

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

Université A/Mira de Béjaïa

Faculté des Sciences Exactes

Département d'Informatique



*Mémoire de fin de cycle*  
*en vue d'obtention du diplôme de master recherche en informatique*  
*spécialité : Réseaux et systèmes distribués*

Thème

---

Protocole hiérarchique basé sur l'évaluation de la  
redondance pour la garantie de la couverture dans les  
réseaux de capteurs sans fil.

---

Mémoire soutenu le 01/07/2013 par :

M<sup>elle</sup> **BERBAR** Nassima

M<sup>elle</sup> **BENKERROU** Hayet

Devant le jury composé de :

|            |  |                     |
|------------|--|---------------------|
| Président  | M <sup>r</sup> <b>ABBACHE</b> Bournane   | M.A.B, U.A.M Béjaïa |
| Rapporteur | M <sup>r</sup> <b>KHANOUCHE</b> M-Essaid | M.A.A, U.A.M Béjaïa |
| Examineur  | M <sup>r</sup> <b>AISSANI</b> SOFIANE    | M.A.B, U.A.M Béjaïa |
| Examineur  | M <sup>r</sup> <b>MIR</b> Foudil         | M.A.A, U.A.M Béjaïa |

Promotion 2012/2013

---

# *Remerciements*

---

*Louange à Dieu, Le miséricordieux, sans lui rien de tout cela n'aurait pu être.*

Nous désirons exprimer nos sincères gratitudee à notre encadreur Mr KHANOUCHE Mohamed Essaid pour nous avoir proposé ce sujet. Nous le remercions également pour la haute qualité de son encadrement, son suivi, sa disponibilité, ses conseils et ses critiques constructives.

Nous tenons à remercier aussi les membres de jury pour avoir accepté d'évaluer notre travail.

Notre reconnaissance va particulièrement à l'ensemble des enseignants du département d'informatique pour tout ce qui nous a été transmis tout au long de notre formation.

Nos remerciements les plus chaleureux à nos parents, pour leur soutien, leurs encouragements et leurs sacrifices.

Nous remercions également tous nos amis, nos collègues et tous ceux qui ont contribué de pré ou de loin pour la réalisation de ce travail.

---

# *Dédicaces*

---

*Je dédie ce modeste travail à :*

**A** mes très chers parents,  
**A** mes frères et soeurs ,  
**A** toute ma famille,  
**A** mes amis Abdou, Mouh, Bill, Cicine, Meheni, Cherif, Hakim et Samou,  
**A** l'association DEFI contre les myopathies,  
**A** mes collègues et camarades, et tous ceux qui m'ont aidé,  
**A** ma binôme Nassima et sa famille ;

***BENKERROU Hayet***

---

# *Dédicaces*

---

*Je dédie ce modeste travail à :*

**A** mes très chers parents,  
**A** mes frères et soeurs ,  
**A** toute ma famille,  
**A** tous mes amis et collègues, et tous ceux qui m'ont aidé,  
**A** ma binôme Hayet et sa famille ;

*BERBAR Nassima*

## LISTE DES ABRÉVIATIONS

RCSFs : Réseaux de Capteurs Sans Fil.

WSN : Wireless Sensor Networks.

TCP/IP : Transmission Control Protocol/Internet Protocol.

ISM band : Industrie, Science and Medical band.

NTP : Network Time Protocol.

DoS : Deny of Service.

TinyOS : Tiny Operating Système.

DHCP : Dynamic Host Configuration Protocol.

DNS : Domain Name System.

HTTP : HyperText Transfer Protocol.

PPP : Point-to-Point ProtocolPoint-to-Point Protocol.

QoS : Qualité de Service.

ECCRA : Energy-efficient Coverage and Connectivity preserving Routing Algorithm under

---

border effects.

LDCC : Layred-Diffusion based Coverage Control Protocol.

LCFC : Low Coverage First Classifying algorithm

CACS : Covrage-Aware Clustering Scheme.

CEC : Coverage-preserving based-Energy Clustering algorithm.

DEECIC : Distributed Energy-Efficient Clustering with Improved Coverage.

CACP : Coverage-Aware Clustering Protocol.

HRP : Hierarchical Routing Protocol.

ECSSCoM : Energy-aware Clustering Sensor Scheduling Coverage Maintenance.

CGNSC : Coverage-Guaranteed Nodes Scheduling Algorithm in Clusters for Wireless Sensor Networks.

HPRC : Hierarchical Protocol based-on Redundancy evaluation for Coverage guarantee.

# TABLE DES MATIÈRES

|  |             |
|--|-------------|
| <b>Table des Matières</b>                                      | <b>iii</b>  |
| <b>Liste des tableaux</b>                                      | <b>viii</b> |
| <b>Table des figures</b>                                       | <b>ix</b>   |
| <b>Intrduction Générale</b>                                    | <b>1</b>    |
| <b>1 Présentation des réseaux de capteurs sans fil</b>         | <b>4</b>    |
| 1.1 Introduction . . . . .                                     | 4           |
| 1.2 Définitions et architectures . . . . .                     | 4           |
| 1.2.1 Qu'est ce qu'un capteur? . . . . .                       | 4           |
| 1.2.2 Architecture d'un capteur . . . . .                      | 5           |
| 1.2.2.1 Architecture matérielle . . . . .                      | 5           |
| 1.2.2.2 Architecture logicielle . . . . .                      | 6           |
| 1.2.3 Réseau de capteur . . . . .                              | 7           |
| 1.3 Contraintes liées aux réseaux de capteurs . . . . .        | 8           |
| 1.3.1 Durée de vie limitée . . . . .                           | 8           |
| 1.3.2 Bande passante limitée . . . . .                         | 8           |
| 1.3.3 Scalabilité . . . . .                                    | 8           |
| 1.3.4 Topologie . . . . .                                      | 9           |
| 1.3.5 Agrégation des données . . . . .                         | 9           |
| 1.4 Communication dans les réseaux de capteurs . . . . .       | 9           |
| 1.4.1 Pile protocolaire dans les réseaux de capteurs . . . . . | 10          |
| 1.5 Domaines d'application . . . . .                           | 12          |
| 1.5.1 Les aplications militaires . . . . .                     | 12          |

|       |   |    |
|-------|---|----|
| 1.5.2 | Les applications de la surveillance . . . . .     | 13 |
| 1.5.3 | Les applications environnementales . . . . .      | 13 |
| 1.5.4 | Les applications médicales . . . . .              | 13 |
| 1.5.5 | Les applications domestiques . . . . .            | 13 |
| 1.5.6 | Les applications commerciales . . . . .           | 14 |
| 1.6   | Quelques problématiques liées aux RCSFs . . . . . | 14 |
| 1.7   | Conclusion . . . . .                              | 16 |

**2 Taxonomie des protocoles de couverture dans les réseaux de capteurs sans**

|            |   |           |
|------------|---|-----------|
| <b>fil</b> |   | <b>17</b> |
| 2.1        | Introduction . . . . .  | 17        |
| 2.2        | Préliminaires . . . . .   | 17        |
| 2.2.1      | Zone d'intérêt . . . . .  | 17        |
| 2.2.2      | Zone de couverture d'un capteur . . . . .   | 18        |
| 2.2.3      | Zone de couverture d'un réseau de capteurs . . . . .  | 18        |
| 2.2.4      | Modélisation du canal radio . . . . .   | 18        |
| 2.2.5      | Types de couverture et redondance des noeuds . . . . .  | 19        |
| 2.2.6      | Ordonnancement et clustering . . . . .  | 20        |
| 2.3        | Modes d'exécution des protocoles . . . . .  | 21        |
| 2.3.1      | Les approches centralisées . . . . .  | 21        |
| 2.3.2      | Les approches distribuées . . . . .   | 21        |
| 2.4        | Types d'informations de contrôle utilisées dans les protocoles . . . . .  | 21        |
| 2.4.1      | Les protocoles basés sur les informations de localisation . . . . .   | 21        |
| 2.4.2      | Les protocoles basés sur la distance . . . . .  | 22        |
| 2.4.3      | Les protocoles basés sur le nombre de sauts . . . . .   | 22        |
| 2.5        | Etude critique de quelques protocoles de couverture . . . . .   | 22        |
| 2.5.1      | Protocoles distribués basés sur le nombre de sauts . . . . .  | 22        |
| 2.5.1.1    | Energy-efficient Coverage and Connectivity preserving Routing<br>Algorithm under border effects : ECCRA . . . . . | 22        |
| 2.5.1.2    | Layred-Diffusion based Coverage Control Protocol : LDCC . . . . .   | 24        |
| 2.5.2      | Protocoles basés sur la connaissance de positions . . . . .   | 26        |
| 2.5.2.1    | Low Coverage First Classifying algorithm : LCFC . . . . .   | 26        |
| 2.5.3      | Les protocoles basés sur le clustering . . . . .  | 28        |
| 2.5.3.1    | Covrage-Aware Clustering Scheme : CACS . . . . .  | 28        |
| 2.5.3.2    | Coverage-preserving based-Energy Clustering algorithm : CEC . . . . .   | 30        |
| 2.5.3.3    | Distributed Energy-Efficient Clustering with Improved Cove-<br>rage : DEECIC . . . . .                            | 31        |



|          |  |           |
|----------|--|-----------|
| 2.5.3.4  | Coverage-Aware Clustering Protocol . . . . .   | 34        |
| 2.5.3.5  | Novel hierarchical routing protocolin : HRP . . . . .  | 38        |
| 2.5.3.6  | Energy-aware Clustering Sensor Scheduling Coverage Maintenance : ECSSCoM . . . . .                                   | 40        |
| 2.5.3.7  | Coverage-Guaranteed Nodes Scheduling Algorithm in Clusters : CGNSC . . . . .   | 41        |
| 2.6      | Comparaison des protocoles de couverture étudiés . . . . .   | 43        |
| 2.7      | Conclusion . . . . .   | 44        |
| <b>3</b> | <b>Protocole hiérarchique basé sur l'évaluation de la redondance pour la garantie de la couverture dans les RCSF</b> | <b>46</b> |
| 3.1      | Introduction . . . . .   | 46        |
| 3.2      | Formulation du problème . . . . .  | 46        |
| 3.3      | Propriétés du réseau et hypothèses . . . . .   | 47        |
| 3.4      | Protocole hiérarchique basé sur l'évaluation de la redondance pour la garantie de la couverture : HPRC . . . . .     | 48        |
| 3.4.1    | Principe général du protocole HPRC . . . . .   | 48        |
| 3.4.2    | Description détaillée des phases de HPRC . . . . .   | 49        |
| 3.4.2.1  | Découverte de voisins . . . . .  | 49        |
| 3.4.2.2  | Formation de clusters et ordonnancement des membres . . . . .  | 49        |
| 3.4.2.3  | Collecte d'informations . . . . .  | 51        |
| 3.4.3    | L'organigramme du protocole HPRC . . . . .   | 51        |
| 3.4.4    | Technique d'évaluation de la redondance . . . . .  | 54        |
| 3.4.5    | Modèle énergétique . . . . .   | 54        |
| 3.5      | Scénario illusratif du protocole HPRC . . . . .  | 56        |
| 3.6      | Conclusion . . . . .   | 68        |
| <b>4</b> | <b>Simulation et évaluation des performances</b>   | <b>70</b> |
| 4.1      | Introduction . . . . .   | 70        |
| 4.2      | Environnement de simulation et paramètres . . . . .  | 70        |
| 4.2.1    | Description du système . . . . .   | 70        |
| 4.2.2    | Les variables du système . . . . .   | 71        |
| 4.2.3    | Evénements discrets du système . . . . .   | 71        |
| 4.3      | Les étapes de création du simulateur . . . . .   | 72        |
| 4.3.1    | Initialisation des variables . . . . .   | 72        |
| 4.3.2    | Création de la topologie . . . . .   | 72        |
| 4.3.3    | Calcul des distances, voisins et poids . . . . .   | 73        |

|         |  |           |
|---------|--|-----------|
| 4.3.4   | Application du protocole HPRC . . . . .    | 73        |
| 4.3.5   | Affichage des résultats . . . . .          | 73        |
| 4.3.5.1 | Le taux de couverture . . . . .            | 73        |
| 4.3.5.2 | Le nombre de noeuds vivants . . . . .      | 73        |
| 4.3.5.3 | La taille des clusters . . . . .           | 74        |
| 4.3.5.4 | L'énergie totale consommée . . . . .       | 74        |
| 4.3.5.5 | La durée de vie . . . . .                  | 74        |
| 4.4     | Evaluation des performances . . . . .      | 74        |
| 4.4.1   | Le taux de couverture . . . . .            | 75        |
| 4.4.2   | Le nombre de noeuds vivants . . . . .      | 76        |
| 4.4.3   | La taille des clusters . . . . .           | 77        |
| 4.4.4   | La consommation totale d'énergie . . . . . | 78        |
| 4.4.5   | La durée de vie du réseau . . . . .        | 78        |
| 4.5     | Conclusion . . . . .                       | 79        |
|         | <b>Conclusion Générale et Perspectives</b> | <b>81</b> |
|         | <b>Bibliographie</b>                       | <b>xi</b> |

## LISTE DES TABLEAUX

|      |   |    |
|------|---|----|
| 2.1  | Comparaison des protocoles de couverture étudiés. . . . .                         | 44 |
| 3.1  | Scénario illustratif du protocole HPRC : résultats de la phase de découverte. . . | 57 |
| 3.2  | Scénario illustratif de HPRC : résultats de la première période. . . . .          | 58 |
| 3.3  | Scénario illustratif de HPRC : l'énergie consommée durant la première période. .  | 60 |
| 3.4  | Scénario illustratif de HPRC : résultats de la deuxième période. . . . .          | 61 |
| 3.5  | Scénario illustratif de HPRC : l'énergie consommée durant la deuxième période.    | 62 |
| 3.6  | Scénario illustratif de HPRC : résultats de la troisième période. . . . .         | 63 |
| 3.7  | Scénario illustratif de HPRC : l'énergie consommée durant la troisième période.   | 64 |
| 3.8  | Scénario illustratif de HPRC : résultats de la quatrième période. . . . .         | 65 |
| 3.9  | Scénario illustratif de HPRC : l'énergie consommée durant la quatrième période.   | 66 |
| 3.10 | Scénario illustratif de HPRC : résultats de la cinquième période. . . . .         | 67 |
| 4.1  | Les variables descriptives du système. . . . .                                    | 71 |
| 4.2  | Les événements discrets du système. . . . .                                       | 72 |
| 4.3  | Les paramètres de simulation : comparaison de HPRC avec DEECIC [34]. . . . .      | 74 |

## TABLE DES FIGURES

|     |   |    |
|-----|---|----|
| 1.1 | architecture matérielle d'un capteur . . . . .  | 5  |
| 1.2 | Architecture d'un réseau de capteurs sans fil . . . . .                               | 7  |
| 1.3 | Pile protocolaire dans les RCSFs. . . . .   | 11 |
| 2.1 | Modélisation des zones de couverture et de communication d'un capteur. . . . .        | 18 |
| 2.2 | Exemple de couverture de surface. . . . .   | 19 |
| 2.3 | Exemple de couverture de cible. . . . .   | 19 |
| 2.4 | Exemple de K-couverture et de redondance d'un noeud. . . . .                          | 20 |
| 2.5 | Protocole LDCC : modélisation en pavage triangulaire. . . . .                         | 25 |
| 2.6 | Protocole CACP : exemple de clusters avec $RCom = H*L$ ( $H = 1, 2, 3...$ ) . . . . . | 35 |
| 2.7 | Protocole CACP : Illustration des positions cibles . . . . .                          | 37 |
| 2.8 | Protocole HRP : le processus d'élection du cluster head . . . . .                     | 38 |
| 2.9 | Protocole HRP : la couverture intra-cluster. . . . .                                  | 39 |
| 3.1 | Organigramme du protocole HPRC. . . . .   | 53 |
| 3.2 | Scénario illustratif du protocole HPRC . . . . .                                      | 56 |
| 3.3 | Scénario illustratif de HPRC : état des capteurs à la période 1. . . . .              | 60 |
| 3.4 | Scénario illustratif de HPRC : état des capteurs à la période 2. . . . .              | 62 |
| 3.5 | Scénario illustratif de HPRC : état des capteurs à la période 3. . . . .              | 64 |
| 3.6 | Scénario illustratif de HPRC : état des capteurs à la période 4. . . . .              | 66 |
| 3.7 | Scénario illustratif de HPRC : état des capteurs à la période 5. . . . .              | 68 |
| 4.1 | Taux de couverture VS Nombre de périodes d'activité. . . . .                          | 75 |
| 4.2 | Nombre de noeuds vivants VS Nombre de périodes d'activité. . . . .                    | 77 |
| 4.3 | La taille des clusters VS Nombre de périodes d'activité. . . . .                      | 78 |
| 4.4 | Consommation totale d'énergie VS Nombre de périodes d'activité. . . . .               | 79 |

4.5 Durée de vie VS Taux de couverture. . . . . 80

# INTRODUCTION GÉNÉRALE

Les réseaux ad hoc sont des réseaux sans fil capables de s'organiser sans infrastructure définie préalablement. L'une des applications de ce type de réseaux est les réseaux de capteurs sans fil ayant été une réponse aux exigences de plusieurs domaines d'application, notamment la surveillance environnementale, le domaine médicale et militaire. Un réseau de capteurs se compose généralement d'un grand nombre de dispositifs électroniques de taille réduite, appelés capteurs, réalisant la collecte des grandeurs physiques, leur traitement et leur transmission via un canal radio. Ces capteurs sont dotés de ressources énergétiques limitées et, comme tout autre type de réseau auto-organisateur, les réseaux de capteurs sans fil soulèvent le problème de la consommation d'énergie.

Vu l'inaccessibilité des environnements de déploiement des capteurs, la recharge et le remplacement des batteries sont considérés quasi-impossibles. En effet, l'optimisation de la consommation énergétique visant l'extension de la durée de vie du réseau, reste une des problématiques fondamentales dans ce type de réseaux. Dans cette optique, plusieurs thématiques ont été ouvertes et différents défis ont été soulevés. Bien que les solutions proposées dans ce domaine soient diverses, il reste encore de nombreuses problématiques exigeant une résolution optimale. Parmi ces problématiques, on cite celle de la couverture ayant comme objectif d'assurer la surveillance complète d'une zone géographique cible avec un nombre réduit de noeuds actifs. Par ailleurs, la majorité des applications exigent un déploiement aléatoire avec une forte densité ayant comme conséquence la redondance élevée de la couverture. Cette redondance augmente les risques de collisions et les dépenses inutiles d'énergie.

Dans un réseau de capteurs dense et à plat, tout noeud achemine les données collectées jusqu'à la station de base ce qui fait que le trafic y circulant soit très élevé, augmentant ainsi la consommation énergétique en transmission et en retransmission en cas de collisions. L'une des

stratégies proposées pour résoudre ce problème est l'ordonnancement d'activité permettant la mise en veille d'un ensemble de noeuds redondants. Cette stratégie a été combinée récemment avec le clustering pour réduire simultanément la redondance et le trafic circulant dans le réseau. L'objectif ici, est d'atteindre la maîtrise de la consommation énergétique et ainsi la prolongation de la longévité du réseau.

A partir de ce constat, nous proposons **HPRC** (Pour **H**ierarchical **P**rotocol based-on **R**edundancy evaluation for **C**overage s' garantie), un protocole garantissant la couverture, se base sur le clustering et l'évaluation de la redondance pour l'ordonnancement d'activité. Le protocole HPRC adopte une politique d'élection tenant compte de :

- La ressource énergétique pour permettre aux noeuds possédant le niveau d'énergie le plus élevé d'accomplir les tâches d'agrégation et de transmission de données.
- La distance moyenne par rapport au voisinage afin de favoriser les capteurs les plus proches de leurs voisins. Ceci est justifié par le fait que lorsque la distance d'émission est réduite, l'énergie dépensée et les pertes de paquets sont réduites.
- Le degré pour privilégier les noeuds ayant plus de voisins afin de minimiser le nombre de clusters et augmenter leur taille ayant comme conséquence la mise en valeur de l'agrégation des données.

Pour l'ordonnancement et la désignation des membres de clusters, la solution HPRC utilise une discrétisation de la zone de couverture des noeuds, basée sur la connaissance de positions géographiques, pour l'évaluation de la redondance.

Les scénarios de simulation et les mesures de performance, comparant HPRC à un autre protocole récent, montrent que notre solution assure une couverture quasi-complète avec une meilleure optimisation d'énergie, ceci est dû aux profits tirés de la politique d'ordonnancement et la technique d'élection adoptées.

### **Ce mémoire est organisé en quatre chapitres :**

Dans le **chapitre 1**, nous présentons les concepts relatifs aux réseaux de capteurs sans fil : architecture, domaines d'application et contraintes de réalisation. Ensuite, nous énumérons les différentes problématiques liées à ce type de réseaux, notamment celles de la couverture et du clustering.

Le **chapitre 2** synthétise une étude exhaustive et critique des protocoles de couverture les

plus récents. Dans cette étude, nous établissons une classification de ces protocoles à base du mode d'exécution et de l'information de contrôle. Ensuite, nous décrivons et nous critiquons chaque approche étudiée. Pour montrer la diversité des solutions proposées dans la littérature, nous dréssons enfin un tableau comparatif en fonction de différents critères tels que la garantie de la couverture, l'assurance de la connectivité, etc.

Dans le **chapitre 3**, nous présentons notre contribution baptisée HPRC (pour Hierarchical Protocol based-on Redundancy evaluation for Coverage garantie), elle combine deux techniques largement utilisées dans la littérature : clustering et ordonnancement afin de garantir la couverture. Dans la deuxième partie de ce chapitre, nous expliquons les étapes du protocole proposé à travers un scénario illustratif.

Le **dernier** chapitre consiste en la validation du protocole HPRC, ceci en évaluant ses performances par des scénarios de simulation. Les résultats obtenus seront comparés à un autre protocole recent en matière de : taux de couverture, durée de vie, nombre de noeuds vivants dans le réseau, taille des clusters et conservation d'énergie.

Nous concluons ce travail par une synthèse les avantages de la contribution apportée à la problématique posée et des perspectives que nous souhaiterons aborder dans un futur proche.



# CHAPITRE 1

## PRÉSENTATION DES RÉSEAUX DE CAPTEURS SANS FIL

### 1.1 Introduction

Les besoins récents dans le domaine d'observation, la surveillance des phénomènes physiques ainsi que l'évolution de la communication, la technologie sans fil et l'électronique ont permis de produire de petits dispositifs à faible coût et à faible puissance appelés capteurs. Ces derniers sont dotés d'une capacité de calcul et de perception leurs permettant d'observer des phénomènes généralement dans des endroits hostiles et inaccessibles. Les réseaux de capteurs sans fil ont été l'objet de plusieurs applications tel que la surveillance environnementale, les maisons intelligentes, la sécurité, la médecine... .

Dans ce chapitre, nous définissons un réseau de capteurs, son modèle, ses composants et nous donnons une présentation des différentes applications, caractéristiques et challenges dans un tel type de réseaux. Notre objectif est de donner une vue générale sur les réseaux de capteurs, en montrant les différentes contraintes pour leurs conceptions ainsi que les voies de recherche dans ce domaine.

### 1.2 Définitions et architectures

#### 1.2.1 Qu'est ce qu'un capteur ?

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

## 1.2.2 Architecture d'un capteur

### 1.2.2.1 Architecture matérielle

Un capteur est composé principalement de quatre unités de base : l'unité de captage, l'unité de traitement, l'unité de communication et l'unité de contrôle d'énergie.

Le schéma de la figure 1.1 illustre l'architecture d'un nœud capteur [2].

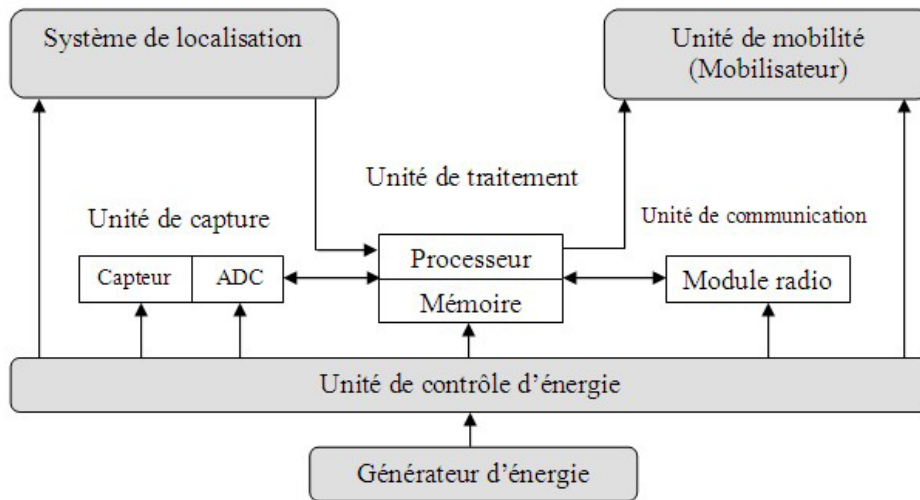


FIGURE 1.1 – architecture matérielle d'un capteur

- **L'unité d'acquisition** : composée d'un capteur qui obtient des mesures sur les paramètres environnementaux et d'un convertisseur Analogique/Numérique qui convertit l'information relevée et la transmet à l'unité de traitement.
- **L'unité de traitement** : composée d'un processeur et d'une mémoire intégrant un système d'exploitation spécifique (TinyOS [3], par exemple). Elle acquiert les informations en provenance de l'unité d'acquisition et les envoie à l'unité de communication. L'unité de traitement est chargée aussi d'exécuter les protocoles de communications permettant de faire collaborer le capteur avec d'autres. Elle peut aussi analyser les données captées.
- **L'unité de communication** : unité responsable de toutes les émissions et réceptions de données via un support de communication radio. Elle peut être de type optique, ou de type radiofréquence.
- **L'unité de contrôle d'énergie** : un capteur est muni d'une batterie pour alimenter tous ses composants. Cependant, à cause de sa taille réduite, la batterie dont il dispose

est limitée et généralement irremplaçable [4].

Dépendant de l'application, un capteur peut comporter des unités additionnelles, système de localisation,...

### 1.2.2.2 Architecture logicielle

Afin qu'un capteur puisse collecter et transmettre des données environnementales, il doit disposer de cinq sous systèmes software de base [3,5].

- **Le système d'exploitation**

Pour supporter les différentes opérations des noeuds, il est important d'avoir un système d'exploitation open source conçu spécifiquement pour les réseaux de capteurs. Les bibliothèques des systèmes d'exploitation incluent des protocoles réseaux, des services distribués, des drivers de capteurs, des outils d'acquisition de données, etc. Nous présentons ci-dessous les systèmes d'exploitation les plus connus :

- TinyOs (Berkley) [3] : est un système d'exploitation open source. Il est conçu pour les réseaux de Microsystèmes autonomes communicants. Il dispose d'une plateforme modulaire et d'une mémoire minimum. Il a un fonctionnement événementiel. Il est utilisé et commercialisé par plus de 500 laboratoires dans le monde.
- NutOs [3] et BTNut [3] : NutOs est un système d'exploitation open source fait pour ETHERNUT (Ethernut embedded ethernet board)<sup>1</sup>. Ce système d'exploitation permet des Multitâches coopératives. Dans ce système d'exploitation, la pile TCP/IP, les protocoles DHCP, DNS, HTTP et PPP sont implémentés. NutOs est la base du système d'exploitation BTNut utilisé dans les BTnodes<sup>2</sup>.
- Think [3] : est une implémentation du modèle Fractal<sup>3</sup> en C. Il a été développé par l'INRIA (Institut National de Recherche en Informatique et en Automatique) et France Telecom RD pour créer des systèmes d'exploitation pour les systèmes embarqués et les applications s'exécutant dessus. Think permet l'allocation dynamique.

- **Les drivers de perception** : ce sont les modules logiciels qui contrôlent les fonctions de base de perception du capteur. Les configurations appropriées doivent être chargées dans le capteur [5].

---

1. est un système d'exploitation disponible pour micro-contrôleurs

2. plate-forme flexible, légère et autonome de communication sans fil et de traitement basée sur une radio Bluetooth et un microcontrôleur

3. modèle général dédié à la construction, au déploiement et à l'administration de systèmes logiciels complexes, tels que les systèmes d'exploitation.

- **Processeur de communication** : ce code contrôle les fonctions de communication telles que le routage, la maintenance de topologie, l'accès au media radio, décryptage, etc [5].
- **Driver de communication** : ce module logiciel contrôle la synchronisation, l'encodage du signal et la modulation. Ce code est implémenté dans la couche physique [5].
- **Intergiciel de traitement de données** : Ce code englobe le traitement des données et toutes les applications de base supportées au niveau du noeud (perception, sauvegarde de donnée, etc) [5].

### 1.2.3 Réseau de capteur

Les réseaux de capteurs sans fil se composent généralement d'un grand nombre de capteurs, communiquants via des liens radio pour le partage d'informations et le traitement coopératif. Ces dispositifs sont déployés aléatoirement dans une zone d'intérêt pour surveiller des phénomènes divers. Après le déploiement initial, les capteurs peuvent s'auto organiser en une infrastructure réseau appropriée, souvent en mode multi-sauts. Les données collectées par ces capteurs sont acheminées directement ou via un routage multi-sauts à un noeud considéré comme "point de collecte", appelé station de base. La communication entre les stations puits et l'utilisateur se fait via Internet ou satellite [2]. Ce processus est illustré dans la figure 1.2.

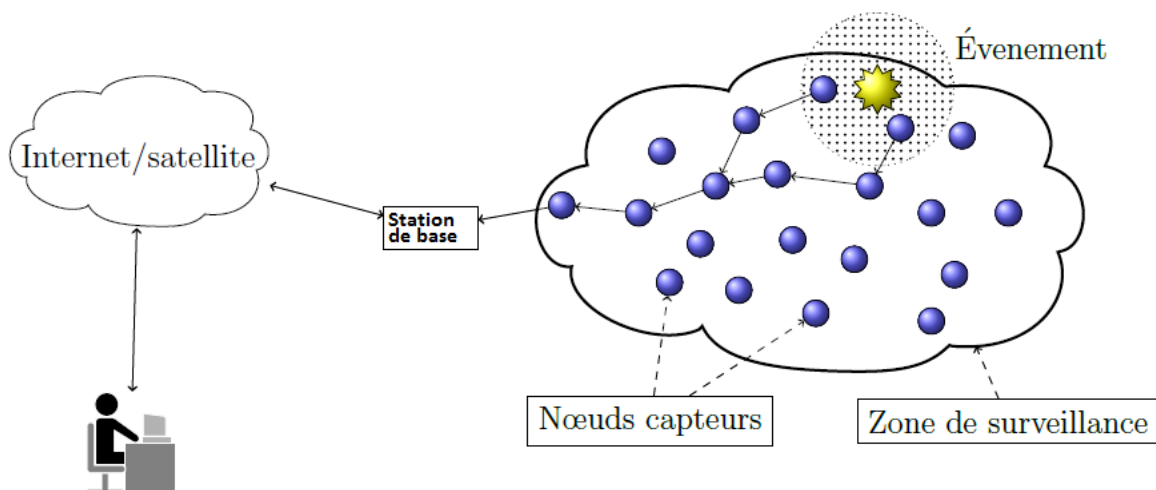


FIGURE 1.2 – Architecture d'un réseau de capteurs sans fil

## 1.3 Contraintes liées aux réseaux de capteurs

### 1.3.1 Durée de vie limitée

D'après les recherches, il existe trois manières de déterminer la fin de la durée de vie d'un réseau [6] :

- Jusqu'à ce que le premier noeud ne pourra plus fonctionner.
- Jusqu'à ce qu'une des tâches du réseau ne pourra pas être satisfaite.
- Jusqu'à ce que le réseau ne sera plus connexe.

La durée de vie n'est pas seulement liée à la consommation individuelle d'un noeud, mais aussi à l'équilibrage de charge entre tous les noeuds du réseau. Le protocole LEACH [7] prend en considération l'énergie moyenne consommée et détermine le nombre de noeuds défaillants comme un critère de durée de vie. En revanche, X. Di [8] tenant compte du taux de couverture, considère la durée de vie terminée lorsque la couverture assurée par le réseau est inférieure au seuil exigé par l'application.

### 1.3.2 Bande passante limitée

Une des caractéristiques primordiales des réseaux basés sur la communication sans fil est l'utilisation d'un médium de communication partagé. Ce partage fait que la bande passante réservée à un noeud est limitée [2].

### 1.3.3 Scalabilité

Le nombre de noeuds d'un réseau dépend de l'application à laquelle est destiné ce réseau. Il peut atteindre plusieurs centaines ou plusieurs milliers. Le nombre de noeuds peut atteindre les centaines dans une région du champ de captage ou cette dernière peut avoir une surface de 10 m<sup>2</sup>. La densité  $\mu$  peut être calculée d'après [9] comme suit :

$$\mu(\text{RCom}) = (N * \mu * \text{RCom}^2) / \mu(\text{R}) \quad (1.1)$$

Ou :

$N$  : le nombre de noeuds dans la région  $A$ .

$\text{RCom}$  : la portée de transmission radio.

$\mu(\text{R})$  : le nombre de noeud dans un rayon de transmission  $R$  d'une région  $A$ .

### 1.3.4 Topologie

Des centaines, voire des milliers de noeuds peuvent être déployés dans un champs de captage. Les capteurs sont distants d'une dizaine de pieds les uns des autres. La densité des noeuds peut atteindre 20 noeuds/m<sup>2</sup> [9]. Le déploiement d'un grand nombre de noeuds exige le contrôle et le maintien de la topologie. La maintenance et le changement de topologie s'effectuent en trois phases :

- **Phase de pré-déploiement et de déploiement** : les capteurs sont généralement déployés aléatoirement dans des zones d'intérêt hostiles (capteurs largués par un avion), ou placés par un robot ou un humain dans les environnements accessibles .
- **Phase de post-déploiement** : après le déploiement, le changement de topologie survient du fait des changements de position des capteurs (obstacle, bruit, etc) ou encore des dysfonctionnements de ces derniers. Le nombre de capteurs opérationnels peut être amené à diminuer.
- **Phase de redéploiement des noeuds additionnels** : des noeuds additionnels peuvent être redéployés à chaque moment pour remplacer les noeuds en panne ou à cause d'un changement dans la distribution des tâches [10].

### 1.3.5 Agrégation des données

L'une des caractéristiques des réseaux de capteurs est la possibilité de réduire la quantité de données circulant dans le réseau, afin de conserver l'énergie, en fusionnant les données par des noeuds particulier du réseau. Ce processus est appelé agrégation de données. L'agrégation exige, non seulement, la transmission des données mais aussi les messages de contrôle, ce qui impose des contraintes sur l'architecture du réseau [11].

## 1.4 Communication dans les réseaux de capteurs

Nous avons constaté dans les réseaux de capteur sans fil qu'il y a deux types de noeuds : noeuds capteurs qui captent et transmettent de l'information et les stations de bases (sinks) qui traitent cette information et jouent aussi le rôle de relais entre les capteurs et les stations utilisateurs.

La topologie multi-sauts que l'acheminement de l'information a créé, donne lieu à quatre types de communications :

- **Communication noeud capteur à un noeud capteur** : ce type de communication directe est employé pour des opérations locales, par exemple pendant le processus de clustering<sup>4</sup> ou le processus de création de route.
- **La communication noeud capteur à un noeud intermédiaire** : les données capturées sont transmises d'un noeud capteur à un noeud intermédiaire. Ce type de communication est souvent unicast.
- **La communication noeud intermédiaire à un noeud capteur** : les requêtes et les signalisations de messages sont souvent multicast, ils sont diffusées par les noeuds intermédiaires pour atteindre un sous-ensemble de noeuds immédiatement.
- **La communication noeud intermédiaire à un noeud intermédiaire** : la communication entre ces noeuds peut être dans la plupart du temps unicast. Noter que chaque noeud est équipé d'une radio d'émissions / réception [12].

### 1.4.1 Pile protocolaire dans les réseaux de capteurs

Dans tous les types de réseaux (filaires ou sans fil), le processus de communication se représente sous forme de couches.

Dans cette architecture, chaque couche a un rôle prédéfini qu'elle assure avec un ensemble de protocoles spécifiques mais à part la communication, les réseaux de capteur doivent assurer d'autres contraintes telles que la mobilité, l'économie d'énergie et la gestion des tâches comme le montre la figure suivante [3] :

- **La couche physique**

Cette couche décrit les procédures et les fonctions mécaniques et électriques nécessaires pour établir, maintenir et libérer une connexion physique entre deux ou plusieurs capteurs. Elle est responsable de la sélection des fréquences (utilisation des bandes ISM), de la détection d'un signal, et du traitement du signal (la modulation).

- **La couche liaison de données**

Cette couche a pour rôle le multiplexage des flux de données, le partage de l'accès au medium et le contrôle d'erreurs. Elle assure une connexion point à point ou point à multipoint fiable dans une communication réseau.

---

4. processus de l'organisation du réseau en groupes, chaque groupe est gérée par un noeud appelé cluster-head.

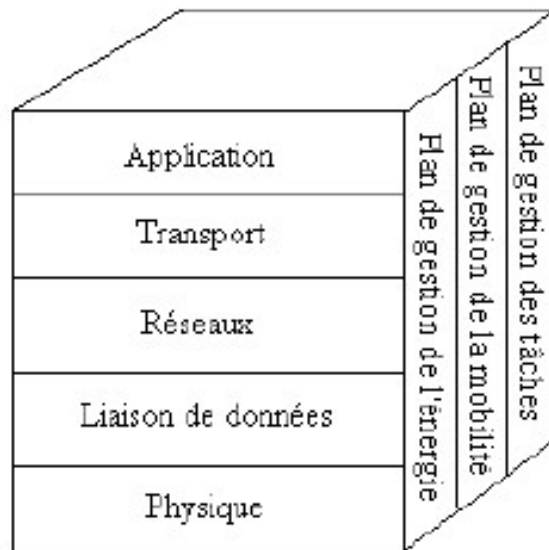


FIGURE 1.3 – Pile protocolaire dans les RCSFs.

- **La couche réseau**

Cette couche est responsable des fonctions de routage et de la gestion de la topologie.

- **La couche transport**

Cette couche est responsable du maintien des flux de données dans les applications utilisées et de la sauvegarde des données dans le cache des capteurs. Elle est particulièrement nécessaire pour accéder au réseau de capteurs par le biais d'un réseau externe comme l'internet. A l'instar des autres couches, de nouvelles approches doivent être mises en place pour faire face aux contraintes inhérentes à ce type de réseau.

- **La couche application**

Cette couche est conçue suivant les fonctionnalités des capteurs. Elle doit fournir des mécanismes :

- pour l'interprétation des données perçues.
- qui rendent transparents à l'utilisateur les logiciels utilisés dans les couches inférieures.
- d'agrégation de données.
- de synchronisation des noeuds, d'authentification, de distribution de clés dans le but d'assurer la sécurité des données [13].

- **Plan de gestion de l'énergie (Power management plane)**



La vie du capteur montre une dépendance forte à l'égard de la vie de la batterie. La consommation de l'énergie dans un réseau de capteurs, peut être divisée selon l'ordre décroissant suivants [14] : la communication (émission et réception), le traitement des données et enfin l'acquisition ou la capture. Ce plan gère la manière dont les noeuds utilisent leur énergie. Par exemple le noeud doit se mettre en sommeil après la réception d'un message en provenance d'un voisin. Il doit aussi informer ses voisins de son incapacité à participer dans le routage quand sa réserve d'énergie atteint un seuil particulier.

- **Plan de gestion de la mobilité (Mobility management plane)**

Puisque les noeuds peuvent être mobiles, un système de gestion de mobilité doit exister. Un tel système doit alors être capable d'enregistrer les mouvements du noeud afin de l'aider à se localiser. Parfois l'utilisateur souhaite changer l'emplacement du noeud, ou bien une nouvelle topologie du réseau nécessite une auto organisation des noeuds. Dans ce cas, le système de gestion de la mobilité doit être capable de commander le noeud pour réaliser les mouvements nécessaires.

- **Plan de gestion des tâches (Task management plane)**

Le niveau de gestion des tâches équilibre la distribution des tâches de captage sur les différents noeuds concernés afin d'optimiser la consommation d'énergie. En effet, la nature des capteurs ou leur niveau d'énergie implique que les noeuds peuvent avoir des rythmes différents de travail [15].

## 1.5 Domaines d'application

La miniaturisation des capteurs, le coût de plus en plus faible, la large gamme des types de capteurs disponibles ainsi que le support de communication sans fil utilisé, permettent aux réseaux de capteurs de se développer dans plusieurs domaines d'application, parmi lesquels nous citons [4] :

### 1.5.1 Les applications militaires

L'exploitation militaire est l'une des principales applications des réseaux de capteurs. Dans ce contexte, l'emploi des réseaux de capteurs peut être dans la surveillance des champs de bataille afin de fournir des renseignements concernant l'emplacement, le nombre, le mouvement, et l'identité des soldats et des véhicules, ou bien encore pour la détection des agents chimiques, biologiques et nucléaires.

### 1.5.2 Les applications de la surveillance

Les réseaux de capteurs sont déployés pour le contrôle et la détection de problèmes structuraux dans les bâtiments et ponts. Ceci aidera à détecter les vibrations qui provoquent des fissures et des altérations dans la structure suite à un séisme ou au vieillissement de cette dernière.

### 1.5.3 Les applications environnementales

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

Dans le domaine de l'agriculture, les capteurs peuvent être utilisés pour réagir convenablement aux changements climatiques.

### 1.5.4 Les applications médicales

Dans le domaine de la médecine, les réseaux de capteurs peuvent être utilisés pour assurer une surveillance permanente des organes vitaux de l'être humain grâce à des micro-capteurs pouvant être avalés ou implantés sous la peau (surveillance de la glycémie, détection de cancers, etc) [16].

### 1.5.5 Les applications domestiques

Avec le développement technologique, les capteurs peuvent être embarqués dans des appareils, tels que les aspirateurs, les fours à micro-ondes, les réfrigérateurs, les magnétoscopes ... [17]. Ces capteurs embarqués peuvent interagir entre eux et avec un réseau externe via Internet pour permettre à un utilisateur de contrôler les appareils domestiques localement ou à distance. Le déploiement des capteurs de mouvement et de température dans les futures maisons dites intelligentes permet d'automatiser plusieurs opérations domestiques telles que : la lumière s'éteint et la musique s'arrête quand la chambre est vide, la climatisation et le chauffage s'ajustent selon les points multiples de mesure, l'alarme est déclenchée par le capteur anti-intrusion quand un étranger veut pénétrer dans la maison.

### 1.5.6 Les applications commerciales

Il est possible d'intégrer des capteurs au processus de stockage et de livraison dans le domaine commercial. Le réseau ainsi formé pourra être utilisé pour connaître la position, l'état et la direction d'un paquet. Il devient alors possible pour un client qui attend la réception d'un paquet, d'avoir un avis de livraison en temps réel et de connaître la localisation actuelle du paquet [18].

## 1.6 Quelques problématiques liées aux RCSFs

Dans cette section nous allons présentés quelques problématiques liées aux réseaux de capteurs sans fil.

- **Systèmes d'exploitation**

Beaucoup de recherches ont été faites pour la conception des systèmes d'exploitation pour les capteurs. Le premier et le plus connu a été développé par l'université de Berkeley TinyOS [3]. Ce système est un modèle basé événement et sa bibliothèque se compose des protocoles réseaux, des pilotes de capteurs et des outils d'acquisition de données.

Pour langage de programmation compatible avec TinyOS, on trouve NESC [19] possédant des dispositifs tels que : conduite des évènements, exécution parallèle et éléments d'orientation etc. Ces derniers correspondent aux composants et au traitement d'évènements de TinyOS.

- **L'agrégation des données**

Pendant l'agrégation nous devons aussi prendre en considération quelques problèmes : les erreurs dans les messages, les messages perdus, la redondance des données, la synchronisation entre les noeud, etc. Plusieurs travaux ont été publiés dans ce domaine, nous citons ceux de Karlof et al [11] qui ont introduit deux algorithmes (exacte et approximatif), pour l'agrégation des données. Le processus se fait en deux niveaux. Premièrement, des agrégateurs locaux sont utilisés pour fusionner les données reçues à partir des noeuds locaux, puis des agrégateurs maîtres sont choisis de façon optimale pour réaliser le deuxième niveau d'agrégation.

- **Localisation des noeuds**

Quand l'information est obtenue par un noeud donné, sa position physique est celle à laquelle l'information a été capturée. Par conséquent, la position d'un noeud dans un réseau de capteur est l'information essentielle pour analyser l'information capturée.

Diverses recherches se sont concentrées sur la localisation des noeuds. Par exemple,

Moore[20] propose un algorithme qui localise avec succès des noeuds dans un réseau de capteur avec des mesures de distance bruyantes. Shang et al [21] emploient MDS (Multi Dimensional Scaling) pour identifier les noeuds. Ceci utilise l'information de connectivité pour savoir quels sont les noeuds impliqués dans telle ou autre communication, pour dériver les endroits des noeuds dans le réseau de capteurs. Cette méthode permet aussi de tirer profit des données additionnelles : les distances estimées entre les voisins ou les positions connues pour certains noeuds ancrés.

- **Synchronisation**

La gestion du temps dans les RCSFs est essentielle pour l'accomplissement des tâches telles que l'estampillage des événements, le contrôle des cycles d'opération ainsi que leur synchronisation. Le NTP (Network Time Protocol) [22] permet l'attribution des estampilles en temps réel en donnant les erreurs maximales, ainsi l'ordre global et partiel des événements peut être obtenu avec une exactitude basée sur les estampilles. Le NTP exige l'échange fréquent de messages de synchronisation des horloges dans chaque noeud.

- **Sécurité**

La sécurité dans les RCSFs est l'une des problématiques critiques, mais un minimum de recherche a été fait comparé à d'autres aspects. Les ressources des noeuds capteurs sont limitées d'où une technologie différente de sécurité pour les RCSFs est exigée.

Wood [23] présente un aperçu sur beaucoup de genres d'attaques DoS (Denial of Service) dans les RCSFs et discute des technologies de défenses.

Karlof [11] a implémenté TinySec comme module de chiffrement pour la couche liaison de données, celui-ci réalise les opérations sur les ressources de TinyOS qui sont très limitées.

- **Routage**

Le routage désigne le mécanisme par lequel les données d'un équipement expéditeur sont acheminées jusqu'à leur destinataire, même si aucun des deux ne connaît le chemin complet que les données devront suivre. Ce mécanisme est très important dans tous les réseaux. Sans routage, la communication, sur n'importe quel réseau, serait impossible. En fait, le rôle du routage dans les réseaux de capteurs est, entre autre, défini comme l'optimisation de la consommation d'énergie de façon à maximiser la durée de vie du réseau.

Beaucoup de protocoles de routage ont été proposés pour les WSNs. Ces derniers prennent en compte plusieurs facteurs, notamment la conservation d'énergie et la topologie dynamique du réseau.

- **Clustering**

le clustering consiste à découper le réseau en groupes de noeuds appelés Cluster. Pour chaque cluster un noeud maître appelé " cluster-head " est élu selon l'état courant du réseau afin d'accomplir des tâches spécifiques. Il existe deux configurations possibles pour les clusters : dans la première, les membres du cluster ne communiquent qu'avec le cluster-head, par contre dans la deuxième, ils peuvent utiliser d'autres noeuds comme passerelles vers le cluster-head.

Le clustering aide à minimiser la consommation d'énergie en réduisant la quantité de données générées par tous les membres du cluster grâce à l'agrégation faite par le cluster-head. De plus, il permet de faciliter la localisation des noeuds [24]. Dans notre travail, nous nous intéressons aux protocoles de clustering qui servent à réaliser la préservation de la couverture.

- **Couverture**

Le problème de la couverture de surface consiste à surveiller tous les points physiques de la zone d'intérêt par au moins un capteur [25] pour assurer une couverture maximale. Dans le cas du déploiement aléatoire à une forte densité, une redondance élevée peut être engendrée. En effet, une même portion peut être surveillée par plusieurs capteurs ce qui épuise inutilement leur énergie. Donc, la thématique posée dans la couverture est : "Comment mettre en veille un ensemble de noeuds redondant tout en assurant la couverture complète avec les capteurs restant actifs?".

L'objectif du chapitre suivant est l'étude de quelques protocoles proposés pour préserver la couverture tout en prolongeant la durée de vie du réseau.

## 1.7 Conclusion

Les réseaux de capteurs sans fil sont une nouvelle technologie qui a surgit après les grands progrès technologiques concernant le développement des capteurs intelligents, des processeurs puissants et des protocoles de communication sans fil. Le faible coût et la facilité de mise en place des capteurs ont permis leur intégration dans une large gamme d'applications : militaires, médicales, etc. Dans ce chapitre, nous avons représenté les réseaux de capteurs sans fil, leur architecture, caractéristiques, contraintes ainsi que les types de communication utilisés. Enfin, nous avons cité les différentes problématiques et plus précisément, celles de la couverture et du clustering que nous allons étudier dans le prochain chapitre.

## CHAPITRE 2

# TAXONOMIE DES PROTOCOLES DE COUVERTURE DANS LES RÉSEAUX DE CAPTEURS SANS FIL

### 2.1 Introduction

La couverture est une métrique de performance importante pour mesurer la QoS (Qualité de Service) dans les réseaux de capteurs sans fil [26]. La problématique liée à la couverture consiste à surveiller tous les points d'une zone d'intérêt en tenant compte de la contrainte d'énergie. Pour atteindre cet objectif, plusieurs stratégies ont été proposées, telles que le placement des noeuds dans les environnements accessibles et l'ordonnancement d'activité dans les environnements hostiles. Dans les stratégies d'ordonnancement, afin de conserver la précieuse énergie et prolonger la durée de vie du réseau, on met en veille les noeuds dits redondants tout en assurant la couverture complète de la zone cible.

Dans ce chapitre, nous abordons une étude critique des protocoles de couverture existant dans la littérature ainsi que leur classification et leur comparaison. Ces protocoles sont généralement basés sur l'ordonnancement d'activité et le clustering.

### 2.2 Préliminaires

#### 2.2.1 Zone d'intérêt

La zone d'intérêt (appelée aussi zone de déploiement) représente l'espace géographique surveillé par un réseaux de capteurs sans fil. Elle est modélisée sous forme d'un plan bidimensionnel  $P$ .

### 2.2.2 Zone de couverture d'un capteur

La zone de couverture du capteur  $C_i$  est modélisée par un disque dont il est le centre (voir figure 2.1) avec un rayon égal à son rayon de captage (nommé **RCap**). Dans un plan  $P$ , l'ensemble de points couverts par le capteur  $C_i$  est représenté comme suit :

$$S(C_i) = \{X/X \in P \wedge d(X, X_i) \leq RCap\}. \quad (2.1)$$

Où  $X_i$  représente les coordonnées de localisation du capteur  $C_i$  et  $X$  représente les coordonnées d'un point quelconque dans le plan  $P$ . La distance euclidienne entre le capteur  $C_i$  et un point dans le plan  $P$  est notée  $d(X, X_i)$ .

### 2.2.3 Zone de couverture d'un réseau de capteurs

La surface  $S$  couverte par un réseau de capteurs est l'union des zones couvertes par tous les noeuds  $C_i$  ( $1 \leq i \leq n$ ) formant le réseau, tel que  $n$  est le nombre de noeuds déployés [24].

$$S = \bigcup_{i=1}^n S(C_i). \quad (2.2)$$

### 2.2.4 Modélisation du canal radio

Un réseau de capteurs sans fil est représenté par un graphe  $G(V, E)$  [24], avec  $V$  l'ensemble des sommets (les noeuds du réseau) et  $E \in V^2$  l'ensemble des arêtes représentant les communications entre les noeuds. Soit la portée de communication  $RCom$ , le modèle du disque unitaire définit l'ensemble des arêtes  $E$  par :

$$E = \{(u, v) \in V^2 / u \neq v \wedge d(u, v) \leq RCom\}. \quad (2.3)$$

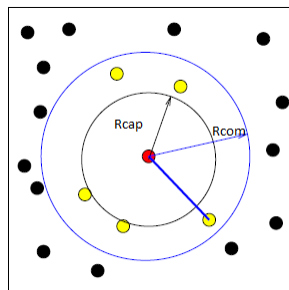


FIGURE 2.1 – Modélisation des zones de couverture et de communication d'un capteur.

### 2.2.5 Types de couverture et redondance des noeuds

Parmi les types de couverture définis dans la littérature [27] selon l'objet à surveiller, nous avons la couverture de surface et la couverture de cible.

- **La couverture de surface** : consiste à surveiller une zone géographique, appelée zone d'intérêt, dans laquelle chaque point doit être surveillé par au moins un noeud [28]. Un exemple de ce type de couverture est illustré par la figure 2.2.

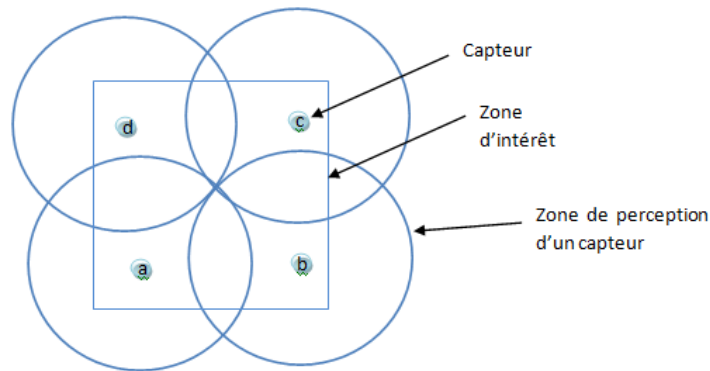


FIGURE 2.2 – Exemple de couverture de surface.

- **La couverture de cible** : modélisée comme un ensemble discret de points dans un champs de captage. Un point  $p$  de la zone  $S$  est couvert par un capteur  $C_i$  s'il se trouve dans la zone de perception de  $C_i$ .  
 $p \in S$  est couvert par  $C_i \Rightarrow d(p, C_i) \leq R_{\text{Cap}}(C_i)$  ou  $p \in S(C_i)$  [24]. La figure ci-après montre un exemple de couverture de cibles.

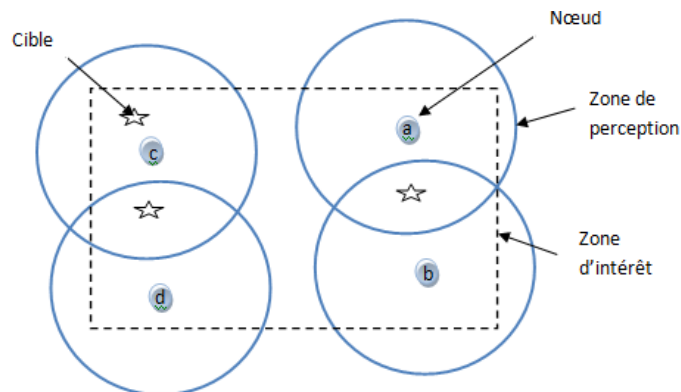


FIGURE 2.3 – Exemple de couverture de cible.



Une autre classification a été définie dans la littérature selon le nombre de capteurs surveillant un objet ou un point de la zone d'intérêt [24] : la 1-couverture et la k-couverture. La forme la plus simple lorsque tout point "p" de la zone d'intérêt est couvert par un et un seul capteur, c-à-d, tout point "p" se trouve dans la zone de perception d'un seul capteur. Dans ce cas, on parle de la 1-couverture. En revanche, on parle de la k-couverture ( $k > 1$ ) lorsque tout point "p" de la zone d'intérêt est couvert par plus d'un capteur. Un noeud est dit redondant si sa zone de perception est couverte par d'autres noeuds (voir figure 2.4).

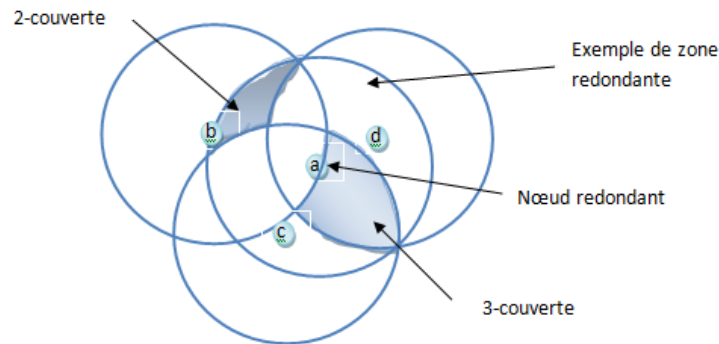


FIGURE 2.4 – Exemple de K-couverture et de redondance d'un noeud.

## 2.2.6 Ordonnancement et clustering

Une des stratégies proposées dans la littérature pour réduire la redondance de couverture est l'ordonnancement d'activité, il consiste à gérer l'activation et la désactivation des noeuds capteurs. Si une zone est surveillée par un capteur alors elle peut être surveillée par plusieurs en raison de la forte densité du réseau. En effet, lorsqu'un noeud est redondant, il peut temporairement passer en mode passif lui permettant de conserver l'énergie. L'objectif principal de l'ordonnancement d'activité consiste à décider de l'état du capteur et pour combien de temps il reste dans cet état. Cette stratégie permet une couverture complète de la zone d'intérêt tout en prolongeant la durée de vie du réseau.

Le clustering consiste à organiser le réseau en clusters. Chaque cluster contient un chef, appelé **cluster-head**, auquel les membres envoient leurs données pour être agrégées et transmises par ce dernier à la station de base. L'objectif du clustering est d'améliorer les performances du réseau en réduisant le trafic y circulant ainsi que les collisions et les effets de retransmission des données (perte du temps et d'énergie). Cette technique a été utilisée récemment dans les protocoles de couverture pour réduire, par le mécanisme d'agrégation, le reste de la redondance non traitée par l'ordonnancement d'activité.

## 2.3 Modes d'exécution des protocoles

Plusieurs protocoles de couverture ont été proposés dans la littérature selon deux types d'approches [29] :

### 2.3.1 Les approches centralisées

Dans les solutions centralisées, l'information de la topologie doit être centralisée dans la station de base, afin de maintenir les informations nécessaires pour chaque noeud et prendre les décisions d'activation qui seront ensuite ordonnées aux noeuds. Ce type d'approches est précis mais l'inconvénient réside dans la difficulté de construire un état global cohérent en cas de défaillance du cite centralisateur.

### 2.3.2 Les approches distribuées

Dans les solutions distribuées, aucune infrastructure ou hiérarchie n'existe. En effet, le comportement de chaque objet n'est influencé que par ceux de ses voisins directs. L'objectif est d'obtenir un comportement global cohérent à partir de comportements individuels. Dans ce type d'approche, chaque noeud décide de sa propre activité en ne se basant que sur l'observation de ses propres voisins.

Pour une meilleure organisation du réseau en niveaux, les approches hiérarchiques (basées sur le clustering) ont été introduite dans les protocoles de couverture [7].

## 2.4 Types d'informations de contrôle utilisées dans les protocoles

En plus du mode d'exécution, nous pouvons classifier les protocoles proposés dans la littérature selon le type d'information de contrôle utilisée, en trois classes :

### 2.4.1 Les protocoles basés sur les informations de localisation

Dans cette classe, l'information de localisation des noeuds est nécessaire pour la spécification d'un protocole et tout noeud doit connaître sa position géographique. L'obtention de cette information nécessite de doter le capteur d'un dispositif de localisation appelé GPS (pour Global Position System) ou d'utiliser une technique de localisation telle que l'ancre mobile [29].

### 2.4.2 Les protocoles basés sur la distance

Certains protocoles utilisent l'estimation de la distance entre les noeuds voisins, au lieu de leurs coordonnées, afin d'ordonner l'activité [30, 31]. Les informations de distance peuvent être obtenues à partir de la force du signal reçu, en utilisant une antenne bidirectionnelle, ou estimées à partir du nombre de voisins, dans le cas où la densité de déploiement du réseau est connue.

### 2.4.3 Les protocoles basés sur le nombre de sauts

Dans cette classe, les protocoles utilisent comme information de contrôle le nombre de sauts qui mesure le nombre de transmissions nécessaires à un capteur pour faire parvenir un message à la station de base. Le nombre de sauts peut être obtenu par inondation originale de la station de base ou d'un autre noeud du réseau.

## 2.5 Etude critique de quelques protocoles de couverture

Les protocoles de couverture visant à minimiser la consommation d'énergie en réduisant la redondance des noeuds et, par conséquent, étendre la durée de vie du réseau, sont nombreux et variés. Chaque protocole est conçu de façon à fonctionner suivant des critères ou des conditions particulières à l'application. Dans ce qui suit, nous présentons certains protocoles de types différents : distribués, centralisés et basés sur le clustering.

### 2.5.1 Protocoles distribués basés sur le nombre de sauts

#### 2.5.1.1 Energy-efficient Coverage and Connectivity preserving Routing Algorithm under border effects : ECCRA

Y. Jin et al [25], ont proposé le protocole ECCRA (pour Energy-efficient Coverage and Connectivity preserving Routing Algorithm under border effects) pour la préservation de la couverture et de la connectivité en se basant sur un protocole de routage. En effet, à chaque instant, tout noeud actif doit être capable de maintenir un chemin composé d'un minimum de noeuds intermédiaires vers la station de base. En outre, cet algorithme peut ordonner dynamiquement les noeuds pour réaliser le taux de couverture et la probabilité de connectivité demandés.

#### Description de l'algorithme

L'opération de collecte de données est divisée en périodes. Dans chaque période, on a deux phases : la phase de découverte de chemins et la phase de transmission. La phase de découverte

de chemins se compose de trois étapes.

La première étape consiste à diviser les noeuds en plusieurs sous-ensembles et garder aléatoirement un seul sous-ensemble en mode actif, pour une certaine durée, tandis que les noeuds des autres sous-ensembles sont dans l'état passif pour conserver leur énergie. Les sous-ensembles sont ordonnancés périodiquement afin d'équilibrer la consommation d'énergie.

Dans la deuxième étape, un ensemble de noeuds à 1-saut de la station de base est d'abord désigné. Ensuite, les autres noeuds choisissent comme pères les noeuds qui sont à leurs portés et ayant un nombre minimal de sauts. Chaque noeud maintient une table de routage dont chaque élément correspond à un père. A la fin de cette étape, un ensemble minimal de noeuds intermédiaires est déterminé pour chaque noeud.

Durant la troisième étape, on construit un chemin connexe avec un nombre minimal de sauts. le principe de cette étape est comme suit : si un noeud  $u$  à  $h$ -sauts n'a pas de père dans son **sous – ensemble <sub>$i$</sub>**  alors il envoie un message COMMAND à un de ses pères choisi aléatoirement. Ensuite, il ajoute le noeud père choisi à son **sous – ensemble <sub>$i$</sub>** . Sinon, il envoie un message NOTIFY à l'un de ses pères choisi aléatoirement (dans le même **sous – ensemble <sub>$i$</sub>** ).

Pour organiser l'envoi des messages "COMMAND/NOTIFY", un temps de backoff est appliqué. En effet, un noeud ne doit transmettre de messages que lorsqu'il reçoit le dernier message de tous ses fils, donc les noeuds les plus éloignés ont le backoff le plus court et un noeud ne peut transmettre qu'après l'expiration de son backoff.

Dans la deuxième phase, après que le chemin minimal ne soit formé pour chaque noeud, les membres de chaque sous-ensemble transmettent périodiquement les données capturées au point de collecte. L'algorithme proposé équilibre la consommation d'énergie du noeud par le mode d'ordonnancement Round-Robin. En outre, chaque sous-ensemble peut maintenir la couverture et la connectivité du réseau simultanément.

## Discussion et critiques

Les résultats de simulation du protocole ECCRA ont montré une bonne préservation du taux de couverture et une probabilité de connectivité proche ou égale à 1. L'inconvénient de cet algorithme réside dans le choix aléatoire du père. En effet, certains noeuds seront plus surchargés par le nombre de fils par rapport à d'autres. Etant donné que le noeud ne peut transmettre ses données à son père que lorsqu'il reçoit les données capturées de tout ses fils,

et les agrège, alors l'énergie consommée par ce noeud lors de l'agrégation et la transmission augmente avec la croissance du nombre de fils, ceci provoque un déséquilibre d'énergie entre les noeuds. Ce problème apparaît généralement lorsque un noeuds n'a pas d'information sur le nombre de fils du père choisi.

De plus, le nombre de messages COMMAND/NOTIFY reste important dans le cas où la connectivité des sous-ensembles est presque nulle , même si les auteurs ont essayé de réduire le trafic avec un back-off mais ce dernier augmente de plus en plus que les noeuds deviennent proches de la station de base. Ceci pose un problème lors du passage à l'échelle car l'intervalle du temps de la phase de création de chemins augmente avec ces back-offs.

### 2.5.1.2 Layred-Diffusion based Coverage Control Protocol : LDCC

B. Wang et al [30], ont proposé un algorithme distribué nommé LDCC (pour Layred-Diffusion based Coverage Control Protocol), il permet de contrôler la redondance en se basant sur le nombre de sauts et la diffusion par couche avec la technique du pavage triangulaire.

Le principe de base du protocole LDCC consiste à utiliser le nombre de sauts, comme information de décision d'état dans la technique du pavage triangulaire. Cette technique permet de diviser la zone d'intérêt en triangles équilatéraux avec une longueur de coté égale au rayon de communication, dont un sommet à H-sauts et les deux autres à H+1-sauts (voir la figure 2.5). Dans le cas où les noeuds utilisent une puissance de transmission fixe, le nombre de sauts représente le nombre de transmission, par noeuds intermédiaires, nécessaires pour qu'un noeud puisse délivrer ses données à la station de base. Le nombre de sauts est généralement obtenu à travers l'inondation lancée par la station de base ou par un autre noeud du réseau.

D'après la figure 2.5, tout noeud sur ou à l'intérieur du cercle de rayon RCom est dit à 1-saut, les noeuds qui sont entre les deux cercles sont dits à 2-sauts et ainsi de suite.

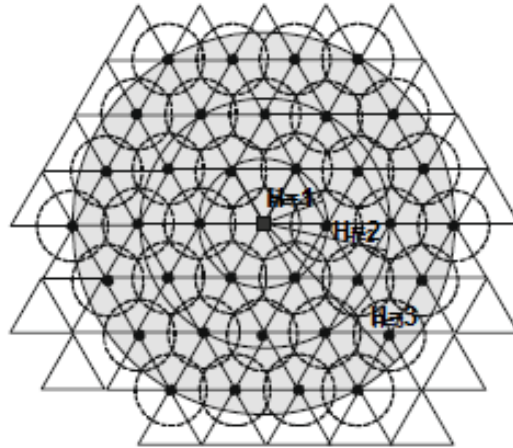


FIGURE 2.5 – Protocole LDCC : modélisation en pavage triangulaire.

### Description du protocole LDCC

La diffusion par couche est utilisée pour décrire le processus d'échange de messages dans lequel des messages de type particulier sont sensés être envoyés par la station de base ou un noeud proche de la station de base. Ces messages sont diffusés par des noeuds, situés dans l'anneau le plus proche de la station de base, vers les noeuds se trouvant dans un anneau plus loin (d'où le principe de diffusion par couche). Une autre caractéristique de ce type de diffusion est qu'un noeud avec un nombre de sauts inférieur ignore tous les messages en provenance d'un noeud avec un nombre de sauts supérieur. L'exécution de LDCC est divisée en périodes. Dans chaque période, la station de base initie le processus de contrôle de la couverture en diffusant un message ACTIVE contenant son nombre de sauts  $H_t=0$ , la puissance de transmission  $P_t$  et la durée d'activation  $T_r$ .

Initialement, tous les noeuds sont dans l'état actif et chacun calcule un temps back-off. A l'expiration de ce dernier, le noeud ayant le nombre de sauts minimal se met en mode actif et envoie un message ACTIVE avec son  $H_t$ , son  $P_t$  et son  $T_r$ . Après la réception du message ACTIVE, le noeud récepteur met à jour son back-off, ou se met à l'état passif selon le  $H_t$  inclut dans le message ACTIVE. La durée d'activation de l'émetteur, transportée par le message ACTIVE, permet au récepteur de savoir pour combien de temps il reste passif.

### Discussion et critiques

Les résultats de simulation ont montré la maintenance d'un taux de couverture élevé. De plus, la diffusion par couche permet de réduire le nombre de messages de contrôle. En effet, les messages provenant des couches supérieures sont ignorés par les noeuds des couches inférieures et les noeuds n'ont pas à envoyer des messages contenant leurs positions géographiques, ainsi

les coûts de collisions et de retransmissions sont minimisés avec la réduction du trafic.

L'inconvénient de ce protocole vient du fait que la technique du pavage triangulaire dans une zone d'intérêt circulaire peut ne pas être applicable en pratique, ceci est en raison du manque de précision à cause du déploiement aléatoire. En effet, la distance entre les noeuds est plus souvent différente de  $R_{Com}$ , ce qui ne permet pas la formation de triangles équilatéraux. De plus, une caractéristique du déploiement aléatoire est que le réseau soit dense au centre de la zone d'intérêt et plus on s'éloigne du centre, la densité diminue. Donc les noeuds proches des frontières ont une faible probabilité de recevoir deux messages ACTIVE car ils n'ont pas beaucoup de voisins, ces noeuds restent alors actifs jusqu'à leur défaillance.

## 2.5.2 Protocoles basés sur la connaissance de positions

### 2.5.2.1 Low Coverage First Classifying algorithm : LCFC

J. Guo et al [31] ont proposé l'algorithme LCFC (pour Low Coverage First Classifying algorithm), basé sur un principe classique reposant sur la subdivision de l'ensemble des noeuds déployés en  $K$  sous-ensembles. En effet, le problème de couverture est subdivisé en  $K$  sous problèmes permettant d'assurer un certain degré de couverture dans chaque sous-ensemble de noeuds. Ce problème de subdivision est considéré par les auteurs comme NP-difficile [32]. La proposition du protocole LCFC a été motivée par les conséquences du déploiement aléatoire telles que les collisions et la retransmission lorsque tous les noeuds sont actifs.

#### Description de l'algorithme

LCFC est un algorithme d'ordonnancement d'activité divisant le réseau en sous-ensembles. L'objectif de ce protocole consiste à assurer la couverture complète de chaque sous-ensemble avec un minimum de noeuds actifs, ainsi que la réduction du chevauchement entre les zones de couverture des noeuds impliquant une meilleure exploitation du rayon de captage **RCap**. On suppose que le réseau est formé de  $N$  noeuds déployés aléatoirement, chaque noeud  $i$  ( $1 \leq i \leq N$ ), couvre une surface  $S_i$  et  $S_{sum}$  représente la zone d'intérêt. La variable  $P_{area}$  permet de calculer le rapport entre la zone couverte par tous les noeuds et la zone d'intérêt. Si  $P_{area}=1$  alors une couverture complète est assurée.

$$P_{area} = | (S_1 \cup S_2 \cup \dots \cup S_N) \cap S_{sum} | / S_{sum}. \quad (2.4)$$

Etant donné l'ensemble  $S$  des noeuds déployés dans une zone carrée de côté  $a$  et  $K$  le nombre de sous-ensembles, chaque noeud doit rejoindre un des sous-ensembles notés  $C_1$ ,

$C_2, \dots, C_k$ . Les  $K$  sous-ensembles doivent satisfaire les conditions suivantes :

- $C_1 \cup C_2 \cup \dots \cup C_k = S$ .
  - $C_i \cap C_j = \emptyset$  pour  $i \neq j$  et  $\forall i$ , maximiser  $P_i$ .
- tel que  $1 \leq i \leq k, 1 \leq j \leq k$  et  $P_i$  représente le taux de couverture dans un sous-ensemble  $C_i$ .

Vu la difficulté du calcul direct du taux de couverture à partir de la formule 2.4, les auteurs ont proposé la discrétisation de la zone d'intérêt. En effet, des points discrets dans des intervalles égaux à  $\Delta d$  sont sélectionnés et le taux de couverture de ces points est considéré comme une approximation du taux de couverture réel. La valeur minimale de  $\Delta d$  représente l'exactitude la plus élevée que l'algorithme peut obtenir. Pour ces points discrets, certains sont couverts par plusieurs capteurs tandis que d'autres sont relativement moins couverts, ces derniers sont appelés les points à basse couverture.

Le principe de l'algorithme LCFC est : lors de la décision quel sous-ensemble rejoindre, un noeud choisit l'ensemble pouvant couvrir plus de points à basse couverture. Avec cette approche, le chevauchement inter zones de captage des noeuds est effectivement réduit et la capacité de captage de chaque noeud est complètement exploitée.

Soit  $C_i$  un sous-ensemble tel que  $1 \leq i \leq K$ . Initialement,  $C_i$  est vide. Après l'exécution de l'algorithme, chaque sous-ensemble  $C_i$  enregistre les identifiants des noeuds qui le rejoignent.  $Cover_i$  est une matrice correspondant à  $C_i$  de taille  $a/\Delta d + 1 * a/\Delta d + 1$ . Pour chaque élément  $C_{mn}$  de  $Cover_i$ ,  $1 \leq m, n \leq a/\Delta d + 1$ , est associé un point discret  $x$ , la valeur de  $C_{mn}$  peut être calculée par la formule suivante :

$$C_{mn} = f_p(x) = \begin{cases} \infty & \exists s, s \in C_i \text{ et } \text{dist}(s, x) \leq RCap \\ 1 / |S_{libre}| & \text{sinon} \end{cases} \quad (2.5)$$

où  $S_{libre} = \{s | s \in \{S - C_1 \cup C_2 \cup \dots \cup C_k\} \text{ et } s \text{ satisfait la condition } \text{dist}(s, x) \leq RCap\}$ .

Le gain  $f(s_j, C_i)$  d'appartenance d'un noeud  $s_j$  à un sous-ensemble  $C_i$  est calculé par la formule suivante :

$$f(s_j, C_i) = \sum_{\text{dist}(s, x) \leq RCap} 1/f_p^2(x). \quad (2.6)$$



Cette formule indique l'idée de priorité de basse couverture. Pour un noeud  $s_j$ , s'il peut couvrir plus de points à basse couverture dans un sous-ensemble  $C_i$ , la valeur de  $f(s_j, C_i)$  est maximale par rapport à celles réalisées dans les autres sous-ensembles et la possibilité de rejoindre  $C_i$  est plus élevée donc le chevauchement sera considérablement réduit.

En résumé, l'algorithme LCFC a deux phases : dans la phase d'initialisation, tous les sous-ensembles  $C_i$  ( $1 \leq i \leq k$ ) sont initialisés à vide et leurs matrices correspondantes  $Cover_i$  sont initialisées à l'infini. Ensuite, dans la phase de classification, on calcule le gain d'appartenance de chaque noeud  $s_j$  à chaque sous-ensemble  $C_i$ . Après quoi, on compare les résultats et on sélectionne le sous-ensemble  $C_m$  réalisant le gain le plus élevé pour le noeud  $s_j$ . Ce dernier sera alors affecté à  $C_m$  et retiré de l'ensemble des noeuds  $S_{libre}$  non appartenant à un sous-ensemble et la matrice  $Cover_m$  sera mise à jour.

## Discussion et critiques

Cet algorithme réduit la redondance inter zones de captage des différents noeuds, ce qui conduit à la réduction de la redondance d'informations capturées. L'inconvénient que nous pouvons citer pour LCFC est qu'il ne considère aucune contrainte sur la connectivité des sous-ensembles, ainsi, les résultats de la simulation montrent que le taux de couverture est amélioré avec l'augmentation de  $K$  (nombre de sous-ensembles), ce qui donne une complexité de plus en plus élevée car le coût de division d'un problème en sous-problèmes augmente avec la taille de ce dernier.

### 2.5.3 Les protocoles basés sur le clustering

#### 2.5.3.1 Covrage-Aware Clustering Scheme : CACS

Namrata et al [33], ont proposés l'algorithme CACS (pour Covrage-Aware Clustering Scheme) permettant la conservation d'énergie et la préservation de la couverture, il est une extension du protocole protocole LEACH [7] afin d'améliorer les performances de ce dernier. En effet, LEACH est un protocole basé sur le clustering appliqué dans les réseaux de capteurs sans fil, il choisit aléatoirement les cluster-heads selon l'ordonnancement Round-Robin.

L'algorithme LEACH est divisé en deux phases : la phase d'installation (set-up) et la phase d'état stationnaire (steady-state). Dans la première phase, chaque noeud décide de devenir un cluster-head ou non, cette décision est basée sur une valeur seuil  $T(u)$  définie selon la formule suivante :

$$T(u) = p/(1-p)^{(r \bmod 1/p)} \text{ si } u \in G. \quad (2.7)$$

Où  $p$  désigne le pourcentage des noeuds désirant être cluster-head, dans la plupart des cas  $p=5\%$ ,  $r$  est la période courante et  $G$  est l'ensemble des noeuds qui n'étaient pas cluster-head durant les périodes précédentes.

L'élection des cluster-heads se fait comme suit : durant la période  $r$ , un noeud  $u$  choisit un nombre aléatoire entre 0 et 1. Si cette valeur est inférieure à  $T(u)$  alors le noeud  $u$  deviendra cluster-head durant cette période, sinon  $u$  devrait rejoindre le cluster-head le plus proche. Dans la deuxième phase, les données seront transférées vers la station de base. Le problème principal de LEACH réside dans le choix aléatoire des cluster-heads, il ne pose aucune contrainte sur leurs distribution ainsi que sur leurs niveau d'énergie. Donc, les cluster-heads peuvent se concentrer dans un même endroit et, par conséquent, il pourrait exister des noeuds isolés.

### L'algorithme proposé

L'extension CACS utilise une architecture centralisée pour élire les cluster-head. L'élection se fait comme suit :

- Chaque noeud transmet son niveau d'énergie et son degré à la station de base.
- La station de base élit le noeud qui a le niveau d'énergie et le degré les plus élevés.
- La station de base diffuse l'information du cluster-head aux autres noeuds.

Dans le protocole CACS, une phase d'ordonnement des noeuds est intégrée avant la phase d'installation. Au début de chaque période, tous les noeuds décident de se mettre soit à l'état actif soit à l'état passif. Les noeuds en veille ne participeront pas à la phase d'état stationnaire. L'ordonnement des noeuds se fait comme suit : chaque noeud recueille les informations de ses voisins, si ces derniers peuvent le couvrir alors il se met à l'état passif.

### Discussion et critiques

Pour CACS, la même phase d'ordonnement que LEACH est reprise. L'ordonnement donne de meilleures performances, il économise de l'énergie et étend donc la vie du réseau. Toutefois, l'algorithme CACS utilise une architecture centralisée pour l'élection du cluster-head, ce qui augmente considérablement le nombre de messages d'information, puisque pendant chaque période, les capteurs devront envoyer leurs informations à la station de base durant la phase d'installation. En outre, ce protocole engendre un coût de communication élevé pouvant causer des pertes de messages et des collisions.

L'insertion de la phase d'ordonnancement d'activité avant la phase de clusterisation, mène à la mise en veille de noeuds pouvant être élus comme cluster-heads. Dans l'ordonnancement aléatoire, le nombre de noeuds actifs peut être important dans un cluster, ce qui provoque la saturation de la mémoire des clusters-head.

### 2.5.3.2 Coverage-preserving based-Energy Clustering algorithm : CEC

X. Di [8] a proposé l'algorithme CEC ( pour Coverage-preserving based-Energy Clustering algorithm), une amélioration de LEACH [7] tenant compte de l'énergie résiduelle permettant d'équilibrer la consommation d'énergie entre les noeuds. Ce protocole est distribué, les noeuds s'organisent dans des clusters et sont sélectionnés comme cluster-head alternativement, donc la consommation d'énergie peut être équilibrée et la durée de vie du réseau peut être prolongée.

#### Description du protocole CEC

L'apport du protocole CEC par rapport à [33], est la combinaison du taux de redondance avec l'énergie résiduelle pour l'élection des cluster-heads.

L'algorithme CEC se déroule en trois étapes. La première consiste à définir le taux de redondance dans la zone couverte par un noeud N. La deuxième est l'amélioration de l'algorithme d'élection du cluster-head. Enfin, la troisième consiste à proposer une approche pour la formation de clusters. Dans la zone d'intérêt, plusieurs points sont couverts par plus d'un capteur. Pour un noeud N, la somme de ces points est représentée comme une zone de chevauchement de couverture, ainsi, en supposant que le noeud N a n voisins, l'étape 1 permet de calculer le taux de chevauchement de couverture pour N comme suit :

$$C_{redondance}(N) = S_{redondance}(N) / S(N) \quad (2.8)$$

Où  $S(N)$  représente la surface couverte par le noeud N et  $S_{redondance}(N)$  représente la zone redondante du noeud N et elle se calcule comme suit :

$$S_{redondance}(N) = \sum_{i=0}^n S_{redondance}^2 - \sum_{i=0}^n S_{redondance}^3 + \sum_{i=0}^n S_{redondance}^4 - \dots + (-1)^{n+1} \sum_{i=0}^n S_{redondance}^{n+1} \quad (2.9)$$

Où  $S_{redondance}^j(N)$  représente la zone redondante entre le noeud N et son voisin j. Dans la deuxième étape, l'énergie du cluster-head se dégrade plus rapidement que celles des autres noeuds donc un noeud ayant une énergie résiduelle élevée a plus de chances de devenir cluster-head. Aussi, un noeud ayant un taux de chevauchement élevé, a plus de chance de devenir cluster-head car sa défaillance n'a presque pas d'influence sur le taux global de

couverture assuré par le réseau. Par exemple, si un noeud est complètement couvert alors il n'aura aucun impact sur le réseau lorsqu'il est défaillant.

L'auteur a amélioré le seuil d'élection du cluster head comme suit :

$$T(i) = \begin{cases} \infty & \frac{p}{1-p * (r \bmod(1/p))} * p_i^{\text{Energie}} * p_i^{\text{couverture}}, i \in G \\ 0, & \text{sinon} \end{cases} \quad (2.10)$$

Où  $p$  est le pourcentage de cluster-heads,  $r$  est la période,  $p_i^{\text{Energie}}$  et  $p_i^{\text{couverture}}$  sont calculés comme suit :

$$p_i^{\text{Energie}} = E_i/E_j \quad (2.11)$$

Où  $E_i$  représente l'énergie résiduelle du noeud  $i$  et  $E_j$  celle de son voisin  $j$ .

$$p_i^{\text{couverture}} = C_n(i)/C_j \quad (2.12)$$

Où  $C_n(i)$  représente le taux de redondance du noeud  $i$  et  $C_j$  celle de son voisin  $j$ .

Le changement du seuil  $T(i)$  montre que le noeud ayant l'énergie résiduelle et le taux de redondance les plus élevés a plus de chances de devenir cluster-head.

Après la désignation du cluster-head, la troisième étape consiste à affecter les noeuds aux clusters, tel que chaque noeud envoie un message JOIN pour rejoindre un cluster.

## Discussion et critiques

L'avantage du protocole CEC réside dans le fait d'exploiter le taux de chevauchement des noeuds en combinaison avec l'énergie résiduelle pour l'élection du cluster-head. En effet, choisir un noeud redondant comme cluster-head, diminue l'impact de sa défaillance sur le réseau. L'auteur a démontré que le protocole a une complexité linéaire  $O(n)$  en nombre de messages échangés. En revanche, l'inconvénient de ce protocole est qu'il n'adopte pas la phase d'ordonnement d'activité visant à mettre en veille périodiquement les noeuds redondants dans chaque cluster.

### 2.5.3.3 Distributed Energy-Efficient Clustering with Improved Coverage : DEECIC

Z-Liu et al [34] ont proposé DEECIC (pour Distributed Energy-Efficient Clustering with Improved Coverage), un algorithme basé sur le clustering et l'identification unique des noeuds.

Dans ce protocole, l'élection des cluster-heads se fait périodiquement en fonction de l'énergie résiduelle. Les auteurs supposent que tous les noeuds du réseau sont aléatoirement répartis, synchrone et n'utilisent aucun mécanisme de géo-localisation ni de GPS pour la détermination de leurs positions géographiques.

L'objectif de cet algorithme, est de sélectionner un faible nombre de cluster-head de telle sorte que chaque noeud du réseau appartient à un cluster. La communication des noeuds avec le cluster-head se fait au plus à 2-sauts par le biais d'un noeud intermédiaire.

### Description de l'algorithme

Dans le protocole DEECIC, un noeud  $S_u$  est considéré comme un voisin d'un noeud  $S_v$  si est seulement si, le noeud  $S_u$  se trouve à porté radio du noeud  $S_v$ . Les auteurs ont défini la fonction  $\text{NodeDegree}(S_v)$  qui calcule le nombre de voisins de  $S_v$  tel que :

$$\text{NodeDegree}(S_v) = \text{somme}(S_u \mid \text{distance}(S_v, S_u) \leq R_{\text{Com}}, S_u \in S, S_u \neq S_v). \quad (2.13)$$

Quatre types de messages ont été définis dans le processus de clustering :

- **MSG-Degré** : message envoyé par un noeud à l'ensemble de ses voisins pour indiquer son  $\text{NodeDegree}$ . Ce message est envoyé durant la phase de formation des clusters ou la phase de migration des cluster-heads.
- **MSG-Etat** : message envoyé par un noeud à l'ensemble de ses voisins pour indiquer son état et son identifiant. Un noeud dans DEECIC peut être dans l'un des quatre états : cluster-head, noeud membre à 1-saut (voisin du cluster-head), noeud membre à 2-saut (voisin d'un noeud membre à 1-saut) et noeud non regroupé.
- **MSG-Rejoindre** : message envoyé par un noeud non regroupé au cluster-head pour informer ce dernier qu'il veut être un noeud membre à 1-saut ou à 2-sauts.
- **MSG-Renoncé** : message envoyé par un cluster-head pour informer ses membre de son retrait.

L'algorithme DEECIC a deux phases, la phase de formation de clusters et la phase de migration du cluster head.

### Phase de formation de clusters

Initialement, tous les noeuds sont actifs pendant l'exécution de l'algorithme. Après avoir reçu un message d'initialisation, en provenance de la station de base, chaque noeud diffuse un message "hello" à tous ses voisins pour indiquer son existence. Afin de réduire les collisions, la diffusion de ce message se fait après un temps tiré aléatoirement entre 0 et  $T_{\max}$ . Chaque noeud calcule donc son degré noté "NodeDegree" et diffuse "MSG-Degré" entre  $T_{\max}$  et  $T_{\max} + T_{Su}$  tel que :

$$T_{Su} = 1 / \alpha e^{\text{NodeDegree}(Su)} \quad (2.14)$$

Où  $\alpha$  est une constante donnée pour assurer que  $0 < T_{Su} \leq T_{\max}$

Chaque noeud attend l'expiration de son temps d'attente avant de décider d'être cluster-head ou non. Si un noeud a le plus grand degré parmi ses voisins, il se déclare cluster-head en envoyant "MSG-Etat" aux noeuds se trouvant dans son rayon de communication.

Un noeud n'appartenant à aucun cluster et recevant un **MSG-Etat** d'un cluster-head, doit choisir un nombre aléatoire sur deux octets qui sera concaténé à l'identifiant (ID) du CH, le résultat est alors son propre ID. Ensuite, le noeud envoie un **MSG-rejoindre** au CH pour l'informer qu'il est membre à 1-saut. Lorsqu'un noeud non regroupé reçoit un "MSG-Etat" en provenance d'un noeud membre à un saut, il se déclare comme étant un noeud membre à 2-saut, il envoie alors un **MSG-rejoindre**. Les noeuds membres à 2-saut choisissent également leurs propre ID, un nombre aléatoire sur 2 octets, qui sera concaténé à l'identifiant du noeud membre à 1-saut. En raison du déploiement aléatoire des noeuds, il peut y avoir encore quelques noeuds qui n'appartiennent à aucun cluster après l'élection. Le protocole DEECIC permet à ces noeuds de former un cluster, et choisissent un nombre aléatoire sur 4-octets comme identifiant. A la fin de cette phase, chaque noeud aura un identifiant unique lui permettant de désigner son cluster.

### Phase de migration de cluster-heads

Une fois que les clusters soient formés, chaque noeud membre à 1-saut crée une liste et stocke les identifiant de ses membres. Cette liste sera ensuite transmise au cluster-head afin que ce dernier identifie ses membres à deux sauts. Durant cette phase, les noeuds membres à deux sauts collectent les données et les transmettent au voisins membres à un saut, ces derniers collectent, agrègent et transmettent les données aux cluster-heads qui s'occupent de l'agrégation et de l'acheminement des résultats jusqu'à la station de base. Lorsque l'énergie résiduelle d'un CH diminue au dessous d'un seuil, il sera remplacé par l'un de ses membre, un MSG-

Renoncé est alors envoyé à ses membre à 1-saut, ce message contient son ID et son énergie seuil.

Après la réception de "MSG-Renoncé" du cluster-head, chaque noeud membre change son état actuel à "non groupé". Au cour de cette phase les noeuds diffusent "MSG-Etat" ainsi que les messages reçus de leurs voisins. Chaque noeud, durant cette même période, enregistre les messages reçus et les utilise pour compter le nombre de noeuds non groupés autour de lui afin de déterminer un CH pour la prochaine période.

### **Discussion et critiques**

Bienqu'aucune phase d'ordonnancement d'activité ne soit prévue dans ce protocole, les résultats de simulation ont montré qu'il prolonge la durée de vie du réseau et améliore la qualité de la couverture.

Dans le déploiement aléatoire, le nombre de noeuds dans un cluster peut être important, ce qui peut provoquer la saturation de la mémoire des cluster-heads lors de l'envoi ou de la réception des messages.

L'acheminement des données collectées par les noeuds membres à 2-sauts nécessite au préalable un passage par des noeuds membre à 1-saut, ce qui augmente la consommation d'énergie.

#### **2.5.3.4 Coverage-Aware Clustering Protocol**

B. Wang et al [35] ont proposé le protocole CACP (pour Coverage-Aware Clustering Protocol) dont le principe consiste à combiner les deux techniques : ordonnancement d'activité et clustering permettant respectivement de réduire l'énergie consommée et le trafic circulant dans le réseau. Le protocole CACP se base sur la technique du pavage hexagonal pour la formation de clusters et la technique du pavage triangulaire pour l'activation des noeuds.

### **Description du protocole CACP**

Dans cet algorithme, il est supposé que tous les noeuds sont à portée de la station de base et que seulement le cluster-head (CH) fait l'agrégation et la transmission de données en provenance de tous les membres du cluster (CMs).

Le protocole CACP vise à affecter aux noeuds les rôles (actif/passif, cluster-head/cluster-membre). Pour cela, les auteurs ont d'abord proposé une méthode permettant de calculer la taille optimale "H\*" des clusters. Ce problème consiste à rechercher un meilleur placement des noeuds pour trouver la taille optimale du cluster en tenant compte de l'énergie consommée par

unité de zone.

Un cluster est modélisé par un hexagone de coté égal au rayon de communication  $R_{Com} = H * L$ , où  $H = 1, 2, 3 \dots$  et  $L = \sqrt{3} R_{Cap}$  (voir figure 2.6). Tous les clusters forment un pavage hexagonal, le cluster-head a les coordonnées  $CH(0, 0)$  et les membres se situent sur les cotés des hexagons centrés en  $CH$ , ces hexagons sont appelés des h-hexagons tel que  $h=1, 2, 3, \dots H$ . Un h-hexagone contient  $6h$  membres et le nombre de membres dans un cluster est  $3H^2 + H$ , chaque cluster-head a 6 cluster-heads voisins et la distance entre deux cluster-heads voisins est  $\sqrt{3}HL$ .

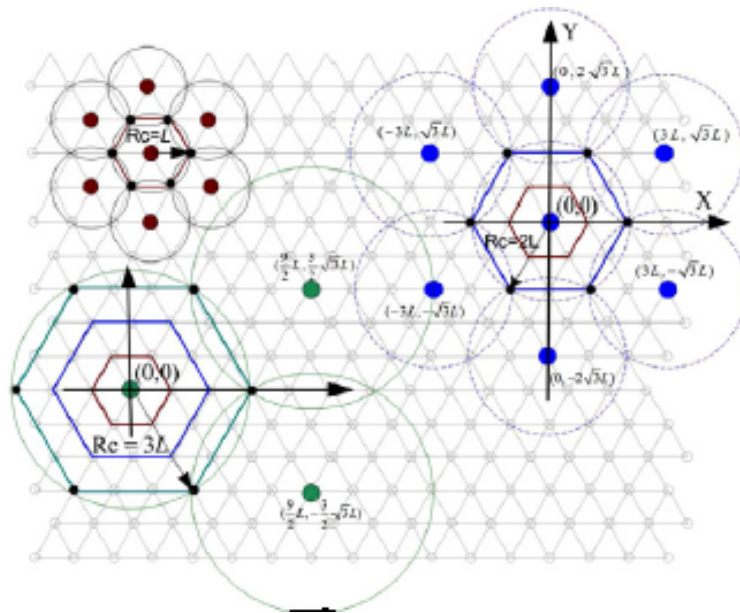


FIGURE 2.6 – Protocole CACP : exemple de clusters avec  $R_{Com} = H * L$  ( $H = 1, 2, 3 \dots$ )

La taille optimale d'un cluster dépend de l'énergie consommée par ses membres qui à son tour dépend de la surface du cluster " $A_h$ " et du trafic total " $B$ " de tous les membres du cluster (CMs). Le cluster-head agrège les  $B$  bits provenant de ses CMs et ses données qu'il a capté, ensuite, il les transmet à la station de base. La consommation d'énergie des CMs dépend des scénarios de routage intra-cluster. En effet, dans le routage à 1-saut, cette consommation dépend de la distance entre un CM et son CH et la taille optimale du cluster est donnée par la formule 5 [35]. En revanche, dans le routage multi-sauts, les informations sont transmises d'un noeud sur un h-hexagone à un noeud sur un (h-1)-hexagone jusqu'à l'arrivée au CH. Cette consommation d'énergie est uniforme car la distance entre les noeuds intermédiaires est la même et la formule 6 [35] permet de calculer la taille du cluster pour ce cas.

### Les phases du protocole CACP



Dans chaque période, le protocole CACP exécute quatre phases. La première consiste en l'échange d'informations de localisation et d'énergie inter voisins de distance  $2 \cdot R_{Cap}$ . La deuxième phase consiste en l'élection du cluster-head où un coût d'élection  $\psi_n$  est calculé pour chaque noeud "n" lui permettant d'évaluer sa contribution dans la préservation de la couverture. Cette métrique combine les informations sur l'énergie restante de l'ensemble couvrant  $C_n$  d'un noeud "n" ainsi que la manière avec laquelle cet ensemble assure la couverture. Pour le calcul de  $\psi_n$ , un algorithme de construction d'un ensemble couvrants  $C(n) \subseteq S(n)$  est proposé, tel que  $S(n)$  est l'ensemble des noeuds ayant leurs zones de perception en chevauchement avec celle du noeud "n". L'idée de cet algorithme consiste à sélectionner en fur et à mesure le noeud qui couvre la plus grande zone non couverte de n. En effet, plus d'un candidat peuvent réaliser une couverture complète avec ceux déjà choisis dans C, mais on sélectionne celui ayant une redondance minimale avec les noeuds déjà appartenant à C. Enfin, en utilisant l'énergie résiduelle du noeud n  $E_R$  et celle de ses noeuds couvrants  $E_{RC}$ , le coût d'élection est calculé comme suit :

$$\psi_n = 1/E_R + \sum_{j=1}^n E_{RC}(C_j) \quad (2.15)$$

Au début de chaque phase d'élection, tout noeud initialise un temps d'attente calculé à base de  $\psi_n$ . Chaque noeud attend l'expiration de son temps d'attente. Si pendant ce temps, le noeud n'entend aucun message d'annonce provenant d'un autre noeud alors il se déclare cluster-head en diffusant un message "HEAD" vers tous les autres noeuds situés sur le rayon de l'héxagone du cluster. Un noeud recevant un message "HEAD", se retire de l'élection et ajoute le CH, sa position ainsi que son énergie à la table des CHs.

La troisième phase concerne la formation des clusters. Après l'élection, les noeuds non-regroupés choisissent un CH dans leurs table de CHs et chaque noeud envoie donc un message JOIN informant le CH qu'il sera membre de ce cluster à la période courante.

La quatrième phase consiste à décider de l'activation des membres de clusters. Pour cela, un algorithme d'activation par couches est proposé. Dans chaque cluster, le CH calcule quelques positions cibles selon la rotation dans le pavage triangulaire (voir la figure 2.7). L'idée de l'algorithme consiste à mettre d'abord en état actif les noeuds les plus proches des positions cibles dans le premier tier du pavage, puis dans le deuxième et ainsi de suite. Chaque CM initialise un temps d'attente dont la durée est liée à sa distance avec la position cible.

Au début de toute phase d'activation, chaque CH diffuse un message "ACTIVE" contenant une information sur la position cible sous forme  $(h(x,y))$  où h est la position cible sur le h-héxagone. A la réception d'un message ACTIVE, le noeud calcule une distance  $d_{min}$  à une de ses positions cibles. Chaque noeud attend l'expiration de son temps d'attente.

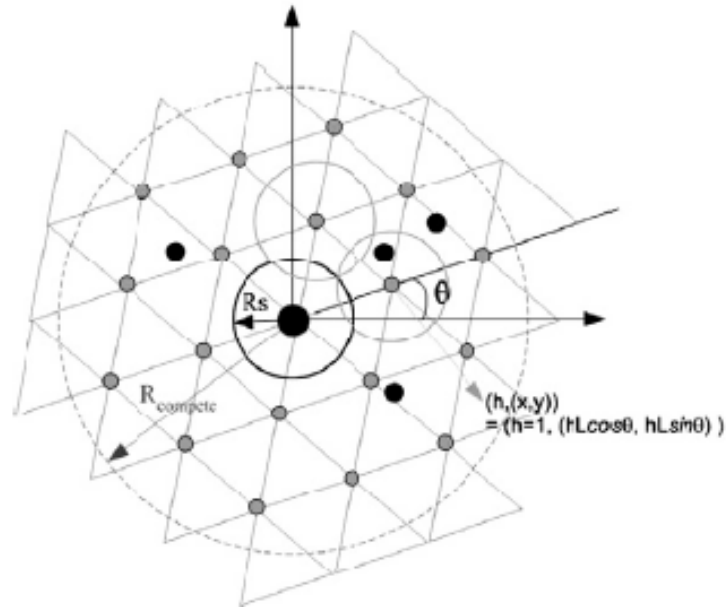


FIGURE 2.7 – Protocole CACP : Illustration des positions cibles

Si après l'expiration de ce temps, un noeud trouve que le taux de redondance de sa zone est inférieur à un certain seuil alors il se déclare comme actif en diffusant un message "ACTIVE" à ses voisins. Ce message contient la position de l'émetteur et la position de sa cible la plus étroite.

Si avant l'expiration de son temps d'attente, un CM "n1" reçoit un message ACTIVE d'un autre CM "n2" alors il ajoute l'information de "n2" à son ensemble de voisins actifs et calcule le taux de redondance dans sa propre zone. Si ce taux est supérieur à un certain seuil, "n1" se met alors à l'état passif dans la période courante.

Pour la communication intra-cluster, le routage multi-sauts est adapté dans le protocole CACP. En effet, un algorithme du plus court chemin, basé sur le nombre de sauts "h", est mis en place. Si  $h=1$  alors le CM envoie directement ses données au CH, sinon il diffuse un message "ROUTE" pour déterminer ses relais.

### Discussion et critiques

La combinaison des techniques de clustering et d'ordonnement d'activité est une solution très puissante pour une meilleure préservation de la couverture. L'activation périodique des membres basée sur le taux de redondance permet ainsi de réduire la consommation et la perte d'énergie.

Néanmoins, un placement idéal dans le cas du déploiement aléatoire n'est pas approprié. L'application des techniques du pavage triangulaire ou hexagonal restent donc restreintes. L'hypothèse d'avoir tous les noeuds à portée de la station de base (chaque noeud se trouve à 1-saut de la station de base) permet d'avoir un réseau dense autour de cette dernière, et cela ne permet pas d'élargir la zone d'intérêt dans le cas où on dispose d'une seule station de base. En plus, le nombre de messages diffusés pour l'élection des cluster-heads, l'activation des membres du cluster et la recherche de relais dans le routage intra-cluster multi-sauts reste important pour le passage à l'échelle. En plus, si le nombre de sponsors d'un noeud est réduit et leur énergie résiduelle est élevée alors, selon le coût d'élection  $\psi$ , il est candidat de devenir CH. Ainsi, le nombre de membres de ce cluster-head et le volume de données à agréger sont réduits.

### 2.5.3.5 Novel hierarchical routing protocolin : HRP

Dans [36], les auteurs ont d'abord décrit une phase de formation de clusters. En effet, chaque noeud diffuse un message "Cf\_Msg" sur le rayon du cluster. Tout noeud recevant "Cf\_Msg" met à jour sa table de voisinage et calcule un taux d'élection (voir formule 1 [36]) dépendant de son énergie résiduelle, des distances les séparant de ses voisins et de la distance le séparant de la station de base. Le noeud réalisant le taux d'élection le plus élevé devient cluster-head pour les membres du cluster formant un réseau complet (voir la figure ci-après).

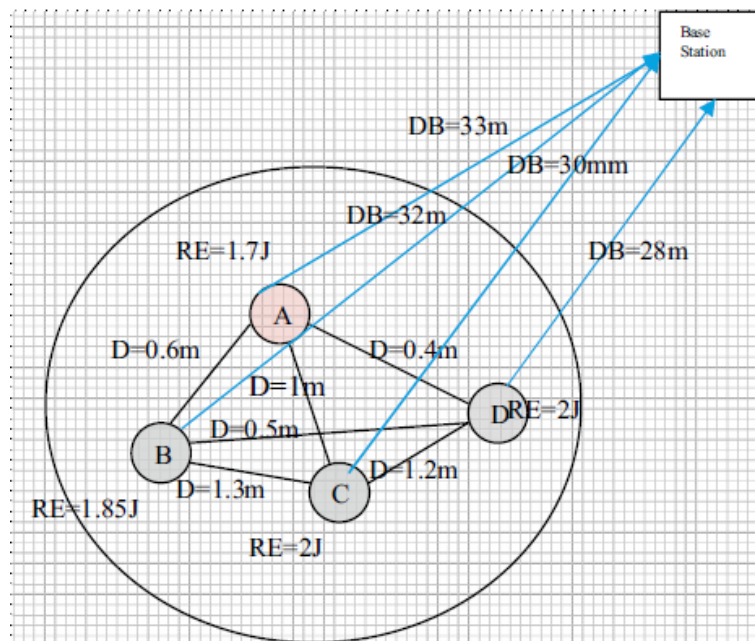


FIGURE 2.8 – Protocole HRP : le processus d'élection du cluster head

Comme il est montré sur la figure 2.8, le cluster-head dans le protocole HRP (pour Hierarchical Routing Protocol) est atteint par un saut depuis tous les membres du cluster.

Après la formation des clusters, les auteurs ont décrit une méthode pour assurer la couverture intra-cluster. Cette méthode consiste à faire un ordonnancement d'activité basé sur l'énergie résiduelle des noeuds. En effet, les noeuds sont classés selon leur niveau d'énergie, dans chaque période, et un sous-ensemble de noeuds ayant la charge la plus élevée est sélectionné pour assurer la couverture du cluster durant cette période. Les noeuds ayant une autonomie faible sont mis en veille (voir figure 2.9).

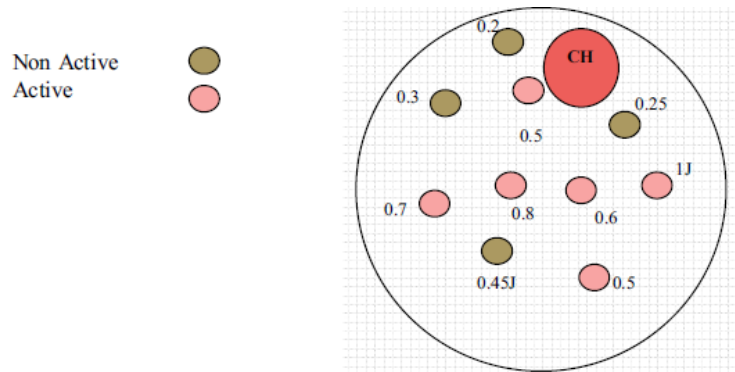


FIGURE 2.9 – Protocole HRP : la couverture intra-cluster.

## Discussion et critiques

Le fait de choisir comme cluster-heads les noeuds les plus proches de l'ensemble de leurs voisins et de la station de base permet de réduire la consommation d'énergie en transmission ainsi que les interférences dans le canal sans fil. Aussi, sélectionner un ensemble de noeuds à énergie élevée comme actif permet d'étendre la longévité du réseau.

Mais le réseau autour d'un cluster ne doit pas être forcément complet et le routage intra-cluster doit être multi-sauts. Ainsi, si les noeuds ayant l'énergie la plus élevée se trouvent tous condensés dans la même région (ils ne sont pas dispersés) et s'ils seront sélectionnés comme actifs pour une période, seulement cette région sera couverte et le reste du cluster devient sans surveillance donc la couverture complète intra-cluster ne peut pas être réalisée.

### 2.5.3.6 Energy-aware Clustering Sensor Scheduling Coverage Maintenance : ECSSCoM

Khanouche et al [37] ont proposé un protocole synchrone nommé ECSSCoM (pour Energy-aware Clustering Sensor Scheduling Coverage Maintenance) dont le principe est tiré du protocole DEECIC [34]. L'amélioration apportée par ECSSCoM consiste en l'intégration de la phase d'ordonnancement d'activité après le clustering afin de réduire la redondance dans la zone d'intérêt. Ce protocole se base sur le calcul d'un temps d'attente en fonction du pourcentage de l'énergie résiduelle et du degré pour l'élection des cluster-heads. Après cette phase, les noeuds restants sont ordonnancés pour assurer la couverture complète de la zone d'intérêt.

#### Description du protocole

Dans cette solution, les noeuds sont supposés synchrones, homogènes et déployés en forte densité. ECSSCoM opère en périodes de longueurs égales, chacune a deux phases : une phase de formation de clusters et d'ordonnancement des membres, et une phase d'activité (capture, agrégation et transmission).

Lors de l'initialisation du réseau, une phase de découverte de voisins est exécutée. Pendant cette phase, chaque noeud  $C_i$  diffuse un message vide "hello" afin de signaler son existence et se met en écoute des messages diffusés par ses voisins directs. Le degré d'un noeud est représenté par le nombre de messages "hello" reçus. Ce degré permet de calculer le temps d'attente dans la phase suivante.

La phase de formation de clusters et l'ordonnancement des membres consiste d'abord à partitionner le réseau en clusters gérés par des cluster-heads, ensuite, ordonnancer l'activité des membres de ces clusters. Avant de prendre la décision d'être cluster-head, membre actif ou passif, chaque noeud calcule un temps d'attente inversement proportionnel à son énergie résiduelle et son degré.

$$\frac{1}{REP^2 * NodeDegree_i} + S_i \quad (2.16)$$

Durant ce temps, le noeud écoute les messages en provenance de ses voisins. Si ce temps expire sans que le noeud reçoive un message **head** d'un de ses voisins dont le temps a expiré en premier, il décide de devenir cluster-head et envoie un message **head** contenant son identifiant.

A la réception d'un message "head", le noeud change la nature de décision en membre actif ou passif. En effet, lors de l'expiration du temps d'attente d'un noeud ayant reçu un

message "head", sans que ce noeud reçoive de messages "join", il décide de devenir membre actif à un saut et envoie un message d'activité "**join**". Un noeud recevant le message "join" déduit qu'il est membre à deux sauts et il décide de se mettre en mode passif lui permettant la conservation de son énergie.

Dans la phase d'activité, les cluster-heads sont responsables de la surveillance de la zone, l'agrégation des données et la transmission du résultat à la station de base, tandis que les membres actifs réalisent le captage des informations et leur transmission aux cluster-heads.

A la fin de la période, les cluster-heads envoient à leurs voisins l'énergie seuil  $E_{\text{SeuilCH}}$ , qui est l'énergie moyenne consommée durant la phase d'activité. Chaque noeud actif recevant ce message, inclut son énergie moyenne consommée durant son activité  $E_{\text{SeuilACT}}$ , et l'envoie à son voisinage direct.

## Discussion et critiques

Le protocole ECSSCoM se base sur un principe simple qui est le temps d'attente permettant à chaque noeud de prendre sa décision. Ce protocole combine la technique de clustering avec celle d'ordonnancement d'activité afin de réduire la redondance tout en gardant un taux élevé de couverture.

Par ailleurs, le coût de communication est élevé vu la nécessité de diffusion de l'énergie consommée par tous les noeuds à la fin de chaque période. La mise en veille de tous les membres à 2-sauts ne garantit pas que toutes les zones surveillées par ces noeuds sont complètement couvertes.

### 2.5.3.7 Coverage-Guaranteed Nodes Scheduling Algorithm in Clusters : CGNSC

Xu BAO et Yongfeng JU [38] ont proposé le protocole CGNSC (pour Coverage-Guaranteed Nodes Scheduling algorithm in Clusters for Wireless Sensor Networks) procédant d'abord à la formation des clusters, puis ces noeuds dans chaque cluster testent s'ils sont redondant. Selon deux conditions. La première consiste à avoir trois voisins telle que la distance entre le noeud et ses trois voisins est inférieure au rayon de captage. La deuxième condition est qu'il soit couvert par des voisins satisfaisant la première condition. Si ces deux conditions sont satisfaites alors le noeud conclut qu'il est redondant et passe à l'état inactif immédiatement.

## Description du protocole CGNSC

Ce protocole est basé sur la connaissance de positions et tous les noeuds sont actifs au

début de chaque période. Pour l'élection des cluster-heads, les noeuds qui ont été passifs dans la période précédente ont plus de chance d'être élus. Dans le processus d'élection, chaque noeud calcule un seuil  $T(n)$  selon la formule suivante :

$$T(u) \bar{p} / (1-p)^{(r \bmod 1/p)} \text{ si } u \in G. \quad (2.17)$$

où  $G$  est l'ensemble des noeuds qui n'ont pas été cluster-heads dans les  $1/P$  périodes précédentes,  $P$  est le pourcentage des CHs et  $r$  est la période courante.

Ensuite, un nombre aléatoire est tiré dans l'intervalle  $[0, 1]$ . Si ce nombre est inférieur au seuil calculé alors le noeud devient cluster-head et diffuse un message "HEAD". Chaque noeud recevant le message HEAD, calcule la distance "d1" par rapport au cluster-head le plus proche et la distance "d2" par rapport à la station de base. Si  $d1 < d2$  alors le noeud rejoint le cluster head le plus proche, sinon il communique directement avec la station de base. Après la formation de clusters, chaque noeud examine sa redondance et prend la décision comme suit :

- Tester s'il y a des noeuds déployés au même point. Si oui, il se met directement en mode passif.
- Tester s'il a des voisins satisfaisant les conditions citées auparavant. Si oui, il se met directement à l'état passif.

Chaque noeud "O" examine la couverture de sa zone modélisée en cercle comme suit :

- Chaque cercle  $S_i$  ( centré en noeud  $S_i$  tel que  $i = 1, 2, 3, \dots, m$ ) voisin du cercle de "O" tel que  $d(S_i, O) < R_{Cap}$ , calcule les coordonnées de tous points d'intersection avec les cercles voisins.
- Si  $m$  est supérieur ou égal à 3 voisins alors, pour chaque cercle voisin  $S_i$ , calculer l'angle formé entre chaque point d'intersection et l'axe des X.
- Les angles formés sont placés selon l'ordre ascendant dans une matrice "M" de  $m$  lignes ( $m$  voisins respectant  $d(\text{voisin}, O) < R_{Cap}$ ) et deux colonnes (le nombre de points d'intersection entre chaque deux cercles est égal à 2).
- Si  $M(i, 2) \geq M(i+1, 1)$  et  $M(m, 2) - M(1, 1) \geq 2\pi$ , le cercle de O est considéré complètement couvert.

## Discussion et critiques

Le protocole CGNSC est basé sur la connaissance des positions permettant une évaluation exacte de la redondance. Mais l'absence de désynchronisation dans la prise de décision peut engendrer une mise en veille simultanée de deux noeuds se couvrant mutuellement et ainsi des

trous de couverture.

## 2.6 Comparaison des protocoles de couverture étudiés

Afin de mieux comprendre la diversité des solutions étudiées dans ce chapitre traitant le problème de la couverture, nous proposons une comparaison à base des critères définis auparavant (mode d'exécution et type d'information de contrôle) et d'autres critères tels que :

- **La garantie de la couverture** : représente la qualité de couverture assurée par le protocole dans la zone cible. Cette caractéristique est primordiale pour déterminer l'efficacité de la solution proposée.
- **La garantie de la connectivité** : la connectivité consiste à assurer que tout noeud peut atteindre la station de base via un ensemble intermédiaire de noeuds formant un chemin connexe pour l'acheminement des données captées. Un protocole de couverture est considéré plus performant en assurant la connectivité du réseau.
- **Le coût de communication** : représente la complexité du protocole en nombre de messages diffusés pour la découverte du voisinage, la clusterisation et l'activation des noeuds.



| Protocole                           | Critère | Mode d'exécution           | Information de contrôle   | Couverture        | Connectivité | Coût de communication |
|-------------------------------------|---------|----------------------------|---------------------------|-------------------|--------------|-----------------------|
| ECCRA, Jin et al, 2008 [25]         |         | Distribué                  | Nombre de saut            | Moyenne           | Oui          | Elevé                 |
| LDCC, Wang et al, 2009 [30]         |         | Distribué                  | Nombre de sauts           | Moyenne           | Oui          | faible                |
| LCFC, Jian et al, 2011 [31]         |         | Distirbué                  | Connaissance de positions | Bonne             | non traitée  | faible                |
| CACS, Namrata et al, 2011 [33]      |         | Centralisé et hiérarchique | Aucune                    | Bonne             | Oui          | Moyen                 |
| DEECIC, Liu et al, 2011 [34]        |         | Distribué et hiérarchique  | Aucune                    | Moyennement bonne | non traitée  | Moyen                 |
| CACP, Wang et al, 2012 [35]         |         | Distribué et hiérarchique  | Connaissance de positions | Très bonne        | Oui          | élevé                 |
| HRP, Zeynali et al 2010 [36]        |         | Distribué et hiérarchique  | Estimation de la distance | Faible            | Oui          | Moyen                 |
| ECSSCoM, Khanouche et al, 2012 [37] |         | Distribué et hiérarchique  | Aucune                    | Bonne             | non traitée  | élevé                 |
| CGNSC, Bao et al, 2012 [38]         |         | Distribué et hiérarchique  | connaissance de positions | Très bonne        | Oui          | faible                |
| CEC, Xin Di, 2012 [8]               |         | Distribué et hiérarchique  | connaissance de positions | Très bonne        | non traitée  | moyen                 |

TABLE 2.1 – Comparaison des protocoles de couverture étudiés.

## 2.7 Conclusion

Diverses recherches ont été menées sur l'axe de la couverture dans les réseaux de capteurs sans fil en utilisant plusieurs stratégies. La stratégie la plus adaptée dans la littérature est celle d'ordonnancement d'activité permettant de réduire la redondance par la mise en veille périodique des capteurs. Dans ce chapitre, notre étude s'est focalisée sur les protocoles d'ordonnancement basés sur le clustering, une technique intégrée récemment dans le problème

de couverture pour une meilleure conservation énergétique.

A base de cette étude critique, nous abordons dans le chapitre suivant, la contribution qui est une approche basée sur le clustering pour la résolution du problème de la couverture.

## CHAPITRE 3

# PROTOCOLE HIÉRARCHIQUE BASÉ SUR L'ÉVALUATION DE LA REDONDANCE POUR LA GARANTIE DE LA COUVERTURE DANS LES RCSF

### 3.1 Introduction

Nous avons montré à travers l'étude des protocoles de couverture que la combinaison des techniques d'ordonnancement d'activité et de clustering permet d'augmenter la longévité du réseau. Cette extension est due, d'une part, à la mise en veille des noeuds redondants et, d'autre part, à l'organisation du réseau en clusters permettant de réduire le trafic, les risques de collisions et la retransmission de données.

Dans ce chapitre, nous proposons une approche distribuée, baptisée HPRC (pour Hierarchical Protocol based-on Redundancy evaluation for Coverage guarantee), basée sur les deux techniques précitées : clustering et ordonnancement d'activité pour garantir la couverture et l'équilibrage des dépenses énergétiques dans les réseaux de capteurs sans fil. Dans sa politique d'élection, le protocole proposé utilise un poids dépendant de l'énergie, la distance moyenne et le degré des noeuds. Quant à l'auto-ordonnancement des membres de clusters, il se base sur l'évaluation de la redondance.

### 3.2 Formulation du problème

Vu l'inaccessibilité des environnements, les réseaux de capteurs sans fil sont fréquemment déployés aléatoirement. Pour éviter les trous de couverture et assurer une surveillance complète de la zone cible, en cas de défaillance de noeuds, un grand nombre d'applications exigent

un réseau fortement dense. Une des conséquences directe du déploiement aléatoire en forte densité, est la redondance des noeuds. Un noeud est dit redondant si sa zone de couverture est complètement surveillée par ses voisins. En effet, un événement peut être capté par plusieurs noeuds voisins à cause de la redondance. Dans un réseau à plat, la perception d'une même donnée par des capteurs voisins et sa transmission à la station de base, provoquent une redondance d'information et une perte inutile d'énergie.

La problématique à traiter dans ce travail est double. Elle consiste, d'une part, à mettre en veille l'ensemble des noeuds redondants et, d'autre part, à organiser les noeuds actifs en clusters chacun doté d'un leader; appelé cluster-head (CH). L'objectif de ce travail consiste donc à combiner la stratégie d'ordonnancement d'activité avec la technique de clustering pour réduire la redondance lors de la collecte des informations et minimiser le volume de données à transmettre à la station de base. Cela permet de conserver la précieuse énergie des capteurs. Les défis à relever par cette combinaison sont divers : (1) la garantie de la couverture complète de la zone cible avec un nombre réduit de noeuds, (2) la réduction de la redondance et du volume de données captées (celà en appliquant l'agrégation au niveau des cluster-heads), ajoutant à ceci (3) l'évitement des collisions et les effets de retransmission dans le canal sans fil.

### 3.3 Propriétés du réseau et hypothèses

Nous considérons un réseau de capteurs sans fil dont les noeuds sont statiques, déployés aléatoirement et uniformément avec une forte densité. Bien que cette densité soit élevée, nous omettons l'hypothèse d'avoir initialement une couverture complète de la zone cible. Nous supposons que les noeuds peuvent avoir connaissance de leurs positions géographiques. Les capteurs formant le réseau ont les mêmes capacités de couverture et de communication ainsi que la quantité d'énergie initiale.

Un noeud a les coordonnées géographiques  $(x, y)$ , une cible située dans le cercle de centre  $(x, y)$  et de rayon **RCap** (rayon de couverture du capteur) peut être détectée par ce capteur [39]. Vu la redondance élevée dans les réseaux de capteurs, une loi de fiabilité est introduite pour évaluer la couverture du réseau. Si une zone est dans le rayon de captage de  $n$  noeuds à l'instant  $t$ , la fiabilité de captage de cette zone est calculée comme suite :

$$R(t) = 1 - \prod_{i=1}^n (1 - r_i(t)) \quad (3.1)$$

Où  $r_i(t)$  est la fiabilité de captage du  $i^{\text{ème}}$  capteur. La zone où la loi de fiabilité satisfait la condition de capture, est définie comme étant la zone de détection effective.

La zone de communication **effective** d'un noeud est délimitée par ses voisins les plus éloignés [25]. Deux noeuds sont dits voisins s'ils se trouvent dans le rayon de communication l'un de l'autre. Le degré d'un noeud est déterminé par le nombre de liens de voisinage dans cette zone.

### 3.4 Protocole hiérarchique basé sur l'évaluation de la redondance pour la garantie de la couverture : HPRC

Le protocole HPRC (pour Hierarchical Protocol based-on Redundancy evaluation for Coverage s'garantee) est une solution distribuée et hiérarchique pour la garantie de la couverture dans les réseaux de capteurs homogènes. Il est basé sur deux techniques : le clustering et l'ordonnement d'activité des noeuds. Le protocole proposé a les caractéristiques suivantes :

- Après le déploiement du réseau, tous les capteurs participent dans une phase de découverte du voisinage.
- Le calcul de la distance entre deux noeuds se base sur les positions géographiques.
- La durée de vie du réseau est divisée en périodes d'activité de longueurs égales.
- Une auto-élection des cluster-heads selon l'énergie, le degré et la distance moyenne par rapport au voisinage. L'auto-ordonnement des membres, quant à lui, tient compte de l'évaluation de la redondance.
- Des dépenses énergétiques équilibrées grâce à l'alternance des rôles (cluster head, membre actif et noeud passif) entre les capteurs.

#### 3.4.1 Principe général du protocole HPRC

Le protocole HPRC opère en périodes d'activité de durées égales. Chaque période se compose de deux phases : une phase de clustering et d'ordonnement des membres, suivie d'une phase de collecte d'informations (capture, agrégation et transmission de données). A ces deux phases, s'ajoute une phase de découverte de voisinage durant laquelle les noeuds s'échangent leurs positions géographiques, et qui est exécutée une seule fois après le déploiement initial du réseau.

Lors de la première phase, une auto-décision d'état est prise par chaque capteur : cluster-head (CH), membre à 1-saut (CM-1S), membre à 2-sauts (CM-2S) ou noeud passif. Les noeuds actifs (cluster-heads et leurs membres) assurent la couverture de la zone cible pendant la phase de collecte d'informations. Pour l'équilibrage de la consommation énergétique, dans chaque période, les noeuds ayant l'énergie et le degré les plus élevés ainsi que la somme des distances la

plus réduite les séparant de leurs voisins, sont élus comme étant cluster-heads. L'ordonnement d'activité des membres se base sur l'évaluation de leur redondance.

### 3.4.2 Description détaillée des phases de HPRC

Après la diffusion des positions lors de la phase de découverte de voisins, chaque période se déroule en deux phases : la phase de formation de clusters et d'ordonnement des membres suivie de la phase de collecte de données. Dans ce qui suit, nous décrivons en détails les différentes phases du protocole HPRC.

#### 3.4.2.1 Découverte de voisins

La phase de découverte de voisinage est exécutée une seule fois lors de l'initialisation du réseau. En effet, après avoir reçu un message d'initialisation en provenance de la station de base, chaque noeud  $\mathbf{i}$  diffuse un message **POS** contenant sa position et se met en écoute des messages envoyés par ses voisins. Ainsi, le degré du capteur  $\mathbf{i}$  (noté  $\text{Deg}_i$ ) est le nombre de messages **POS** reçus. En se basant sur les positions reçues dans ces messages, le noeud  $\mathbf{i}$  calcule la distance  $d(\mathbf{i}, \mathbf{j})$  le séparant de chaque voisin  $\mathbf{j}$ , il pourra alors déterminer la distance moyenne (notée  $\text{Avg\_Dist}_i$ ) le séparant de ses voisins  $V_i$  :

$$\text{Avg\_Dist}_i = \frac{\sum_{j=1}^{\text{Deg}_i} d(\mathbf{i}, \mathbf{j})}{\text{Deg}_i} \quad (3.2)$$

A l'issue de cette phase, tout noeud obtient son degré et la distance moyenne le séparant de ses voisins. Ces deux paramètres permettront le calcul du poids dans la phase suivante.

#### 3.4.2.2 Formation de clusters et ordonnancement des membres

Cette phase se déroule en deux étapes : (1) election des cluster-heads, (2) ordonnancement et désignation des membres.

Lors de la phase d'élection, chaque noeud  $\mathbf{i}$  calcule un poids (noté  $\text{Poids}_i$ ) dépendant de son énergie résiduelle à l'instant  $\mathbf{t}$  (notée  $\text{ER}_i(\mathbf{t})$ ), son degré ( $\text{Deg}_i$ ) et sa distance moyenne ( $\text{Avg\_Dist}_i$ ). L'élection est basée sur le poids pour plusieurs raisons : (1) la consommation énergétique des cluster-heads est plus élevée que celle des membres en raison de l'agrégation des données et leur transmission à la station de base, (2) le choix des capteurs les plus proches de leurs voisins comme étant des cluster-heads, réduit les dépenses énergétiques lors de l'envoi des données par les membres, (3) avoir des cluster-heads ayant le plus de voisins permet

d'augmenter la taille des clusters et ainsi le volume de données agrégées, ce qui minimise la consommation énergétique lors de l'acheminement de ces données.

$$\text{Poids}_i = (\text{ER}_i(t) * \frac{1}{\text{AVG\_Dist}_i}) * \text{Deg}_i \quad (3.3)$$

Le noeud  $i$  diffuse ensuite son poids dans un message **POIDS** ( $i$ ,  $\text{Poids}_i$ ) contenant aussi son identifiant. A partir de l'énergie courante (notée  $\text{ECourante}_i$ ), chaque capteur estime son énergie résiduelle à l'instant  $t$  pour l'utiliser dans le calcul du poids :

$$\text{ER}_i(t) = \text{ECourante}_i - \text{E}_{\text{émission}}(\text{POIDS}) - (\text{Deg}_i * \text{E}_{\text{réception}}(\text{POIDS})) \quad (3.4)$$

A la réception des messages **POIDS** ( $j$ ,  $\text{Poids}_j$ ), le noeud  $i$  compare son poids, s'il possède le poids le plus élevé parmi ses voisins, alors il se déclare cluster-head en diffusant un message **HEAD**( $i$ ).

Pour l'ordonnancement et la désignation des membres, tout noeud  $i$ , ne possédant pas le poids maximal parmi ses voisins, évalue sa couverture en calculant le taux de redondance (noté  $\text{TxRed}_i$ ) de sa zone de captage. La technique d'évaluation de la redondance est détaillée plus loin.

A la réception du premier message **HEAD**( $j$ ), en provenance d'un voisin  $j$ , le noeud  $i$  compare son taux de redondance à un seuil prédéterminé (noté (**RedSeuil**)). Si le capteur  $i$  n'est pas suffisamment couvert, il envoie un message **JOIN-1S**( $j$ ) pour rejoindre le cluster-head  $j$ . Dans le cas contraire, le noeud  $i$  calcule un temps d'attente (noté  $\text{BackOff}_i$ ) inversement proportionnel à son poids.

$$\text{BackOff}_i = \frac{1}{\text{Poids}_i} \quad (3.5)$$

Avant que son temps d'attente expire, lorsque le noeud  $i$  reçoit un message **PASSIF**( $j$ ), en provenance d'un voisin  $j$  ayant décidé d'être passif, il réévalue sa redondance. Si le noeud  $i$  n'est plus couvert, alors il envoie **JOIN-1S**( $i$ ). Dans le cas contraire, le noeud  $i$  attend la fin de son temps pour passer à l'état inactif et envoyer un message **PASSIF**( $i$ ) s'il demeure toujours couvert.

Pour un noeud  $i$  n'ayant reçu aucun message **HEAD**, deux cas se présentent : s'il reçoit un message **PASSIF**( $j$ ) ou **JOIN-2S**( $k$ ) (en provenance d'un voisin  $j$ ) alors il retire le noeud  $j$  de son voisinage et ré-exécute le processus d'élection, s'il reçoit un message **JOIN-1S**( $j$ ) en provenance d'un voisin  $k$  alors il devient membre à 2-sauts et envoie **JOIN-2S**( $k$ ).

A la fin de cette phase, tout noeud se trouve dans l'un des quatre états : cluster-head (CH), membre à 1-saut (CM-1S), membre à 2-sauts (CM-2S) ou membre passif (PM). L'algorithme 1 représente le pseudo-code de la phase 1 du protocole HPRC.

### 3.4.2.3 Collecte d'informations

Dans cette phase, les noeuds ayant décidé d'être cluster-heads, membres à 1-saut ou à 2-sauts assurent la couverture de la zone cible. En effet, les cluster-heads s'occupent du captage, de l'agrégation et de la transmission des données à la station de base. En revanche, les membres à un saut effectuent le captage, l'agrégation de données en provenance des CM-2S et la transmission de ces données aux clusters heads. Les membres à deux sauts, quant à eux, sont seulement responsables du captage des informations et de leur transmission aux CM-1S.

A la fin de cette phase, si l'énergie d'un noeud  $i$  diminue au dessous d'un seuil (noté  $E_{seuil}$ ), il devient défaillant et diffuse un message **FAIL**( $i$ ). Les noeuds recevant ce message, mettent à jour leurs degrés ainsi que leur distance moyenne avant d'entamer une nouvelle période. Il est à noter que  $E_{seuil}$  est soit :  $ESeuil(CH)_i$  pour un cluster head  $i$ , soit  $ESeuil(CM)_j$  pour un membre  $j$ .

### 3.4.3 L'organigramme du protocole HPRC

La figure 3.1 illustre le fonctionnement du protocole proposé sous forme d'organigramme.



**Algorithm 1** Le pseudo-code du protocole HPRC

---

```

1: if ( $ER_i < E_{seuil}$ ) then
2:   Le noeud est défaillant
3:   Envoyer un message FAIL( $i$ )
4: else
5:   if (La réception d'un message FAIL( $j$ )) then
6:     Mise à jour du degré ( $Deg_i$ )
7:     Mise à jour de la distance moyenne ( $Avg\_Dist_i$ )
8:   end if
9:    $Poids_i = (ER_i / Avg\_Dist_i) * Deg_i$ 
10:  Envoyer un message POIDS ( $Poids_i, i$ )
11:  if ( $Poids_i = \text{Max}(Poids_j), \forall j \in V_i$ ) then
12:    Devenir Cluster – head
13:    Envoyer un message HEAD ( $i$ )
14:  else
15:    Calculer le taux de redondance ( $TxRed_i$ )
16:    if (HEAD( $j$ ) reçu) then
17:      if ( $TxRed_i \geq Red_{seuil}$ ) then
18:         $BackOff_i = 1 / Poids_i$ 
19:        while ( $BackOff_i \neq 0$ ) et ( $TxRed_i \geq Red_{seuil}$ ) do
20:          Lors de La réception d'un message( PASSIF( $k$ )) faire
21:          Ré – évaluer  $TxRed_i$ 
22:          Fin faire
23:        end while
24:        if ( $TxRed_i < Red_{seuil}$ ) then
25:          Envoyer un message JOIN – 1S ( $j$ )
26:        else
27:          Envoyer un message PASSIF ( $i$ )
28:          Devenir passif
29:        end if
30:      else
31:        Envoyer un message JOIN – 1S ( $j$ )
32:      end if
33:    else
34:      if ((PASSIF( $j$ ) reçu) || (JOIN – 2S ( $k$ ) reçu)) then
35:        Aller à 11
36:      else
37:        Envoyer un message JOIN – 2S ( $j$ )
38:      end if
39:    end if
40:  end if
41: end if

```

---

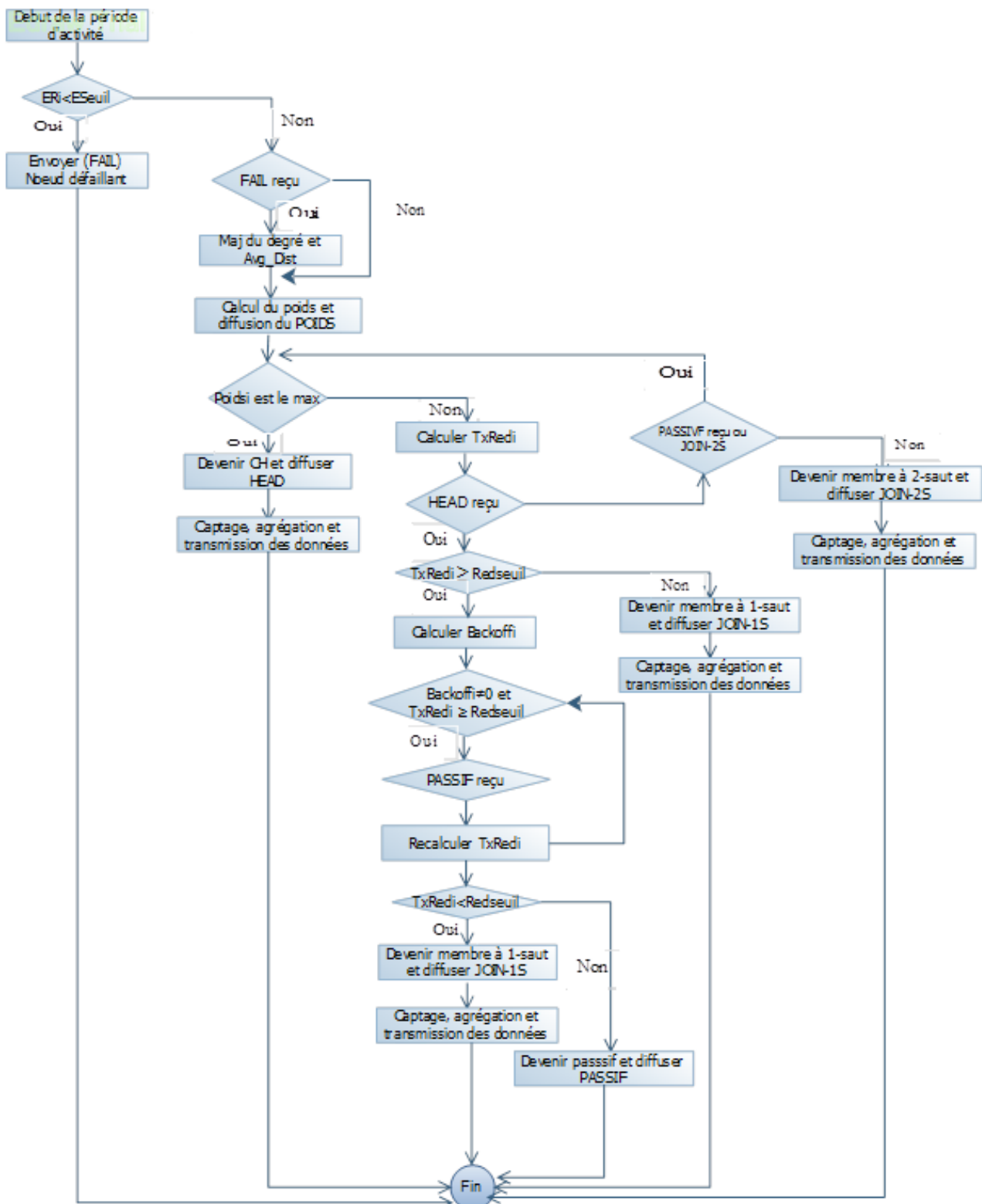


FIGURE 3.1 – Organigramme du protocole HPRC.

### 3.4.4 Technique d'évaluation de la redondance

Une des techniques largement utilisées dans la littérature (par exemple [31],[35],[38],[40]) pour l'évaluation de la redondance, consiste en la division de la zone d'intérêt en points discrets. Cette technique permet d'évaluer la couverture d'un point donné en comparant la distance le séparant d'un capteur par rapport au rayon de captage. Si cette distance est au dessous du rayon de captage d'au moins deux capteurs alors ce point est dit redondant. Le taux de couverture de tous les points de la zone est considéré comme une approximation du taux de couverture réel.

Afin d'assurer une surveillance quasi-totale de la zone cible, la politique de mise en veille des capteurs doit tenir compte du taux de redondance de ces derniers. Pour évaluer la redondance des noeuds dans le protocole HPRC, nous avons opté pour la division de la zone de couverture de chaque capteur en cellules de  $(\mathbf{RCap} / 20 * \mathbf{RCap} / 20) \text{ m}^2$ . Une cellule de la zone de perception d'un capteur  $\mathbf{i}$  est dite couverte par un voisin  $\mathbf{j}$ , si son centre est couvert. Le centre d'une cellule de la zone de couverture du capteur  $\mathbf{i}$  est couvert par un voisin  $\mathbf{j}$ , si la distance le séparant de  $\mathbf{j}$  est inférieure au rayon de captage. Le taux de redondance  $\text{TxRed}_i$  est alors le rapport entre le nombre de cellules couvertes (par  $V_i$  : les voisins de  $\mathbf{i}$ ) et le nombre total de cellules.

$$\text{TxRed}_i = \frac{\text{Nombre de cellules couvertes par } V_i}{\text{Nombre total de cellules}} \quad (3.6)$$

### 3.4.5 Modèle énergétique

Pour le protocole HPRC, nous adaptons le modèle énergétique utilisé dans DEECIC [34]. Ce modèle est basé principalement sur celui proposé par Heinzelman et al [7]. En effet, l'énergie consommée par un membre à 2-sauts  $\mathbf{j}$  dépend de la distance le séparant du membre à 1-saut  $\mathbf{i}$  auquel il est associé :

$$E(\text{CM-2S}_j) = 1 * E_{\text{elec}} + 1 * \epsilon_{fs} * \text{dist}^2(\text{CM-2S}_j, \text{CM} - 1\text{S}_i) \quad (3.7)$$

L'énergie consommée par un membre à 1-saut  $\mathbf{i}$  est calculée en fonction du nombre de ses membres à 2-sauts :

$$E(\text{CM} - 1\text{S}_i) = 1 * E_{\text{elec}} * \text{NBR}(\text{CM-2S}(\text{CM} - 1\text{S}_i)) + 1 * E_{\text{DA}} * (\text{NBR}(\text{CM-2S}(\text{CM} - 1\text{S}_i)) + 1) + 1 * (E_{\text{elec}} + \epsilon_{fs} * \text{dist}^2((\text{CM-1S})_i, \text{CH}_k)). \quad (3.8)$$

L'énergie consommée par le cluster-head  $\mathbf{k}$  dépend du nombre de ses membres à 1-saut :

$$E(\text{CH})_k = l * E_{elec} * \text{NBR}(\text{CM-1S}(\text{CH}_k)) + l * E_{DA} * (\text{NBR}(\text{CM-1S}(\text{CH}_k)) + 1) + l * (E_{elec} + \epsilon_{fs} * \text{dist}^4(\text{CH}_k, \text{SB})). \quad (3.9)$$

Selon les formules (3.7)-(3.10), l'énergie consommée par un cluster-head est plus élevée que celle consommée par ses membres. En supposant qu'un noeud agit continuellement comme cluster-head, dans ce cas, l'énergie seuil est calculée comme suit :

$$E_{\text{Seuil}}((\text{CH})_k) = E(\text{CH})_k * f \quad (3.10)$$

Les membre du cluster compare leur énergie résiduelle à un seuil nommé  $E_{\text{Seuil}}(\text{CM})$  correspondant à l'énergie moyenne consommée pendant les périodes précédentes.

$$E_{\text{Seuil}}(\text{CM}) = (\sum_{i=1}^{n-1} E_{\text{periode}i} + E_{\text{periode}n}) / n. \quad (3.11)$$

Où :

- $E_{elec}$  représente l'énergie électronique du transmetteur ;
- $\epsilon_{mp}$  et  $\epsilon_{fs}$  représentent les énergies d'amplification ;
- $E_{DA}$  représente l'énergie d'agrégation consommée par un CH ou un M-1S ;
- $f$  représente le nombre de trames transmises par un cluster-head dans une période ;
- $l$  représente la taille du paquet de données ;
- $d(i, j)$  représente la distance en mètres séparant le capteur  $i$  du voisin  $j$ .
- $E_{\text{periode}i}$  représente l'énergie consommée pendant la période "i" pour un M-1S ou un M-2S.

### 3.5 Scénario illustratif du protocole HPRC

Pour illustrer le fonctionnement du protocole proposé, nous considérons une zone d'intérêt totalement couverte par vingt capteurs déployés aléatoirement (figure 3.2). Initialement tous les capteurs ont le même niveau d'énergie et possèdent des rayons de couverture et de communication semblables. La portée de transmission d'un capteur étant supérieure à son rayon de captage.

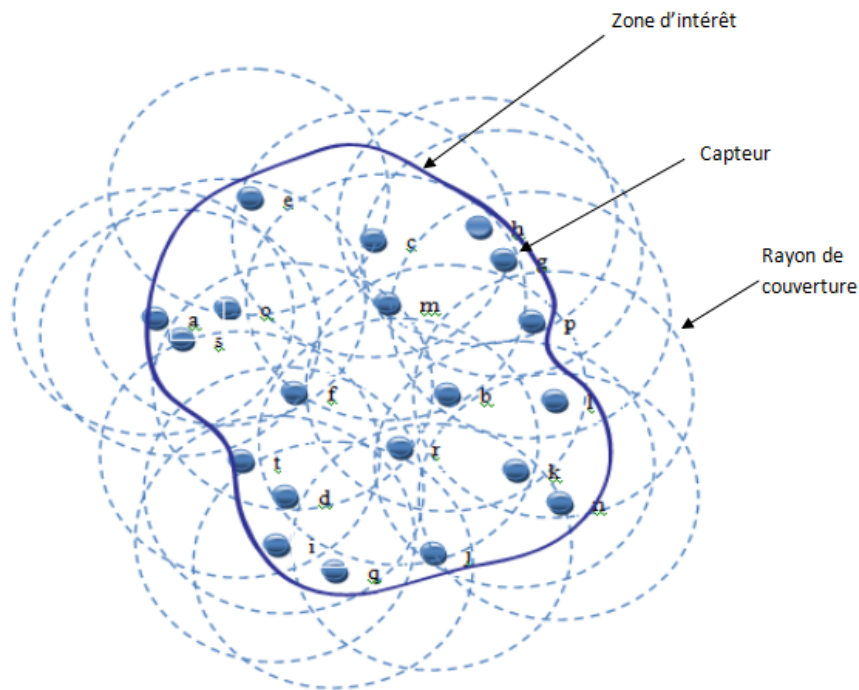


FIGURE 3.2 – Scénario illustratif du protocole HPRC

#### Phase de découverte de voisins

Cette phase permet à tout capteur de connaître son voisinage à 1-saut et la distance par rapport à ses voisins. Il pourra alors calculer son degré et la distance moyenne le séparant de ses voisins (formule 3.1).

Dans ce scénario, nous considérons que l'énergie initiale des capteurs est de 5 joules, l'émission d'un message coûte 0.2 joule et sa réception coûte 0.1 joule. Les résultats de la phase de découverte de voisins sont décrits dans le tableau suivant :

| Noeuds | Deg <sub>i</sub> | Voisins et distance   | Avg_Dist |
|--------|------------------|---|----------|
| a      | 5                | s : 5, o : 10, e :23, t : 25, f : 22  | 17       |
| b      | 10               | c :24, f :25, j :28, k :18, l :13, m :16, n :23, p :12, r :18               | 19.7     |
| c      | 8                | b :24, e :15, f :25, g :15, h :16, o :25, p :27                             | 19.75    |
| d      | 6                | f :17, i :8, j :22, q :19, r :17, t :9                                      | 15.33    |
| e      | 7                | a :19, c :15, f :21, h :28, m :22, o :19, s :22                             | 22.28    |
| f      | 11               | a :22, b :25, c :25, d :17, e :26, i :25, m :28, o :17, r :21, s :18, t :12 | 20.54    |
| g      | 6                | b :21, c :16, h :9, l :28, m :15, p :15                                     | 16.66    |
| h      | 5                | c :13, e :28, g :9, m :22, p :25  | 19.4     |
| i      | 6                | d :8, f :25, j :23, q :8, r :24, t :14                                      | 17       |
| j      | 8                | b :28, d :22, i :22, k :16, l :27, n :25, q :16, r :14                      | 18.5     |
| k      | 6                | b :17, j :16, l :10, n :7, p:23, r :16                                      | 14.83    |
| l      | 7                | b:13, g :28, j :27, k :10, n :11, p :13, r :22                              | 17.71    |
| m      | 9                | b :16, c :10, e :22, f :18, g :18, h :22, o :24, p :24, r :25               | 19.88    |
| n      | 6                | b :23, j :23, k :16, l :11, p :26, r :25                                    | 20.66    |
| o      | 7                | a:10, c :23, e :19, f:17, m :24, s :7, t :24                                | 17.71    |
| p      | 8                | b :12, c :27, g :15, h :25, k :12, l :13, m :24, n :26                      | 19       |
| q      | 5                | d :19, i :8, j :15, r :12, t :8   | 14       |
| r      | 11               | b :18, d :17, f :25, i :24, j :14, k :16, l :22, m :25, n :25, q :21, t :24 | 20.63    |
| s      | 5                | a :5, e :22, f :18, o :7, t :21   | 14.6     |
| t      | 8                | a :25, d :8, f :12, i :14, o :24, q :8, r :24, s :21                        | 17       |

TABLE 3.1 – Scénario illustratif du protocole HPRC : résultats de la phase de découverte.

### Première période d'activité

Dans cette phase, tout capteur calcule son poids (formule 3.2). Le noeud ayant le poids le plus élevé parmi ses voisins sera élu cluster-head. L'exécution du protocole HPRC, permet l'obtention des résultats illustrés dans le tableau ci-dessous :

| Noeuds | Deg <sub>i</sub> | ER <sub>i</sub> | Poids <sub>i</sub> | T <sub>x</sub> Red ≥ Red <sub>Seuil</sub> | BackOff | Décision | Etat   |
|--------|------------------|-----------------|--------------------|---|---------|----------|--------|
| a      | 5                | 4.3             | 1.2647             | non                                       | 0.7907  | 10       | CM-1C  |
| b      | 10               | 3.8             | 1.9191             | oui                                       | 0.5210  | 2        | Passif |
| c      | 8                | 4               | 1.6305             | oui                                       | 0.6123  | 6        | Passif |
| d      | 6                | 4.2             | 1.6941             | oui                                       | 0.5902  | 5        | Passif |
| e      | 7                | 4.1             | 1.2893             | non                                       | 1.2893  | 11       | CM-1C  |
| f      | 11               | 3.7             | 1.98               | -   | -       | -        | CH     |
| g      | 6                | 4.2             | 1.5162             | oui                                       | 0.6595  | 16       | Passif |
| h      | 5                | 4.3             | 1.1082             | non                                       | 0.9821  | 17       | CM-1C  |
| i      | 6                | 4.2             | 1.4823             | oui                                       | 0.6746  | 8        | Passif |
| j      | 8                | 4               | 1.7534             | -   | -       | -        | CH     |
| k      | 6                | 4.2             | 1.7004             | oui                                       | 0.5880  | 12       | Passif |
| l      | 7                | 4.1             | 1.6201             | oui                                       | 0.6285  | 13       | Passif |
| m      | 9                | 3.9             | 1.7727             | oui                                       | 0.5641  | 4        | Passif |
| n      | 6                | 4.2             | 1.2209             | non                                       | 0.8190  | 15       | CM-1C  |
| o      | 7                | 4.1             | 1.5589             | oui                                       | 0.6414  | 7        | Passif |
| p      | 8                | 4               | 1.5625             | -   | -       | -        | CH     |
| q      | 5                | 4.3             | 1.5357             | non                                       | 0.5076  | 14       | CM-1C  |
| r      | 11               | 3.7             | 1.96               | oui                                       | 0.5076  | 1        | Passif |
| s      | 5                | 4.3             | 1.4726             | oui                                       | 0.6790  | 9        | Passif |
| t      | 8                | 4               | 1.8823             | oui                                       | 0.5352  | 3        | Passif |

TABLE 3.2 – Scénario illustratif de HPRC : résultats de la première période.

En effet, le capteur "f" possède le poids le plus élevé parmi ses voisins, alors il se déclare comme cluster-head et envoie un "HEAD" à ses voisins à 1-saut. Les noeuds (b, c, d, e, g, h ; i, j, k, l, m, n, o, p, q, r, s, t) évaluent leurs taux de redondance pour décider de leur activité. Lors de la réception de "HEAD" par les capteurs t, d, r, b, m, c, e, o, s, a et i chacun évalue sa redondance et la compare à  $R_{seuil}$ . Si le taux de redondance du capteur dépasse le seuil alors il calcule un délai d'attente inversement proportionnel à son poids (formule 3.2).

Le temps d'attente du capteur "r" expire en premier. N'ayant reçu aucun message passif, le noeud "r" se met à l'état passif et envoie un "PASSIF" à ses voisins. Lorsque les noeuds t, d, i, q, j, k, n, l, b, m et f reçoivent "PASSIF" envoyé par le noeud "r", ils ré-évaluent leur redondance.

Etant toujours couvert, lorsque le temps d'attente des capteurs "b" et "t" expire, ils se mettent à l'état passif et envoient "PASSIF" à leurs voisins. Lors de la réception du message

passif par "m" et "d", ils ré-évaluent leurs redondance. Ce dernier étant au delà du seuil, les capteurs "m" et "d" décident alors de se mettre à l'état passif et envoient un "PASSIF" après l'expiration de leurs temps d'attente.

Lorsque le noeud "j" reçoit "PASSIF" en provenance de "b", il devient le noeud possédant le poids le plus élevé parmi ses voisins. IL se déclare alors cluster-head en envoyant un "HEAD" qui sera reçu par les noeuds n, k, l et q. Ces derniers comparent leur taux de redondance puis déclenche un temp d'attente.

Le temp d'attente des noeuds "c", "o" et "i" expire et leur taux de redondance est toujours supérieur au seuil, ils décide alors de se mettrent à l'état inactif et envoyer "PASSIF" à leur voisinage. Le noeud "s" reçoit "PASSIF" envoyé par "o", il évalue sa redondance et trouve qu'il est couvert, il pourra donc se mettre en mode passif en envoyant "PASSIF" à ses voisins.

Lors de la réception de "PASSIF" par "p" et qui a été envoyé par "c", le capteur "p" devient le noeud ayant le poids max parmi ses voisins. Donc il se déclare cluster-head et envoi "HEAD" qui sera reçu par les noeuds "h" et "g".

Lors de la réception de "PASSIF" de "s" par les noeuds "a" et "e", ils ré-évaluent leur taux de redondance puis le comparent et le trouve inférieure au seuil. Ils décident donc d'être actif et envoient un "JOIN-1S" à leurs voisins.

Lorsque le temps d'attente du capteur "k" expire, et son taux de redondance est supérieur au seuil, alors il se met à l'état passif et envoi un "PASSIF" à son voisinage. Le noeuds "l" reçoit le message passif de "k", il ré-évalue sa redondance. Cette dernière étant supérieur au seuil, il décide donc de se mettre à l'état passif et envoie "PASSIF".

Lorsque le noeud "q" reçoit "PASSIF" de "i", il ré-évalue sa redondance. Cette dernière étant inférieure au seuil, donc le noeud "q" décide d'être un membre actif à 1-saut et envoie "JOIN-1S" à ses voisins.

Lors de la réception de "PASSIF" de "l" par les noeuds "n" et "g", ils ré-évaluent leur taux de redondance puis le comparent, le noeuds "n" le trouve inférieure au seuil donc il décide d'être actif et envoi "JOIN-1S", tandis que le noeud "g" le trouve supérieur au seuil donc il décide de se mettre à l'état passif en envoyant "PASSIF". Le noeud "h" reçoit se dernier, il ré-évalue son taux de redondance puis le compare et le trouve inférieur au seuil donc il décide d'être actif et envoi "JOIN-1S".



La figure 3.3 illustre la décision prise par chaque capteur lors de la première période.

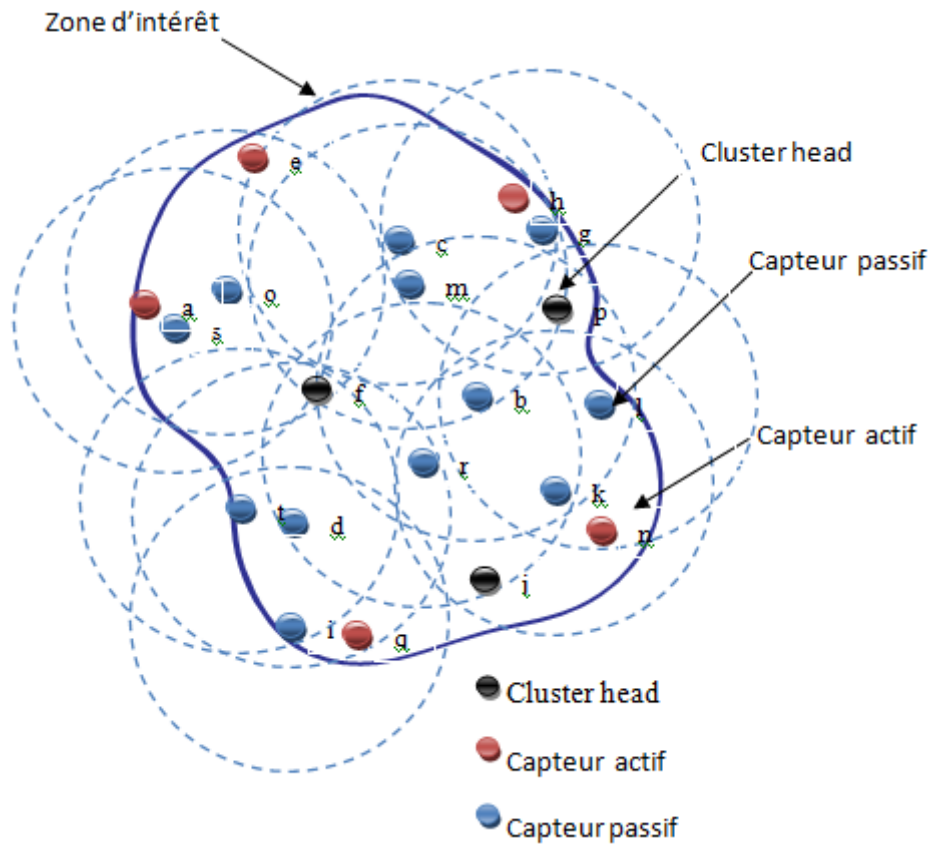


FIGURE 3.3 – Scénario illustratif de HPRC : état des capteurs à la période 1.

À la fin de chaque période d'activité, les noeuds capteurs dont l'état est cluster head ou actif, calculent l'énergie moyenne consommée lors des périodes précédentes.

### Deuxième période d'activité

Le tableau ci-dessous illustre les valeurs de la consommation d'énergie durant la première période :

|                   |     |     |     |     |     |     |     |     |     |     |
|-------------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Noeud             | a   | b   | c   | d   | e   | f   | g   | h   | i   | j   |
| Energie consommée | 1.3 | 0.3 | 0.5 | 0.5 | 1   | 1.9 | 0.6 | 1   | 0.6 | 1.7 |
| Noeud             | k   | l   | m   | n   | o   | p   | q   | r   | s   | t   |
| Energie consommée | 0.5 | 0.6 | 0.5 | 1.4 | 0.5 | 2.2 | 1.3 | 0.3 | 0.4 | 0.4 |

TABLE 3.3 – Scénario illustratif de HPRC : l'énergie consommée durant la première période.

L'exécution du protocole HPRC durant la deuxième période, permet l'obtention des résultats illustrés dans le tableau ci-dessous :

| Noeuds | Deg <sub>i</sub> | ER <sub>i</sub> | Poids <sub>i</sub> | T <sub>x</sub> Red ≥ Red <sub>Seuil</sub> | BackOff | Décision | Etat   |
|--------|------------------|-----------------|--------------------|---|---------|----------|--------|
| a      | 5                | 3               | 0.8823             | non                                       | -       | -        | CM-2C  |
| b      | 10               | 3.4             | 1.7171             | oui                                       | 0.5210  | 2        | Passif |
| c      | 8                | 3.5             | 1.4693             | -   | -       | -        | CH     |
| d      | 6                | 3.7             | 1.800              | oui                                       | 0.7812  | 7        | Passif |
| e      | 7                | 3.1             | 0.8732             | non                                       | 1.1452  | 15       | CM-1C  |
| f      | 11               | 1.8             | 0.9677             | non                                       | 1.0333  | 10       | CM-1C  |
| g      | 6                | 3.6             | 1.2996             | oui                                       | 0.7694  | 13       | Passif |
| h      | 5                | 3.3             | 0.7656             | oui                                       | 1.3061  | 14       | Passif |
| i      | 6                | 3.6             | 1.3074             | oui                                       | 0.7648  | 6        | Passif |
| j      | 8                | 2.3             | 1.0222             | oui                                       | 0.9782  | 9        | Passif |
| k      | 6                | 3.7             | 1.4979             | oui                                       | 0.6500  | 4        | Passif |
| l      | 7                | 3.5             | 1.3833             | oui                                       | 0.7229  | 5        | Passif |
| m      | 9                | 3.4             | 1.5962             | oui                                       | 0.6264  | 2        | Passif |
| n      | 6                | 2.8             | 0.8138             | non                                       | 1.2286  | 11       | CM-1C  |
| o      | 7                | 3.6             | 1.3688             | oui                                       | 0.7305  | 12       | Passif |
| p      | 8                | 1.8             | 0.7031             | non                                       | 1.4222  | 16       | CM-1C  |
| q      | 5                | 3               | 1.0714             | non                                       | 0.9333  | 8        | CM-1C  |
| r      | 11               | 3.4             | 1.8181             | -   | -       | -        | CH     |
| s      | 5                | 3.9             | 1.3356             | non                                       | -       | -        | CM-1C  |
| t      | 8                | 3.6             | 1.5929             | oui                                       | 0.6277  | 3        | Passif |

TABLE 3.4 – Scénario illustratif de HPRC : résultats de la deuxième période.

La figure 3.4 illustre la décision prise par chaque capteur lors de la deuxième période.

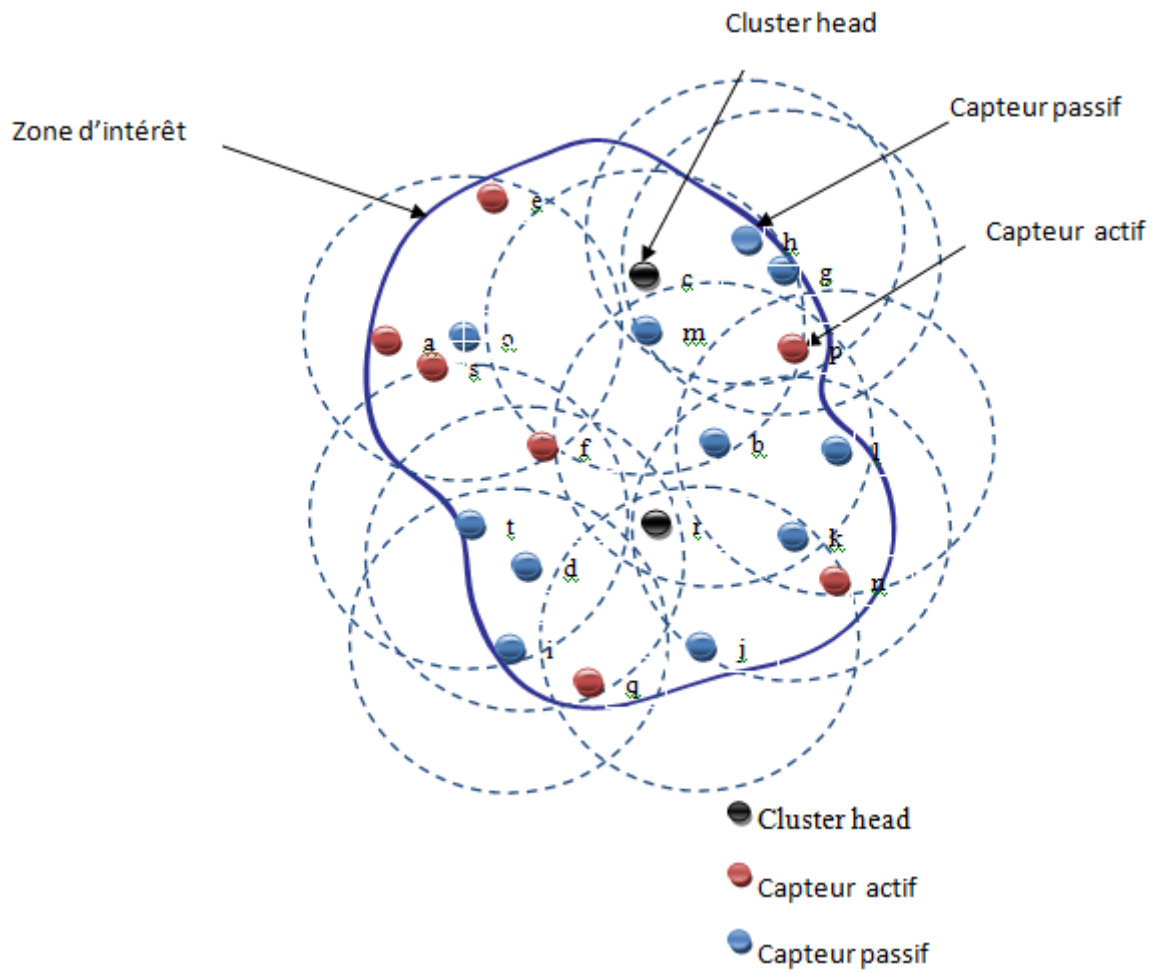


FIGURE 3.4 – Scénario illustratif de HPRC : état des capteurs à la période 2.

### Troisième période d'activité

Le tableau ci-dessous illustre les valeurs de la consommation d'énergie durant la deuxième période :

|                          |     |     |     |     |     |     |     |     |     |     |
|--------------------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| <b>Noeud</b>             | a   | b   | c   | d   | e   | f   | g   | h   | i   | j   |
| <b>Energie consommée</b> | 1   | 0.3 | 1.4 | 0.3 | 1   | 0.6 | 0.7 | 0.5 | 0.4 | 0.7 |
| <b>Noeud</b>             | k   | l   | m   | n   | o   | p   | q   | r   | s   | t   |
| <b>Energie consommée</b> | 0.4 | 0.5 | 0.4 | 1   | 0.4 | 0.9 | 0.8 | 2.2 | 0.5 | 0.4 |

TABLE 3.5 – Scénario illustratif de HPRC : l'énergie consommée durant la deuxième période.

L'exécution du protocole HPRC durant la troisième période, permet l'obtention des résultats illustrés dans le tableau ci-dessous :

| Noeuds | Deg <sub>i</sub> | ER <sub>i</sub> | Poids <sub>i</sub> | T <sub>x</sub> Red ≥ Red <sub>Seuil</sub> | BackOff | Décision | Etat   |
|--------|------------------|-----------------|--------------------|---|---------|----------|--------|
| a      | 5                | 2               | 0.5882             | non                                       | 1.7001  | 14       | CM-1C  |
| b      | 10               | 3.1             | 1.5656             | -   | -       | -        | CH     |
| c      | 8                | 2.1             | 0.8571             | oui                                       | 1.1667  | 9        | Passif |
| d      | 6                | 3.4             | 1.349              | oui                                       | 0.7412  | 2        | Passif |
| e      | 7                | 2.1             | 0.6600             | non                                       | 1.5151  | 16       | CM-1C  |
| f      | 11               | 1.2             | 0.6451             | non                                       | 1.5501  | 12       | CM-1C  |
| g      | 6                | 2.9             | 1.0469             | oui                                       | 0.9552  | 8        | Passif |
| h      | 5                | 2.8             | 0.7216             | -   | -       | -        | CH     |
| i      | 6                | 3.4             | 1.1307             | oui                                       | 0.8844  | 7        | Passif |
| j      | 8                | 1.6             | 0.7111             | oui                                       | 1.4062  | 11       | Passif |
| k      | 6                | 3.3             | 1.3360             | oui                                       | 0.7485  | 3        | Passif |
| l      | 7                | 3               | 1.1857             | oui                                       | 0.8433  | 5        | Passif |
| m      | 9                | 3               | 1.4084             | oui                                       | 0.7100  | 1        | Passif |
| n      | 6                | 1.8             | 0.5278             | non                                       | 1.8946  | 15       | CM-1C  |
| o      | 7                | 3.2             | 1.3107             | oui                                       | 0.8259  | 4        | Passif |
| p      | 8                | 0.9             | 0.3515             | oui                                       | 2.8449  | 17       | Passif |
| q      | 5                | 2.2             | 0.7857             | non                                       | 1.2727  | 10       | CM-1C  |
| r      | 11               | 1.2             | 0.6417             | oui                                       | 1.5583  | 13       | Passif |
| s      | 5                | 3.4             | 1.1643             | oui                                       | 0.8588  | 6        | Passif |
| t      | 8                | 3.2             | 1.4159             | -   | -       | -        | CH     |

TABLE 3.6 – Scénario illustratif de HPRC : résultats de la troisième période.

La figure 3.5 illustre la décision prise par chaque capteur lors de la troisième période.

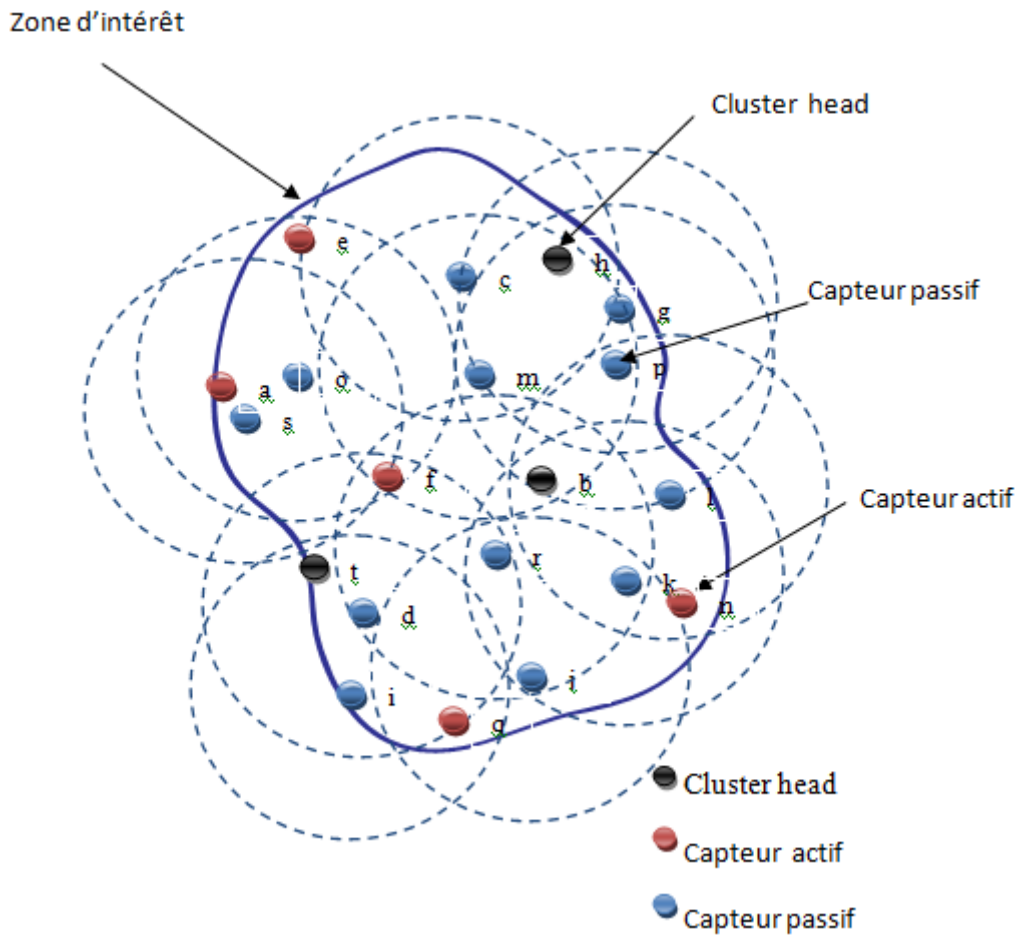


FIGURE 3.5 – Scénario illustratif de HPRC : état des capteurs à la période 3.

### Quatrième période d'activité

Le tableau ci-dessous illustre les valeurs de la consommation d'énergie durant la troisième période :

|                          |     |     |     |     |     |     |     |     |     |     |
|--------------------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| <b>Noeud</b>             | a   | b   | c   | d   | e   | f   | g   | h   | i   | j   |
| <b>Energie consommée</b> | 1   | 1   | 0.6 | 0.4 | 1.3 | 1.2 | 0.5 | 1.7 | 0.4 | 0.7 |
| <b>Noeud</b>             | k   | l   | m   | n   | o   | p   | q   | r   | s   | t   |
| <b>Energie consommée</b> | 0.3 | 0.4 | 0.3 | 1.1 | 0.3 | 0.9 | 1.2 | 0.9 | 0.4 | 1   |

TABLE 3.7 – Scénario illustratif de HPRC : l'énergie consommée durant la troisième période.

L'exécution du protocole HPRC durant la quatrième période, permet l'obtention des résultats illustrés dans le tableau ci-dessous :

| Noeuds | Deg <sub>i</sub> | ER <sub>i</sub> | Poids <sub>i</sub> | T <sub>x</sub> Red ≥ Red <sub>Seuil</sub> | BackOff | Décision | Etat       |
|--------|------------------|-----------------|--------------------|---|---------|----------|------------|
| a      | 4                | 1               | 0.2544             | non                                       | 3.9308  | 10       | CM-1C      |
| b      | 7                | 2.1             | 0.7217             | non                                       | 1.3858  | 6        | CM-1C      |
| c      | 6                | 1.5             | 0.5154             | oui                                       | 1.9402  | 7        | Passif     |
| d      | 4                | 3               | 0.8287             | oui                                       | 1.2067  | 3        | Passif     |
| e      | 6                | 0.8             | 0.2234             | non                                       | 4.4762  | 11       | CM-1C      |
| f      | -                | 0               | -                  | -   | -       | -        | Défaillant |
| g      | 5                | 2.4             | 0.7057             | non                                       | -       | -        | CM-1C      |
| h      | 4                | 1.1             | 0.2444             | non                                       | -       | -        | CM-2C      |
| i      | 4                | 2.8             | 0.8459             | -   | -       | -        | CH         |
| j      | 7                | 0.9             | 0.3345             | oui                                       | 2.9895  | 8        | Passif     |
| k      | 4                | 3               | 0.9615             | -   | -       | -        | CH         |
| l      | 5                | 2.6             | 0.7428             | oui                                       | 1.1362  | 5        | Passif     |
| m      | 6                | 2.7             | 0.8681             | oui                                       | 1.1519  | 2        | Passif     |
| n      | 4                | 0.7             | 0.1535             | oui                                       | 6.5146  | 12       | Passif     |
| o      | 6                | 2.9             | 0.9764             | -   | -       | -        | CH         |
| p      | -                | 0               | -                  | -   | -       | -        | Défaillant |
| q      | 4                | 1               | 0.2762             | non                                       | 3.6205  | 9        | CM-1C      |
| r      | -                | 0.3             | -                  | -   | -       | -        | Défaillant |
| s      | 5                | 3               | 0.8746             | oui                                       | 1.1433  | 1        | Passif     |
| t      | 8                | 2.2             | 0.7942             | non                                       | 1.2591  | 4        | CM-1C      |

TABLE 3.8 – Scénario illustratif de HPRC : résultats de la quatrième période.

La figure 3.6 illustre la décision prise par chaque capteur lors de la quatrième période.

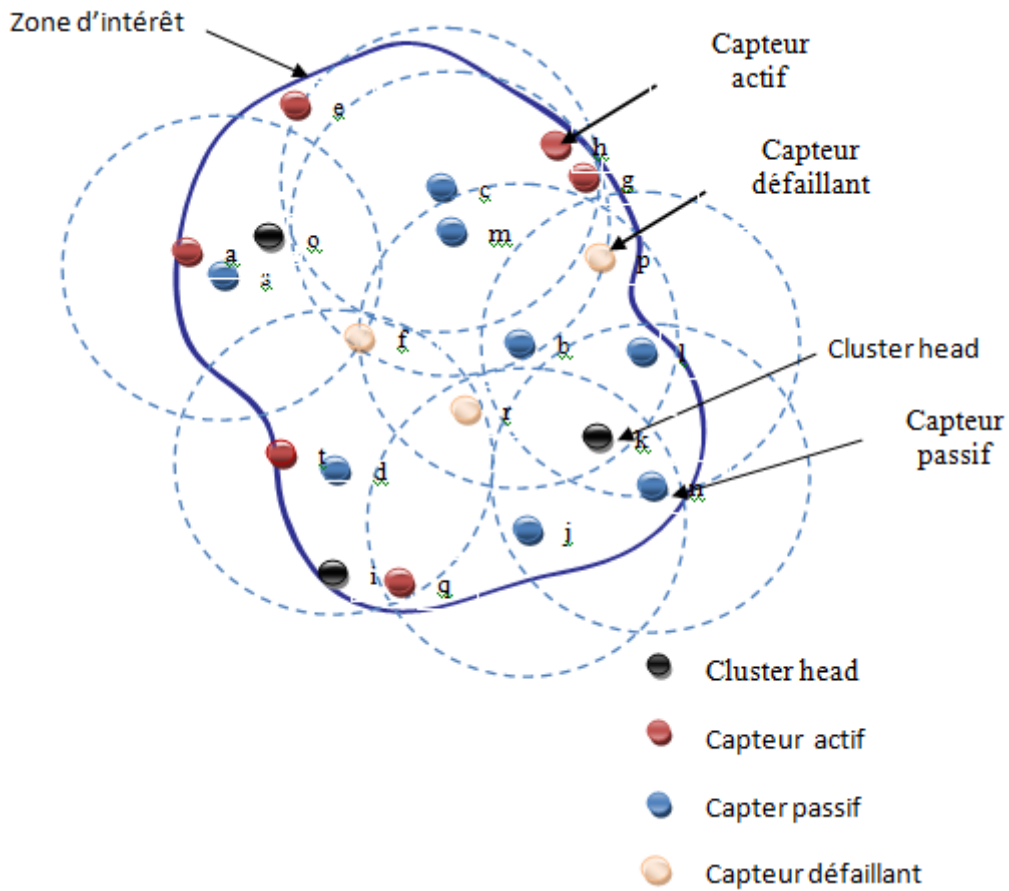


FIGURE 3.6 – Scénario illustratif de HPRC : état des capteurs à la période 4.

### Cinquième période d'activité

Le tableau ci-dessous illustre les valeurs de la consommation d'énergie durant la quatrième période :

|                          |   |     |     |     |     |   |   |   |     |     |
|--------------------------|---|-----|-----|-----|-----|---|---|---|-----|-----|
| <b>Noeud</b>             | a | b   | c   | d   | e   | f | g | h | i   | j   |
| <b>Energie consommée</b> | 1 | 1   | 0.4 | 0.3 | 0.8 | 0 | 1 | 1 | 1   | 0.7 |
| <b>Noeud</b>             | k | l   | m   | n   | o   | p | q | r | s   | t   |
| <b>Energie consommée</b> | 1 | 0.3 | 0.3 | 0.7 | 1.3 | 0 | 1 | 0 | 0.3 | 0.8 |

TABLE 3.9 – Scénario illustratif de HPRC : l'énergie consommée durant la quatrième période.

L'exécution du protocole HPRC durant la cinquième période, permet l'obtention des résultats illustrés dans le tableau ci-dessous :

| Noeuds | Deg <sub>i</sub> | ER <sub>i</sub> | Poids <sub>i</sub> | T <sub>x</sub> Red ≥ Red <sub>Seuil</sub> | BackOff | Décision | Etat       |
|--------|------------------|-----------------|--------------------|---|---------|----------|------------|
| a      | -                | 0               | -                  | -   | -       | -        | Défaillant |
| b      | 5                | 1.1             | 0.2989             | oui                                       | 3.3456  | 4        | Passif     |
| c      | 4                | 1.1             | 0.2350             | non                                       | 14.2553 | 1        | CM-1C      |
| d      | 2                | 2.7             | 0.6352             | -   | -       | -        | CH         |
| e      | -                | 0               | -                  | -   | -       | -        | Défaillant |
| f      | -                | 0               | -                  | -   | -       | -        | Défaillant |
| g      | 4                | 1.4             | 0.0.2947           | oui                                       | 3.3932  | 6        | Passif     |
| h      | -                | 0.1             | -                  | -   | -       | -        | Défaillant |
| i      | 2                | 1.8             | 0.3272             | non                                       | 3.0562  | 3        | CM-1C      |
| j      | -                | 0.2             | -                  | -   | -       | -        | Défaillant |
| k      | 2                | 2               | 0.2962             | non                                       | 3.3760  | 5        | CM-1C      |
| l      | 3                | 2.3             | 0.4063             | -   | -       | -        | CH         |
| m      | 4                | 2.4             | 0.5647             | -   | -       | -        | CH         |
| n      | -                | 0               | -                  | -   | -       | -        | Défaillant |
| o      | 4                | 1.6             | 0.0.3385           | non                                       | 3.0441  | 2        | CM-1C      |
| p      | -                | 0               | -                  | -   | -       | -        | Défaillant |
| q      | -                | 0               | -                  | -   | -       | -        | Défaillant |
| r      | -                | 0.3             | -                  | -   | -       | -        | Défaillant |
| s      | 2                | 2.7             | 0.3857             | -   | -       | -        | CH         |
| t      | 4                | 1.4             | 0.3349             | oui                                       | 2.9859  | 1        | Passif     |

TABLE 3.10 – Scénario illustratif de HPRC : résultats de la cinquième période.

La figure 3.7 illustre la décision prise par chaque capteur lors de la cinquième période. A cette étape, le protocole s'arrête car le capteur q, i et e sont défaillants, ils n'ont pas assez d'énergie pour devenir cluster head, et ils ne sont couverts par aucun de leurs voisins, donc la zone de déploiement n'est pas totalement couverte par les noeuds restants.



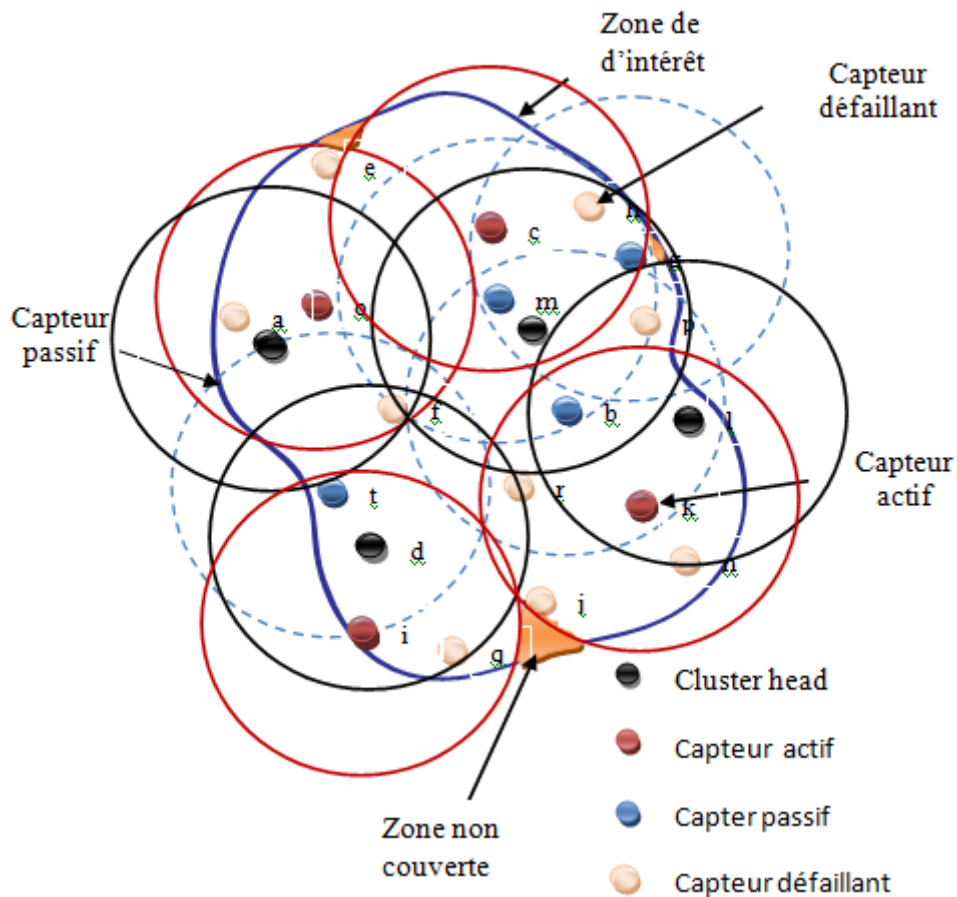


FIGURE 3.7 – Scénario illustratif de HPRC : état des capteurs à la période 5.

### 3.6 Conclusion

Dans ce chapitre, nous avons présenté un protocole distribué et hiérarchique, baptisé HPRC, permettant la préservation de la couverture et la réduction du trafic circulant dans les réseaux de capteurs homogènes. Cette approche se base sur deux principes : le premier consiste en le calcul d'un poids en fonction de l'énergie résiduelle, le degré et la distance moyenne pour l'élection des cluster-heads, et le deuxième consiste en l'évaluation de la redondance pour l'ordonnement des membres de clusters. Le protocole HPRC requière la connaissance de positions des capteurs qui offre une meilleure estimation lors de l'évaluation de la redondance d'un noeud et une décision appropriée, ce qui permet d'éviter les trous de couverture.

Pour évaluer les performances du protocole HPRC, le chapitre suivant abordera la validation

de cette approche par des scénarios de simulation. Cet outil permettra le calcul de certains paramètres tels que, le taux de couverture, la consommation énergétique et le nombre de noeuds actifs.

## CHAPITRE 4

# SIMULATION ET ÉVALUATION DES PERFORMANCES

### 4.1 Introduction

Ce chapitre est consacré à la validation de notre protocole, baptisé HPRC, par des scénarios de simulation. Pour cela, nous avons développé un simulateur en JAVA permettant de comparer les performances de la solution proposée avec celle du protocoles DEECIC [34] en matière de : taux de couverture, nombre de noeuds vivants, taille des clusters (nombre moyen de membres par cluster), énergie consommée et longévité du réseau.

### 4.2 Environnement de simulation et paramètres

#### 4.2.1 Description du système

Notre système modélise une zone de déploiement bidimensionnelle contenant des capteurs et une station de base. Les différentes entités de ce système sont :

**La station de base** : elle est responsable de l'envoi des requêtes et de la collecte des données du réseau.

**Les capteurs** : ceux-ci ont comme tâches la couverture de la zone d'intérêt, la détection des événements ainsi que leur traitement et transmission. Notre modèle consiste en un réseau composé de capteurs statiques, homogènes et déployés aléatoirement avec une forte densité sur la zone à surveiller.

**Les messages** : lors de la découverte du voisinage et de la formation des clusters, un certain nombre de messages sont échangés entre les capteurs : POS, POIDS, HEAD, JOIN-1S, JOIN-2S, PASSIF et FAIL.

### 4.2.2 Les variables du système

Le tableau suivant représente les différentes variables utilisées pour l'implémentation du simulateur.

| Variable               | Définition  | Type   | Unité de mesure |
|------------------------|---|--------|-----------------|
| RCap                   | Rayon de captage  | Entier | Mètre           |
| RCom                   | Rayon de communication                                  | Entier | Mètre           |
| ER                     | Energie courante  | Réel   | Joule           |
| pcouverture            | Taux de redondance                                      | Réel   | -               |
| (x, y)                 | Coordonnées géographiques                               | Réel   | Mètre           |
| id                     | Identifiant   | Entier | -               |
| dist(i, j)             | La distance séparant un noeud de son voisin             | Réel   | Mètre           |
| voisin                 | Le voisinage d'un capteur                               | Entier | -               |
| poids                  | Le poids d'un noeud                                     | Réel   | Joule/Mètre     |
| poidsvoisins           | Les poids des voisins                                   | Réel   | -               |
| degre                  | Le nombre de voisins d'un noeud                         | Entier | -               |
| etat                   | Etat d'un noeud   | Entier | -               |
| Energie_Periode        | La quantité d'énergie consommée pendant une période     | Réel   | Joule           |
| Energie_Periode-1Saut  | La quantité d'énergie consommée par un membre à 1-saut  | Réel   | Joule           |
| Energie_Periode-2Sauts | La quantité d'énergie consommée par un membre à 2-sauts | Réel   | Joule           |
| Energie_Periode-CH     | La quantité d'énergie consommée par un cluster-head     | Réel   | Joule           |
| Nombre_NOEUDS          | Le nombre de noeuds du réseau                           | Entier | -               |
| NBR_PERIODE            | Le nombre d'itérations                                  | Entier | -               |

TABLE 4.1 – Les variables descriptives du système.

### 4.2.3 Événements discrets du système

Les variables descriptives définies auparavant sont mises à jour par un ensemble d'événements discrets dans le temps. Le tableau ci-dessous illustre les principaux événements qui peuvent survenir dans le système.

| Événement  | Description  |
|--|--|
| Calcul du poids                                      | Cet événement déclenche l'élection de cluster-heads  |
| Début d'élection                                     | Cet événement déclenche l'exécution du protocole HPRC  |
| Evaluation de la couverture                          | Cet événement se produit après l'élection  |
| Début d'ordonnancement et de désignation des membres | Cet événement est déclenché lors de la réception de "HEAD"   |
| Message d'association à 1-saut                       | Cet événement se produit lorsque la redondance d'un noeud est inférieure au seuil                              |
| Message d'association à 2-sauts                      | Cet événement a lieu lors de la réception du message d'association 1-saut                                      |
| Calcul du temps d'attente                            | Cet événement se produit lorsque la redondance d'un noeud est supérieure au seuil                              |
| Message de mise en veille                            | Cet événement se produit lorsque le temps d'attente expire et la redondance d'un noeud est supérieure au seuil |
| La réévaluation de la couverture                     | Cet événement est déclenché lors de la réception d'un message "PASSIF"   |
| La réélection  | Cet événement est déclenché lors de la réception d'un message "PASSIF" ou d'un message d'association à 2-sauts |

TABLE 4.2 – Les événements discrets du système.

### 4.3 Les étapes de création du simulateur

L'implémentation de notre simulateur est réalisé en plusieurs étapes. Les résultats obtenus sont la moyenne de 100 simulations pour sa comparaison avec DEECIC [34]. Les différentes étapes se résument comme suit :

#### 4.3.1 Initialisation des variables

Cette étape permet de déterminer les mesures de la zone d'intérêt, la tailles du réseau et les caractéristiques des capteurs (rayons de communication et de captage, Energie initiale,etc.).

#### 4.3.2 Création de la topologie

Les noeuds constituant notre réseau sont déployés aléatoirement sur une zone de (100 \* 100) m<sup>2</sup> afin d'avoir un modèle de simulation de HPRC semblable à celui de DEECIC [34].

En variant le nombre de période, la topologie reste la même jusqu'à ce que la durée de vie du réseau se termine. Chaque noeud est représenté par des coordonnées  $(x, y)$ .

### 4.3.3 Calcul des distances, voisins et poids

Nous constituons la table de voisinage en comparant la distance séparant deux noeuds par rapport au rayon de communication. Ensuite, nous créons les tables de distances et de poids des voisins permettant l'élection et l'évaluation de la redondance.

### 4.3.4 Application du protocole HPRC

L'application de l'algorithme HPRC n'exige pas une couverture complète de la zone initialement. A la fin de toute période, chaque noeud actif compare son énergie résiduelle à la moyenne des énergies consommées pendant les périodes antérieures, si elle est strictement inférieure, le noeud se déclare défaillant.

### 4.3.5 Affichage des résultats

L'affichage des résultats permet de tracer les courbes comparatives en se basant sur plusieurs paramètres de performance tels que : le taux de couverture, le nombre de noeuds vivants, la taille des clusters, la consommation totale d'énergie et la durée de vie. Ces paramètres sont les critères principaux pour décider de la performance d'un protocole de couverture utilisant le clustering.

#### 4.3.5.1 Le taux de couverture

Il représente le rapport moyen entre la surface couverte par les capteurs actifs et la surface totale de la zone cible. Pour notre comparaison (HPRC et DEECIC), nous examinons la couverture de 100 points obtenus par la division de la zone de déploiement en carrés de  $(100 * 100)$   $m^2$  et le nombre de points couverts divisé par 100 (le nombre total de points) représente le taux de couverture de la zone cible.

#### 4.3.5.2 Le nombre de noeuds vivants

Il représente le nombre moyen de capteurs ayant une énergie pouvant servir au fonctionnement durant une autre période d'activité.

### 4.3.5.3 La taille des clusters

Elle représente en moyenne le rapport entre le nombre de noeuds actifs et le nombre de clusters. En d'autres termes, c'est la moyenne de membres par cluster.

### 4.3.5.4 L'énergie totale consommée

Elle représente la quantité d'énergie moyenne consommée en joules par l'ensemble des capteurs du réseau.

### 4.3.5.5 La durée de vie

Elle représente le nombre moyen de périodes durant lesquelles le réseau assure un taux de couverture donné (dépendant de l'application).

## 4.4 Evaluation des performances

Pour évaluer les performances de HPRC le long de la durée de vie du réseau, nous avons opté pour la comparaison avec DEECIC [34] en adoptant le même modèle énergétique (détaillé en chapitre 3) et en variant le nombre de périodes avec 200 noeuds déployés aléatoirement. Les paramètres de simulation sont représentés dans le tableau 4.3. Quant aux résultats, ils sont la moyenne de 100 simulations.

| Paramètre   | valeur                       |
|---|------------------------------|
| Taille du paquet  | 4000 bits                    |
| Energie électronique $E_{elec}$                         | 50 nJ/bit                    |
| Energie d'amplification des membres $\epsilon_{fs}$     | 10 pJ/bit/m <sup>2</sup>     |
| Energie d'amplification du cluster-head $\epsilon_{mp}$ | 0.0013 pJ/bit/m <sup>2</sup> |
| Energie d'agrégation $E_{DA}$                           | 5 nJ/bit/signal              |
| Energie de capture $E_{cap}$                            | négligeable                  |
| Rayon de captage $RCap$                                 | 10 m                         |
| Rayon de communication $RCom$                           | 20 m                         |
| Taille de la zone                                       | (100 * 100)m <sup>2</sup>    |

TABLE 4.3 – Les paramètres de simulation : comparaison de HPRC avec DEECIC [34].

#### 4.4.1 Le taux de couverture

La figure 4.1 illustre la variation du taux de couverture à travers les périodes d'activité.

Les résultats montrent que jusqu'à la période 12, le taux de couverture fourni par DEECIC est légèrement supérieur à celui assuré par HPRC (100% contre 99%). Cela est dû au fait que dans DEECIC, tous les noeuds sont actifs, donc une couverture complète de la zone cible. En revanche, le protocole HPRC met en veille une partie des noeuds redondants conduisant ainsi à une légère dégradation du taux de couverture.

Après la douzième période, on remarque une forte dégradation du taux de couverture pour le protocole DEECIC alors que HPRC préserve une couverture acceptable (au dessus de 90%). Cela est expliqué par le fait que dans le protocole DEECIC, tous les noeuds restent actifs conduisant à une consommation énergétique élevée, et par conséquent, une augmentation du nombre de noeuds défaillants, donc la zone n'est plus couverte. Dans HPRC, l'ordonnancement d'activité permet l'équilibrage de charge et une stabilité du nombre de noeuds actifs durant chaque période, ce qui préserve la couverture.

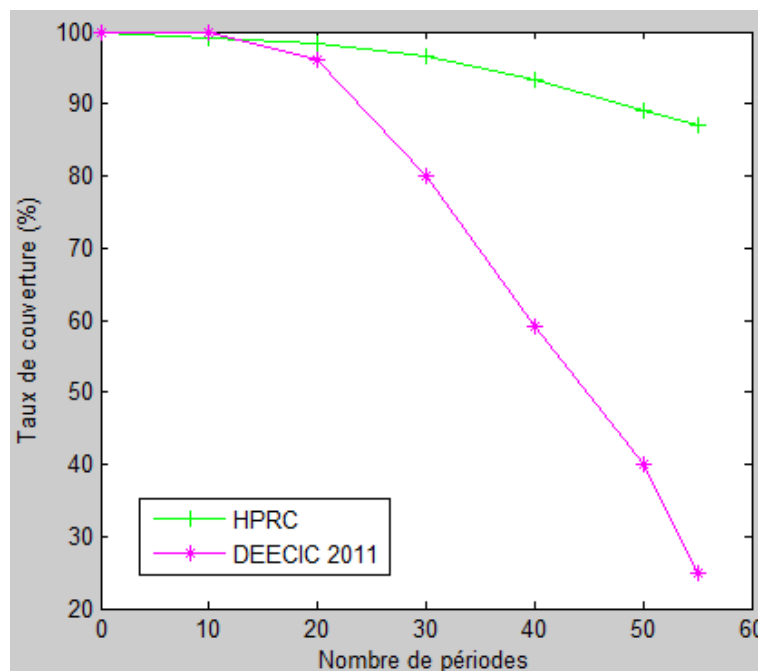


FIGURE 4.1 – Taux de couverture VS Nombre de périodes d'activité.



### 4.4.2 Le nombre de noeuds vivants

La figure 4.2 illustre la variation du nombre de noeuds vivants en fonction du nombre de périodes.

Nous constatons de la figure que le protocole HPRC possède plus de noeuds vivants en le comparant à DEECIC le long des périodes d'activité. En effet, la politique d'élection adoptée par HPRC permet le choix de nouveaux cluster-heads à chaque début d'une période, et l'ordonnement d'activité met en veille les membres redondants, ce qui permet d'équilibrer la charge et de réduire la défaillance des noeuds. Dans DEECIC, bien que certains noeuds sont inutiles à la couverture, leur activité permanente épuise leur énergie. En outre, un noeud sélectionné comme étant cluster-head reste dans cet état jusqu'à sa défaillance. Ce partage non-équitable des tâches du réseau, provoque une dégradation continue du nombre de noeuds vivants.

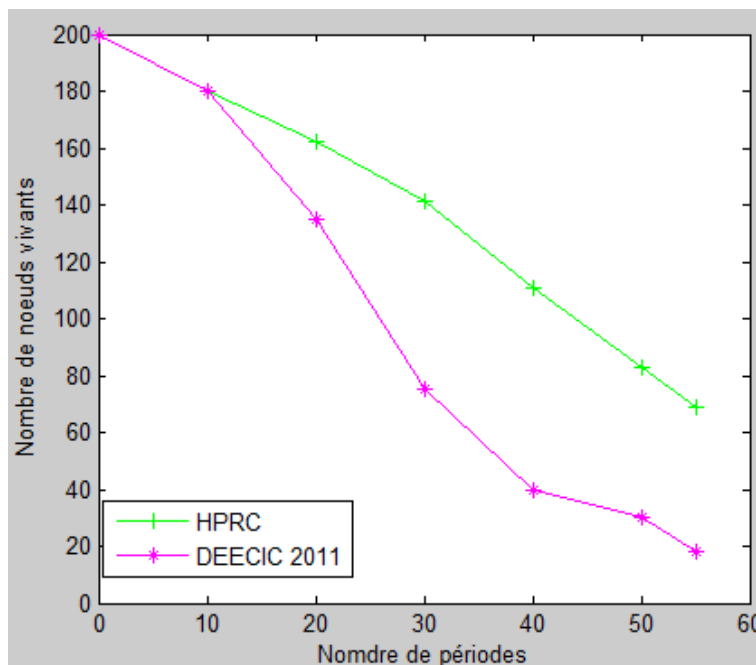


FIGURE 4.2 – Nombre de noeuds vivants VS Nombre de périodes d'activité.

### 4.4.3 La taille des clusters

La figure suivante illustre sous forme d'un histogramme la taille moyenne des clusters en fonction du nombre de périodes.

Nous remarquons que durant les dix premières périodes, le nombre moyen de membres par cluster induit par DEECIC est plus élevé que celui de HPRC car initialement tous les noeuds sont vivants (et actifs) dans DEECIC et ils sont repartis sur le nombre de clusters formés. Par

contre dans HPRC, seulement les noeuds désignés comme actifs seront regroupés, dans des clusters, suivant leur décision en membres à un saut ou à deux sauts. En effet, cela est lié à l'ordonnement d'activité et non pas aux nombre de clusters formés.

Après la dixième période, la taille des clusters dans DEECIC diminue car le nombre de noeuds vivants décroît (voir la figure 4.3). Pour HPRC, nous observons une légère variation du nombre moyen de membres par cluster car le nombre de noeuds actifs reste stable d'une période à une autre. Cette stabilité permet à HPRC de fournir une agrégation efficace de données.

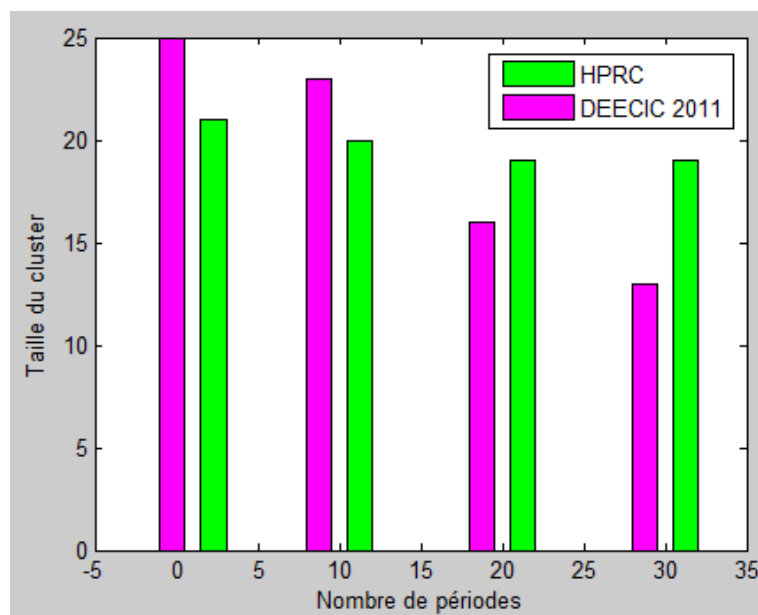


FIGURE 4.3 – La taille des clusters VS Nombre de périodes d'activité.

#### 4.4.4 La consommation totale d'énergie

La figure 4.4 montre la consommation totale d'énergie induite par HPRC et DEECIC à travers les périodes d'activité.

Les résultats indiquent que l'énergie consommée par HPRC est considérablement inférieure à celle de DEECIC. En effet, le protocole HPRC utilise, dans chaque période, une politique de mise en veille d'un ensemble de noeuds redondants, ce qui permet à ces noeuds de conserver leur précieuse énergie. Dans DEECIC, tous les noeuds sont actifs durant la période, ce qui épuise rapidement leur énergie conduisant ainsi à leur défaillance.

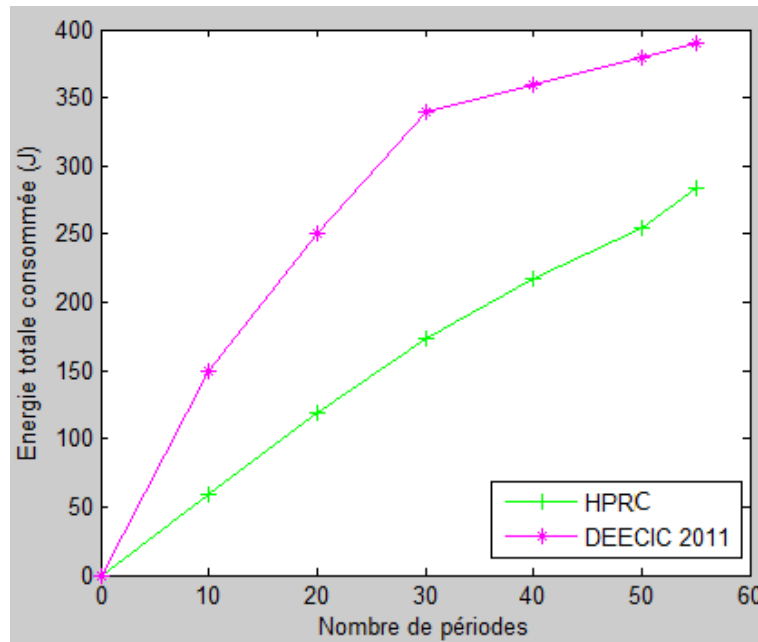


FIGURE 4.4 – Consommation totale d'énergie VS Nombre de périodes d'activité.

#### 4.4.5 La durée de vie du réseau

Pour la problématique de couverture, la longévité d'un réseau est définie comme étant la durée séparant le déploiement du moment où le taux de couverture exigé par l'application n'est plus assuré [8]. La figure 4.5 représente la mesure de la durée de vie (en nombre de périodes) par rapport au taux de couverture.

Le graphe montre que pour une couverture complète (avec un taux égale à 100%), DEECIC est meilleur en le comparant à HPRC en raison de l'activité de l'ensemble des noeuds déployés. Ensuite, nous observons que le protocole HPRC prolonge la durée de vie du réseau plus que DEECIC. Pour un taux de couverture égal à 90% (qui est exigé par un large nombre d'applications), le réseau a une durée de vie de 45 périodes dans le cas du protocole HPRC et 23 périodes pour DEECIC. Cette extension induite par HPRC est due à l'ordonnancement d'activité et à la politique d'élection permettant de conserver leur énergie. Cependant, DEECIC ne présente pas une bonne maîtrise énergétique, ce qui augmente la défaillance des noeuds ayant comme conséquence directe les trous de couverture et ainsi une durée de vie réduite.

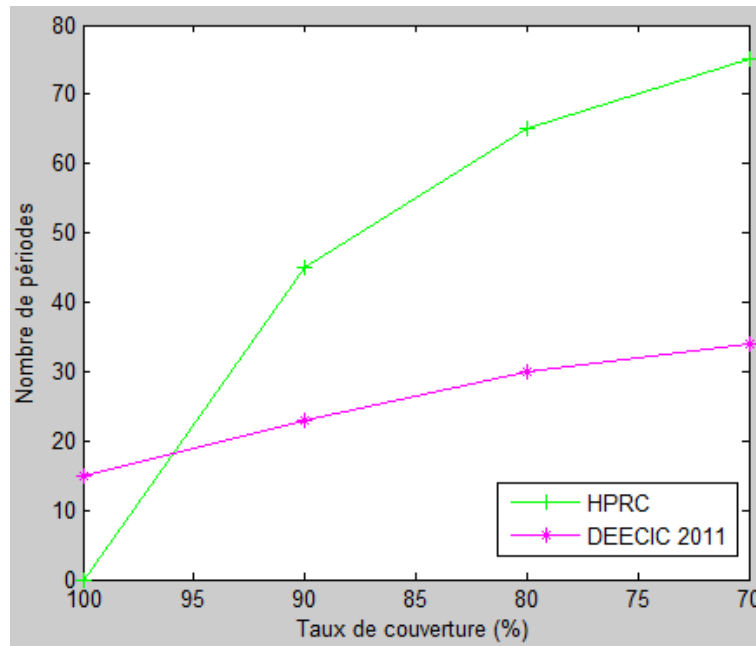


FIGURE 4.5 – Durée de vie VS Taux de couverture.

## 4.5 Conclusion

Dans ce chapitre, nous avons évalué les performances de notre approche **HPRC** en la comparant avec un protocole récents : DEECIC [34] en variant le nombre de périodes. Cette comparaison est faite à base de plusieurs paramètres de performance, tels que le taux de couverture, le nombre de noeuds vivants, la taille des clusters, la durée de vie et la consommation énergétique.

Les résultats obtenus montrent que le protocole HPRC préserve un taux de couverture quasi-totale avec un nombre élevé de noeuds vivants, ceci en présentant une meilleure maîtrise de la consommation énergétique. Ce qui lui permet d'étendre la longévité du réseau.

## CONCLUSION GÉNÉRALE ET PERSPECTIVES

Les réseaux de capteurs sans fil ont été conçus pour répondre aux besoins des diverses applications, telles que les applications de surveillance. Les caractéristiques du médium sans fil, l'inaccessibilité aux environnements de déploiement et la contrainte énergétique ont fait du problème de la couverture un aspect fondamental pour ce type de réseau. Ce problème consiste à réduire d'une façon optimale le chevauchement entre les zones de perception des capteurs en assurant la couverture complète de la zone cible. L'une des stratégies de résolution proposées pour cette problématique est l'ordonnancement d'activité des nœuds. Cependant, lors du passage à l'échelle, les protocoles de couverture à plat n'assurent pas une conservation optimale d'énergie, d'où l'idée de combiner l'ordonnancement avec l'hiérarchisation virtuelle du réseau dans une même solution. Dans cette optique, nous avons proposé un nouveau protocole de couverture appelé HPRC (Pour Hierarchical Protocol based-on Redundancy evaluation for Coverage garantie) basé sur le Clustering et adoptant une politique de mise en veille utilisant l'évaluation de la redondance des capteurs.

En effet, HPRC permet d'élire les cluster-heads à base d'un poids calculé en fonction du degré, de la distance par rapport au voisinage et de l'énergie (la dernière la plus cruciale). Ce choix est motivé par le fait que la réduction de la distance moyenne par rapport au voisinage permet de minimiser les dépenses énergétiques en transmission et en cas de pertes des paquets. Un degré élevé a l'avantage d'augmenter la taille des clusters et, par conséquent, le volume de données à agréger. Toutefois, l'ordonnancement et la désignation des membres sont effectués selon le critère de redondance des nœuds évaluée par la technique de discrétisation de la zone de perception des capteurs.

Pour la validation du protocole proposé, nous avons développé notre propre simulateur en JAVA. Les scénarios de simulation réalisés et les résultats comparatifs de HPRC avec

une autre solutions récente, ont montré que la politique d'élection et la technique d'ordonnement adoptées offrent de meilleures performances en termes de : (1) garantie de la couverture, (2) maintien d'un nombre élevé de noeuds vivants, (3) réduction du nombre de clusters et (4) efficacité énergétique ayant comme conséquence l'extension de la longévité du réseau.

Nos perspectives futures dans cette thématique sont :

- Examen du support de l'hétérogénéité par le protocole HPRC. En effet, un réseau de capteurs peut être composé de noeuds possédant des caractéristiques variées ; rayon de couverture, portée de communication et ressources énergétiques.
- Validation par méthodes mathématiques et analytiques telles que la modélisation par chaînes de Markov, files d'attente, les automates d'états finis, etc. Ces méthodes sont plus puissantes que la simulation car elles offrent des résultats exacts. En effet, le défi à relever par cette perspective est d'adapter les formules de calcul des métriques de performance au problème de la couverture.
- Traitement du problème de la couverture dans le cas de la mobilité des capteurs.
- La prise en considération du déploiement des capteurs dans un plan à trois dimensions.

## BIBLIOGRAPHIE

[1] : I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless sensor networks : a survey. Computer Networks, vol 38 (4), pp 393-422, 2002.

[2] : Ian F. Akyildiz, Weilian Su, Yogesh Sankarasubramaniam, and Erdal Cayirci. A Survey on Sensor Networks. IEEE Communications Magazine, Vol 40 (8), pp 102-114, 2002.

[3] : Hill, J. et al. System architecture directions for networked sensors . Proceeding ACM International Conference on Architectural Support for Programming Languages and Operating Systems, pp 93-104, 2000.

[4] : Noury Norbert. Du signal à l'information : le capteur intelligent Exemples industriels et en médecine. Thèse d'habilitation en Informatique, Laboratoire des Techniques de l'Imagerie, de la Modélisation et de la cognition, (TIMC, IMAG), université de Grenoble, 2002.

[5] : K. SOHRABY, D. MINOLI, T. ZNATI. Livre : " WIRELESS SENSOR NETWORKS Technology, Protocols, and Applications", 2007.

[6] : S. Mahfoudh, P. Minet. An energy efficient Routing based on OLSR in wireless Ad Hoc and sensor networks. 22nd International Conference on Advanced In-formation networking and Application, 2008.

[7] : Heinzelman WB, Chandrakasem AP, Balakrishnan H. An application-specific protocol architecture for wireless microsensor networks. IEEE Trans on wireless Communica-tions, vol 4(1) pp. 660-670, 2002.

[8] : X. Di. A Novel Coverage-Preserving Clustering Algorithm for Wireless Sensor Networks. International Conference on Medical Physics and Biomedical Engineering, pp. 1054-1059, 2012.

[9] : Nirupama Bulusu, John S. Heidemann, Deborah Estrin, Tommy Tran : Self-configuring localization systems : Design and Experimental Evaluation. ACM Trans. Embedded Comput. Syst. vol 3(1), pp 24-60, 2004.

[10] : V. Rodoplu, T.H. Meng. Minimum Energy Mobile Wireless Network, IEEE JSAC, vol 17(8), pp 1333-1344, 1999.

[11] : Karlof. C et al. TinySec : a link layer security architecture for wireless sensor networks. Proc. SenSys, 2004.

[12] : Nieberg, S. Dulman, P. Havinga T.Lodewijk, v. Hoesel, J. Wu, "Collaborative Algorithms for Communication in Wireless Sensor Networks". 2003.

[13] : W.Su and I.F Akyildiz .The Jitter Time Stamp Approach for Clock Recovery of Real-Time VBR Traffic, IEEE/ACM Transactions on Networking, vol 9 (6), pp.746-755, 2001.

[14] : Z. Shelby, Carlos A. Pomalaza Reiz, J. Haapola. Energy Optimisation in Multihop wireless embedded and sesor networks, PIMRC, 221-225, 2004.

[15] : A. Boukerche, I. Chatzigiannakis, S-E. Nikolettseas. Pwer-Efficient Data Propagation Protocols for Wireless Sensor Networks, Simulation, vol 81(6), pp. 399-411, 2005.

[16] : Andrews, P. Johnson and D.C. Remote continuous monitoring in the home. Telemedicine and Telecare, Vol 2 (2), pp. 107-113, 1996.

[17] : E.M. Petriu, N.D. Georganas, D.C. Petriu, D. Makrakis, and V.Z. Groza. Sensor-based information appliances. IEEE Instrumentation Measurement Magazine. Vol 3(4), pp. 31-35, 2000.

[18] : Michael Fitzgerald. Technnology Review, Tracking a Shopper's Habits, vol 19(4), pp. 475-493. 2008.

[19] : D. Gray et al. The nesC language : A holistic approach to networked embedded



systems, vol 26(2-3), pp. 203-216, 2003.

[20] : D.Moore et al. "Robust distributed network localization with noisy range measurements". Proc. SenSys, 2004.

[21] : Y. Shang et al. Improved MDS Based Localization. Proc. Infocom, 2004.

[22] : D.L. Millis. Internet Time Synchronization : The Network Time Protocol . IEEE trans. Communications, vol 39(10), pp. 1482-1493, 1991.

[23] : A. Wood et al. Denial of service in sensor networks . IEEE Computer, vol 35(10), pp. 54-62, 2002.

[24] : Antoine Gallais, François Ingelrest, Jean Carle, David Simplot-Ryl. Maintien de la couverture de surface dans les réseaux de capteurs avec une couche physique non idéale. Rapport scientifique, IRCICA/LIFL, Univ. Lille 1, CNRS UMR 8022, INRIA Futurs, POPS research group, FRANCE, page 3, 2006.

[25] : Yan Jin , Ju-Yeon Jo , Ling Wang , Yoohwan Kim , Xiaozong Yang. An energy-efficient coverage and connectivity preserving routing algorithm under border effects in wireless sensor networks. Computer communications, vol 31, 2398-2407, 2008.

[26] : S. Meguerdichian, F. Koushanfar, M. Potkonjak, M. Srivastava. Coverage problems in wireless ad-hoc sensor networks. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE, vol 3, pp. 1380-1387, 2001.

[27] : S. Goel, T. Imielinski. Prediction-based monitoring in sensor networks. ACM SIGCOM Computer Communication Review. Vol 31(5), pp. 82-98, 2001.

[28] : Bang WANG, Huazhong, University of Science and Technology. Coverage Problems in Sensor Networks. ACM Comput, vol 43(4), Article NO : 32, 2011.

[29] : A.MAKHOUL. Réseaux de capteurs : localisation, couverture et fusion de données. Thèse de doctorat en informatique, Université de Franche-Comté, 2008.

[30] : Bang WANG, Cheng Fu, Hock Beng Lim. Layered Diffusion-based Coverage Control in Wireless Sensor Networks. Computer Networks, vol 53 (17), pp. 1114-1124, 2009.

[31] : GUO Jian, SUN Lijuan, WANG Ruchuan, XIAO Fu. A low coverage first classifying algorithm for wireless sensor networks. *Procedia Engineering*, vol 15 (7), pp. 3825-3830, 2011.

[32] : S. Slijepcevic, M. Potkonjak. Power efficient organization of wireless sensor networks. *Proceeding of the IEEE International Conference on Communications, IEEE Communication Society*, vol 4(1), pp. 472-476, 2001

[33] : Namrata S. Baghele, K.N. Hande. A Coverage aware Clustering Scheme for power constrained Wireless Sensor Networks. *IJICT Journal*. vol 1(2) pp. 64-68, 2011.

[34] : Zhixin Liu, Qingchao Zheng, Liang Xu, Xinping Guan. A distributed energy-efficient clustering algorithm with improved coverage in wireless sensor networks. *Future Generation Computer Systems*, vol 28(5) : 167-739, 2011.

[35] : Bang WANG, Hock Beng LIM, Di MA. A coverage-aware clustering protocol for wireless sensor networks. *Computer Networks*, vol 56(5), pp. 1599-1611, 2012.

[36] : Mohammad Zeynali, Amir Mollanijad, Leili Mohammad Khanly. "Novel hierarchical routing protocol in wireless sensor networks" : *Elsevier Ad-Hoc Networks Journal*, vol 3(2) pp. 292-301, 2010.

[37] : M-E. Khanouche, F-S. Ouada, S. Ouguigui. Energy-aware Clustering Sensor Scheduling coverage Maintenance. *IEEE International Conference on Green Computing and Communications, Conference on Internet of Things, and Conference on Cyber, Physical and Social Computing*, pp. 171-178, 2012.

[38] : Xu BAO, Yongfung JU. A Coverage-Garanteed Nodes Scheduling Algorithm in clusters for Wireless sensor Networks. *IEEE Trans communications*, pp. 2265-2268, 2012.

[39] : X. WANG, J. MA, S. Wang. Parallel energy-efficient coverage optimization with maximum entropy clustering in wireless sensor networks. *Journal of parallel and distributed computing*, vol 69(3), pp. 838-847, 2009.

[40] : X. Wang, G. Xing, Y. Zhang, C. Lu, R. Pless, and C. Gill, Integrated coverage and connectivity configuration for energy conservation in sensor networks, *ACM Trans. Sensor Networks*, vol 1(1), pp. 36-72, Aug. 2005.

# Résumé

Bien que la problématique de la couverture dans les réseaux de capteurs sans fil soit largement abordée, les exigences des applications (en matière de fiabilité) et la contrainte d'énergie laissent cette problématique à la recherche d'une résolution optimale. Toutefois, les avancées récentes dans cet axe font recours à la combinaison de deux techniques : le clustering et l'ordonnancement d'activité, ceci pour une meilleure extension de la longévité du réseau. Notre contribution baptisée **HPRC** (pour **H**ierarchical **P**rotocol based-on **R**edundancy evaluation for **C**overage garantie) est une solution distribuée tirant profits (réduire la redondance et minimiser le trafic circulant dans le réseau) des deux techniques précitées. Dans **HPRC**, la politique adoptée pour l'élection des cluster-heads (CH) utilise un poids calculé en fonction de l'énergie, le degré et la distance moyenne par rapport au voisinage du capteur. Les noeuds non-élus optent pour une évaluation exacte de leur redondance afin de devenir membres à un saut (CM-1S), membres à deux sauts (CM-2S) ou noeuds passifs.

Les scénarios de simulation réalisés pour comparer les performances de **HPRC** avec celles d'un autre protocole récent, montrent une couverture quasi-complète de la zone cible, un nombre élevé de noeuds vivants et une consommation énergétique réduite, ce qui prolonge la longévité du réseau.

**Mots clés :** Réseaux de capteurs sans fil, Couverture, Clustering, Ordonnancement d'activité, Évaluation de la redondance, Optimisation d'énergie.

# Abstract

Although the issue of coverage in wireless sensor networks is widely discussed, application requirements (in reliability) and energy constraint leave this issue to the search for an optimal resolution. However, recent advances in this axis are using a combination of two techniques : clustering and activity scheduling in order to extend the network lifetime. Our contribution called HPRC (for Hierarchical Protocol based-on Redundancy evaluation for Coverage guarantee) is a distributed solution pulling profits (reduce redundancy and minimize the traffic flowing through the network) of the two aforementioned techniques. In the policy adopted for the election of cluster-heads (CH), HPRC uses a weight calculated by using energy, the degree and the average distance from neighboring of the node. The non-elected nodes evaluate their redundancy to join a cluster-head as one-hop Members (CM-1S), two-hop Members (CM-2S) or decide to be passive.

Simulation scenarios performed to compare the performance of HPRC with those of an other recent protocol. This comparison shows an almost complete coverage of the target area, a large number of alive nodes and reduced energy consumption, thereby extending the network lifetime.

**Keywords :** coverage, clustering, activity scheduling, redundancy evaluation, energy optimization.