

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



**Université de Batna 2**  
**Faculté de Mathématiques et**  
**d'Informatique**  
**Département d'Informatique**



## **Thèse**

*En vue de l'obtention du diplôme de*  
**Doctorat en Informatique**

### **Auto-adaptation dans les RCSF hétérogènes pour des** **e-applications**

*Présentée Par*

**Naidja Miloud**

Soutenue le: 24 /05/2018

#### **Membres du jury:**

<i>Président:</i>	Dr. SEGHIR Rachid	(Université de Batna 2).
<i>Rapporteur:</i>	Pr. BILAMI Azeddine	(Université de Batna 2)
<i>Examineurs:</i>	Pr. MAAMRI Ramdane	(Université de Constantine 2)
	Dr. BOUBICHE Djelleddine	(Université de Batna 2)
	Dr. HEDJAZI Djalal	(Université de Batna 2)



*Dédicace*

*A mon très cher père*

*A ma très chère mère*

*A ma très chère grande mère*

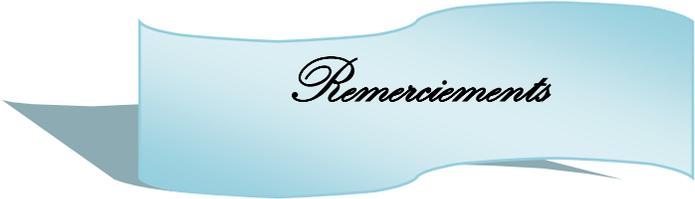
*A mon très cher frère et mes sœurs*

*A toute ma famille*

*A ma famille de Pharma Invest*

*Je dédie ce travail à mes parents. Aucun mot n'est assez fort pour leur exprimer la reconnaissance sincère que je leur porte pour la richesse de leurs enseignements.*

*Merci du fond du cœur*



## *Remerciements*

*Je tiens d'abord à remercier le Bon Dieu pour m'avoir prodigué le courage, la force et la patience tout au long de mon parcours.*

*J'exprime ma profonde gratitude à mon directeur de thèse **Pr. BILAMI Azeddine**, professeur à l'université de BATNA 2, pour avoir dirigé mes travaux de thèse. Il a pris le temps pour me guider, me donner des conseils. Je le remercie pour sa disponibilité, son écoute, son expérience, sa sympathie et ses encouragements qui m'ont permis de mener à bien cette thèse. Je tiens à remercier aussi les membres du laboratoire « L@STIC » pour m'avoir encouragé et soutenu lors des différents exposés de mes rapports de recherche.*

*J'adresse également mes très sincères remerciements à **Dr. SEGHIR Rachid**, Maître de conférence -A-, à l'université de Batna 2, d'avoir accepté de présider mon jury.*

*J'exprime ma plus profonde gratitude au **Pr. MAAMRI Ramdane**, Professeur à l'université de Constantine 2, **Dr. BOUBICHE Djalleddine** et **Dr. HEDJAZI Djalal** Maîtres de conférence –A- à l'université de Batna 2 de m'avoir fait l'honneur d'évaluer cette thèse; qu'ils soient assurés de ma reconnaissance pour l'intérêt qu'ils ont apporté à ce travail.*

*Je ne saurais oublier de remercier **Dr. BOUAM Souheila** pour l'accueil, l'aide, la sympathie sa disponibilité et son soutien dans les démarches administratives. Je remercie **Dr. SEDRATI Maamar** pour ses encouragements.*

*Un grand merci à **Dr. ALIOUAT Zibouda**, Maître de conférences à l'université de Sétif pour ses conseils, ses remarques, ses encouragements et son soutien (professionnel et personnel). Merci infiniment Madame.*

*Merci à mes amis, mes collègues et toute personne qui a contribué à la réalisation de ce travail.*

*Finalement sur un plan plus personnel, je veux remercier les personnes sans qui ce travail n'aurait jamais vu le jour: mon père, ma mère et ma grande mère pour leurs sacrifices, leur patience et tout ce qu'ils ont fait pour m'apporter le bonheur. Qu'ils sachent à travers ces quelques mots combien je leurs suis reconnaissant.*

## ملخص

حسب طبيعة التطبيق، تم اقتراح بنى وبروتوكولات مختلفة لشبكات الاستشعار. وتعتمد طوبولوجيا نشر أجهزة الاستشعار في الشبكات اللاسلكية على التطبيق، كما تؤثر على أداء بروتوكولات الطبقة الفرعية تماما كأي بروتوكول توجيه. حقيقة، إن ملاءمة بروتوكول توجيه معين يعتمد أساسا على قدرات أجهزة الاستشعار اللاسلكي مقيده، متجانسة مع نفس الخصائص، غير متجانسة مع قدرات مختلفة (الخ.) وعلى متطلبات التطبيق.

يكون نشر أجهزة الاستشعار في الشبكة إما حتميا أو منظما ذاتيا. حيث تم اقتراح الكثير من مخططات التوجيه و التكيف الذاتي لشبكات الاستشعار اللاسلكية. وتعتبر الطوبولوجيا الهرمية أكثر نجاحا في هذا المجال لأنها تلبي متطلبات الحفاظ على الطاقة و جودة الخدمة في بيئة متجانسة. و الملاحظ عند مختلف الباحثين هو تطبيق نفس خوارزمية التوجيه على مستوى جميع أجهزة الاستشعار لكل المجموعات. على هذا الأساس قمنا باقتراح حلين في مجال التكيف الذاتي لشبكات الاستشعار اللاسلكية المتجانسة وغير المتجانسة، حيث يعتمد الاقتراح الأول على تطبيق إستراتيجيات مختلفة لكل مجموعة من شأنها أن تكون الأنسب اعتمادا على التطبيق وإعدادات خاصة، هذا ما يجعل الشبكة غير متجانسة أي قيام كل مجموعة من أجهزة الاستشعار بتطبيق إستراتيجية قد تكون مماثلة للمجموعات الأخرى و قد تكون مختلفة خلال زمن الرقابة في ظل اختلاف الظروف، حسب مجموعة من الشروط والتي تهدف إلى الجمع بين الحفاظ على الطاقة وبالتالي تمديد عمر الشبكة، كما تقوم بتحسين جودة الخدمة من خلال وصول نسبة كبيرة من المعطيات إلى المحطة الرئيسية.

في هاته الرسالة قمنا بتشكيل بروتوكول يعتمد على مختلف الاستراتيجيات لبروتوكولات أخرى معروفة مثل هيب، أبتين، ليتش، بيغاسيس وغيرها... مع الأخذ بعين الاعتبار لنقاط القوة والضعف لديهم، والتركيز على عدم تجانس الشبكة من عدة جوانب كاختلاف البروتوكولات السائدة و الاعتماد على طريقة محكمة لإرغام بعض الأجهزة على النوم إلى غاية الحاجة إليهم، وفي الأخير بينت المحاكاة التي قمنا بها باستعمال برنامج الان اس 2 نقص في استعمال الطاقة وبالتالي زيادة في عمر الشبكة مقارنة بالبروتوكولات الأخرى.

وفي نفس الوقت، يقدم الاقتراح الثاني بروتوكول توجيه هيب الذاتي التنظيم: بروتوكول توجيه الهجين، ويعتبر هذا الأخير امتداد للبروتوكول هيب، والذي يقدم حلا فعالا للتنظيم الذاتي على مستوى العقد العادية، رؤساء المجموعات و المحطة القاعدة. وبالتالي، يضمن هذا الحل نقل جميع البيانات بشكل صحيح إلى المحطة القاعدة على الرغم من وجود أنواع مختلفة من الأعطال، بسبب الطاقة، وفشل الأجهزة، وأخطاء الاتصالات، والهجمات الخبيثة الخ.

**كلمات البحث:** عدم تجانس بروتوكولات التوجيه، شبكات الاستشعار اللاسلكية، شبكات الاستشعار اللاسلكية غير متجانسة، نظام التكيف الذاتي.

## Abstract

Depending on the application, different architectures and protocols have been considered for sensor networks. Topological deployment of the sensors in WSNs is application dependent and affects the performance of sub layer protocols as the routing protocol. Really, the appropriateness of a particular routing protocol mainly depends on the capabilities of the nodes (constrained, homogenous with same characteristics, heterogeneous with different capacities...) and on the application requirements.

The network deployment is either deterministic or self-organizing. Many self-adaptation routing schemes have been proposed for sensor networks. The most relevant of them consider a hierarchical topology and aim to meet energy conservation and QoS requirements in a homogeneous environment. In such networks, one specific algorithm is commonly applied by all nodes inside clusters. On this basis we have proposed two solutions for the self-adaptation in homogeneous and heterogeneous WSN. The first proposal depends on heterogeneous routing by applying different strategies according to specific parameters at the same time inside different clusters; each cluster can adopt different strategies that are the most suitable for a specific application at different moments under different conditions. This approach leads to a new self-adaptation protocol based on heterogeneity of the routing process in a multi hop clustering WSN. The proposal uses a set of mechanisms that have been adopted in well known protocols (HEEP, APTEEN, LEACH, PEGASIS...) taking into account their strengths and weaknesses. Simulations under NS2 show that our proposal, based on heterogeneous routing protocol, prolongs the network lifetime with different ratios compared to HEEP, PEGASIS and others.

The second proposal presents a routing protocol HEEP-SO (HEEP-Self-Organized): a hybrid routing protocol as an extension of HEEP (Hybrid Energy Efficiency Protocol), this solution is effective for self-organization in the level of ordinary nodes, CHs and SINK. Thus, this solution is appropriate for applications that require reliability and fault tolerance because it ensures that all data will be transmitted correctly to the base station despite the presence of different types of faults, due to energy, hardware failure, communication errors, and malicious attacks ... etc.

**Key words:** Heterogeneous Routing Protocol; WSN; Heterogeneous WSN; Self-Adaptation.

## Résumé

En fonction de l'application, différentes architectures et protocoles ont été considérés pour les réseaux de capteurs. La topologie de déploiement des capteurs dans les RCSF dépend de l'application et affecte les performances des protocoles de sous-couche comme le protocole de routage. Vraiment, la pertinence d'un protocole de routage particulier dépend principalement des capacités des nœuds (contraintes, homogènes avec les mêmes caractéristiques, hétérogènes avec des capacités différentes ...etc.) et sur les exigences de l'application.

Le déploiement du réseau est soit déterministe, soit auto-organisé. De nombreux schémas de routage et d'auto-adaptation ont été proposés pour les réseaux de capteurs. La topologie hiérarchique dans les RCSF répond aux exigences de conservation d'énergie et de QoS dans un environnement homogène. Dans de tels réseaux, un algorithme spécifique est appliqué au niveau de tous les nœuds capteurs. Sur cette base, nous avons proposé deux solutions pour l'auto-adaptation dans les RCSF homogènes et hétérogènes. La première proposition considère un routage hétérogène en appliquant différentes stratégies qui seront les plus appropriées dépendamment d'une application en fonction des paramètres spécifiques dans chaque cluster. Cette approche conduit à un nouveau protocole d'auto-adaptation basé sur l'hétérogénéité des protocoles de routage dans les RCSF. La proposition utilise un ensemble de mécanismes qui ont été adoptés dans des protocoles bien connus (HEEP, APTEEN, LEACH, PEGASIS ...) en tenant compte de leurs forces et faiblesses. Les simulations effectuées sous NS2 montrent que notre proposition, basée sur un protocole de routage hétérogène, prolonge la durée de vie du réseau par rapport à HEEP, PEGASIS et autres.

La deuxième solution présente un protocole de routage HEEP-AO (HEEP-Auto-Organisé): un protocole de routage hybride, qui est une extension du protocole HEEP. La solution est efficace pour l'auto-organisation au niveau des nœuds ordinaires, cluster-heads et station de base. Elle est appropriée pour des applications nécessitant une fiabilité et une tolérance aux pannes car elle garantit que toutes les données soient transmises correctement à la station de base malgré la présence de différents types de défauts, dus à l'énergie, aux pannes matérielles, aux erreurs de communication et aux attaques malveillantes ... etc.

**Mots-clés:** Protocole de routage hétérogène; RCSF; RCSF hétérogène; Auto-adaptation.

# ***Table des matières***

Introduction générale .....	1
-----------------------------	---

## **Partie 1 : Introduction au domaine de recherche.**

### **Chapitre 1 : Généralités sur les réseaux de capteurs sans fil.**

1. Introduction .....	7
2. Un Capteur sans fil.....	7
3. Architecture matérielle d'un capteur sans fil .....	8
4. Un réseau de capteur sans fil.....	10
5. Spécificités des réseaux de capteurs sans fil.....	11
6. La pile protocolaire des RCSF.....	13
7. Les systèmes d'exploitation dans les RCSF.....	15
8. Classification des RCSF.....	16
8.1. Selon le mode d'acquisition et de livraison des données au puits.....	16
8.2. Selon la distance entre les nœuds capteurs et le puits .....	17
8.3. Selon le modèle de mobilité dans le réseau .....	18
8.4. Selon les capacités des nœuds du réseau.....	18
9. Les technologies de transmission dans les RCSF. ....	19
10. Conclusion.....	20

### **Chapitre 2 : Etat de l'art : Auto-adaptation, routage, et applications dans les RCSF.**

1. Introduction.....	22
2. Auto-adaptation .....	22
2.1. Définition et rôle .....	22

2.2.	Concepts et défis.....	23
2.3.	Les objectifs de l'auto-adaptation.....	24
2.4.	Solutions proposées dans la littérature.....	24
3.	Routage .....	26
3.1.	Les principales classes des protocoles de routage dans les RCSF.....	26
3.1.1.	Selon les paradigmes de communication.....	26
3.1.2.	Selon l'établissement de la route.....	27
3.1.3.	Selon le fonctionnement du protocole.....	28
3.1.4.	Selon la topologie du réseau.....	29
3.2.	Caractéristiques d'un protocole hiérarchique.....	31
3.2.1.	Formation de clusters.....	31
3.2.2.	La communication intra et inter-cluster.....	31
3.2.3.	Le niveau d'agrégation de données.....	32
3.3.	Les principaux protocoles hiérarchiques existants .....	33
4.	Applications .....	36
4.1.	E-application .....	36
4.2.	Exemples d'applications .....	36
5.	Conclusion .....	39

### **Chapitre3 : Etat de l'art : Les RCSF hétérogènes.**

1.	Introduction.....	41
2.	Les conditions de fonctionnement.....	41
3.	Quelques modèles d'hétérogénéité.....	42
3.1.	L'hétérogénéité matérielles.....	42
3.2.	L'hétérogénéité logicielle .....	43
4.	Exemple de « e-application » dans les RCSF hétérogènes .....	45
5.	Conclusion.....	48

## **Partie 2 : Contribution.**

### **Chapitre4 : Approches d'auto-adaptation dans les RCSF hétérogènes.**

1. Introduction .....	51
2. Motivation.....	51
3. Etude comparative des protocoles de routage.....	52
3.1. Quelques avantage et limites des protocoles de routage.....	52
4. Préparation de notre RCSF hétérogène.....	56
5. Proposition d'une auto-adaptation dans un protocole de routage hétérogène.....	60
5.1. Sélection basée sur la consommation d'énergie, la distance et la densité.....	60
6. Le processus de migration .....	67
7. Algorithme détaillé d'ADAPTRP.....	69
7.1. ADAPTRP: ADAPTative Routing Protocols.....	69
8. Conclusion .....	71

### **Chapitre 5 : Approches d'auto-organisation dans les RCSF homogènes.**

1. Introduction .....	73
2. La proposition d'auto-organisation.....	73
3. HEEP-Auto-Organisé (HEEP-AO) .....	73
4. Nouvelle Méthode d'Auto-Organisation dans les RCSF.....	75
4.1. Le déroulement d'auto-organisation .....	75
4.2. Les hypothèses.....	75
5. Le mécanisme d'auto-organisation.....	76
5.1. L'Auto-Organisation dans le cas d'une Panne de réception.....	76

5.2.	L'Auto-Organisation en cas d'une Panne d'émission.....	78
5.3.	L'Auto-Organisation en cas d'une Panne d'énergie.....	79
5.4.	L'Auto-Organisation en cas d'une Panne de capture.....	80
5.5.	L'Auto-Organisation en cas d'une Panne de traitement.....	81
6.	Conclusion.....	82

## **Chapitre 6 : Mise en œuvre et simulation**

1.	Introduction.....	84
2.	Environnement de simulation.....	84
2.1.	Objectif de la simulation.....	85
2.2.	Choix du simulateur NS2.....	85
2.3.	Choix de plateforme OS.....	86
2.4.	Le choix du langage.....	86
3.	Etapas d'implémentation de notre protocole.....	86
3.1.	Préparation de l'environnement d'implémentation.....	87
3.2.	Implémentation.....	89
4.	Évaluation des performances du protocole ADAPTRP.....	90
5.	Évaluation des performances du protocole HEEP-AO.....	95
6.	Conclusion.....	98

<b>Conclusion générale</b> .....	<b>99</b>
----------------------------------	-----------

## ***Liste des figures***

1	Structure de la thèse.....	4
<b>Chapitre 1</b>		
1.1	Un capteur sans fil.....	8
1.2	Architecture d'un capteur.....	8
1.3	Réseaux de capteurs sans fil.....	10
1.4	Modèle en couches pour la communication dans les RCSF.....	14
1.5	Les différents modes de communications pour les RCSF.....	17
<b>Chapitre 2</b>		
2.1	Classification des protocoles de routage pour les RCSF.....	27
2.2	Routage à plat dans un RCSF.....	29
2.3	Routage hiérarchique dans un RCSF.....	30
2.4	Architecture de communication intra et inter-cluster.....	32
2.5	Architecture de communication du protocole HEEP .....	35
2.6	Quelques applications des réseaux de capteurs sans fil.....	37
<b>Chapitre 3</b>		
3.1	Exemple d'un cluster dans un RCSF et actionneur.....	42
3.2	Hétérogénéité de l'unité de capture.....	43
3.3	Topologie hétérogène exploitant quatre canaux dans un RCSF.....	45
3.4	Exemple de e-application dans les RCSF hétérogènes .....	46
3.5	Les systèmes d'irrigation existants.....	48
<b>Chapitre 4</b>		
4.1	Modèle d'hétérogénéité dans les RCSF.....	57
4.2	Routage Hétérogène pour les RCSF.....	57

4.3	Exemple de réseau de capteurs hétérogène.....	59
4.4	Modèle de consommation d'énergie .....	61
4.5	Le processus de migration.....	68

## Chapitre 5

5.1	Architecture de communication du protocole HEEP-AO.....	74
5.2	Auto-organisation en cas d'une panne de réception.....	77
5.3	Auto-organisation en cas d'une panne d'émission.....	78
5.4	Auto-organisation en cas d'une panne d'énergie.....	79
5.5	Auto-organisation en cas d'une panne de capture.....	80
5.6	Auto-organisation en cas d'une panne de traitement.....	81

## Chapitre 6

6.1	Positions des nœuds et de la station de base.....	90
6.2	Quantité de données reçues en fonction de la durée de la simulation.....	92
6.3	Quantité de données reçues par la SB en fonction de l'énergie consommée.....	92
6.4	Nombre de nœuds vivants en fonction de la durée de la simulation.....	93
6.5	Nombre de nœuds vivants en fonction de la quantité de données reçues par la SB.....	94
6.6	Quantité d'énergie consommée en fonction de la durée de la simulation.....	94
6.7	Quantité de données reçues (panne de la SB).....	95
6.8	Quantité de données reçues (panne de la SB, des CHs et nœuds ordinaires).....	96
6.9	Nœuds défectueux détectés dans HEEP et HEEP-AO.....	97

## ***Liste des tableaux***

### Chapitre 4

Tableau 4.1	Comparaison des différents protocoles de routage hiérarchique.....	56
Tableau 4.2	Détermination du seuil distance et énergie.....	66

### Chapitre 6

Tableau 6.1	Les paramètres de simulation.....	91
-------------	-----------------------------------	----

# ***Liste des algorithmes***

## Chapitre 4

Algorithme 4.1	Calcul de la précision perdue $La_N$ .....	63
----------------	--	----

## *Liste des équations*

### Chapitre 4

Equation 4.1	$E_K(P_j)$ .....	61
Equation 4.2	$C1E_K(P_j)$ .....	61
Equation 4.3	$DistK(P_j)$ .....	62
Equation 4.4	$C1DistK(P_j)$ .....	62
Equation 4.5	$DNK$ .....	62
Equation 4.6	$C3DNk$ .....	63
Equation 4.7	$E_{total}$ .....	65

## ***Glossaire des acronymes***

ADAPTRP: ADAPTative Routing Protocols

ADP : Adaptive dynamic programming

AODV : Ad hoc On-Demand Distance Vector

APTEEN : AdaPtive Threshold sensitive Energy Efficient sensor Network protocol

BS : Base station

CAN : Convertisseur analogique/numérique

CDMA : Code division multiple access

CH : Cluster Head

DPA : Analyse de puissance différentielle

DSR : Dynamic Source Routing protocol

EEIHN : energy-efficient improvement in heterogeneous networks

FZRP : Fisheye Zone Routing Protocol

GPS: Global Positioning System

HEED : Hybrid Energy-Efficient Distributed clustering

HEEP: Hybrid Energy Efficiency Protocol

H-PEGASIS : Hierarchical-PEGASIS

HWSN: Heterogeneous Wireless Sensor Network

IoT: Internet of Things

LEACH : Low Energy Adaptive Clustering Hierarchy

LEACH-C : LEACH-Centralisé

MAC: Message Authentication Code

NS : Network simulator

PEGASIS : Power-Efficient Gathering in Sensor Information Systems

RAP : Routage avec Agrégation Partielle

RAT : Routage avec Agrégation Totale

RCSF : Réseau de capteurs sans fil

RSA : Routage Sans Agrégation

SB : Station de base

SBP : Station de base primaire

SBS : Station de base secondaire

SEP: Stable election protocol

TCL : Tool Command Language

TCP: Transmission Control Protocol

TDMA : Time division multiplexed access

TEEN : Threshold sensitive Energy Efficient sensor Network protocol

TinyOS: Tiny operating system

TinySec: Tiny Security

UDP: User Datagram Protocol

USB Universal Serial Bus

WiFi: Wireless Fidelity

WiMAX: Worldwide Interoperability for Microwave Access

WLAN : Wireless Local Area Network

WMAN : Wireless Metropolitan Area Network

WPAN : Wireless Personal Area Network

WSN: Wireless Sensor Network

WWAN : Wireless Wide Area Network



# ***Introduction générale***

## **Motivations et Problématique**

Aujourd'hui nous voyons naître un grand nombre d'applications basées sur de nouveaux concepts comme les réseaux intelligents, les maisons intelligentes, et les transports intelligents; ce sont des systèmes d'infrastructure qui relient notre monde plus que nous n'aurions jamais cru possible. Ces concepts sont tous impliqués dans un concept unique "l'internet des objets", où grâce à l'utilisation de capteurs; toute l'infrastructure physique est associée étroitement aux technologies de l'information et de la communication; où une surveillance et une gestion intelligente et sécurisée peuvent être réalisées grâce à l'utilisation de dispositifs intégrés en réseau [1]. Les RCSF sont considérés comme une méthode révolutionnaire de collecte d'informations pour construire le système d'information et de communication qui améliorera grandement la fiabilité et l'efficacité des systèmes d'infrastructure. Par rapport à la solution filaire, les RCSF offrent un déploiement plus simple et une meilleure flexibilité des périphériques.

Dans un tel système dynamique sophistiqué, les dispositifs sont interconnectés pour collecter et transmettre à des utilisateurs d'applications via Internet des informations de mesure utiles et des instructions de contrôle à partir de réseaux de capteurs. On peut citer à titre d'exemple : applications médicales pour le contrôle de la sante (e-health), ou Capteurs et objets connectés pour l'agriculture numérique, ou encore pour la gestion et le contrôle des ressources hydriques, ...etc.

Ces applications à travers l'internet 'e-applications' ont pour caractéristique commune l'utilisation des nouvelles technologies d'information et de communication et l'Internet, pour améliorer la qualité de la surveillance en facilitant l'accès à des ressources et des services, ainsi que les échanges des informations et la collaboration à distance. Dans les RCSF, Les utilisateurs communiquent avec l'environnement de captage via les modes de communication synchrones et asynchrones supportés par des outils web le plus souvent intégrés à des plateformes, cette technique offre la possibilité de véhiculer des interactions entre les différents utilisateurs et l'environnement cible.

Par contre, on peut noter des différences de ces applications au niveau des objectifs et des paramètres d'intérêt ; si pour certaines c'est la fiabilité, la disponibilité ou la durée de

vie du réseau qui sont primordiales, pour d'autres c'est la puissance de calcul, le temps de réponse (applications temps réel), la qualité du service, ... qui sont importantes.

Pour répondre aux divers besoins (hétérogènes) de ces applications (limiter le nombre de communications, préserver l'énergie, fiabilité, rapidité des communications,...), les réseaux de capteurs sans fil utilisent différentes stratégies de routage.

## **Contexte de la thèse**

Dans cette thèse on se réfère à des RCSF homogènes et hétérogènes. A l'inverse des réseaux homogènes, les réseaux hétérogènes sont des réseaux constitués de nœuds de capteurs ayant des capacités différentes, telles qu'une puissance de calcul et une plage de détection différentes. Dans ces réseaux le contrôle de déploiement et de topologie est plus complexe que dans les RCSF homogènes.

La plupart des protocoles efficaces en énergie conçus pour les réseaux de capteurs sans fil sont hiérarchiques c.à.d. basés sur la technique de clustering [2]. Avec des avantages particuliers liés à l'évolutivité et à une communication efficace, le clustering constitue un moyen efficace et évolutif de concevoir et d'organiser des réseaux à grande échelle pour la communication de données avec l'efficacité énergétique [3]. Dans une architecture hiérarchique, les nœuds ayant une capacité énergétique plus élevée peuvent être utilisés pour traiter et envoyer l'information tandis que les nœuds de faible énergie peuvent être utilisés pour effectuer la détection à proximité de la cible. Certains des protocoles de routage dans ce groupe incluent une hiérarchie de clustering adaptative basse énergie (LEACH) [4], LEACH-C [4], LEACH-M [5], l'amélioration de l'efficacité énergétique dans les réseaux hétérogènes (EEIHN) [6], algorithme approché de programmation dynamique (ADP) [7] et approche efficace de sélection de cluster-head pour le traitement collaboratif des données (CHSCDP) [8].

De nombreux articles, dans la littérature, traitent de la capacité d'auto-adaptation des protocoles de routage dans les RCSF. Au meilleur de notre connaissance, nous présentons dans cette thèse le premier travail qui suggère une auto-adaptation basée sur l'hétérogénéité du processus de routage.

## **Contributions**

Nous proposons différents protocoles de routage dans différents clusters pour les RCSF hiérarchiques. En d'autres termes, le routage est variable dans le réseau et variable en même temps dans le cluster lui-même (*Figure 4.2*). La modification des protocoles est faite

par un algorithme de migration proposé pour acheminer l'évolution du protocole à un autre en fonction du changement de l'environnement.

Ce concept améliore encore la consommation d'énergie des données reçues par la station de base (SB) et la durée de vie du réseau. Après cela, pour enrichir notre travail et en fonction de l'énergie initiale, nous utilisons l'idée de Smaragdakis et al [9] en nous concentrant sur l'ajout de nœuds sophistiqués qui ont plus de capacité énergétique que les nœuds normaux. L'utilisation d'une telle idée améliore l'efficacité énergétique et prolonge la durée de vie du réseau. L'avantage du réseau est dans ses nœuds sophistiqués qui peuvent être utilisés pour effectuer des tâches plus complexes telles que les coordinateurs, connus sous le nom Cluster Heads (CH). Par conséquent, nous obtenons un RCSF hétérogène avec en plus d'une hétérogénéité des protocoles de routage au niveau du réseau. En outre, dans notre proposition, nous présentons un autre algorithme basé sur la désactivation des nœuds inutiles pour renforcer le concept d'auto-adaptation dans les réseaux de capteurs sans fil hétérogènes (HWSN).

En résumé, nous proposons deux mécanismes pour l'auto-adaptation dans les réseaux de capteurs sans fil : le premier est une solution efficace pour l'auto-adaptation dans les RCSF hétérogènes. La solution est basée sur un nouveau protocole proposé et baptisé ADAPTRP. Ce protocole repose sur une politique de migration vers quelques protocoles de routage hiérarchiques. Le deuxième mécanisme est une proposition qui concerne l'auto-organisation dans les RCSF homogènes. C'est un protocole de routage hybride efficace en énergie et en durée de vie du réseau, appelé HEEP-AO qui est une extension du protocole HEEP.

## **Organisation de la thèse**

La suite du manuscrit est organisée en deux parties, la première consiste à présenter le domaine de recherche, tandis que la deuxième est consacrée à nos contributions en termes d'auto-adaptation dans les RCSF homogènes et hétérogènes comme l'indique la *figure 1*. La première partie est structurée en trois chapitres. Le premier chapitre est destiné à introduire les réseaux de capteurs sans fil. Le deuxième chapitre met le point sur l'auto-adaptation, routage, et applications dans les RSCF. Ensuite, les RCSF hétérogènes vont être abordé dans le troisième chapitre. La deuxième partie de notre travail est aussi divisée en trois chapitres, dont les deux premiers chapitres consistent à présenter respectivement notre protocole de routage ADAPTRP, ainsi que le protocole de routage HEEP-AO. Le dernier chapitre est consacré à notre étude expérimentale, dans le but d'évaluer les performances de nos contributions.

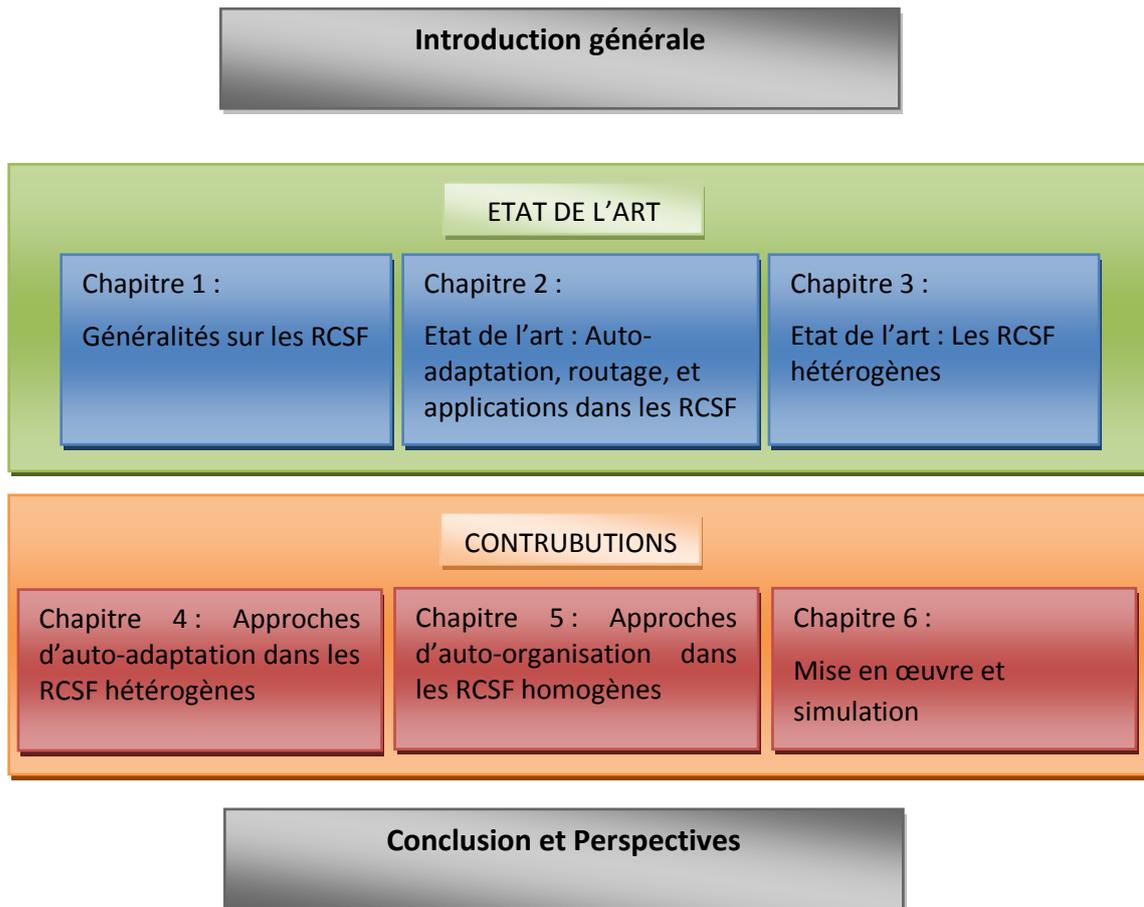


Figure1. Structure de la thèse.

# **Partie 1 : Introduction au domaine de recherche**

# **Chapitre 1**

## **Généralités sur les Réseaux de capteurs Sans fil**

### 1. Introduction

En 1998, l'équipe des professeurs Pister et Khan de l'université de Berkeley en Californie réalise le projet Smart Dust [10, 11, 12]. Ce projet consiste à créer le premier microsystème d'une taille millimétrique capable de mesurer des informations grâce à des capteurs et de les communiquer vers l'extérieur également. Le résultat de ce projet a permis aux nœuds de capteurs communicants et aux réseaux de capteurs sans fils de voir le jour.

Les capteurs sont généralement déployés dans un environnement et sont attachés à une station de base appelée puits. Ces capteurs doivent coopérer entre eux pour s'auto adapter afin de constituer un Réseau de Capteurs Sans Fil (RCSF).

Un réseau de capteurs sans fil est un réseau formé d'un grand nombre de nœuds de capteurs dans lesquels chaque nœud de capteur est constitué de quatre composants de base: une unité de détection, une unité de traitement, une unité d'émission-réception et une unité d'énergie.

Les RCSF sont constitués généralement d'un grand nombre de capteurs homogènes ou hétérogènes à faible ressources déployés autour ou dans une zone à observer en vue de collecter et transmettre des données environnementales vers un ou plusieurs points de collecte, d'une manière autonome. Les spécificités les plus frappantes de ces nœuds sont leurs capacités d'auto-adaptation, de coopération, leur rapidité de déploiement, leur tolérance aux pannes et leur faible coût. En ce qui concerne leurs caractéristiques (absence d'infrastructure, contrainte d'énergie, topologie dynamique, nombre important de capteurs, sécurité physique limitée, capacité réduite des nœuds,...etc.).

Dans ce chapitre nous commençons par introduire les réseaux de capteurs sans fil, leurs concepts et caractéristiques.

### 2. Un Capteur sans fil

Un nœud capteur est composé de plusieurs éléments ou modules correspondant chacun à une tâche particulière d'acquisition, de traitement, ou de transmission de données. Il comprend également une source d'énergie [13, 14].

Les capteurs (*Figure 1.1*) sont des dispositifs de taille extrêmement réduite avec des ressources de calcul, de stockage, de communication et d'énergie très limitées, autonomes, capable de traiter des informations et de les transmettre, via les ondes radio, à une autre entité (capteurs, unité de traitements) sur une distance limitée à quelques mètres.



Figure 1.1. Un capteur sans fil.

### 3. Architecture matérielle d'un capteur sans fil

Généralement, un capteur sans fil permet de mesurer des grandeurs physiques ou des paramètres d'usage (détection) afin d'évaluer ou contrôler l'environnement, mais également de communiquer avec d'autres entités via un médium radio. L'architecture matérielle d'un nœud est composée de quatre unités de base : une unité de capture (*Sensing unit*), une unité de traitement (*Processing unit*), une unité d'émission/réception (*Transceiver unit*) et une unité d'énergie (*Power unit*). Il peut contenir également, suivant son domaine d'application, des modules supplémentaires tels qu'un système de localisation (GPS), ou bien un système générateur d'énergie (cellule solaire). On peut même trouver des micro-capteurs, un peu plus volumineux, dotés d'un système mobilisateur chargé de déplacer le micro-capteur en cas de nécessité. La *figure 1.2* représente l'architecture de capteur sans fil.

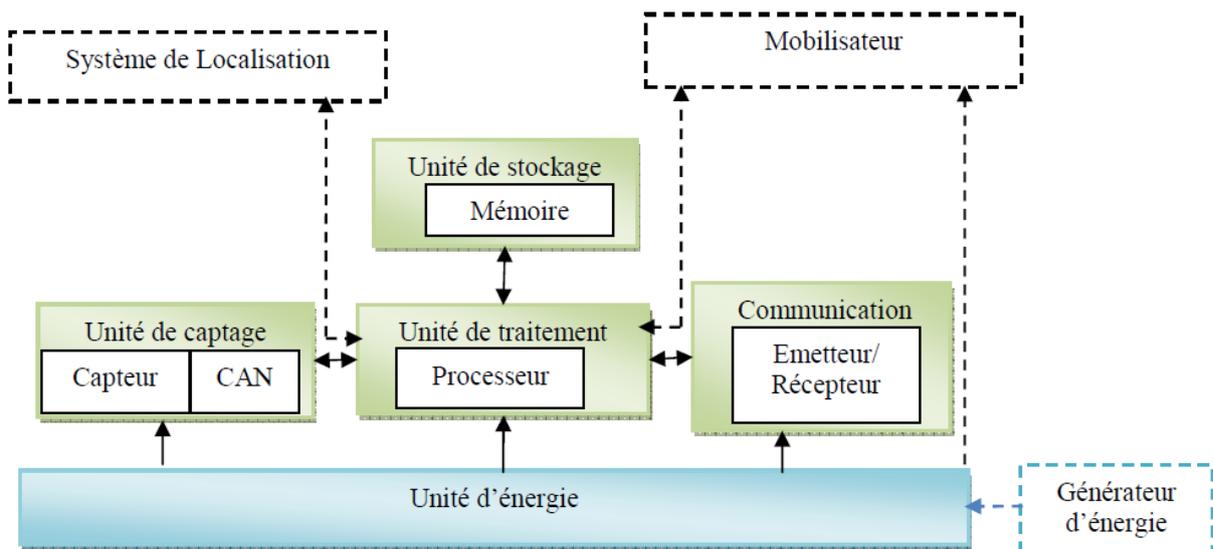


Figure 1.2. Architecture d'un capteur [15].

### ➤ L'unité de capture

Elle comporte deux composants, un dispositif de capture physique qui prélève l'information de l'environnement local : capteur analogique ou numérique, et un adaptateur soit un convertisseur analogique/numérique CAN ou bien un adaptateur logique (de niveau de tension). Pour les RCSF, les capteurs sont choisis selon l'application voulue, ils existent des milliers de capteurs citons les capteurs de température, de pression, de présence, d'humidité, de niveau, de lumière, de vibration, de distance, infrarouge, caméra, etc.

### ➤ L'unité de traitement

On appelle généralement Mote la carte physique utilisant le système d'exploitation pour fonctionner. Celle-ci a pour cœur le bloc constitué du processeur et des mémoires RAM et Flash. Cet ensemble est à la base du calcul binaire et du stockage, temporaire pour les données et définitif pour le système d'exploitation. Cette unité est chargée d'exécuter les protocoles de communications qui permettent de faire collaborer le nœud avec les autres nœuds du réseau. Elle peut aussi analyser les données captées pour alléger la tâche du nœud puits [16]. Ils existent différents choix pour les unités de traitement : les microcontrôleurs (ATmega, PIC, rPIC...), les DSPs, les FPGA ou les PSOC, selon l'application l'unité de traitement est choisie, sa tâche est d'exécuter les algorithmes de commande de l'application, control du protocole de communication et la gestion d'énergie [17].

### ➤ L'unité de transmission

Les composants utilisés pour réaliser la transmission sont des composants classiques. Cette unité comporte deux module : un module radiofréquence d'émission / réception permettant la communication sans fil entre les différents nœuds du réseau, et un module série permettant directement l'interfaçage entre l'utilisateur et le nœud, facilitant ainsi la reprogrammation de l'unité de traitement. Ainsi on retrouve les mêmes problèmes que dans tous les réseaux sans fil : la quantité d'énergie nécessaire à la transmission augmente avec la distance. Pour les réseaux sans fil classiques (LAN, GSM) la consommation d'énergie est de l'ordre de plusieurs centaines de milliwatts, et on se repose sur une infrastructure alors que pour les réseaux de capteurs, le système de transmission consomme environ 20 mW et possède une portée de quelques dizaines de mètres [18]. Pour augmenter ces distances tout en préservant l'énergie, le réseau utilise un routage multi-sauts.

### ➤ L'unité d'énergie

Pour des réseaux de capteurs sans fil autonomes, l'alimentation est une composante cruciale. Il y a essentiellement deux aspects : premièrement, stocker l'énergie et la fournir sous la forme requise ; deuxièmement, tenter de reconstituer l'énergie consommée par un réapprovisionnement grâce à une source externe au nœud capteur, telles les cellules solaires. Le stockage de l'énergie se fait traditionnellement en utilisant ses piles. À titre indicatif, ce sera souvent une pile normale d'environ (2,2 – 2,5) Ah fonctionnant à 1,5 V [18, 19].

### 4. Un réseau de capteur sans fil

Un réseau de capteurs sans fil (RCSF), ou *Wireless Sensor Network (WSN)* est composé de centaines ou de milliers de capteurs. Ces appareils, appelés en anglais **nodes**, sont alimentés par des piles et sont typiquement déployés de façon plus ou moins aléatoire dans une zone géographique appelée zone de captage, ou zone d'intérêt. Généralement, ces capteurs font des mesures périodiques et utilisent une communication sans fil pour acheminer les données collectées à un dispositif plus puissant appelé nœud puits (sink), ou station de base (base station), qui les traite en calculant par exemple leur maximum, moyenne ou médiane [20]. Ensuite, La station de base transmet ces données par Internet ou par satellite à l'ordinateur central pour analyser ces données et prendre des décisions. La figure suivante présente un RCSF.

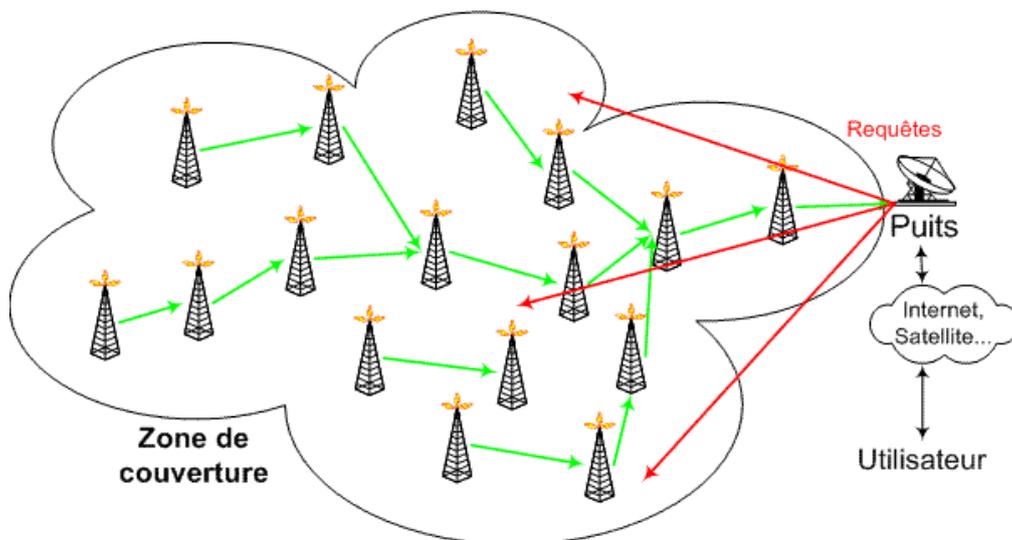


Figure 1.3: Réseaux de capteurs sans fil.

### 5. Spécificités des réseaux de capteurs sans fil

Les réseaux de capteurs sans fil sont des réseaux ad-hoc spécifiques avec un nombre de nœuds plus conséquents, une énergie limitée et une puissance de calcul plus faible que les réseaux ad-hoc classiques. Ce sont ces particularités que nous introduisons dans la partie suivante [21].

#### ➤ La topologie

La topologie que l'on retrouve classiquement au sein des réseaux de capteurs sans fil est un ensemble de capteurs qui sont déposés de manière hétérogène sur une zone ou des objets voire des individus mouvants. Chaque nœud peut communiquer avec les autres nœuds qui sont situés dans sa zone de couverture.

Les réseaux de capteurs sans fil sont plus souvent reliés à une ou plusieurs bases. Ces bases ont pour mission de récupérer les informations circulant sur les réseaux, et de les stocker ou bien de les envoyer directement via une liaison internet ou une liaison GSM.

Ces bases peuvent être par exemple un ordinateur portable ou un capteur de puissance plus important que les autres nœuds classiques. Elles peuvent avoir un rôle de gestionnaire du réseau et elles font souvent le lien entre l'utilisateur et le réseau.

#### ➤ Le routage

Pour limiter le nombre de communications coûteuses en énergie, les réseaux de capteurs sans fil utilisent des protocoles de routage efficaces. Une solution souvent utilisée est la clustérisation, qui divise le réseau en plusieurs clusters. Dans chacun de ces clusters, un nœud maître (Cluster-Head) est élu et aura pour mission de récupérer les informations des nœuds du cluster dont il a la charge pour les transmettre aux autres clusters et inversement. Le choix du nœud maître sera fait en désignant par exemple le nœud avec l'énergie la plus importante, pour augmenter la vie du réseau.

#### ➤ La tolérance aux fautes

Dans les réseaux de capteurs sans fil, un ou plusieurs capteurs peuvent ne pas fonctionner correctement. En effet les capteurs sont des entités sensibles aux altérations d'états comme des phénomènes climatiques (humidité, chaleur, électromagnétisme) ou du fait d'une batterie faible.

Dans ce cas de figure, le réseau doit être capable de détecter ce type d'erreur et d'y remédier, en cherchant par exemple à modifier ses tables de routage pour trouver un autre chemin permettant de transmettre l'information et de maintenir le réseau toujours

opérationnel. De la même manière, les capteurs doivent pouvoir détecter des capteurs défaillants qui envoient des informations erronées du fait de leur état.

### ➤ **Mise à l'échelle**

Le nombre de capteurs utilisés dans les réseaux de capteurs sans fil peut varier de quelques entités à plusieurs dizaines de milliers. C'est d'ailleurs la principale utilité des réseaux de capteurs qui doivent pouvoir s'auto organiser à une grande échelle et être efficace quelque soit le nombre. Pour cela les protocoles des réseaux de capteurs sans fil doivent être capables de fonctionner et de s'adapter selon le nombre de nœuds.

### ➤ **Une énergie limitée**

Les capteurs ont une durée de vie limitée par ce qu'ils sont équipés de batteries avec une énergie limitée. De plus, les réseaux de capteurs sans fil quand ils sont déployés, le sont souvent dans des zones difficiles d'accès pour l'homme. Il est donc difficile de pouvoir changer les batteries des capteurs. Si le nombre des capteurs dépasse la centaine d'entités, il est encore plus difficile d'intervenir pour trouver le capteur défaillant et charger sa batterie. Les capteurs sont en général déployés pour ne plus être modifiés.

La consommation de l'énergie des réseaux de capteurs sans fil doit être la plus préservée possible. Dans ce but, les capteurs actuels ont des périodes de veille durant leur temps d'inactivité pour préserver leur batterie.

Il est fortement nécessaire de limiter le nombre de communications entre capteurs et si possible le nombre de calculs car les communications sont les actions les plus coûteuses en termes d'énergie. Les calculs le sont, mais dans une moindre importance.

### ➤ **Une faible puissance de calcul**

Malgré les progrès récents dans la fabrication de capteurs de plus en plus puissants, les nœuds capteurs sont limités en puissance de calcul. Cette faible puissance ne permet pas d'utiliser des algorithmes complexes dans les réseaux de capteurs sans fil.

### ➤ **Les contraintes matérielles**

La principale contrainte matérielle est la taille du capteur qui doit être assez réduite, ainsi que la résistance du capteur aux susceptibles cassures et accidents.

### ➤ **La sécurité**

La sécurité physique des nœuds capteurs ainsi que la sécurité des communications inter-capteurs sont des caractéristiques très intéressantes dans le domaine des RCSF.

### ➤ Une densité importante des nœuds

Les RCSF sont caractérisés par leur forte densité [22]. Cette densité peut atteindre, selon le type d'application, 20 nœuds/m<sup>3</sup> [23].

## 6. La pile protocolaire des RCSF

Les réseaux de capteurs sans fil sont des réseaux qui se basent sur un support de communication sans fil. Cette solution de communication doit être couplée avec les contraintes matérielles et énergétiques des capteurs.

Les standards de communication utilisés pour les réseaux filaires et pour les autres types de réseaux sans fil (WiFi, WiMAX, Bluetooth...etc.) ne sont plus adaptés aux RCSF car les mécanismes de communication et de sécurité développés pour ces standards ne prennent pas en considération les ressources limitées des capteurs au niveau des couches protocolaires. Ainsi, il était primordial de développer une nouvelle norme de communication garantissant, à la fois, une fiabilité maximale et un respect aux limites des capteurs essentiellement celle liée à la consommation d'énergie. Cependant, la majorité des articles scientifiques, qui traitent la thématique des RCSF, se basent sur la pile protocolaire qui a été proposée par Akyildiz et al. [23], le rôle de ce modèle consiste à standardiser la communication entre les composants du réseau afin que différents constructeurs puissent mettre au point des produits (logiciels ou matériels) compatibles. Ce modèle (*Figure 1.4*) comprend 5 couches qui ont les mêmes fonctions que celles du modèle OSI (*Open System Interconnection*), ainsi que 3 couches pour la gestion d'énergie (*Power Management Plane*), la gestion de la mobilité (*Mobility Management Plane*), et la gestion des tâches (*Task Management Plane*). Le but d'un système en couches est de séparer le problème en différentes parties (les couches) selon leur niveau d'abstraction. Chaque couche du modèle communique avec une couche adjacente (celle du dessus ou celle du dessous). Chaque couche utilise ainsi les services des couches inférieures et en fournit à celle de niveau supérieur.

La couche application assure l'interface avec les applications et présente le niveau le plus proche des utilisateurs, géré directement par les logiciels. La couche transport aide à gérer le flux de données si le réseau de capteurs l'exige. Elle permet de diviser les données issues de la couche application en segments pour les délivrer, ainsi elle réordonne et rassemble les segments venus de la couche réseau avant de les envoyer à la couche application. La couche réseau prend soin de router les données fournies par la couche transport. Dans cette couche, les métriques considérées par les chercheurs pour déterminer la route la plus

## Chapitre 1 : Généralités sur les RCSF

---

optimisée dans les réseaux RCSF sont : L'énergie nécessaire pour transmettre le paquet d'une manière fiable, et L'énergie disponible dans chaque nœud capteur. Le type d'adressage le plus utilisé dans les RCSF est l'adressage géographique. Le protocole MAC (Media Access Control) de la couche liaison est responsable du multiplexage des données, du contrôle d'erreurs, de l'accès au media, ...etc. Elle assure la liaison point à point et multipoint dans un réseau de communication. La couche physique assure la sélection de fréquence, la génération de la fréquence porteuse, la détection du signal, la modulation/démodulation et le cryptage/décryptage des informations.

En plus de ces cinq couches, la pile protocolaire des réseaux de capteurs comporte trois niveaux. Ces niveaux sont responsables du contrôle de l'énergie consommée, des mouvements des nœuds et de la distribution des tâches à travers toute la pile protocolaire, ils permettent aux capteurs de coordonner leurs tâches et minimiser la consommation d'énergie.

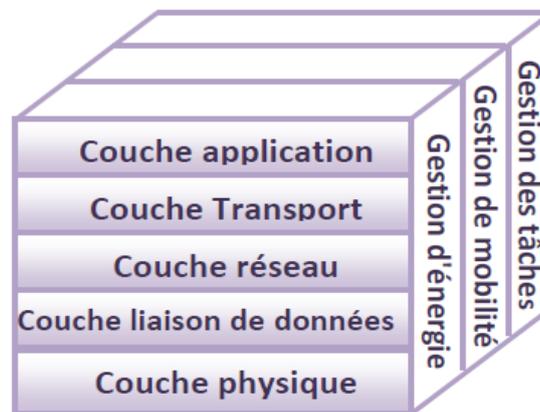


Figure 1.4: Modèle en couches pour la communication dans les RCSF [23].

D'abord, La gestion d'énergie adopte quelques mécanismes pour contrôler et essayer d'optimiser l'énergie consommée par les capteurs. Ainsi, lorsqu'un capteur atteint un niveau critique d'énergie il informe ses voisins de son incapacité à participer au routage. Ensuite, la gestion de la mobilité détecte et enregistre les mouvements des nœuds capteurs afin de leur permettre de garder continuellement un chemin vers le nœud puits et de maintenir une image récente de la carte du réseau. En fin, le niveau de gestion des tâches : Lors d'une opération de capture dans une région donnée, les nœuds composant le réseau ne travaillent pas obligatoirement au même rythme, cela dépend essentiellement de la nature du capteur, de son niveau d'énergie et de la région dans laquelle il a été déployé, ce niveau doit donc assurer l'équilibrage et la distribution des tâches sur les différents nœuds concernés afin d'optimiser la consommation d'énergie.

### 7. Les systèmes d'exploitation dans les RCSF

Puisque les capteurs sont considérés comme des dispositifs à ressources limitées, l'implémentation des applications dans les RCSF nécessite des systèmes d'exploitation et des langages de programmation des capteurs bien spécifiques. Il existe plusieurs systèmes d'exploitation conçus pour les nœuds capteurs sans fil. Les rôles de chaque système d'exploitation sont d'assurer les services basiques suivant : l'abstraction de la partie matérielle, la gestion des tâches, la gestion de la mémoire, la gestion de puissance et la gestion des périphériques. TinyOS (tiny operating system) et Contiki sont parmi les systèmes d'exploitation les plus utilisés depuis une petite dizaine d'années [24].

**TinyOS** [25] est un système d'exploitation Open-Source, initialement développé au sein de l'université de Berkeley en Californie. TinyOS propose un modèle de programmation orienté évènement, sa conception a été entièrement réalisée en NesC, langage orienté composants s'inspirant du C classique. TinyOs incorpore une architecture basée sur une association de composants, qui réduit la taille du code nécessaire et fournit une plateforme flexible pour mettre en application de nouveaux protocoles de communication. Cela s'inscrit dans le respect des contraintes de mémoires qu'observent les capteurs, pourvus de ressources très limitées dues à leur miniaturisation.

Pour autant, la bibliothèque des composants de TinyOs est particulièrement complète, puisqu'on y retrouve des protocoles réseaux, des pilotes de capteurs, et des outils d'acquisition de données. Un programme s'exécutant sur TinyOs est constitué d'une sélection de composants systèmes et de composants développés spécifiquement pour l'application à laquelle il sera destiné (mesure de température, du taux d'humidité...). TinyOs s'appuie sur un fonctionnement évènementiel, c'est à dire qu'il ne devient actif qu'à l'apparition de certains évènements, par exemple l'arrivée d'un message radio. Le reste du temps, le capteur se trouve en état de veille, garantissant une durée de vie maximale connaissant les faibles ressources énergétiques des capteurs. Ce type de fonctionnement permet une meilleure adaptation à la nature aléatoire de la communication sans fil entre capteurs.

**Contiki** [26] est un système d'exploitation écrit en langage C, portable open source multitâche développé pour une série de plates-formes comprenant des microcontrôleurs tels que le TI MSP430 et Atmel AVR, qui sont utilisés dans les familles de Telos, de Tmote et de Mica. Contiki est spécialement conçu pour respecter les contraintes des RCSF, en particulier, celles qui sont liées aux limitations de l'espace mémoire (il en occupe environ 32 kilooctets de ROM et 4 kilooctets de RAM). Les processus Contiki utilisent le *protothreading* [27] ; un style de programmation qui présente un bon compromis entre la programmation

## Chapitre 1 : Généralités sur les RCSF

---

évènementielle et la programmation par multithreading. Comparé à TinyOS, qui est statiquement lié à la version compilée, Contiki permet de substituer des programmes et des pilotes (drivers) en temps réel et sans recompilation.

D'autres systèmes d'exploitation connus pour les réseaux de capteurs sont décrits dans la liste non exhaustive suivante :

**SOS** [28] : système d'exploitation développé par l'université de Los Angeles en Californie (UCLA) écrit en langage C et qui reprend le système de programmation évènementielle de TinyOS.

**Mantis OS** [29] : système d'exploitation dédié aux réseaux de capteurs développé par l'université du Colorado (USA) et écrit en langage C. Contrairement à TinyOs qui est basé sur un modèle de programmation évènementielle, Mantis OS s'articule autour d'un modèle commandé par l'exécution de processus.

**Nut/OS** [30] : Système d'exploitation multitâches pour systèmes embarqués avec une pile TCP/IP.

### 8. Classification des RCSF

Selon plusieurs critères bien spécifiques, comme le mode d'acquisition et de livraison des données au puits, la distance entre les nœuds capteurs et le puits, le modèle déployé dans le réseau, les capacités des nœuds du réseau, les RCSF peuvent être classés en plusieurs classes [19, 31, 32].

#### 7.1. Selon le mode d'acquisition et de livraison des données au puits

Dans les RCSF, le mode d'acquisition et de livraison des données au puits dépend de l'application et de ses exigences. Il existe quatre modes de communication pour les réseaux de capteurs :

**Le mode continu (time-driven)** : ou le mode périodique, les capteurs doivent périodiquement réveiller leurs émetteurs pour envoyer les données captées au puits (*Figure 1.5(b)*). Le type d'application visé concerne les applications de type "surveillance" où le but principal est d'avoir une information régulière de la zone surveillée. Un exemple d'application dans le monde agricole [33]. Les capteurs sont déployés dans un champ pour surveiller l'état du sol. Ils envoient périodiquement des informations sur l'état du sol pour optimiser les apports d'eau et de nutriments.

**Le mode événementiel (eventdriven):** les capteurs envoient leurs mesures vers le puits lorsqu'il y a un évènement qui se produit (*Figure 1.5(a)*). Ce type de modèle est recommandé pour les applications de surveillance d'évènements critiques où le but principal est l'obtention d'une information sur l'évènement le plus rapidement possible. Par exemple, pour une application de gestion de déchets et de conteneurs de verres [34]. Quand les capteurs détectent un dépassement d'un seuil de remplissage d'un conteneur, ils envoient cet évènement détecté vers le nœud puits. Ce dernier va informer les autorités de la commune pour envoyer un camion pour décharger ce conteneur.

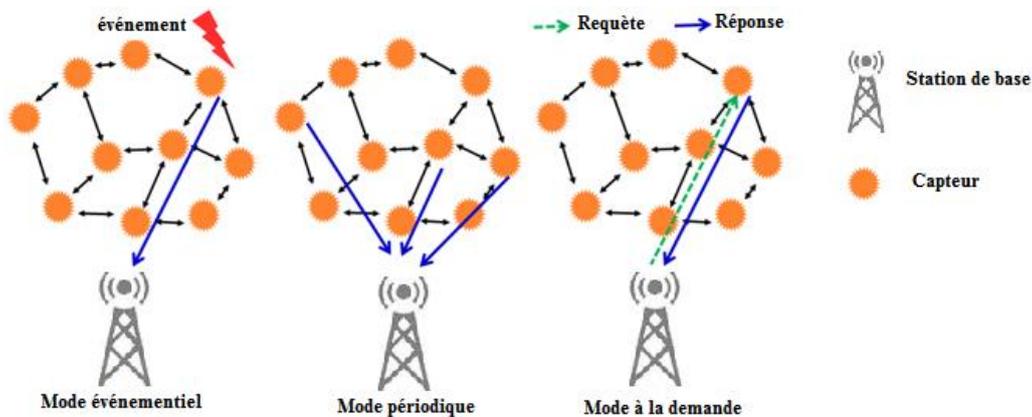


Figure 1.5 : Les différents modes de communications pour les RCSF [34].

**Le mode à base de requête (query-driven) :** Dans le modèle à la demande, les capteurs mesurent des phénomènes et stockent ces mesures dans leur mémoire. Ils envoient ces mesures seulement lorsqu'ils reçoivent des requêtes de la station de base (*Figure 1.5(c)*). Un exemple typique de ces applications est la télé relève de compteur d'eau [34]. Dans ces applications des capteurs sont déployés avec les compteurs d'eau pour surveiller la consommation des clients. Le nœud puits peut être présenté comme un technicien ayant un appareil de collecte de donnée. Une fois devant l'immeuble où il doit collecter des informations, le technicien envoie avec son nœud puits une demande de collecte d'information. Quand la demande est reçue par le(s) nœud(s) destination, ce(s) dernier(s) va (vont) répondre en envoyant les informations requises par le nœud puits.

Le modèle hybride est une combinaison des trois modèles précédents.

### 7.2. Selon la distance entre les nœuds capteurs et le puits

Dans cette classification, on distingue trois modes de communication ; un seul saut, multi-sauts et hybride. Dans un RCSF à un seul saut, les nœuds capteurs envoient leurs

## **Chapitre 1 : Généralités sur les RCSF**

---

données captées directement au puits sans passer par aucun autre nœud intermédiaire. Cependant, dans les RCSF multi-sauts, la distance entre les quelques nœuds capteurs et la station de base dépasse leur portée maximale. De ce fait, chaque nœud joue à la fois le rôle d'initiateur de données et de routeur également. En multi-sauts, chaque nœud capteur envoie ses informations au nœud voisin, ce dernier retransmet les informations à son nœud voisin jusqu'à ce que les informations arrivent à la station de base. Dans les RCSF hybride, les nœuds capteurs envoient leurs données captées avec une combinaison des deux méthodes précédentes.

### **7.3. Selon le modèle de mobilité dans le réseau**

Cette classification consiste en une combinaison entre la mobilité des nœuds capteurs et celle de la station de base. Par cette combinaison, nous pouvons distinguer deux grandes catégories de réseaux: réseaux statiques et réseaux dynamiques ou mobiles.

Un réseau de capteurs statique ne considère aucune mobilité des nœuds, ni de la station de base. Cela exige une mise en place initiale des infrastructures de communication afin de créer le chemin entre la station de base et les capteurs.

Dans un réseau de capteurs mobile les nœuds, ou la station de base sont mobiles, Comment un nœud peut-il être mobile ? Il peut être attaché à un robot, à un être humain, véhicule, etc. L'objectif des réseaux de capteurs mobiles est de recueillir plus d'informations sur un environnement en utilisant moins de nœuds capteurs, et de permettre au réseau d'organiser lui-même ses nœuds. En outre, il devient capable de déplacer ses capteurs dynamiquement selon les changements environnementaux, ce qui le rend adaptable à l'évolution de son environnement.

On peut par exemple avoir un réseau constitué d'un ensemble de nœuds capteurs mobiles et d'une station de base fixe. Le but de tels réseaux est la plupart du temps l'exploration de zones inaccessibles ou dangereuses. Un autre exemple est un réseau constitué de capteurs fixes servant à la surveillance d'occurrence d'événements sur une zone géographique et d'une station de base fixe.

### **7.4. Selon les capacités des nœuds du réseau**

Dans cette catégorie, on distingue deux types de réseaux : les réseaux homogènes des hétérogènes [19]. Dans un réseau de capteurs homogène, tous les nœuds ont les mêmes capacités du point de vue énergie, calcul et stockage. Alors que, dans un réseau de capteurs

hétérogène il y a quelques nœuds sophistiqués qui ont plus de capacité de traitement, capacité énergétique et de communication que les nœuds normaux.

### **8. Les technologies de transmission dans les RCSF**

Dans un réseau de capteurs, les nœuds sont reliés par une architecture sans fil. Pour permettre des opérations sur ces réseaux dans le monde entier, le média de transmission doit être standardisé. On utilise le plus souvent l'infrarouge [35], le Bluetooth [36], et les communications radio Zig Bee [37].

#### **➤ Les liaisons infrarouges (IEEE 802.11)**

L'infrarouge utilise une longueur d'onde de 850nm à 950nm pour le signal. Cette longueur d'onde est proche de la bande du visible par l'homme. C'est une lumière infrarouge diffusée. Les liaisons infrarouges permettent de créer des liaisons sans fil de quelques mètres avec des débits pouvant monter à quelques mégabits par seconde. Cette technologie est largement utilisée dans la domotique (télécommandes), elle souffre toutefois des perturbations dues aux interférences lumineuses.

#### **➤ La technologie Bluetooth (IEEE 802.15.1)**

Le Bluetooth est un standard de communication permettant l'échange bidirectionnel de données à très courte distance en utilisant des ondes radio, son objectif est de simplifier les connexions entre les appareils électroniques en supprimant des liaisons filaires. Cette technologie a été lancée par Ericsson en 1994, proposant un débit théorique de 1 Mbps lui permettant une transmissions de la voix, des données et des images [38], d'une portée maximale d'une trentaine de mètres. Bluetooth est une technologie peu onéreuse, grâce à sa forte intégration sur une puce unique de 9 mm sur 9 mm [39] ; Elle présente également l'avantage de fonctionner sur des appareils à faible puissance d'où une faible consommation d'énergie [38].

#### **➤ La technologie Zig Bee (IEEE 802.15.4)**

Zig Bee est un réseau WPAN de faible débit, faible consommation énergétique, de courte portée et d'une bonne souplesse d'installation [40], ce qui la rend particulièrement adaptée pour être directement intégrée dans de petits appareils électroniques (capteurs, appareils électroménagers...). Les réseaux ZigBee permettent d'offrir des débits jusqu'à 250 Kbits/s

## **Chapitre 1 : Généralités sur les RCSF**

---

dans la bande classique des 2,4GHz. Les RCSF constituent une des applications que cette norme peut couvrir [41].

### **9. Conclusion**

Ce chapitre nous a permis de voir le fonctionnement général d'un nœud capteur sans fil. Ce dernier est composé de différents modules matériels (unité de traitement, unité de captage, unité de transmission et l'énergie) et des logiciels qui les commandent.

Dans ce chapitre, nous avons présenté des concepts généraux, ils s'avèrent différents des réseaux sans fil conventionnels dû à un ensemble de caractéristiques ; la faible puissance de calcul, le facteur d'énergie limitée, les contraintes matérielles, ces spécificités dévoilent un large domaine de recherche dans les RCSF. Le chapitre suivant est consacré pour introduire l'auto-adaptation, le routage, et applications dans les RCSF.

## **Chapitre 2**

### **Etat de l'art : Auto-adaptation, Routage, et Applications dans les RCSF**

### 1. Introduction

Comme nous l'avons souligné dans le chapitre précédent, les réseaux de capteurs jouent un rôle très important. Donc leur auto-adaptation est plus que nécessaire pour assurer la réception des données par la station de base via les différents protocoles de routage. Ce type de réseaux est souvent déployé pour accomplir des missions critiques ou pour surveiller nos activités et applications quotidiennes.

Dans ce chapitre, nous présentons les concepts, les défis et les objectifs d'auto-adaptation sur lesquels s'est focalisée particulièrement notre recherche, nous présentons aussi quelques solutions d'auto-adaptation proposées dans la littérature.

Nous nous intéressons particulièrement aux protocoles de routage hiérarchiques, nous présentons en premier lieu les principales classes des protocoles de routage dans les RCSF, dans la seconde partie nous nous intéressons aux principales caractéristiques d'un protocole de routage hiérarchique, et enfin nous examinons les différentes approches proposées pour le routage hiérarchique de données sur les réseaux de capteurs sans fil (exemples de protocoles de routage hiérarchiques).

À la fin de ce chapitre, nous expliquons quelques exemples d'application dans les RCSF. En effet, les spécificités des RCSF ont permis de supporter des applications qui sont inappropriées pour d'autres types de réseaux sans fil.

### 2. Auto-adaptation

Dans un RCSF, les nœuds sont déployés soit d'une manière aléatoire ou fixés par un humain ou un robot. Ainsi, un nœud capteur doit avoir des capacités d'une part, pour s'auto-adapter dans le réseau, et d'autre part pour collaborer avec les autres nœuds dans le but de reconfigurer dynamiquement le réseau en cas de changement de topologie du réseau [42].

#### 2.1. Définition et rôle

Un système composé de plusieurs objets, est dit adapté s'il possède une structure et un ensemble de procédures [43]. La structure vise à créer une relation entre les objets pour faciliter la communication entre eux. L'ensemble des procédures a pour rôle de maintenir la structure et l'utilisation de celle-ci pour répondre à des besoins bien déterminés [44].

L'auto-adaptation est utilisée dans de nombreuses branches de la science [20, 45]. Dans les RCSF, un système est dit auto-adapté s'il s'adapte de façon autonome, sans l'intervention

d'un agent extérieur à contrôle centralisé. Ainsi, les capteurs doivent interagir localement avec d'autres nœuds d'une façon distribuée. Un système auto-adapté est basé sur:

- **L'interaction locale:** le comportement émergent du système est plus que des interactions simples entre ses différents éléments sans contrôle externe.
- **L'émergence d'une structure globale:** l'objectif principal de l'auto-adaptation est de produire des structures stables qui sont construites dans un temps cohérent limité.
- **Adaptation à l'environnement et à la robustesse:** la structure émergente d'un système doit être adaptée à l'environnement et doit répondre aux changements locaux.
- **Grande échelle:** cette propriété est le résultat d'un contrôle central et d'interactions internes [46].

### 2.2. Concepts et défis

L'infrastructure limitée ou l'absence d'infrastructure et la notion décentralisée dans les réseaux de capteurs sans-fil, pose le problème d'auto-adaptation. Ce concept d'auto-adaptation est un grand domaine d'intérêt pour la communication, les réseaux informatiques et les RCSF. Les nœuds capteurs, les cluster-heads et les stations de base interagissent directement entre eux de manière distribuée pour atteindre un but commun.

Par rapport au scénario précédent et au concept d'auto-adaptation, nous avons identifié les défis suivants:

#### ➤ **Comment choisir la meilleure auto-adaptation ?**

Comme déjà indiqué, les réseaux de capteurs sont caractérisés par la pénurie de ressources. L'échange de messages ainsi que les opérations nécessaires pour atteindre la nouvelle auto-adaptation doivent être minimisés pour conserver les ressources du réseau et assurer le bon fonctionnement. En conséquence, la nouvelle auto-adaptation choisie dans les RCSF devrait être la plus appropriée.

#### ➤ **Comment traiter les objectifs contradictoires ?**

Dans le processus de détermination de la meilleure auto-adaptation, on pourrait se concentrer sur une seule dimension, tout en sacrifiant les autres. Par exemple, l'économie d'énergie et le coût de l'auto-adaptation peuvent être considérés comme des objectifs contradictoires. Ensuite, pour résoudre ce genre de situations, nous devons considérer plus de critères dans le processus de sélection. Il est nécessaire de trouver un équilibre approprié entre ces objectifs.

### 2.3. Les objectifs de l'auto-adaptation

Les principaux objectifs de l'auto-adaptation sont :

#### ➤ **Minimiser la consommation énergétique**

La mise en état d'endormissement de certains nœuds permet de conserver de l'énergie dans le réseau [47]. L'auto-adaptation dans le cas de changement des protocoles de routage augmente la durée de vie de réseau. D'autres propositions d'auto-adaptation permettent de définir une puissance de transmission pour chacun des nœuds afin de réduire le coût énergétique total du réseau [34, 48, 49].

#### ➤ **Améliorer les performances du réseau**

Plusieurs travaux ont montré que la présence de l'auto adaptation améliore les performances du réseau (en terme de charge du réseau, de consommation énergétique, de délais de transmission, nombre des nœuds activés, ...etc.) et facilitent sa gestion et son extensibilité. Pour cela, nous avons cherché à minimiser l'échange de messages dans le RCSF, et aussi de réduire le nombre de retransmissions, impactant la consommation d'énergie du réseau.

#### ➤ **Partager et gérer les ressources disponibles**

Un protocole d'auto-adaptation doit faciliter la gestion des ressources dans le RCSF. En effet, chaque protocole de routage doit être plus efficace quand il est déployé dans un réseau. Par conséquent, la présence des nœuds riches en ressources énergétique permet de répartir les informations de routage entre les nœuds capteurs tout en prenant compte leurs capacités de stockage limitées.

### 2.4. Solutions proposées dans la littérature

Selon la littérature, il y a plusieurs travaux sur les mécanismes d'auto-adaptation dans les RCSF. Nous pouvons citer par exemple les travaux rapportés dans [50, 51, 52, 53, 54, 55]. Cependant, beaucoup d'entre eux [50, 51, 52] se concentrent sur la façon de réaliser cette adaptation sur le réseau au lieu de chercher une configuration appropriée de l'ensemble du système comme nous le proposons dans ce mémoire. Dans [50], un mécanisme d'adaptation basé sur un algorithme permettant à un réseau de capteurs de s'adapter en fonction du contexte actuel et des critères de réseau prédéfinis qui incluent la densité de couverture, la tolérance aux pannes, la durée de vie du système et ainsi de suite. L'idée clé de leur travail est basée sur le calcul de la couverture des nœuds, afin de permettre aux réseaux d'être adaptatifs. Dans le travail consigné dans [51], l'énergie est l'exigence la plus importante dans l'adaptation des RCSF. Les auteurs décrivent une approche pour adapter le comportement

## Chapitre 2 : Auto-adaptation, routage, et applications dans les RCSF

---

des applications en cours en interceptant les appels effectués aux services du système d'exploitation et en modifiant leurs effets au moment de l'exécution.

La personnalisation est obtenue grâce à de petits fragments de byte code interprété, appelés adaptateurs, injectés dans le réseau par la station de base. Ils se concentrent sur la façon de reconfigurer le système en utilisant un mécanisme d'injection de code. Dans [52], une réflexion middleware orienté service pour les RCSF est proposée. Le middleware fournit une couche d'abstraction entre les applications et l'infrastructure de réseau sous-jacente. Il conserve l'équilibre entre les exigences de QoS de l'application et la durée de vie du réseau en surveillant les états d'exécution du réseau et de l'application et en effectuant une adaptation réseau chaque fois qu'il le faut.

Un autre point intéressant d'auto-adaptation dans les RCSF est discuté en [56], où les auteurs ont développé une technique pour conduire des systèmes électriques alimentés par batterie dans de nombreuses applications en fonction du sujet intéressant pour la conception électronique sans batterie. Les auteurs dans [57], se concentrent sur l'idée d'auto-adaptation dans le «Internet of Things» (IoT), ils ne considèrent pas seulement comment les nouveaux périphériques pourraient conduire à des chiffrements plus sophistiqués et robustes dans des environnements à ressources limitées, mais aussi la façon dont les nouveaux dispositifs peuvent rendre ces chiffres plus résistants aux attaques telles que l'analyse de puissance différentielle (DPA).

Dans [53, 54], les auteurs essayent d'aborder l'adaptation en tenant compte des objectifs contradictoires. Cependant, ils utilisent un mécanisme d'adaptation biologique pour reconfigurer le système en ce qui concerne uniquement la latence, le coût et le taux de réussite. Dans notre travail, les dimensions sont plus génériques et couvrent plus d'objectifs. Dans [55], les auteurs se concentrent sur la façon de trouver la configuration la plus appropriée pour adapter le système à une nouvelle situation contextuelle. Ils présentent une approche pour permettre la sélection d'une nouvelle configuration dans les RCSF homogènes lorsque plusieurs configurations satisfont le contexte actuel. En particulier, ils appliquent des optimisations multi-objectives pour faire face à des objectifs contradictoires. Les optimisations sont basées sur la consommation de ressources, la QoS offerte par le réseau et le coût de reconfiguration. Ce sont des dimensions importantes à considérer dans la configuration à atteindre [58].

### 3. Routage

Sur la base des résultats des chercheurs dans [59, 60, 61, 62, 63, 64, 65], on remarque que la majorité des protocoles de routage dans les RCSF sont des protocoles basés sur la notion de cluster (hiérarchiques) quelque soit leur type proactif, réactif ou hybride, ces protocoles sont les meilleurs en terme d'énergie, durée de vie de réseau, évolutivité, ...etc.

L'objectif fondamental du routage hiérarchique [66, 67] est de réduire la consommation d'énergie lors de l'établissement de routes, afin de diminuer le nombre de messages transmis à la destination, le RCSF construit une communication multi-sauts au sein d'un cluster et effectue le concept d'agrégation et la fusion des données. En effet, la plupart des formations de clusters sont fondées sur la réserve d'énergie des capteurs et sur les capteurs qui sont à proximité de cluster-head.

La technique de clusterisation (hiérarchique) permet de regrouper le réseau en sous ensemble dits clusters pour préserver l'énergie et faciliter la gestion du réseau où chaque cluster a un super nœud appelé cluster-head (CH), dont le choix du cluster-head se base généralement sur les trois critères à savoir l'identification, l'énergie résiduelle, et la distance vers la station de base. Chaque cluster-head gère un cluster et communique avec d'autres cluster-heads.

#### 3.1. Les principales classes des protocoles de routage dans les RCSF

Les protocoles de routage proposés dans les RCSF peuvent être classés suivant des critères différents. *La figure 2.1* résume cette classification qui se base sur quatre critères : la topologie du réseau, mode de fonctionnement du protocole, les paradigmes de communication et selon le mode d'établissement des chemins.

##### 3.1.1. Selon les paradigmes de communication

Globalement, il existe trois paradigmes de communication dans les RCSF [67] : le paradigme basé-localisation, le paradigme centré-données, et le paradigme centré-nœuds.

###### ➤ Basé-localisation

La plupart des protocoles de routage pour les RCSF nécessitent des informations de localisation pour calculer la distance entre deux nœuds particuliers afin de pouvoir estimer la consommation d'énergie. En raison de l'absence de plan d'adressage pour les réseaux de capteurs comme les adresses IP, les informations de localisation peuvent être utilisées pour acheminer des données. [68].

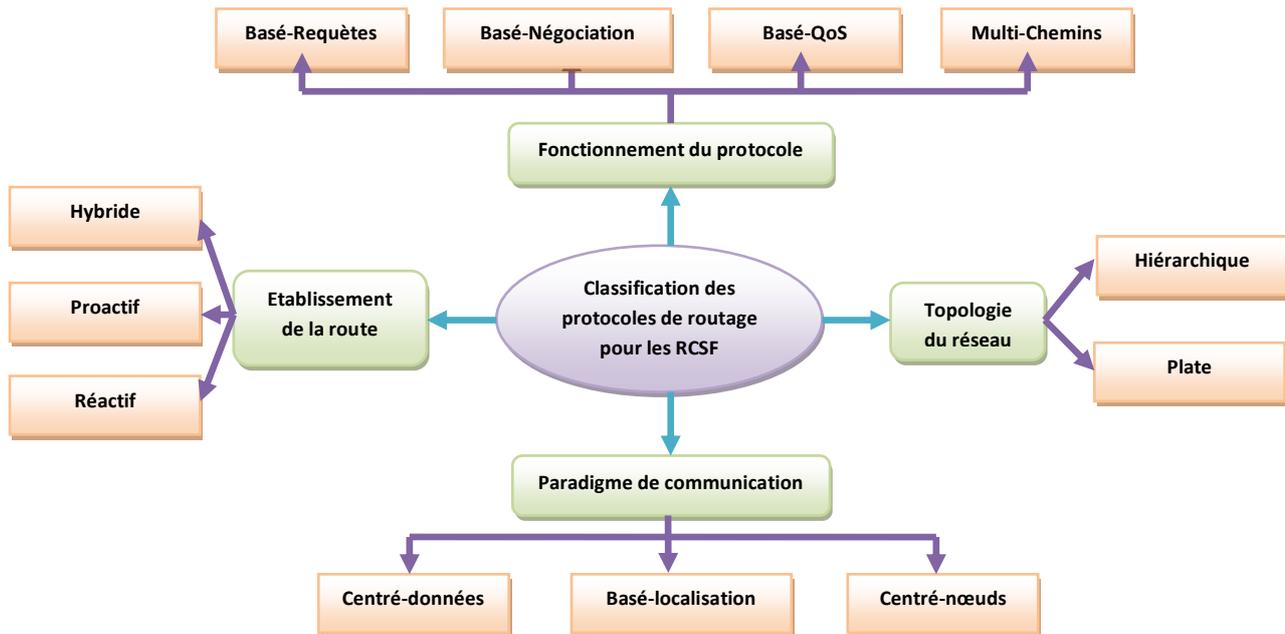


Figure 2.1. Classification des protocoles de routage pour les RCSF.

### ➤ Centré-données

Dans un RCSF, les données sont plus importantes que le nœud lui-même, ce qui rend son identification inutile. Dans le paradigme centré-données, le routage ne se fait pas en fonction d'une adresse de destination, mais l'identification de la communication se fait par les données disponibles au niveau des capteurs. Ainsi, le système peut être vu comme une base de données distribuée, où les données seront propagées de proche en proche pour arriver à la station de base.

### ➤ Centré-nœuds

La communication basé sur ce mode de routage est le paradigme le plus populaire actuellement pour les RCSF où les destinations sont spécifiées en fonction des adresses numériques (ou identifiants) des nœuds [69].

### 3.1.2. Selon l'établissement de la route

L'établissement de la route détermine la manière de création des chemins pendant le routage dans le réseau. Nous distinguons trois catégories : les protocoles proactifs, réactifs, et hybrides.

### ➤ Proactif

Ce type des protocoles établissent au préalable les meilleures routes pour chaque nœud capteur vers l'ensemble des destinations possibles dans le RCSF. Grâce à une technique périodique de construction des tables de routage par l'échange des messages, ces protocoles maintiennent une vision globale du réseau. L'inconvénient majeur est la sauvegarde des routes même si elles ne sont pas utilisées.

### ➤ Réactif

Dans le cas d'un protocole réactif s'il y a des changements soudains détectés dans l'environnement de captage, ces protocoles assurent une réaction instantanée aux changements. Ce type de protocole est utilisé dans les applications dites à contraintes temporelles [67].

### ➤ Hybride

Les protocoles de routage hybrides combinent entre les deux caractéristiques des protocoles proactifs en cas d'établissement de la route et réactifs au delà de la zone de voisinage lorsqu'un capteur doit transmettre des données à un autre capteur [70].

### 3.1.3. Selon le fonctionnement du protocole

Selon ce critère, les protocoles de routage peuvent être classifiés en quatre classes : routage basé sur la négociation, routage multi-chemins, routage basé sur la qualité de service, et routage basé sur les requêtes.

#### ➤ Basé-QoS

Les protocoles basés sur ce modèle de routage doivent équilibrer entre la qualité de données et la consommation d'énergie. Ce type de routage est très recommandés pour les applications de surveillance afin d'assurer certaines métriques de QoS, par exemple, énergie, retard, largeur de bande passante, ...etc.

#### ➤ Basé-Négociation

L'objectif de ce type de routage élimine la redondance des données et garantit que seuls les informations utiles seront transmises par la suppression des informations en double et l'empêchement des données redondantes d'être envoyées vers la station de base ou au nœud capteurs suivant. L'idée de base est la négociation par une série de messages avant chaque transmission de données [71].

### ➤ Basé-Requête

Pour interroger les capteurs dans ce type de routage, la station de base génère des requêtes qui sont exprimées soit par un langage comme SQL (Structured Query Language) ou bien par un schéma valeur-attribut. Les nœuds émetteurs envoient les données demandés à travers le chemin inverse de la requête.

### ➤ Basé sur multi-chemins

Les protocoles de routage basé sur le concept multi-chemins envoient leurs données sur un ensemble de chemins menant d'un nœud source à une destination afin d'augmenter la performance des communications dans le réseau. Cette solution est efficace en cas de détection de dégradation ou de rupture des chemins actifs dans les RCSF multi-sauts.

### 3.1.4. Selon la topologie du réseau

En général, le routage dans les réseaux de capteurs peut être classé, selon la structure du réseau, en en deux classes [23], routage à plat et routage hiérarchique.

### ➤ Le routage à plat

Dans cette classe de routage, les nœuds capteurs ont typiquement les mêmes fonctionnalités et rôles, donc ils sont tous dans le même niveau de privilège. Ainsi, La transmission des données de la part d'un capteur dépend de sa position dans le réseau, c'est-à-dire que si un nœud capteur veut envoyer un message vers la station de base et que celle-ci est en dehors de sa portée radio, pour atteindre l'objectif de l'arrivée du message à la station de base, il envoie son message à un nœud intermédiaire (dans sa portée radio) et la même procédure se répète récursivement.

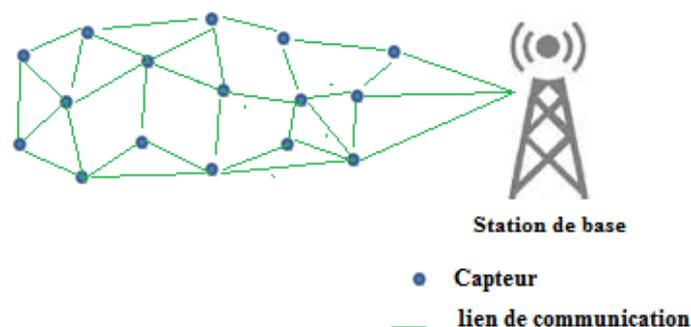


Figure 2.2. Routage à plat dans un RCSF.

L'avantage de cette topologie (*Figure 2.2*) est la tolérance aux pannes; si un nœud intermédiaire tombe en panne, la communication entre l'émetteur et le récepteur demeure possible s'il existe un chemin alternatif. Par conséquent, multiples routes menant vers la station de base donne le risque d'épuiser rapidement les réserves énergétiques des capteurs où chaque nœud transmet les paquets reçus à ses voisins par diffusion même si ces derniers ont déjà reçu ces paquets d'une autre source ; les données seront diffusées de la même façon jusqu'à ce que le réseau soit inondé, c'est la technique du FLOODING (inondation) [72].

### ➤ Le routage hiérarchique

Le routage hiérarchique initialement proposé dans les réseaux câblés, le principe est de restructurer le réseau en groupes dits clusters dont chacun est vu comme un sous réseau représenté par un chef de groupe (Cluster-Head) qui relie les membres de son groupe à la station de base. Les cluster-heads disposent d'une ressource énergétique importante par rapport aux autres, ils peuvent être utilisés pour traiter et envoyer les informations tandis que les nœuds à faible énergie peuvent être utilisés pour exécuter la tâche de la capture d'information dans la proximité de la cible [73]. Ainsi, La communication entre le chef du cluster et ses membres peut être directe (en un seul saut) ou indirecte pour les nœuds distants (en multi-sauts). *La figure 2.3* présente un simple modèle de cette topologie.

Le routage hiérarchique est une façon de minimiser la consommation énergétique, autre avantage marquant de cette topologie optimise les problèmes due à la densité du réseau qui produit la surcharge de la station de base, Ceci aboutit à une prolongation de la durée de vie du réseau.

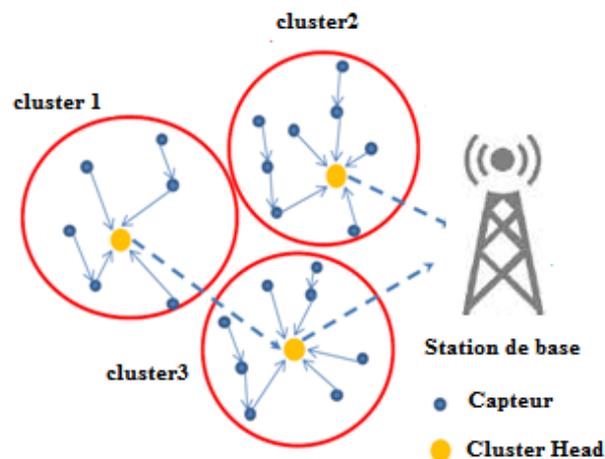


Figure 2.3. Routage hiérarchique dans un RCSF.

### 3.2. Caractéristiques d'un protocole hiérarchique

#### 3.2.1. Formation de clusters

Les algorithmes de formation des clusters représentent la première étape de la hiérarchisation. La première phase est l'algorithme d'élection du cluster-head qui peut constituer l'initialisation du cluster, où le réseau est divisé en groupes, et un représentant pour chaque groupe est élu. Le cluster-head consomme une quantité importante d'énergie par rapport aux autres nœuds, parce qu'il coordonne le fonctionnement des nœuds membres de son cluster et agrège les données. Pour éviter la dissipation d'énergie, une rotation de ce rôle de cluster-head est organisée par la station de base (centralisé), ou bien exécuté en coopération au niveau de chaque nœud du réseau (distribué). La rotation est effectuée en fonction de la consommation de l'énergie du cluster-head ou périodiquement [74].

Les algorithmes de formation des clusters peuvent être classés selon leur façon de contrôler

- la taille des clusters définie soit par le nombre maximum de nœuds dans le cluster, ou par la taille en km du cluster dans un contexte géographique,
- le nombre maximum de sauts entre chaque membre et son cluster-head,
- le nombre maximum de clusters dans le réseau,
- un critère particulier selon l'application considérée,
- des fonctions d'ajout ou de retrait de nœuds,
- les différents chemins de routage à partir et vers la station de base.

Chaque nœud dans ce type de RCSF a une adresse hiérarchique de topologie composée des numéros des groupes sur le chemin de la racine à la station de base. Une passerelle peut être atteinte de la racine par l'intermédiaire de plus d'une voie d'accès.

#### 3.2.2. La communication intra et inter-cluster

La majorité des solutions proposées pour les communications inter-clusters et intra-clusters [75] dans les réseaux de capteurs sans fil s'appuient soit sur une vue à plat ou bien sur une vue hiérarchisée. Des schémas de routage différents sont utilisés au sein des clusters et entre les clusters.

Dans la communication inter-cluster, les cluster-heads communiquent avec la station de base soit directement (un seul saut) (*Figure 2.4. (B)*), soit indirectement (plusieurs sauts) via

des nœuds appelés « Nœuds Gateway » (Figure 2.4. (A)). Ces nœuds généralement sont des CH. La communication multi-sauts permet d'augmenter l'évolutivité du réseau et de réduire la consommation d'énergie.

Dans le deuxième cas (intra-cluster), la communication entre le cluster-head et leurs membres peut se faire, soit en un seul saut, soit en plusieurs sauts. La figure 2.4. (A, et B) représente une communication intra-cluster en un seul saut (direct), les données sont envoyés directement au cluster-head par une transmission assez puissante, ceci engendre une consommation importante d'énergie si la distance entre l'émetteur et le récepteur est grande. Pour réduire cette consommation d'énergie, on utilise une communication intra-cluster en plusieurs sauts, dans ce mode de communication, chaque membre envoie ses données au plus proche membre de son cluster jusqu'à atteindre le cluster-head qui les transmet à la station de base soit directement ou indirectement.

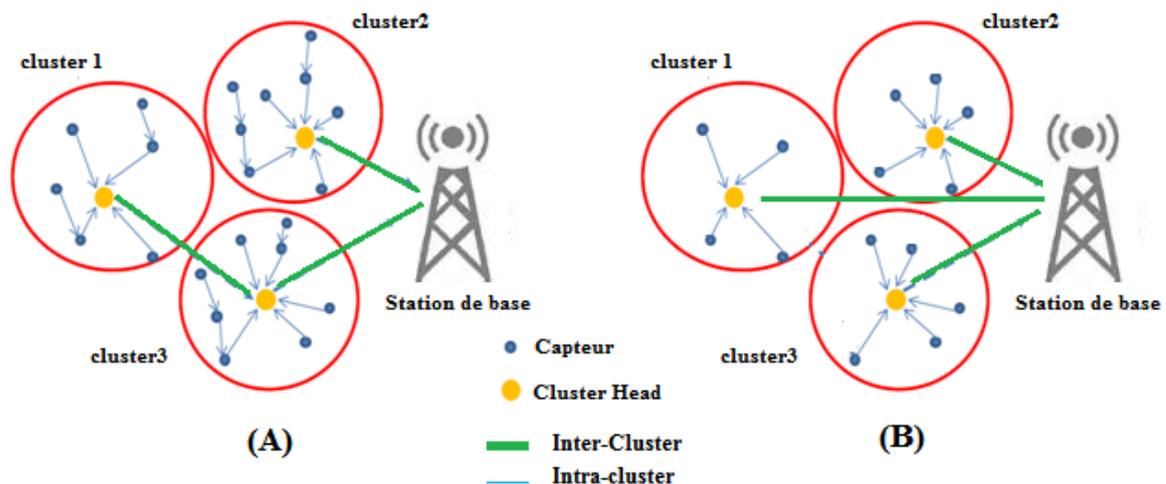


Figure 2.4. Architecture de communication intra et inter-cluster.

### 3.2.3. Le niveau d'agrégation de données

L'agrégation de données est une opération effectuée pour réduire la taille des données échangées entre les capteurs [76], i.e. le concept d'agrégation permet d'éviter la transmission d'informations redondantes et donc d'économiser la latence et l'énergie des capteurs [77], avec comme conséquence le prolongement de la durée de vie du réseau [78].

Les auteurs dans [79] proposent trois scénarios d'agrégation : (i) le Routage Sans Agrégation (RSA), (ii) le Routage avec Agrégation Partielle (RAP) et (iii) le Routage avec Agrégation Totale (RAT). Les résultats montrent que le RSA minimise les délais de

communication, le RAP réduit la consommation énergétique et le RAT prolonge la durée de vie des cluster-heads.

### 3.3. Les principaux protocoles hiérarchiques existants

Plusieurs exemples de protocoles hiérarchiques sont décrits dans la littérature, nous allons présenter les plus référencés:

#### ➤ **LEACH** (Low-Energy Adaptive Clustering Hierarchy)

Introduit par Heinzelman et Al. dans [4], est l'un des protocoles de routage hiérarchiques les plus populaires. Les auteurs proposent un algorithme de clustering distribué appelé LEACH (la formation des clusters et l'élection des cluster-heads se font au niveau des nœuds). LEACH suppose que les nœuds sont homogènes et que l'acheminement des paquets vers la station de base se fait en un seul saut via les cluster-heads. Le choix des CH se fait d'une façon aléatoire selon un algorithme spécifique d'élection basé sur une fonction de probabilité qui prend en compte différents critères comme l'énergie disponible des nœuds et attribue ce rôle aux différents nœuds selon la politique de gestion Round-Robin (tour à tour pour garantir une dissipation équitable d'énergie entre les nœuds).

Dans le but de réduire la quantité d'informations transmise à la station de base, les cluster-heads agrègent les données capturées par les nœuds membres appartenant à leur propre cluster, et envoient un paquet agrégé à la station de base [67].

#### ➤ **LEACH-C**

Les auteurs dans [5] proposent l'algorithme centralisé (LEACH-C) pour éviter les inconvénients du protocole LEACH. C'est un algorithme itératif, dans lequel la formation des groupes et l'élection des CH sont centralisés, c'est-à-dire dans chaque itération, la station de base affecte des rôles pour les différents nœuds du réseau. Ensuite, le fonctionnement continue de la même manière que pour LEACH.

#### ➤ **TEEN** (Threshold sensitive Energy Efficient sensor Network protocol)

Introduit dans [80] par A. Manjeshwar et Agrawal, le protocole TEEN est conçu pour les applications se basant sur le mode de communication événementiel (les applications dédiées pour détecter les changements soudains). L'architecture du réseau de capteurs est basée sur un groupement hiérarchique multi-niveaux où les nœuds les plus proches forment des clusters. TEEN est une hybridation du clustering hiérarchique et parmi les protocoles centrés-données. Il utilise la stratégie de LEACH pour la création des clusters. Après

l'établissement des clusters, le CH diffuse aux membres de son groupe le seuil maximal et minimal. Les nœuds capteurs surveillent leurs environnements continuellement. Une fois que la valeur des données dépasse le seuil maximal, ou diminue le seuil minimal le nœud transmet les données récoltées.

➤ **APTEEN** (AdaPtive Threshold sensitive Energy Efficient sensor Network protocol)

Proposé dans [61] par A. Manjeshwar et Agrawal, est une réforme de TEEN dédié à la fois aux applications se basant sur le mode de communication événementiel et à celles se basant sur le mode de communication périodique. La formation des clusters est faite par la station de base puis des valeurs des attributs sont diffusées par les cluster-heads vers tous les nœuds du réseau. Une agrégation de données est aussi réalisée par les Cluster-Head dans le but de minimiser l'énergie.

➤ **PEGASIS** (Power-Efficient GATHERing in Sensor Information Systems)

Proposé par Lindsey et Raghavendra en l'an 2002 dans [81]. L'idée de ce protocole qui est une amélioration de LEACH est que chaque nœud émet ou reçoit uniquement vers ou depuis des nœuds directement voisins à ce dernier. D'une autre façon, est un protocole basé sur la formation des chaînes des nœuds assurant la liaison entre voisins. La communication avec la station de base est assurée par un nœud représentatif d'une chaîne déjà présélectionnée [82]. Une procédure d'agrégation des informations est mise en œuvre au niveau des nœuds représentatifs afin de faciliter l'acheminement des données vers la destination.

➤ **H-PEGASIS** (Hierarchical-PEGASIS)

H-PEGASIS est une extension du protocole PEGASIS. Ainsi, L'objectif principal de ce protocole est de réduire les retards de transmission des paquets vers la station de base [83, 84]. Une chaîne est formée comme un arbre hiérarchique par tous les nœuds capteurs. Un nœud est choisi à un niveau particulier et ce nœud transmet des données aux nœuds dans le niveau supérieur de la hiérarchie jusqu'à ce qu'il arrive à la station de base. Cette méthode assure la transmission simultanée de données et réduit les délais dans PEGASIS.

➤ **HEED** (a Hybrid Energy-Efficient Distributed clustering)

Proposé par Younis et Fahmy dans [85], ce protocole réalise une meilleure distribution des clusters. La méthode avec laquelle sont sélectionnés les CH est l'un de ses aspects le plus important, où la construction des CH se fait selon les deux paramètres. Le premier dépend

de l'énergie résiduelle des nœuds, le deuxième dépend du coût des communications intra-cluster [86, 87, 88]. Ensuite, les CH correspondants allouent à chacun de leurs membres des temps de traitement différents afin de gérer les conflits lors de la transmission des données. Ils disposent de tâches de mise en veille et de réveille leurs permettant de gérer au mieux leurs ressources périodiquement [86]. L'envoi des données depuis un cluster-head vers la station de base se fait en sauts multiples.

### ➤ **HEEP** (Hybrid Energy Efficiency Protocol)

Proposé par Boubiche Djallel Eddine et BILAMI Azeddine en l'an 2011 dans [90], l'organisation des nœuds appartenant à la même grappe (cluster) sous forme d'une chaîne permettent d'améliorer et de réguler la dissipation d'énergie permettant de réduire la charge sur le cluster-head. En effet, les nœuds communiquent uniquement avec leurs proches voisins et non pas directement avec leur cluster-head, ce que économise d'avantage l'énergie et offre une meilleure utilisation de la bande passante. L'agrégation des données au niveau de chaque nœud dans la chaîne réduit la quantité de données échangées entre les nœuds et leur cluster-head, ce qui a pour effet de préserver les réserves énergétiques de ces derniers.

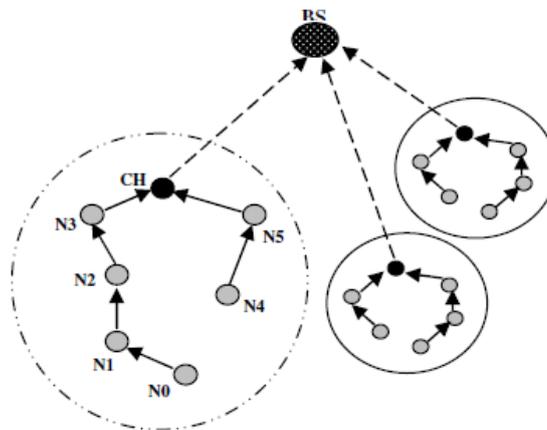


Figure 2.5. Architecture de communication du protocole HEEP [90].

La figure 2.5, montre comment les nœuds seront organisés à l'intérieur des clusters. Le nœud N0 transmet ses données à son proche voisin N1 qui à son tour agrège les données reçues avec les siennes et les transmet à son voisin jusqu'à atteindre le cluster-head qui les transmet directement à la station de base. Donc, dans cette nouvelle organisation (grappe à chaînes), tous les nœuds du cluster vont transmettre leurs données collectées à leurs cluster-

heads respectifs en se reliant à travers la chaîne, tandis que chaque CH doit recevoir les données collectées par les nœuds de même cluster.

### 4. Applications

Les réseaux de capteurs ont un large éventail d'applications diverses. Chaque application a ses propres spécifications et contraintes différentes des autres applications. Il n'y a pas de protocole réseau capable de répondre pleinement aux critères de toutes les applications. Par conséquent, les protocoles de routage conçus doivent calculer un chemin optimal et garantir la livraison précise des données à la station de base.

Les RCSF sont destinés à surveiller et observer des phénomènes physiques comme : la température, l'humidité, la pression, le taux de bruit, le mouvement des véhicules et la vitesse. Parmi les domaines d'application de ce type de réseau, nous pouvons citer les domaines militaire, environnemental, domestique, santé, sécurité, écologie, traçabilités, ...etc. [91, 92].

#### 4.1. e-application

e-application est une application électronique dont une des grandes caractéristiques est « d'utiliser des technologies application et l'Internet, pour améliorer la qualité de la surveillance en facilitant l'accès à des ressources et des services, ainsi que les échanges des informations et la collaboration à distance ». Dans les RCSF, Les utilisateurs communiquent avec l'environnement de captage via les modes de communication synchrones et asynchrones supportés par des outils web le plus souvent intégrés à des plateformes, cette technique offre la possibilité de véhiculer des interactions entre les différents utilisateurs et l'environnement suivi.

#### 4.2. Exemples d'applications

Aujourd'hui, il est unanimement reconnu que les Technologies de l'Information et de la Communication (TIC) jouent un rôle capital pour le soutien et l'assistance de la majorité des activités réalisées au sein des organisations [93].

Des exemples d'applications potentielles dans ces différents domaines sont exposés dans la *figure 2.6*.

##### ➤ Applications militaires

Les applications dans les RCSF sont très utiles pour l'espionnage militaire et la surveillance des champs de bataille. L'objectif d'utiliser e-applications dans ce domaine est

## Chapitre 2 : Auto-adaptation, routage, et applications dans les RCSF

la reconnaissance et la classification des cibles, par exemple. Le déploiement des capteurs sur les champs de bataille ou les endroits stratégique peut être réalisé manuellement ou aléatoirement ou bien les deux en même temps. Dans le premier cas, des troupes de reconnaissance peuvent marcher dans le champ de bataille pour positionner et cacher stratégiquement des capteurs. Comme les capteurs sont, par définition, des dispositifs très petits, ils seront à priori faciles à dissimuler. Dans le deuxième cas, lorsque les capteurs sont déployés aléatoirement, par exemple en larguant les capteurs depuis un avion pose le risque que des nœuds tombent dans une mauvaise position.

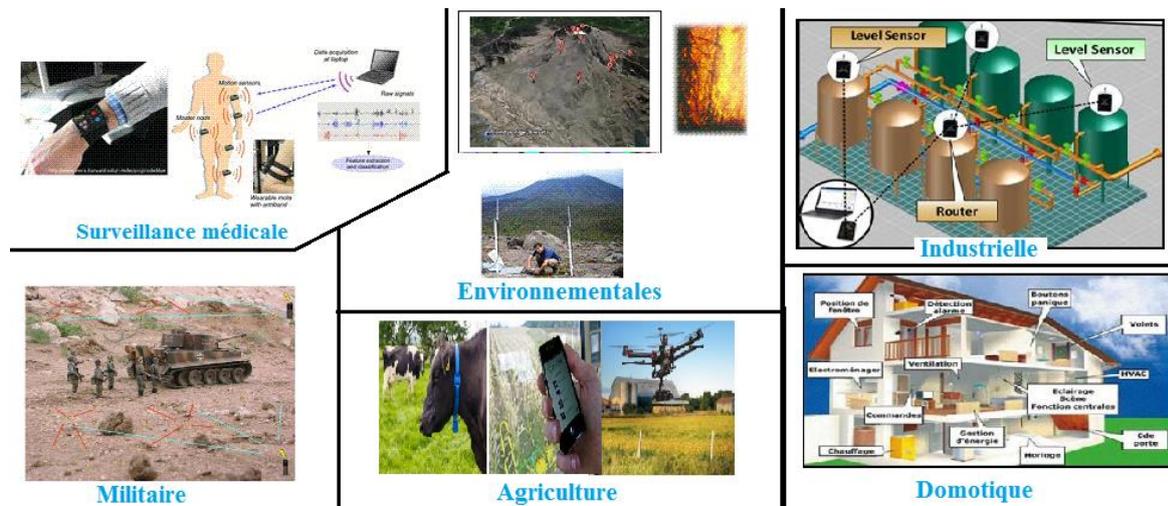


Figure 2.6. Quelques applications des réseaux de capteurs sans fil.

### ➤ Applications liées à la sécurité

Les RCSF peuvent être appliqués pour la sécurité des zones privées et publiques. Néanmoins, le véritable intérêt des RCSF pour des e-applications n'est pas dans la surveillance des établissements fermés seul, mais aussi dans les endroits ouverts. Pour ce type d'application, une longue liste de produits est disponible sur le marché. Des capteurs sans fil de mesure, micro-caméras et autres dispositifs existent déjà pour la surveillance. Ils s'appuient sur des méthodes, des protocoles de routage, et des technologies de communication. Des capteurs limités en énergie pourraient être mis en place, avec l'ajout des capteurs riche en ressources pour protéger des parcs, les industries forestières ou agricoles, des zones sauvages, et d'autres zones liés a la protection des ressources naturelles.

### ➤ Applications environnementales

Les e-applications dans les RCSF sont aussi utiles pour la vigilance environnementale. Comme les nœuds consomment très peu d'énergie, ils peuvent être déployés dans des endroits stratégiques pour de longues périodes de temps (de l'ordre de plusieurs mois), afin d'obtenir des informations d'intérêt scientifique sur de larges zones géographiques. Grâce à cela, les observateurs de la nature peuvent étudier le comportement et les habitudes des diverses espèces animales, en obtenant des données, qui peuvent être très éloignées, sans avoir besoin de se déplacer physiquement. Des systèmes pour les tâches de gardes de parcs, comme la détection des incendies de forêt, pourraient être développés. A titre d'exemples, on peut citer les cas suivants:

- ✓ la dispersion de thermo-capteurs à partir d'un avion sur une forêt pour signaler un éventuel début d'incendie dans le champ de captage,
- ✓ le semis de nœuds capteurs en même temps avec les graines dans les champs agricoles pour pouvoir identifier les zones sèches et rendre l'irrigation plus efficace,
- ✓ le déploiement de nœuds capteurs sur les sites industriels, les centrales nucléaires ou dans les raffineries de pétrole pour détecter des fuites de produits toxiques (gaz, produits chimiques, éléments radioactifs, pétrole, ...etc.) et alerter les utilisateurs dans un délai suffisamment court pour permettre une intervention efficace.

### ➤ Applications médicales

Surveillance permanente des patients et une possibilité de collecter des informations physiologiques de meilleure qualité facilitant ainsi le diagnostic de maladies grâce à des micro-capteurs qui pourront être ingérés ou implantés sous la peau.

- ✓ les micro-caméras qui peuvent être ingérées et sont capables, sans avoir recours à la chirurgie, de transmettre des images de l'intérieur d'un corps humain,
- ✓ la création d'une rétine artificielle composée d'une centaine de micro-capteurs pour améliorer la vision de l'œil.

### ➤ Applications écologiques

L'intégration de plusieurs micro-capteurs dans le système de climatisation et de chauffage des immeubles. Ainsi, la climatisation ou le chauffage ne sont déclenchés qu'aux endroits où il y a des personnes présentes et seulement si c'est nécessaire. Le système distribué peut aussi maintenir une température homogène dans les pièces. Utilisée à grande échelle, une telle application permettrait probablement de réduire la demande mondiale en énergie.

### ➤ Applications de traçabilité et de localisation

Suite à une avalanche il est nécessaire de localiser les victimes enterrées sous la neige en équipant les personnes susceptibles de se trouver dans des zones à risque par des capteurs. Ainsi, les équipes de sauvetage peuvent localiser plus facilement les victimes. Contrairement aux solutions de traçabilité et de localisation basées sur le système de GPS (Global Positioning System), les réseaux de capteurs peuvent être très utiles dans des endroits clos comme les mines par exemple.

### ➤ Applications du trafic routier

Des réseaux de capteurs peuvent être déployés pour le suivi et le contrôle de la circulation routière ou bien la surveillance du déplacement d'objets dans un champ. Des algorithmes d'analyse peuvent être utilisés pour faire le dénombrement des véhicules ou des personnes pour estimer le niveau de trafic en fonction des heures de la journée.

## 5. Conclusion

Dans ce chapitre, nous avons présenté une étude qualitative pour comprendre l'auto-adaptation, le routage et les e-applications dans les RCSF.

D'abord, nous avons fait une étude sur quelques concepts tels que les objectifs, rôles et défis d'auto-adaptation dans les RCSF, Nous avons présenté les solutions proposées dans la littérature, et fédéré les concepts nécessaires à la compréhension des problématiques abordées dans le reste de ce manuscrit.

Par la suite, nous avons présenté les principales classes des protocoles de routage dans les RCSF, et les protocoles de routage hiérarchiques qui ont été proposés comme solution pour assurer le service de routage dans les RCSF a grand échelle, nous avons étudié aussi les principaux caractéristiques de ce type de routage.

Enfin, nous avons présenté les travaux e-applications développés dans plusieurs domaines se basant sur les RCSF. Le chapitre suivant est consacré à l'introduction des RCSF hétérogènes.

## **Chapitre 3**

### **Etat de l'art: Les RCSF hétérogènes**

### 1. Introduction

Les réseaux de capteurs sans fil se composent de plusieurs milliers de capteurs qui peuvent collecter des informations précises dans des environnements éloignés et périlleux.

Tous les nœuds des réseaux homogènes sont identiques en termes de complexité du matériel et d'énergie. Néanmoins, en fonction de l'application, il est évident que certains capteurs peuvent avoir des rôles différents, ou des périphériques spécifiques [94], comme la coordination du protocole de routage et l'agrégation des données, par conséquent, leur durée de vie expire rapidement avant celle de ses autres nœuds. Dans l'approche proposée Leach (Low Energy Adaptive Clustering Hierarchy) [95], les auteurs proposent la première solution, cette solution permet de faire tourner le rôle du Cluster-Head périodiquement, mais le problème des ressources limitées est resté actif, ils n'ont pas de matériel assez puissant et également le réseau homogène peut souffrir des mauvaises performances [96]. Pour améliorer les performances du réseau, une clé des futures recherches dans ce domaine est sans doute liée au contexte hétérogène, donc il est mieux de former un réseau avec un nombre de nœuds de haute capacité c'est-à-dire dans le réseau il y a quelques nœuds sophistiqués qui ont plus de capacité de traitement, capacité énergétique et de communication que les nœuds normaux [97].

*P. Samundiswary et al* dans [98], implémentent un réseau de capteurs sans fil hétérogènes, l'idée est pour atteindre le nœud station de base et ils impliquent moins de multi-sauts que les réseaux homogènes. Les résultats de comparaison démontrent que l'énergie consommée en utilisant le réseau de capteurs hétérogènes est réduite de 92,5% de moins que le réseau de capteurs homogène. Aussi, également, le taux de livraison dans les réseaux hétérogènes est meilleur que celui dans les réseaux homogènes [99].

À la fin de ce chapitre, nous détaillons un exemple de « e-application » dans le domaine d'agriculture pour les RCSF hétérogènes, cet exemple est plus proche de notre solution proposée dans le chapitre suivant (*chapitre 4*), sans risque de compromettre la robustesse du système.

### 2. Les conditions de fonctionnement

Plusieurs conditions doivent être remplies pour qu'un réseau de capteur hétérogène fonctionne, d'abord, il faut que les capteurs parlent le même langage et comprennent le langage des autres nœuds du même RCSF, la communication doit être assurée entre tous les nœuds du réseau. En plus, les capteurs sont équipés des redirecteurs adéquats pour

s'adresser à la station de base d'un autre système d'exploitation. Enfin, les stations de bases sont équipées des services adéquats pour recevoir les informations des nœuds capteurs d'un autre système d'exploitation.

### 3. Quelques modèles d'hétérogénéité

L'hétérogénéité est la possibilité que les nœuds dans un réseau soit différents. Il peut s'agir de l'hétérogénéité logicielle ou matérielle [100].

Les RCSF Hétérogènes représentent une nouvelle tendance adéquate et plus économique pour le contrôle et la surveillance des environnements à travers la capture, l'agrégation de mesure relatives à différents phénomènes physiques. Ces réseaux permettent d'opérer sur un même réseau sans avoir à intervenir pour l'adapter à chaque fois une nouvelle tâche ou à une nouvelle application. Une étude faite par les auteurs dans [101] a montré qu'un réseau hétérogène proprement déployé peut tripler le taux moyen de livraison et peut prolonger jusqu'à cinq fois plus la durée de vie du réseau.

#### 3.1. L'hétérogénéité matérielle

➤ **Hétérogénéité des nœuds capteurs** : L'auteur dans [102] propose un modèle d'hétérogénéité matérielle, ce modèle contient un autre type de nœud dans le réseau nommé le nœud actionneur (*Figure 3.1*). Ainsi, un WSAAN (*Wireless Sensors and Actuators Networks*) est un réseau hétérogène par nature puisqu'il est constitué au moins de deux types de nœuds : les nœuds capteurs qui collectent les données de l'environnement où ils sont déployés, et les nœuds actionneurs qui ne sont pas responsables seulement d'agir sur l'environnement physique mais aussi, d'un point de vue réseau, sont responsables de recueillir les données récoltées par les nœuds capteurs.

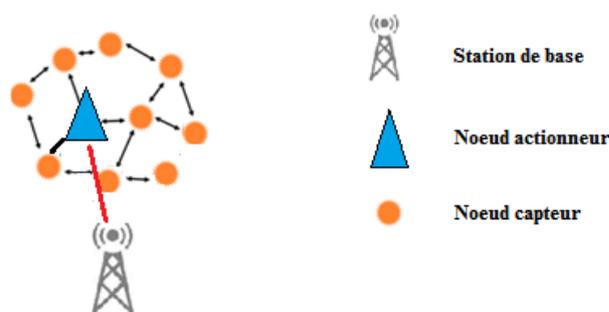


Figure 3.1. Exemple d'un cluster dans un RCSF et actionneur.

➤ **Hétérogénéité de l'unité de capture** : L'unité de capture ou d'acquisition est composée de deux sous unités, un dispositif de capture physique qui prélève l'information de l'environnement local et un convertisseur analogique/numérique appelé ADC (Analog to Digital Converters) qui gère la conversion des signaux analogiques provenant de l'unité de capture en signaux numériques interprétables par l'unité de traitement [103].

Au cours des dernières décennies, la nécessité d'observer et de contrôler les phénomènes physiques tels que la température, la pression, la luminosité, ...etc. est essentielle pour de nombreuses applications industrielles et scientifiques. Par exemple, dans le domaine de l'écologie, la surveillance des polluants comme Ozone, NO<sub>2</sub> ou CO<sub>2</sub> pourrait augmenter considérablement la qualité de vie dans les villes [23, 104, 105, 106]. Dans un RCSF homogène, chaque capteur observe et contrôle un seul phénomène, un tel réseau de surveillance de la température, un autre pour la surveillance de la pression, ...etc. Mais avec le concept d'hétérogénéité, il est possible d'avoir un capteur avec plusieurs unités de capture (Figure 3.2), donc les données capturées sont envoyées périodiquement aux clusters-head pour collecter, agréger, traiter et envoyer le résultat final à la station de base.

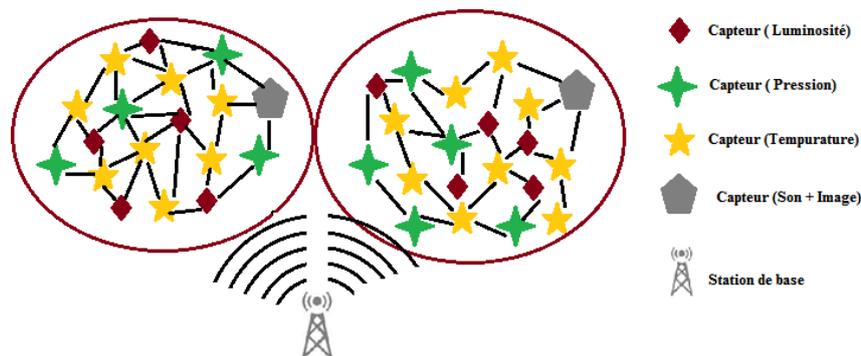


Figure 3.2. Hétérogénéité de l'unité de capture.

### 3.2. L'hétérogénéité logicielle

➤ **Hétérogénéité de systèmes d'exploitation** : Les systèmes d'exploitation représentent un socle sur lequel s'appuient les programmes d'application. Ils servent de lien ou d'interface entre l'architecture matérielle et la partie logicielle. Dans les réseaux de capteurs sans fil, les systèmes d'exploitation conservent ce rôle. Sachant que certaines architectures sont plus puissantes que d'autres, il est logique d'extrapoler le concept au niveau logiciel. On peut donc avoir des systèmes d'exploitation qui offrent plus de caractéristiques, ou qui utilisent moins de mémoire flash/ram. Donc une possibilité non seulement d'hétérogénéité matérielle mais aussi logicielle. C'est-à-dire plusieurs systèmes

d'exploitation ou bien les mises-à-jour visant des systèmes d'exploitation bien particulières dans un RCSF. Ceci mène à ce qu'on appelle "hétérogénéité complète matérielle - logicielle".

➤ **Hétérogénéité du Protocole de routage** : Suivant la manière de création et de maintenance des routes, les protocoles de routage peuvent être répartis en trois catégories: proactifs, réactifs et hybrides.

**Protocole proactif** : Le principe de base des protocoles proactifs est que chaque nœud du réseau a, dans sa table de routage, un chemin vers l'ensemble des nœuds du réseau. Ce mode de fonctionnement implique des échanges constants entre les différents nœuds pour prendre en compte les potentiels changements de topologie du réseau. Le principal avantage de ce type de protocole est que chaque nœud dispose immédiatement d'un chemin vers n'importe quel autre nœud du réseau. Les protocoles DSDV [107] et OLSR [108] appartiennent à cette famille.

**Protocole réactif** : Dans un protocole réactif, les chemins sont construits au fur et à mesure des communications entre les nœuds. Au départ, la table de routage d'un nœud est vide. Quand un nœud va vouloir communiquer avec un autre, la première étape est la recherche d'un chemin valide par l'envoi de messages à travers le réseau. Si un tel chemin est découvert, la communication pourra débuter. Par rapport à un protocole proactif, le nombre de messages pour établir les routes est moindre. Des exemples de protocoles réactifs sont AODV [109] et DSR [110].

**Protocole hybride** : Les protocoles de routage hybrides combinent les aspects proactif et réactif de différentes manières [111]. Par exemple, dans [112, 113], une zone au fonctionnement proactif est définie autour de chaque nœud. Au-delà de cette zone, un protocole réactif est utilisé. D'autres protocoles comme FZRP [114] utilisent des méthodes différentes à base de constructions d'arbres de routage.

Certain protocoles de routages offrent plus de caractéristiques que d'autres, par exemple ils utilisent moins des messages de communications qui consomment trop d'énergie. Avec le concept d'hétérogénéité, il existe une possibilité d'avoir plusieurs protocoles de routages dans un même RCSF. C'est-à-dire les nœuds utilisent différent protocoles de routage de la même famille (proactif ou réactif ou hybride) ou bien à partir des différentes familles.

➤ **Hétérogénéité du Canal de communication** : Les auteurs dans [102] considèrent le RCSF hétérogène à 2 niveaux comme illustré dans *la figure 3.3*. Le réseau de niveau 1 appelé réseau de capteurs fonctionne sur des canaux de fréquences 802.15.4. Notons que les nœuds du niveau 1 utilisent seulement la couche physique du standard 802.15.4. Pour le mécanisme de transmission (couche MAC), les auteurs utilisent un

## Chapitre 3 : Les RCSF hétérogènes

CSMA/CA classique c'est-à-dire "non slotté et non beacon activé". Le réseau de niveau 2 appelé réseau de puits (réseau entre les nœuds actionneurs) fonctionne sur un unique canal 802.11. Les auteurs supposent que le canal 802.11 choisi pour le réseau de niveau 2 n'interfère pas avec les canaux 802.15.4 choisis pour le réseau de niveau 1 [98]. Les nœuds du niveau 2 implémentent toute la pile protocolaire du standard 802.11. Tous les émetteurs radios du réseau de niveau 1 ont la même portée radio.

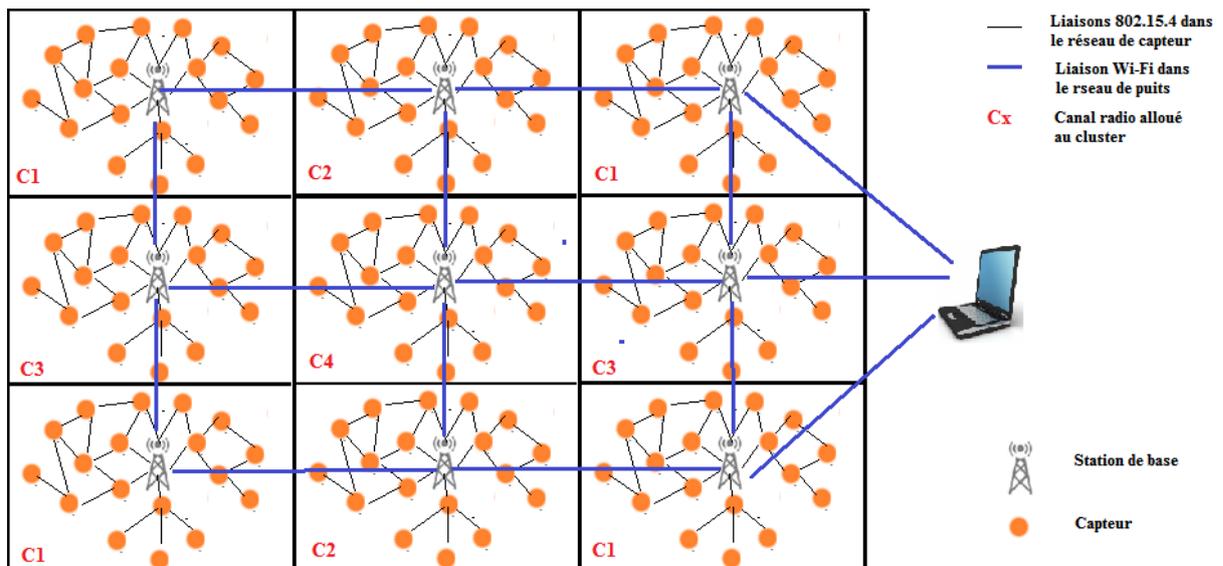


Figure 3.3 : Topologie Hétérogène exploitant quatre canaux dans un RCSF.

En outre, un seul émetteur-récepteur radio est suffisant pour utiliser dynamiquement plusieurs canaux de fréquences [115] et le délai pour changer de canal n'a pas d'incidence sur la capacité du réseau [116]. Par exemple, la latence d'un émetteur-récepteur radio CC2420 pour basculer d'un canal à un autre est d'environ 200 $\mu$ s. Nous avons vérifié cette propriété sur un nœud réel.

### 4. Exemple de « e-application » dans les RCSF hétérogènes

Dans cette partie, nous expliquons un exemple de « e-application » dans les RCSF hétérogènes, ce qui correspond à notre première proposition. Il s'agit de l'agriculture.

Le métier d'agriculteur se spécialise mais les surfaces à gérer, les contraintes, les réglementations, les tâches quotidiennes s'amplifient. Les capteurs de mesure et le numérique aident les agriculteurs à réduire la pénibilité et les temps d'astreinte de ce métier.

## Chapitre 3 : Les RCSF hétérogènes

La naissance des sociétés pour répondre à la demande des agriculteurs, l'objectif de ces entreprises est le développement des capteurs simples avec des interfaces web interactives.

### ➤ La plateforme

Ce travail a besoin de deux processus de base: l'installation logicielle et l'installation matérielle.

L'étape d'installation logicielle doit commencer par l'installation d'un système d'exploitation pour les RCSF comme TinyOs sous Ubuntu 10.10 par exemple, ensuite, il faut installer une application Web pour créer l'interface de gestion, nous proposons PhpMyAdmin, les capteurs telosb, MicaZ et le langage NesC.

Dans la deuxième étape d'installation matérielle, il faut utiliser un capteur telosb comme Receiver et des capteurs MicaZ riches en ressources comme Sender dont chacun représentant un rôle bien spécifique et des capteurs d'humidité du sol comme EC-5 (hétérogénéité des capteurs), les détails sont illustrés dans la représentation graphique (Figure 3.4). Le MicaZ communique avec la station de base via une liaison sans fil et la station de base communique avec l'ordinateur via le câble USB (hétérogénéité de communication). L'agriculteur peut s'utiliser la technologie d'Internet pour surveiller le processus à distance via la communication avec l'ordinateur afin d'améliorer le service ou d'intervenir en cas d'urgence.

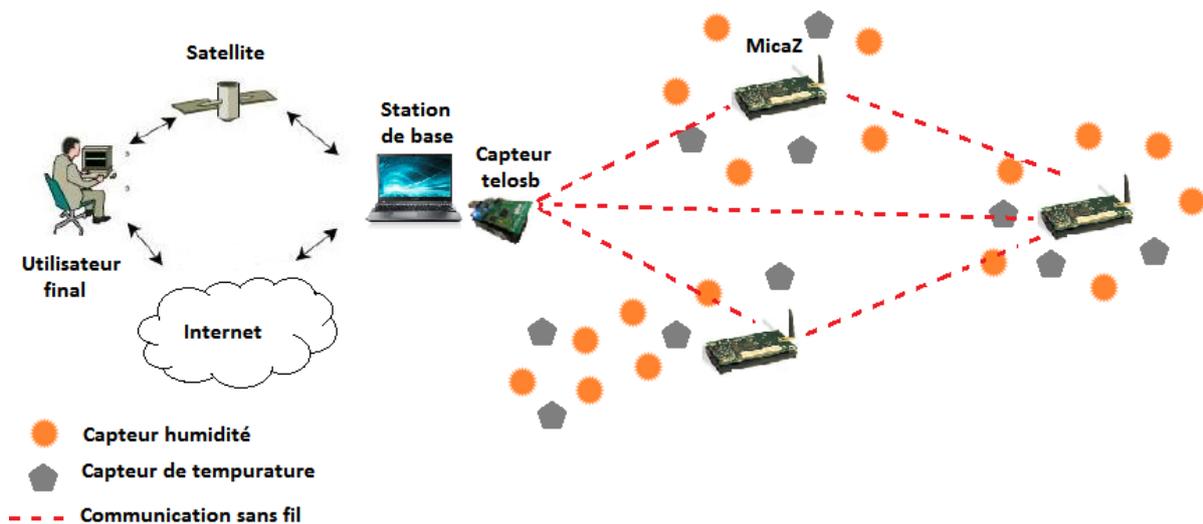


Figure 3.4. Exemple de « e-application » dans les RCSF hétérogènes.

### ➤ L'architecture de l'application

L'eau (L'eau) et le soleil sont deux éléments importants voire vitaux à la vie de la majorité des plantes. L'irrigation est l'opération qui consiste à apporter artificiellement de l'eau à des végétaux cultivés [117].

Cette technologie augmente l'efficacité de l'irrigation, ce qu'elle nous permet de suivre le taux d'humidité et la température de manière à conserver au maximum les plantes en vie, car chaque plante a besoin d'un taux d'humidité et température spécifique, d'où une irrigation inadaptée, ou mal conçue, ou bien une augmentation de la température peut être source de beaucoup de problèmes.

L'utilisation de RCSF dans l'agriculture permet de prendre la décision appropriée, où les nœuds de capteurs collaborent entre eux et interagissent avec l'environnement de captage sur lesquels ils sont déployés.

Dans cette partie, nous présentons une solution complète d'irrigation pour l'agriculture à la base de RCSF hétérogène qui correspond à notre première contribution. Son principe est de créer un système d'irrigation automatisé qui peut mesurer la quantité efficace de l'eau nécessaire par les plantes. Et par conséquent réduire la consommation d'eau dans la zone de captage.

Cette technologie consiste à utiliser des capteurs à faibles ressources placés dans la zone des racines des plantes ou par un déploiement aléatoire, ces capteurs peuvent être programmés pour enregistrer des mesures telles que la température et l'humidité du sol. Ensuite, l'endroit de captage doit être divisé en morceaux, et à l'intérieur de chaque partie un capteur riche en ressource de type MicaZ (déploiement déterministe) pour jouer le rôle d'un cluster-head.

Selon les valeurs contrôlées, les simples capteurs répartis dans l'espace recueillent la température et l'humidité pour les envoyer vers les capteurs riches en ressources (de type MicaZ) où les informations sont agrégées afin de les envoyer vers le capteur telosb qui communique directement avec la station de base. Ce processus a été fait selon le processus de migration suivant des changements dans le RCSF hétérogène. Ensuite, sans intervention humaine, le programmeur déclenche automatiquement l'irrigation goutte à goutte ou Irrigation gravitaire Lorsque l'humidité est faible et Irrigation par aspersion lorsque la température élevée ou des vents forts pour protéger les feuilles des plantes de la sécheresse (*Figure 3.5.*), ce mécanisme se répète aux jours et heures voulus, pour la durée programmée.

## Chapitre 3 : Les RCSF hétérogènes

---

Quand une plante est jeune, elle nécessite plus d'attention et une fois qu'elle est un peu plus âgée, elle peut s'adapter à l'environnement et compter sur elle-même, ce qui explique le succès de notre protocole de routage proposé dans ce domaine. En d'autres termes, on s'intéresse dès le départ à fournir les informations nécessaires dans le domaine de l'agriculture. Après une certaine période, seules les informations concernant les nouveaux changements sont envoyées vers la station de base.



Figure 3.5. Les systèmes d'irrigation existants.

### 5. Conclusion

Dans ce chapitre, nous avons présenté les RCSF hétérogènes et les conditions de fonctionnement, Ensuite, nous avons identifié quelques modèles d'hétérogénéité (matériels et logiciels). Nous nous concentrons sur les WSAN comme exemple d'hétérogénéité des nœuds capteurs, et l'hétérogénéité de systèmes d'exploitation, du protocole de routage, et du Canal de communication comme exemples d'hétérogénéité logiciels.

Par ailleurs, nous avons donné un exemple de « e-application » dans les RCSF hétérogènes; cet exemple peut être celui de l'application de notre première contribution que nous allons présenter dans le chapitre suivant.

## **Partie 2 : Contributions**

## **Chapitre 4**

# **Approches d'auto-adaptation dans les RCSF hétérogènes**

### 1. Introduction

Selon différentes situations, un réseau de capteurs sans fil hétérogène exige que chaque nœud de capteur soit suffisamment indépendant et souple pour assurer la fonction d'auto-adaptation [118]. Dans le réseau de capteur il n'y a aucune infrastructure fixe disponible pour la gestion de réseau, ce qui rend la gestion d'auto-adaptation des réseaux de capteurs sans fil hétérogènes plus difficile.

L'adaptation dans les réseaux de capteurs sans fil est le processus par lequel un logiciel est modifié afin de prendre en compte un changement au niveau de l'environnement de captage [119]. Il s'agit d'un processus en trois phases. Il faut détecter les changements, décider de la réaction la plus appropriée à la situation détectée, et enfin réaliser les traitements décidés [120]. L'adaptation peut être statique ou dynamique. L'adaptation statique est le processus réalisé avec reprise du logiciel suite aux changements détectés, par contre l'adaptation est dite dynamique si ce processus continue son exécution malgré le changement en cours rencontré. Cette dynamique s'avère nécessaire dans différents domaines d'application surtout dans les RCSF où un arrêt peut être très coûteux financièrement ou dangereux du point de vue humain ou environnemental. Pour remédier à ces problèmes, nous avons proposé des approches d'auto-adaptation dans les RCSF hétérogènes.

Dans ce chapitre, nous proposons un protocole nommé ADAPTRP (ADAPTative Routing Protocols), ce dernier répond aux grands problèmes dans les RCSF par une politique de migration vers quelques protocoles de routage hiérarchiques. Pour réaliser une telle migration, la station de base applique des algorithmes basés sur la consommation d'énergie, la distance et la densité des nœuds.

### 2. Motivation

Pour mener à bon escient notre travail, nous nous concentrons sur quatre points: d'abord, sur de nombreux articles dans la littérature qui ont traité la capacité d'auto-adaptation des protocoles de routage dans les RCSF. La décision qui adopte le processus d'auto-adaptation est définie par des algorithmes et des mécanismes, étant donné que la configuration basée sur multi-objectifs peut satisfaire le nouveau contexte. Compte tenu de la rareté des ressources dans les RCSF hétérogènes, l'auto-adaptation basée sur un critère multi-objectif doit être optimale, en tenant compte de certaines dimensions telles que la consommation de ressources et le coût de reconfiguration. Cependant, une telle sélection ne doit pas se faire au détriment d'autres dimensions. Par conséquent, nous définissons une

## **Chapitre 4 : Approches d'auto-adaptation dans les RCSF hétérogènes**

---

approche basée sur un principe multi-dimensions dans les RCSF tels que la consommation de ressources, la distance, la densité et la qualité du service (QoS) en même temps.

Ensuite, dans notre proposition, le routage est considéré comme hétérogène dans le cluster lui-même en fonction des changements dans l'environnement (Figure 4.2.). À notre connaissance, cette idée n'a pas été abordée auparavant et pour la réalisation de cette idée, un algorithme de migration est proposé pour acheminer l'évolution du protocole vers un autre.

Puis, selon l'énergie initiale, nous adoptons dans notre approche l'idée proposée par les auteurs dans [9], en ajoutant des nœuds sophistiqués ayant plus de capacité d'énergie que les nœuds normaux. Cette idée améliore l'efficacité du routage et prolonge la durée de vie du réseau, parce que les nœuds sophistiqués peuvent être utilisés pour effectuer des tâches plus complexes telles que les chefs de clusters et des coordinateurs.

Enfin, nous proposons un nouveau mécanisme dans notre protocole de routage, en fonction de la désactivation des nœuds inutiles, pour renforcer l'idée d'auto-adaptation dans les RCSF hétérogènes.

### **3. Etude comparative des protocoles de routage**

Les capteurs se communiquent par de nombreuses stratégies sans fil et ces stratégies de communication sont administrées par des protocoles de routage. Dans cette partie, on réalise une étude comparative des protocoles de routage vus dans le chapitre 3 pour choisir le meilleur protocole dans chaque cluster. On s'intéressera spécialement à la classe des protocoles qui se basent sur le routage hiérarchique. En effet, ce type de protocole présente divers avantages, tels que plus d'évolutivité, moins de charge, moins de consommation d'énergie et plus de robustesse [65].

#### **3.1. Quelques avantages et limites des protocoles de routage**

##### **➤ LEACH**

##### **Avantages de LEACH**

1. Égalité de chances entre les nœuds pour devenir un cluster-head ;
2. Le cluster-head ne peut pas être sélectionné comme cluster-head dans le tour suivant, de sorte que la charge est partagée entre les nœuds ;

## **Chapitre 4 : Approches d'auto-adaptation dans les RCSF hétérogènes**

---

3. L'auto-configuration des clusters se fait indépendamment de la station de base (algorithme distribué) [121] ;
4. La consommation d'énergie est partagée sur tous les nœuds prolongeant ainsi la durée de vie du réseau ;
5. les cluster-heads agrègent les données de leur cluster grâce aux traitements (agrégation, compression) locaux des données [4], et envoient un paquet agrégé à la station de base pour réduire la quantité d'informations à transmettre à la base de la station.

### **Inconvénients de LEACH**

1. Sans justifier leur choix, les auteurs fixent le pourcentage optimal de CH pour le réseau à 5% du nombre total de nœuds. Cependant, la topologie, la densité et le nombre de nœuds peuvent être différents dans chaque cluster.
2. Aucune suggestion n'est faite concernant le moment de la réélection des CH (temps d'itération).
3. Les cluster-heads éloignés de la station de base meurent rapidement par rapport à ceux qui sont proches de la station [4].
4. Nous ne pouvons pas avoir un cluster-head durant un round si les nombres aléatoires générés par tous les nœuds du réseau sont supérieurs à la probabilité  $P(t)$ .
5. L'utilisation d'une communication à un seul-saut au lieu d'une communication multi-saut diminue l'énergie des nœuds.
6. Manque de communication inter-clusters [122].

### **➤ PEGASIS**

#### **Avantages de PEGASIS**

1. PEGASIS réduit la consommation énergétique due à la formation dynamique de clusters [123] et diminue le nombre de transmissions de données en raison de la chaîne d'agrégation des données.
2. La charge d'énergie est répartie uniformément dans le réseau. Cela implique que tous les nœuds des capteurs agissent dans le réseau comme un cluster-head.

#### **Inconvénients de PEGASIS**

1. PEGASIS n'est pas adapté aux réseaux avec des topologies variant dans le temps.
2. La communication a des retards très longs.

## Chapitre 4 : Approches d'auto-adaptation dans les RCSF hétérogènes

---

3. Le réseau n'est pas très évolutif car tous les nœuds doivent avoir une connaissance globale du réseau pour la construction de la chaîne.

### ➤ H-PEGASIS

#### Avantages de H-PEGASIS

1. la diminution des délais de transmissions des paquets vers la station de base [123].
2. L'adaptation de la transmission simultanée de messages.
3. La suggestion d'une solution au problème de la collecte de données en considérant la métrique EDP (Energy x Delay Product) [124].
4. Un protocole basé sur le type de hiérarchie chaîne et clustering au même temps.

#### Inconvénients de H-PEGASIS [125, 126]

1. Comparé à LEACH, H-PEGASIS réduit les frais généraux de création de cluster mais il n'est pas adapté au réseau lourdement chargé.
2. H-PEGASIS ne prend pas en compte la condition d'énergie du saut suivant quand il choisit un chemin de routage.
3. Il n'est pas adapté au réseau de capteurs où la connaissance globale n'est pas facile à obtenir.

### ➤ HEED

#### Avantages de HEED

1. Utilisation de deux paramètres importants pour l'élection d'un CH, c'est une méthode de regroupement entièrement distribuée;
2. Les niveaux de faible concentration de clusters favorisent une augmentation de la réutilisation spatiale, tandis que des niveaux élevés de concentration de clusters sont nécessaires pour la communication entre les clusters. Cela fournit une distribution uniforme de CH dans le réseau et l'équilibrage de charge;
3. Les communications de mode multi-saut entre les CH et la SB favorisent une plus grande conservation de l'énergie et une évolutivité contrairement au mode à un seul saut.
4. Afin de former les clusters, les nœuds ont uniquement besoin d'information locale (nœuds voisins) [86].

### Inconvénients de HEED

1. Les nœuds sont forcés de devenir un CH et ces CHs peuvent être formés dans un autre cluster ou ne peuvent pas avoir de membres associés à eux.
2. HEED souffre de coûts consécutifs car il nécessite plusieurs itérations pour former des clusters (beaucoup de paquets sont diffusés) [23].
3. Certains CH, surtout près de la station de base, peuvent mourir plus tôt parce que ces CH ont plus de charge de travail, et apparaîtra dans le réseau.
4. Ce protocole se focalise plutôt sur la prolongation de la durée de vie du réseau, plutôt que sur l'ensemble des besoins d'un réseau de capteurs [86].

#### ➤ TEEN

### Avantages de TEEN

1. Moins de transmission de données parce qu'il utilise deux seuils [67]; ce qui implique une économie en énergie.
2. TEEN est un protocole pour réagir aux applications dites à contraintes temporelles [80] qui convient à des situations réactives et à des applications critiques dans le temps.
3. Plus économe en énergie que les protocoles proactifs (la transmission d'un message consomme plus que la collecte d'informations sensorielles) [123].

### Inconvénients de TEEN

1. La possibilité que la SB ne soit pas en mesure de faire la distinction entre les nœuds morts et ceux en vie [88].
2. Inadéquation pour les applications exigeant des rapports périodiques [66].
3. Un nœud peut attendre pendant son slot de temps.
4. Un slot affecté peut être gaspillé si le nœud n'a aucune donnée à transmettre [65].

#### ➤ APTEEN

### Avantages de APTEEN

1. Combinaison des deux politiques proactive et réactive [66].
2. Le support de trois types de requêtes : historiques, persistantes, obtention d'un instantané de la vue du réseau actuel [65].

## Chapitre 4 : Approches d'auto-adaptation dans les RCSF hétérogènes

### Inconvénients de APTEEN

1. Complexité additionnelle requise pour la mise en place des paramètres supplémentaires [65].

Ci-dessous *la table 4.1*, montre les avantages et les inconvénients des protocoles vus dans le chapitre 3.

Protocoles	Classification	CH	Nbr de sauts	TDMA	Efficacité d'énergie	Données transmises dans un temps t	Propriété	Densité
LEACH	Proactive	oui	Un seul	oui	Très faible	Très fort	L'un des premiers protocoles de clustering et le plus connu	Pas de problème parce que le réseau est hétérogène
LEACH-C	Proactive	oui	Un seul	oui	Très faible	Très fort	Sélection du CH centralisée par rapport au LEACH	Pas de problème parce que le réseau est hétérogène
HEED	Hybrid	oui	Un seul	oui	Très faible	Très fort	Sélection du CH améliorée par rapport au LEACH	Pas de problème parce que le réseau est hétérogène
PEGASIS	Reactive	non	Multi	non	Moyenne	Moyenne	Prolonger 2 à 3 fois la durée de vie par rapport LEACH	N'assume pas la densité
H-PEGASIS	Proactive	oui	Multi	non	Faible	Fort	Pour réduire le retard dans PEGASIS	Meilleur que PEGASIS
TEEN	Reactive	oui	Multi	non	Très fort	Très faible	Les applications sensibles aux délais	N'assume pas la densité
APTEEN	Hybrid	oui	Multi	oui	Très fort	Très faible	Protocole événementiel et périodique	Meilleur que TEEN, Faible que LEACH
HEEP	Hybrid	oui	Multi	oui	Faible	Fort	Sélection du CH : LEACH, Communication : H-PEGASIS	Meilleur que PEGASIS

Tableau4.1. Comparaison des différents protocoles de routage hiérarchique.

### 4. Présentation de notre RCSF Hétérogène

Nous considérons un RCSF hétérogène à plusieurs paramètres comme illustré dans la figure 4.1. Nous considérons que les caractéristiques importantes d'hétérogénéité sont basées sur le rôle d'un capteur (nœud ordinaire, cluster-head ou station de base), l'état du nœud (activé ou désactivé), le déploiement (aléatoire ou déterminé), la fonction de l'unité de capture (température, humidité, ...etc.) et l'environnement de capture (eau, terre ou air) où l'hétérogénéité peut comporter des capteurs sur la terre, suspendus dans l'air ou flotter sur l'eau. Pour augmenter la durée de vie du réseau, le système doit être auto-adapté pour économiser de l'énergie. Pour ce faire, plusieurs options sont possibles.

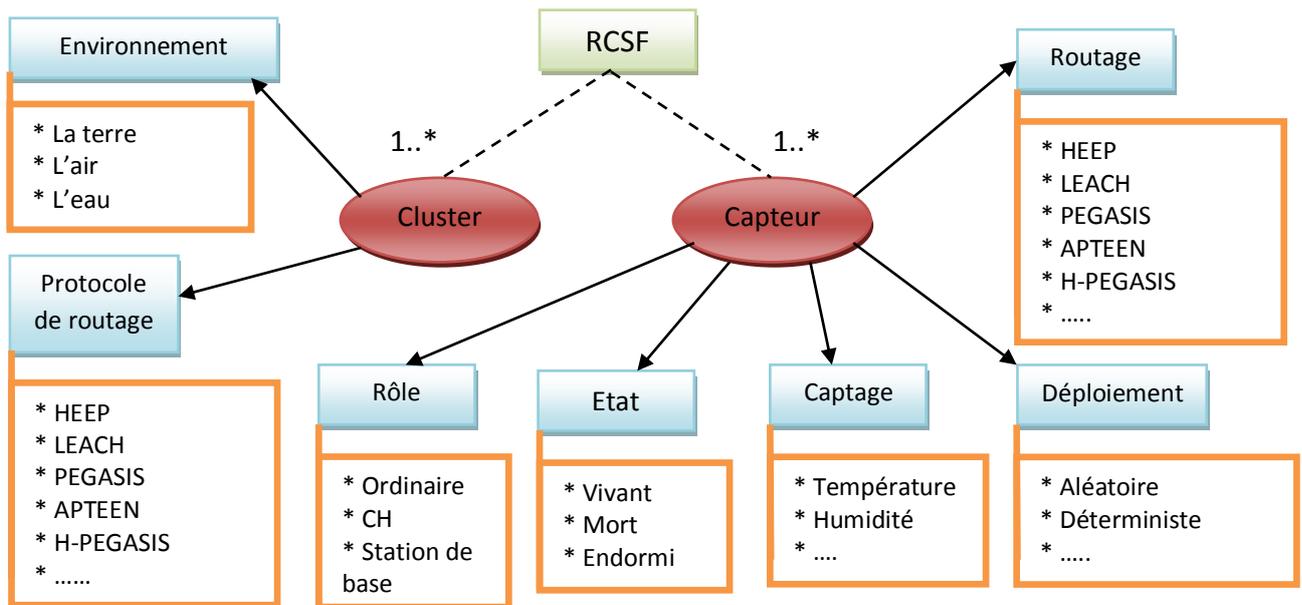


Figure 4.1. Modèle d'hétérogénéité dans les RCSF.

### ➤ Hétérogénéité de routage

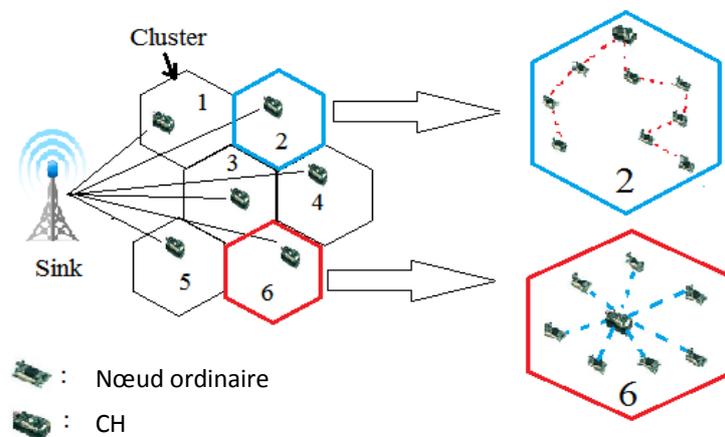


Figure 4.2. Routage Hétérogène pour les RCSF.

Les protocoles de routage peuvent être divisés en trois catégories en ce qui concerne la manière adoptée pour l'établissement des routes et la maintenance des routes: proactive, réactive et hybride. Certains protocoles de routage offrent des fonctionnalités différentes des autres, certains utilisent moins de messages de communication et d'autres consomment plus d'énergie. Avec le concept d'hétérogénéité, il existe une possibilité d'avoir plusieurs protocoles de routage dans un seul RCSF (Figure 4.2.). C'est-à-dire que les nœuds de capteurs utilisent des protocoles de routage différents de la même famille ou de familles

différentes. Pour économiser de l'énergie, nous pouvons choisir le protocole de routage le plus efficace en termes d'énergie, en ce qui concerne les liens de communication et la densité des nœuds dans chaque cluster.

### ➤ **Hétérogénéité des rôles des nœuds capteurs**

Bien que les réseaux de capteurs sans fil homogènes soient utilisés dans de nombreuses applications, il existe un nombre croissant d'applications nécessitant l'utilisation d'un autre type de nœuds dans le réseau. Les RCSF hétérogènes sont composés d'au moins deux types de nœuds: les nœuds capteurs qui collectent et traitent les données environnementales où ils sont déployés; et les nœuds riches en ressources qui possèdent une grande capacité de traitement, une grande mémoire et une puissance de transmission. Ces nœuds ne sont pas seulement responsables de la détection de l'environnement physique, mais ils sont également responsables de l'agrégation des données collectées par les nœuds capteurs et transmettent les données agrégées à la station de base afin de surveiller et / ou de contrôler un ou plusieurs phénomènes donnés [23].

### ➤ **Hétérogénéité d'état des nœuds des capteurs**

Une autre façon d'économiser de l'énergie est la désactivation des nœuds dans le réseau. Par exemple, nous pouvons décider de désactiver 20% ou 30% des nœuds. Mais comment sélectionnons-nous les nœuds? Nous pourrions le faire de façon aléatoire ou arbitraire. Cependant, cette désactivation doit être effectuée avec précaution, car nous pouvons désactiver un cluster-head ou une station de base qui peut désactiver le réseau ou une partie de celui-ci. De plus, en fonction de la topologie du réseau, nous pouvons créer des nœuds inactifs (en mode veille), mais cela peut affecter sérieusement la précision du réseau.

### ➤ **Hétérogénéité des nœuds capteurs**

La coexistence des nœuds riches en ressources et des nœuds à faibles ressources ouvre de nouvelles occasions pour proposer des nouveaux mécanismes et des algorithmes qui tiennent compte de cette hétérogénéité et aussi en tirer profit.

Les nœuds riches en ressources ont une puissance de transmission élevée, la communication entre ces nœuds peut être avantageusement à long distance contrairement à la communication entre les nœuds à faibles ressources [89].

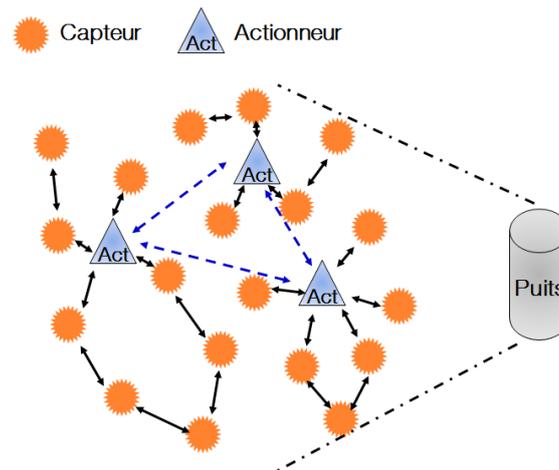


Figure 4.3. Exemple de réseau de capteurs hétérogène.

Dans notre proposition, le RCSF contient quelques nœuds sophistiqués qui ont plus de capacité de traitement, énergétique et de communication que les nœuds normaux (Figure 4.3). L'avantage de ce réseau est que ces nœuds sophistiqués peuvent être utilisés pour exécuter les tâches plus complexes comme les coordinateurs, les chefs de groupe (Cluster Head), ...etc. son inconvénient est qu'il est difficile de mettre en place un tel réseau du fait qu'au moins chaque type de nœuds du réseau aura un code (programme) propre à lui. Ce qui augmente le coût de développement [122].

### ➤ Hétérogénéité de déploiement

Dans notre RCSF hétérogène, nous commençons par le déploiement des nœuds capteurs aléatoire par ce que le recours à ce type de déploiement est alternatif et le coût de placer chaque nœud capteur d'une façon planifiée est élevé, donc, le coût est trop élevé lorsque l'environnement de captage est caractérisé par une grande taille de réseau.

En général, dans les applications de contrôle, de multimédia et de surveillance corporelle le déploiement déterministe est le plus adéquat. Dans notre travail, après le déploiement aléatoire où on a une connaissance préalable de la zone de surveillance, nous plaçons les nœuds sophistiqués d'une façon déterministe afin d'atteindre le déploiement de nœuds introduit également une hétérogénéité dans le RCSF.

### 5. Proposition d'une auto-adaptation dans les protocoles de routage hétérogène

Le processus d'auto-adaptation dans les protocoles de routage hétérogène doit remplacer l'adaptation actuelle par une nouvelle adaptation pour répondre au changement de la nouvelle situation du contexte.

Dans cette thèse, nous proposons une nouvelle adaptation non seulement pour satisfaire le contexte, mais aussi pour améliorer le protocole de routage concernant d'autres aspects.

Ainsi, compte tenu des différentes dimensions cela peut conduire à des objectifs contradictoires qui posent à leur tour une question intrigante sur la façon dont nous pouvons faire face à ces conflits.

Dans cette section, nous définissons une approche pour trouver l'auto-adaptation appropriée de protocoles de routage hétérogènes en tenant compte simultanément de multiples dimensions, telles que la consommation de ressources et la qualité de service (QoS). L'avantage de notre approche est double: (i) la sélection d'une auto-adaptation appropriée dans des protocoles de routage hétérogènes selon le besoin et (ii) la manière adoptée de faire face aux objectifs contradictoires.

#### 5.1. Sélection basée sur la consommation d'énergie, la distance et la densité

Il existe de nombreux problèmes d'optimisation multi-objectifs dans les RCSF. Dans de tels problèmes, les objectifs à optimiser sont normalement en conflit les uns avec les autres, ce qui implique qu'il n'existe pas de solution unique pour ces problèmes, mais une série de solutions de compromis. Ces solutions sont optimales en ce sens qu'aucune autre solution dans l'espace de recherche des solutions n'est supérieure à celles-ci lorsque tous les objectifs sont considérés simultanément [66].

##### ➤ Optimisation basée sur la consommation d'énergie

Heinzelmann et al., proposent dans [5] un modèle radioélectrique de consommation d'énergie, *figure 4.5*. Ainsi, l'énergie nécessaire pour transmettre un message de  $s$  bits sur une distance de  $d$  mètres est donnée par  $E_{Tx}(s, d)$  et pour recevoir le même message est donné par  $E_{Rx}(s)$ .

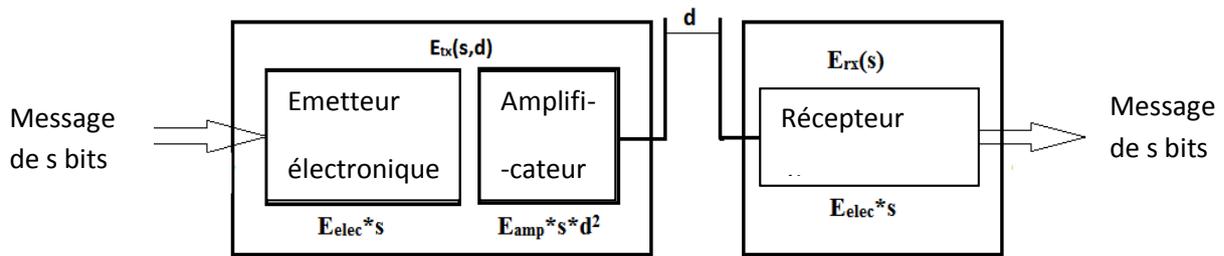


Figure 4.5. Modèle de consommation d'énergie [5].

$$E_{Tx}(s, d) = E_{Tx\ elec}(s) + E_{Tx\ ampli}(s, d)$$

$$\text{Ensuite, } E_{Tx}(s, d) = (E_{elec} * s) + (E_{amp} * s * d^2).$$

$$E_{Rx}(s) = E_{Rx\ elec}(s) \rightarrow E_{Rx}(s) = E_{elec} * s.$$

$E_{elec}$  et  $E_{AMP}$  désignent respectivement la puissance de transmission électronique et l'amplification, et ils sont représentés sur la figure 4.5.

Dans ce cas, la station de base calcule l'énergie consommée par tout protocole de routage "j" dans un cluster "k":  $E_k(P_j)$  et choisit le protocole qui consomme le moins d'énergie.

Fonction objectif:

$$E_k(P_j) = \min (\sum (i = 1 \text{ à } |N|) (E_{ti}(s, d) + E_{ri}(s) + E_{agi}(s))) \quad \dots\dots\dots (4.1)$$

$E_{ti}(s, d)$ : la puissance d'émission consommée par le ième nœud ;

$E_{ri}(s)$ : l'énergie de réception consommée par le ième nœud;

$E_{agi}(s)$ : l'énergie d'agrégation consommée par le ième nœud;

$D$ : la distance entre l'émetteur et le récepteur;

$S$ : le nombre de bits.

Nous avons une contrainte de base sur les protocoles de routage qui n'utilisent pas le concept d'agrégation:

$$C1E_k(P_j) = \min (\sum (i = 1 \text{ à } |N|) (E_{ti}(s, d) + E_{ri}(s))) \quad \dots\dots\dots (4.2)$$

### ➤ Optimisation basée sur la distance

Dans un WSN, un nœud est localisé à l'aide de ses coordonnées. Soient deux nœuds  $N_1$  et  $N_2$  situés dans la zone considérée. Nous appelons  $(x_1, y_1)$  les coordonnées du capteur

## Chapitre 4 : Approches d'auto-adaptation dans les RCSF hétérogènes

$N1$  et  $(x2, y2)$  celles de  $N2$ , alors la distance euclidienne sur le plan entre  $N1$  et  $N2$  est  $N1N2 = \text{Sqr}((x2-x1)^2 + (y2-y1)^2)$

Fonction objectif:

$$\text{DistK}(Pj) = \min (\sum (i = 1 \text{ à } |N|) \text{Sqr}((xi + 1 - xi)^2 + (yi + 1 - yi)^2)) \dots\dots\dots (4.3)$$

Dans notre cas, l'estimation de la distance se fait sur la base des liens de communication dans le protocole de routage "Pj". Par exemple, dans le cas où on utilise l'approche multi-hop, le nœud  $N1$  transmet ses données à son voisin  $N2$  le plus proche, alors  $N2$  à son tour, agrège les données reçues avec ses propres données et les transmet à son autre voisin, et ainsi de suite jusqu'à atteindre le CH. Ce dernier transmet à son tour les données agrégées directement à la BS. Mais en cas d'utilisation d'un seul saut, la fonction objective devient:

$$C1\text{DistK}(Pj) = \min (\sum (i = 1 \text{ à } |N|) \text{Sqr}((xCH-xi)^2 + (yCH-yi)^2)) \dots\dots\dots \text{Equation 4.4}$$

### ➤ Optimisation basée sur la densité

Fonction objectif:

$$\text{DNK} = \min (\sum (i = 1 \text{ à } |N|) (\text{SNi})) \dots\dots\dots (4.5)$$

DNK: densité des nœuds capteurs dans un cluster k.

SNi: i<sup>ème</sup> nœud de capteur.

Ici, nous recherchons un équilibre entre la précision offerte par l'ensemble du réseau (QoS) et l'énergie consommée. Dans la fonction objective, nous réduisons le nombre de nœuds capteurs actifs dans le réseau. En particulier, l'expression d'un (SNi) indique que le i<sup>ème</sup> nœud capteur est actif (SNi = 1) ou non actif (SNi = 0). Ensuite, dans notre modèle, la station de base et les clusters ne peuvent être désactivés, car leur désactivation introduit un coût supplémentaire sur l'auto-adaptation, puisque d'autres nœuds du cluster devraient remplacer le CH désactivé.

En conséquence, nous avons les trois contraintes de base suivantes pour optimiser DNK:

C1DNk: (SNi / Sni CH: (SNi) = 1) tous les nœuds qui sont CH sont toujours actifs.

C1DNk: (SNi / Sni SINK: (SNi) = 1) tous les nœuds qui sont des puits sont toujours actifs.

Avec les deux contraintes précédentes, la minimisation des nœuds actifs atteint l'état d'auto-adaptation où tous les nœuds ordinaires seront désactivés, mais cela serait inadmissible. Ensuite, nous devons éviter la désactivation de nombreux nœuds consécutifs afin d'obtenir une précision appropriée. Pour cette raison, nous définissons la contrainte suivante:

## Chapitre 4 : Approches d'auto-adaptation dans les RCSF hétérogènes

---

$$C3DNk: 1-laN \geq X, X > 0 \quad \dots\dots\dots (4.6)$$

Dans cette contrainte, nous supposons que la précision est comprise entre 0 (exclu) et 1 (inclus). La valeur  $laN$  représente la précision perdue dans le réseau en fonction des nœuds qui sont désactivés. Pour calculer  $laN$ , nous utilisons l'Algorithme 4.1. Dans cet algorithme, nous supposons que tous les nœuds ont la même contribution sur la précision. En conséquence, nous calculons la précision perdue en termes de nœuds désactivés. Par conséquent, nous calculons la précision perdue comme étant la moyenne arithmétique des nœuds désactivés (voir la ligne 19 et la ligne 30, Algorithme 4.1). Cependant, dans les RCSF, la précision est également affectée par la distance entre les nœuds qui sont encore actifs. Ensuite, dans la ligne 21, nous examinons les voisins de chaque nœud et augmentons par " $m-2$ " la précision perdue lorsqu'un des voisins n'est pas actif, le " $m$ " étant le nombre de nœuds dans le réseau. Comme l'une des conditions préalables de l'algorithme est que la station de base et les cluster-heads sont toujours actifs, la satisfaction de cette condition est garantie par les contraintes  $C1DNk$  et  $C2DNk$ . En outre, le " $laN$ " ne sera jamais supérieur à un.

L'algorithme suivant est une amélioration de la proposition [5]. Notre amélioration considère le mode de communication pour déterminer la distance entre le nœud et son CH ou le nœud et son prochain nœud, pour faciliter le calcul de la latence. Nous ajoutons le seuil de distance pour supprimer les nœuds qui ont une faible capacité d'énergie et qui ne peuvent envoyer leurs données.

---

### Algorithme 4.1          Calcul de la précision perdue $La_N$

---

$S$  : l'ensemble des nœuds capteurs.  
 $N_{SN}$ : Un ensemble de voisins pour chaque nœud  $SN \in S$ ;  
 $La_N$ : La précision perdue à partir des capteurs des nœuds désactivés;  
 $thld$  : la distance optimale entre deux nœuds voisins;  
 $N_{CH}$ : l'ensemble des CH;  
 $dist$ : la distance;  
 $com$ : mode de communication ( 0  $\rightarrow$  un seul saut, 1  $\rightarrow$  multi-saut) ;  
 $N_{cluster}$ : les membres de cluster "k".

```
1:  $La_N \leftarrow 0$ 
2:  $m \leftarrow size$ 
3:  $com \leftarrow communication$ 
4:   for all ( $SN \in S$ ) do
5:     if ( $com == 1$ ) then
6:        $dist = dist(SN, N_{SN})$ 
7:     else
```

## Chapitre 4 : Approches d'auto-adaptation dans les RCSF hétérogènes

---

```
8:         dist = dist(SN, Ncluster)
9:     end if
10:
11:     if (SN == NCH || dist >= thld) then
12:         SN:activate() = true
13:     else
14:         SN:activate() = false
15:     end if
16:
17:     if (com == 1) then
18:     if (SN:activate() = false) then
19:         LaN ← LaN + (1/m)
20:         SN:visit ← (true)
21:         for all (NE ∈ NSN) do
22:             if (NE:visited() = false and NE:active() = false) then
23:                 LaN ← LaN + (1/m2)
24:                 NE:visit(true)
25:             end if
26:         end for
27:     end if
28:     else /*com = 0
29:     if (SN:activate() = false && SN:visit = false) then
30:         LaN ← LaN + (1/m)
31:         SN:visit ← (true)
32:     end if
33: end for
```

---

Afin de prolonger la durée de vie du réseau, il convient d'utiliser des méthodes éco-énergétiques adaptées aux caractéristiques des réseaux de stockage.

Ainsi, en fonction de l'énergie initiale, Smaragdakis et al., ont proposé un schéma SEP [9] pour un réseau de capteurs sans fil hétérogènes à deux niveaux composé de deux types de nœuds. Les nœuds sophistiqués étaient équipés de plus d'énergie que les nœuds normaux au début du déploiement. Nous utilisons cette méthode en ajoutant des nœuds sophistiqués qui ont une plus grande capacité d'énergie que les nœuds normaux, avec un déploiement déterministe, ce qui améliore l'efficacité et prolonge la vie du réseau. Dans ce cas, nous obtenons l'auto-adaptation dans des protocoles de routage hétérogènes pour les WSN hétérogènes. Selon notre connaissance, ce concept est le premier et est unique dans le cadre des réseaux de capteurs; cela n'a jamais été abordé auparavant.

Le modèle proposé ici pour les RCSF hétérogènes considère deux types de nœuds: les nœuds normaux et les nœuds sophistiqués. On suppose que  $N$  est le nombre total des capteurs du réseau, il y a  $(1-\alpha)*N$  nœuds normaux et  $\alpha*N$  nœuds sophistiqués. Les nœuds

## Chapitre 4 : Approches d'auto-adaptation dans les RCSF hétérogènes

sophistiqués ont  $\gamma$  fois plus d'énergie que les nœuds normaux. Ensuite, les énergies initiales des nœuds normaux et des nœuds avancés sont respectivement  $E_{init}$  et  $\gamma * E_{init}$ . Par conséquent, l'énergie totale du réseau est la suivante

$$\begin{aligned}
 E_{total} &= (1 - \alpha) * N * E_{init} + \alpha * N * \gamma * E_{init} \\
 &= N * E_{init} (1 - \alpha + \alpha * \gamma) \\
 &= N * E_{init} + N * E_{init} * \alpha * (\gamma - 1) \quad \dots\dots (4.7)
 \end{aligned}$$

Cela montre que le réseau hétérogène a  $\alpha * (\gamma - 1)$  plus de nœuds normaux virtuels que de réseaux homogènes avec l'énergie  $N * E_{init}$ .

### ➤ Définition du seuil de distance

Le seuil de distance est la distance idéale pour laquelle l'énergie moyenne est consommée. Notre modèle d'expérimentation est établi sur 100 nœuds standards avec le nœud 101 comme station de base. Au début de la simulation, le budget énergétique initial est égal à 2 joules pour tous les nœuds capteurs, puis nous ajoutons des nœuds sophistiqués de 16 joules (nous utilisons l'idée de Smaragdakis et al.). Cela implique qu'on a réalisé l'hétérogénéité des nœuds capteurs par l'ajout de nœuds qui ont plus de capacité énergétique, et l'hétérogénéité de déploiement parce qu'on utilise l'idée des auteurs avec un déploiement déterministe. Par contre, le premier déploiement est effectué de manière aléatoire.

Round	Noeud	Energie (joules)	Nœud suivant	Distance (m)	CH
20	0	0,18373121	1	4,74	0
20	1	0,205422126	88	15,14	0
20	3	0,250111845	55	2,87	0
20	4	0,222322356	87	7,46	0
20	5	0,212733536	56	3,91	0
20	6	0,251467103	94	3,79	0
20	7	0,250145264	35	2,07	0
20	8	0,239376785	51	13,44	0
20	9	0,225807924	34	4,36	0
20	10	0,198174069	81	1,76	0
20	11	0,213898216	10	5,41	0
20	12	0,884246803	101 (BS)	125,19	1

## Chapitre 4 : Approches d'auto-adaptation dans les RCSF hétérogènes

20	13	0,236721021	44	6,22	0
20	14	0,205415191	17	7,37	0
20	15	0,222250309	61	2,44	0
20	16	0,169038726	26	2,53	0
20	17	0,201629752	29	2,93	0
20	18	0,224707607	13	6,41	0
20	19	0,253928005	90	24,47	0
20	20	0,248901357	7	1,55	0
<b>Seuil de distance</b>			4.11375 m		
<b>Seuil d'énergie</b>			0.219815968 joule		

Tableau 4.2. Détermination du seuil distance et énergie

Ensuite, nous considérons un tour (round) aléatoire et nous étudions un échantillon de nœuds pour déterminer le seuil de distance. Dans notre cas, 20 nœuds sont distribués aléatoirement, ce qui est équivalente à 20% du nombre total de nœuds capteurs au début de la simulation ( $100 * 20\% = 20$  nœuds). Ce pourcentage représente exactement un cluster. Nous choisissons des nœuds provenant de différents clusters, c'est-à-dire que ces nœuds sont sélectionnés à partir de différents protocoles. Ce faisant, nous pouvons calculer la valeur adéquate du seuil de distance en fonction de l'énergie consommée, quel que soit le protocole utilisé. Dans ce cas, nous pourrions calculer la valeur adéquate du seuil de distance en fonction de l'énergie consommée.

A partir du *tableau 4.2*, nous remarquons que l'énergie consommée par chaque nœud est non seulement liée à la distance, mais aussi à la quantité d'informations transférées, et peut être généralisée par rapport au protocole de routage utilisé dans le cluster.

Dans notre travail, nous calculons le seuil de distance et le seuil d'énergie à partir de nœuds qui ne sont pas cluster-head, car ces derniers sont toujours en mode veille et responsables d'agrégation et transmission des données avec une grande distance par rapport aux nœuds ordinaire ; ce implique que les CHs consomment beaucoup d'énergie. Nous prenons un échantillon aléatoire de 20% des nœuds dans le RCSF, nous trouvons dans notre choix trois cluster-heads identifiés respectivement (88, 51 et 90) et la SB qui prend le numéro 101. Le résultat de l'étude figure dans *le tableau 4.2*. Après avoir analysé la situation, nous avons obtenu des résultats qui nous permettent d'évaluer le seuil de notre travail. Nous avons trouvé le taux d'énergie égal à 0.219815968 joule et le seuil de distance égal à 4.11375 m dans chaque tour (round).

### 6. Le processus de migration

Tout d'abord, nous nous sommes concentrés sur le protocole de routage HEEP, ce protocole a été réalisé au sein de notre laboratoire de recherche L@STIC, après avoir concentré notre étude sur un large groupe de protocoles tels que LEACH, LEACH-C, PEGASIS, H-PEGASIS, HEED, TEEN, APTEEN, ...etc.

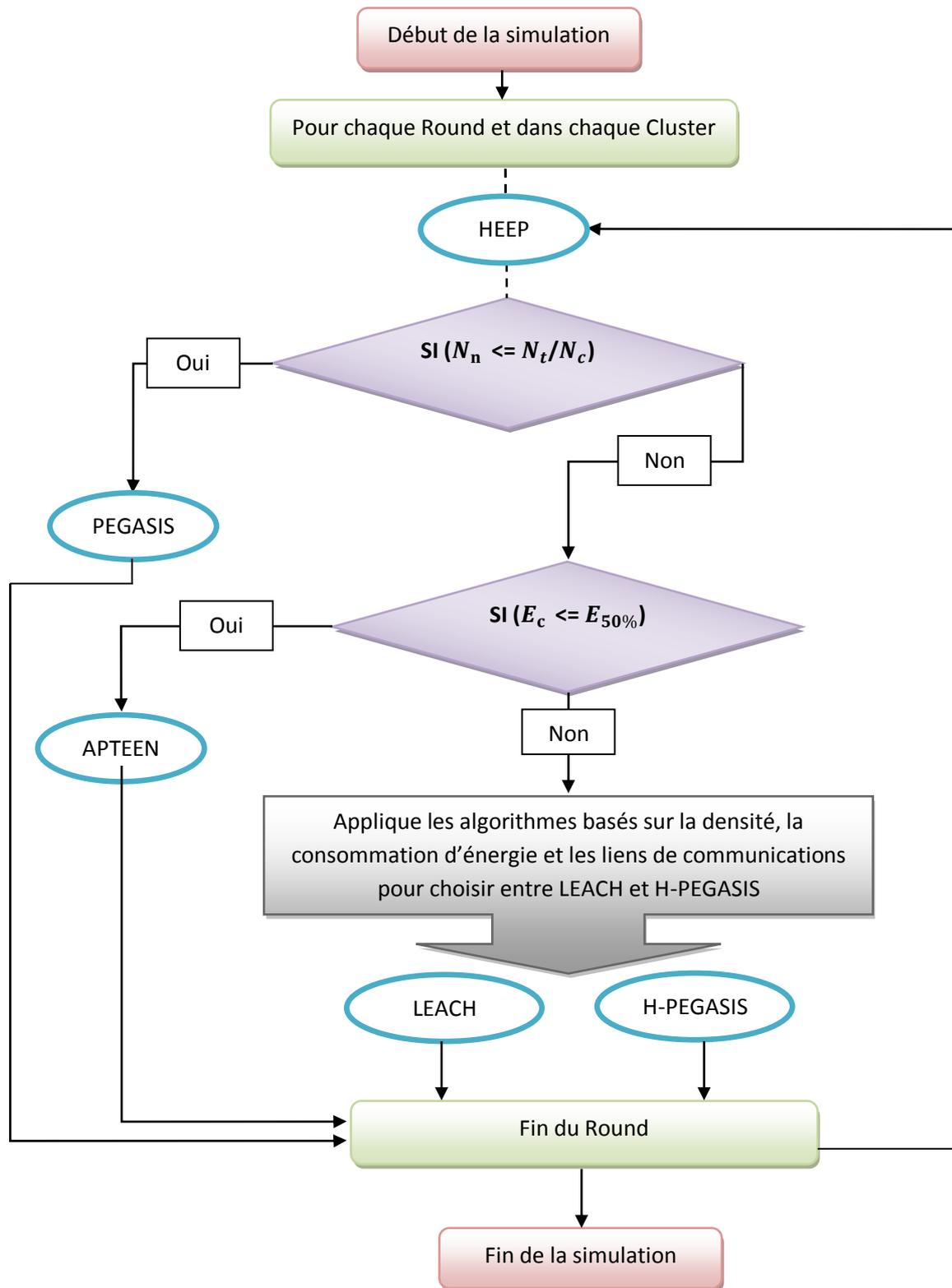
Nous notons que dans les deux protocoles (LEACH et LEACH-C), la quantité d'informations envoyées à la SB est très élevée par rapport à d'autres protocoles, mais la consommation d'énergie est très élevée. Dans ce cas, nous utilisons (LEACH et LEACH-C) lorsque le RCSF a une grande capacité d'énergie, l'avantage est d'envoyer une quantité de données maximale aux stations de base (*Figure 4.5*). Avec la notion d'hétérogénéité, le problème de surcharge pour les CH est limité par la capacité d'énergie réservée à ce type de nœuds.

Le mode de communication, l'efficacité énergétique et la quantité d'informations envoyées à la station de base dans le protocole HEED sont presque identiques avec LEACH et LEACH-C (la différence est négligeable). Dans ce cas, nous choisissons un protocole de communication unique qui utilise un seul saut (soit LEACH, soit LEACH-C ou HEED).

Le protocole PEGASIS atteint rapidement ses limites d'exploitation dans le contexte des réseaux très denses, mais augmente la durée de vie du réseau deux fois par rapport au protocole LEACH. Pour cela, si le nombre de nœuds capteurs descend au-dessous d'un seuil prédéterminé, c'est-à-dire le nombre de capteurs dans un cluster lors du démarrage de l'application divisé par le nombre de clusters;  $\text{threshold} = N_{\text{total}} / N_{\text{clusters}}$ , dans ce cas, nous considérons le réseau comme un cluster unique et la SB remplace le CH de ce cluster (*Figure 4.5*).

Selon la comparaison entre TEEN et APTEEN, nous constatons qu'APTEEN est le protocole de routage le plus proche de HEEP, car il utilise le concept de TDMA dans sa partie de communication compatible avec HEEP. Ces deux protocoles sont plus efficaces en termes d'énergie, mais le nombre de paquets envoyés à la SB est très faible car les capteurs envoient périodiquement les données lorsqu'ils détectent des changements soudains (lorsqu'ils détectent une valeur qui dépasse l'intervalle approprié).

Dans notre travail, la station de base fait une comparaison entre l'énergie totale d'un cluster et la moitié de l'énergie totale dans le même cluster (à partir de l'énergie initiale de chaque nœud au début de la simulation), nous proposons d'utiliser le protocole APTEEN lorsque l'énergie est presque égale à 50% de l'énergie initiale dans le cluster, pour prolonger la durée de vie du réseau (*Figure 4.5*)



$N_n$  : Nombre des nœuds actuel.

$N_t$  : Nombre des nœuds total (début de la simulation).

$N_c$  : Nombre de clusters dans le réseau.

$E_{50\%}$  : 50% d'Énergie total dans ce cluster (début de la simulation).

$E_c$  : Énergie total du cluster.

Figure 4.5. Le processus de migration

A la fin, nous obtenons cinq protocoles de routage hétérogènes, le protocole racine HEEP qui combine les deux protocoles LEACH dans sa phase d'initialisation et PEGASIS dans sa phase de transmission, le protocole LEACH-C qui utilise dans la communication à un seul saut, basé sur la notion de clustering et le mode périodique, le protocole H-PEGASIS pour la communication multi-saut. Basé sur la notion de clustering et le mode périodique, le protocole APTEEN pour les applications événementiel, et PEGASIS qui considère le réseau comme un seul cluster. Mais l'ensemble des cinq protocoles de routage utilisent le concept de TDMA, cet avantage facilite le travail de la station de base et nous donne un RCSF hétérogène.

### 7. Algorithme détaillé d'ADAPTRP

Notre travail adresse plusieurs concepts dont les plus importants sont l'hétérogénéité et l'auto-adaptation. Sur cette base, nous résumons comment le processus de migration des protocoles de routage fonctionne d'une position à une autre suivant des changements dans le RCSF (Figure 4.5). Notre idée est initialement basée sur l'équilibrage entre l'énergie et les données transmises. Après un moment, lorsque la durée de vie du RCSF s'approche de la fin, nous nous sommes concentrés sur comment le réseau va survivre et donc nous avons pris en compte la conservation de l'énergie, pour sélectionner le protocole le plus approprié. Nous avons implémenté un ensemble de processus et mécanismes au début de chaque tour (round).

#### 7.1. ADAPTRP: ADAPTative Routing Protocols

C'est un protocole de routage adaptatif basé sur le concept de TDMA. L'objectif principal de ce protocole est d'améliorer la durée de vie des réseaux de capteurs sans fil hétérogènes en réduisant la consommation d'énergie nécessaire pour créer et adapter des clusters et cluster-heads. L'ADAPTRP utilise un ensemble de mécanismes qui ont été adoptés dans des protocoles bien connus (HEEP, APTEEN, LEACH, PEGASIS et H-PEGASIS) en tenant compte de leurs forces et de leurs faiblesses [127].

Le fonctionnement du protocole ADAPTRP est basé sur des cycles ou *Round*. Chaque cycle est constitué de deux phases : la phase d'initialisation et la phase de transmission.

Pendant la phase d'initialisation, la station de base exécute l'algorithme d'élection de Cluster-Heads (CH), les clusters sont formés, et la station de base sélectionne un protocole de routage la plus appropriée pour ce cluster. La phase de transmission est subdivisée en trames. La trame, à son tour, est subdivisée en time slots. La longueur de la trame est

## Chapitre 4 : Approches d'auto-adaptation dans les RCSF hétérogènes

---

variable d'un cluster à l'autre selon le nombre des nœuds du cluster. Dans cette phase, la durée est comparativement plus longue que celle de la phase d'initialisation.

### ➤ La phase d'initialisation

Le protocole ADAPTRP offre un algorithme d'auto-adaptation pour les RCSF hétérogènes, la station de base sélectionne des cluster-heads en fonction des informations échangées entre les nœuds dans le RCSF. Dans notre protocole, pendant chaque phase d'initialisation la station de base reçoit des informations sur l'état du nœud, l'emplacement et l'énergie résiduelle de chaque nœud. La station de base utilise les informations précédentes pour calculer la valeur moyenne de l'énergie des nœuds de réseau, puis trouve le nombre de cluster-heads et divise le réseau en clusters. Le fonctionnement du protocole à chaque round comprend les étapes suivantes:

Au début de la simulation, le nombre total de nœuds capteurs  $N$  représentent 100%, chaque nœud du réseau envoie les informations concernant le niveau d'énergie résiduelle et l'emplacement à la station de base. Après cela, la station de base calcule le nombre de ces nœuds, si ce nombre est inférieur à  $N / P$ , où  $N$  est le nombre total de nœuds du réseau au début de la simulation et  $P$  est le nombre de cluster-heads. Dans ce cas, la station de base considère le RCSF comme un seul cluster, ce qui signifie que chaque nœud identifie son successeur et son prédécesseur à partir des informations diffusées par la station de base. C'est exactement la migration du protocole HEEP vers le protocole PEGASIS, où la station de base représente le cluster-head de tous le RCSF. Sinon, si le nombre de nœuds dans le RCSF est supérieur à " $N / P$ "; alors la station de base analyse les détails et sélectionne les cluster-heads les plus appropriés pour ce cycle (plus grande capacité d'énergie). Ensuite, dans chaque cluster, la station de base calcule l'énergie résiduelle, si cette énergie diminue à moins de 50% (à partir de l'énergie initiale de chaque nœud au début de la simulation), alors la station de base se concentre sur la conservation de l'énergie plutôt que sur l'envoi des données. Dans pareil cas, la station de base utilise l'approche du protocole de routage APTEEN. Sinon, si l'énergie résiduelle d'un cluster est supérieure à 50% (à partir de l'énergie initiale de chaque nœud au début de la simulation), la station de base se concentre sur l'envoi des données plutôt que sur la préservation de l'énergie; c.à.d. le protocole HEEP migre vers: LEACH ou H-PEGASIS. Pour choisir la migration entre les deux protocoles, la station de base applique les algorithmes basés sur la densité, la consommation d'énergie et les liaisons de communication (*chapitre 4, section 5.1*).

### ➤ La phase de transmission

Cette phase est plus longue que la phase précédente, et permet la collecte de données captées et leur envoi vers la station de base. Dans notre proposition, le RCSF considéré est hétérogène, parce que le protocole de routage ADAPTRP utilise plusieurs protocoles de routage en même temps.

Après un certain temps, le réseau va passer à un nouveau round. Ce processus est répété jusqu'à ce que le nombre des nœuds dans le réseau atteigne  $P$  où  $P$  est le nombre de cluster-heads. Donc, c'est la fin de la vie du réseau est atteinte.

## 8. Conclusion

Dans le chapitre 2 nous avons vu que les protocoles de routage hiérarchiques apportent de nombreux avantages en termes de prolongement de la durée de vie du réseau et de consommation d'énergie. Nous avons donc opté pour la conception d'un protocole de routage adapté aux topologies hiérarchiques. Ce chapitre a pour objectif d'étudier la proposition de protocole ADAPTRP qui consiste à réaliser un mécanisme de migration vers quelques protocoles de routage hiérarchiques. Notre processus de migration est basé sur le principe de multiple objectifs tels que : la densité, l'énergie, la distance et la qualité de service. Pour éviter les objectifs contradictoires et pour choisir la meilleure migration vers un protocole de routage dans un cluster, la station de base, au début de chaque round, utilise les différentes fonctions et procédures étudiées dans *la section 5.1*. Le chapitre suivant traite l'auto-organisation dans les RCSF homogènes.

## **Chapitre 5**

# **Approches d'auto-organisation dans les RCSF homogènes**

### 1. Introduction

Un grand nombre des capteurs peuvent être mis en réseau pour fonctionner sans surveillance pour une variété d'applications : militaires, gestion des catastrophes, domotique, santé, agriculture, ...etc.

Plusieurs recherches se sont focalisées sur l'auto-organisation pour obtenir une utilisation efficace de l'énergie et de la capacité du réseau [23]. Donc, L'auto-organisation est un processus par lequel un ensemble d'éléments, interagissant entre eux, produisent une organisation qui tend à se maintenir sur une certaine durée, sans qu'il y ait eu volonté a priori de produire cette organisation. Le défi est d'organiser de manière dynamique et spontanée les nœuds pour former un réseau ayant une durée de vie maximale, tout en satisfaisant des contraintes de qualité de service.

Nous présentons ici un protocole de routage hybride efficace en énergie et en durée de vie du réseau appelé HEEP-AO (extension du protocole HEEP). Enfin, on traite l'évaluation des performances de la proposition et des résultats obtenus par simulation.

### 2. La proposition d'auto-organisation

L'absence de sécurité physique pour les capteurs, et la nature vulnérable des communications radios sont des caractéristiques qui augmentent les risques de pannes sur le réseau, donc, il est nécessaire de considérer la sécurité [38], l'auto-organisation en cas des pannes dans le réseau comme critères indispensable dans la conception des protocoles.

En outre, dans les RCSF, les nœuds sont sujets à de nombreuses pannes, dues principalement à l'épuisement des batteries et aux destructions physiques. Ainsi, la panne de nœuds entraîne la perte des liens de communication et donc un changement significatif dans la topologie globale du réseau. Ceci peut affecter d'une façon considérable la connectivité du réseau et diminuer, en conséquence, sa durée de vie.

### 3. HEEP-Auto-Organisé (HEEP-AO)

L'objectif principal de notre travail est la mise en œuvre d'une solution qui se charge d'améliorer le protocole de routage HEEP de telle sorte qu'il soit auto-organisé en cas des pannes dans le réseau. Notre premier but est d'atteindre un niveau d'auto-organisation acceptable sans dégrader les performances du réseau. De ce fait, nous avons établi un

nouveau protocole appelé HEEP-AO qui est en mesure de pallier aux limites de HEEP dans un environnement non idéal. Dans notre proposition HEEP-AO, au niveau de la phase de construction des clusters, une élection d'un vice-CH est établie par la SB dans chaque cluster, ce vice-CH prend le rôle de la réception des données des membres du CH principale en cas de panne. (Figure 5.1). Ensuite, nous avons intégré l'auto-organisation en cas de panne au niveau de la station de base, dans ce cas une deuxième station de base (Station de Base Secondaire SBS) sera chargée à la réception des données envoyées par les CHs.

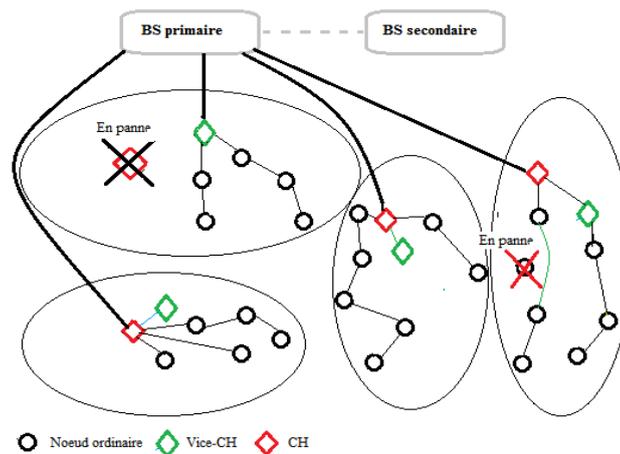


Figure 5.1. Architecture de communication du protocole HEEP-AO.

L'idée proposée est comme suit : La station de base choisit le nœud le plus proche du CH dans chaque cluster, caractérisé par une grande capacité d'énergie parmi les membres du même cluster comme étant le vice-CH. Tous les membres de cluster connaissent le vice-CH choisi, dans ce cas, le vice-CH reste toujours en mode veille (WakeUp) pour capter les données envoyées par les membres du cluster. Pendant le slot time réservé au vice-CH, si le CH reçoit des données de son vice-CH, il lui répond par un message d'acquiescement ACK pour lui indiquer qu'il n'est pas en panne. Dans le cas contraire, si le Vice-CH ne reçoit pas d'acquiescement, il sera alors chargé de transmettre les données reçues vers la SB. En faisant cela, les données sur les nœuds du cluster pourront toujours atteindre la BS; de sorte que nous n'avons plus besoin d'élire un nouveau CH à chaque fois que ce dernier rencontre des problèmes (mort ou panne), il suffit seulement d'utiliser le vice-CH déjà sélectionné.

### 4. Nouvelle Méthode d'Auto-Organisation dans les RCSF

L'auto-organisation peut s'intégrer dans diverses couches de la pile protocolaire d'un RCSF (i.e. au niveau de la couche MAC, réseau, transport ou application). Notre travail a porté sur la couche réseau accomplissant un des services fondamentaux assuré par les RCSF qui ne sont autres que le routage. Sachant que le protocole HEEP sur lequel repose notre solution est de type best effort (i.e. aucune qualité de service), HEEP-AO (HEEP-Auto-organisé) sera en mesure de détecter, localiser les défaillances et organiser automatiquement le réseau (quelque soit leur type de problèmes survenus). Pour ce raison, HEEP-AO sera un protocole de routage hiérarchique multi-saut, intégrant un système d'auto-organisation à HEEP basé sur les algorithmes décrits plus bas.

Dans les protocoles de routage hybrides comme HEEP, on distingue trois types de nœuds : les nœuds ordinaires qui servent au captage, les nœuds CH et la station de base. La panne peut donc se produire au niveau de n'importe quel nœud.

Dans ce qui suit, la description de l'auto-organisation pour les pannes les plus fréquentes dans le protocole HEEP et leurs conséquences sur ce dernier. Parmi les pannes les plus courantes, on distingue: les pannes d'émission, de réception, d'énergie, captage ou de traitement au niveau des différents nœuds.

#### 4.1. Le déroulement d'auto-organisation

##### C'est quoi la panne ?

C'est un cas parmi la liste des quatre principaux problèmes dans les réseaux de capteurs sans fil (Energie, émission, réception, traitement, ou captage).

##### C'est qui ?

Soit la station de base, soit le cluster-Head, ou un nœud ordinaire.

##### L'auto-organisation

Chaque panne a une solution quelque soit le type de nœud (Station de base, cluster-Head ou nœuds ordinaire).

#### 4.2. Les hypothèses

Pour être effectif, un diagnostic cohérent doit tenir compte des hypothèses réalistes. Ainsi, les hypothèses suivantes permettent de mieux cerner les problèmes :

- Le problème de capture d'un nœud est diagnostiqué au niveau du son successeur associé en vérifiant que la valeur de captage reçue appartient à l'intervalle de

vraisemblance (par exemple intervalle de température entre  $-20^{\circ}$  et  $60^{\circ}$ ), ou bien au niveau local s'il n'a aucune valeur captée ;

- Les problèmes d'énergie, réception et traitement pouvant survenir au niveau d'un nœud sont localement diagnostiqués au niveau du nœud défaillant;
- Le problème d'émission peut survenir au niveau d'un nœud est diagnostiqué au niveau du son successeur;
- Un nœud défaillant au niveau captage peut être mis à profit au niveau traitement, stockage et transmission. De même, un défaillant en réception peut être utilisé en capture, émission et stockage. Et un défaillant en traitement peut être utilisé en capture, transmission et stockage.
- Le choix de la période du 2ieme chance dépend de type d'application ;

La période du 2ieme chance : c'est la période donnée à un nœud pour la confirmation de sa défaillance, c'est-à-dire à l'instant « t » si un nœud génère une défaillance de traitement par exemple, on enregistre ce nœud comme un nœud en doute et on attend un temps « r » c'est le temps de 2ieme chance, donc, à l'instant « t+r » on lance de nouveau le traitement si ce nœud en doute génère une autre défaillance dans ce cas, il existe un diagnostic pour que le réseau reste avec un bon fonctionnement sinon pas de problème.

- On ne peut pas utiliser un nœud qui n'envoie pas de données, de même qu'un nœud qui a un problème d'énergie.

## 5. Le mécanisme d'auto-organisation

### 5.1. L'auto-organisation dans le cas d'une panne de réception

➤ **Au niveau d'un nœud ordinaire** : deux types de messages peuvent être reçus par un nœud ordinaire: le message INFO émis par la station de base et DATA émis par le nœud prédécesseur. Si le nœud ordinaire ne reçoit pas les messages « INFO » il agira comme un CH, donc il enverra ses données directement vers la station de base. S'il ne reçoit pas un message de DATA alors ce nœud est enregistré comme un nœud douteux, dans cette situation, on donne une deuxième chance à ce nœud, si le même problème réception revient, alors ce nœud défaillant envoie un message d'alerte vers son propre prédécesseur pour que le prochaine envoie des données sera orienté vers le successeur du nœud défaillant ( *figure 5.2*).

➤ **Au niveau d'un nœud CH** : le seul message reçu par le CH est de type DATA depuis son prédécesseur. Dans ce cas, le CH est enregistré comme un nœud douteux en lui accordant une deuxième chance. Si le même problème de réception persiste alors le vice-CH sera chargé de transmettre les données reçues vers la station de base (Figure 5.2). Notons que le vice-CH est sélectionné dans la phase d'ordonnancement. L'apport de cette stratégie est l'augmentation de nombre de DATA envoyé vers la station de base.

➤ **Au niveau de la station de Base** : la station de base secondaire (SBS) envoie un message Hello vers la station de base primaire (SBP), à la réception de ce message, la SBP répond à ce message pour garantir le bon fonctionnement. Le message reçu par la SB est de type DATA. En cas d'un manque de réponse, les données n'atteindront jamais l'utilisateur final, dans ce cas la SBS sera chargée pour réceptionner les données envoyées par les CHs.

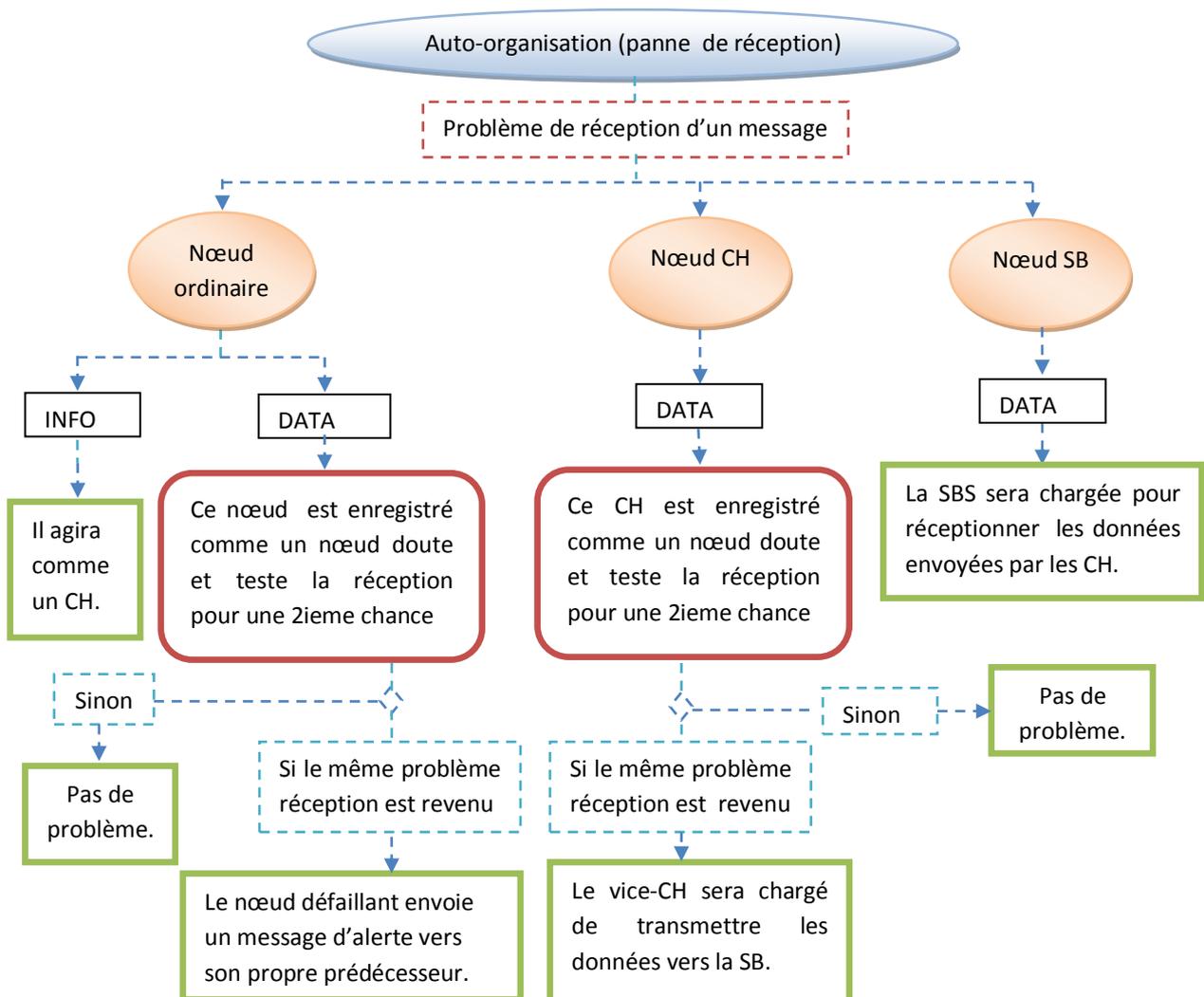


Figure 5.2. Auto-organisation en cas d'une panne de réception.

## 5.2. L'auto-organisation en cas d'une panne d'émission

➤ **Au niveau d'un nœud ordinaire** : les messages envoyés par un nœud ordinaire sont deux types : les messages de type INFOn contenant l'identificateur du nœud ordinaire et sa position dans le réseau et les messages de type DATA. Les nœuds n'ayant pas envoyés de message INFOn ne recevrons pas le time-slot pour l'envoi des données au successeur, ne seront pas rattachés à aucun CH et se comporterons en tant que CH assurant ainsi l'envoi de leurs données par eux-mêmes. En cas de non envoi d'un message DATA pendant son time-slot, le nœud ordinaire sera marqué comme un nœud douteux dans une structure de données. Le successeur de ce nœud défaillant lui donne une deuxième chance et si le même problème persiste, le successeur envoie un message vers le prédécesseur du nœud défaillant pour l'informer de lui adresser les données.

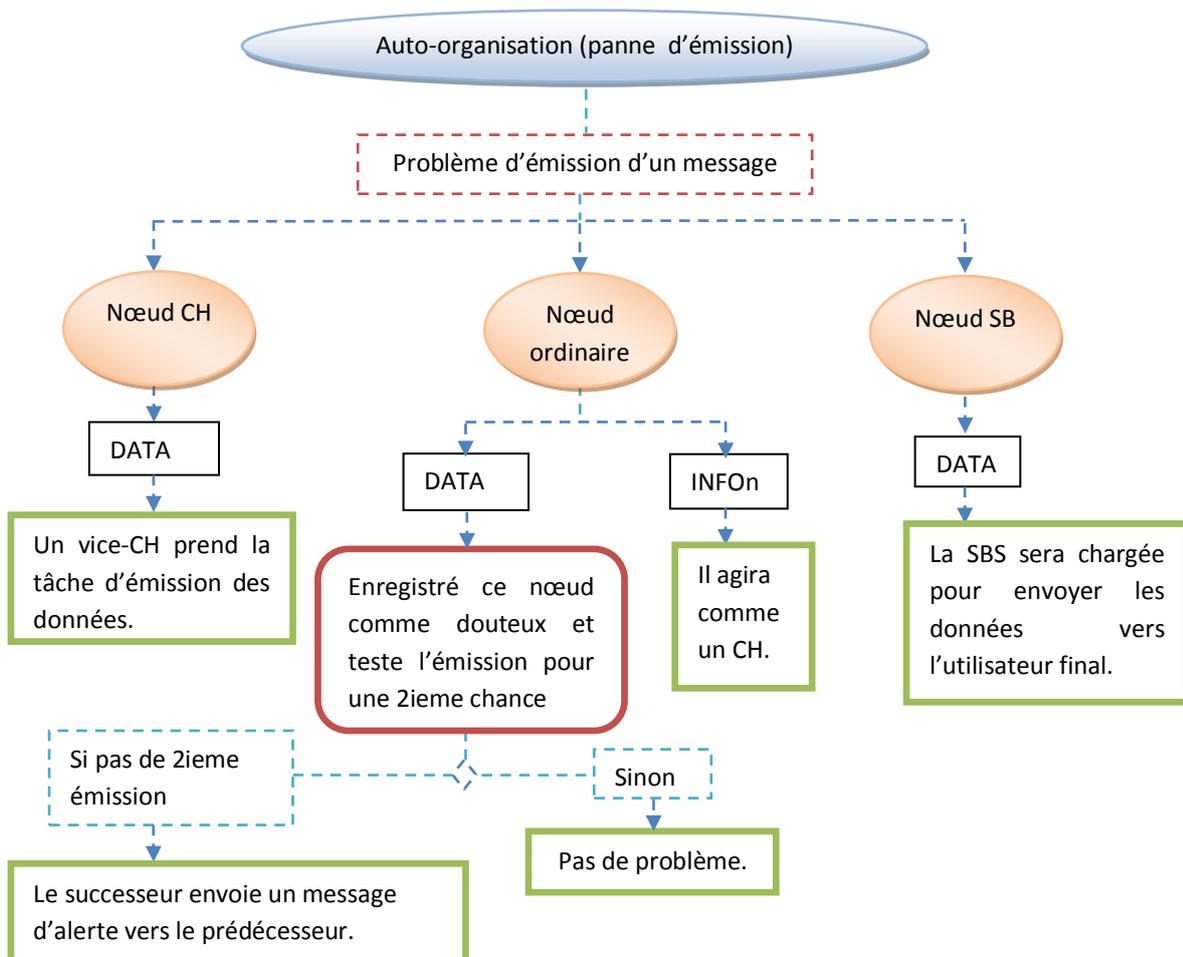


Figure 5.3. Auto-organisation en cas d'une panne d'émission.

➤ **Au niveau d'un nœud CH** : le message DATA est le seul message envoyé par un nœud CH vers la station de base. En cas de problème avec le CH (pas d'envoi de messages DATA), un vice-CH prend la tâche d'émission des données.

➤ **Au niveau de la station de base** : le message envoyé par la station de base est de type DATA. Si cette dernière n'a pas envoyé le message DATA, les données n'atteindront jamais l'utilisateur final. Dans ce cas une station de base secondaire (SBS) prendra le relais pour émettre les données (Figure 5.3).

### 5.3. L'auto-organisation en cas d'une panne d'énergie

L'avantage principal d'auto-organiser le réseau dans le cas d'une « panne d'énergie » est l'augmentation du nombre de DATA envoyés vers la station de base. Il est à noter que la station de base ne peut pas tomber en panne dans ce cas (panne d'énergie) parce qu'elle dispose d'une réserve d'énergie illimitée.

Au niveau du cluster, quand un nœud donné atteint un seuil critique d'énergie, il est déclaré comme un nœud défaillant dit nœud *crash*.

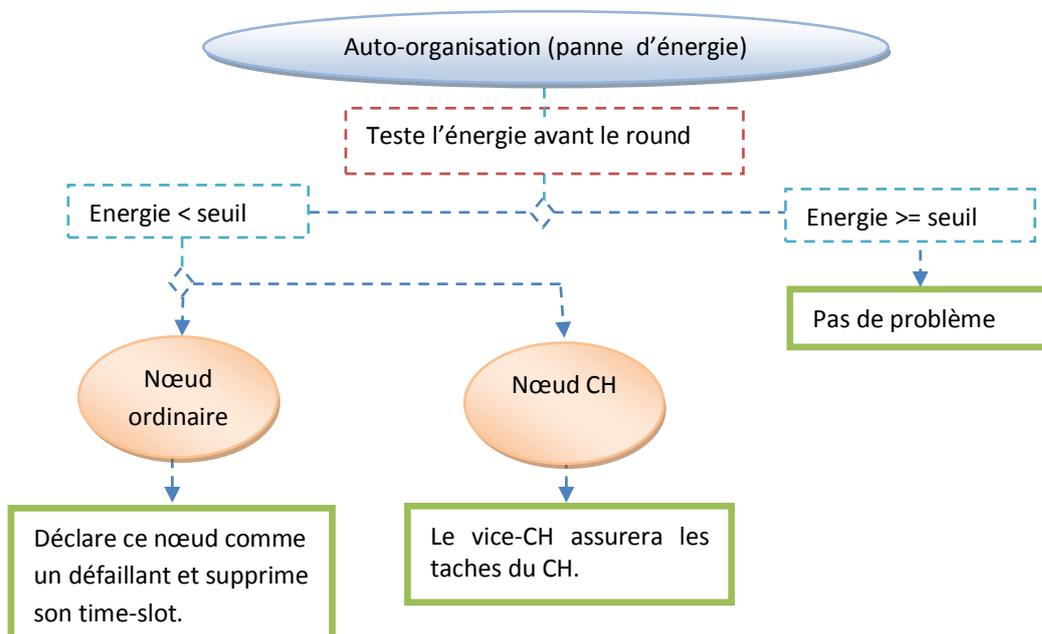


Figure 5.4. Auto-organisation en cas d'une panne d'énergie.

➤ **Au niveau d'un nœud ordinaire** : Si le nœud en panne est un nœud ordinaire alors la station de base supprime ce nœud et enlève le time-slot d'envoi (dans le TDMA)

pour gagner un espace de transmission (on peut gagner un ou plusieurs frame d'envoi de plus).

➤ **Au niveau d'un nœud CH** : en cas d'un manque d'énergie au niveau de ce nœud, c'est le vice-CH qui assure les tâches du CH (Figure 5.4).

### 5.4. L'auto-organisation en cas d'une panne de capture

Généralement, un nœud qui ne capte pas n'envoie pas, dans ce cas on a besoin de faire appel au diagnostic de transmission et exactement au diagnostic d'émission.

Pour les applications utilisant des valeurs capturées proches (i.e. température) le problème ne se pose pas, mais si la différence entre les valeurs des nœuds voisins dépasse un certain seuil, deux cas sont possibles. Le premier, si le nœud est un cluster-head alors il fonctionne normalement (i.e. reçoit des DATA et les envoie vers la station de base). Le deuxième cas, le nœud est ordinaire : le diagnostic se fait au niveau de son successeur ; si ce successeur est un CH alors, il envoie un message d'alerte à la station de base, sinon il envoie un message d'alerte contenant l'identificateur du nœud défaillant vers son CH, ensuite ce dernier renvoie ce message vers la station de base. Si le nombre de messages d'alerte augmente au niveau de la station de base alors, il faut procéder à un contrôle au niveau de la zone où le problème est apparu (Figure 5.5).

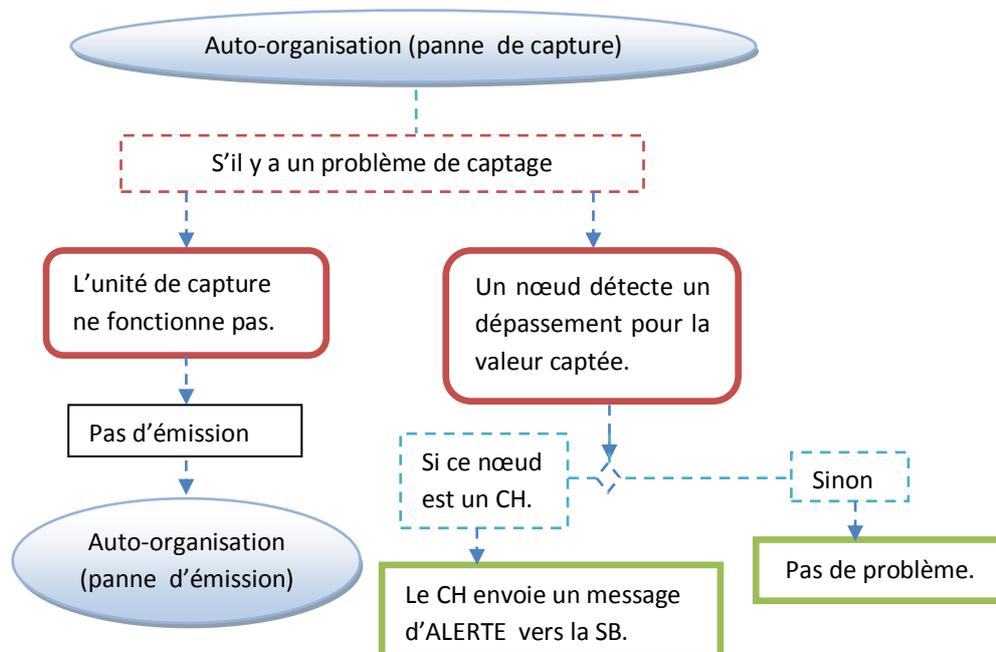


Figure 5.5. Auto-organisation en cas d'une panne de capture.

### 5.5. L'Auto-Organisation en cas d'une Panne de traitement

Chaque nœud exécute un calcul pour chaque début de round dans le cadre d'une vérification de l'unité de traitement, pour cela, on prend par exemple : le calcul de  $(a+b)$  ?

Chaque nœud fait le calcul à son niveau ensuite, il compare le résultat obtenu avec le résultat déclaré au départ.

Par exemple, dans chaque nœuds on déclare  $(a=5)$ ,  $(b=7)$ ,  $(c=12)$ , ensuite, chacun fait son propre calcul «  $a + b$  » et compare avec «  $c$  » si les résultats sont identiques, aucun problème n'est a signalé, sinon les nœuds erronés sont enregistrés pour leur donner une deuxième chance. Ensuite, si le résultat d'un nœud enregistré est vrai alors il n'y'a pas de problème, sinon le nœud défaillant ne sera plus accepté comme Cluster-Head ou bien une station de base et dans ce cas alors le vice-CH ou la deuxième station de base prend les taches dans le réseau, mais dans le cas ou le nœud défaillant est un nœud ordinaire, il est utilisé pour le captage, la transmission ou un espace de stockage et on l'utilise pas pour le traitement (Figure 5.6).

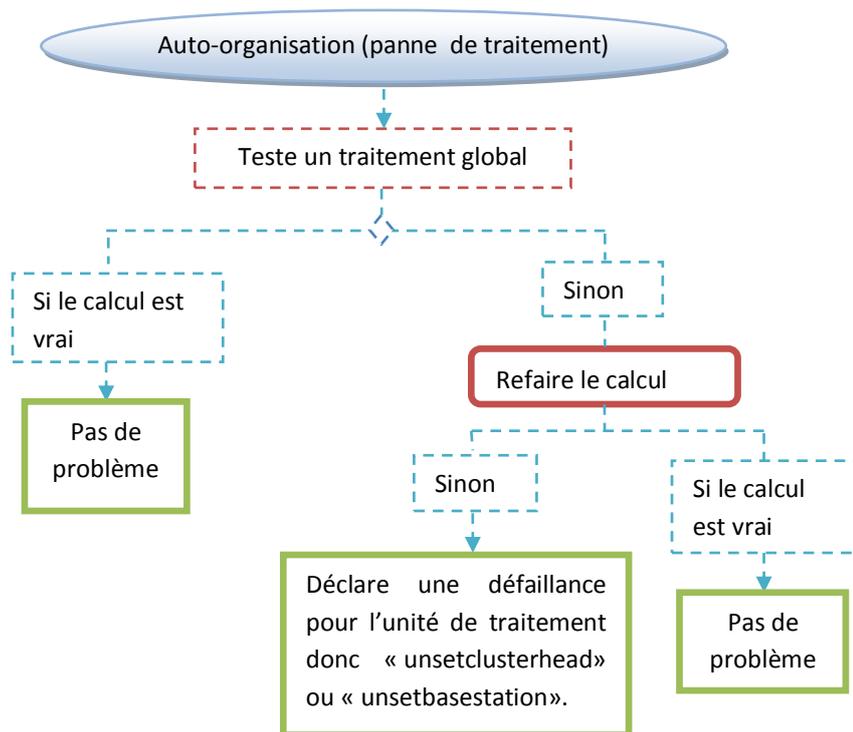


Figure 5.6. Auto-organisation en cas d'une panne de traitement.

### 6. Conclusion

Dans ce chapitre, une méthode d'auto-organisation a été proposée (HEEP-AO) pour les nœuds défaillants dans les RCSF basée sur le protocole de routage HEEP. Cette méthode consiste à traiter les défaillances dans le réseau après détection et localisation d'une panne. Les résultats des simulations dans le chapitre suivant ont montré l'efficacité de la proposition au regard du protocole de référence utilisé. Le chapitre suivant traite l'implémentation des deux protocoles au niveau de l'environnement de simulation NS2 sous Ubuntu 10.10.

# **Chapitre 6**

## **Mise en œuvre et simulation**

### 1. Introduction

Tel qu'on l'a montré au cours de l'étape de conception, l'objectif principal de notre première contribution est la mise en œuvre d'une solution qui se charge d'auto-organiser le protocole de routage HEEP. Notre premier but est d'atteindre un niveau d'auto-organisation acceptable sans dégrader les performances du réseau. De ce fait, nous avons établi un nouveau protocole **HEEP-AO** qui est en mesure de détecter et localiser les défaillances dans le protocole de routage HEEP.

Dans notre deuxième contribution, le protocole **ADAPTRP** est le résultat de notre problématique dans le cadre de la mise en œuvre d'une solution efficace pour réaliser l'auto-adaptation dans les RCSF hétérogènes.

L'objectif de ce chapitre est donc de démontrer l'efficacité du protocole HEEP-AO par rapport au protocole racine HEEP, et une autre comparaison de notre proposition ADAPTRP avec les protocoles HEEP, LEACH, PEGASIS, HPEGASIS et APTEEN. Nous avons pu montrer à travers les résultats de simulation obtenus que les objectifs de nos protocoles aient été atteints.

Pour cela, nous commencerons par définir les outils nécessaires pour l'implémentation et la simulation des deux protocoles. Ensuite, nous décrirons la mise en œuvre de toutes les structures de données et processus décrits lors de la conception. Nous terminerons ce chapitre par une présentation des résultats relevés lors des tests de performances des deux protocoles HEEP-AO et ADAPTRP.

La simulation des réseaux de capteurs consiste principalement en la reproduction du comportement et du fonctionnement des nœuds capteurs dans un environnement informatique.

Il sera donc question, pour ce chapitre, de faire la présentation d'un simulateur NS2, puis aborder les simulateurs eux-mêmes.

### 2. Environnement de simulation

Dans cette section, nous présentons les outils utilisés pour la mise en œuvre des protocoles HEEP-AO et ADAPTRP. Nous commençons tout d'abord par le simulateur NS2 [128] pour simuler et évaluer les différents paramètres tels que les données envoyées, l'énergie consommée et le nombre des nœuds vivant. Nous parlons ensuite du UBUNTU, le système d'exploitation adapté aux RCSF. Nous terminons cette partie par la présentation du

langage de programmation TCL avec lequel nous avons programmé les codes des deux protocoles.

### 2.1. Objectif de la simulation

Le déploiement d'un réseau de capteurs nécessite une phase de tests avant sa mise en place afin de s'assurer du bon fonctionnement du réseau. Une expérimentation directe effectuée sur le terrain peut se révéler coûteuse, irrationnelle ou même impossible. Il serait même inconcevable dans notre étude de mettre en œuvre un RCSF, de changer les paramètres pour comparer deux ou plusieurs algorithmes de routage.

La simulation des réseaux de capteurs consiste principalement en la reproduction du comportement des nœuds capteurs et des interactions entre eux. C'est une étape incontournable pour l'évaluation des modèles d'application ou des protocoles de communication. De plus, la simulation offre un gain considérable en temps, une flexibilité en permettant la variation des paramètres et une meilleure visualisation des résultats.

### 2.2. Choix du simulateur NS2

Beaucoup de simulateurs pour les réseaux informatiques ont été développés pour répondre aux attentes des utilisateurs. Parmi ces simulateurs, nous pouvons citer : OPNET, QualNet, NetSim, NS2, OMNeT++, ...etc. Nous avons choisi le simulateur NS2 (Network simulateur, version 2).

NS-2 est un logiciel de simulation open source gratuit (libre) à événements discrets. Il est développé dans le cadre du projet VINT qui regroupe plusieurs laboratoires de recherche comme AT&T institut de recherche à Berkeley (ACIRI), Xerox PARC et Sun Microsystems.

Ce simulateur est développé en C++ et OTCL (Object Tools Command Language), est utilise le langage TCL comme langage de création des scénarios de simulation. Il supporte les réseaux sans fil et filaires, avec plusieurs protocoles des différentes couches (Physique, MAC, réseaux, transport ... etc.). L'outil NAM (Network Animator) associé au simulateur NS permet de visualiser des animations de la simulation (transfert des paquets d'un nœud à un autre, taille des paquets, remplissage des files d'attente, ...etc.)

#### ➤ Avantages

- ✓ NS est un logiciel de simulation multicouches ;
- ✓ Son développement est orienté objet ;
- ✓ Permet l'ajout de composants à la demande ;
- ✓ C'est un outil complètement libre et disponible pour plusieurs plateformes ;

- ✓ C'est l'un des simulateurs gratuits les plus documentés ;
- ✓ Vu sa popularité, plusieurs protocoles sont à priori disponibles, aussi bien pour les réseaux ad-hoc que pour les réseaux filaires.

### ➤ Inconvénients

- ✓ Initialement conçu pour les réseaux filaires, le support des réseaux sans fil a été ajouté ultérieurement ;
- ✓ Contient peu de paramètres de configuration dans ses modèles standards ;
- ✓ Introduit des dépendances entre les modules ce qui rend difficile l'ajout de nouveaux modèles;
- ✓ Les performances du simulateur sont assez limitées et ne semblent pas adaptées pour des simulations de réseaux importants.

### 2.3. Choix de plateforme OS

NS2 étant disponible sous deux grandes plateformes OS :

- a. Sous Windows : en l'installant sous un émulateur tel que Cygwin.
- b. Sous Unix.

Après étude des deux environnements, nous avons choisi Linux-UBUNTU. Ce dernier offre tous les outils logiciels nécessaires (langages, outils graphiques, ...etc.).

### 2.4. Le choix du langage

Nous avons choisi comme langage d'implémentation de nos protocoles HEEP-AO et ADAPTRP le langage TCL (Tool Command Language). Ce dernier est connu comme étant un langage de commandes interprétées extensif. En effet, les programmes écrits en TCL sont des fichiers texte constitués de commandes TCL qui sont traitées via un interpréteur TCL au moment de l'exécution. L'avantage d'implémenter notre algorithme en TCL est de pouvoir facilement l'interpréter avec l'interpréteur TCL, ce qui nous permet de simuler son fonctionnement afin d'évaluer ses performances.

Nous avons choisi comme plate-forme d'implémentation, le simulateur NS2 version NS2.34, sous le système d'exploitation LUNIX UBUNTU 10.10.

## 3. Etapes d'implémentation de notre protocole

L'implémentation de notre protocole passe par plusieurs étapes à savoir :

### 3.1. Préparation de l'environnement d'implémentation

La préparation de l'environnement d'implémentation consiste à installer le simulateur de réseau NS2 sous le système d'exploitation UBUNTU. On a choisi la version NS2.34 puisqu'elle prend en considération la topologie des réseaux sans fil et le protocole que nous étudions HEEP se base sur ce simulateur.

L'installation de NS2.34 sous UBUNTU 10.10 s'effectue suivant les étapes qui se résument en :

- ✓ télécharger le paquet ns-allinone-2.34.tar.gz
- ✓ placer le paquet ns-allinone-2.34.tar.gz dans le home par exemple (/home/dadi/ns-allinone-2.34.tar.gz)
- ✓ Cliquer à droite de la souris sur ns-allinone-2.34.tar.gz ensuite choisir extrait here pour extraire le contenu dans le même répertoire.

- ✓ Ouvrir le terminal (Applications→Accessories→Terminal)

- ✓ Changer vers la direction de ns-allinone-2.34

```
$ cd /home/dadi/ns-allinone-2.34
```

- ✓ Taper la commande suivante :

```
$ sudo apt-get install build-essential autoconf automake libxmu-dev gcc-4.3
```

- ✓ Ouvrir Makefile.in qui existe dans la localisation ns-allinone-2.34/otcl-1.13/Makefile.in trouver la ligne suivante :

```
CC= @CC@
```

et changer vers:

```
CC= gcc-4.3
```

- ✓ Pour devenir un super utilisateur, taper la commande suivante :  
\$ sudo su

- ✓ Pour démarre l'installation taper «./install »

- ✓ Si l'installation est terminé sans erreurs, il faut ajouter les variables d'environnement dans un fichier « .bashrc »

```
$ gedit ~/.bashrc
```

- ✓ Ajouter les lignes suivantes à la fin de ce fichier

```
# LD_LIBRARY_PATH
```

```
OTCL_LIB=/home/dadi/ns-allinone-2.34/otcl-1.13
NS2_LIB=/home/dadi/ns-allinone-2.34/lib
X11_LIB=/usr/X11R6/lib
USR_LOCAL_LIB=/usr/local/lib

export
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$OTCL_LIB:$NS2_LIB:$X11_LIB:$USR_LOCAL_LIB

# TCL_LIBRARY

TCL_LIB=/home/dadi/ns-allinone-2.34/tcl8.4.18/library
USR_LIB=/usr/lib
export TCL_LIBRARY=$TCL_LIB:$USR_LIB

# PATH

XGRAPH=/home/dadi/ns-allinone-2.34/bin:/home/dadi/ns-allinone-
2.34/tcl8.4.18/unix:/home/dadi/ns-allinone-2.34/tk8.4.18/unix
# Note: the above two lines starting from XGRAPH should come in the same line
NS=/home/dadi/ns-allinone-2.34/ns-2.34/
NAM=/home/dadi/ns-allinone-2.34/nam-1.14/
PATH=$PATH:$XGRAPH:$NS:$NAM
```

- ✓ A la fin taper \$ source ~/.bashrc
- ✓ Installer xgraph à l'aide de la commande: sudo apt-get install xgraph

Après avoir installé le simulateur de réseau NS2.34, il faut installer le protocole de routage HEEP. Pour ce faire, il faut suivre les étapes suivantes :

- ✓ Obtenir le package d'installation du protocole HEEP « ns-234-heap.tar.gz »

Le paquet est disponible dans notre laboratoire de recherche L@STIC.

- ✓ Obtenir le fichier bash « heap-setup.sh »

Le fichier est disponible dans notre laboratoire L@STIC.

- ✓ placer les deux dans le chemin suivant : « /home/dadi/ns-allinone-2.34/ns-2.34 »

- ✓ décompresser le package d'installation dans le même répertoire « ns-2.34 »

✓ on remarque que le répertoire d'accueil de « ns-2.34 » est autre que "/opt/...", alors nous avons besoin de trouver "/opt/ns-allinone-2.34" et le remplacer par "/home/dadi/ns-allinone-2.34" pour le dossier "Heap-setup.sh".

✓ déplacer vers le répertoire `"/home/dadi/ns-allinone-2.34/ns-2.34"` et patch le fichier `"Heep-setup.sh"`

```
wsn@ubuntu:~$ cd /home/dadi/ns-allinone-2.34/ns-2.34/
```

```
wsn@ubuntu:/home/dadi/ns-allinone-2.34/ns-2.34$ sudo bash heep-setup.sh
```

✓ trouver `"/opt/ns-allinone-2.34"` et le remplacer par `"/home/dadi/ns-allinone-2.34"` pour les deux fichiers:

«Makefile»: on le trouve dans le répertoire `"/home/dadi/ns-allinone-2.34/ns-2.34"`.  
«Makefile.in»: on le trouve dans le répertoire `"/home/dadi/ns-allinone-2.34/ns-2.34"`.

✓ remplacer CC et CPP de « Makefile » et « Makefile.in » existe dans « ns-234-heap » par

```
CC = gcc-4.3
```

```
CPP = g++-4.3
```

✓ Exécuter les commandes suivantes :

```
wsn@ubuntu:/opt/ns-allinone-2.34/ns-2.34$ ./configure
```

```
wsn@ubuntu:/opt/ns-allinone-2.34/ns-2.34$ make clean
```

```
wsn@ubuntu:/opt/ns-allinone-2.34/ns-2.34$ make
```

✓ Si l'exécution de la commande précédente est réussie alors on peut tester HEEP par l'exécution de la commande « `./test` »

```
wsn@ubuntu:/opt/ns-allinone-2.34/ns-2.34$ ./test
```

✓ Vérifier le fichier `"heap.err"`, s'il n'y a pas d'erreurs, aller vers le fichier `"heap.out"` et voir le résultat.

### 3.2. Implémentation

Après avoir préparé notre environnement d'implémentation, on entame l'étape d'implémentation de nos protocoles qui consiste à modifier le code d'implémentation du protocole HEEP afin qu'il corresponde à l'architecture de fonctionnement de nos protocoles. Pour cela, on a ajouté diverses fonctions et procédures pour l'auto-organisation ou l'auto-adaptation.

En effet, le code d'implémentation proposé est composé de plusieurs fichiers TCL dont chacun joue un rôle bien précis. Ainsi, et pour réutiliser ce dernier, nous avons dû analyser et comprendre le rôle de chacun de ces fichiers. Etant donné qu'un fichier TCL est présenté

comme une classe qui contient plusieurs sous classes, les protocoles proposés sont vu comme un ensemble de classes qui se communiquent pour réaliser une tâche commune.

Les protocoles proposés sont des applications spécifiques des protocoles d'auto-organisation et d'auto-adaptation. Il est implémenté comme sous classe par rapport à la classe Applications du simulateur NS2.

### 4. Évaluation des performances du protocole ADAPTRP

Cette section présente les paramètres et les résultats de la simulation, dans notre travail, nous utilisons le simulateur NS2 avec une centaine de nœuds répartis d'une façon aléatoire et déterministe dans une zone de 100m \* 100m, les coordonnées de position de la station de base sont SB (50,175) comme indiqué dans *la figure 6.1*. Nous supposons que tous les nœuds ont une position fixe pendant toute la période de simulation. Les paramètres de notre simulation sont donnés dans *le tableau 6.1*.

Nous avons effectué la simulation pour évaluer les performances de notre mécanisme d'auto-adaptation sur les cinq protocoles de routage: HEEP, LEACH, PEGASIS, H-PEGASIS, APTEEN et collecter tous ces protocoles dans un protocole nommé ADAPTRP: ADAPTative Routing Protocols.

ADAPTRP donne de meilleurs résultats en ce qui concerne le nombre de données envoyées à la station de base, le nombre de nœuds défaillants détectés et l'énergie consommée. Les résultats sont illustrés dans les figures suivantes.

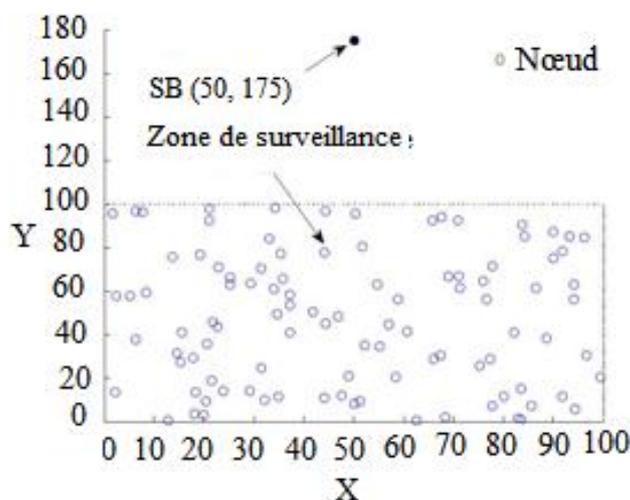


Figure 6.1. Positions des nœuds et de la station de base

<b>Paramètres</b>	<b>Valeur</b>
Localisation de la SB	(50,175)
Nombre des noeuds	100
Nombre de clusters	5
Nombre des nœuds sophistiqués	20
La taille des paquets	512 Bits
La taille des paquets de contrôle	8 Bits
Energie des nœuds initiaux	70 nœuds →2J BS → 5000J
Energie des nœuds ajoutés	30 nœuds →16J
Simulator	NS2

Tableau 6.1 : paramètres de simulation

Dans les figures du résultat de la simulation, les deux protocoles LEACH et H-PEGASIS ne sont pas listés pour deux raisons: premièrement, pour éviter la charge de la figure, deuxièmement parce que dans [90] les auteurs ont montré que le protocole HEEP donne le meilleur résultat par rapport au LEACH et H-PEGASIS en fonction de nombreux paramètres (durée de vie, nœuds vivants, énergie et données).

### ➤ Flux de données transmis

La figure 6.2 représente la quantité de données envoyée en fonction du temps de simulation. On remarque que le nombre de paquets envoyés par notre protocole de routage ADAPTRP (Multi-protocole) est strictement supérieur à la quantité de donnée envoyée par les autres protocoles (HEEP, PEGASIS et APTEEN).

Dans le protocole APTEEN, il existe une très petite quantité de données car APTEEN a conçu des événements critiques temporels visant à capturer des collectes de données périodiques. Dans ce protocole, les capteurs transmettront leurs données détectées en fonction des valeurs de seuil. En revanche, nous trouvons une plus grande quantité de données avec le protocole PEGASIS en raison du mode de communication, mais cette

quantité est plus petite que celles de l'ADAPTRP car notre nouveau protocole combine les avantages de cinq protocoles.

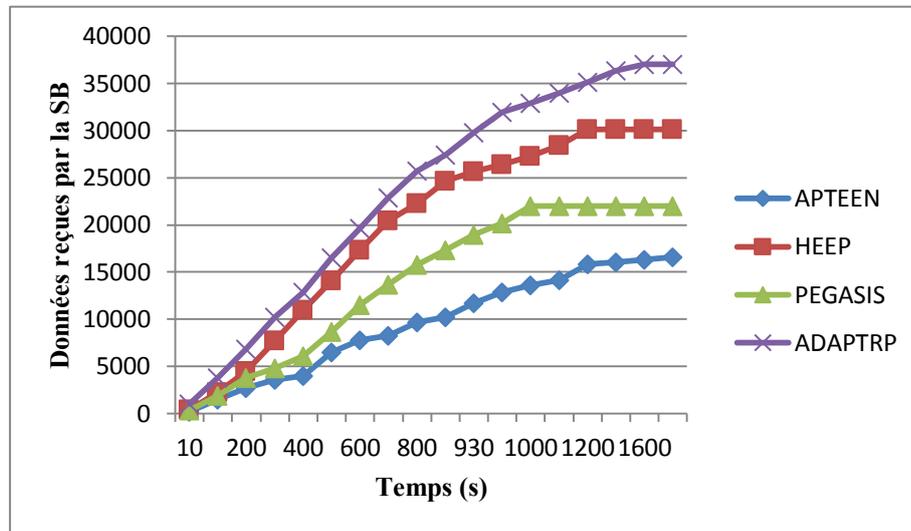


Figure 6.2. Quantité de données reçues en fonction de la durée de la simulation.

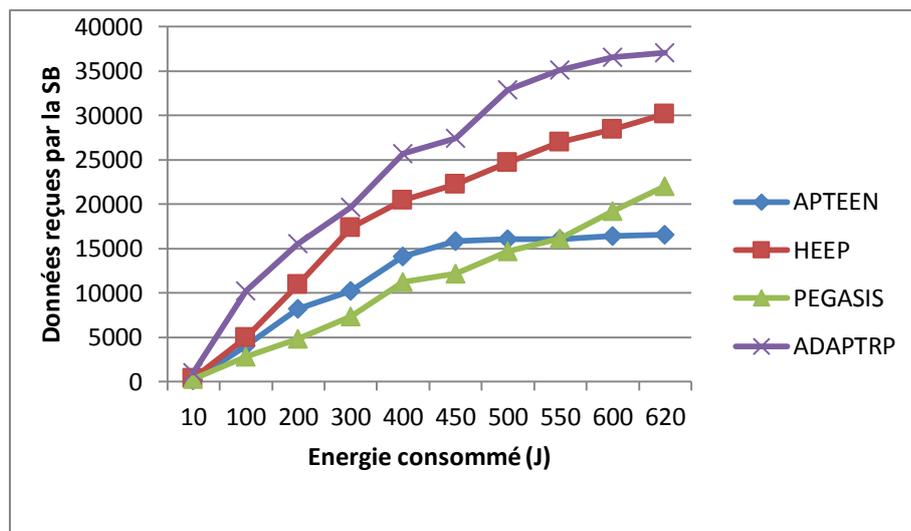


Figure 6.3. Quantité de données reçues par la SB en fonction de l'énergie consommée.

La figure 6.3 représente la quantité de données reçue par la station de base en fonction de l'énergie consommée. A partir de cette figure, on remarque que le nombre de paquets envoyés par notre protocole de routage ADAPTRP (Multi-protocole) vers la station de base est strictement supérieur à la quantité envoyée par les autres protocoles (HEEP, PEGASIS et APTEEN).

Donc, il est évident que considérant la quantité de données reçue par la station de base en fonction de la durée de la simulation ou la quantité de données transférée vers la station de base en fonction de la consommation d'énergie, le protocole ADAPTRP est meilleur que les autres. Parce que dans le protocole ADAPTRP, la station de base est consciente des informations sur l'emplacement et l'énergie de tous les nœuds du réseau, ce qui permet à la station de base d'appliquer les algorithmes basé sur la densité, la consommation d'énergie et les liens de communication (*chapitre4, section 5.1*) pour choisir le meilleur protocole de routage dans chaque cluster, et sélectionner les cluster-head.

### ➤ Durée de vie

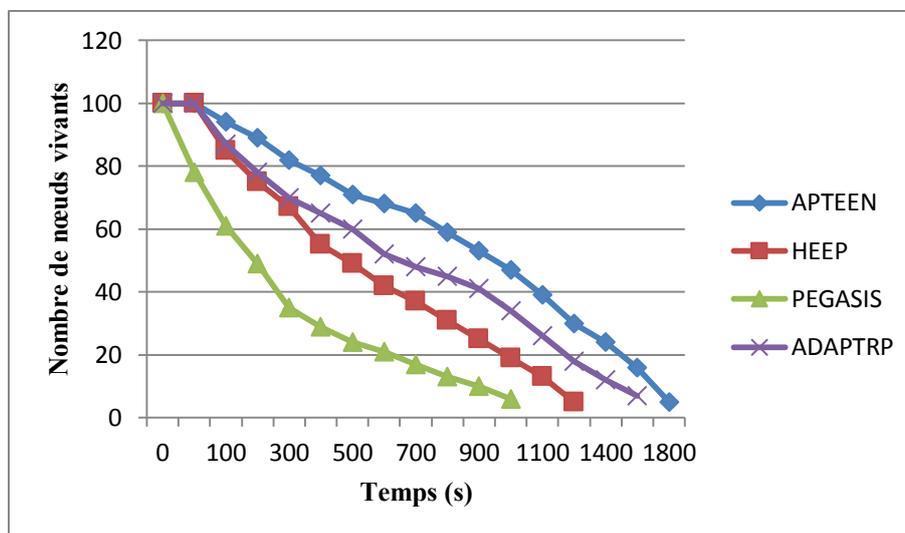


Figure 6.4. Nombre de nœuds vivants en fonction de la durée de la simulation.

Le premier objectif d'auto-adaptation pour les RCSF est la détection du meilleur protocole de routage pour chaque cluster afin d'augmenter la durée de vie; pour approfondir l'évaluation des performances de notre protocole, nous avons mesuré le pourcentage de nœuds morts pendant toute la durée de la simulation, ce qui nous a donné le résultat, comme montré dans le graphique représenté dans la figure 6.4. Initialement, tous les nœuds sont vivants, mais après un certain temps de simulation, nous observons qu'APTEEN augmente le bénéfice de l'énergie et prolonge la durée de vie du réseau car il ne transmet les données que s'il détecte un changement de la valeur captée. À la suite de cette comparaison, nous voyons qu'ADAPTRP, basé sur des protocoles de routage hétérogènes; prolonge la durée de vie du réseau de 10% à 25% par rapport à HEEP, et de 25% à 37% par rapport à PEGASIS.

Par conséquent, le protocole ADAPTRP est meilleur que les autres protocoles. Dans la figure 6.5, nous observons que la durée de vie du réseau pour le protocole ADAPTRP est beaucoup plus longue que celle des autres protocoles, car le protocole ADAPTRP prend en compte les avantages et évite les inconvénients des autres protocoles pour prolonger la durée de vie du réseau.

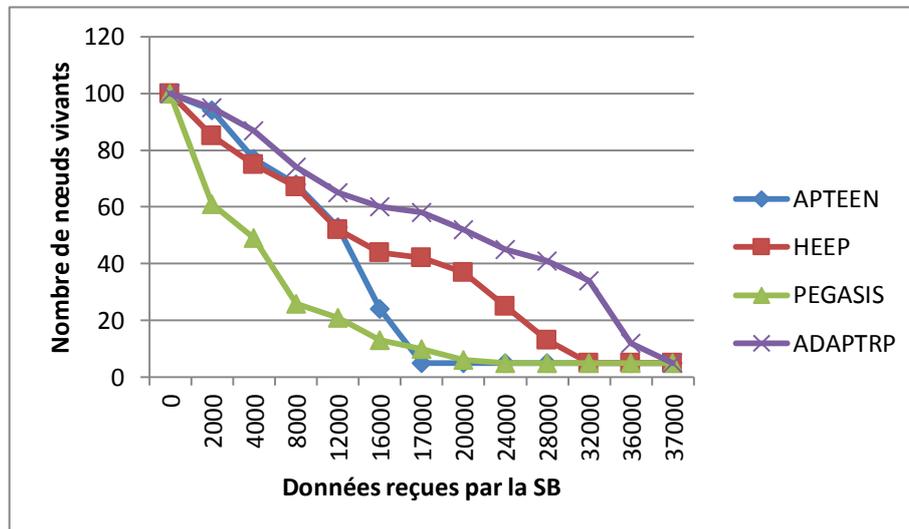


Figure 6.5. Nombre de nœuds vivants en fonction de la quantité de données reçues par la SB.

### ➤ L'énergie consommée

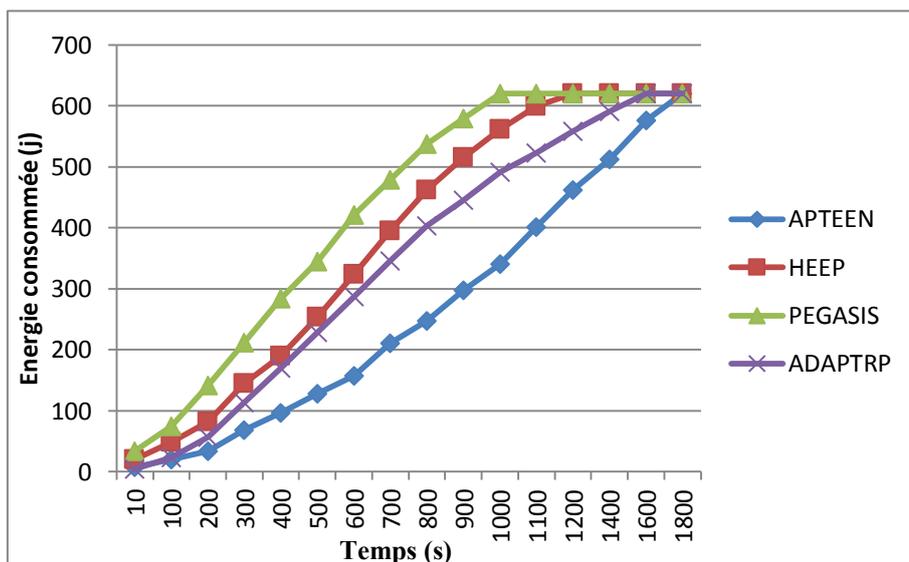


Figure 6.6. Quantité d'énergie consommée en fonction de la durée de la simulation.

On remarque dans *la figure 6.6* que la consommation d'énergie mise en évidence par le protocole APTEEN est plus faible par rapport aux autres protocoles; ceci est dû à un manque de consommation d'énergie dans la transmission et la réception entre les nœuds. D'autre part, nous constatons une amélioration de la consommation d'énergie par notre propre protocole par rapport à HEEP et PEGASIS car ce nouveau protocole ADAPTRP utilise le concept d'optimisation basé sur la consommation d'énergie pour la sélection du meilleur protocole de routage dans chaque cluster.

### 5. Évaluation des performances du protocole HEEP-AO

Le modèle d'expérimentation utilise 100 nœuds ordinaires, le nœud 101 comme station de base et le nœud 102 comme la station de base secondaire. L'ensemble est dispersé sur une surface de 100\*100 m<sup>2</sup>.

Au début de la simulation, l'énergie initiale est égale à deux (02) joules pour tous les nœuds capteurs sauf les deux nœuds spécifiques (i.e: station de base primaire et secondaire) ont une énergie énorme : 5000 Joules. En plus tous les nœuds ont une position fixe durant toute la période de simulation.

La simulation conduite évalue dans un premier temps les performances du mécanisme d'auto-organisation proposé dans les RCSF, puis la simulation des deux protocoles de routages HEEP et HEEP-AO. Ce dernier donne de meilleurs résultats pour le nombre de données reçues par la station de base et le nombre des nœuds défaillants détectés, ...etc.

#### ➤ Flux de données transmis (Panne SB)

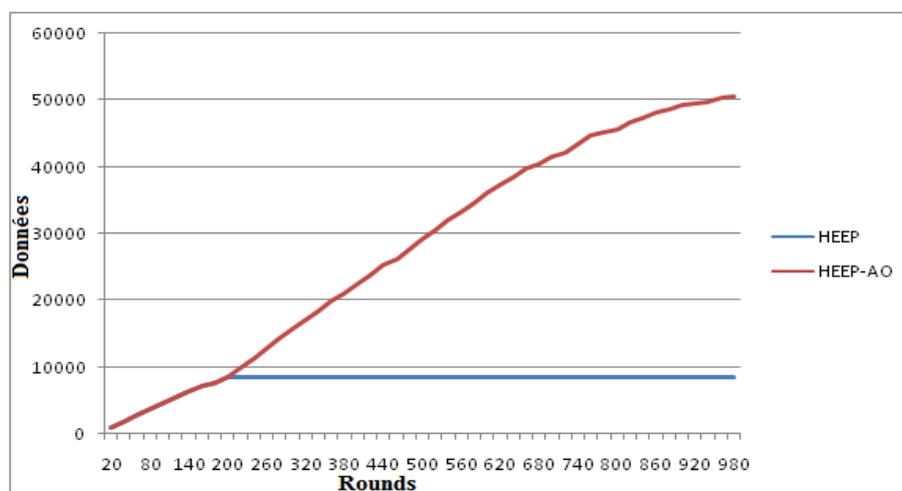


Figure 6.7. Quantité de données reçues (panne de la SB).

La figure 6.7 représente la quantité des données en réception dans le protocole HEEP utilisant une seule station de base et le protocole HEEP-AO avec deux stations de base en fonction du temps de simulation. On remarque que le nombre de paquets envoyés par le protocole HEEP-AO est nettement supérieur à celui envoyé par le protocole HEEP. On observe que le nombre de données reçues est de 8475 depuis l'instant ( $t=200$  s) pour le protocole HEEP suite à une panne de la station de base et que tous les paquets envoyés par les CH depuis cet instant sont perdus par contre dans le protocole HEEP-AO proposé, on constate une augmentation dans le nombre de données reçues; grâce à la deuxième SB (SBS) qui prendra le relais qui minimisera les paquets perdus.

### ➤ Flux de données transmis (Panne SB+CH+nœuds ordinaires)

Dans ce cas, on injecte les trois pannes au niveau de la SB, des CH et nœuds ordinaires en même temps et à l'instant  $t=200$ s.

Dans l'intervalle de temps  $[0...200$ s], on observe que le nombre de données reçues augmente dans HEEP-AO par rapport au protocole HEEP, à cause des vice-CH qui récupèrent les données et assurent le rôle des CH défaillants.

A l'instant ( $t=200$ s), la SB dans HEEP et la SB principale dans HEEP-AO sont toutes les deux en pannes, mais on remarque que le nombre des données reçues dans HEEP est fixe depuis l'instant de panne, par contre dans HEEP-AO ce nombre continu à croître grâce à la présence de la deuxième SB (SBS) qui assure la réception des données (Figure 6.8).

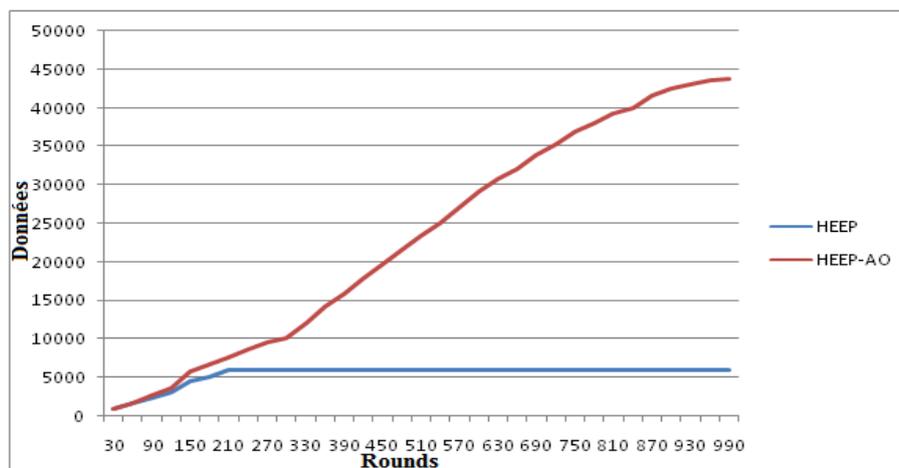


Figure 6.8. Quantité de données reçues (panne de la SB, des CHs et nœuds ordinaires).

### ➤ Nœuds défaillants détectés

Le premier objectif de l'auto-organisation dans les RCSF est le diagnostic qui relève de la détection et la localisation des nœuds défaillants. La figure 6.9 est dédiée à ce phénomène (le nombre des nœuds défaillants détectés en fonction du temps de simulation). Initialement, il n'y a aucun nœud défaillant mais après un certain temps de simulation : temps d'injection des fautes au niveau de quelques nœuds, on remarque que le protocole HEEP ne détecte pas les nœuds défaillants mais seulement les nœuds morts par épuisement d'énergie. Par contre, dans HEEP-AO, on remarque que la détection des nœuds défaillants est plus efficace, car la panne est détectée quelque soit son type (énergie, transmission, captage et traitement).

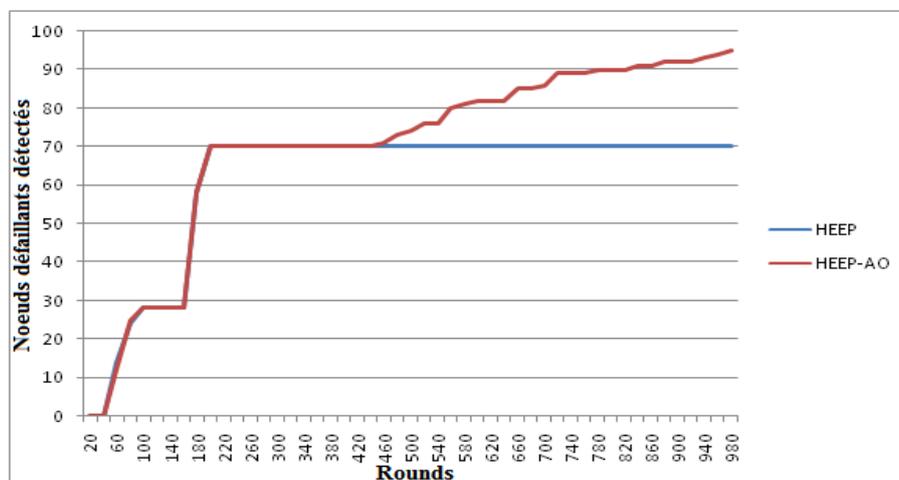


Figure 6.9. Nœuds défaillants détectés dans HEEP et HEEP-AO.

La consommation d'énergie par le protocole de routage HEEP-AO proposé est strictement supérieure à celle de HEEP, parce que l'énergie consommée par un nœud capteur est due essentiellement aux opérations de captage, traitement et communication de données. La quantité de données envoyées vers la station de base est proportionnelle à la consommation de l'énergie, ce qui est également vrai pour les traitements et les messages échangés lors du diagnostic et la tolérance aux différentes pannes tel que :

- Les messages échangés entre les CH et les vice-CH ;
- Les CH et les vice-CH qui restent toujours en WakeUp ;
- L'envoi des données vers la station de base par les vice-CH.

Cela implique que les résultats obtenus reflètent bien la réalité.

### 6. Conclusion

Dans ce chapitre, nous avons présenté quelques environnements de simulation existants sur le marché. Nous avons également présenté NS2 en général avant de passer au choix de la plateforme OS et le choix du langage. Ensuite, nous avons présenté l'implémentation ainsi que l'évaluation des deux protocoles ADAPTRP et HEEP-AO. Le langage Tcl, sous UBUNTU 10.10 et NS2, ont été utilisés lors de cette évaluation.

Après avoir implémenté notre protocole ADAPTRP, nous avons comparé et montrer les performances de notre protocole dans notre première contribution. L'étude comparative entre le protocole proposé et quelques protocoles de référence tels que HEEP, LEACH, PEGASIS, H-PEGASIS, et APTEEN est faite sur différentes métriques à savoir l'énergie consommée, la durée de vie du réseau, le nombre de paquets reçus par la station de base avec différentes tailles. Les résultats obtenus ont montré que le protocole ADAPTRP est plus performant par rapport aux autres.

Enfin, nous avons également implémenté dans notre deuxième contribution l'auto-organisation qui vise HEEP afin de considérer les effets négatifs qui le concernent en absence d'une auto-adaptation efficace. En effet, une simple réinjection de données peut falsifier les résultats dans HEEP. Cependant, ces défaillances sont détectées par le protocole HEEP-AO grâce aux algorithmes d'auto-organisation qu'il offre.

Par ailleurs, nous avons constaté que les tests de performances effectués sur la consommation d'énergie, les nœuds défaillants détectés et les données transmises, ont montré que le protocole HEEP-AO répond bien aux critères de performances souhaités. En effet, ce protocole permet de ne pas surcharger les nœuds capteurs, et de ne pas dégrader les performances du réseau.

# ***Conclusion générale***

La technologie des réseaux de capteurs sans fil a été largement utilisée durant ces deux dernières décennies. La ressource énergétique limitée est l'un des goulots d'étranglement qui limite son utilisation. Les capteurs sans fil sont généralement déployés de manière aléatoire ou déterministe dans des environnements difficiles. Le manque de ressources dans ce domaine représente également la contrainte la plus imposante et engendre plusieurs défis de conception.

Dans ce domaine et comme première contribution, nous avons proposé une nouvelle approche pour permettre la sélection d'une nouvelle adaptation dans les RCSF hétérogènes, en prenant en compte des optimisations multi-objectives pour des stratégies de routage adéquates dans tout le réseau en cluster tout en choisissant le meilleur protocole de routage dans chaque cluster.

Les optimisations sont basées sur la consommation de ressources, la distance entre nœuds, le nombre de nœuds, la QoS offerte par le réseau, et le coût d'auto-adaptation, qui sont des dimensions importantes à considérer dans l'auto-adaptation à atteindre. Jusqu'à présent dans le domaine de recherche des RCSF, de nombreux chercheurs ont associé un protocole de routage unique, mais l'intérêt majeur de notre approche est l'hétérogénéité des protocoles de routage, au lieu d'avoir un seul protocole de routage, il y a une possibilité d'en avoir plusieurs où chaque cluster a son propre protocole, et c'est ce qui a conduit à l'émergence de résultats positifs lors des simulations.

Dans la deuxième contribution, nous avons proposé une méthode pour l'auto-organisation des nœuds défaillants dans les RCSF, spécifiquement appliquée au protocole de routage HEEP. Cette méthode consiste à gérer la défaillance dans le réseau après sa détection et sa localisation. Les résultats des simulations ont montré l'efficacité de notre proposition par rapport au protocole de référence.

Nous proposons en perspective comme suite à nos contributions pour améliorer les performances :

1. de considérer la méthode proposée avec une redondance adaptée aux capteurs de nœuds et éventuellement de mettre en œuvre les deux protocoles ADAPTRP et HEEP-AO sur les capteurs réels comme le Mica Z mote.
2. D'exploiter la mobilité des nœuds capteurs riches en ressources ou celle de la station de base. Ainsi, la station de base (nœud mobile) se déplace vers le cluster-head pour

réceptionner les données; ce processus réduit la distance de transmission et permet par conséquent de conserver l'énergie. Dans ce cas, nous utilisons un modèle de mobilité par clusters pour la gestion des cluster-heads, et un autre modèle de mobilité pour le déplacement de la station de base.

3. Le processus de migration vers les protocoles de routage dans notre première contribution est réalisé par le protocole ADAPTRP proposé, dans lequel la formation des clusters, l'élection des cluster-heads, et la migration sont centralisés. Comme perspective future, nous pouvons proposer le même protocole d'une façon distribuée (ADAPTRP -D) où la formation des clusters, l'élection des cluster-heads, et la migration se font au niveau des nœuds.
4. Auto-organiser le RCSF hétérogène en cas des pannes par l'introduction d'un mécanisme permettant la détection et la localisation de la panne par le cluster-head ou la station de base, ou par les nœuds ordinaires.

# ***Bibliographie***

- [1] SAHRAOUI, Somia et BILAMI, Azeddine. Compressed and distributed host identity protocol for end-to-end security in the IoT. In : *Next Generation Networks and Services (NGNS)*. IEEE, 2014. p. 295-301.
- [2] KOUCHERYAVY, Andrey et SALIM, Ahmed. Cluster-based perimeter-coverage technique for heterogeneous wireless sensor networks. In: *Ultra Modern Telecommunications & Workshops, 2009. ICUMT'09*. p. 1-7.
- [3] ABUSAIMEH, Hesham et YANG, Shuang-Hua. Dynamic cluster head for lifetime efficiency in WSN. *International journal of automation and computing*, 2009, vol. 6, no 1, p. 48-54.
- [4] HEINZELMAN, Wendi Rabiner, CHANDRAKASAN, Anantha, et BALAKRISHNAN, Hari. Energy-efficient communication protocol for wireless microsensor networks. In : *System sciences, 2000. Proceedings of the 33rd annual Hawaii international conference*. p. 10. vol. 2.
- [5] HEINZELMAN, Wendi B., CHANDRAKASAN, Anantha P., et BALAKRISHNAN, Hari. An application-specific protocol architecture for wireless microsensor networks. *IEEE Transactions on wireless communications*, 2002, vol. 1, no 4, p. 660-670.
- [6] TUAH, Norah, ISMAIL, Mahamod, et JUMARI, Kasmiran. Energy-efficient improvement for heterogeneous wireless sensor networks. *Information Technology Journal*, 2012, vol. 11, no 12, p. 1687.
- [7] XIAO, Wendong et SONG, Ruizhuo. Adaptive dynamic programming for sensor scheduling in energy-constrained wireless sensor networks. In: *15th International Conference on Information Fusion (FUSION)*, 2012. p. 991-996. Singapore.
- [8] QIANG, Yan, PEI, Bo, WEI, Wei, et al. An efficient cluster head selection approach for collaborative data processing in wireless sensor networks. *International Journal of Distributed Sensor Networks*, 2015, vol. 11, no 6, p. 794518.
- [9] SMARAGDAKIS, Georgios, MATTA, Ibrahim, et BESTAVROS, Azer. SEP: A stable election protocol for clustered heterogeneous wireless sensor networks. Boston University Computer Science Department, 2004.
- [10] ATWOOD, Bryan, WARNEKE, Brett, et PISTER, Kristofer SJ. Preliminary circuits for smart dust. In: *Mixed-Signal Design. SSMSD. 2000 Southwest Symposium*. p. 87-92.

- [11] WARNEKE, Brett, LAST, Matt, LIEBOWITZ, Brian, et al. Smart dust: Communicating with a cubic-millimeter computer. *Computer*, 2001, vol. 34, no 1, p. 44-51.
- [12] WARNEKE, Brett A. et PISTER, Kristofer SJ. An ultra-low energy microcontroller for smart dust wireless sensor networks. In: *Solid-State Circuits Conference (ISSCC)*. 2004. p. 316-317. San Francisco, CA, USA
- [13] BENAZZOUZ, Mohamed. Surveillance de tout point d'une zone d'intérêt à l'aide d'un réseau de capteur multimédia sans fil. 2013. Mémoire de magistère. Ecole nationale supérieure d'informatique Oued- Smar Alger Algérie.
- [14] BENABBASSI, Youssef. Application de la redondance pour la surveillance par réseau de capteurs sans fil: Cas du réseau de capteurs image sans fil. 2014. Thèse de doctorat. Université de Mascara.
- [15] ANASTASI, G., CONTI, M., FRANCESCO, M. D. et PASSARELLA, A. Energy conservation in wireless sensor networks: A survey. *Elsevier Ad hoc networks*, 2009, vol. 7, no 3, p. 537-568.
- [16] ABDELFAH, Fares. Développement d'une bibliothèque de capteurs. projet de master en informatique, 25/04/2008.
- [17] SOUFIANE, KAISSARI. optimisation et distribution des algorithmes multimédia sur les réseaux de capteurs. Projet de master en génie électronique 2016. Ecole normale supérieure d'enseignement technique – Rabat.
- [18] MELODIA, Tommaso, POMPILI, Dario, GUNGOR, Vehbi C., et al. Communication and coordination in wireless sensor and actor networks. *IEEE transactions on mobile computing*, 2007, vol. 6, no 10.
- [19] KARL, Holger et WILLIG, Andreas. Protocols and architectures for wireless sensor networks. John Wiley & Sons, 2007. ISBN:9780470095126. DOI:10.1002/0470095121.
- [20] CASTELLUCCIA, Claude et FRANCILLON, Aurélien. Protéger les réseaux de capteurs sans fil. *SSTIC08*, 2008.
- [21] MARTINS, David et GUYENNET, Herve. Etat de l'art-Sécurité dans les réseaux de capteurs sans fil. In: *SAR-SSI: 3rd conference on Security of Network Architectures and Information Systems*. 2008. p. 167--181. Loctudy, France.
- [22] FLEURY, Eric, CHELIUS, Guillaume et MIGNON, Thierry. minimisation de l'énergie dans les réseaux de capteurs. 2003. Laboratoire CITI/INSA de Lyon.

- [23] AKYILDIZ, Ian F., SU, Weilian, SANKARASUBRAMANIAM, Yogesh, et al. A survey on sensor networks. *IEEE Communications magazine*, vol. 40, no 8, p. 102-114. Georgia Institute of Technology, Atlanta, USA. Août 2002.
- [24] RANDRIANARISAINA, Aina. Modélisation de la consommation d'énergie En vue de la conception conjointe (matériel/logiciel) des applications embarquées. Application aux réseaux de capteurs sans fil (WSN). 2015. Thèse de doctorat. UNIVERSITE DE NANTES.
- [25] HILL, Jason, SZEWCZYK, Robert, WOO, Alec, et al. System architecture directions for networked sensors. *ACM SIGOPS operating systems review*, 2000, vol. 34, no 5, p. 93-104.
- [26] DUNKELS, Adam, GRONVALL, Bjorn, et VOIGT, Thiemo. Contiki-a lightweight and flexible operating system for tiny networked sensors. In: 29th Annual IEEE International Conference on Local Computer Networks, 2004. p. 455-462. Tampa, FL, USA.
- [27] DUNKELS, Adam, SCHMIDT, Oliver, VOIGT, Thiemo, et al. Protothreads: simplifying event-driven programming of memory-constrained embedded systems. In: Proceedings of the 4th international conference on Embedded networked sensor systems. *Acm*, 2006. p. 29-42. Boulder, CO, USA.
- [28] MO, Wei, QIAO, Daji, et WANG, Zhengdao. Mostly-sleeping wireless sensor networks: Connectivity, k-coverage, and  $\alpha$ -lifetime. In: Proceedings of the 43rd Annual Allerton Conference on Communication, Control, and Computing. 2005. p. 1-10. Monticello, Illinois, USA.
- [29] CARDEI, Mihaela, THAI, My T., LI, Yingshu, et al. Energy-efficient target coverage in wireless sensor networks. In: INFOCOM. 24th annual joint conference of the IEEE computer and communications societies. 2005. p. 1976-1984. Miami.
- [30] WANG, Wei, SRINIVASAN, Vikram, et CHUA, Kee-Chaing. Using mobile relays to prolong the lifetime of wireless sensor networks. In: Proceedings of the 11th annual international conference on Mobile computing and networking. *ACM*, 2005. p. 270-283. Gologne, Germany.
- [31] TILAK, Sameer, ABU-GHAZALEH, Nael B., et HEINZELMAN, Wendi. A taxonomy of wireless micro-sensor network models. *ACM SIGMOBILE Mobile Computing and Communications Review*, 2002, vol. 6, no 2, p. 28-36.
- [32] SINGH, Shio Kumar, SINGH, M. P., et SINGH, D. K. Applications, classifications, and selections of energy-efficient routing protocols for wireless sensor networks. *International Journal of Advanced Engineering Sciences and Technologies (IJAEST)*, 2010, vol. 1, no 2, p. 85-95.

- [33] DE SOUSA, G., CHANET, J. P., JACQUOT, A., et al. Data collection and management solution for wireless sensor networks. In: AgEng 2010: International Conference on Agricultural Engineering. p. 6. Clermont-Ferrand, France.
- [34] ROMDHANI, Bilel. Exploitation de l'hétérogénéité des réseaux de capteurs et d'actionneurs dans la conception des protocoles d'auto-organisation et de routage. 2012. Thèse de doctorat. INSA de Lyon.
- [35] PUGH, Jim et MARTINOLI, Alcherio. Local Range and Bearing Sensing Using Infrared Transceivers in Mobile Robotics. Swarm-Intelligent Systems Research Group. Doi: 10.1.1.147.6376. 30/06/2004.
- [36] LAN-MAN Standards Committee of the IEEE Computer Society – 802.15.1 IEEE Standard for Information technology, Part 15.1: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Wireless Personal Area Networks (WPANS) – IEEE Std 802.15.1. 2002.
- [37] BAKKARI, Mohammed, BENTAMA, Adnan, RACHIDI, Abdelhafid, et KHATORY, Abdellah. Implémentation du Wireless Personal Area Network (WPAN) dans les PME marocaines: cas du protocole ZigBee. Colloque international sur le monitoring des systèmes industriels- CIMSI 2014, marrakech – Maroc.
- [38] ATHMANI, Samir. Protocole de sécurité Pour les Réseaux de capteurs Sans Fil. 2010. mémoire de magister. Université de Batna 2.
- [39] PUJOLLE, G., SALVATORI, O., et NOZICK, J. Les réseaux Editions 2005. éditions Eyrolles, 2005.
- [40] CAMUS, Manuel. Architecture de réception RF très faible coût et très faible puissance. Application aux réseaux de capteurs et au standard ZigBee. 2008. Thèse de doctorat. Université Paul Sabatier-Toulouse III.
- [41] ZHENG, Jianliang et LEE, Myung J. Will IEEE 802.15. 4 make ubiquitous networking a reality?: a discussion on a potential low power, low bit rate standard. IEEE Communications magazine, 2004, vol. 42, no 6, p. 140-146.
- [42] HOLGER, Karl et ANDREAS, Willig. A short survey of wireless sensor networks. 2003. Technical Report. Telecommunication Networks Group. Technische Universität Berlin.
- [43] HEYLIGHEN, Francis, GERSHENSON, Carlos, STAAB, S., et al. Neurons, viscose fluids, freshwater polyp hydra-and self-organizing information systems. IEEE Intelligent Systems, 2003, vol. 18, no 4, p. 72-86.

- [44] PREHOFER, Christian et BETTSTETTER, Christian. Self-organization in communication networks: principles and design paradigms. *IEEE Communications magazine*, 2005, vol. 43, no 7, p. 78-85.
- [45] ASHBY, W. Ross. Principles of the self-organizing dynamic system. *The Journal of general psychology*, 1947, vol. 37, no 2, p. 125-128.
- [46] HEURTEFEUX, Karel et VALOIS, Fabrice. Protocoles d'auto-organisation: une étude qualitative au cours de la vie des réseaux de capteurs. In : *Journées Doctorales en Informatique et Réseaux*. 2008. Villeneuve d'Ascq, France.
- [47] INGELREST, François, SIMPLOT-RYL, David, et STOJMENOVIC, Ivan. A Dominating Sets and Target Radius Based Localized Activity Scheduling and Minimum Energy Broadcast Protocol for Ad Hoc and Sensor Networks. In: *The Third Annual Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net 2004)*.
- [48] KIROUSIS, Lefteris M., KRANAKIS, Evangelos, KRIZANC, Danny, et al. Power consumption in packet radio networks. In: *Annual Symposium on Theoretical Aspects of Computer Science*. Springer, Berlin, Heidelberg, 1997. p. 363-374.
- [49] PARK, Joongseok et SAHNI, Sartaj. Power assignment for symmetric communication in wireless sensor networks. *International Journal of Distributed Sensor Networks*, 2009, vol. 5, no 2, p. 185-200.
- [50] ZHANG, Hui, NIXON, Paddy, et DOBSON, Simon. Multi criteria adaptation mechanisms in homological sensor networks. In: *11th IEEE International Conference on Communication Systems (ICCS 2008)*. pp. 937-942. Singapore.
- [51] AVVENUTI, Marco, CORSINI, Paolo, MASCI, Paolo, et al. An application adaptation layer for wireless sensor networks. *Pervasive and Mobile Computing*, 2007, vol. 3, no 4, p. 413-438.
- [52] DELICATO, Flávia C., PIRES, Paulo F., RUST, Luiz, et al. Reflective middleware for wireless sensor networks. In: *Proceedings of the 2005 ACM symposium on Applied computing*. p. 1155-1159.
- [53] BOONMA, Pruet et SUZUKI, Junichi. Evolutionary constraint-based multiobjective adaptation for self-organizing wireless sensor networks. In: *Bio-Inspired Models of Network, Information and Computing Systems (Bionetics 2007)*. p. 111-119.
- [54] BOONMA, Pruet et SUZUKI, Junichi. MONSOON: A coevolutionary multiobjective adaptation framework for dynamic wireless sensor networks. In: *Proceedings of the 41st Annual Hawaii International Conference on System Sciences. HICSS 2008*. p. 497-497.

- [55] GAMEZ, Nadia, ROMERO, Daniel, FUENTES, Lidia, et al. Constraint-based self-adaptation of wireless sensor networks. In: Proceedings of the 2nd International Workshop on Adaptive Services for the Future Internet and 6th International Workshop on Web APIs and Service Mashups. ACM, 2012. p. 20-27.
- [56] YUAN, Jiann-Shiun et BI, Yu. Process and temperature robust voltage multiplier design for RF energy harvesting. *Microelectronics Reliability*, 2015, vol. 55, no 1, p. 107-113.
- [57] BI, Yu, HU, X. Sharon, JIN, Yier, et al. Enhancing hardware security with emerging transistor technologies. In : Great Lakes Symposium on VLSI, 2016 International. p. 305-310.
- [58] BOURMADA, Amal et BILAMI, Azeddine. Cross-layer energy efficient protocol for QoS provisioning in wireless sensor network. *International Journal of Systems, Control and Communications*, 2017, vol. 8, no 3, p. 230-249.
- [59] CHAKRABORTY, Shounak et KHAN, Ajoy Kumar. Evaluation of wireless sensor network routing protocols with respect to power efficiency. In: 5th International Conference on Computational Intelligence and Communication Networks (CICN), 2013. p. 123-128. Mathura, UP, India.
- [60] OVSTHUS, Knut, KRISTENSEN, Lars M., et al. An industrial perspective on wireless sensor networks—a survey of requirements, protocols, and challenges. *IEEE Communications Surveys & Tutorials*, 2014, vol. 16, no 3, p. 1391-1412.
- [61] MANJESHWAR, Arati et AGRAWAL, Dharma P. APTEEN: A hybrid protocol for efficient routing and comprehensive information retrieval in wireless sensor networks. In: *ipdps*. IEEE, 2002. p. 0195b.
- [62] ASERI, T. C., et al. Comparison of routing protocols in wireless sensor network using mobile sink-A survey. In: *Engineering and Computational Sciences (RAECS), 2014 Recent Advances*. p. 1-4.
- [63] PARVIN, Shamsad et RAHIM, Muhammad Sajjadur. Routing protocols for wireless sensor networks: a comparative study. In: *International Conference on Electronics, Computer and Communication (ICECC'2008)*. p. 891-894. University of Rajshahi, Bangladesh.
- [64] BHARTI, Anjali et SHARMA, Kanika. Comparative Study of Clustering based Routing Protocols for Wireless Sensor Network. *International Journal of Computer Applications*, 2013, vol. 66, no 21.

- [65] BHATTACHARYYA, Debnath, KIM, Tai-hoon, et PAL, Subhajit. A comparative study of wireless sensor networks and their routing protocols. *Sensors*, 2010, vol. 10, no 12, p. 10506-10523.
- [66] AKKAYA, Kemal et YOUNIS, Mohamed. A survey on routing protocols for wireless sensor networks. *Ad hoc networks*, 2005, vol. 3, no 3, p. 325-349.
- [67] AL-KARAKI, Jamal N. et KAMAL, Ahmed E. Routing techniques in wireless sensor networks: a survey. *IEEE wireless communications*, 2004, vol. 11, no 6, p. 6-28.
- [68] YE, Fan, CHEN, Alvin, LU, Songwu, et al. A scalable solution to minimum cost forwarding in large sensor networks. In: *Tenth International Conference on Computer Communications and Networks*, 2001. p. 304-309. Scottsdale, AZ, USA.
- [69] BIRADAR, Rajashree V., PATIL, V. C., SAWANT, S. R., et al. Classification and comparison of routing protocols in wireless sensor networks. *Special Issue on Ubiquitous Computing Security Systems*, 2009, vol. 4, no 2, p. 704-711.
- [70] MÜHLETHALER, Paul. Routage dans les réseaux ad hoc. *Techniques de l'ingénieur*, 2004.
- [71] CHAKKOR, Saad. E-diagnostic de processus physiques à base des méthodes de haute résolution Application: machines éoliennes. 2015. Thèse de doctorat. Faculté des Sciences de Tétouan, Maroc.
- [72] HEINZELMAN, Wendi Rabiner, KULIK, Joanna, et BALAKRISHNAN, Hari. Adaptive protocols for information dissemination in wireless sensor networks. In: *Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking*. 1999. p. 174-185. Seattle, WA, USA.
- [73] BELLAL, Juba, MOUSSAOUI, Adel, et al. Routage Hiérarchique avec Optimisation de Consommation d'Energie dans les Réseaux de Capteurs Sans-Fil. 2016. Thèse de doctorat. Université A/Mira de Bejaia.
- [74] BOUCHAKOUR, Khaled. Routage hiérarchique sur les réseaux de capteurs sans fil. 2012. Thèse de doctorat. Ecole Nationale Supérieure d'Informatique.
- [75] RACHID, BECHAR. Surveillance distribuée d'un réseau de capteurs sans fil à base d'agents. 2015. Thèse de doctorat. Université Ahmed Ben Bella d'Oran1 Es Senia.
- [76] BELFKIH, Abderrahmen, SADEG, Bruno, DUVALLET, Claude, et al. Les bases de données dans les réseaux de capteurs sans fil. *Technique et Science Informatiques*, 2014, vol. 33, no 9-10, p. 739-776.
- [77] RAJAGOPALAN, Ramesh et VARSHNEY, Pramod K. Data aggregation techniques in sensor networks: A survey. *IEEE Commun. Surveys Tutorials*, vol. 8, no. 4, pp. 48–63, Oct. 2006.

- [78] SIVARANJANI, S., RADHAKRISHNAN, S., et THANGARAJ, C. Adaptive delay and energy aware data aggregation technique in wireless sensor networks. *Mobile communication and power engineering*, 2013, vol. 296, no 5, p. 41-49.
- [79] NOLOT, Florent, NIANG, Ibrahima, FLAUZAC, Olivier, et al. Routage et agrégation de données dans les réseaux de capteurs sans fil structurés en clusters auto-stabilisants. *REVUE AFRICAINE DE LA RECHERCHE EN INFORMATIQUE ET MATHÉMATIQUES APPLIQUÉES*, 2016, vol. 21.
- [80] MANJESHWAR, Arati et AGRAWAL, Dharma P. TEEN: a routing protocol for enhanced efficiency in wireless sensor networks. In: null. *IEEE*, 2001. p. 30189a.
- [81] LINDSEY, Stephanie et RAGHAVENDRA, Cauligi S. PEGASIS: Power-efficient gathering in sensor information systems. In : *Aerospace conference proceedings*, 2002. *IEEE*. p. 3. Big Sky, MT, USA.
- [82] XIANGNING, Fan et YULIN, Song. Improvement on LEACH protocol of wireless sensor network. In: *International Conference on Sensor Technologies and Applications (SensorComm 2007)*. p. 260-264. Valencia, Spain.
- [83] RANA, Hetal, VHATKAR, Sangeeta, et ATIQUE, Mohommad. Comparative Study of PEGASIS Protocols in Wireless Sensor Network. *IOSR Journal of Computer Engineering (IOSR-JCE) e-ISSN*, 2014, p. 2278-0661.
- [84] CHANDRASEKARAN, V. A review on hierarchical cluster based routing in wireless sensor networks. *Journal of global research in computer science*, 2012, vol. 3, no 2, p. 12-16.
- [85] YOUNIS, Ossama et FAHMY, Sonia. Distributed clustering in ad-hoc sensor networks: A hybrid, energy-efficient approach. In: *INFOCOM 2004. Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies*. Hong Kong, China.
- [86] WAWARE, Sanjay, SARWADE, Dr Nisha, et GANGURDE, Pallavi. A review of power efficient hierarchical routing protocols in wireless sensor networks. *International Journal of Engineering Research and Applications (IJERA)*, 2012, vol. 2, no 2, p. 1096-1102.
- [87] SINGH, Shio Kumar, SINGH, M. P., et SINGH, D. K. A survey of energy-efficient hierarchical cluster-based routing in wireless sensor networks. *International Journal of Advanced Networking and Application (IJANA)*, 2010, vol. 2, no 02, p. 570-580.
- [88] PAWAR, Khushboo et KELKAR, Y. A survey of hierarchical routing protocols in wireless sensor network. *International Journal of Engineering and Innovative Technology*, 2012, vol. 1, no 05, p. 50–54.

- [89] BOURMADA, Amal et BILAMI, Azeddine. Hybrid Energy Efficient Protocol with Service Differentiation for QoS Provisions in Multi-Hop WSNs. *International Journal of Embedded and Real-Time Communication Systems (IJERTCS)*, 2015, vol. 6, no 2, p. 27-47.
- [90] BOUBICHE, Djallel Eddine et BILAMI, Azeddine. HEEP (Hybrid Energy Efficiency Protocol) based on chain clustering. *International Journal of Sensor Networks*, 2011, vol. 10, no 1-2, p. 25-35.
- [91] UMAR, Idris Abubakar, HANAPI, Zurina M., et ADNAN, Ali Idarous. Performance Analysis of State-Free Geographic Forwarding Protocols for Wireless Sensor Networks. *International Journal of Engineering and Technology (IJET)*. ISSN: 0975-4024. Vol 8 No 6 Dec 2016-Jan 2017.
- [92] BOUBICHE, Djallel Eddine. Une approche Inter-Couches (cross-layer) pour la Sécurité dans les RCSF. 2013. Thèse de doctorat. Université de Batna 2.
- [93] HEDJAZI, Djalal. Conception d'un modèle coopératif de support de la télémaintenance industrielle. 2011. Thèse de doctorat. Université de Batna 2.
- [94] Ali, JORIO. Le Clustering basé sur la Classification Spectrale pour l'Optimisation d'Energie dans les Réseaux de Capteurs Sans Fil Homogènes. 2015. These de doctorat. UNIVERSITÉ MOHAMMED V, Rabat – Maroc.
- [95] BUSNEL, Yann. Systèmes d'information collaboratifs et auto-organisants pour réseaux de capteurs large-échelle: " De la théorie à la pratique". 2008. Thèse de doctorat. Université Rennes 1.
- [96] LIU, Yong-Min et JIANG, Xin-Hua. A protocol model for wireless sensor network. In : *International Conference on Networks Security, Wireless Communications and Trusted Computing*. NSWCTC'09, 2009. p. 588-591. Ukraine.
- [97] MUTHALAGI, R. et RANI, P. Leela. A Novel Opinion Dynamics Model Based Clustering Protocol in High-level Heterogeneous WSN. *International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT)*. Volume 2. Issue 3. ISSN: 2456-3307. 2017.
- [98] SAMUNDISWARY, P., PRIYADARSHINI, Padma, et DANANJAYAN, P. Performance evaluation of heterogeneous sensor networks. In : *Future Computer and Communication (ICFCC 2009)*. International Conference. p. 264-267. Kuala Lumpur, Malaysia.
- [99] HANDY, M. J., HAASE, Marc, et TIMMERMANN, Dirk. Low energy adaptive clustering hierarchy with deterministic cluster-head selection. In: *Mobile and Wireless Communications Network*, 2002. 4th International Workshop. p. 368-372.

- [100] GALOS, Mihai. Reconfiguration dynamique et simulation fine modélisée au niveau de transaction dans les réseaux de capteurs sans fil hétérogènes matériellement-logiciellement. 2012. Thèse de doctorat. Ecole Centrale de Lyon.
- [101] YARVIS, Mark, KUSHALNAGAR, Nandakishore, SINGH, Harkirat, et al. Exploiting heterogeneity in sensor networks. In: INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. p. 878-890. Miami.
- [102] ROMDHANI, Bilel. Exploitation de l'hétérogénéité des réseaux de capteurs et d'actionneurs dans la conception des protocoles d'auto-organisation et de routage. 2012. Thèse de doctorat. L'Institut National des Sciences Appliquées de Lyon.
- [103] BOULANOUAR, Ibtissem. Algorithmes de suivi de cible mobile pour les réseaux de capteurs sans fils. 2014. Thèse de doctorat. Paris Est.
- [104] GAMEZ, Nadia, FUENTES, Lidia, et ARAGÜEZ, Miguel. Autonomic computing driven by feature models and architecture in FamiWare. *Software Architecture*, 2011, p. 164-179.
- [105] KRISHNAMACHARI, Bhaskar. *Networking wireless sensors*. Cambridge Press: United Kingdom, 2005.
- [106] XU, Ning. A survey of sensor network applications. *IEEE communications magazine*, 2002, vol. 40, no 8, p. 102-114.
- [107] PERKINS, Charles E. et BHAGWAT, Pravin. Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers. In: *ACM SIGCOMM computer communication review*. ACM, 1994. p. 234-244.
- [108] CLAUSEN, Thomas et JACQUET, Philippe. Optimized link state routing protocol (OLSR). DOI: 10.17487/RFC3626. 2003.
- [109] PERKINS, Charles, BELDING-ROYER, Elizabeth, et DAS, Samir. Ad hoc on-demand distance vector (AODV) routing. DOI: 10.17487/RFC3561. 2003.
- [110] JOHNSON, David, HU, Yin-chun, et MALTZ, David. The dynamic source routing protocol (DSR) for mobile ad hoc networks for IPv4. 2007.
- [111] ABOLHASAN, Mehran, WYSOCKI, Tadeusz, et DUTKIEWICZ, Eryk. A review of routing protocols for mobile ad hoc networks. *Ad hoc networks*, 2004, vol. 2, no 1, p. 1-22.
- [112] RAMASUBRAMANIAN, Venugopalan, HAAS, Zygmunt J., et SIRER, Emin Gün. SHARP: A hybrid adaptive routing protocol for mobile ad hoc networks. In: *Proceedings of the 4th ACM international symposium on Mobile ad hoc networking & computing*. 2003. p. 303-314.

- [113] HAAS, Zygmunt J. A new routing protocol for the reconfigurable wireless networks. In : 6th International Conference Universal Personal Communications Record, 1997. p. 562-566. San Diego, CA, USA.
- [114] YANG, Chun-Chuan et TSENG, Li-Pin. Fisheye zone routing protocol: A multi-level zone routing protocol for mobile ad hoc networks. *Computer Communications*, 2007, vol. 30, no 2, p. 261-268.
- [115] DARTIES, Benoit, THEOLEYRE, Fabrice, et DUDA, Andrzej. A divide-and-conquer scheme for assigning roles in multi-channel wireless mesh networks. In: *Local Computer Networks*, 2009. IEEE 34th Conference. p. 277-280. Zurich, Switzerland.
- [116] KYASANUR, Pradeep et VAIDYA, Nitin H. Capacity of multichannel wireless networks under the protocol model. *IEEE/ACM Transactions on Networking*, 2009, vol. 17, no 2, p. 515-527.
- [117] CHAOUICHE, Hanane. Réalisation d'un prototype d'un réseau de capteurs sans fil dédié à l'agriculture de précision. 2015. Thèse de doctorat.
- [118] MALEH, Yassine et EZZATI, Abdellah. Etude et développement d'un protocole symétrique pour sécuriser les communications des RCSF. 2015. Thèse de doctorat. Faculté des Sciences et Technique de Settat.
- [119] KETFI, Abdelmadjid, BELKHATIR , Noureddine et CUNIN, Pierre-Yves. Adapting applications on the fly. In *ASE '02 : Proceedings of the 17th IEEE International Conference on Automated Software Engineering*, pp. 313. IEEE Computer Society. Washington, DC, USA.
- [120] GRONDIN, Guillaume, BOURAQADI, Noury, et VERCOUTER, Laurent. Assemblage automatique et adaptation d'applications à base de composants. In : *CAL*. 2008. p. 121.
- [121] DEHNI, Lahcene, BEENNABI, Y., et KRIEF, Francine. LEA2C: Une nouvelle approche de routage dans les réseaux de capteurs pour l'optimisation de la consommation d'énergie. *Université Paris*, 2003, vol. 13.
- [122] BRAMAN, Alakesh et UMAPATHI, G. R. A comparative study on advances in LEACH routing protocol for wireless sensor networks: A survey. *International journal of advanced research in computer and communication engineering*, 2014, vol. 3, no 2, p. 5683-5690.
- [123] AKILAL, Abdallah, GHEBBI, Sofiane, et al. Conception d'un protocole de routage hiérarchique pour les réseaux de capteurs sans fil. 2016. Thèse de doctorat. Université A/Mira de Bejaia.

- [124] ZELKOWITZ, M. *Advances in Computers: Architectural Advances*, volume 69. Elsevier Science, Academic Press, New York, 8, 2011.
- [125] SINGH, Shio Kumar, SINGH, M. P., et SINGH, D. K. A survey of energy-efficient hierarchical cluster-based routing in wireless sensor networks. *International Journal of Advanced Networking and Application (IJANA)*, 2010, vol. 2, no 02, p. 570-580.
- [126] KACIMI, RAHIM. *Techniques de conservation d'énergie pour les réseaux de capteurs sans fils*. 2009. Thèse de Doctorat. Institut national polytechnique de Toulouse, France.
- [127] NAIDJA, Miloud et BILAMI, Azeddine. A dynamic self-organising heterogeneous routing protocol for clustered WSNs. *International Journal of Wireless and Mobile Computing*, 2017, vol. 12, no 2, p. 131-141.
- [128] J. S. Rauthan, S. Mishra. An improved Cluster Based Multi-hop Routing in Self-Organizing Wireless Sensor Networks. *International Journal of Engineering Research Technology (IJERT)* Vol.1, Issue 4, June – 2012.