS est qu'il utilise tous les nœuds pour transmettre ou recevoir des données avec ses plus proches voisins. Il déplace les données reçues de nœud à nœud, puis les données seront agrégées jusqu'à ce qu'elles atteignent tous la station de base. Donc, chaque nœud du réseau est tour à tour un chef de file de la chaîne, ainsi que responsable pour transmettre l'ensemble des données recueillies et fusionnées par la chaîne de nœuds au niveau de la station de base [9].

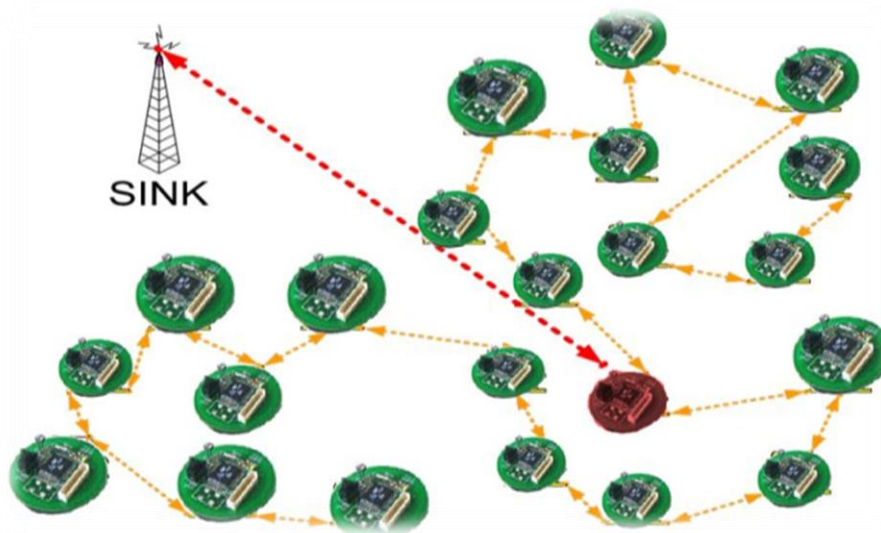


Figure 2.5 : Formation des chaînes dans PEGASIS

II.4.3 TEEN et APTEEN (Adaptive Threshold-sensitive Energy Efficient sensor Network protocol)

Les protocoles Threshold sensitive Energy Efficient sensor Network protocol (TEEN) [7] et Adaptive Threshold sensitive Energy Efficient sensor Network protocol (APTEEN) conviennent pour les applications critiques. Dans les deux protocoles, le facteur clé est la valeur de l'attribut mesuré. La caractéristique supplémentaire d'APTEEN est la capacité de changer la périodicité et les paramètres de TEEN en fonction des besoins des utilisateurs et des applications.

TEEN est conçu pour être sensible à des changements soudains des attributs tels que la température. La réactivité est importante pour les applications critiques dont le réseau fonctionne dans un mode réactif. L'architecture du réseau de capteurs est basée sur un groupement hiérarchique où les nœuds forment des clusters et ce processus va se répéter jusqu'à ce que la station de base soit atteinte [8].

APTEEN est une extension de TEEN qui fait à la fois la collection des captures périodique de données et qui réagit aux événements critiques. Quand la station de base forme des clusters, les clusters head diffusent les attributs, les valeurs des seuils, ainsi que le calendrier de transmission à tous les nœuds. Le cluster-head effectue également l'agrégation de données afin d'économiser l'énergie.

II.4.4 CTLMN (Clustering Technique for Large multi hop Mobile wireless Networks)

Lin et Chu [16] proposent une technique pour un large réseau Ad hoc. La structure de cluster est contrôlée par la distance égale au nombre de sauts. Cette technique repose sur la manière dont les nœuds sont regroupés dans un cluster en utilisant le nombre maximum de sauts R qui indique le rayon du cluster.

II.4.5 HEED (Hybrid, Eenergy-Efficient, Distributed approach)

Proposé par O. YOUNIS et autres en 2004 [15] ; HEED s'exécute d'une manière distribuée comme LEACH à la différence du mode d'élection des CHs où il utilise une formule qui fait intervenir l'énergie et le coût de la communication. Ainsi, seuls les nœuds ayant une énergie résiduelle plus importante peuvent devenir des CHs. HEED est caractérisé par:

- ✍ Une meilleure distribution des CHs sur le réseau, car la probabilité que deux nœuds adjacents deviennent CH en même temps, est minimale à la différence de LEACH;
- ✍ La consommation d'énergie est différente d'un nœud à un autre;
- ✍ La probabilité d'élire un CH peut être ajustée pour assurer une connectivité inter-CH avec un rayon de transmission prédéfini.

L'algorithme HEED s'exécute en trois phases :

- **Initialisation** : l'algorithme fixe un seuil C_{prob} pour limiter le nombre initial des CHs car chaque nœud du réseau calcule une probabilité CH_{prob} pour devenir CH par la formule suivante:

$$\mathbf{CH_{prob}} = \mathbf{C_{prob}} * (\mathbf{E_{R\u00e9siduel}} / \mathbf{E_{Max}}) \quad (2.2)$$

Avec :

- E_{Max} : Est l'nergie initiale du n\u00e9ud ;
- $E_{R\u00e9siduel}$: Est l'nergie restante.
- CH_{prob} : Est toujours sup\u00e9rieure \u00e0 un seuil P_{min} inversement proportionnel \u00e0 E_{Max} .

- **Phase r\u00e9p\u00e9titive** : chaque n\u00e9ud du r\u00e9seau et apr\u00e8s plusieurs it\u00e9rations choisi son CH qui minimise le co\u00fbt de transmission. Si aucun CH n'est entendu, il se d\u00e9clare CH et annonce sa d\u00e9cision ; et pour que l'algorithme converge le n\u00e9ud double son CH_{prob} \u00e0 chaque nouvelle it\u00e9ration jusqu'elle devienne \u00e9gale \u00e0 1. De cette fa\u00e7on, le n\u00e9ud est soit dans un \u00e9tat « Tentative status » si $CH_{prob} < 1$, soit il est dans un \u00e9tat « Final status » si $CH_{prob} = 1$.

- **Phase finale** : chaque n\u00e9ud d\u00e9cide d\u00e9finitivement de son statut; il est soit membre du cluster avec un co\u00fbt minimal, soit il est CH. Le protocole HEED, en consid\u00e9rant l'nergie comme facteur pour l'\u00e9lection des CHs, vise \u00e0 prolonger la dur\u00e9e de vie du r\u00e9seau ; quoique, la formation des clusters \u00e0 un saut risque de g\u00e9n\u00e9rer un nombre important de clusters si le nombre de n\u00e9uds est important. Ainsi, on se rapproche d'une topologie plate de r\u00e9seau. En plus, HEED pr\u00e9voit lui aussi, une communication directe entre les CHs et la station de base ce qui oblige les CHs lointains \u00e0 d\u00e9penser plus d'nergie et par cons\u00e9quence HEED est consid\u00e9r\u00e9 comme un protocole non favorable au passage \u00e0 l'\u00e9chelle.

Pour permettre une bonne mise \u00e0 l'\u00e9chelle des r\u00e9seaux hi\u00e9rarchiques, la communication en multi-sauts est employ\u00e9e. Ce type de communication permet de r\u00e9duire la consommation de l'nergie en utilisant des transmissions radio pour des distances r\u00e9duites au lieu des transmissions \u00e0 grande port\u00e9e, qui demande plus d'nergie pour \u00e9viter l'affaiblissement du signal caus\u00e9 par l'environnement entourant les n\u00e9uds du r\u00e9seau.

En plus, la communication multi-sauts garantit une meilleure qualit\u00e9 de r\u00e9ception de donn\u00e9es en minimisant les collisions gr\u00e2ce \u00e0 la petite port\u00e9e du signal qui n'affecte pas les transmissions voisines[12].

II.5 Protocoles de routage non hi\u00e9rarchiques

II.5.1 DSDV (Destination Sequenced Distance Vector)

DSDV est un protocole proactif de routage à vecteur de distance. Chaque nœud du réseau maintient une table de routage contenant le saut suivant et le nombre de sauts pour toutes les destinations possibles. Des diffusions de mises à jour périodiques tendent à maintenir la table de routage complètement actualisée à tout moment [12].

Afin d'éviter le bouclage (loop-freedom), DSDV utilise les numéros de séquence (Sequence Number) pour indiquer la « nouveauté » d'une route. Une route R est considérée plus favorable qu'une autre R', si R a un numéro de séquence plus grand ; si ces deux routes ont le même numéro de séquence, alors R est plus favorable s'il possède un nombre inférieur de sauts. Le numéro de séquence pour une route est initialisé par le nœud émetteur et incrémenté pour chaque nouvel avertissement de route. Quand un nœud détecte un lien brisé vers une destination D, il met à jour le nombre de sauts pour l'entrée de la destination D dans sa table avec la valeur infini et incrémente son numéro de séquence.

Les boucles de routes peuvent survenir lorsque des informations incorrectes de routage sont présentes dans le réseau après un changement dans la topologie du réseau (lien brisé par exemple). Dans ce contexte, l'utilisation des numéros de séquence adapte DSDV à une topologie dynamique de réseau comme dans un réseau Ad-hoc. DSDV utilise des mises à jour étiquetées lorsque la topologie change. La transmission des mises à jour est retardée afin d'introduire un effet d'amortissement quand la topologie change rapidement [12].

II.5.2 .GSR (Global State Routing)

Le protocole GSR est un protocole similaire au protocole DSDV décrit précédemment. Ce protocole utilise les idées du routage basé sur l'état des liens (Link State, LS), et les améliore en évitant le mécanisme inefficace d'inondation des messages de routage. GSR utilise une vue globale de la topologie du réseau, comme c'est le cas dans les protocoles basés sur LS. Le protocole utilise aussi une méthode, appelée la méthode de dissémination, utilisée dans le DBF (Distributed Bellman-Ford) [12].

De la même manière que les protocoles LS, les messages de routage sont générés suivant les changements d'états des liens. Lors de la réception d'un message de routage, le nœud met à jour sa table de topologie (représentée par un graphe dans LS) et cela dans le cas où le numéro de séquence du message reçu est supérieur à la valeur du numéro de séquence sauvegardé dans la table (exactement comme le fait le protocole DSDV). Par la

suite, le nœud reconstruit sa table de routage et diffuse les mises à jour à ses voisins. Le calcul des chemins peut se faire avec n'importe quel algorithme de recherche des plus courts chemins. Par exemple, l'algorithme du GSR utilise l'algorithme de Dijkstra modifié de telle façon qu'il puisse construire la table des nœuds suivants (NEXT HOP) et la table de distance Table_D, en parallèle avec la construction de l'arbre des plus courts chemins (l'arbre dont la racine est le nœud source) [12].

II.5.3 FSR (Fisheye State Routing)

Le protocole FSR (Fisheye State Routing) est basé sur l'utilisation de la technique "œil de poisson" (fisheye) et utilisé dans le but de réduire le volume d'information nécessaire pour représenter les données graphiques. Dans la pratique, l'œil d'un poisson capture avec précision, les points proches du point focal. La précision diminue quand la distance, séparant le point vu et le point focal augmente.

Dans le contexte du routage, l'approche du "fisheye" matérialise, pour un nœud, le maintien des données concernant la précision de la distance et la qualité du chemin d'un voisin direct, avec une diminution progressive du détail et de la précision, quand la distance augmente.

La diminution de la précision est assurée en changeant les fréquences de mise à jour, et cela en utilisant des périodes d'échanges différentes pour les différentes entrées de la table de routage. Les entrées qui correspondent aux nœuds les plus proches sont envoyées aux voisins avec une fréquence élevée et donc avec une période d'échange relativement petite [10].

Afin de réduire le volume de messages échangés sans toucher à la consistance et à la précision des données de routage, FSR utilise la technique œil de poisson.

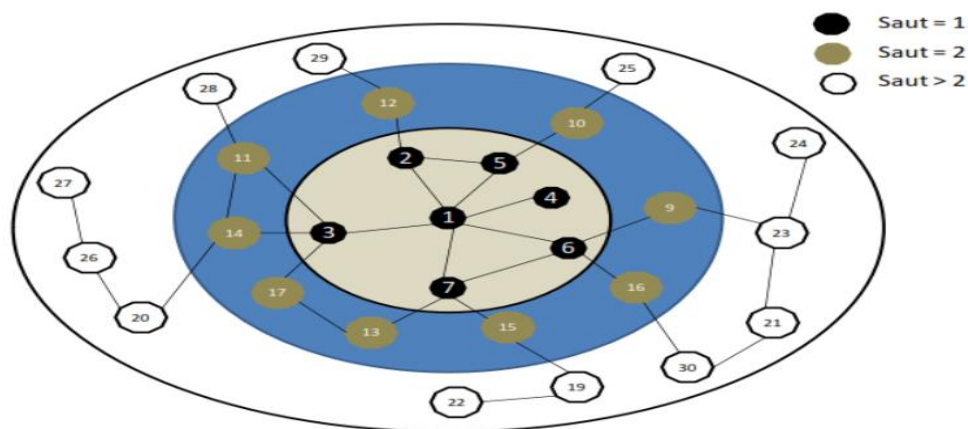


Figure 2.6 : Technique "œil de poisson" dans le protocole FSR.

La Figure 2.6 illustre cette technique. Dans cette figure, on définit la portée, ou le champ de vision de l'œil de poisson, avec un nœud du centre d'identificateur égal à 1. La portée est définie en termes de nœuds qui peuvent être atteints en passant par un certain nombre de sauts. Dans la Figure 2.6, trois portées sont montrées pour 1, 2 et supérieures à 2 respectivement. En conséquence, les nœuds sont colorés en noir, gris et blanc. Le nombre de niveaux et le rayon de chaque portée va dépendre de la taille du réseau. Le nœud de centre (le nœud 1) maintient les données les plus précises des nœuds appartenant au cercle le plus proche ; la précision diminue progressivement, pour les cercles moins proches du centre.

II.5.4 AODV (Ad-hoc On Demand Distance Vector)

C'est un protocole à vecteur de distance, comme DSDV, mais il est réactif plutôt que proactif comme DSDV. En effet, AODV ne demande une route que lorsqu'il en a besoin. AODV utilise les numéros de séquence d'une façon similaire à DSDV pour éviter les boucles de routage et pour indiquer la « nouveauté » des routes. Une entrée de la table de routage contient essentiellement l'adresse de la destination, l'adresse du nœud suivant, la distance en nombre de sauts (i.e. le nombre des nœuds nécessaires pour atteindre la destination), le numéro de séquence destination, le temps d'expiration de chaque entrée dans la table [13].

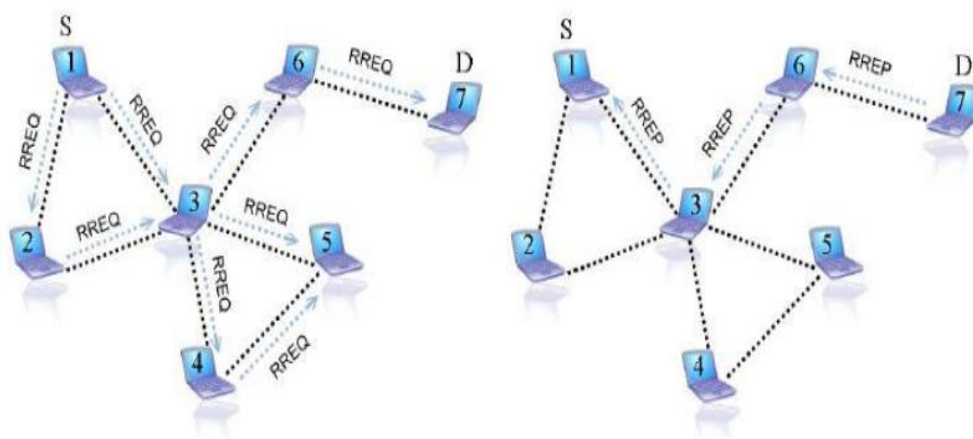


Figure 2.7 : Fonctionnement de la procédure de demande de route dans AODV.

La Figure 2.7 illustre le mécanisme de création des routes d'AODV. Il y a d'abord diffusion de la demande de route. Ensuite, la destination envoie une réponse, qui, grâce

aux informations recueillies par les nœuds lors de la diffusion de la demande de route, crée une route de la destination vers la source.

A noter que le protocole de routage AODV n'assure pas la détection du meilleur chemin existant entre la source et la destination. Les nœuds voisins sont détectés par des messages périodiques HELLO (un message particulier de RREP). Si un nœud x ne reçoit pas un message HELLO d'un voisin y par lequel il envoie des données, ce lien est considéré brisé et une indication de défaillance de lien est envoyée à ses voisins actifs. Ces derniers propagent l'indication à leurs voisins qui utilisaient le lien entre x et y . Lorsque le message du lien brisé atteint finalement les sources affectées, celles-ci peuvent choisir d'arrêter l'envoi des données ou de demander une nouvelle route en envoyant un nouveau message RREQ.

II.5.5 DSR (Dynamic Source Routing)

Le protocole "Routage à Source Dynamique" (DSR) [6] est basé sur l'utilisation de la technique "routage source". Dans cette technique, la source des données détermine la séquence complète des nœuds à travers lesquelles les paquets de données seront envoyés.

Un site initiateur de l'opération de « découverte de routes » diffuse un paquet requête de route. Si l'opération de découverte est réussie, l'initiateur reçoit un paquet réponse de route qui liste la séquence de nœuds à travers lesquelles la destination peut être atteinte. Le paquet requête de route contient donc un champ enregistrement de route dans lequel sera accumulée la séquence des nœuds visités durant la propagation de la requête dans le réseau.

L'utilisation de la technique "routage source" fait que les nœuds de transit n'aient pas besoin de maintenir les informations de mise à jour pour envoyer les paquets de données, puisque ces derniers contiennent toutes les décisions de routage.

Dans ce protocole, il y a une absence totale de boucle de routage car le chemin source destination fait partie des paquets de données envoyés.

II.5.6 SPIN (Sensor Protocols for Information via Negotiation)

L'idée derrière le SPIN est d'échanger des informations sur les données à envoyer en utilisant des paquets de signalisations spéciales nommées meta-DATA. Ceci permet d'éviter le problème des données redondantes. Chaque nœud, s'intéressant à la donnée référencée par ce paquet meta-DATA, peut les récupérer en envoyant un paquet de requête [14].

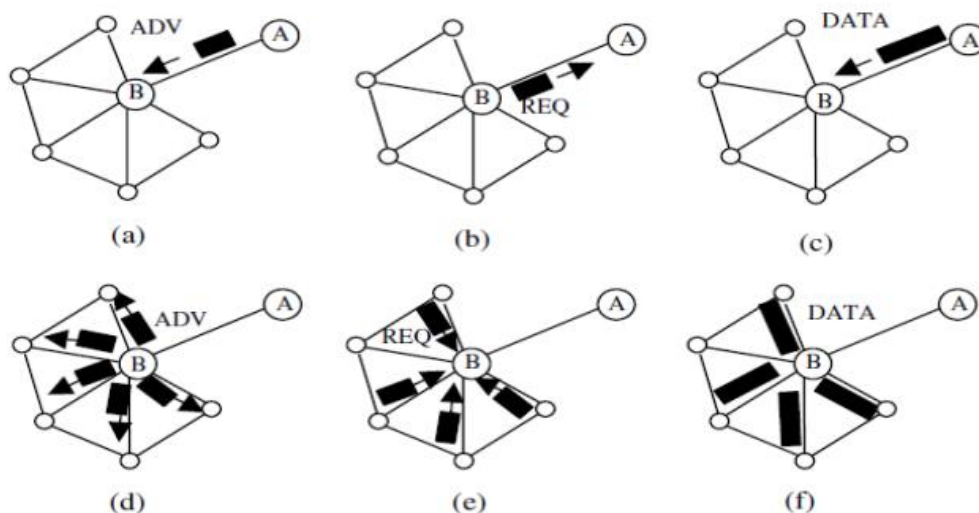


Figure 2.8 : Fonctionnement du protocole SPIN.

Le protocole SPIN utilise essentiellement trois types de paquets ADV/REQ/DATA. Un nœud voulant émettre une donnée commence par envoyer un paquet ADV. Ce paquet ADV consiste d'une méta-donnée sur les données à émettre. Les méta-données peuvent décrire plusieurs aspects comme le type des données et la localisation de son origine. Les nœuds qui reçoivent ce paquet vérifient si les données les intéressent. Si oui, ils répondent par un paquet REQ. Le nœud qui a initié la communication envoie alors un paquet DATA pour chaque réponse REQ reçue. Un nœud peut parfaitement ne pas répondre aux messages ADV, par exemple dans le but d'économiser son énergie. Ensuite chaque nœud qui fait office de relais peut très bien agréger ses propres données aux données qui sont déjà contenues dans le paquet.

II.5.7 OLSR (optimized link State ROUTING)

Comme son nom l'indique, OLSR est un protocole proactif à état des liens optimisé ; il permet d'obtenir aussi des routes de plus court chemin. Alors que dans un protocole à état des liens, chaque nœud déclare ses liens directs avec ses voisins à tout le réseau, dans le cas d'OLSR, les nœuds ne déclarent qu'une sous-partie de voisinage grâce à la technique des relais multipoints (MultiPoint Relaying, MPR) [14].

II.5.8 GPSR (Greedy perimeter Stateless Routing)

La topologie a un caractère relativement provisoire dû à la mobilité des nœuds dans les réseaux Ad-hoc et de capteurs mobiles. Pour cette raison, les protocoles de

routing les plus étudiés pour ce type de réseaux sont les protocoles de routing géographique car ils permettent d'éviter la surcharge d'informations échangées entre les nœuds qui cherchent à obtenir la topologie du réseau ou à construire les tables de routing.

Ce protocole de routing géographique se base sur le fait que tous les nœuds connaissent leur position, par exemple, grâce à un équipement GPS (Global Positioning System) ou encore par un système de positionnement distribué [12].

II.6 Conclusion

La technologie des réseaux de capteurs sans fil reste très prometteuse, et leur défi majeur est de trouver des protocoles de routing qui permettent, à la fois, de :

- Consommer le moins d'énergie possible,
- Assurer la connectivité du réseau et la couverture du champ surveillé,
- Assurer une livraison fiable et rapide,
- Tolérer aux pannes,
- S'adapter aux changements de topologie ...

Afin de concevoir un protocole de routing qui prend en compte toutes les contraintes citées au-dessus, des méthodes et algorithmes bio-inspirés ont été proposés dans la littérature. Parmi ces algorithmes on trouve l'Optimisation par les Essaims des Particules (PSO). Cette dernière sera l'objet du chapitre suivant.



CHAPITRE 03
OPTIMISATION PAR
ESSAIMES DES PARTICULES



III.1.Introduction

A l'ère où les logiciels les plus rapides du marché ne peuvent plus résoudre certains problèmes (complexité), l'homme s'inspire de plus en plus de la nature qui l'entoure pour mettre en place des algorithmes simulant le comportement des animaux, leurs actions ou réactions vis-à-vis d'un problème et même les méthodes qu'ils utilisent pour y faire face .

Même si ces méthodes sont sujettes à de nombreuses controverses, ils permettent de trouver facilement et rapidement la solution la plus proche de l'optimale si ce dernier existe, et reste un moyen très efficace pour le traitement de problèmes complexes pouvant prendre des années de calculs sans résultats.

Dans ce sujet, Dans ce chapitre nous allons présenter ce méta heuristique bio-inspirée, celle-ci est dite « optimisation par essaims de particule » dont l'idée directrice est la simulation du comportement collectif des oiseaux à l'intérieur d'une nuée.

Pour cela, nous allons commencer par introduire le côté biologique de l'algorithme. Nous allons par la suite décrire le mécanisme de l'algorithme qui constitue d'un point de vue informatique une source d'inspiration assez riche.

III.2 Optimisation Par Essaims de Particule :

L'optimisation par Essaim de particule ou bien (PSO Particle swarm optimization) est une technique utilisée pour explorer l'espace de recherche d'un problème quelconque pour trouver l'ensemble des paramètres qui maximise/minimise un objectif particulier. Cet objectif est atteint en suivant un algorithme dédié que l'on verra par la suite [17].

III.3 UN PEU D'HISTOIRE

L'optimisation par Essaim de particule ou bien (PSO Particle swarm optimization), a été inventée par Russel Eberhart (ingénieur en électricité) et James Kennedy (socio-psychologue) en 1995. Au départ J. Kennedy et R. Eberhart cherchaient à simuler la capacité des oiseaux à voler de façon synchrone et leur aptitude à changer brusquement de direction tout en restant en une formation optimale. Le modèle qu'ils ont proposé a ensuite été étendu en un algorithme simple et efficace d'optimisation [19].

III.4 Topologie du voisinage :

La topologie du voisinage défini avec qui chacune des particules va pouvoir communiquer. Il existe de nombreuses combinaisons.

Le voisinage constitue la structure du réseau social. En général, pour une nuée d'oiseaux, le voisinage suit trois types de topologies [19] :

- Topologie en étoile (Figure1 (a)) : le réseau social est complet, donc une communication complète et une attirance vers la meilleure particule chaque particule est reliée à toutes les autres, i.e l'optimum du voisinage est l'optimum global.
- Topologie en anneau (Figure1 (b)) : chaque particule communique avec n voisines (en général, $n = 3$), immédiates et tend à se déplacer vers la meilleure dans son voisinage local, c'est la topologie la plus utilisée.
- Topologie en rayon (Figure1 (c)) : une particule "centrale" est connectée à toutes les autres. Seule cette particule centrale ajuste sa position vers la meilleure, si cela provoque une amélioration l'information est propagée aux autres.

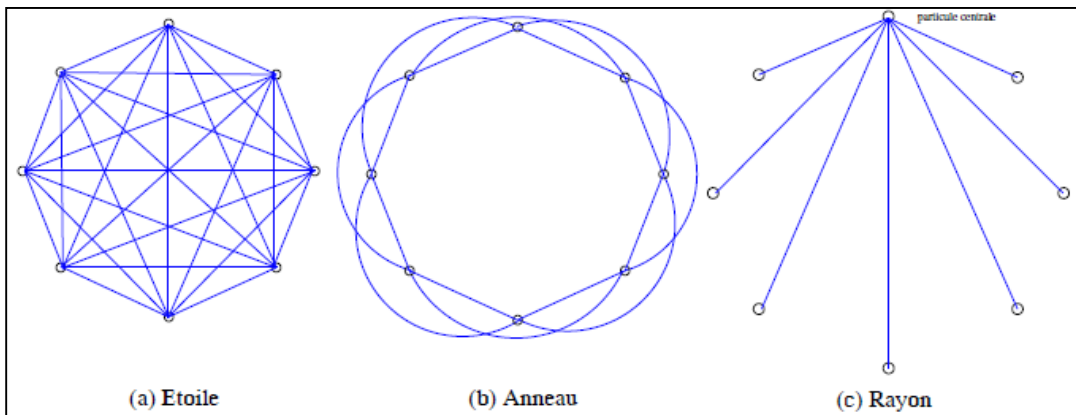


Figure 3. 1 : différents types de topologie pour un essaim de particules.

III. 5 L'ALGORITHME DE PSO

Chaque particule représente une solution potentielle dans l'espace de recherche. La nouvelle position d'une particule est déterminée en fonction de sa propre valeur et celle de ses voisines. Soit $\vec{x}_i(t)$ la position de la particule P_i au temps t , sa position est modifiée en ajoutant une vitesse $\vec{v}_i(t)$ à sa position courante [20] :

$$\vec{x}_i(t) = \vec{x}_i(t - 1) + \vec{v}_i(t) \quad (3.1)$$

La vitesse de chaque particule est mise à jour suivant l'équation suivante :

$$v_i(t + 1) = \omega v_i(t) + c_1 r_1 [x_{pi}(t) - x_i(t)] + c_2 r_2 [g(t) - x_i(t)] \quad (3.2)$$

$v_i(t)$ est la vitesse de particule i à l'instant t et $x_i(t)$ est la position de particule i à l'instant t , les paramètres w , c_1 , et c_2 ($0 \leq w \leq 1.2$, $0 \leq c_1 \leq 2$, et $0 \leq c_2 \leq 2$) sont des coefficients constants fixés par l'utilisateur, r_1 et r_2 sont des nombres aléatoires tirés à chaque itération, $g(t)$ est la meilleure solution trouvée jusqu'à l'instant t et $x_{pi}(t)$ est la meilleure solution trouvée par le particule i .

Remarque :

$\omega v_i(t)$ c'est le Terme d'inertie

$c_1 r_1 [x_{pi}(t) - x_i(t)]$ c'est le Composant cognitive

$c_2 r_2 [g(t) - x_i(t)]$ c'est le composant social

C'est le vecteur vitesse qui dirige le processus de recherche et reflète la "sociabilité" des particules.

Les variables et paramètres de l'algorithme :

N nombre de particules

$\vec{x}_i(t)$ Position de la particule P_i

\vec{v}_i vitesse de la particule P_i

$pbest_i$ Meilleure fitness obtenue pour la particule P_i

\vec{x}_{pbest_i} Position de la particule P_i pour la meilleure fitness

\vec{x}_{gbest_i} Position de la particule ayant la meilleure fitness de toutes

ρ_1, ρ_2 valeurs aléatoires positives

Initialisations :

Initialiser aléatoirement la population.

Traitement :

Répéter

Pour i de 1 à N faire

Si $(F(\vec{x}_i) > pbest_i)$ **Alors**

$pbest_i \leftarrow F(\vec{x}_i)$

$\vec{x}_{pbest_i} \leftarrow \vec{x}_i$

Fin Si

Si $(F(\vec{x}_i) > gbest_i)$ **Alors**

$gbest_i \leftarrow F(\vec{x}_i)$

$\vec{x}_{gbest_i} \leftarrow \vec{x}_i$

Fin Si

Fin Pour

Pour i de 1 à N faire

$\vec{v}_i \leftarrow \vec{v}_i + \rho_1(\vec{x}_{pbest_i} - \vec{x}_i) + \rho_2(\vec{x}_{gbest_i} - \vec{x}_i)$

$\vec{x}_i \leftarrow \vec{x}_i + \vec{v}_i$

Fin Pour

Jusqu'à ce que (le processus converge)

Fin.

La variation de la vitesse est proportionnelle à l'éloignement d'une solution par rapport à la solution globale.

Les variables aléatoires ρ_1 et ρ_2 peuvent être définies de la façon suivante :

$$\begin{cases} \rho_1 = r_1 c_1 \\ \rho_2 = r_2 c_2 \end{cases}$$

r_1 et r_2 suivent une loi uniforme sur $[0..1]$ et c_1 et c_2 sont constantes et représentent une accélération positive, avec $c_1 + c_2 < 4$.

Le critère de convergence peut être un nombre fixe d'itérations, suivant la fitness ou bien la variation lorsqu'elle tend vers 0[21].

On remarque qu'il y a Six paramètres qui rentrent en ligne de compte :

1. La dimension du problème.
2. Le nombre de particules.
3. Les valeurs des coefficients ρ .

4. La taille du voisinage.
5. La vitesse maximale.
6. L'inertie.

La vitesse peut être limitée par une vitesse maximale V_{max} et une vitesse minimale V_{min} pour éviter que les particules se déplacent trop rapidement ou trop lentement d'une région à une autre dans l'espace de recherche.

Un facteur d'inertie Φ peut être appliqué à la vitesse Pour contrôler l'influence de celle-ci.

III.5.1 Vitesse maximale et coefficient de constriction

Afin d'éviter que les particules ne se déplacent trop rapidement dans l'espace de recherche, passant éventuellement à côté de l'optimum, il peut être nécessaire de fixer une vitesse maximale (notée \vec{v}_{max}) pour améliorer la convergence de l'algorithme. Cependant, on peut s'en passer si on utilise un coefficient de constriction κ introduit par Maurice CLERC et qui permet de resserrer l'hyper-espace de recherche [18].

L'équation de la vitesse devient alors :

$$K=1-\frac{1}{\rho}+\frac{\sqrt{\rho^2-4\rho}}{2} \quad (3.3)$$

Avec $\rho = \rho_1 + \rho_2 > 4$.

III.5.2 Critères d'arrêt

La convergence vers la solution optimale globale n'est pas garantie dans tous les cas de même si les expériences dénotent la grande performance de la méthode. De ce fait, il est fortement conseillé de doter l'algorithme d'une porte de sortie en définissant un nombre maximum d'itération. L'algorithme doit alors s'exécuter tant que l'un des critères de convergence suivant n'a pas été atteint [22] :

- Nombre des itérations maximale a été atteint ;
- la variation de la vitesse est proche de 0 ;
- le fitness de la solution est suffisant.

III.9 Conclusion

Les méta-heuristiques constituent une stratégie de résolution de plus en plus privilégiée. Une des particularités importantes des méta-heuristiques, réside dans l'absence d'hypothèses particulière sur la régularité de la fonction objective. Aucune hypothèse sur la continuité de cette fonction n'est requise, ses dérivées successives ne sont pas nécessaires, ce qui rend très vaste le domaine d'application.

L'optimisation par essaims de particules présente un fort potentiel d'application pratique, mais le choix de paramètres reste l'un des problèmes de l'optimisation par particules d'essaim, c'est très difficile de trouver des bons paramètres adaptés à la structure du problème.

Les résultats obtenues par PSO sont très satisfaisant et confirment bien la validité de l'algorithme, sa simplicité d'implémentation lui donne un avantage conséquent, néanmoins par l'expérimentation, on a remarqué que la qualité des solutions ne peut être garantie même en augmentant le nombre d'itérations.



IMPLÉMENTATION ET DISCUSSION



IV.1 Introduction

Les capteurs sont conçus pour fonctionner durant des mois voire des années. Ainsi, la capacité énergétique des capteurs doit être utilisée efficacement afin de maximiser la durée de vie du réseau. A noter qu'une fois qu'un nœud capteur a épuisé son énergie, il est considéré comme défaillant. Ainsi, il y a une forte probabilité de perdre la connectivité du réseau.

Dans ce chapitre, Nous présenterons une nouvelle protocole qui est Adaptive Particle Swarm Optimization for clustering (APSOC) comme une solution proposées à la problématique de la consommation d'énergie, ainsi les problèmes concernent la distance de transmission dans les réseaux de capteurs sans fil par la stratégie hiérarchique qui réduirait aussi le nombre de transmission, et avec l'agrégation des données dans le RCSF par conséquent moins d'énergie consommé.

IV.2 Description détaillée de l'algorithme proposé

IV.2.1 Formation des CHs

Les CHs initial sont choisi aléatoirement ; mais Dans les autres ronds on déterminer la position des nouveaux CHs à l'aide de la méthode APSOC basant sur les ancien CHs selon leur Pbest , Gbest et la vitesse par l'équation [1] notant que chaque particule est considéré le rôle comme étant un CH :

$$\vec{x}_i(t) = \vec{x}_i(t - 1) + \vec{v}_i(t) \quad (4.1)$$

Avec :

$\vec{x}_i(t)$ Est la position de nouveau CH ;

$\vec{x}_i(t - 1)$ Est la position ancienne de CH ;

$\vec{v}_i(t)$ Est la vitesse

Initialement on associe à chaque nœud une vitesse initiale aléatoirement, mais elle sera mise à jour à chaque nouveau ronds par l'équation 2 :

$$v_i(t + 1) = \omega v_i(t) + c_1 r_1 [x_{pi}(t) - x_i(t)] + c_2 r_2 [g(t) - x_i(t)] \quad (4.2)$$

Tel que :

- $v_i(t + 1)$: La nouvelle vitesse ;
- $x_{pi}(t)$: La meilleure position de CHs obtenu par, l'énergie consommée par la diffusion de message d'annonce comme étant un CH et la réception de message d'abonnement des membres ;
- $g(t)$: Est la meilleur $x_{pi}(t)$ (CH qui a le moins énergie consommé) ;
- $x_i(t)$: Est la position de CH ;
- c_1, c_2, r_1, r_2 et ω sont fixé à 1.

Après la mise à jour de vitesse on obtient une nouvelle position qu'elle va nous aidera a trouvé les nouveaux CHs, dont on calcule la distance entre chaque nœuds et ce nouveaux positions les nœuds qui a la minimums distance sont les nouveaux CHs.

IV. 2.2 Algorithme de APSOC

Algorithme APSOC

$c1 = c2 = w = 1$

$r1 = r2 = [0..1]$

Debut

1. *Génération des CHs aléatoire*

Répéter jusqu'à ce qu'a (t+1 < max_generation)

Pour i de 1 à nbr des CHs Faire

2. *mise à jour de vitesse pour chaque CH* $v_i(t + 1) = \omega v_i(t) + c1r1[xp_i(t) - x_i(t)] + c2r2[g(t) - x_i(t)]$

3. *Calcule de nouvelle position*

$$\vec{x}_i(t) = \vec{x}_i(t - 1) + \vec{v}_i(t)$$

Pour j de 1 à nbr des Nœud Faire

4. *Chois de min distance(Nœud(j), $\vec{x}_i(t)$);*

Nœud(j) devenir CH;

j=j+1;

Fin pour

Fin pour

Calculer la fonction objective

Fin

Visualiser le résultat

IV 2.3 Fonction objective

L'objective de cette fonction est de minimisé l'énergie consommé par les CHs par apport à l'énergie totale de réseaux ainsi que la distance entre les CHs et la station de base suivant l'équation :

$$F(\text{config}) = \frac{\sum_{i=1}^{nbch} E(ch_i) - \text{TotaleEn}}{\sum_{i=1}^{nbch} (dis_{ch, sb})} \quad (4.3)$$

Avec :

Config : Est la configuration de réseaux a évaluées selon la fonction objective

$nbch$: Est le nombre des CHsE(ch) : Energie consommé lorsque un CHs envoie un message a station de base

TotaleEn: Totale énergie de réseaux

$dis_{ch, sb}$: Distance entre chaque CHs et la station de base

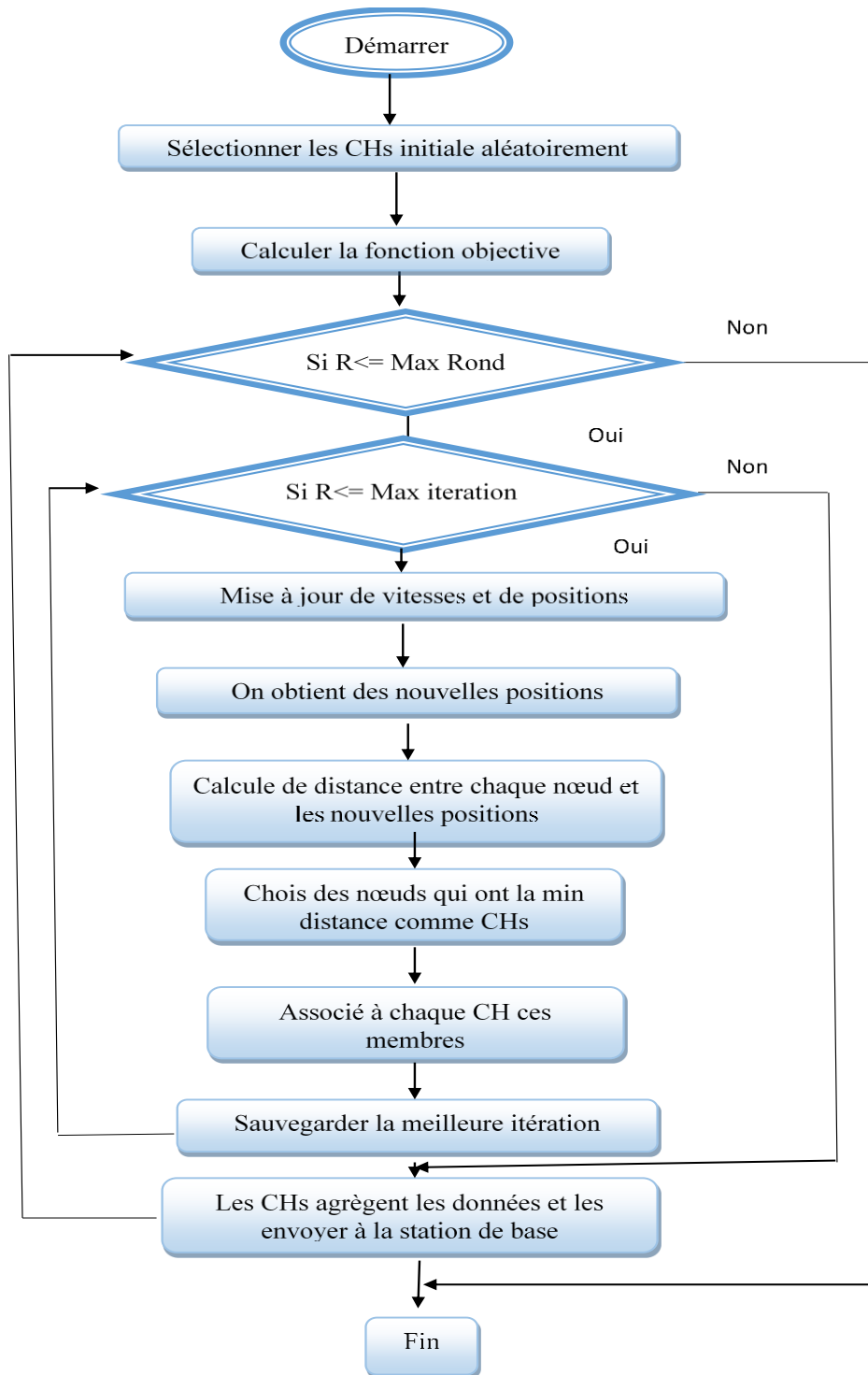


Figure 4.1 : L'organigramme de la méthode PSO

IV.3 A propos de l'environnement MATLAB

Nous avons choisi comme langage d'implémentation pour notre protocole, on a utilisé le MATLAB (MATrix LABoratory), qui est un langage de calcul scientifique de haut niveau et un environnement interactif pour le développement des applications, la visualisation et l'analyse de données, ou encore le calcul numérique.

Nous utilisons MATLAB®13.0 pour résoudre des problèmes de calcul scientifique plus rapidement. Le langage MATLAB met à la disposition du développeur les opérations vectorielles et matricielles, fondamentales pour les problèmes d'ingénierie et scientifiques. Il permet un développement et une exécution rapide à l'égard de langage MATLAB nous pouvons programmer et tester des algorithmes plus rapidement qu'avec les langages traditionnels, car il n'est pas nécessaire d'effectuer les tâches de programmation de bas niveau, comme la déclaration des variables, la spécification des types de données et l'allocation de la mémoire.

Les IHM (Interface Homme Machine), sont appelés GUI (Graphique User Interfaces) dans MATLAB. Elles permettent à des objets graphique (boutons, menus, cases à cocher, ...) d'interagir avec un programme informatique [23].

L'environnement de MATLAB possède 4 fenêtres : a. Au centre l'invite de commande (command window). b. En haut à droite le contenu de l'espace courant de travail (workspace). c. A gauche la liste des fichiers du répertoire courant (current folder). d. En bas à droite l'historique des commandes tapées (command history) [24].

IV.4 Les étapes d'exécution de l'application

IV.4.1 Description et paramètres de simulation

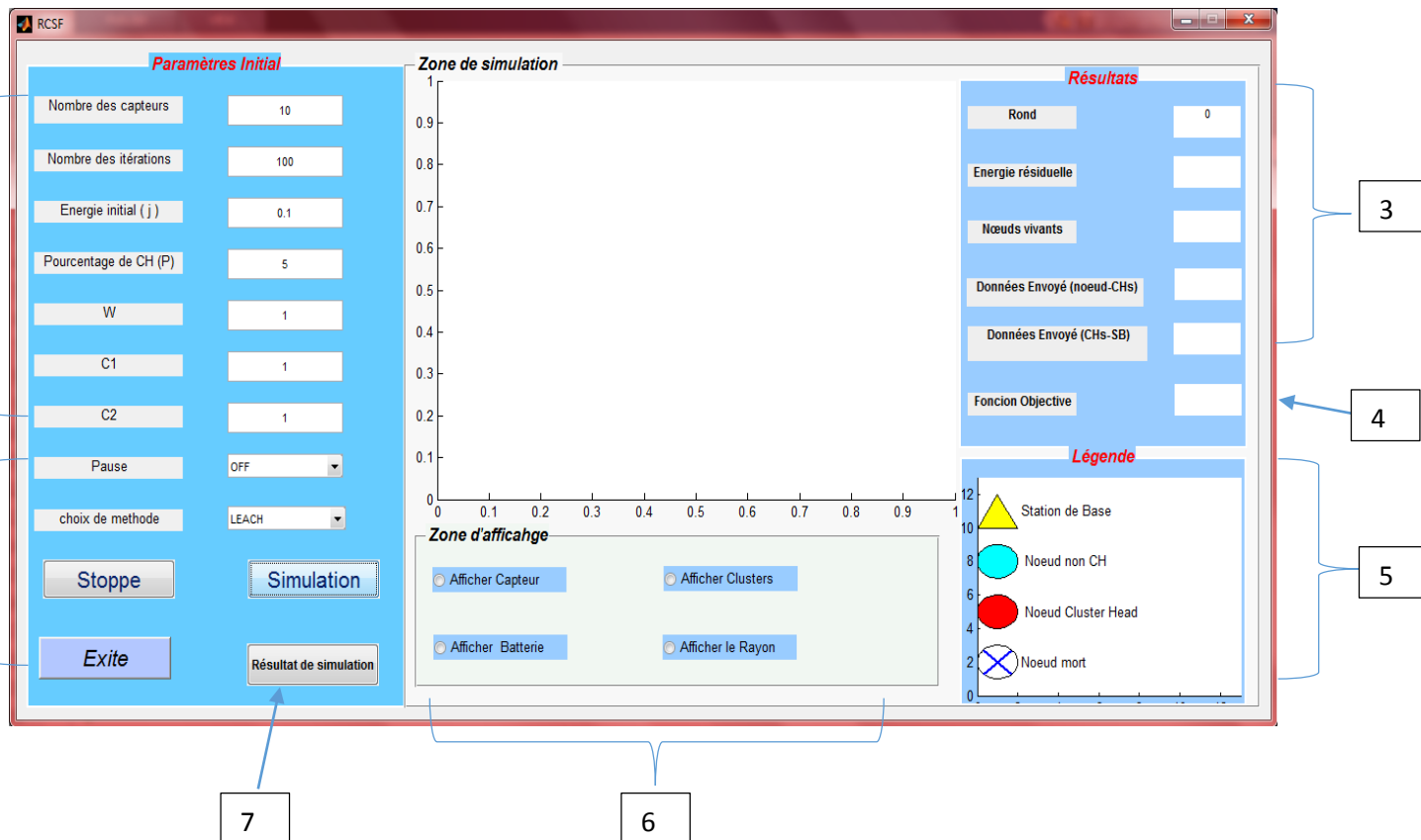


Figure 4.2 : L'interface principale de simulation.

1 : zone des paramètres initial : détermine le paramètre initiale et les paramètres de APSOC ;

2 : zone de contrôle de simulation :

- Choix de méthode : Pour choisir le protocole de simulation (Leach ou APSOC) ;
- Stoppe : pour arrêter la simulation à n'importe quel instant ;
 - ☞ Si pause est en état ON : la simulation sera continuer ;
 - ☞ Si pause est en état OFF : La simulation sera redémarrer de début ;
- Bouton de simulation : la simulation de protocole choisie précédemment ;
- Exit : Pour sortir de l'application.

3 : zone de résultat : le nombre des donnée transmettre dans le réseau, l'énergie résiduelle et le nombre des nœuds vivant ;

4 : Résultat de la fonction objective ;

5 : légende.

6 : zone d'affichage : l'affichage des caractéristiques des nœuds ;

7 : Bouton pour l'affichage des résultats.

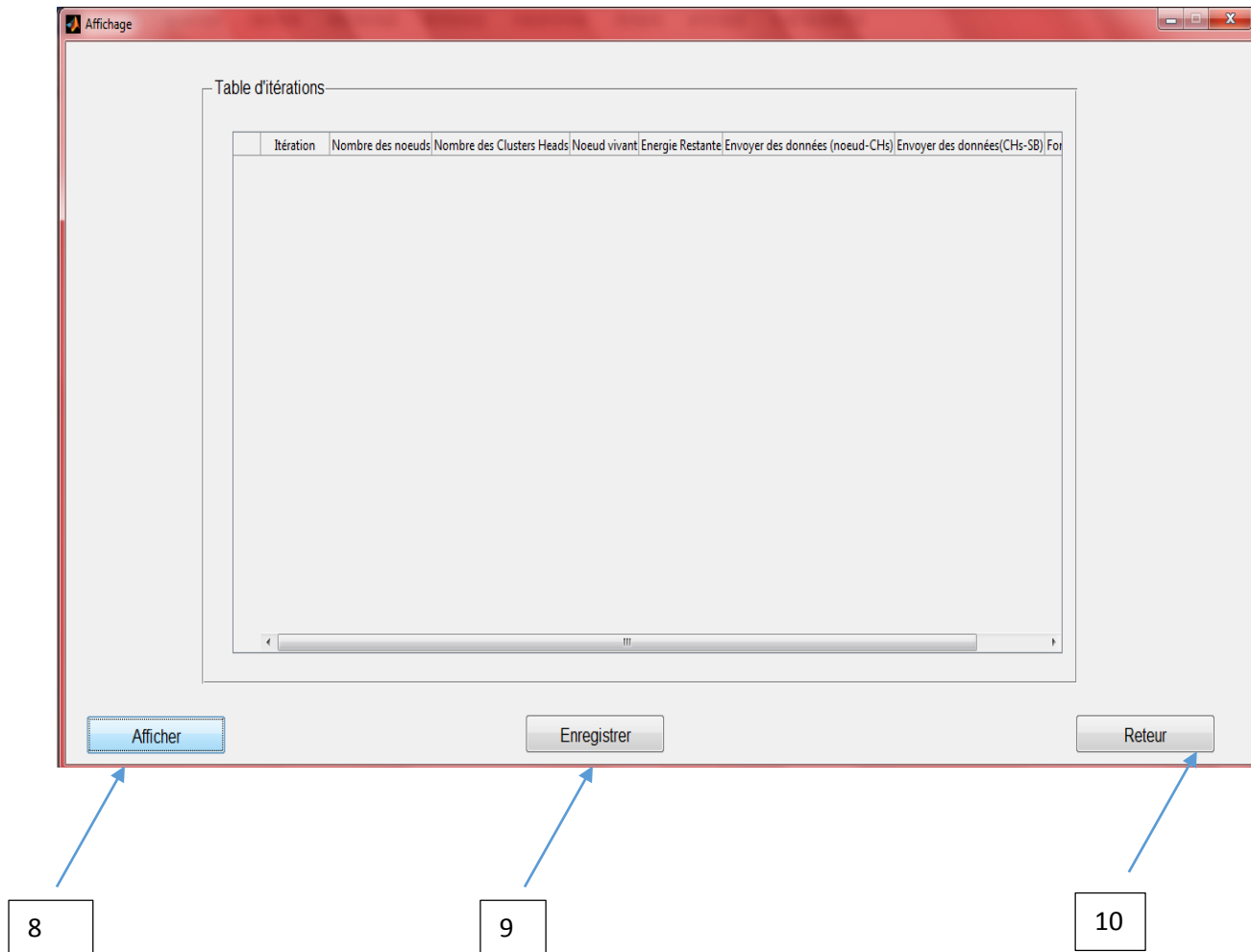


Figure IV.3 : L'interface d'affichage.

8 : Pour afficher le résultat de simulation ;

9 : Pour enregistrer le résultat dans un fichier Excel ;

10 : Pour retourner à la figure de simulation.

IV.4.2 Paramètres de simulation

✍ Energie de capture

Un capteur utilise son énergie pour réaliser trois actions principales : l'acquisition, la communication et le traitement des données.

- **Acquisition** : L'énergie consommée pour effectuer l'acquisition n'est pas très importante. Néanmoins, elle varie en fonction du phénomène et du type de surveillance effectué.
- **Communication** : Les communications consomment beaucoup plus d'énergie que les autres tâches. Elles couvrent les communications en émission et en réception. La figure 1 présente un modèle d'antenne et les règles de consommation d'énergie associées [24].

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

$$E_{Tx}(k, d) = E_{Tx}(l) + E_{Tx_amp}(k, d) \quad (4.4)$$

$$E_{Tx}(k, d) = \begin{cases} k \cdot E_{elec}(k, d) + k \cdot \mathcal{E}_{friss} \cdot d^2 \\ k \cdot E_{elec}(k, d) + k \cdot \mathcal{E}_{two_ray_amp} \cdot d^4 \end{cases} \quad (4.5)$$

Pour recevoir un message de k bits, le récepteur consomme :

$$E_{Rx}(k) = E_{Rx_elec}(k) = k \cdot E_{elec} \quad (4.6)$$

Avec :

E_{elec} : Energie de transmission/réception électronique ;

k : Taille d'un message ;

d : Distance entre l'émetteur et le récepteur ;

E_{TX-amp} : Energie d'amplification ;

\mathcal{E}_{amp} : Facteur d'amplification ;

$d_{crossover}$: Distance limite pour laquelle les facteurs de transmission changent de valeur.

- **Traitement des données** : L'énergie consommée pour les opérations de calculs est beaucoup plus faible que l'énergie de communication. Un exemple est cité dans. L'énergie nécessaire pour transmettre 1 KB sur une distance de 100m est approximativement équivalente à l'énergie nécessaire pour exécuter 3 millions d'instructions avec une vitesse de 100 millions d'instructions par seconde (MIPS). Ce niveau peut être dépassé en fonction des circuits installés dans les nœuds et des fonctionnalités requises [24].

Par la suite, elle sera calculée en appliquant la formule suivante :

$$E_{DA} = 5 \text{ nJ/bit/signal}$$

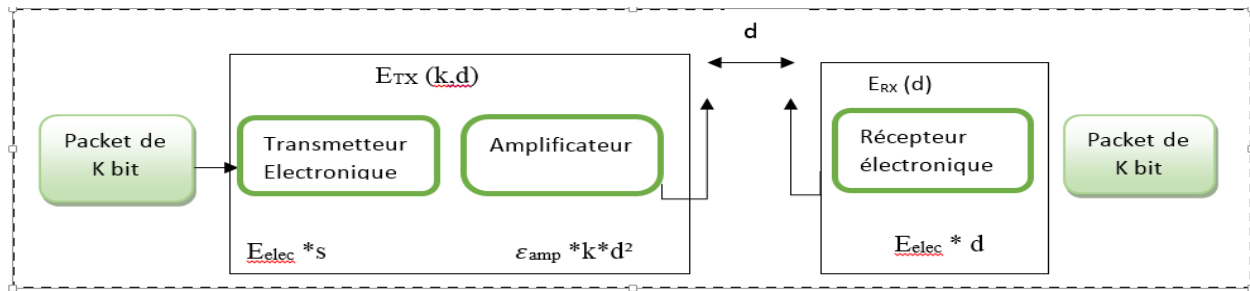


Figure IV.4 : Modèle de consommation d'énergie pour

IV.5 Simulation et résultat

Les résultats de simulation que nous avons obtenus et présenté dans cette partie, ces résultats seront comparés avec celles résultats de protocole LEACH a fin de validé la performance de notre protocole.

L'évaluation de notre protocole ce fait en terme d'économie d'énergie et de prolongement de la durée de vie global du réseau et selon le nombre total des nœuds morts dont on analysant les résultats fournis.

Nos paramètres de simulation sont les mêmes que dans et nous assumons que tous les nœuds ont une position fixe durant toute la période de simulation et la station de base est positionnée an centre de la zone du captage.

Zone de couverture	100*100 m²
Nombre de nœuds	100
L'énergie initiale des noeuds	0.1 J
Langur de paquet	6400
Eelec	50 nJ/bit
Efs	10 pJ/bit/m ²
Emp	0.0013 pJ/bit/m ²
Eda	5 pJ/bit/sig
Type de distribution	aléatoire

Tableau IV.1 : Les paramètres de simulation.

C1 et C2	Coefficients d'accélération (1)
w	1
r1 et r2	Prend une valeur [0..1] à chaque itération

Tableau IV.2 : Les paramètres de l'algorithme APSOC.

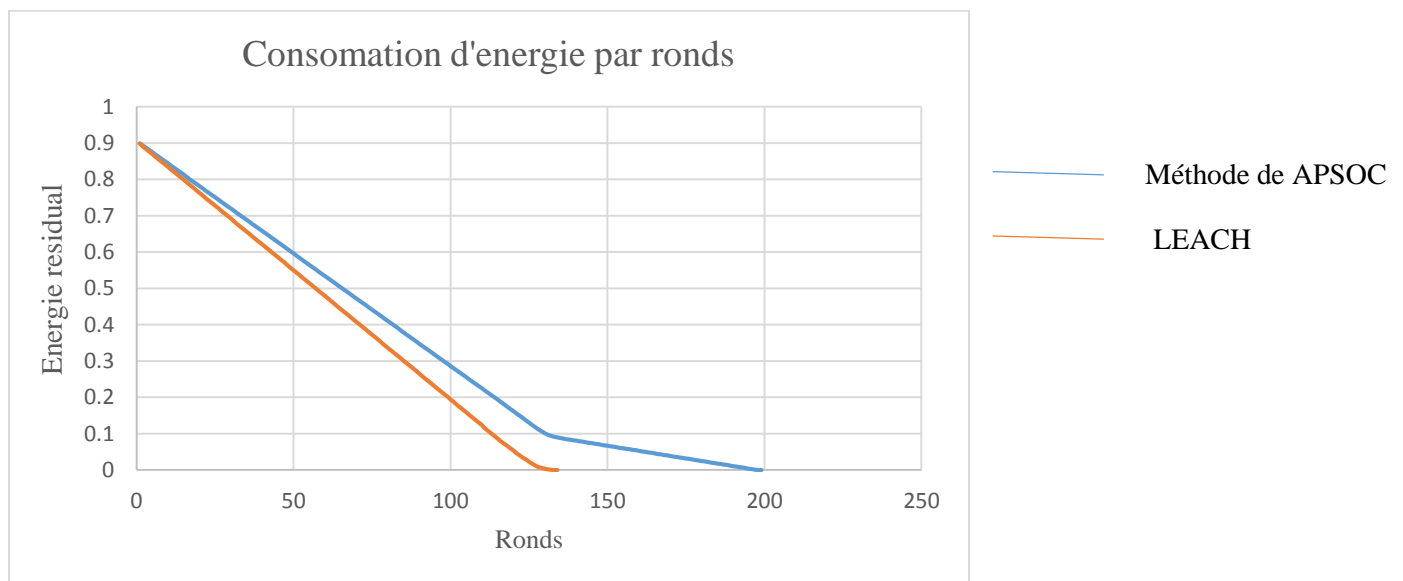


Figure IV.6 : la consommation d'énergie après chaque rond.

On remarque que à chaque nouveau rond l'énergie se diminue pour les deux protocoles APSOC et LEACH avec l'augmentation des ronds, mais avec la méthode APSOC, le réseau réserve leur énergie ce qu'il permet de prolonger la durée de vie de réseaux et reste vivant a plusieurs ronds jusqu'à 200, cela due au meilleur choix des CHs et aussi à la meilleur distribution.

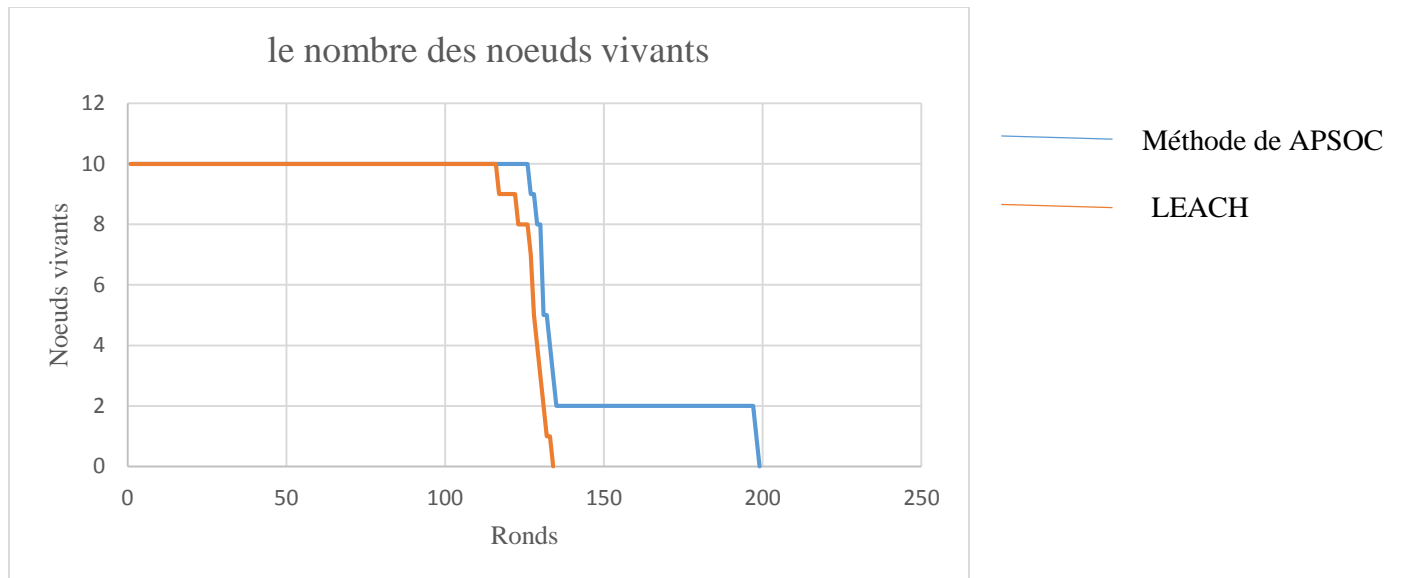


Figure IV.7 : Nombre des nœuds vivants.

On remarque dans ce graphe que, le nombre des nœuds vivant diminue avec l'augmentation des nombre des ronds du réseau Tel que dans la méthode APSOC les nœuds reste vivant jusqu'à 200 ronds par contre dans Leach après le ronds 134 les nœuds sont totalement mort. Pour chacun de ces 2 protocoles, la diminution se continué jusqu'au la mort de tous les nœuds (fin de duré de vie).

D'où, nous pouvons constater que, en termes d'optimisation de la durée de vie du réseau, notre protocole APSOC est toujours plus performant par rapport au LEACH.

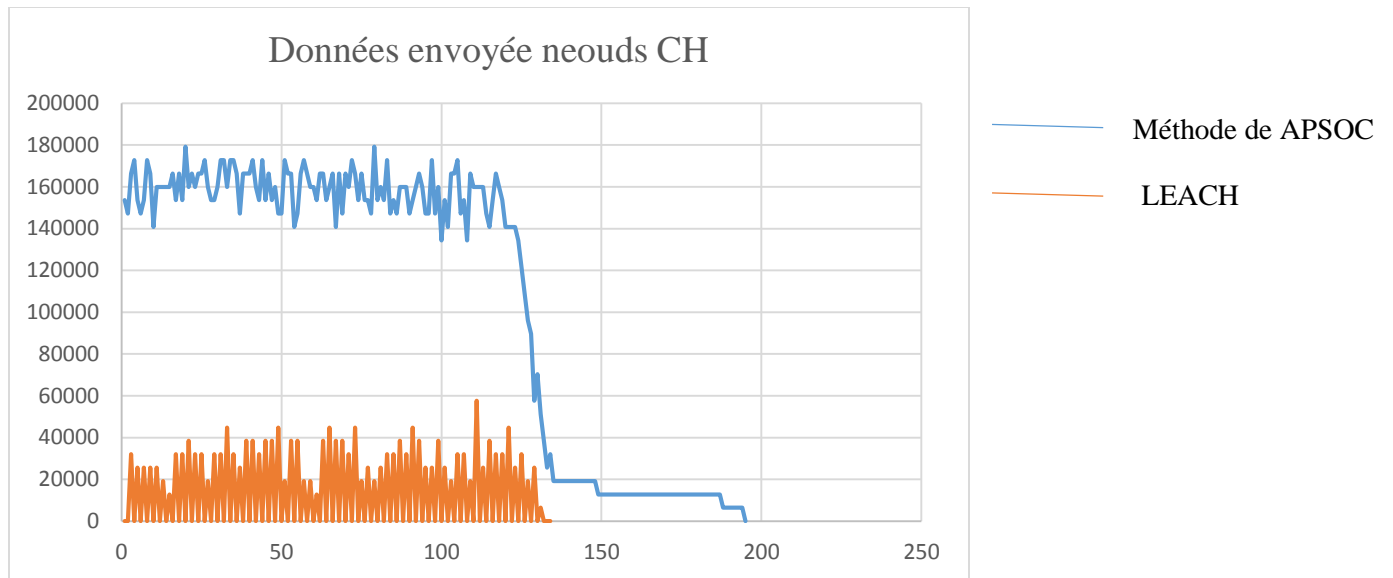


Figure IV.8 : Données envoyées de CHs vers station de base.

IV.6 Conclusion

Dans ce chapitre, nous avons présenté les résultats de la simulation de notre protocole APSOC, effectuée sur l'environnement MATLAB13. En se basant sur les résultats de la stimulation Les effectuées nous avons démontré que le protocole APSOC permet de améliorer la performance de réseau par rapport au protocole LEACH, augmente le gain d'énergie, et par conséquent, prolonge considérablement la durée de vie du réseau.

Conclusion

L'utilisation des batteries par les capteurs est une contrainte critique dans les réseaux de capteurs. Aussi les capteurs sont parfois déployés sans surveillance et en grand nombre, de sorte qu'il est difficile de changer ou de recharger leurs batteries. Pour cela, les algorithmes et les protocoles de communication pour les réseaux de capteurs sans fil doivent minimiser la consommation d'énergie. Mais le taux de consommation d'énergie des capteurs reste très variable selon les protocoles utilisés.

Dans cette thèse, nous nous sommes intéressés au problème de l'énergie et du routage dans les réseaux de capteurs sans fil. Nous avons proposé des solutions permettant d'offrir une meilleure prise en compte des ressources énergétiques du réseau.

Nous avons simulé le fonctionnement de protocole APSOC avec le simulateur MATLAB 13 et comparé avec deux autres protocoles de la littérature à savoir LEACH, afin de montrer sa performance. Les simulations ont montré des bons résultats, une consommation énergétique très réduite donc une prolongation de la durée de vie des réseaux.

Comme une perspective, et pour améliorer les résultats obtenus, nous envisageons, par la suite, les améliorations suivantes:

- ✓ Une technique de routage APSOC multi-sauts (en utilisons des niveaux des CHs jusqu'à l'aboutissement de station de base) a fin de minimisé la distance d'envoi ;
- ✓ Une hybridation entre notre algorithme avec d'autre méthode de la littérature.

Références

- [1] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci , “A Survey on Sensor Networks”. *IEEE Communications Magazine*, August 2002
- [2] Belkheir Khaled et Haned Ahmed ; « Réseaux WiFi ad hoc ». Mémoire d’ingénieur; Institut de télécommunication d’Oran ; Juin 2008.
- [3] Boudjaadar Amina ; « Plateforme basée Agents pour l’aide à la conception et la simulation des réseaux de capteurs sans fil ». Thèse de Magistère ; Université de Skikda ; 2009/2010.
- [4] Sofiane MOAD, La consommation d’énergie dans les réseaux de capteurs sans fil, Etude bibliographique Master Recherche 2 en Informatique IFSIC-Rennes 1 , 2008.
- [5] M. Badet, W. Bonneau. " Mise en place d’une plateforme de test et d’expérimentation", Projet de Master Technologie de l’Internet 1ere année, Université Pau et des pays de l’Adour. 2006.
- [6] D. B. Johnson, D. A. Maltz, and J. Broch, “DSR The Dynamic Source Routing Protocol for Multihop Wireless Ad Hoc Networks”. In *Ad Hoc Networking*, edited by Charles E. Perkins, chapter 5, Addison-Wesley, 2001, pp. 139–172.
- [7] A. Manjeshwar and D. P. Agrawal, “TEEN : A Protocol for Enhanced Efficiency in Wireless Sensor Networks”. 1st International Workshop on Parallel and Distributed Computing Issues in Wireless Networks and Mobile Computing, San Francisco, CA, April 2001.
- [8] K. Akkaya, and M. Younis, “A Survey on Routing Protocols for Wireless Sensor Networks”. *Journal of Ad Hoc Networks*, Vol. 3, No. 3, May 2005, pp. 325-349.
- [9] S. Lindsey and C. Raghavendra, “PEGASIS: Power-Efficient Gathering in Sensor Information Systems”. *Proceedings of the IEEE Aerospace Conference*, vol. 3, Big Sky, MT, USA, March 2002, pp. 1125-1130.
- [10] M. I. Razzak, S. A. Hussain, A. A. Minhas and M. Sher, “Collaborative Image Compression in Wireless Sensor Networks”. *International journal of computational cognition* (<http://www.ijcc.us>), vol. 8, no. 1, march 2010.
- [11] W..R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, “An Application-Specific Protocol Architecture for Wireless Micro sensor Networks”, *IEEE Transactions on the wireless communications*, Vol. 1, No. 4, pp. 660-670, Oct. 2002.
- [12] K. Beydoun . « Conception d’un protocole de routage hiérarchique pour les réseaux de capteurs » Thèse de doctorat, Spécialité : Informatique, l’u.f.r des sciences et techniques de l’université de Franche-Comté, 2009.

- [13] J.N. Al-Karaki and A.E. Kamal, "Routing techniques in wireless sensor networks: a survey. Wireless Communications". IEEE, Decembre 2004.
- [14] E.DHIB « Routage avec QoS temps réel dans les réseaux de Capteurs »Ingénieur en Télécommunications option : Ingénierie des réseaux, école supérieure de communication de Tunis,2006/2007.
- [15] O. YOUNIS, S. FAHMY, «HEED: A hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks», IEEE Transactions on Mobil Computing, vol. 3, no. 4, pp.366-379, 2004.
- [16] Lin H., Chu Y. « A clustering technique for large multihop mobile wireless networks ». Vehicular Technology Conference Proceedings. 2000, Vol. 7, pp. 1545-1549.
- [17] Guillaume CALAS , Optimisation par essaim particulaire, [SCIA] EPITA 2009.
- [18] Maurice Clerc and James Kennedy, The particle swarm - explosion, stability, and convergence in a multidimensional complex space. IEEE Trans. Evolutionary Computation, 6(1) :58–73, 2002.
- [19] Jerome Onwunalu, Louis J. Durlofsky, « *Application of a particle swarm optimization algorithm for determining optimum well location and type* », Smart Fields Meeting, 8 Avril 2009.
- [20] Maurice Clerc, « *L'optimisation par essaim particulaire* », 2003.
- [21] Antoine Dutot et Damien Olivier, « *Optimisation par essaim de particules Application au problème des n-Reines* », Laboratoire Informatique du Havre, Université du Havre, 25 rue Philippe Lebon, 76600 Le Havre.
- [22] James Blondin, « *Particle Swarm Optimization: A Tutorial* », 4 Septembre 2009.
- [23] V. Kawadia and P. R. Kumar, "Power Control and Clustering in Ad Hoc Networks", IEEE INFOCOM. (2003).
- [24] T. Kwon and M. Gerla, "Clustering with Power Control", In Proceedings MILCOM '99, volume 2. (1999).