

MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE SCIENTIFIQUE
UNIVERSITÉ MOULOUE MAMMÉRI DE TIZI-OUZOU
FACULTÉ DE GENIE ÉLECTRIQUE ET D'INFORMATIQUE
DÉPARTEMENT D'INFORMATIQUE



*En vue de l'obtention du diplôme de MASTER 2 en INFORMATIQUE spécialités
RESEAUX MOBILITES ET SYSTEMES EMBARQUES*

MEMOIRE DE FIN D'ETUDES

Thème

Conception d'un algorithme de clustering pour les Réseaux de capteurs sans fil

Proposé et Dirigé Par :
M^r TALBI Said

Réalisé Par :
CHABANE Hocine
TICILIA Kahina

Année Universitaire : 2012 / 2013

Remerciements

Nous tenons à remercier en premier lieu le TOUT PUISSANT DIEU qui nous a donné la volonté nécessaire, la force et la bonne santé, la patience et le courage de mener à bon terme ce mémoire de fin d'études.

Nous exprimons notre gratitude à notre promoteur Monsieur Talbi Said qui nous a fait l'honneur de diriger ce travail. Nous le remercions pour sa disponibilité et ses conseils qui nous ont été d'un apport considérable tant sur l'aspect scientifique que sur la méthode de recherche. Sans son appui et son expérience, rien n'aurait été possible,

Nos vifs remerciements vont également aux membres du jury, pour l'intérêt qu'ils ont porté à notre recherche en acceptant d'examiner notre travail et de l'enrichir par leurs propositions.

Enfin, nous remercions tous ceux qui ont contribué de près ou de loin à la réalisation de ce travail.

Résumé

Les réseaux de capteurs sans fils, l'une des technologies marquantes du 20ème siècle, composés de centaines voir de milliers de capteurs sans fils minuscules à capacités réduites qui communiquent à l'aide du sans fils, déployés dans la zone du phénomène à surveiller ou prêt de celui-ci. Au début ils furent utilisés dans des applications militaires, mais l'efficacité ainsi que les services assurés par ces derniers, les ont favorisé à être appliqués dans d'autres domaines tels que la surveillance de l'environnement, le suivi des malades, la détection des incendies, la prévention des catastrophes naturelles et divers autres domaines. Vu leurs capacités limitées, de plus qu'ils sont dotés d'une ressource énergétique très réduite, les nœuds d'un réseau de capteurs travaillent en collaborations, après avoir détecté des grandeurs, ils collaborent afin de les transmettre vers un utilisateur final à savoir la station de base. Le principal défi d'un tel réseau et sa capacité d'atteindre les objectifs cernés par l'application toute en minimisant l'énergie consommée.

L'objectif de ce travail, est de structurer le réseau afin de minimiser l'énergie consommée par les protocoles de routage. Pour cela nous avons procédé par la création d'une structure hiérarchique basée sur la technique du clustering qui consiste à former des groupes de nœuds gérés par un leader ayant des capacités le favorisant à accomplir ce rôle

Mot clefs: WNS, Cluster He ad, auto-organisation, clustering, Omnet++

Table des matières

| | |
|--|-----------|
| Introduction générale..... | 1 |
| Chapitre 1 Généralités sur les réseaux de capteurs..... | 2 |
| 1.1 Introduction..... | 3 |
| 1.2 Réseaux sans fil..... | 3 |
| 1.2.1 Définition et Caractéristiques..... | 3 |
| 1.2.2 Classification des réseaux sans fil (étendu / infrast)..... | 3 |
| 1.3 Réseaux de capteurs sans fil..... | 7 |
| 1.3.1 Architecture d'un WSN..... | 8 |
| 1.3.2 Architecture d'un capteur | 10 |
| 1.3.3 Communication dans WSNs (Pile Protocolaire) | 13 |
| 1.3.4 Caractéristiques des réseaux de capteurs..... | 16 |
| 1.3.5 WSNs VS Ad Hoc | 18 |
| 1.3.6 Consommation de l'énergie dans les WSNs..... | 18 |
| 1.3.7 Axes de recherche | 21 |
| 1.3.8 Application..... | 25 |
| 1.3.8 Autres | 29 |
| 1.4 Conclusion | 30 |
| Chapitre 2 Auto-organisation dans les WSNs..... | 31 |
| 2.1 Objectif | 32 |
| 2.2 Auto-organisation basée sur la mise en plan CDS..... | 32 |
| 2.2.1 CDS-regle k (Connecting Dominating Set regle k) | 32 |
| 2.2.2 CDS-IDS (Connected Dominating Set-Infependent Dominating Set) | 33 |
| 2.2.3 Legos..... | 34 |
| 2.3 Auto-organisation basée sur l'élagage de lien | 35 |
| 2.3.1 Relative Neighborhood RNG | 35 |
| 2.3.2 LMST | 36 |
| 2.4 Auto-organisation en Cluster | 37 |
| 2.4.1 Low Energy Adaptive Clustering Hierarchy (LEACH) | 37 |
| 2.4.2 Hybrid Energy-Efficient Distributed Clustering (HEED) | 39 |
| 2.4.3 Power-Efficient GATHERing in Sensor Information System (PEGASIS) | 40 |

| | |
|--|-----------|
| 2.4.4 Far-Legos (<i>Far Low Energy Self-OrGanisatiOn Scheme</i>) | 41 |
| 2.5 Conclusion | 42 |
| Chapitre 3 Vers une Nouvelle Approche d'Auto-Organisation..... | 43 |
| 3.1 Objectif | 44 |
| 3.2 Déroulement du protocole Leach et ces limite | 44 |
| 3.2.1 Présentation | 44 |
| 3.2.2 Pourquoi se baser sur LEACH | 46 |
| 3.2.3 Itérations de l'algorithme | 47 |
| 3.2.4 Phase d'élection des Cluster-Heads | 47 |
| 3.2.5 Phase de formation des clusters | 48 |
| 3.2.6 Ordonnancement intra-cluster..... | 48 |
| 3.2.7 Transmission de données | 48 |
| 3.2.8 Limite du protocole Leach | 48 |
| 3.3 EDBUC (Energy and Density Based Unequal Cluster)..... | 49 |
| 3.3.1 Hypothèses | 49 |
| 3.3.2 Modèle d'énergie utilisée..... | 50 |
| 3.3.3 Itérations de l'algorithme EDBUC..... | 50 |
| 3.4 Diagramme de l'algorithme EDBUC | 54 |
| 3.5 Conclusion | 56 |
| Chapitre 4 Simulation | 57 |
| 4.1 Introduction..... | 58 |
| 4.2 Etude de quelques simulateurs existants..... | 58 |
| 4.2.1 TOSSIM | 58 |
| 4.2.2 NS2..... | 60 |
| 4.2.3 GloMoSim | 61 |
| 4.2.4 OMNeT++ Framework | 63 |
| 4.3 Environnement de simulation choisi OMNeT++..... | 64 |
| 4.3.1 Présentation | 64 |
| 4.3.2 Architecture..... | 64 |
| 4.3.3 La librairie MF | 66 |
| 4.3.4 La librairie INET..... | 66 |
| 4.3.5 La couche PHY | 66 |
| 4.3.6 La couche MAC | 66 |
| 4.3.7 La couche IP | 66 |

| | |
|-------------------------------------|-----------|
| 4.3.8 La couche RTP..... | 67 |
| 4.3.9 La couche Application..... | 67 |
| 4.4 Résultats des simulations..... | 67 |
| 4.4.1 Critère de performances | 67 |
| 4.4.2 Scénario de simulation | 68 |
| 4.4.3 Discussion des résultats..... | 69 |
| 4.5 Conclusion | 72 |
| Conclusion générale | 74 |
| Référence..... | 75 |
| Webliographie | 78 |

Liste des figures

Figure 1.1 Classification des réseaux sans fil selon l'étendu géographique.

Figure 1.2 Réseau avec infrastructure.

Figure 1.3 Réseau sans fil sans infrastructure.

Figure 1.4 Acheminement des données d'une source vers le nœud puits et l'utilisateur.

Figure 1.5 Diffusion d'une requête par le nœud puits.

Figure 1.6 Architecture plate d'un WSN.

Figure 1.7 Structures hiérarchiques d'un WSN.

Figure 1.8 Architecture physique d'un capteur.

Figure 1.9 La pile protocolaire des réseaux de capteurs.

Figure 1.10 Le modèle de consommation d'énergie pour la communication.

Figure 1.11 La consommation d'énergie en capture, traitement et communication.

Figure 1.12 Exemples de nœuds capteurs.

Figure 1.13 Application militaire des réseaux de capteurs.

Figure 1.14 Surveillance d'activités des personnes âgées.

Figure 1.15 Application des RCSF dans la surveillance de l'environnement des oiseaux marins.

Figure 2.1 Principe du protocole CDS-règle k.

Figure 2.2 Exemple d'un IDS.

Figure 2.3 Topologie Legos.

Figure 2.4 Elagage des liens avec l'algorithme RNG.

Figure 2.5 Topologie basée sur LEACH.

Figure 2.6 Exemple d'affectation de 4 rangs aux capteurs autour d'un actionneur.

Figure 3.1 cluster-heads formés à l'instant t

Figure 3.2 cluster-heads formés à l'instant t'

Figure 3.3 Diagramme représentant la construction de cluster.

Figure 4.1 L'architecture de GloMosSim.

Figure 4.2 Modules composés, modules simples et les connexions entre les modules conception en NED.

Figure 4.3 Moyenne d'énergie consommée par les nœuds.

Figure 4.4 L'écart type de l'énergie consommée.

Figure 4.5 La durée de vie du réseau pour 25 nœuds.

Liste des Tableaux

Tableau 1.1 WSN vs Ad Hoc.

Tableau 3.1 Structure d'un message Hello

Tableau 3.2 Structure d'un message WeightMsg.

Tableau 3.3 Structure d'une table de voisinage.

Tableau 3.4 Structure d'un message CHadv.

Tableau 3.5 Structure d'un message JoinClusterDecision.

Tableau 4.1 Tableau qui résume une comparaison entre LEACH et EBUC.

Tableau 4.2 Pourcentage de nœuds en vie pour une topologie de 25 nœuds.

Introduction générale

L'apparition des communications sans fil accessibles sur des équipements mobiles (portable, PDA, laptops...), l'évolution des dispositifs de calcul et les progrès dans l'infrastructure de communication ont abouti à la croissance rapide des réseaux sans fil. Ceux-ci sont géographiquement étendus (GSM, Wimax), locaux (802.11, Zigbee) ou personnels (Bluetooth). On trouve également les NFC (Near Field Communication) utilisés pour de nouveaux services (paiement des transports, affichage d'informations contextuelles...) et des RFID qui permettent d'optimiser le fonctionnement des processus internes de l'entreprise, comme la logistique et la production par exemple. Dans les réseaux cellulaires comme le GSM basés sur les deux modes de communication filaire et sans fils, chaque antenne couvre une zone géographique, et quand l'utilisateur se déplace, son téléphone mobile change de cellule. Ainsi on dit que ces réseaux sont structurés suivant une infrastructure définie. Un autre groupe de réseaux sont sans infrastructure, les entités le constituant doivent être capables de s'auto-organiser, c'est le cas des réseaux Ad Hoc (MANet) et les réseaux de capteurs sans fils (WSN) [BEY09]. Le grand essor des progrès réalisés ces dernières décennies dans les domaines de la microélectronique, micromécanique et l'apparition de nœud senseurs petits et bon marché, est l'implosion des WSN (Wireless Sensor Network ou réseaux de capteurs sans fils) qui se composent d'un grand nombre de dispositifs de détection, traitement et d'acheminement.

Les WSNs ont suscité l'attention énorme du milieu universitaire et de l'industrie partout dans le monde, un grand nombre de recherche ont été effectués pour explorer et résoudre de divers problèmes de conception et d'application.

Ce qui distingue les réseaux de senseurs des réseaux traditionnels (système cellulaire et mobile, réseaux ad hoc) est le niveau plus dense de déploiement de nœuds, manque de fiabilité, contrainte de stockage, d'énergie et de calcul.

Ces contraintes ont présenté de nouveaux défis dans le développement et le déploiement des WSN.

On envisage que dans un avenir proche, les WSNs seront employés couramment dans divers domaines et révolutionneront notre vie. [AME 10]

Chapitre 1

Généralités sur les réseaux de capteurs

1.1 Introduction

Les réseaux de capteurs sans fil, sont considérés comme le plus grand essor des avancements technologiques du 20ème siècle. Ces réseaux utilisés essentiellement pour surveiller des zones inaccessibles par l'être humain, regroupent des entités minuscules intelligentes capables de détecter, traiter et communiquer des données. Initialement, ils furent utilisés dans le domaine militaire, par la suite, leurs sucées a vite envahie d'autres domaines tels que la nature, l'industrie, les milieux publics ou encore des domaines beaucoup plus critiques à savoir la médecine.

Le plus grand défi de ces réseaux est l'absence d'une infrastructure centralisée telle que dans les réseaux cellulaire (GSM). Les entités le constituant doivent être en mesure de s'auto-organiser afin d'accomplir les taches principales (routage). Les réseaux de capteurs sans fils ou WSNs étaient et restent une voix de recherche ouverte, notamment le routage visé par la majorité des travaux de recherche.

Dans ce présent chapitre, nous allons initier par une définition générale des réseaux sans fils, afin de pouvoir définir les WSNs et donner leurs caractéristiques matérielles, les standards appropriés, et les différents cas d'utilisation. Nous allons finir par citer quelques voies de recherches ouvertes dans ce domaine.

1.2 Réseaux sans fil

1.2.1 Définition et Caractéristiques

Un réseau sans fils, est un réseau informatique qui interconnecte les différents équipements entre eux par des ondes radios. La norme la plus utilisée actuellement pour ces réseaux est la norme IEEE 802.11. [BEY09]

Dans le réseau sans infrastructure, il n'y a pas lieu d'une infrastructure préexistante (point d'accès par exemple) pour gérer le réseau : chaque membre du réseau peut recevoir et envoyer ses informations mais il agit aussi comme routeur pour transférer les données vers d'autres nœuds du réseau. Ce type de réseau est appelé réseau Ad-hoc.

1.2.2 Classification des réseaux sans fil (étendu / infrast)

Les réseaux sans fil se classifient généralement suivant l'étendu géographique, et le mode avec ou sans infrastructure :

1.2.2.1 Classification selon l'étendu géographique

Quatre catégories principales sont distinguées, suivant la portée et le taux de transmission radio disponibles. Les applications varient souvent selon le type de réseau, c'est pourquoi chaque type de réseau repose sur des normes de communications différentes :

– **Les WPAN** (*Wireless Personal Area Network*)

Les réseaux sans fil personnels ou Wireless Personal Area Network (WPAN), sont des réseaux sans fil à très faible portée (de l'ordre d'une dizaine de mètres). Ils sont le plus souvent utilisés dans le cadre du Wearable Computing ou informatique vestimentaire qui consiste à faire communiquer entre eux des matériels présents sur une personne (par exemple une oreillette et un téléphone portable). Ils sont également utilisés pour relier des équipements informatiques entre eux : par exemple pour relier une imprimante ou un PDA (Personal Digital Assistant ou assistant personnel) à un ordinateur de bureau.

Pour mettre en œuvre de tels réseaux, la principale technologie est IEEE 802.15.1 ou Bluetooth. Elle fut proposée par Ericsson en 1994 et fournit un taux de transmission radio théorique de 1 Mbit/s pour une portée maximale d'une trentaine de mètres. Elle est très peu gourmande en énergie ce qui la rend particulièrement intéressante pour être intégrée dans de petits équipements autonomes sans fils comme les PDA. La technologie infrarouge ou IrDA est également utilisée dans ce type de réseaux. Cette technologie est cependant beaucoup plus sensible que Bluetooth aux perturbations lumineuses et nécessite une vision directe entre les éléments souhaitant communiquer ce qui la limite bien souvent à un usage de type télécommande.

– **WLAN** (*Wireless Local Area Network*)

Les réseaux locaux sans fils ou Wireless Local Area Network (WLAN) sont généralement utilisés à l'intérieur d'entreprises, d'universités mais également chez les particuliers depuis le développement des réseaux à haut débit. Ces réseaux sont principalement basés sur la technologie IEEE 802.11 soutenue par le WECA (Wireless Ethernet Compatibility Alliance) ou sur la technologie HiperLan1 et son remplaçant Hiperlan 2 soutenue par l'ETSI (*European Telecommunications Standards Institute*). Ils offrent des taux de transmission radio théoriques pouvant atteindre 54 Mbit/s pour IEEE 802.11g ou HiperLan2 c'est-à-dire bien plus élevés que la technologie Bluetooth. La portée de ces technologies est également beaucoup plus importante: de l'ordre de 300 mètres en extérieur et

100 mètres à l'intérieur de bâtiments pour IEEE 802.11b et d'une centaine de mètres pour Hiperlan2.

– **WMAN** (*Wireless Metropolitan Area Network*)

Les réseaux métropolitains sans fils ou *Wireless Metropolitan Area Network* (WMAN) également appelés boucle locale radio (BLR) étaient à l'origine prévus pour interconnecter des zones géographiques difficiles d'accès à l'aide d'un réseau sans fils. Actuellement ces réseaux sont utilisés dans certaines villes américaines (San Francisco) pour fournir un accès Internet aux habitants. Les réseaux basés sur la technologie IEEE 802.16 ont une portée de l'ordre de quelques dizaines de kilomètres (50km de portée théorique annoncée) et un taux de transmission radio théorique pouvant atteindre 74 Mbit/s pour IEEE 802.16-2004 plus connue sous le nom commercial de WiMAX. C'est également dans cette catégorie que peuvent être classés les réseaux téléphoniques de troisième génération utilisant la norme UMTS (*Universal Mobile Telecommunication System*) pour transmettre de la voix et des données. Cette norme UMTS propose des taux de transmission radio théoriques pouvant aller jusqu'à 2 Mbit/s sur des distances de plusieurs kilomètres.

– **WWAN** (*Wireless Wide Area Network*)

Les réseaux sans fils étendus ou *Wireless Wide Area Network* (WWAN) regroupent notamment les différents réseaux téléphoniques de première et deuxième génération mais également les réseaux satellitaires. Les réseaux cellulaires téléphoniques reposent sur des technologies comme GSM (*Global System for Mobile Communication*), GPRS (*General Packet Radio Service*). Les réseaux satellites s'appuient quant à eux sur les normes comme DVB-S (*Digital Video Broadcasting-Satellite*) pour transmettre l'information et proposent des débits élevés (de l'ordre de 40 Mbit/s pour la norme DVB-S). Ces réseaux se posent en concurrents directs des réseaux filaires traditionnels en proposant des services et une qualité de services de plus en plus proches de celle que l'on retrouve dans les réseaux filaires. Ils permettent en particulier de toucher des zones totalement enclavées dans lesquelles les réseaux de types WMAN ne peuvent être utilisés comme les déserts ou les zones montagneuses dépourvues d'infrastructures terrestres.

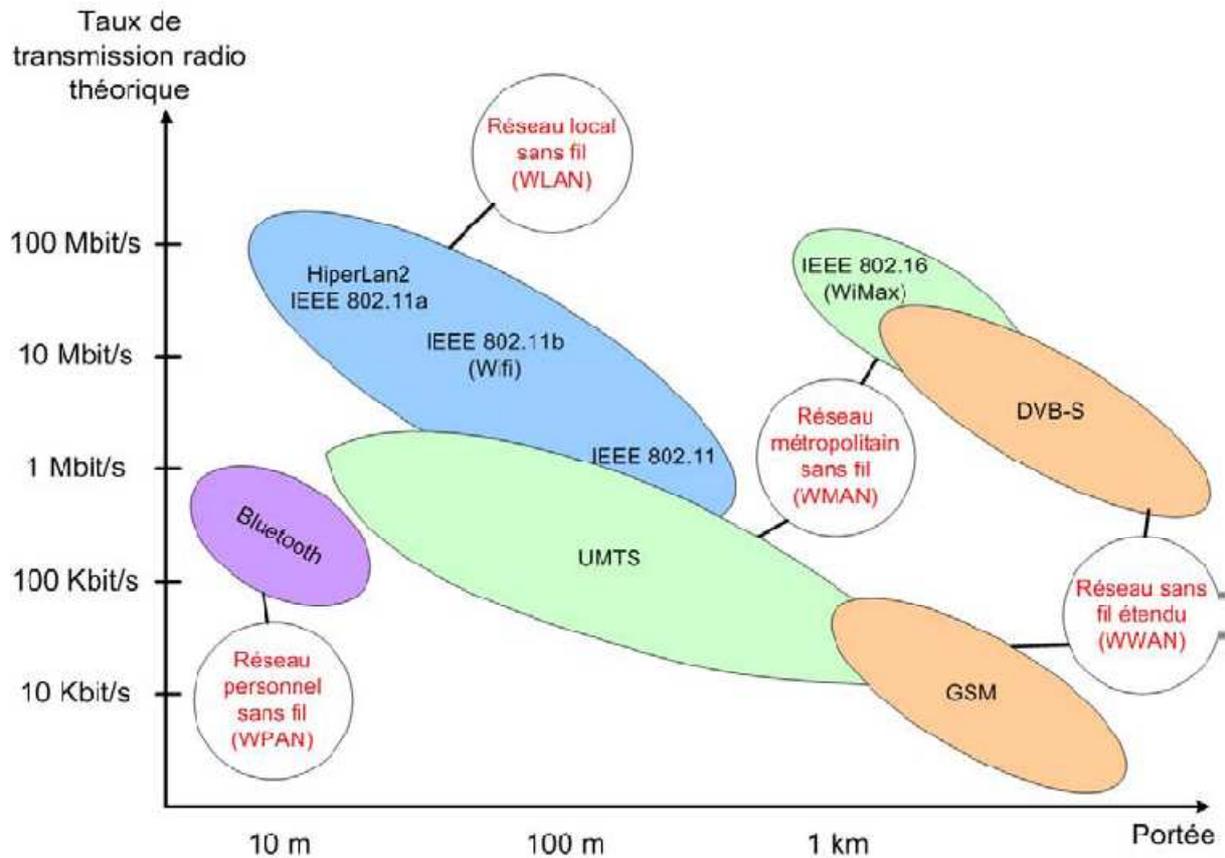


Figure 1.1 Classification des réseaux sans fil selon l'étendu géographique.

1.2.2.2 Classification selon l'infrastructure mise en place

– Réseau avec infrastructure

Le mode infrastructure est défini pour fournir aux différentes stations des services spécifiques sur une zone de couverture déterminée par la taille du réseau. Les réseaux d'infrastructure sont établis en utilisant des points d'accès, ou AP (*Access Point*), qui jouent le rôle de station de base pour une cellule de base, appelée BSS (*Basic Set Service*). Un point d'accès sur un réseau sans fil équivaut à un concentrateur (hub) sur un réseau filaire. Chaque terminal sans fil reçoit donc tout le trafic circulant sur le réseau. Lorsque le réseau est composé de plusieurs BSS, chacun d'eux est relié à un système de distribution, ou DS (*Distribution System*), par l'intermédiaire de leur point d'accès (AP) respectif. Un groupe de BSS interconnectés par un système de distribution forme un ESS (*Extended Set Service*). Le système de distribution est responsable du transfert des paquets entre différents BSS d'un même ESS. Les points d'accès peuvent être reliés entre eux par des liaisons radio ou filaires et un terminal peut alors passer d'un point d'accès à un autre en restant sur le même réseau

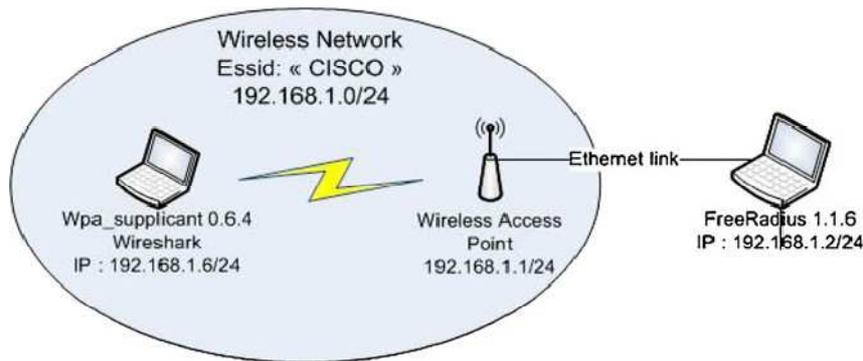


Figure1.2 Réseau avec infrastructure.

– Réseaux sans infrastructure

Dans le mode de fonctionnement sans infrastructure, un groupe de terminaux s'associe pour former un IBSS (*Independent Basic Set Service*), dont le rôle consiste à permettre aux stations de communiquer sans l'aide d'une quelconque infrastructure. Chaque station peut établir une communication avec n'importe quelle autre station dans l'IBSS sans être obligée de passer par un point d'accès. Les réseaux ad hoc et les réseaux de capteurs sans fils font partie des réseaux fonctionnant en mode sans infrastructure. De tels réseaux ont pour particularité de s'auto créer, s'auto organiser et s'auto administrer

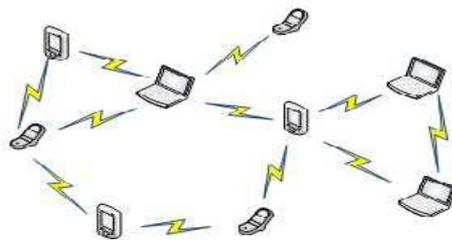


Figure1.3 Réseau sans fil sans infrastructure.

1.3 Réseaux de capteurs sans fil

Un réseau de capteurs sans fils (WSN) est un type particulier des réseaux Ad Hoc, il est composé d'un grand nombre de nœuds capteurs dispersés aléatoirement (souvent par un avion ou un hélicoptère) dans une zone géographique dite *champs de captage*. Ces nœuds qui jouent, à la fois, le rôle des hôtes et des routeurs, coopèrent à l'observation d'un phénomène, au captage des données et leurs transmissions vers un autre nœud dit nœud puits (*sink node*). Un nœud puits récolte les données captées afin de les traiter et les transmettre via satellite ou

Internet vers un utilisateur, comme il diffuse des requêtes aux capteurs sur le type de données requises.

Dans un réseau de capteurs sans fil, l'infrastructure fixe et l'administration centralisée sont absentes, il est donc nécessaire de prévoir des algorithmes d'auto organisation, de plus, les nœuds d'un réseau de capteurs doivent être assez durs pour résister aux conditions environnementales (par exemple les feux) et consommer le moins d'énergie pour prolonger la durée de vie du réseau ; en effet, si dans un WSN le nombre de connexions perdues dues au dysfonctionnement de ses capteurs est important le réseau devient inutile, il est donc nécessaire que la batterie d'un capteur dure le plus longtemps possible, étant donné que dans la plupart des applications, il est impossible d'intervenir pour son remplacement. [KHE08] [ABA12] [BEC09]

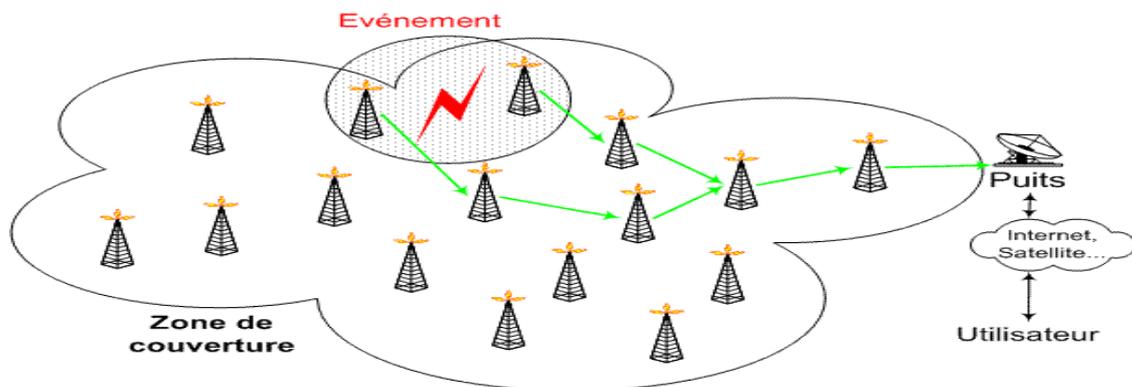


Figure 1.4 Acheminement des données d'une source vers le nœud puits et l'utilisateur.

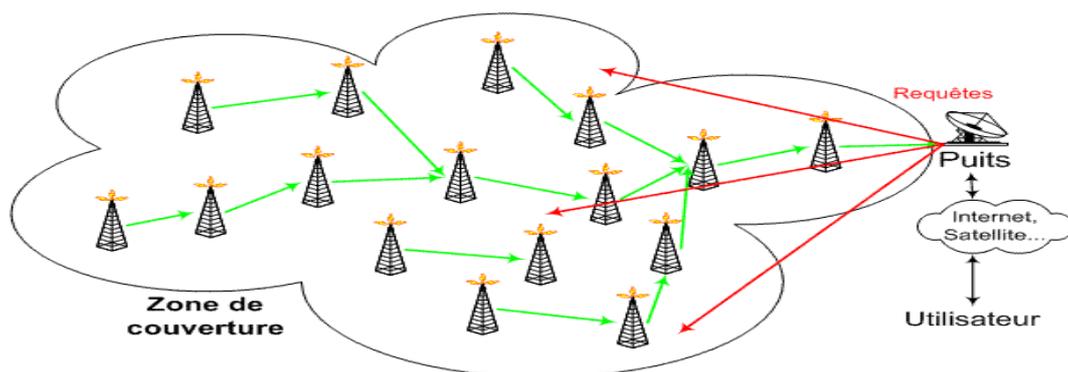


Figure 1.5 Diffusion d'une requête par le nœud puits.

1.3.1 Architecture d'un WSN

Selon le rôle joué par chaque nœud dans les différentes fonctions du réseau (détection d'événement, routage, agrégation de données...), on distingue deux structures :

1.3.1.1 Structure Plate

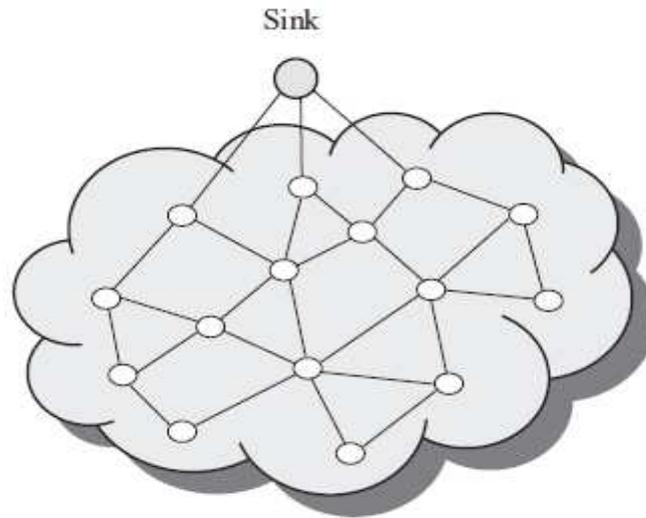


Figure 1.6 Architecture plate d'un WSN.

Comme le montre la figure 1.6, dans une architecture plate, tous les nœuds jouent le même rôle et coopèrent de la même façon dans la fonction du routage. Cette approche est simple et facile à mettre en œuvre, cependant elle nécessite l'échange d'un grand nombre de messages de contrôle, afin de garder le réseau actif à tous moments, de plus, les nœuds proches du nœud sink vont participer plus que les autres nœuds du réseau, ce qui va induire un déséquilibre sur la quantité de l'énergie résiduelle au sein du réseau.

Cette façon d'organiser un réseau donne des résultats seulement lorsque il s'agit d'un WSN de petite taille, cependant, elle reste inadéquate dans le cas des réseaux volumineux.

1.3.1.2 Structure Hiérarchique

On les appelle des réseaux auto-organisable, cette auto-organisation consiste à diviser le réseau virtuellement en un ensemble de nœuds géographiquement proches. Chaque ensemble de nœuds ou également nommé cluster est mené d'un chef de groupe (cluster-head), responsable sur l'acquisition des données captés par les membres, leurs agrégation, ensuite les envoyer vers le nœud sink (figure 1.7), un cluster-head sert aussi comme relais pour d'autre cluster-heads.

Cette méthode présente alors une solution intéressante pour simplifier et optimiser les fonctions et les services du réseau. En particulier, elle permet un routage plus efficace en réduisant le trafic de contrôle dans le réseau et en simplifiant le processus d'aiguillage des

données. Plusieurs Algorithmes de clustering ont été proposés et évalués dans la littérature. Ces algorithmes ont des caractéristiques différentes et sont conçus pour satisfaire certains objectifs selon le contexte pour lequel les clusters doivent être formés (routage, sécurité, conservation de l'énergie, etc.).

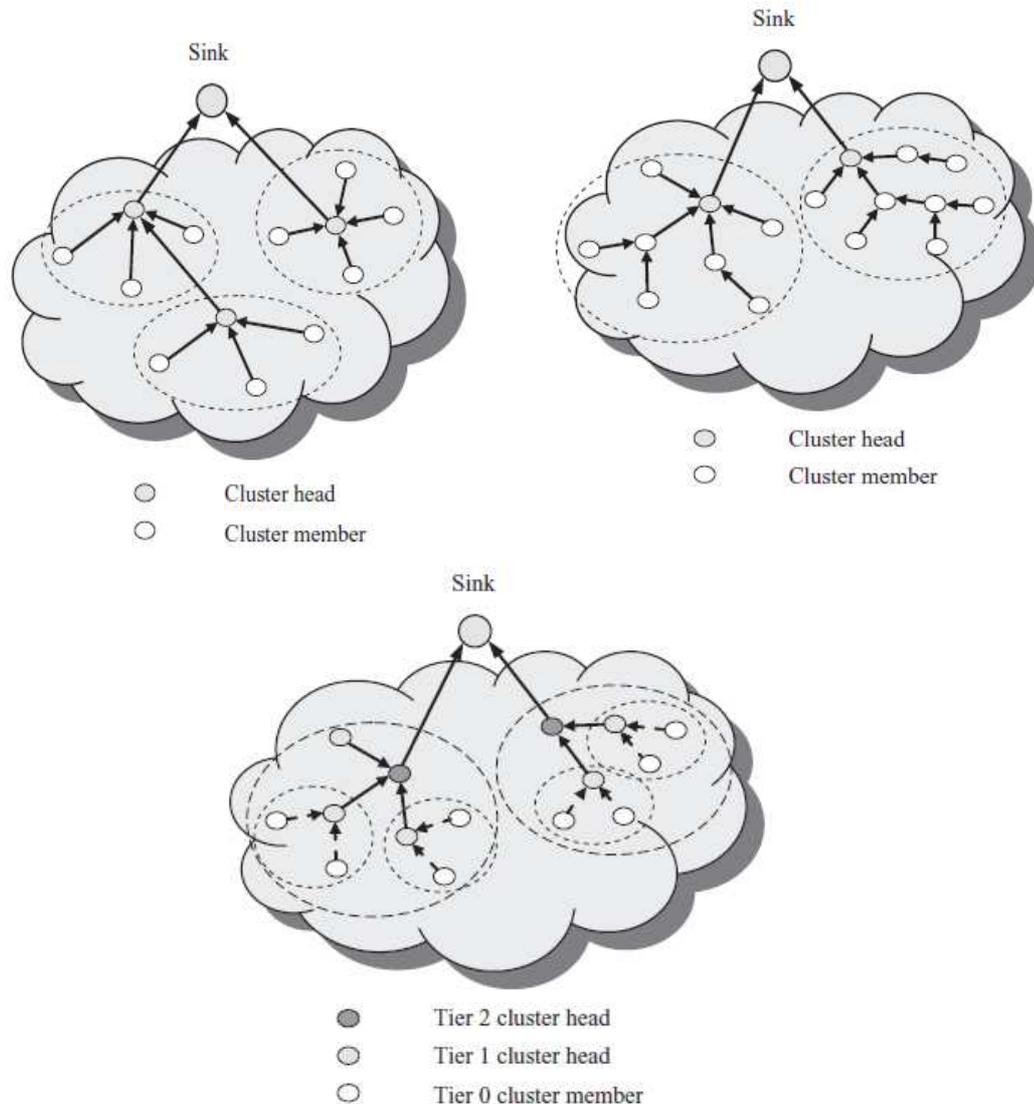


Figure 1.7 Structures hiérarchiques d'un WSN.

1.3.2 Architecture d'un capteur

Un nœud capteur contient quatre unités de base : l'unité de captage, l'unité de traitement, l'unité de transmission, et l'unité de contrôle d'énergie. Il peut contenir également, suivant son domaine d'application, des modules supplémentaires tels qu'un système de localisation (GPS), ou bien un système générateur d'énergie (cellule solaire). On peut même trouver des micro-capteurs, un peu plus volumineux, dotés d'un système mobilisateur chargé de déplacer le micro-capteur en cas de nécessité [KHE08] [CHA08] [KAC09].

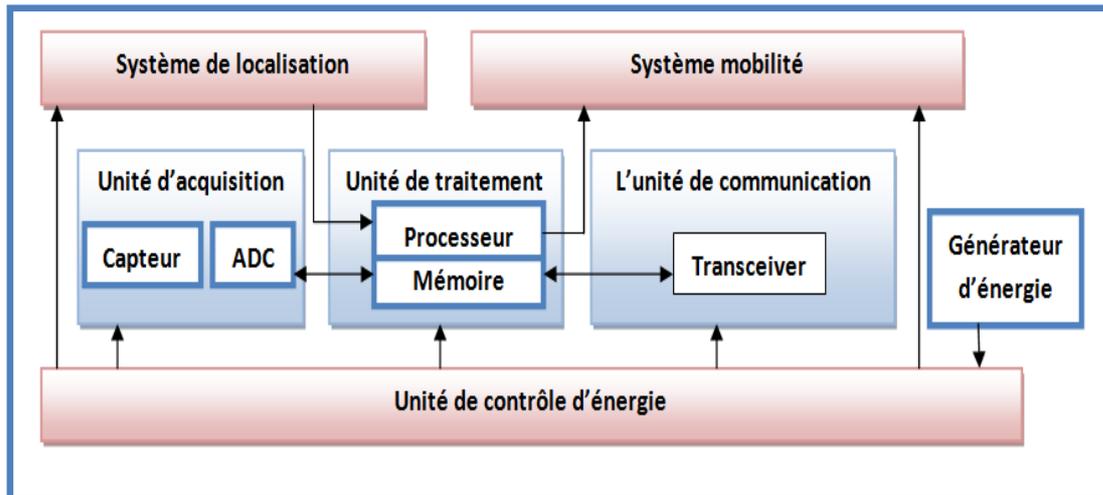


Figure 1.8 Architecture physique d'un capteur [KHE 08].

– Unité de traitement

L'unité de traitement est composée de deux interfaces, une interface pour l'unité d'acquisition et une deuxième pour l'unité de transmission. Cette unité est également composée d'un processeur et d'un système d'exploitation spécifique. Elle acquiert les informations en provenance de l'unité d'acquisition et les envoie à l'unité de transmission.

– Unité de transmission

Elle effectue toutes les émissions et réceptions des données sur un médium sans fil. Trois médiums de communication sans fil peuvent être utilisés: optique (LASER), infrarouge et radiofréquences (RF). Le laser et l'infrarouge n'utilisent aucune antenne mais leurs capacités de diffusion sont limitées, ces médiums de communication sont robustes vis-à-vis des interférences électriques. Néanmoins, ne pouvant pas établir de liaisons à travers des obstacles, elles présentent l'inconvénient d'exiger une ligne de vue permanente entre les entités communicantes. Les unités de transmission de type radio-fréquence (RF) comprennent des circuits de modulation, démodulation, filtrage et multiplexage; ceci implique une augmentation de la complexité et du coût de production du micro-capteur.

La RF nécessite une antenne et une portée radio en fonction de l'énergie consommée. Elle reste le moyen le plus répandu pour la communication des capteurs.

Le système de transmission des réseaux de capteurs consomme environ 20 mW et possède une portée de quelques dizaines de mètres. Pour augmenter ces distances tout en préservant l'énergie, le réseau utilise un routage multi-sauts [KHE08] [BEC09].

– **Unité de captage (LED)**

Le capteur est généralement composé de deux sous-unités, le récepteur qui fournit des signaux analogiques, basés sur le phénomène observé, au convertisseur (transducteur) Analogique/Numérique. Ce dernier transforme ces signaux en un signal numérique compréhensible par l'unité de traitement.

– **Unité de control d'énergie**

Un capteur est muni d'une ressource énergétique pour alimenter tous ses composants. Cependant, en conséquence de sa taille réduite, la ressource énergétique dont il dispose est limitée et généralement irremplaçable. Cette unité peut aussi gérer des systèmes de rechargement d'énergie à partir de l'environnement observé telles que les cellules solaires, afin d'étendre la durée de vie totale du réseau.

– **Système de localisation géographique**

Certaines applications WSNs, nécessitent de connaître la situation géographique du nœud capteur, afin d'accomplir les tâches de détection et de routage, pour cela, un nœud doit être équipé d'un système de localisation géographique. Ce système peut se composer d'un module GPS pour un nœud du haut niveau, ou bien d'un algorithme de localisation qui fournit des informations sur l'emplacement du nœud en effectuant des calculs distribués.

– **Un mobilisateur**

Cette unité sert à déplacer le nœud lorsque c'est nécessaire. Cependant le déplacement d'un nœud implique une haute consommation en énergie, le mobilisateur exige donc des ressources énergétiques étendues.

– **Le régénérateur de l'énergie**

Un nœud capteur peut être équipé d'un générateur électrique supplémentaire pour des applications où une plus longue vie du réseau est essentielle. Pour des applications extérieures, des piles solaires sont utilisées pour générer l'alimentation électrique. De même, des techniques de récupération d'énergie pour l'énergie thermique ou cinétique peuvent également être utilisées.

1.3.3 Communication dans WSNs (Pile Protocolaire)

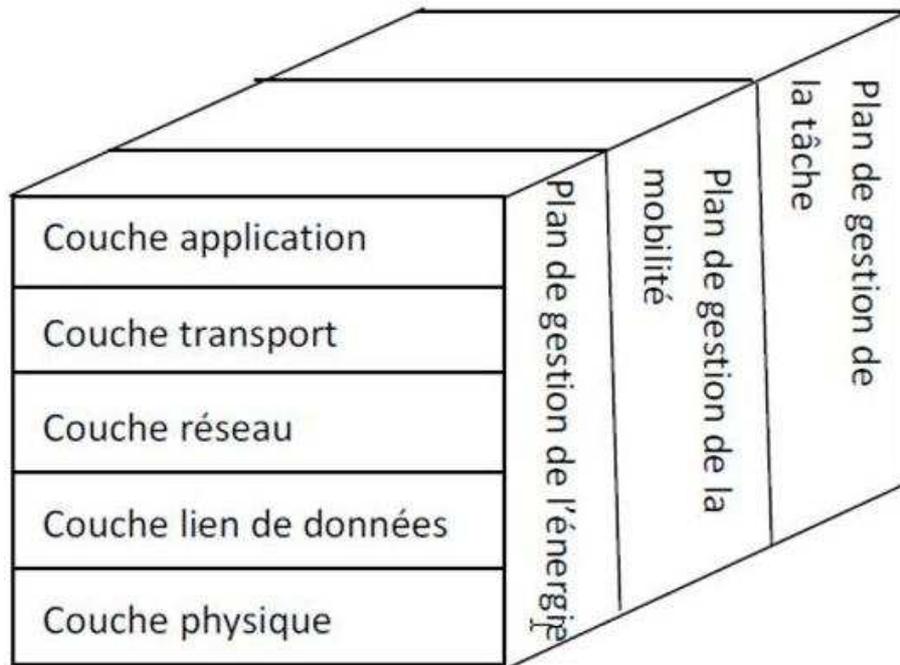


Figure 1.9 La pile protocolaire des réseaux de capteurs.

La pile protocolaire utilisée pour un nœud puits (sink) et un nœud senseur est présentée ci-dessus. Cette pile prend en charge le problème de consommation d'énergie, intègre le traitement des données transmises dans les protocoles de routage, et facilite le travail coopératif entre les capteurs.

Elle est composée de la couche application, transport, réseau, liaison de données, physique, ainsi que de trois plans qui sont : le plan de gestion d'énergie, plan de gestion de tâches et le plan de gestion de mobilité.

1.3.3.1 Couche Physique

Cette couche est responsable sur le support de transmission qui achemine les données entre les nœuds capteurs. Elle est responsable sur la sélection de fréquence, la génération de la fréquence de porteuse, la détection du signal, la modulation et le cryptage de la donnée. Deux médias peuvent être utilisés pour communiquer dans les réseaux de capteurs sans fils : l'infrarouge et la radiofréquence. Néanmoins, la communication par radiofréquence est la plus utilisée, car les paquets transportés sont petits, les débits de transmission sont bas, et la

réutilisation de fréquence est considérable. La communication sans fils sur de longues distances consomme le plus d'énergie, c'est pourquoi, les concepteurs de la couche physique pour un réseau de capteurs doit prendre en considération la contrainte énergétique.

1.3.3.2 Couche Liaison de données

Cette couche se charge essentiellement de multiplexer le flux de données, détecter les trames, contrôler l'accès au média de transmission (MAC : *Media Access Control*) et contrôler les erreurs. Elle garantit également des connexions point-à-point ou point-à-multipoints fiables. Le protocole MAC (*Media Access Control*) doit réussir à créer une infrastructure pour le réseau, et établir des liens de communication pour le transfert de données. Les protocoles MAC traditionnels ont comme objectifs l'offre de meilleures qualités de services et une bande passante efficace. Ces protocoles ne peuvent pas être appliqués sur les réseaux de capteurs puisque la minimisation de la consommation de l'énergie est un objectif secondaire pour ces protocoles, mais une contrainte à valeur pour les réseaux de capteurs. De là se montre la nécessité de conception de protocoles qui doivent être capables de réduire au minimum les collisions avec l'émission des voisins et de minimiser les retransmissions.

Le contrôle d'erreurs dans les réseaux de capteurs ne peut pas s'appuyer sur les techniques traditionnelles existantes, le mode ARQ (*Automatic Repeat Request*) est limité par le coût additionnel d'énergie dans la retransmission et le temps de servitude, le mode FEC (*Forward Error Control*) repose sur des techniques de décodage très complexe pour la correction des erreurs, donc inutiles pour les réseaux de capteurs. Des codes de contrôles d'erreurs avec une faible complexité de codage et de décodage conviennent le mieux et sont considérés comme les meilleures solutions pour les réseaux de senseurs.

1.3.3.3 Couche Réseau

Dans un réseau de capteurs, les nœuds capteurs sont déployés aléatoirement dans un champ de captage, à l'intérieur du phénomène observé ou très proche de celui-ci. Pour permettre la communication dans le réseau, des protocoles de routage spéciaux basés sur la communication multi-sauts sont nécessaires entre les nœuds capteurs et le nœud puits du réseau. Toutefois, les techniques liées au routage dans les réseaux ad hoc, ne peuvent pas répondre aux exigences uniques des réseaux de capteurs sans fil, notamment dans l'absence d'un adressage global, et la priorité absolue donnée à la quantité d'énergie consommée par les

nœuds. En effet, la conception de la couche réseau dans un réseau de capteurs doit être guidée par les principes suivants :

- L'efficacité en consommation d'énergie est une considération prioritaire.
- Tous les protocoles dans les réseaux de capteurs prennent en charge les données d'application qui circulent dans le réseau (*Data centric*).
- L'agrégation des données est utile quand elle ne cache pas l'effort collaboratif des nœuds capteurs.
- Un réseau de capteur idéal garantit un adressage basé-attributs et ses nœuds sont conscients de leur localisation.

1.3.3.4 Couche Transport

Cette couche constitue une interface entre la couche application et la couche réseau, les principales fonctionnalités de cette couche sont:

- Le multiplexage/démultiplexage des messages entre les applications et la couche réseau.
- Le contrôle de haut niveau des données.
- La régularisation de la quantité des données injectées dans le réseau.

Les protocoles de la couche transport sont particulièrement nécessaires si le réseau de capteurs est prévu d'être accessible à partir d'autres réseaux externes tels qu'Internet. Les mécanismes de communication du protocole TCP ne sont pas compatibles avec l'environnement des réseaux de capteurs, ces derniers présentent des insuffisances hardwares (énergie limitée, mémoire limitée), un nœud ne peut pas stocker des accusés de réception coûteux. «*TCP splitting*» peut être nécessaire pour habilitier le réseau de capteurs d'interagir avec les réseaux externes. Dans cette approche, les connexions TCP se terminent au niveau des nœuds puits, dès lors, des protocoles spéciaux au niveau de la couche transport seront utilisés pour assurer la communication entre le nœud puits et les nœuds capteurs. Ainsi, la communication entre les utilisateurs et le nœud puits sera via les protocoles TCP ou UDP, par Internet ou par satellite. D'autre part, la communication entre le nœud puits et les nœuds capteurs peuvent être purement via le protocole UDP, à cause des limitations de mémoire connues dans les nœuds capteurs.

1.3.3.5 Couche Application

Cette couche, fournit des mécanismes pour permettre à l'utilisateur d'interagir avec le réseau par l'intermédiaire d'un réseau étendu en lui fournissant des interfaces pour la création

et la diffusion des requêtes et un moyen d'interpréter les réponses reçues, également ils rendent transparents le matériel et les logiciels utilisés dans les couches inférieures.

Bien que plusieurs domaines d'application ont été proposés et définis pour les réseaux de capteurs sans fil, la conception des protocoles agissants dans la couche application reste largement inexploitée.

1.3.4 Caractéristiques des réseaux de capteurs

Comme nous l'avons mentionné auparavant, un réseau de capteurs sans fils est un type particulier de réseaux ad-hoc. De ce fait, les WSNs héritent de quelques caractéristiques des réseaux ad-hoc. Cependant ils ont d'autres caractéristiques particulières [GUP02, VIS04]:

- **Ressources limitées:** les capteurs d'un WSN sont généralement de très petite taille. Ce facteur de taille limite considérablement la quantité de ressources qui peuvent être mises dans ces nœuds. En conséquence, la capacité de traitement et de mémoire est très limitée.
- **La durée de vie limitée:** dans un WSN, les capteurs sont limités par la contrainte d'énergie. Ils fonctionnent habituellement sans surveillance dans des régions géographiques éloignées et souvent inaccessibles. L'intervention humaine pour recharger ou changer les batteries est généralement impossible.
- **Scalabilité:** le nombre de capteurs dans un réseau de capteurs peut atteindre des centaines ou des milliers (selon l'application).
- **Bande passante limitée:** en raison de la puissance limitée, les capteurs ne peuvent pas supporter des débits élevés.
- **Déploiement aléatoire et dense:** dans un WSN, les capteurs sont généralement déployés d'une façon dense et plus ou moins aléatoire. La forte densité est souvent liée à des raisons de fiabilité.
- **Topologie dynamique:** vu le déploiement aléatoire des WSN, il est généralement impossible de déterminer une configuration à priori du réseau. De plus, un manque d'énergie ou une défaillance inattendue à cause de l'environnement rude peuvent empêcher certains capteurs de faire leur travail de capture et éventuellement de communiquer avec les autres nœuds.
- **Auto-organisation:** pour remédier au problème de changement non prédictible de topologie, une auto-organisation du réseau s'avère nécessaire. C'est-à-dire que les nœuds doivent savoir localiser les nœuds voisins et établir des routes pour que l'information puisse circuler à travers le réseau.

- **Le schéma de communication:** contrairement au réseau ad-hoc qui utilise un schéma de communication point à point, un WSN doit assurer trois schémas:

Many-to-one: il est généralement utilisé pour acheminer les rapports de données vers le puits. En raison des redondances, les lectures doivent être agrégées dans des zones particulières du réseau.

One-to-many: ce schéma est utilisé par le puits pour communiquer avec les capteurs, afin de propager des requêtes ou contrôler le fonctionnement des nœuds. Ces communications peuvent être générales ou ciblées c'est à dire destinées à une certaine région du réseau.

Communications locales: ce sont les communications à un saut entre des capteurs voisins.

- **Agrégation des données:** dans les WSNs, les données produites par les nœuds capteurs sont très corrélées vu le déploiement dense des capteurs. Ceci implique d'éventuelles redondances des données capturées. Pour remédier à ce problème et éviter la surcharge du réseau et le gaspillage d'énergie, les données corrélées seront fusionnés par des nœuds particuliers du réseau. Ce processus est appelé agrégation de données.
- **Sécurité limitée:** vue l'absence d'infrastructure dans les WSNs et la technologie sans fil sous-jacente, les réseaux de capteurs sans fils sont plus sensibles aux attaques qui menacent les données transmises. De plus, les techniques conventionnelles utilisées pour faire face à ces attaques ne sont plus applicables dans les WSNs à cause des limitations de ressources connues dans ce type de réseau (puissance de calcul et mémoire) [KHE 05, GAY 03].

1.3.5 WSNs VS Ad Hoc

Un réseau de capteur est un type particulier de réseaux Ad Hoc, ces deux types ont en commun l'absence de l'infrastructure et la communication à l'aide du sans fils. Le tableau suivant, illustre les différences entre ces deux réseaux.

| | Réseau de capteurs | Réseaux Ad Hoc |
|----------------------------------|--|--|
| Composition | Petit micro-capteurs | Portable, PDA.... |
| Flot de communication | plusieurs à un (many to one) | Plusieurs à plusieurs (many to many) |
| Communication | Diffusion /Agrégation | Point à point |
| Topologie | Mobilité faible | En constante évolution et mobilité forte |
| Relations entre les nœuds | Collaborant pour le même objectif | Chaque nœud à son propre objectif |
| Identifications des nœuds | Très grand nombre de nœuds n'ayant pas tous une ID | Présence de la notion d'ID |
| Objectifs du réseau | Objectif ciblé | Générique / communication |
| Contrainte clé | Ressource énergétiques | Débit /QOS |

Tableau 1.1 WSN vs Ad Hoc.

1.3.6 Consommation de l'énergie dans les WSNs

La consommation d'énergie est le point le plus crucial dans les réseaux de capteurs sans fil, car elle influe directement sur la durée de vie du réseau. Les capteurs disposent généralement d'une ressource énergétique à durée de vie limitée (<0.5 Ampère-heure, 1.2 V) et toute intervention humaine pour recharger et changer la batterie est la plus part du temps exclue.

Sachant que les réseaux de capteurs sont basés sur la communication multi-sauts, chaque nœud a un double rôle: il capte et/ou mesure l'information, de plus il doit participer au routage des paquets transmis. Tout dysfonctionnement engendre un changement significatif de la topologie globale et peut nécessiter une réorganisation totale du réseau. C'est pour cela que la consommation énergétique est primordiale dans les WSNs. Les concepteurs négligent souvent les autres métriques de performance telle que la durée de transmission et le débit, au profit du facteur de consommation d'énergie.

La consommation d'énergie dans un capteur est causée par les trois fonctions suivantes: la capture, la communication et le traitement de données.

– *La Capture*

L'énergie consommée pour effectuer la capture varie en fonction de la nature du matériel et de l'application. Par exemple, une capture à intervalles réguliers consomme moins d'énergie qu'une surveillance continue [KHE 04]. Comme le montre la figure 1.9, cette énergie est négligeable par rapport à l'énergie de communication.

– *La Communication*

Parmi les trois phases citées, cette phase consomme la plus grande quantité d'énergie. Ceci est dû à la multitude de composants électroniques intégrés au circuit responsable de cette opération. Cette phase concerne les deux étapes d'émission et de réception de données. La figure 1.9 montre un modèle d'énergie du premier ordre. Pour transmettre un message de k bits sur une distance de d mètres l'émetteur consomme:

$$E_{Tx}(k, d) = E_{elec}(k) + E_{amp}(k, d) = E_{elec} * k + E_{amp} * k * d^\alpha$$

Et le récepteur consomme:

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

Avec :

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

k : Taille d'un message en bits.

d : Distance entre l'émetteur et le récepteur.

E_{amp} : Energie d'amplification.

α : Sa valeur dépend des conditions de l'environnement. Elle est comprise entre 2 et 6

En effectuant cette tâche de communication, le capteur peut fonctionner selon trois modes:

- le mode *Actif* où il exécute une fonction ou transmet un message ;
- le mode *Idle* où il est seulement à l'écoute d'éventuels messages à recevoir;
- le mode *Sleep* où il est en veille (*Figure 1.10*).

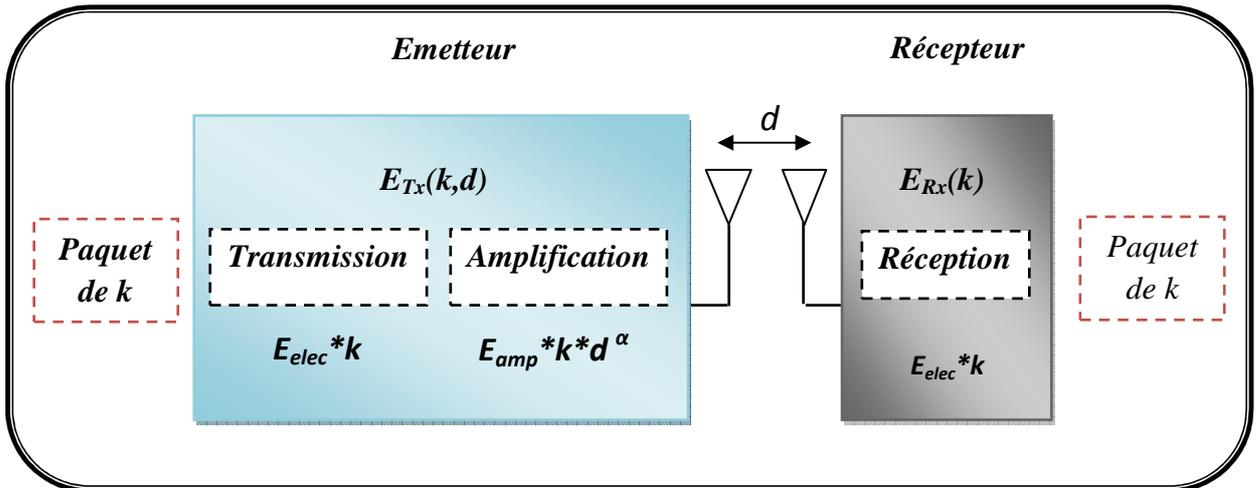


Figure 1.10 La modèle de consommation d'énergie pour la communication [HEI 02].

– **Le traitement de données**

L'énergie consommée pour les calculs est beaucoup plus faible que l'énergie consommée pour les opérations de communication. D'après [POT 00], la transmission de 1kb de données sur une distance de 100 mètres consomme une énergie de 3 Joules. Tandis qu'un processeur avec une capacité de calcul de 100 MIPS (Million of Instructions Per Second) peut exécuter 300 millions d'instructions avec la même quantité d'énergie. La *Figure 1.11* montre les différents niveaux de consommation d'énergie en capture, traitement et communication

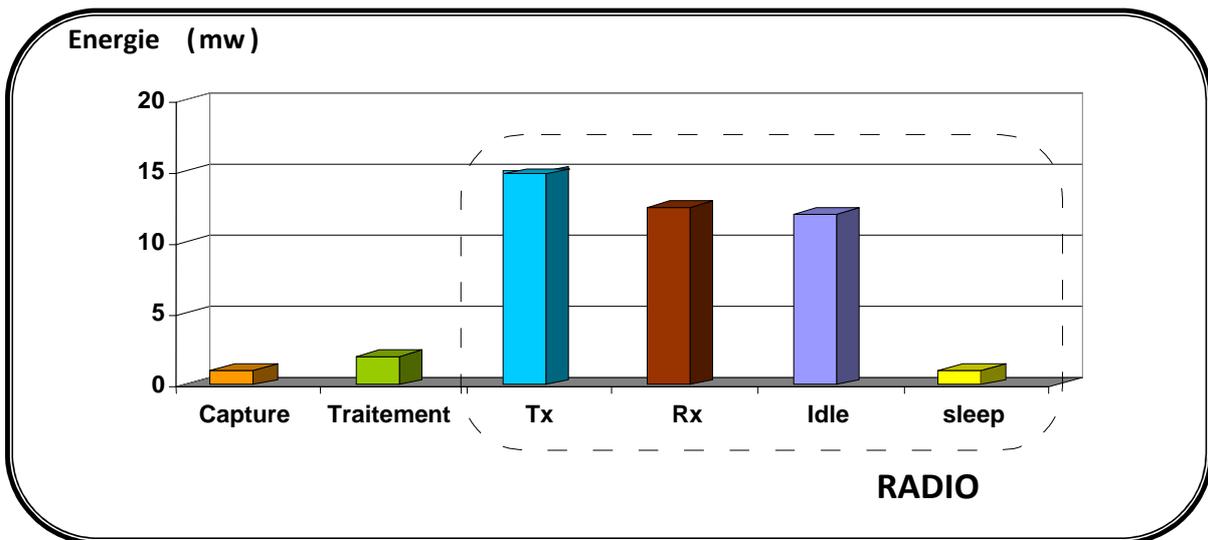


Figure 1.11 La consommation d'énergie en capture, traitement et communication [POM 04].

1.3.7 Axes de recherche

L'évolution et le déploiement réussi des réseaux de capteurs sans fil (WSN) dans le monde ont ouvert la voie pour un accroissement des activités de recherche, développement et standardisation dans ce domaine. Les mécanismes mis en œuvre deviennent de plus en plus complexes, ceci pour assurer des fonctions exigées par certaines applications: qualité de service, sécurité, mobilité... Les contraintes liées au déploiement, la topologie et l'énergie ainsi que la taille miniaturisée des capteurs motivent une grande partie des problématiques de recherches concernant les réseaux de capteurs sans fil.

Les réseaux de capteurs présentent un champ d'application très vaste, selon MIT's Technology Review, il s'agit de l'une des dix nouvelles technologies qui bouleverseront le monde et notre manière de vivre et de travailler. La batterie est un composant important d'un capteur. En général, elle n'est ni remplaçable ni rechargeable. Elle peut être, en partie, alimentée par une unité génératrice d'énergie comme les cellules solaires. De petite taille, elle fournit donc une quantité d'énergie très limitée à l'échelle de 1 à 2J par nœud. Elle limite ainsi la durée de vie du capteur et influe sur le fonctionnement global du réseau. C'est pourquoi, les protocoles permettant d'économiser l'énergie occupent, aujourd'hui, un axe important de recherche dans ce domaine.

Un capteur assure l'acquisition, le traitement de données et les communications. C'est cette dernière tâche qui est la plus consommatrice d'énergie. Un bon schéma de gestion d'énergie doit donc, en priorité, prendre en compte les communications. Les contraintes liées au déploiement, à la topologie et à l'énergie motivent une grande partie des problématiques de recherches concernant les réseaux de capteurs sans fil. La majorité des recherches sur le routage se concentre sur l'optimisation de la consommation d'énergie, de plus la livraison de la bonne donnée au bon moment doit être assurée. L'aspect sécurité est un concept clé dans les domaines où la confidentialité est exigée tel que le domaine militaire, des algorithmes de cryptographie et des mécanismes d'authentification, doivent être mis en place au sein de ces réseaux.

1.3.7.1 Architecture matérielle

L'architecture matérielle des capteurs a fait l'objet de plusieurs projets de recherche ces dernières années. Le but principal reste toujours la conception de dispositifs peu coûteux, plus petits, et avec une faible dissipation d'énergie. A titre d'exemple citons quelques projets.

Le projet *WINS* (*Wireless Integrated Network Sensors*) lancé par *UCLA* et *Rockwell* a permis de développer des capteurs intégrant la capture, le traitement, et la communication (figure 1.12-a). Ces capteurs sont fabriqués en utilisant la technologie *LWIM* (*Low-power Wireless Integrated Microsensor*). Une nouvelle génération de nœuds *WINS* est développée par la société *Sensoria Corporation* [web 01] permet de contrôler et de traiter plusieurs signaux. Les données captées et les messages sont échangés à travers les nœuds via des modems utilisant la radiofréquence [HAM 07].

Le projet *Smart-Dust* développé à l'université de Berkeley vise à développer des nœuds capteurs de taille millimétrique. L'objectif est la miniaturisation des nœuds capteurs de sorte qu'ils aient la taille d'une particule de poussière (*dust*). *WeC Mote* (figure 1.12-b) et *CCR Mote* étaient les deux premiers types de nœuds capteurs développés dans ce projet. *Rene*, *Mica Mote* et *Mica2 Mote* (Figure 1.12-c) sont venus par la suite. La famille de *Mote* emploie *TinyOS*, un système d'exploitation open-source compact et simple basé sur les événements. *Telos* (Figure 1.12-d) est la nouvelle génération de plate-forme de *Mote* qui utilise un microcontrôleur différent et un canal radio de technologie *Zigbee*. Le projet de Berkeley a également construit le *Spec Mote*. (Figure 1.12-e), un nœud d'usage universel de taille réduite.

L'objectif du projet *μAMPS* (micro-Adaptive Multi-domain Power-aware Sensors) est de concevoir des capteurs efficaces en énergie (Figure 1.12-f), des logiciels pour ces nœuds, et des protocoles de communication [HAM 07].

Le projet *Crossbow* [WEB 01] a été lancé en 1995 à l'université de Berkeley pour exploiter les capteurs *MEMS*. Actuellement, l'entreprise 'Crossbow Technology' produit plusieurs plateformes de capteurs. Les capteurs *MICA2* (868/916 MHz), *MICAZ* (2.4 GHz) et *IRIS* (2.4 GHz) sont les plus connus. Ces capteurs sont équipés de processeurs fonctionnant avec un système d'exploitation très réduit (*TinyOS* généralement). En 2008, 'Crossbow' a lancé 'Eko Pro Series System'. Cette plateforme vise le domaine de l'environnement. Elle peut être utilisée pour différentes applications telles que le changement de climat, la biodiversité, la qualité de l'eau, la contamination d'eaux souterraines, la contamination des sols, l'utilisation des ressources naturelles, la gestion des déchets ou la pollution atmosphérique.

Beaucoup d'autres fabricants existent actuellement sur le marché: *PicoRadio*, *Digital Sun*, *Rockwell Science Center*, *Millennial Net*, *Dust Networks*, *Intel*, *Ember Corporation*, *Ferro Solutions*, *Senera*, *Sensicast* et *Xsilog*.

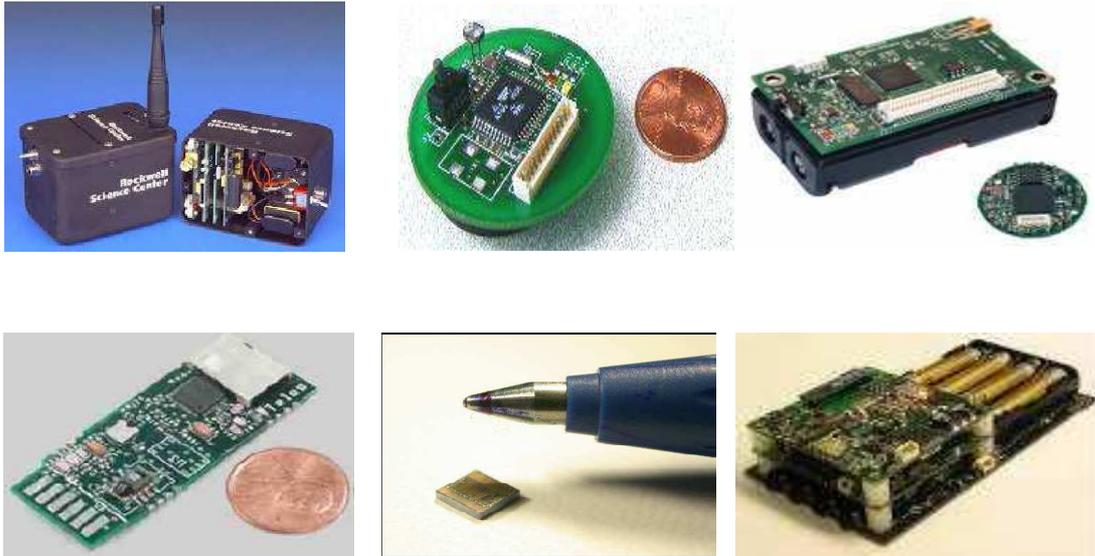


Figure 1.12 Exemples de nœuds capteurs.

1.3.7.2 Déploiement d'un WSN

Le problème de déploiement dans les réseaux de capteurs sans fils peut être formulé de la façon suivante: étant donné la région cible d'un WSN, comment et où doit-on placer les nœuds capteurs ? [KRI 05].

Le nombre important de nœuds utilisés dans un réseau de capteurs empêche leur déploiement suivant un plan soigneusement établi. Cependant, un schéma général pour le déploiement initial doit être conçu de telle sorte à garantir la connectivité entre les nœuds et la couverture de toute la région cible (champ de captage). Ceci est indispensable afin d'éviter toute perte de données. La limitation d'énergie et la robustesse doivent aussi être prises en compte.

La densité et la manière de déployer les nœuds dépendent de plusieurs facteurs. Citons entre autres: la nature de l'application, les capacités individuelles des nœuds, les caractéristiques de propagation radio, et la topologie de la région [MEH 01].

Après la phase de déploiement, un WSN est exposé fréquemment à des changements. En effet, la panne matérielle constitue un événement très commun à cause de l'épuisement de l'énergie ou la destruction du nœud. Une bonne gestion du réseau exige le déploiement de capteurs additionnels afin de remplacer ceux qui sont en panne.

1.3.7.3 Localisation des nœuds

Dans le problème de localisation, il est possible de distinguer deux aspects: la localisation des nœuds capteurs et la localisation d'une cible externe [PAR 05].

Le premier aspect s'intéresse à l'emplacement des capteurs dans le réseau. Vu le déploiement arbitraire des nœuds, leur localisation s'avère nécessaire pour pouvoir fournir des données significatives à l'utilisateur et éventuellement utiliser les positions des capteurs dans les protocoles de la couche MAC et de la couche réseau. Le deuxième aspect concerne la localisation d'une cible externe. Cette cible peut être fixe ou mobile. Cet aspect est généralement lié aux applications de type 'Detection/Tracking' [PAR 05].

Théoriquement, un système GPS résout le problème de localisation. Toutefois, équiper chaque capteur d'un récepteur GPS constitue souvent une solution irréalisable vu le coût prohibitif d'un tel équipement, de la réserve énergétique limitée des capteurs et du mauvais fonctionnement de cette technologie à l'intérieur de locaux. Plusieurs solutions alternatives centralisées et d'autres distribuées ont été proposés [ESS 07].

A l'heure actuelle, les approches de localisation utilisées se décomposent en deux phases:

- l'estimation des distances qui se fait en fonction des propriétés du signal (le temps de propagation, l'atténuation ou la forme) ;
- la dérivation des positions qui vise l'identification des positions de chaque nœud en se basant sur la position de ses voisins [MAT 05].

1.3.7.4 Synchronisation

Comme dans tout autre type de réseaux à architecture distribuée, le problème de synchronisation est un point critique dans les réseaux de capteurs sans fils.

Le système temporel dans un WSN vise les mêmes objectifs que dans les systèmes distribués traditionnels : la précision et la validité. Cependant, ceci doit être garanti en respectant des exigences spécifiques aux réseaux de capteurs sans fil, notamment le temps et l'énergie nécessaires. De ce fait, les méthodes de synchronisation classiques doivent être adaptées.

1.3.7.5 Sécurité

Les applications de capteurs sans fils sont très nombreuses, et touchent plusieurs domaines critiques. Dans ce genre d'applications, la fiabilité est un enjeu majeur.

Depuis les premières années, les recherches dans les WSN ont été orientées vers les problèmes de déploiement, de localisation, de routage, de clustering, etc. L'aspect de sécurité a été souvent négligé, ce qui a permis la découverte de différents types d'attaques ciblant toutes les couches protocolaires.

Un réseau de capteurs sans fils peut être attaqué de deux manières :

- par des capteurs ayant les mêmes capacités que ceux présents dans le réseau. Dans ce cas, l'attaquant est limité par les mêmes contraintes que les autres nœuds;
- par des calculateurs plus puissants, comme les PDA et les laptops. Donc, l'attaquant bénéficie de plus de puissance, et de nouvelles attaques peuvent être menées.

SPINS avec ses deux blocs de sécurité SNEP & μ Tesla a été l'un des premiers protocoles sécurisés dans les réseaux de capteurs sans fils. D'autres protocoles ont été proposés par la suite, INSENS [DEN 06], SecRout[YIN 06], etc.

1.3.8 Application

Le coût de plus en plus faible, la taille de plus en plus réduite des capteurs, la large gamme de types de capteurs disponibles (optique, thermique, vibration...) et le support de communication sans fils utilisé, permettent aux réseaux de capteurs d'intégrer plusieurs domaines d'application, tels que le domaine militaire, l'environnement, le domaine médical et l'industrie.

1.3.8.1 Militaire



Figure1.13 Application militaire des réseaux de capteurs.

Le déploiement rapide, l'auto-organisation, la tolérance aux pannes, et le coût réduit rendent le concept de réseau de capteurs une meilleure approche pour les champs de bataille.

On cite comme exemples des applications militaires auxquels les réseaux de capteurs peuvent être appliqués : [KHE08] [AKY02]

- Le contrôle des forces, équipement et munition

Les chefs de troupes et les commandants des opérations militaires, peuvent utiliser les réseaux de capteurs pour se renseigner sur l'état des troupes et la disponibilité des équipements. Chaque troupe peut véhiculer plusieurs capteurs qui rapportent leurs états. Les données captées par les nœuds capteurs seront collectées dans un nœud puits et envoyées au chef de la troupe qui à son tour les transforme au niveau plus haut après avoir collecté le maximum d'information nécessaires sur l'état des troupes.

- Reconnaissance et surveillance du champ de bataille

Les réseaux de capteurs sans fils peuvent être facilement déployés sur un champ d'une bataille pour surveiller de prêt les activités des forces ennemies.

- Ciblage

Une autre application importante des réseaux de capteurs dans le domaine militaire consiste en leur incorporation dans les systèmes de guidage des munitions intelligentes.

- Détection et reconnaissance des attaques nucléaires, biologiques et chimiques

Les réseaux composés des capteurs de réactions chimiques et biologiques, peuvent être utilisés pour la détection précise et instantanée des agents chimiques de l'ennemie. Cela est très important dans le terme de diminuer les risques engendrés par ce genre d'attaques.

1.3.8.2 Médicales

Le domaine médical profite des progrès des réseaux de capteurs, ces derniers sont utiles pour le contrôle des malades, aussi bien le mouvement des médecins au sein de l'hôpital, l'administration des médicaments ainsi que la télésurveillance et la collecte des informations physiologiques humaines.



Figure1.14 Surveillance d'activités des personnes âgées.

– Télésurveillance des informations physiologiques humaines

En utilisant les réseaux de capteurs sans fil, cette opération fournit des informations qui peuvent être stockées pendant une longue période, et utilisées par la suite pour des fins multiples, notamment les explorations médicales. Les réseaux de capteurs installés peuvent également détecter et surveiller le comportement des personnes âgées et permettre une intervention rapide en cas de nécessité. Par conséquent, Ces petits nœuds capteurs permettront aux sujets surveillés une meilleure liberté de mouvements, ainsi qu'aux médecins une identification plus rapides de certains symptômes prédéfinis. De plus, ils assurent aux sujets une meilleure qualité de vie comparée à celle dans les centres hospitaliers. Dans ce cadre, une maison de santé intelligente a été conçue à l'université de Grenoble (France) pour démontrer la faisabilité de tels systèmes. Loren & al ; décrivent une application biomédicale qui utilise les réseaux de capteurs et consiste à construire une rétine artificielle servant à aider les personnes malvoyantes, cette rétine contient une centaine de micro-capteurs munis d'une communication sans fil employée pour le contrôle, l'identification et la validation de l'image captée.

– Le Suivi et la surveillance des médecins et des patients au sein de l'hôpital

Les médecins portent des capteurs pour pouvoir les localiser facilement, de plus chaque patient aura des capteurs dont chacun à un rôle bien déterminé, par exemple un capteur pour détecter les battements du cœur.

- L'administration des médicaments

Si les nœuds capteurs peuvent être attachés aux médicaments, cela permettra de minimiser la probabilité de prescrire des mauvais traitements aux patients, car ces derniers auront, de la même manière, des capteurs qui identifient leurs allergies et les médicaments qui leur sont adéquats. Des systèmes informatisés peuvent aider à diminuer les possibilités d'effets secondaires causés par les médicaments inadéquats.

1.3.8.3 Environnement

La surveillance des environnements inaccessibles par l'être humain est rendue possible grâce au déploiement d'un nombre de capteurs dans l'endroit visé, citons par exemple :



Figure 1.15 Application des RCSF dans la surveillance de l'environnement des oiseaux marins.

- Détection des incendies de forêts

Des millions de capteurs utilisant les fréquences radio ou la communication par voix optique peuvent être déployés dans une forêt dans le but de détecter les incendies avant qu'ils deviennent incontrôlable. Les capteurs utilisés pour cette tâche doivent être assez durs et robustes pour faire face aux effets environnementaux (tels que les obstacles), de plus, ils doivent être équipés par des systèmes de rechargement d'énergie efficaces tels que les panneaux solaires pour pouvoir fonctionner sur une longue durée.

- Schématisation de la bio-complexité de l'environnement

Cette opération requière des approches sophistiquées pour intégrer des informations précises à travers des échelles spatiales et temporelles. En effet, les avancées technologiques dans le

domaine de capture distante et de collection automatique de données ont permis une résolution temporelle, spectrale et spatiale plus élevée, tout en réduisant le coût de l'opération de schématisation par unité de surface. Grâce à ces avancées également, les capteurs ont la possibilité d'être connectés à Internet pour permettre aux utilisateurs distants d'observer et surveiller constamment la bio-complexité de l'environnement.

Une unité de l'université de Californie a entrepris un travail de schématisation de la bio complexité de l'environnement, dans lequel 3 grilles de surveillance contenant 25 à 100 nœuds capteurs sont implantés pour fournir des vues multimédias fixes et délivrer des informations liées à l'environnement.

– Détection des inondations

Parmi les exemples d'utilisation des réseaux de capteurs pour la détection des inondations on trouve le système ALERT (*Automated Local Evaluation in Real Time*) déployé aux Etats Unis. Ce dernier englobe plusieurs types de capteurs hydrologiques qui détectent la pluie, le niveau d'eau, ainsi que d'autres capteurs météorologiques qui servent à la détection de la température, la pression, etc. tous ces types de capteurs fournissent les informations nécessaires à une base de données centralisée via une communication radio, un modèle de prévision des inondations est adopté pour analyser les données reçues et générer les avertissements éventuels. Cependant plusieurs projets de recherche tel que COUGAR (*Cornell University*) sont entrain d'élaborer des approches distribuées pour l'interaction avec les nœuds capteurs et la collecte de données, et par conséquent éviter le mode centralisé qui ne permet pas aux utilisateurs de changer dynamiquement le comportement du système et nécessite un taux de communication et d'interaction relativement élevé.

1.3.8 Autres

Grâce aux capteurs intelligents pouvant être intégrés aux appareils électroménagers (aspirateur, réfrigérateurs, fours-micro-ondes,...) l'utilisateur peut aisément contrôler ces appareils, à distance ou localement.

Les réseaux de capteurs possèdent également d'autres applications dans le domaine commercial, parmi lesquelles on peut énumérer : la surveillance de l'état du matériel, la gestion des inventaires, le contrôle de qualité des produits, la construction des espace d'achat intelligents, le contrôle de l'environnement dans les bâtiments administratives, le contrôle des robots dans les environnement de fabrications automatiques, les jouets interactifs, les musées interactifs, le contrôle et l'automatisation des processus d'usinage, le diagnostic des machines,

le transport, la détection et la surveillance des vols de voitures, le dépistage des véhicules, l'instrumentation des chambres blanches consacrées aux traitements des semi-conducteurs, etc...

1.4 Conclusion

Dans ce chapitre, nous avons défini et décrit brièvement ce qu'est un réseau sans fils, un réseau Ad-hoc et ses caractéristiques ainsi qu'un réseau de capteurs sans fil. Nous avons décrit le capteur, ses fonctionnalités, sa plateforme, son architecture comme nous avons cité la plateforme MICA. Nous avons cité les caractéristiques d'un réseau de capteurs, sa topologie et nous avons présenté quelques applications. Les communications dans les WSNs font aussi partie de ce chapitre. Dans la suite, nous aborderons le routage et l'auto-organisation dans un WSN, nous allons présenter quelques protocoles hiérarchiques qui vont nous servir lors de la conception de notre approche.

Chapitre 2

Auto-organisation dans les Réseaux de capteurs sans fil

2.1 Objectif

Comme nous l'avons déjà défini dans le chapitre précédent, un réseau de senseurs est un ensemble de nœuds capteurs qui sont dynamiquement et arbitrairement éparpillés d'une manière où l'interconnexion entre les nœuds peut changer à tout moment. Dans la plupart des cas, l'unité destination ne se trouve pas obligatoirement dans la portée de l'unité source ce qui implique que l'échange des données entre deux nœuds quelconques, doit être effectué par des stations intermédiaires. La gestion de cet acheminement de données, ou routage, implique l'établissement d'une certaine architecture globale que l'on doit tenir compte de la mobilité des unités et de la versatilité du médium physique.

Les protocoles de routage sont donc conçus dans le but de sélectionner un chemin entre une station source, et une destination. Les protocoles de routage hiérarchiques qui font à la fois l'auto-organisation d'un réseau et la découverte de chemin, sont devenus les approches les plus adaptés afin de remédier aux problèmes liés à l'autonomie d'un capteur, et à la réserve énergétique limitée.

Dans ce chapitre, nous allons souligner la notion d'auto-organisation, les avantages qu'elle apporte, comme nous allons aussi citer les différentes méthodes d'auto-organisation avec des protocoles associées.

2.2 Auto-organisation basée sur la mise en plan CDS.

Cette technique affecte deux états possibles pour les nœuds du réseau : Nous trouvons ainsi des nœuds dominants et des nœuds dominés. L'ensemble des nœuds dominants est de sorte que tous les nœuds du réseau sont, soit dans cet ensemble, soit voisin d'un nœud dominant de cet ensemble. L'ensemble de nœuds dominants est dit connecté si et seulement s'il existe un chemin entre chaque paire de nœuds composant cet ensemble. Parmi les principaux protocoles d'auto-organisation basés sur la mise en place d'un CDS, nous citons :

- CDS-règle k (*Connected Dominating Set-règle k*),
- CDS-IDS (*Connected Dominating Set-Independent Dominating Set*)
- Legos (*Low Energy Self-OrGanisatiOn Scheme*).

2.2.1 CDS-regle k (Connecting Dominating Set regle k)

Soit S un ensemble connecté dominant CDS [LU 07]. S est défini comme un sous-ensemble de l'ensemble des sommets V , qui satisfait les trois points suivants :

- Chaque nœud de l'ensemble V est soit dominant (dans l'ensemble S), soit dominé.

- Un nœud dominé a un voisin à un saut appartenant à l'ensemble S.
- L'ensemble S est connecté.

Le CDS-règle k permet la construction d'un CDS en deux phases : une phase de marquage et une phase d'élagage de la règle k :

- Durant la première phase de marquage, chaque nœud se marque quand il possède deux voisins non-connectés entre eux comme le montre la figure 2.1 (a).
- Après cette première phase, le processus d'élagage de la règle k est appliqué par chaque nœud marqué. Si l'ensemble des voisins d'un nœud dominant sont couverts par un ensemble dominant de k nœuds voisins, et que ce nœud a le plus petit identifiant, alors ce dernier devient un nœud dominé comme le montre la figure 2.1 (b).

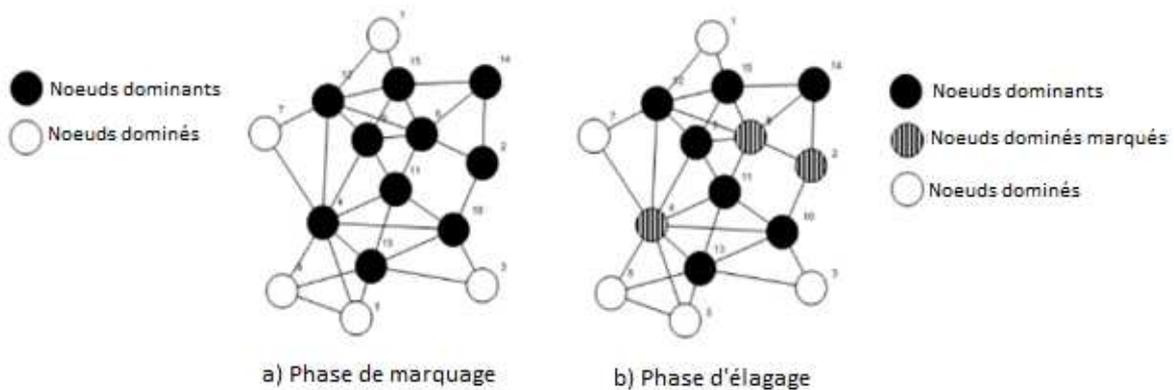
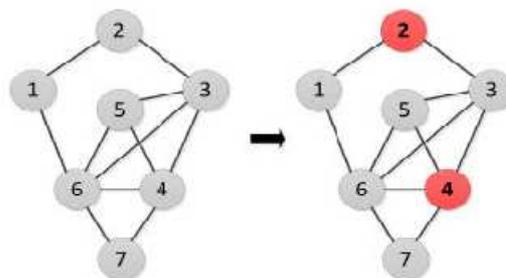


Figure 2.1 Principe du protocole CDS-règle k.

L'algorithme de CDS-règle k est complètement localisé. Il définit un ensemble connecté dominant de plus fort poids dans le voisinage. Chaque nœud appliquant cet algorithme a besoin des informations sur son voisinage à deux sauts. Dans cette optique, le recours aux paquets *hello* est indispensable.

2.2.2 CDS-IDS (Connected Dominating Set-Independent Dominating Set)

Un ensemble dominant indépendant (IDS) est un ensemble dominant où chaque nœud n'est pas adjacent à aucun autre nœud de l'ensemble. Ainsi un nœud appartenant à un IDS, est éloigné d'au moins deux sauts et d'au plus trois sauts d'un autre nœud dominant comme le



montre la figure suivante :

Figure 2.2 Exemple d'un IDS.

Les auteurs proposent une version distribuée de construction d'un CDS. L'algorithme élaboré est distribué et est composé de trois étapes, il construit :

- Un arbre couvrant.
- Un ensemble dominant indépendant.
- Un arbre dominant.

L'IDS construit par l'algorithme dans [PEN 02] est de sorte que chaque nœud de l'IDS est à exactement deux sauts d'un autre nœud de l'IDS. Les auteurs utilisent pour cela une notion de rang utilisée pour faire un ordonnancement total parmi les nœuds. Ce rang est calculé en se basant sur le nombre de sauts séparant chaque nœud à la racine de l'arbre et sur l'identifiant unique de chaque nœud. L'IDS est construit à partir du nœud racine. Ce dernier se déclare noir et envoie un message **BLACK**. Les nœuds recevant ce message **BLACK** se déclarent gris et envoient un message **GREY**. Un nœud recevant un message **GREY** de l'ensemble de ses voisins de rang inférieur, se déclare noir et envoie un message **BLACK**.

Les nœuds déclarés noirs appartiennent à l'IDS. Le CDS est construit aussi à partir du nœud racine, les nœuds noirs désignent un parent de rang inférieur. Les parents de tous les nœuds de l'IDS seront considérés comme des nœuds dominants : ils forment avec les nœuds de l'IDS l'ensemble dominant connecté.

2.2.3 Legos

Legos [LU 07] est un protocole d'auto-organisation visant à générer une topologie logique en épine dorsale non-orienté sous l'hypothèse d'un déploiement progressif ou sous l'hypothèse que les nœuds ne se réveillent pas simultanément pour rejoindre la topologie logique.

Avec Legos, les nœuds sont dans l'un des trois états suivants : *Leader L*, *Gateway G* ou *Member M*. Les nœuds *Members* sont des nœuds dominés et les nœuds *Leaders* et *Gateways* sont des nœuds dominants. Chaque nœud *Member* est relié à un seul *Leader*. Ce dernier est en charge de toutes les communications dans son voisinage à 1 saut. Les *Leaders* sont espacés de 2 sauts. Un nœud *Gateway* est en charge de l'interconnexion des deux ou plusieurs nœuds *Leaders* (voir figure 2.3). Les *Leaders* diffusent périodiquement des messages pour annoncer leurs présences. Un nœud, détectant un *Leader* dans son voisinage, s'attache à lui. Une fois

attaché à un *Leader*, les communications entrantes et sortantes de ce nœud sont gérées par ce *Leader*.

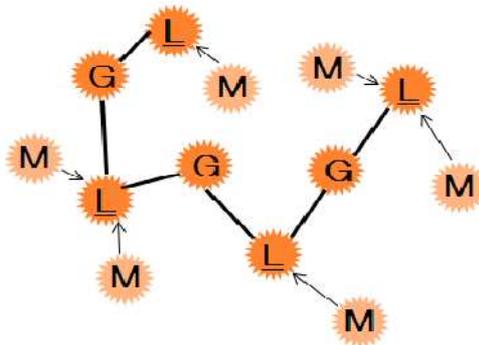


Figure 2.3 Topologie Legos [LU 07].

2.3 Auto-organisation basée sur l'élagage de lien

Cette technique utilise l'ajustement de la puissance de transmission pour éliminer les liens redondants et inutiles dans le réseau. Parmi les protocoles utilisant cette technique nous citons:

- RNG (*Relative Neighborhood Graph*)
- LMST (*Local Minimum Spanning Tree*).

2.3.1 Relative Neighborhood RNG

Relative Neighborhood Graph (RNG) est une famille de graphe proposée par Toussaint en 1980. L'idée est de couper l'arête la plus longue dans chaque triangle dans le graphe. L'idée de construction d'une topologie logique basée sur le graphe RNG est d'économiser les liens les plus longs en les coupant pour transmettre à plus faible puissance. La construction du graphe RNG repose sur la connaissance du voisinage direct. Ainsi cette construction est basée sur un algorithme localisé. Celui-ci respecte les propriétés de connexité du graphe initial. En d'autres termes si le graphe initial est connexe alors le graphe RNG l'est également.

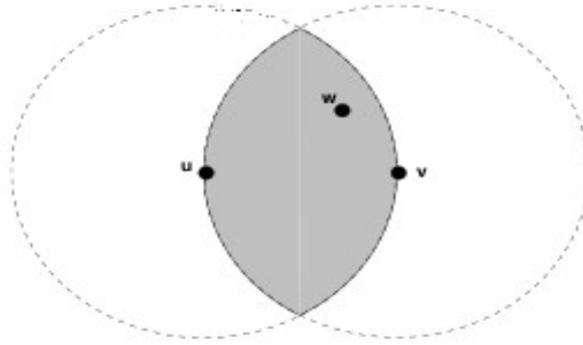


Figure 2.4 Elagage des liens avec l'algorithme RNG.

Sur la figure 2.4 Un arc (u,v) appartient au RNG s'il n'y a pas un nœud w dans la zone grise. Cette zone est l'intersection de deux cercles de rayon d (u,v), l'un est de centre u et l'autre est de centre v .

Le RNG peut être déduit localement par chaque nœud en utilisant seulement les distances qui le séparent de ses voisins. Avec un système de localisation (similaire à un GPS), les nœuds se contentent d'envoyer périodiquement un message "HELLO" avec leurs coordonnées. Ainsi, chaque nœud met à jour une liste de voisinage avec leurs coordonnées. Le calcul du RNG donnera donc si un nœud appartient au RNG local. On aura dans ce cas besoin d'une visibilité à un saut. Cependant, si le réseau ne possède pas un système de localisation, les nœuds calculent leur distance mutuelle (en utilisant la puissance du signal ou le délai d'émission). Chaque nœud en envoyant un message "HELLO" propage sa liste de RNG avec leurs distances. L'information nécessaire pour qu'un nœud prenne la décision est une information à 2-sauts. Dans les deux cas, avec le GPS ou l'estimation de la distance, on ne propage pas l'information plus que 2 sauts, par conséquent, la construction du RNG est plus rapide et nécessite beaucoup moins de messages "HELLO".

2.3.2 LMST

Les algorithmes du *Minimum Spanning Tree* MST ont été largement utilisés pour la diffusion dans le réseau et pour le routage. Cependant, la détermination du MST nécessite une connaissance complète de la topologie du réseau. Cette connaissance ne peut pas être envisagée dans les réseaux WSNs ou les WSANs. Dans [LI 05] une version de MST locale (Local MST ou LMST) dans laquelle chaque nœud détermine localement un ST sur son voisinage à deux sauts. Pour cela, les nœuds doivent s'échanger des paquets *hello* contenant la liste des voisins directs. Si deux nœuds sont voisins dans leurs topologies logiques de MST respectives, alors l'arête entre les deux nœuds est ajoutée dans le sous-graphe du LMST. Il est

à souligner que le sous-graphe résultant n'est pas un arbre. Il existe des boucles à cause de l'exécution localisée de l'algorithme.

2.4 Auto-organisation en Cluster

L'auto-organisation en clusters consiste à organiser le réseau en zones de captage, plus homogènes. Dans le cas des WSNs, les clusters sont généralement identifiés par un nœud particulier appelé cluster-head, ce dernier permet de coordonner entre les membres de son cluster, d'agréger leurs données collectées et de les transmettre à la station de base. Dans le cas des WSANs la formation de cluster se fait autour des actionneurs qui sont plus riche en ressources. Parmi les protocoles utilisant cette technique nous citons :

- LEACH (*Low Energy Adaptive Clustering Hierarchy*)
- HEED (*Hybrid Energy Efficient Distributed clustering*)
- PEGASIS (*Power-Efficient GATHERing in Sensor Information Systems*)
- Far-Legos (*Far Low Energy Self-Organization Scheme*)

2.4.1 Low Energy Adaptive Clustering Hierarchy (LEACH)

Proposé par (Heinzelman et al, 2000) : LEACH est le protocole de routage hiérarchique le plus populaire. Il choisit les cluster-head aléatoirement d'une manière probabiliste pour une période Dt , selon une politique « Round-Robin » qui consiste à répéter la procédure de formation de cluster-head à chaque intervalle de temps prédéfini, afin d'équilibrer la consommation d'énergie entre les nœuds. Chaque nœud se décide donc de s'élire cluster-head en calculant une métrique probabiliste comparé à un seuil $T(n)$, il informe par la suite son voisinage de sa décision. Chaque nœud non cluster-head décide de rejoindre le cluster le plus proche de lui en utilisant le minimum d'énergie de transmission.

Le seuil $T(n)$ se calcule comme suit :

$$T(n) = \frac{P}{1 - P(r \bmod (1/p))} \quad \text{si } n \in G$$

Où p : le pourcentage des CHs sur le réseau (généralement 5%);

- r : numéro du round en cours;
- G : l'ensemble des nœuds qui n'était pas CH dans les $(1/p)$ rounds précédentes.

Chaque nœud d'un cluster communique directement avec le cluster-head (un seul saut), en utilisant la technique TDMA instaurée par ce dernier. Cette technique permet de minimiser les collisions en allouant à chaque nœud un temps privé pour transmettre ses données vers son CH et chaque cluster-head communique directement avec la station de base. LEACH présente deux principales insuffisances : la première étant sa métrique d'élection de cluster-heads qui est probabiliste, dans ce fait, un nœud ayant une quantité d'énergie insuffisante peut être élu cluster-head. La seconde est que tout nœud supposé pouvoir atteindre la station de base, pratiquement ce n'est pas toujours vrai. Dans un réseau de petite taille, l'épuisement de l'énergie d'un cluster-head peut être dû au fait qu'il coordonne un nombre élevé de membres. Par contre, dans un grand réseau, cela est dû au fait que le cluster-head nécessite une transmission de forte puissance pour communiquer avec la station de base.

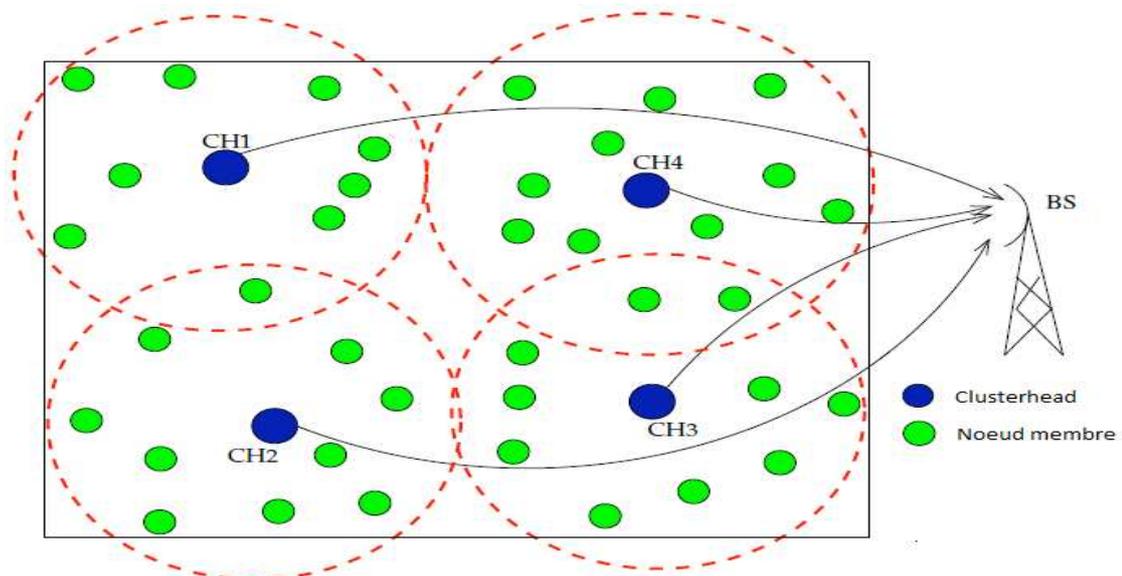


Figure2.5 Topologie basée sur LEACH.

Des variantes de LEACH ont été proposées afin de remplir ses vides, nous citons par la suite deux de ses variantes :

– M-LEACH (Multi-hop LEACH)

Il s'agit d'une variation de LEACH proposé par (Mhatre et Rosenberg, 2004). M-LEACH suppose que les membres d'un cluster peuvent être à plus d'1 saut de son cluster-head, de plus, il autorise des communications inter-cluster multi-sauts. M-LEACH ne suppose pas que tout cluster-head peut atteindre la station de base, cela permet de réduire l'énergie dissipée par cluster-head ainsi d'augmenter la stabilité de la structure par rapport à

LEACH. Par contre, M-LEACH ne résout pas le problème d'un nœud faible en énergie pouvant devenir cluster-head et fragiliser ainsi la robustesse de la structure.

– C- LEACH

Une autre amélioration de LEACH a été proposée par les auteurs dans (Heinzelman et al, 2002). C-LEACH inclut la quantité d'énergie résiduelle des nœuds dans le critère d'élection des cluster-heads. Par contre, cette approche est centralisée car l'ensemble du processus de clustering est contrôlé par la station de base.

2.4.2 Hybrid Energy-Efficient Distributed Clustering (HEED)

Proposé par O. YOUNIS et autres en 2004, HEED s'exécute d'une manière distribuée comme LEACH, à la différence qu'il utilise une métrique basée sur l'énergie résiduelle d'un nœud et le coût de la communication pour choisir les cluster-heads. HEED offre une meilleure distribution des cluster-heads dans le réseau comparé à LEACH, la probabilité que deux nœuds adjacents soient cluster-head est réduite. Les nœuds ne consomment pas leur énergie de la même façon.

HEED ajuste le rayon du cluster qui définit la puissance de transmission à utiliser pour une diffusion inter-cluster. Ses itérations s'exécutent sur trois phases:

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

$$CH_{prob} = C_{prob} * (ERésiduel / EMax);$$

Où :

–EMax : l'énergie initiale du nœud et ERésiduel est l'énergie restante.

–CHprob : toujours supérieure à un seuil p_{min} inversement proportionnel à EMax.

2. **Phase répétitive:** chaque nœud du réseau et après plusieurs itérations choisi son CH qui minimise le coût de transmission. Si aucun CH n'est entendu, il se déclare CH et annonce sa décision; et pour que l'algorithme converge le nœud double son CH_{prob} à chaque nouvelle itération jusqu'elle devienne égale à 1. De cette façon, le nœud est soit dans un état « Tentative status » si $CH_{prob} < 1$, soit il est dans un état « Final status » si $CH_{prob} = 1$.

3. **Phase finale:** chaque nœud décide définitivement de son statut; il est soit membre du cluster avec un coût minimal, soit il est CH.

Le protocole HEED vise à prolonger la durée de vie du réseau en prenant en considération l'énergie résiduelle. Cependant, la formation des clusters à un saut risque de générer un nombre important de clusters si le nombre de nœuds est important. Ainsi, on se rapproche d'une topologie plate de réseau. En plus, HEED prévoit aussi, une communication directe entre les CHs et la station de base ce qui oblige les CHs lointains à dépenser plus d'énergie et par conséquent HEED est considéré comme un protocole non favorable au passage à l'échelle.

2.4.3 Power-Efficient GATHERing in Sensor Information System (PEGASIS)

Power-Efficient GATHERing in Sensor Information Systems (PEGASIS) est une version améliorée de LEACH (Yousef Yaser) proposé par Lindsey et Raghavendra. Ce protocole forme des chaînes plutôt que des clusters, il se base sur la notion que chaque nœud communique avec ses voisins les plus proches, il ajuste donc sa portée radio pour des communications à courte distance afin de minimiser l'énergie consommée. Les données collectées sont transmises d'un nœud à un autre qui les agrège jusqu'à ce qu'elles arrivent à un nœud particulier qui les transmet à la station de base, cela garantit une meilleure conservation en énergie. Pour communiquer avec le puits, le processus est organisé en rounds; au cours de chaque round un seul nœud est autorisé à communiquer avec le puits directement. Ce privilège est accordé à l'ensemble des nœuds du réseau à tour de rôle. Les simulations ont montré que PEGASIS peut prolonger de deux à trois fois la durée de vie d'un réseau de capteurs que LEACH, cela est dû à l'agrégation de données qui diminuent le nombre de transmission, et à l'élimination de la phase de formation de clusters, coûteuse en termes d'énergie. Cependant, PEGASIS exige toujours un ajustement dynamique de la topologie puisqu'un nœud devrait connaître le niveau d'énergie de ses voisins avant de relayer ses données. Cela génère une surconsommation d'énergie importante surtout pour les réseaux à grande échelle. En plus, PEGASIS suppose que tous les nœuds du réseau peuvent atteindre la station de base ce que nécessite une transmission réglable avec un surcoût énergétique non négligeable. En plus, le délai de livraison des données est très important lorsque la chaîne formée est très longue.

2.4.4 Far-Legos (Far Low Energy Self-OrGanisatiOn Scheme)

Far-Legos propose une solution pour les capteurs qui ne se trouvent pas dans une zone d'activité d'un actionneur, cela est rendu possible en appliquant la stratégie de l'ensemble dominant dans ces régions non couvertes par des actionneurs, donnant ainsi naissance à un protocole hybride. Dans un premier temps, chaque actionneur affecte des rangs (gradients qui vont servir pour la collecte des données) pour les capteurs qui se situent dans sa zone d'activité, l'affectation de ces rangs sera assurée par une diffusion avec des puissances successives décrémentées comme le montre la figure ci-dessous. Ainsi les capteurs plus proches du nœud actionneur auront un plus petit rang.

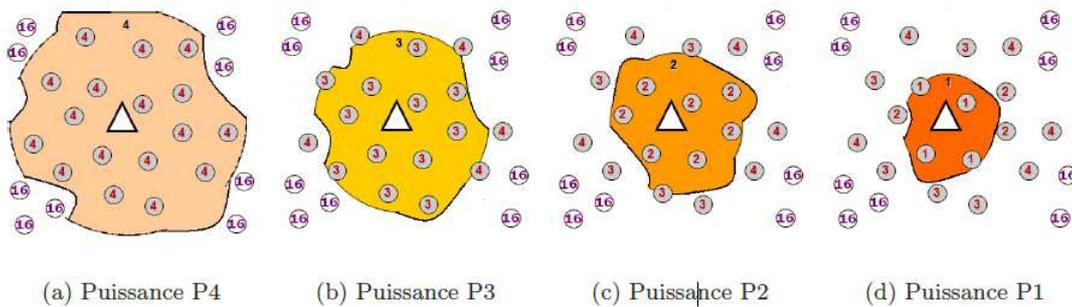


Figure 2.6 Exemple d'affectation de 4 rangs aux capteurs autour d'un actionneur [ROM 10].

Dans un second temps, une topologie logique est construite dans les zones non-couvertes par aucun actionneur. La topologie construite dans les zones non-couvertes se base sur le protocole d'auto-organisation Legos [LU 07]. En effet, Legos se base sur la construction d'une épine dorsale. Chaque nœud non couvert commence la construction de la topologie à $t = T_{start_Legos}$, calculé comme dans l'équation ci-dessous.

$$T_{start_Legos} = TimeOut_{NeighborDiscovery} + Rand(X) + TimeOut_{LeaderDetection}$$

Comme décrit dans la formule, chaque nœud capteur attend un $TimeOut_{NeighborDiscovery}$ qui correspond à la durée pendant laquelle un nœud capteur peut recevoir des messages *Hello* de la part des nœuds dans son voisinage. Puis, il attend un temps aléatoire qui dépendra de la situation du nœud dans la zone non-couverte. Par conséquent, les nœuds à proximité de la zone couverte ont une chance d'être les premiers à commencer la construction de la topologie. Quand au $TimeOut_{LeaderDetection}$, celui-ci correspond à la période séparant deux messages diffusés par les leaders de la topologie Legos [LU 07] dans la zone non couverte.

2.5 Conclusion

Dans ce chapitre nous avons étudié des techniques d'auto organisations parmi plusieurs techniques proposées dans la littérature. Nous avons essayé durant de noter le meilleur de ces techniques et les combiner avec d'autres pour avoir une approche qui permettra d'avoir de meilleurs résultats.

Chapitre 3

Vers une Nouvelle Approche d'Auto-Organisation

3.1 Objectif

Durant les chapitres précédents, nous avons tiré des connaissances sur les réseaux de capteurs sans fil. Dans ce qui suit, nous avons la base nécessaire pour contribuer dans ce domaine par la présentation d'une approche d'auto-organisation basée sur la technique de *clustering*.

L'objectif que nous nous sommes fixé est d'avoir une structure de réseau qui permet de consommer le moins d'énergie avec la meilleure distribution de cette consommation sur l'ensemble du réseau, et donc prolonger significativement la durée de vie de celui-ci. D'après ce que nous avons vu dans le chapitre 2, l'organisation en clusters permet de réduire la consommation en énergie qu'un réseau plat, en minimisant les communications intra-cluster sur de courtes distances, de plus l'utilisation des techniques telle que TDMA, réduit considérablement les collisions. L'agrégation de données qui se fait au niveau des cluster-heads réduit la quantité de données échangées dans le réseau. La plupart des protocoles proposés notamment ceux que nous avons étudié (HEED, LEACH, PEAGASIS...) supposent que tous les capteurs peuvent transmettre directement vers la station de base ; cette hypothèse reste vraie et valable seulement dans les réseaux à petite dimension, cependant dans des réseaux à grande échelle, une telle supposition s'avère trop coûteuse en terme d'énergie.

Dans notre travail, notre intérêt s'est porté sur le protocole LEACH, donc dans ce présent chapitre nous allons initialement présenter le protocole LEACH avant de présenter notre solution.

3.2 Déroulement du protocole Leach et ses limite

Nous allons dans un premier temps parler du protocole de routage hiérarchique basé clustering le plus populaire LEACH proposé par Heinzelman et Chandrakasan (2000) :

3.2.1 Présentation

LEACH est un protocole auto organisable hiérarchique basé sur la technique du clustering, il utilise une fonction aléatoire pour élire les cluster-heads afin de distribuer uniformément la consommation d'énergie sur l'ensemble des nœuds du réseau.

Dans LEACH les nœuds s'organisent pour former des clusters, chaque cluster est mené par un cluster-head. Dans certains algorithmes de clustering classiques, les cluster-heads sont choisis et fixés tout au long de la durée de vie du système, il est facile à constater que les

cluster-heads mal choisis épuisent leurs batteries rapidement, mettant fin à la durée de vie des nœuds du cluster. Afin de remédier à ce problème, LEACH adopte une fonction aléatoire périodique pour élire les cluster-heads, ce qui ne va pas engendrer l'épuisement rapide des batteries des nœuds, de plus dans LEACH, les données sans agrégées au niveau des clusters avant d'être délivré à la station de base, cela permet de réduire la quantité de données qui circulent dans le réseau. LEACH s'exécute en un ensemble d'itérations comme suit :

- Au début d'un round, chaque nœud se décide de devenir cluster-head indépendamment des autres nœuds du réseau, suivant une probabilité bien définie, il diffuse ensuite un message pour informer ses voisins de son statut. Chaque nœud décide auquel cluster appartenir en se basant sur l'énergie de communication minimale ;
- Une fois les clusters formés, chaque cluster-head organise son groupe en allouant à chaque membre un intervalle de temps dans lequel il peut transmettre ses données. En dehors de cet intervalle approprié, un nœud peut éteindre sa radio, question d'économiser son énergie. Une fois le cluster-heads à reçu toutes les données des nœuds de son groupe, il les agrège et envoie les données compressés vers la station de base. Etant donné que la station de base se situe loin des autres nœuds, donc sa communication consomme plus d'énergie. Toutefois comme il n'y a que quelques cluster-heads, cela touche un petit nombre de nœuds.

Afin d'équilibrer l'énergie consommée, les cluster-heads élus ne sont pas fixés (**figure 3.1** et **figure 3.2**), autrement dit si à un instant t l'ensemble des cluster-heads est C , à l'instant t' l'ensemble des cluster-heads est C' différent de C . La décision de devenir cluster-head dépend de la probabilité calculé au niveau du nœud, de cette façon seuls les nœuds ayant la plus grande probabilité sont élu CH.

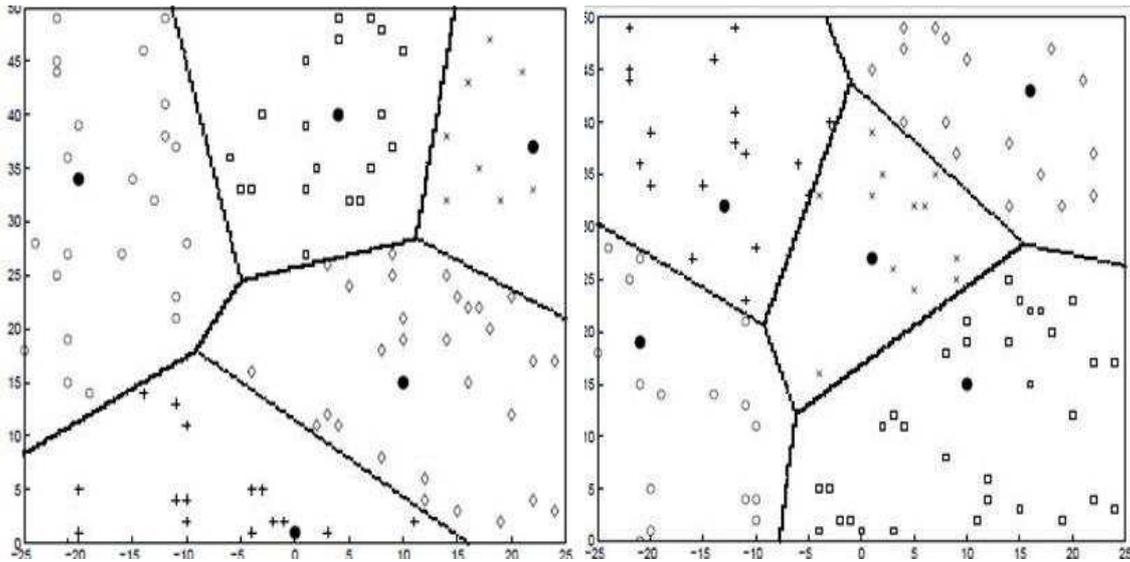


Figure 3.1 cluster-heads formés à l'instant t . Figure 3.2 cluster-heads formés à l'instant t' .

Le système fixe le nombre de clusters en fonction de quelques paramètres comme la topologie du réseau, le coût des opérations de communication et de calcul. Généralement le nombre moyen de cluster-heads représente 5% des nœuds du réseau.

LEACH suppose que chaque nœud du système peut atteindre la station de base en un seul saut. Les données sont donc communiquées des nœuds vers le cluster-head, et une fois l'agrégation est faite, le cluster-head les envoie directement vers la station de base.

3.2.2 Pourquoi se baser sur LEACH

Parmi ses points forts, on cite son mode d'exécution en round qui permet de définir les cluster-heads à tour de rôle, ce qui permet d'éviter le problème d'épuisement rapide des batteries de certains nœuds, donc de prolonger la durée de vie du réseau. De plus l'agrégation de données diminue la quantité de données circulées au sein du réseau.

Afin d'arriver à une nouvelle approche de clustering, nous avons choisi LEACH comme protocole de base.

LEACH est parmi les premiers protocoles de routage hiérarchiques conçus pour les réseaux de capteurs sans fils. La robustesse de ses notions de bases, l'a rendu très populaire. Certaines de ses notions notamment celles concernant l'agrégation de données et sa technique d'exécution en round ont été retenues, d'autres comme sa métrique de sélection d'un cluster-head et son algorithme de routage ont été améliorés. Ainsi, la majorité des travaux de recherche présentés juste après l'apparition de LEACH se sont appuyés sur.

3.2.3 Itérations de l'algorithme

L'algorithme LEACH est organisé en cycle, où chaque cycle débute par une phase **set-up** ; qui correspond à l'organisation des clusters, suivi d'une phase dite **steady-state**, qui correspond à l'envoi de données vers la station de base. C'est cette dernière phase qui permet d'exploiter l'auto-organisation de la première.

3.2.4 Phase d'élection des Cluster-Heads

Au début d'un round, chaque nœud se décide de devenir cluster-head. Cette décision dépend du pourcentage des cluster-heads dans le réseau (défini à priori), et le nombre de fois que le nœud en question a accompli le rôle d'un cluster-head. Le nœud n génère un nombre aléatoire entre 0 et 1, si ce nombre est inférieur à un seuil $T(n)$ le nœud devient cluster-head pour le round courant. Le seuil est fixé comme suit :

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

Où

- p le pourcentage des CHs sur le réseau (généralement 5%).
- r numéro du round en cours.
- G l'ensemble des nœuds qui n'étaient pas CH dans les $(1/P)$ rounds précédents.

En utilisant ce seuil, chaque nœud accomplira le rôle d'un cluster-head dans les prochains $1/P$ rounds. Durant le courant round, tous les nœuds ont la probabilité P pour devenir chef du groupe, ainsi les nœuds désignés cluster-head pour ce round ne peuvent pas être réélus dans les prochains $1/P$ rounds. La probabilité pour que les nœuds restants vont devenir cluster-head doit augmenter, du moment qu'il y'a moins de nœuds admissibles. Après $1/P$ round tous les nœuds sont passés par le rôle cluster-head.

Chaque nœud élu cluster-head, diffuse un message **ChAdv** pour le reste des nœuds. En utilisant le protocole CSMA, l'énergie de transmission est la même pour tous les cluster-heads, les nœuds non cluster-head doivent garder leurs récepteur en écoute afin de recevoir les messages **ChAdv**. Une fois tous les messages **ChAdv** sont envoyés, chaque nœud décide à quel cluster appartenir durant ce round, cette décision est fondée sur la puissance du signal avec lequel le message **ChAdv** a été reçu, ainsi le **ChAdv** ayant le plus fort signal, nécessite

le moins d'énergie pour être communiqué donc il est le plus approprié à être choisi. En cas d'égalité de forces des signaux, un sera choisi aléatoirement.

3.2.5 Phase de formation des clusters

Après avoir décidé à quel cluster appartenir, un nœud doit informer son cluster-head de sa décision en lui envoyant un message.

3.2.6 Ordonnancement intra-cluster

Le cluster-head reçoit tous les messages des nœuds qui ont choisi son groupe, en se basant sur la cardinalité du cluster formé, il crée une table TDMA ou il alloue pour chaque membre un intervalle de temps dans lequel il peut transmettre ses données, afin d'éviter la collision intra-cluster. Cette table sera diffusée sur tous les membres

3.2.7 Transmission de données

Une fois les clusters créés et les tables TDMA sont construites, la transmission de données peut commencer. On suppose que tous les nœuds ont des données à transmettre, ils les transmettent pendant leurs intervalles de temps appropriés aux cluster-heads. Un nœud non cluster-head peut mettre sa radio en OFF si son tour n'est pas encore arrivé, cela lui permet d'économiser son énergie. Par contre un nœud cluster-head doit toujours garder son récepteur actif. Une fois qu'il a reçu toutes les données des autres nœuds, il effectue des traitements afin d'avoir des données compressées qui seront envoyées en un seul signal à la station de base.

3.2.8 Limite du protocole Leach

Plusieurs critiques sont apportés au protocole LEACH relatives à ses notions de base :

- LEACH suppose que chaque nœud peut atteindre la station de base, cela devient coûteux en énergie lorsque le nœud se trouve très loin ce qui rend le protocole moins apte pour le passage à l'échelle ;
- L'agrégation faites au niveau des cluster-heads engendre une surconsommation de l'énergie ;
- LEACH ne garantis pas une distribution homogène des cluster-heads sur le réseau, car le seul critère d'élection du cluster-head est une probabilité aléatoire. Cela n'empêche pas une concentration des cluster-heads dans une région limitée au détriment de l'ensemble du réseau ;

- La rotation du rôle de cluster-head génère une surconsommation d'énergie, car chaque rotation de cluster-head nécessite une phase de diffusion pour faire connaître le nouveau cluster-head.

3.3 EDBUC (*Energy and Density Based Unequal Cluster*)

Après avoir étudié le protocole LEACH, et cerné ses insuffisances, nous avons pensé à un nouvel algorithme d'auto-organisation basé sur LEACH que nous avons nommé EDBUC (*Energy and Density Based Unequal Clustering*).

EDBUC utilise la densité d'un nœud et son niveau d'énergie résiduelle comme métrique d'élection des cluster-heads. Cette métrique a deux grands avantages, le 1er étant que l'utilisation de la densité permet d'avoir un nombre réduit de clusters et uniformément distribués sur le réseau. Le deuxième intérêt de cette métrique est que l'utilisation de l'énergie permet d'élire des nœuds ayant la plus grande quantité d'énergie pour accomplir le rôle d'un cluster-head ce qui permet d'une part d'équilibrer la consommation d'énergie sur le réseau, d'autre part économiser l'énergie du réseau et prolonger sa durée de vie. EDBUC contrôle la cardinalité des clusters formés en se référant au rayon entre le cluster-head et la station de base. En effet, la taille des clusters diminue en s'approchant de la station de base.

3.3.1 Hypothèses

Notre algorithme reprend certaines hypothèses de LEACH. Afin de ne pas reprendre les défauts de ce dernier, nous avons ajouté quelques hypothèses :

- Le réseau comporte une seule station de base et qui est stationnaire.
- Les nœuds sont déployés aléatoirement ;
- Chaque nœud connaît ces coordonnées ainsi sa distance vers la station de base ;
- EDBUC adopte la politique « Round-Robin », qui consiste à répéter la procédure de formation de cluster-head à chaque intervalle de temps prédéfini. Afin de mettre à jour les cluster-head et avoir une consommation d'énergie équilibrée ;
- EDBUC adopte une transmission directe du CH vers la station de base;
- Le nombre de cluster est indéfini au préalable ;
- Un nœud peut contrôler sa puissance d'émission ;

3.3.2 Modèle d'énergie utilisée

Suite à la réception ou à l'émission d'un message, un nœud doit mettre à jour son niveau d'énergie. L'énergie consommée à la réception est beaucoup plus inférieure à la quantité consommée à l'émission.

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

$$E_{Tx}(k, d) = E_{elec}(k) + E_{amp}(k, d) = E_{elec} * k + E_{amp} * k * d^{\alpha}$$

Et le récepteur consomme:

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

Avec :

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

k : Taille d'un message en bits.

d : Distance entre l'émetteur et le récepteur.

E_{amp} : Energie d'amplification.

α : Sa valeur dépend des conditions de l'environnement. Elle est comprise entre 2 et 6 .

On considère les paramètres suivants :

3.3.3 Itérations de l'algorithme EDBUC

EDBUC s'exécute en round, chaque round comporte deux phases. Durant la 1ere phase on élit les cluster-heads et on forme les groupes. La 2eme phase de EDBUC est identique à la 2eme phase de LEACH qui comporte une étape d'ordonnancement intra-clusters et une étape de transmission de données:

3.3.3.1 Phase de formation de cluster

Dans cette phase, la station de base émet un message pour déclencher un nouveau round. A la réception de ce message chaque nœud envoie un message Hello pour découvrir son voisinage. Une fois tous les voisins sont connus chaque nœud calcule son poids et le transmet vers ses voisins dans un message dit **WeightMsg** afin de déterminer les cluster-heads. A la fin de cette phase, le rôle de chaque nœud est déterminé, un nœud peut être

cluster-head ou nœud membre (nœud ordinaire), et les groupes sont formés. Cette phase peut être découpée en plusieurs étapes :

3.3.3.2 Découverte de voisinage

Au début d'un round, la station de base diffuse un message NewRound. A la réception de ce message, chaque nœud calcule son rayon R_i qui correspond à sa portée de transmission comme suit :

$$R_i = (1 - C * ((D_{max} - d(i, SB)) / (D_{max} - D_{min})) * R_{max}$$

Avec :

C : étant une constante appartenant à l'intervalle $[0, 1]$;

D_{max} : la distance maximale par rapport à la station de base ;

D_{min} : la distance minimale par rapport à la station de base ;

$d(i, SB)$: la distance entre le nœud i et la station de base ;

R_{max} : est la portée maximale d'un nœud

Le nœud envoie par la suite un message Hello contenant son identifiant, avec une puissance d'émission qui correspond à la déperdition du signal sur le rayon calculé.

3.3.3.3 Déperdition du signal

Consiste à estimer la puissance du signal émis par chaque nœud en fonction de la distance entre l'émetteur et le récepteur en question selon la formule :

$$PLD = PLD_0 + 10 * PATH_{LOSS_{EXPONENT}} * \log_{10} \left(\frac{d}{d_0} \right) + normal(0, \sigma)$$

$$RSSI_{dBm} = TSSI_{dBm} - PLD$$

Avec

$RSSI_{dBm}$: La puissance du signal à la réception en dB

$TSSI_{dBm}$: La puissance de transmission

PLD : path-loss, la perte en dB pour une distance d de l'émetteur

d_0 : Distance de référence

$PATH_{LOSS_EXPONENT}$: indique le taux avec lequel le signal s'atténue en fonction de la distance.

$normal(0, \sigma)$: la loi normale centrée avec un écart-type σ .

3.3.3.4 Structure d'un message Hello :

| Field | Size |
|-------------|------|
| Identifiant | 16 |

Tableau3.1 Structure d'un message Hello

3.3.3.5 Election des cluster-heads

Chaque nœud qui reçoit un message Hello, construit une table de voisinage qui contiendra l'identifiant du nœud émetteur. Ensuite, il calcule son poids et le diffuse par le biais d'un message **WeightMs**. La formule qui permet de calculer ce poids est :

$$\text{Poids} = a * (E_j / E_{\text{max}}) + b * (D_j / D_{\text{max}}).$$

Avec : $a + b \leq 1$;

E_j : l'énergie résiduelle du nœud j ;

E_{max} : l'énergie initiale du nœud ;

D_j : la densité du nœud j ;

D_{max} : la taille maximal du réseau ;

3.3.3.6 Structure d'un message WeightMsg.

| Field | Size |
|------------|------|
| IDEmetteur | 16 |
| Weight | 64 |

Tableau3. 2 Structure d'un message WeightMsg.

A la réception d'un message WeightMsg, le nœud ajoute un champ Wheight a sa table de voisinage qui va contenir la valeur du poids correspondant à l'identifiant de l'émetteur dans la table. Chaque nœuds compare son propre poids a celui de ses voisins dans le cas ou il possède le plus grand poids, il s'auto-élit cluster-head. L'élection des cluster-heads s'arrête au bout d'un moment prédéfini, les CHs émettent un message CHadv pour informer les autres nœuds de leurs statuts CH, ainsi les inviter à joindre leurs groupes. Un nœud non cluster-head doit garder son récepteur en écoute afin de recevoir les messages CHadv. Une fois tous les messages CHadv reçus, chaque nœud décide a quel cluster appartenir durant ce round, cette décision est fondée sur la puissance du signal avec lequel le message CHadv à été reçu, ainsi le CHadv ayant le plus fort signal, nécessite le moins d'énergie pour être communiqué donc il est le plus approprié à être choisi. En cas d'égalité de forces des signaux, un sera choisi aléatoirement.

3.3.3.7 Structure d'une table de voisinage

| ID du nœud | Poids |
|------------|-------|
|------------|-------|

Tableau3.3 Structure d'une table de voisinage.

3.3.3.8 Structure d'un message CHadv

| Field | Size |
|------------|------|
| IDEmetteur | 16 |

Tableau3.4 Structure d'un message CHadv.

3.3.3.9 Formation des clusters

Après avoir décidé à quel cluster appartenir, un nœud doit informer son cluster-head de sa décision en lui envoyant un message **JoinClusterDecision**.

3.3.3.10 Structure d'un message JoinClusterDecision

| Field | Size |
|----------------|------|
| ID source | 16 |
| ID destination | 16 |

Tableau3. 5 Structure d'un message JoinClusterDecision.

3.3.3.11 Phase de communication des données

Cette phase comporte deux étapes :

– Ordonnancement intra-cluster

Un cluster-head reçoit tous les messages **JoinClusterDecision** des nœuds qui ont choisi son groupe, en se basant sur la cardinalité du cluster formé, il crée une table TDMA ou il alloue pour chaque membre un intervalle de temps dans lequel il peut transmettre ses données, afin d'éviter les collisions intra-cluster. Cette table sera diffusée sur tous les membres.

– Transmission de données

Une fois les clusters créés et les tables TDMA sont construites, la transmission de données peut commencer. On suppose que tous les nœuds ont des données à transmettre, ils les transmettent pendant leurs intervalles de temps appropriés aux cluster-heads. Un nœud non cluster-head peut mettre sa radio en OFF si son tour n'est pas encore arrivé, cela lui permet d'économiser son énergie. Par contre un nœud cluster-head doit toujours garder son récepteur actif. Une fois qu'il a reçu toutes les données des autres nœuds, il effectue des traitements afin d'avoir des données compressées qui seront envoyées en un seul signal à la station de base.

3.4 Diagramme de l'algorithme EDBUC

Le diagramme suivant résume les étapes de l'algorithme EDBUC

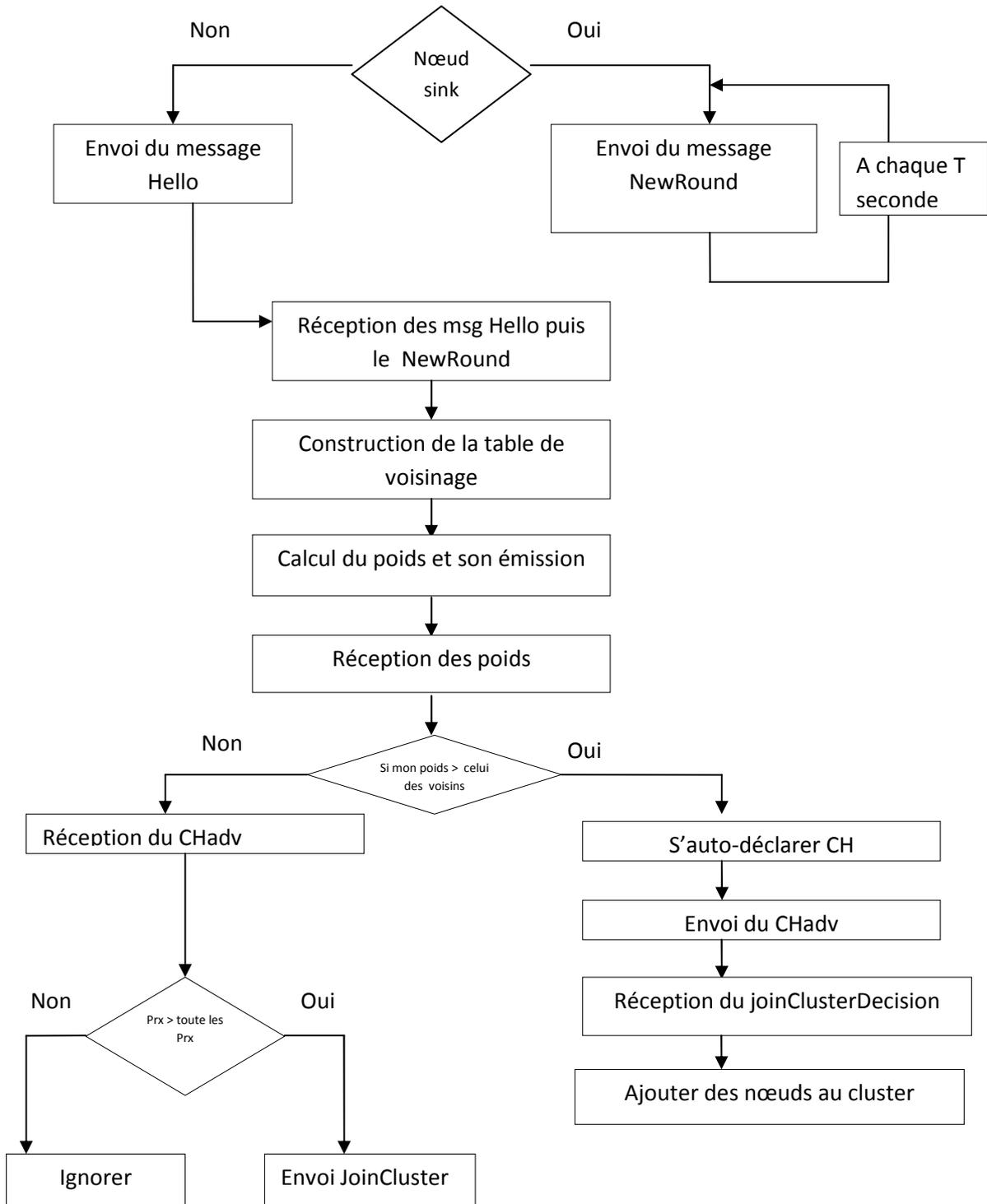


Figure3. 3 Diagramme représentant la construction de cluster.

3.5 Conclusion

Dans ce chapitre nous avons étudié LEACH, souligné ces insuffisances, et nous avons présenté un nouvel algorithme basé sur LEACH que nous avons nommé EDBUC. L'objectif de notre algorithme est d'assurer une consommation d'énergie plus efficace par rapport à LEACH, donc assurer une durée de vie du réseau plus grande, et avoir une meilleure distribution des CH au sein du réseau. Pour cela, nous avons choisis les nœuds ayant une grande quantité d'énergie et un nombre élevé de voisins pour accomplir le rôle d'un CH. A la fin de la présentation de l'algorithme, nous avons fait une comparaison qui montre l'amélioration apportée à LEACH en terme de consommation d'énergie et de répartition des clusters et de ses membres ce protocole sera une base pour l'implémentation d'un routage multi saut.

Prochainement nous allons implémenter et simuler EDBUC, sous OMNeT++.

Chapitre 4

Simulation

4.1 Introduction

Une des étapes les plus cruciales dans le développement de notre approche est la simulation.

Dans cette section on prendra soin d'étaler le procédé suivi dans la simulation de notre algorithme et les différents critères de performance.

4.2 Etude de quelques simulateurs existants

Dans ce qui suit, quelques environnements de simulation utilisés pour évaluer les performances des protocoles et des architectures dans un réseau sans fils sont présentés. Certains sont dédiés aux WSNs, d'autres sont plus généraux (conçus pour les réseaux classiques) mais peuvent être adaptés pour les WSNs. Aucun des simulateurs présentés (libres ou commercialisés) n'est parfait. Chacun présente des avantages et des inconvénients.

4.2.1 TOSSIM

TOSSIM est un simulateur discret basé sur la programmation par événements. Il a été conçu pour simuler les réseaux de capteurs en utilisant la plateforme TinyOS.

4.2.1.1 TinyOS

TinyOS est un système d'exploitation open-source conçu pour les applications embarquées fonctionnant en réseaux, et en particulier, pour les réseaux de capteurs sans-fil. Actuellement, il représente le système d'exploitation le plus connu dans le monde des réseaux de capteurs sans fils et il est utilisé par plus de 500 universités et centres de recherches à travers le monde. Ce système, conçu et maintenu par l'université de Berkeley et de nombreux contributeurs, est orienté "composants" afin de faciliter l'implémentation des WSNs, tout en minimisant la taille du code afin de respecter les contraintes de mémoire.

L'avantage principal de *TinyOs* est sa taille extrêmement réduite en termes de mémoire. Il ne requiert que quelques kilo-octets (3500 octets pour le code du système, 4500 octets pour d'autres programmes et applications associées) car il se base sur une association de composants, réduisant la taille du code nécessaire à sa mise en place. Un autre point fort de *TinyOS* réside dans sa bibliothèque de composants. Elle est particulièrement complète puisqu'on peut retrouver des protocoles de réseaux, des pilotes de capteurs et des outils d'acquisition des données. L'ensemble de ces composants peut être utilisé tel quel ou être adapté à une application précise.

La conception de TinyOS a été entièrement réalisée en NesC, un langage orienté composant syntaxiquement très proche du C.

4.2.1.2 TOSSIM

TOSSIM [LEV 03] est un simulateur discret basé sur la programmation par événements. Il a été conçu afin de simuler les réseaux de capteurs qui utilisent la plateforme TinyOS. Il offre un comportement très proche de ce qui se passe dans le monde réel avec ce type de réseau.

TOSSIM fournit plusieurs modèles de communication. Au niveau radio, deux modèles peuvent être utilisés :

- Le modèle ‘simple’ propose une communication idéale sans erreurs de transmission ;
- Par contre, le modèle ‘lossy’ prend en considération les erreurs de transmission et la perte de paquets.

TOSSIM est souvent utilisé avec une interface graphique (TinyViz) pour une meilleure compréhension et visualisation de l’état du réseau. Cette application graphique, réalisée en JAVA, donne un aperçu du réseau de capteurs à tout instant, ainsi que des divers messages émis. Elle permet de déterminer un délai entre les itérations des capteurs afin de permettre une analyse pas à pas du déroulement des actions.

👉 Avantages

- TOSSIM simule fidèlement le comportement d’un réseau de capteurs utilisant la plateforme TinyOS.
- L’interface graphique TinyViz permet la visualisation des échanges radios, conjointement aux messages de débogage ; ce qui permet d’avoir une vue globale de l’activité du réseau à tout instant.
- TinyViz offre la possibilité de ralentir la simulation par un délai afin d’observer le déroulement des événements, ce qui est très intéressant lorsque le réseau est surchargé de messages.

👉 **Inconvénients**

- L'inconvénient principal de TOSSIM est qu'il exige l'exécution du même code pour tous les capteurs. Afin de pallier à ce problème, les capteurs doivent être identifiés par des numéros et des conditions doivent être utilisées dans le code pour différencier le rôle de chaque capteur.
- Le manque de souplesse et d'extensibilité.
- Le simulateur et l'interface de visualisation ont été conçus séparément.

4.2.2 NS2

NS est le simulateur à événements discrets le plus répandu dans le domaine des réseaux. L'appellation « NS-2 » est utilisée pour désigner la deuxième version du simulateur NS. Il constitue un très bon support pour la description et la simulation des réseaux IP (sans fil ou filaires) avec des protocoles des couches réseau, transport et application.

Le simulateur NS2 est écrit en C++ avec une interface textuelle (ou shell) utilisant le langage OTcl (Object Tool Command Language) qui représente l'extension objet au langage de commande Tcl. Le langage C++ sert à décrire le fonctionnement interne des composants de la simulation (définir les classes). Quant au langage OTcl, il fournit un moyen flexible et puissant de contrôle de la simulation comme le déclenchement d'événements, la configuration du réseau, les caractéristiques des liens physiques, la collecte de statistiques, etc.

NS bénéficie de toutes les possibilités qu'offrent les techniques objets comme l'héritage, le polymorphisme, la surcharge, etc

L'outil NAM (Network Animator) associé au simulateur NS permet de visualiser des animations de la simulation (transfert des paquets d'un nœud à un autre, taille des paquets, remplissage des files d'attente ...).

👉 **Avantages**

- NS est un logiciel de simulation multicouche.
- Son développement est orienté objet.
- C'est un outil complètement libre et disponible pour plusieurs plateformes (windows, UNIX, etc).

- Vu sa popularité, plusieurs protocoles sont à priori disponibles, aussi bien pour les réseaux ad-hoc que pour les réseaux filaires.
- Permet l'ajout de composants à la demande.

Inconvénients

- Initialement conçu pour les réseaux filaires, le support des réseaux sans fil a été ajouté ultérieurement.
- Il existe peu de paramètres de configuration dans les modèles prédéfinis.
- Les performances du simulateur sont assez limitées et ne semblent pas adaptées à des simulations de réseaux importants (d'après NS-2 est très lent lorsque le nombre de nœuds dépasse 100).
- Difficulté d'ajouter de nouveaux modèles à cause des dépendances entre modules.
- Difficile à intégrer dans d'autres applications.
- L'analyse des résultats est en général peu aisée, car ils sont représentés dans un seul fichier retraçant l'ensemble des envois, réceptions et suppressions de paquets.
- Il n'est pas destiné aux réseaux de capteurs sans fils et peu de modèles existent pour ce type de réseau.

4.2.3 GloMoSim

GloMoSim (Global Mobile Simulator) est un environnement de simulation pour les réseaux à grande échelle. Développé à UCLA (University of California, Los Angeles), ce simulateur exploite le parallélisme offert par le langage PARSEC (PARallel Simulation Environment for Complex systems) qui est une extension du langage C. Il donne la possibilité d'exécuter un modèle de simulation avec plusieurs protocoles dans différentes architectures parallèles.

Actuellement, GloMoSim supporte des protocoles pour les réseaux purement sans fil et dans le futur proche, de nouvelles versions pourront simuler un réseau filaire aussi bien qu'un réseau hybride.

Comme le montre la figure 4.1, GloMoSim est conçu selon une architecture basée "couches". Cette architecture est similaire à celle du modèle OSI. Un certain nombre de protocoles prédéfinis ou développés par l'utilisateur peuvent être utilisés au niveau de chaque couche.

Afin de garantir la liaison, GloMoSim définit des APIs communes entre les couches voisines. Chaque API spécifie les services et les paramètres échangés entre les couches voisines.

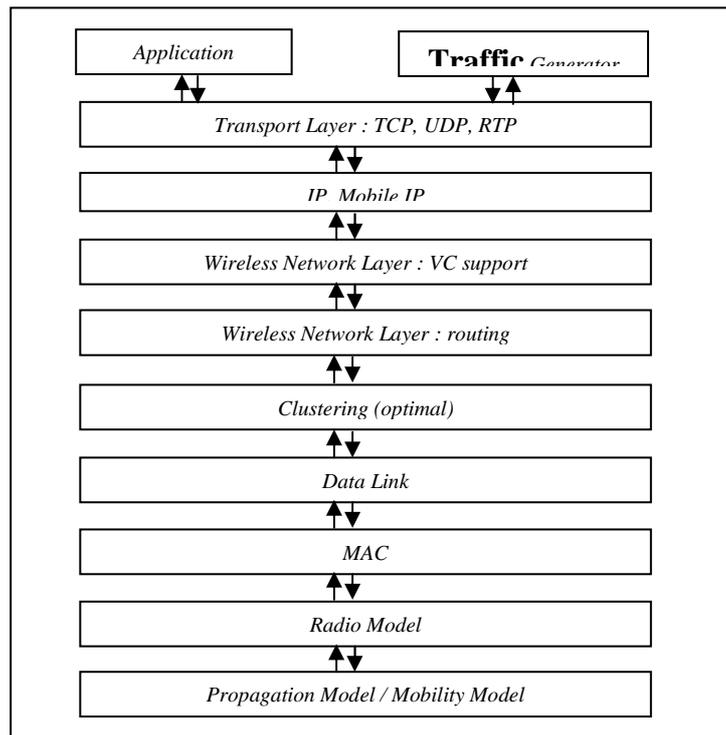


figure 4.1: L'architecture de GloMosim [ZEN 98].

👉 **Avantages**

- Un logiciel de simulation multicouche.
- Plusieurs protocoles sont prédéfinis au niveau de chaque couche.
- GloMoSim est plus rapide en termes de temps d'exécution que NS2.

👉 **Inconvénients**

- afin de pouvoir exécuter un modèle de simulation sous GloMoSim, il est nécessaire d'avoir quelques connaissances de PARSEC. L'apprentissage de ce langage peut se révéler difficile.

- GloMoSim est peu documenté.

4.2.4 OMNeT++ Framework [BOU 12]

Développé par András Varga [VAR 05], Objective Modular Network Test-bed in C++ (OMNeT++) est un simulateur libre à événements discrets. Il est basé "composants" et propose un framework de simulation modulaire.

OMNeT++ était initialement conçu pour simuler les communications réseaux. Cependant, grâce à son architecture générique et flexible, il est actuellement utilisé avec succès dans plusieurs autres champs tels que la simulation de systèmes complexes de traitement et de communications de données, la simulation de réseaux de files d'attente, la modélisation des systèmes multiprocesseurs et pas mal d'autres systèmes distribués.

👉 Avantages

- OMNeT++ est basé sur une approche modulaire qui facilite énormément la combinaison et la réutilisation des modèles conçus.
- L'approche orientée-objet permet l'extension des classes de base du noyau de la simulation (via l'héritage).
- Le développement sur OMNeT++ se fait en langage C++. Ce langage est très connu. De plus il facilite l'intégration d'OMNeT++ avec d'autres environnements de développement.
- Les bibliothèques d'OMNeT++ sont très riches en termes de supports d'entrées/sorties, de manipulation des statistiques, de représentation graphique, de génération des nombres aléatoires, etc.
- Le langage NED utilisé pour la description des modules est très simple. De plus, OMNeT++ offre une interface graphique (GNED) pour la construction des modules avec une génération automatique de la description NED.
- OMNeT++ permet la visualisation de l'état du réseau et les messages échangés à tout instant. De plus, il offre deux interfaces graphiques pour la visualisation des statistiques (Plove and Scalars).
- Le chargement des paramètres de simulation peut être dynamique grâce à l'utilisation des fichiers *.ini

👉 Inconvénients

- OMNeT++ n'est pas conçu pour les réseaux de capteurs sans fil. Il faut donc l'enrichir avec les modèles adéquats à ce type de réseau.

- Plusieurs concepteurs s'orientent vers OMNeT++ afin de l'enrichir avec des modèles de simulation destinés aux WSNs. Cependant, ces travaux sont généralement indépendants et il n'y a aucune plateforme qui les regroupe.

4.3 Environnement de simulation choisi OMNeT++

4.3.1 Présentation

OMNeT++ (Objective Modular Network Test-bed in C++) est un environnement de simulation à événement discret, modulaire et orienté objet basé sur le C++. Son architecture générique, a permis son utilisation dans divers domaines:

- Modélisation de réseaux de communication filaires et sans fil ;
- Modélisation de protocoles ;
- Modélisation des files d'attente réseaux ;
- Modélisation des multiprocesseurs et d'autres systèmes matériels distribués ;
- La validation d'architectures matérielles ;
- L'évaluation des aspects de la performance des systèmes logiciels complexes ;
- En général, la modélisation et la simulation d'un système où l'approche d'événements discrets est adaptée, et qui peut être avantageusement cartographié en entités qui communiquent par l'échange des messages.

OMNeT++ est un projet open source dont le développement a commencé en 1992 par Andras Vargas à l'université de Budapest. Actuellement, Ce simulateur est utilisé par des dizaines d'universités pour la validation de nouveaux matériels et logiciels, ainsi que pour l'analyse de performance et l'évaluation de protocoles de communication.

4.3.2 Architecture

Un modèle OMNeT++ est assemblé à partir de composants réutilisables dits modules. Les modules sont soit simples ou composés à partir d'autres modules simples ou composés. Ils peuvent être reliés entre eux par des portes (d'autres systèmes pourraient appeler ports), ou combinés pour former des modules composés. La profondeur d'un module composé n'est pas limitée. Les feuilles d'un modèle sont des modules simples qui représentent les classes C++. Pour chaque module simple correspond un fichier .cc et un fichier .h.

Les modules communiquent par envoi de messages pouvant représenter des paquets, des trames d'un réseau informatique, des clients dans une file d'attente ou bien d'autres types d'entités en attente d'un service. Les messages sont envoyés via les portes, mais il est également possible de les envoyer directement vers leurs modules de destination, celui-ci est utile pour les simulations sans fils. Les portes représentent l'entrée et l'interface de sortie des modules. Les messages sont envoyés à travers les portes de sortie et arrivent par les portes d'entrée. Une porte d'entrée et la porte de sortie peuvent être reliées par une connexion.

En OMNeT, un modèle entier est appelé réseau, qui est lui-même un module composé réutilisable. La conception d'un réseau se fait dans un fichier .ned et les différents paramètres de chaque modules sont spécifiés dans un fichier .ini. OMNET++ génère à la fin de chaque simulation deux nouveaux fichiers omnet.vec et omnet.sca qui permettent de tracer les courbes et calculer des statistiques.

Dans la figure 4.2 les boîtes représentent des modules simples (fond gris) et des modules composés. Les flèches reliant les petites boîtes représentent les connexions et les portes.

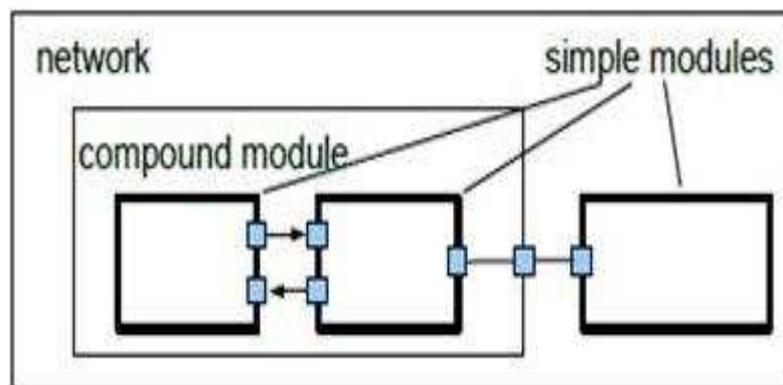


Figure4. 2 Modules composés, modules simples et les connexions entre les modulesConception en NED.

L'utilisateur définit la structure du modèle (les modules et leurs interconnexions) par le biais du langage NED. Les composants d'une description NED sont les déclarations des modules simples, les définitions des modules composés et des réseaux.

La déclaration d'un module simple décrit les interfaces de ce module et ses paramètres Les Définitions des modules composés comportent la déclaration des interfaces externes du module (lesportes et les paramètres), et la définition de ses sous-modules et leurs

interconnexions. La définition d'un réseau est un module composé qualifié comme un modèle de simulation autonome.

4.3.3 La librairie MF

La librairie MF est une extension du simulateur OMNET++. Elle a été développée par une équipe de chercheurs à l'université de Berlin. La dernière version a été proposée par Marc Loebbers en Octobre 2003. Il est un bon support pour la simulation des réseaux sans infrastructure et mobile. Il peut être utilisé pour la simulation de :

- réseaux sans fils,
- réseaux mobiles,
- réseaux Ad-hoc.

4.3.4 La librairie INET

Dans ce paragraphe, nous allons présenter une étude de l'existant de la librairie INET.

Nous allons décrire plus précisément l'implémentation des couches PHY, MAC, IP et RTP dans INET.

4.3.5 La couche PHY

Andras Varga s'est basée par l'implémentation de la couche PHY faite dans MF par Marc Lœbbers Dans la librairie INET version Octobre 2006 la couche physique IEEE 802.11b implémenté. Le calcul au niveau PHY de la puissance reçue se fait à l'aide de la formule de Friis, c'est-à-dire que seul le modèle Free-Space est implémenté comme modèle de propagation.

4.3.6 La couche MAC

La couche MAC IEEE 802.11b est implémentée dans la librairie INET, elle supporte le mécanisme RTS/CTS. Il n'y a que l'algorithme DCF implémenté, le PCF n'est pas implémenté. Le débit de transmission physique est constant pendant la simulation. Aucun des algorithmes d'adaptation du débit de transmission physique n'est implémenté. Les paquets multipoints au niveau MAC sont envoyés de la même manière que les paquets broadcast. Le filtrage de ces paquets se fait au niveau IP.

4.3.7 La couche IP

Le routage multicast au niveau IP est déjà implémenté mais de manière statique. C'est à dire que l'utilisateur doit configurer les adresses IP et les adresses de groupes multicast dans

les tables de routages avant la simulation. C'est au niveau IP que se passe le filtrage des paquets multicast.

4.3.8 La couche RTP

La couche RTP n'est pas encore intégrée dans la librairie INET. Dans la version 20061020, il y a une implémentation de la couche RTP réalisé par Matthias Opptiz et ajouté par Andras Vargas mais elle n'est pas intégrée. Le problème de cette implémentation est qu'elle a été faite dans une ancienne version de l'année 2001 et que l'architecture globale de la librairie INET a changé.

4.3.9 La couche Application

De nombreuses applications sont implémentées dans la librairie INET qui utilise soit le protocole UDP ou le protocole FTP.

4.4 Résultats des simulations

4.4.1 Critère de performances

– Consommation d'énergie

La consommation d'énergie dans les réseaux de capteurs est un paramètre fondamental lors de la phase des tests. Elle concerne essentiellement les communications (transmissions et réceptions).

– Durée de vie du réseau

La durée de vie d'un RCSF est définie en utilisant trois métriques FND (First Node Dies), HNA (Half of the Nodes Alive), et LND (Last Node Dies). Dans la plupart des applications, la durée de vie du réseau suit la métrique LND i.e. le temps qui s'écoule jusqu'à ce que le dernier capteur cesse de fonctionner. Cependant, cette métrique ne devrait pas être utilisée dans certaines applications critiques telles que les applications de détection et de surveillance d'intrusion, dans lesquelles le taux de couverture de la zone d'intérêt devrait être assez élevé. Ainsi, pour que le réseau accomplisse ses missions pour lesquelles il a été déployé avec succès, nous exigeons souvent que le taux de capteurs vivants soit supérieur à une certaine valeur seuil.

Dans notre travail, nous avons évalué la durée de vie d'un réseau de capteurs dans un seul contexte. Ce contexte suit la métrique LND i.e. le pourcentage des nœuds restant en vie.

– **Le nombre de Cluster Formé par Round**

L'énergie résiduelle est liée directement au le nombre de clusters formés de cela ce critère est pertinent dans notre comparaison.

– **Ecart type de l'énergie consommée (Standard deviation)**

L'**écart type**, aussi orthographié **écart-type** (en anglais *standard deviation*), est une notion mathématique définie en probabilités et appliquée à la statistique. En probabilités l'écart type est une mesure de la dispersion d'une variable aléatoire réelle ; en statistique il est une mesure de dispersion de données. Il est défini comme la racine carrée de la variance. Il a la même dimension que la variable aléatoire ou de la variable statistique en question.

Il se calcule dans notre cas de la manière suivante :

$$\mathbf{EnrRes} = \mathbf{EnrRes} - (\mathbf{E_elec} + \mathbf{EnrSpent}) \dots (1)$$

EnrRes : l'énergie résiduelle.

EnrSpent : l'énergie consommée lors d'une réception ou transmission.

$$\mathbf{TotalEnrRes} = \mathbf{TotalEnrRes} + \mathbf{EnrRes} \dots (2)$$

TotalEnrRes : est la somme des énergies résiduelles au niveau des nœuds capteurs.

$$\mathbf{AvgES} = \sum_{i=1} (\mathbf{1000.00} * (\mathbf{numNodes} - 1) - \mathbf{TotalEnrRes}) / (\mathbf{numNodes} - 1) \dots (3)$$

numNodes : le nombre de nœuds.

1000.00 : est l'énergie initiale au début de la simulation mesuré en Mili Joules.

$$\mathbf{Sd} = ((\mathbf{1000.00} - \mathbf{TotalEnrRes}) - \mathbf{AvgES}) * ((\mathbf{1000.00} - \mathbf{TotalEnrRes}) - \mathbf{AvgES}) \dots (4)$$

TotalEnrRes : l'énergie résiduelle calculée dans la formule (2).

AvgES : Moyenne d'énergie consommée évaluée dans la formule (3).

1000.00 : est l'énergie initiale de chaque nœud au début de la simulation.

4.4.2 Scénario de simulation

La simulation se déroulera de la manière suivante :

- Définition des topologies de réseaux constitués de (25, 50, 75, 100, 125, 150, 175 leurs déploiement est fait aléatoirement).
- Les deux protocoles (LEACH, EDBUC) ont été simulés sur les mêmes topologies définies au préalable.

- Pour les formules qu'on a utilisées on a pris les valeurs suivantes :

La formule de calcul du poids :

$$\text{Poids} = a * (E_j / E_{\text{max}}) + b * (D_j / D_{\text{max}}).$$

$$a = 0,6;$$

$$b = 0,4;$$

La formule qui permet de calculer le rayon:

$$R_i = (1 - C * ((D_{\text{max}} - d(i, SB)) / (D_{\text{max}} - D_{\text{min}}))) * R_{\text{max}}$$

Avec :

$$C = 0,7 ;$$

$$D_{\text{max}} = 282;$$

$$D_{\text{min}} = 1 ;$$

$$R_{\text{max}} = 50 ;$$

- La moyenne d'énergie consommé en 200 secondes ce qui est l'équivalent de 20 round;
- L'écart type en 200 secondes;
- La durée de vie a été étudiée sur une topologie de 25 nœuds durant 200 secondes ;

4.4.3 Discussion des résultats

4.4.3.1 Moyenne d'énergie consommée (MEC)

Sur la figure 4.3, on observe l'évolution de la moyenne d'énergie consommé en fonction du nombre des nœuds, dans toutes les topologies l'énergie consommé pour EDBUC est plus petite que l'énergie consommée dans LEACH, et en remarque que l'augmentation des nœuds ne signifie pas forcément une diminution de la consommation d'énergie cela est du au déploiement aléatoire qui différent de topologie en topologie.

| Nombre de nœuds | | 25 | 50 | 75 | 100 | 125 | 150 | 175 |
|-----------------|--------------|---------|---------|---------|---------|---------|---------|---------|
| LEACH | MEC | 643.153 | 591.027 | 797.111 | 552.845 | 647.104 | 608.92 | 969.27 |
| | ET | 205.153 | 215.097 | 166.218 | 204.687 | 168.79 | 143.36 | 126.626 |
| | Couverture % | 24 | 84 | 69,33 | 89 | 60 | 82 | 0 |
| | nœud mort % | 8 | 10 | 24 | 5 | 5,6 | 2,66 | 95,43 |
| | Nb cluster | 1 | 4 | 4 | 3 | 2 | 4 | 0 |
| EBUC | MEC | 366.668 | 388.551 | 367.8 | 298.303 | 347.122 | 342.184 | 697.268 |
| | ET | 226.543 | 149.423 | 112.198 | 75.1432 | 101.278 | 104.494 | 288.309 |
| | Couverture % | 64 | 72 | 93,33 | 94 | 99,2 | 90 | 33,91 |
| | nœud mort % | 0 | 4 | 0 | 0 | 0 | 0 | 34,28 |
| | Nb cluster | 5 | 3 | 10 | 7 | 11 | 8 | 7 |

Tableau4.1 Tableau qui résume une comparaison entre LEACH et EBUC.

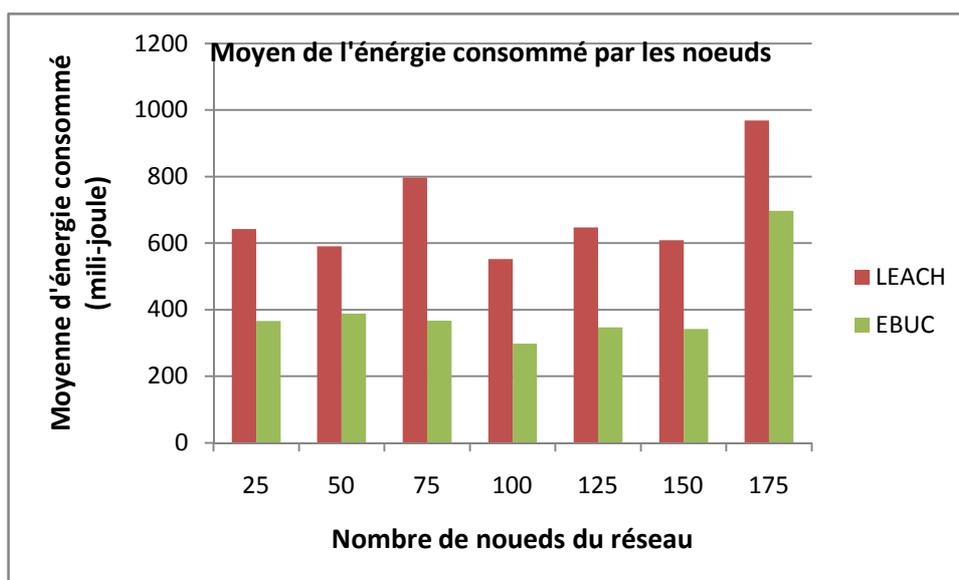


Figure4.3 Moyenne d'énergie consommée par les nœuds.

4.4.3.2 Ecart type de l'énergie consommée (ET)

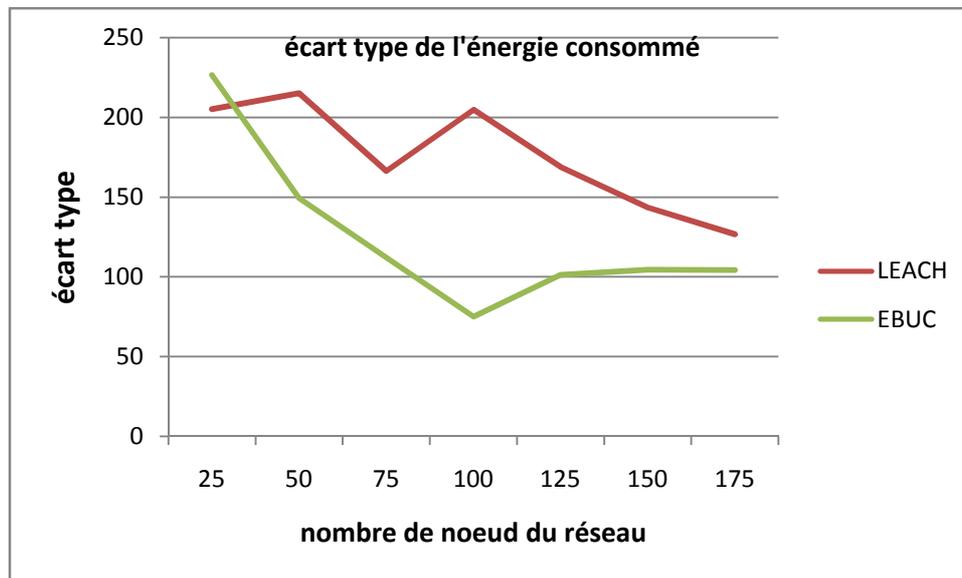


Figure 4.4 L'écart type de l'énergie consommée.

L'écart type de l'énergie consommée diminue en fonction de l'augmentation du nombre de nœuds, ce qui signifie que la consommation d'énergie est de plus en plus équilibrée en fonction de la densité des nœuds, cela s'explique par la répartition des tâches sur les nœuds, plus on a de nœuds plus on a de possibilité de répartir la consommation d'énergie.

4.4.3.3 Pourcentage des nœuds épuisant leur énergie

Afin d'étudier la durée de vie du réseau en fonction du nombre de nœuds déployés, nous avons suivi la diminution des nœuds d'une topologie de 25 nœuds dans l'intervalle (100 1000 secondes).

Il est clair, sur ces figures ci-dessous, que la durée de vie du réseau offerte par le protocole EDBUC est plus importante que celle de protocole LEACH, Ce gain revient aussi à l'équilibrage dans la distribution de la charge et la prise en considération de la densité dans EDBUC.

On peut également constater que la durée de vie du réseau augmente avec la densification du réseau ce qui est attendu du moment que l'énergie moyenne consommée diminue en augmentant le nombre de nœuds de réseau.

Ces résultats s’expliquent par la réduction des distances de transmission entre les nœuds voisins avec l’augmentation du nombre de nœuds déployés dans le réseau.

| Temps (s) | 100 | 200 | 300 | 400 | 500 | 600 | 700 | 800 | 900 | 1000 |
|-----------|------|-----|-----|-----|-----|-----|-----|-----|-----|------|
| LEACH | 100% | 82% | 68% | 48% | 20% | 8% | 4% | 4% | 0% | 0% |
| EBUC | 100% | 88% | 84% | 72% | 52% | 44% | 40% | 36% | 16% | 8% |

Tableau4. 2 Pourcentage de nœuds en vie pour une topologie de 25 nœuds.

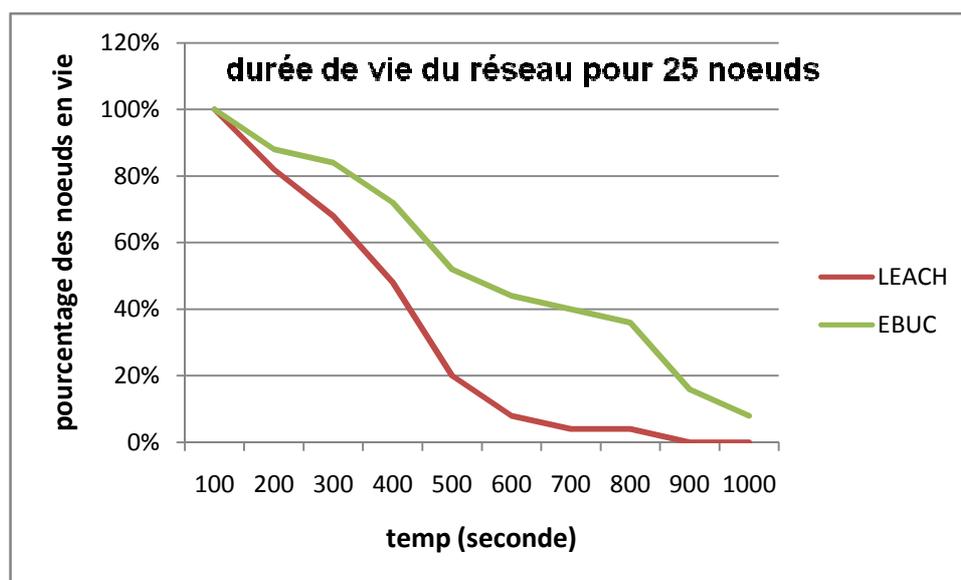


Figure4.5 La durée de vie du réseau pour 25 nœuds.

4.5 Conclusion

Dans ce dernier chapitre, nous avons présenté l’environnement de simulation utilisé OMNET++ et son principe de fonctionnement, comme nous avons implémenté l’algorithme de clustering proposé EDBUC (Energy and Density Blancing Unequal Clusters). Les résultats des simulations effectuées sous OMNeT++ ont montré que ce dernier améliore significativement la consommation d’énergie par rapport à son ancêtre LEACH, donc il assure une durée de vie assez prolongée, cela est du essentiellement à la robustesse des critères

utilisés pour le choix des clusters-heads. En effet, l'algorithme a permis une meilleure distribution des cluster-heads, avec un nombre réduit de clusters et dont la taille change suivant le rayon par rapport à la station de base.

Conclusion générale

Les WSNs constituent un axe de recherche très fertile, ils peuvent être utilisés dans divers domaines, tels que la surveillance des champs de bataille, diagnostique des panne de machines, les espaces intelligent, le suivi des stocks...etc Cependant il reste toujours de nombreuses problématiques liées à ce domaine.

Parmi les problèmes visés par plusieurs recherches, ceux relatives aux routages, en effet quand les protocoles de routages sont implémentés sur une structure plate, leurs performances diminuent notamment ils consomment une grande quantité d'énergie due au trafic généré et à l'absence de l'agrégation de données.

Dans ce projet nous avons mis en place un nouvel algorithme de clustering EDBUC, qui permet d'organiser les nœuds du réseau en cluster, dans le but de réduire la consommation d'énergie. Au cours de ce travail, les études que nous avons effectuées dans le 1^{er} et le 2eme chapitre, nous ont permis de découvrir un champ de recherche très intéressant qui touche pratiquement tous les domaines d'application. Cela nous a permis aussi de nous initier à la recherche et nous visons à contribuer dans ce domaine qui est en pleine évolution.

La principale perspective est de continuer sur ce travail en lui développons un routage adéquat à l'auto organisation faite, implémenter notre approche sur des capteurs réels.

Référence

1. **[ABA 12]** ABED & BACHA « Description des comportements d'un réseau de capteurs sans fils à l'aide de SMA. » Ecole nationale Supérieur de l'Informatique (E.S.I) 2012
2. **[AKY 02]** I.F. AKYILDIZ, Weilian SU, Yogesh Sankarasubramaniam, and Erdal CAYIRCI, A survey on sensor networks, *IEEE Communications Magazine*, pp.102-114, August 2002
3. **[AME 10]** Ian F. AKYILDIZ Mehmet Can VURAN, « Wireless Sensor Networks » John Wiley & Sons Ltd, 2010.
4. **[BEC 09]** Walid BECHKIT « Un nouveau protocole de routage avec conservation d'énergie dans les réseaux de capteurs sans fil » Ecole nationale Supérieur de l'Informatique (E.S.I) 2009
5. **[BEY 09]** Kamal BEYDOUN « conception d'un protocole de routage hiérarchique pour les réseaux de capteurs sans fils »Thèse doctorat, Université Franche-Comté France 2009
6. **[BOU12]** Khaled BOUCHAKOUR « Routage hiérarchique sur les réseaux de capteurs sans fil: Protocole KhLCH (K-hop Layered Clustering Hierarchy). » ESI, 2012
7. **[CHA 08]** Yacine CHALLAL « Réseaux de capteurs sans fils » 2008
8. **[DEN 06]** J. DENG, R. HAN, and S. MISHRA, *Insens: Intrusion-Tolerant Routing for Wireless Sensor Networks.*, *Computer Communications* 29, no. 2, pp.216–230, 2006.
9. **[ESS 07]** M. ESSLOH, C. RICHARD, H. SNOUSSI, *Localisation distribuée dans les réseaux de capteurs sans fils par résolution d'un problème quadratique*, Colloque GRETSI, Troyes, 11-14 septembre 2007.
10. **[GAY 03]** V. GAYRAUD, L. NUAYMI, F. DUPONT, S. GOMBAULT, and B. THARON, *La Sécurité dans les Réseaux Sans Fil ad-hoc*, SSTIC03, 2003.
11. **[GUP 02]** A. GUPTA, N.KUMAR, *Term Paper Report on Acquiring information from smart badges in an emergency network*“, *Advanced Computer Networks*, 2002.

12. **[HAM 07]** A. HAMZI, *Plateforme pour l'aide à la conception et à la simulation des réseaux sans fil, mémoire pour obtenir le grade de magister en informatique, Institut National de formation en Informatique, 2007.*

13. **[KAC 09]** Rahim KACIMI « *Techniques de conservation d'énergie pour les réseaux de capteurs sans » doctorat de l'Université de TOULOUSE France, 2009*

14. **[KHE 08]** Manel KHELIFI « *Optimisation de la consommation de l'énergie et maximisation de la durée de vie des réseaux capteurs sans fil, Université Abderrahmane Mira de Bejaïa, 2008*

15. **[KHE 05]** I. KHEMAPECH, I.DUNCAN and A. MILLER *A Survey of Wireless Sensor Networks Technology*, in *PGNET, Proceedings of the 6th Annual PostGraduate Symposium on the Convergence of Telecommunications, Networking & Broadcasting* (M. Merabti and R. Pereira, eds.), (Liverpool, UK), EPSRC, June 2005

16. **[KHE 04]** I. KHELLADI, N. BADACHE, *Les réseaux de capteurs: état de l'art, rapport de recherche, laboratoire LSI USTHB, février 2004.*

17. **[KRI 05]** B. KRISHNAMACHARI, *Wireless sensor networks, Cambridge University press, 2005.*

18. **[LEACH2000]** Wendi Rabiner Heinzelman, Anantha Chandrakasan, and Hari Balakrishnan « *Energy-Efficient Communication Protocol for Wireless Microsensor Networks* », 2000 ;

19. **[LI 05]** N. Li, J. C. Hou, and L. Sha "Design and analysis of an mst-based topology control algorithm" *IEEE Transactions on Wireless Communications*, 4(3) :1195–1206, 2005.

20. **[LU 07]** J. L. Lu, F. Valois, and D. Barthel "Low-energy self-organization scheme for wireless ad hoc sensor networks" In *Proceedings of the 4th Annual Conference on Wireless on Demand Network Systems and Services, WONS07, pages 138–145, Obergurgl, Tyrol, Austria 2007.*

21. **[MAT 07]** M. VAN DER HAEGEN, *Réseaux de senseurs sans fil: problèmes de localisation, Mémoire proposé en vue de l'obtention du grade de licencié en informatique, Université Libre de Bruxelles, 2007.*

22. **[MEH 01]** S. MEHROTRA, *Distributed Algorithms for Tasking Large Sensor Networks, Thesis submitted to the Faculty of the Virginia Polytechnic Institute and State University, July 2001.*

23. **[PAR 05]** P. PARREND, *Localisation dans les réseaux de capteurs sans fils, INSA Lyon, Janvier 2005.*

24. **[PEN 02]** W. Peng-Jun, K. M. Alzoubi, and O. Frieder "Distributed construction of connected dominating set in wireless ad hoc networks" *In Proceedings of the 21st Annual Joint Conference of the IEEE Computer and Communications Societies, INFOCOM02, Pages 1597–1604, New York, USA, 2002.*

25. **[POM 04]** C. POMALAZA-REZ, *Wireless Ad Hoc & Sensor Networks, University of Oulu, Finland, 2004.*

26. **[POT 00]** G.J.POTTIE, W.J.KAISER, *Wireless integrated network sensors, Communications of the ACM 43, pp.551 –558, May 2000.*

27. **[VIS 04]** S.C. VISWESWARA, *An Automatic, Adaptive, ad-hoc Algorithm for Power Conservation in Sensor Networks using Switch-off, Master's Thesis, department of computer science, faculty of North Carolina State University, July 2004.*

28. **[YIN 06]** J. YIN and S. MADRIA, *Secrout: A secure routing protocol for sensor networks, AINA 06: Proceedings of the 20th International Conference on Advanced Information Networking and Applications (Washington, DC, USA),vol. 1, IEEE Computer Society, pp. 393–398, 2006.*

Webliographie

[Lien1] : <http://mobed.yonsei.ac.kr/retos/index.html>

[Lien2] : <http://mantisos.org/index/tiki-index.php.html>

[Lien 3] : <http://bonsplanswifi.voila.net/defwifi.html>

[Lien 4] : <http://www.tomshardware.fr/articles/bluetooth-2.1,1-17232.html>

[Lien5] : <http://www.generation-nt.com/bluetooth-sig-bluetooth-2-1-edr-approbation-actualite-43750.html>

[Lien 6] : <http://french.bluetooth.com/Bluetooth/Technology/>