

Résumé

Les progrès récents dans la microélectronique, la communication sans fil et les technologies de capteurs à faible coût ont permis l'émergence et l'évolution des réseaux de capteurs sans fil (WSN) comme un nouveau paradigme du réseau informatique. Les réseaux de capteurs sans fil (WSN) ont été considérés comme des outils prometteurs pour de nombreuses applications dépendantes de la localisation telles que la surveillance de la zone, la recherche et le sauvetage, le suivi et la navigation mobiles, etc. Ces applications s'attendent souvent à une connaissance de l'emplacement exact des nœuds. De plus, l'information géographique des nœuds capteurs peut être essentielle pour améliorer la gestion du réseau, la topologie, la planification, le routage des paquets et la sécurité. Bien que la localisation joue un rôle important dans tous ces systèmes, elle-même est un problème difficile en raison de ressources extrêmement limitées disponibles à chaque nœud capteur à faible coût avec une efficacité énergétique élevée.

Dans cette thèse nous avons proposé des solutions permettant de localiser un nœud en réduisant son coût, minimisant la consommation d'énergie et prolongeant la durée de vie du réseau à savoir : Le couplage du routage géographique et les services de localisation, une méthode hybride qui combine le protocole de routage Greedy Perimeter Stateless Routing et les services de localisation Grid Location Service et Hierarchical Location Service afin de réduire les coûts de localisation et d'augmenter non seulement les performances de localisation, mais aussi les performances du routage dans les réseaux de capteurs sans fils (WSNs).

Afin de prolonger la durée de vie du réseau WSN, nous avons optimisé le protocole de routage hiérarchique Low Energy Adaptive Clustering Hierarchy Centralized en définissant une nouvelle approche de sélection des cluster-heads et une méthode qui permet à chaque cluster d'avoir un nombre égal de nœuds qui améliore la durée de vie du réseau, la région stable et le débit du réseau de capteurs. Le schéma proposé équilibre la répartition de l'énergie dans le réseau.

Mots clés : Réseaux de Capteur sans fils(WSNs), protocole de routage, services de localisation, routage hiérarchique.

Abstract

Recent advancements in the world of microelectronics and wireless technologies have led to the development and evolution of wireless sensor networks (WSN) as a new computing paradigm. Wireless sensor networks (WSN) have been considered as a promising solution for many locations dependent applications such as area surveillance, search, tracking and navigation, etc. These Applications often expect knowledge of the exact location of nodes. This information is used to improve network management and topology or to identify the specific positions of certain nodes to enhance packet routing and security. Although localization plays an important role in all these systems, itself is a difficult problem due to extremely limited resources available at each low-cost sensor node with high energy efficiency.

In this thesis, we proposed some solutions to locate a node by reducing its cost, minimizing energy consumption and extending the lifetime of the network. Firstly we propose to couple the geographic routing and location services as a hybrid method that combines the Greedy Perimeter Stateless Routing protocol and the Grid Location Service and Hierarchical Location to reduce location costs and increase not only location performance but also routing performance in sensor networks (WSNs).

In order to extend the life of the WSN network, we have optimized the Low Energy Adaptive Clustering Hierarchy Centralized hierarchical routing protocol by defining a new cluster-heads selection approach and a method that allows each cluster to have an equal number of nodes which improves the lifetime of the network, the stable region and the network throughput. The proposed scheme balances the distribution of energy in the network.

Keywords: Wireless Sensor Networks (WSNs), routing protocol, location services, hierarchical routing.

AVANT PROPOS

Nom et Prénom de l'auteur :

Rania KHADIM

Intitulé du travail :

Contributions à l'amélioration des services de localisation dans les réseaux ad hoc de capteurs

Nom et Prénom du (ou des) directeur(s) de recherche :

Pr. Abdelhakim MAADEN

Pr. Mohammed ERRITALI

Laboratoire(s) où les travaux ont été réalisés :

Mathématiques et Applications

Date de commencement de ce travail :

2014/2015

Rapporteur(s) autre(s) que l'encadrant(s) :

Pr. Mohamed BAHAJ (Faculté des Sciences et Techniques, Settat)

Pr. Mohamed FAKIR (Faculté des Sciences et Techniques, Béni-Mellal)

Pr. Younes IDRISSEI KHAMLICHI (Ecole Nationale des Sciences Appliquées, Fès)

Principales publications et communications auxquelles ce travail a donné lieu :

Publications:

- Rania Khadim et all. « *Performance study of IEEE 802.15.4 under OPNET modeler for wireless sensor networks* » .TELKOMNIKA Indonesian Journal of Electrical Engineering. Vol. 16, 1 (October 2015), pp 98-105.
- Rania Khadim et all. « *Recognition of 3D Objects from 2D Views Features* » .IGI-GLOBAL: Journal of Electronic Commerce in Organizations (JECO), vol. 13, issue 2, April 2015, pp 50-58.
- Rania Khadim et all. « *Rang free localization schemes for wireless sensor networks* ». TELKOMNIKA Indonesian Journal of Electrical Engineering. Vol 16, No.2, November (2015), pp 323-332.
- Rania Khadim et all. « *Performance Evaluation of Hierarchical Location-Based Services Coupled to the Greedy Perimeter Stateless Routing Protocol for Wireless Sensor*

Networks ». International Journal of Electronics and Communications Engineering, Vol 10, No 8(2016).

- Rania Khadim et all. « *Hierarchical Location-Based Services for Wireless Sensor Networks* ». In Proceedings of the IEEE 13th International Conference on Computer Graphics, Imaging and Visualization (CGIV), pp. 457-463. Beni-Mellal, Morocco, March 2016.
- Rania Khadim et all. « *Impact of Location Data Freshness on Routing in Wireless Sensor Networks* ». Europe and MENA Cooperation Advances in Information and Communication Technologies Volume 520 of the series Advances in Intelligent Systems and Computing pp 373-382.

Communications :

- Rania Khadim et all. « Recognition of 3D objects from 2D views features ». The second International Conference on Business Intelligence (CBI'15). FST-Béni-Mellal, Maroc. 23-25 Avril 2015.
- Rania Khadim et all. « Performance Study of IEEE 802.15.4 Under OPNET Modeler for Wireless Sensor ». The second International Conference on Business Intelligence (CBI'15). FST-Béni-Mellal, Maroc. 23-25 Avril 2015.
- Rania Khadim et all. « Rang-Free Localization Schemes for Wireless Sensor Networks ». Conference of Advanced Information Technology Services and Systems (AIT2S-15).
- Rania Khadim et all. « Hierarchical Location-Based Services for Wireless Sensor Networks ». 13th International Conference on Computer Graphics, Imaging and Visualization (CGIV 2016).
- Rania Khadim et all. « Impact of Location Data Freshness on Routing in Wireless Sensor Networks ». Europe Middle East and North Africa Conference on Technology and Security to Support Learning (EMENA-TSSL 2016).
- Rania Khadim et all. « Coupling of geographic location-based service and routing for wireless sensor networks ». 17th International Arab Conference on Information Technology (ACIT' 2016), Sultan Moulay Slimane University, Beni Mellal, Morocco.

Remerciements

*Soyons reconnaissants aux personnes qui nous donnent
du bonheur ; elles sont les charmants jardiniers
par qui nos âmes sont fleuries.*

Marcel Proust

Le temps met tout en lumière.

Thalès

En tout premier lieu, je remercie le bon Dieu, tout puissant, de m'avoir donné la force pour survivre, ainsi que l'audace pour dépasser toutes les difficultés.

Je tiens à exprimer tous mes sincères remerciements à **M. Abdelhakim MAADEN** professeur à la Faculté des Sciences et Techniques de Beni-Mellal, Université Sultan Moulay Slimane, pour avoir dirigé cette thèse. Ses encouragements dans mes moments de doute et la confiance qu'il m'a accordée m'ont permis de mener à bien ce travail. Je tiens encore à lui exprimer ma profonde gratitude pour son aide et tous ses précieux conseils.

Un merci particulier à **M. Mohammed ERRITALI** professeur à la Faculté des Sciences et Techniques de Beni-Mellal pour sa façon efficace et amicale avec laquelle il a suivi ce travail. Outre son questionnement constructif, la cohérence de ses suggestions, et la perspicacité de ses analyses, je souhaite le remercier pour son soutien et ses encouragements permanents, ainsi que pour le climat de confiance et de convivialité dans lequel il m'a permis de travailler.

Ensuite, je tiens à remercier **M. MOHAMED BAHAJ** professeur à la Faculté des Sciences et Techniques de Settat, **M. Youness IDRISSE KHAMLI** professeur à l'Ecole Nationale des Sciences Appliquées Fès, et **M. Mohamed FAKIR** professeur à la Faculté des Sciences et Techniques de Béni Mellal, pour avoir consacré du temps dans la lecture de cette thèse ainsi que pour avoir soumis leurs précieux jugements sur la qualité et le contenu de ce travail.

Mes remerciements s'adressent également à **M. Said MELLIANI** professeur à la Faculté des Sciences et Techniques de Béni Mellal, **Mme. IDRISSE NAJLA** professeur à la Faculté des Sciences et Techniques de Béni Mellal et **M. RACHID ELAYACHI** professeur à la Faculté des Sciences et Techniques de Béni Mellal, membres du jury, qui ont bien accepté de siéger au jury de cette thèse. Qu'ils trouvent ici l'expression de ma reconnaissance !

Dédicace

À la lune et le soleil de ma vie mes CHERS PARENTS

Aucune dédicace ne saurait exprimer mon respect, mon amour éternel et ma considération pour les sacrifices que vous avez consenti pour mon instruction et mon bien être.

Je vous remercie pour tout le soutien et l'amour que vous me portez depuis mon enfance et j'espère que votre bénédiction m'accompagne toujours.

Que ce modeste travail soit l'exaucement de vos vœux tant formulés, le fruit de vos innombrables sacrifices, bien que je ne vous en acquitterai jamais assez.

Puisse Dieu, le Très Haut, vous accorder santé, bonheur et longue vie et faire en sorte que jamais je ne vous déçoive.

À mon Mari

Aucun mot ne saurait t'exprimer mon profond attachement et ma reconnaissance pour l'amour, la tendresse et la gentillesse dont tu m'as toujours entouré.

Cher mari j'aimerais bien que tu trouves dans ce travail l'expression de mes sentiments de reconnaissance les plus sincères car grâce à ton aide et à ta patience avec moi que ce travail a pu voir le jour. . .

Que dieu le tout puissant nous accorde un avenir meilleur.

À mon cher adorable Frère

En témoignage de mon affection fraternelle, de ma profonde tendresse et reconnaissance, je te souhaite une vie pleine de bonheur et de succès et que Dieu, le tout puissant, te protège et te garde.

À ma chère petite nièce Alaa

Aucune dédicace ne saurait exprimer tout l'amour que j'ai pour toi, Ta joie et Ta gaieté me comblent de bonheur. Puisse Dieu te garder, éclairer ta route et t'aider à réaliser à ton tour tes vœux les plus chers.

À la mémoire de ma grand-mère chérie

*J'aurais tant aimé que tu sois présente
Que Dieu ait ton âme dans sa sainte miséricorde*

*A toute personne que je connais et qui me sont chers et tous
ceux qui m'aiment*

Rania

Table des matières

Résumé	1
Abstract.....	2
Remerciements	5
Dédicace	6
Liste des Acronymes	13
Liste des Figures.....	17
Liste des Tableaux	19
Chapitre 1	20
Contexte Général	20
I. Introduction	20
II. Motivation	20
III. Objectifs de la thèse	24
IV. Organisation de la thèse	27
Chapitre 2	29
Réseaux de Capteurs sans fils	29
I. Introduction	29
II. Définition	30
2.1. Capteur	30
2.2. Capteur intelligent	30
III. Architecture matérielle d'un capteur	30
3.1. Classification des capteurs.	32
3.2. Caractéristiques des capteurs.....	33
IV. Architecture d'un réseau de capteurs sans fil	33
V. Classification des RCSFs	34
5.1. Selon le mode d'acquisition et de livraison des données au puits	34
5.2. Selon la distance entre les nœuds capteurs et le puits	35
5.3. Selon le modèle de mobilité dans le réseau.....	35
5.4. Selon les capacités des nœuds du réseau.....	36
VI. Caractéristiques des réseaux de capteurs sans fil	36
6.1. Type de service.....	36
6.2. Qualité de service (QoS)	36
6.3. Tolérance aux fautes.....	37
6.4. Durée de vie	37
6.5. Scalabilité	37
6.6. Maintenance	37

6.7.	Programmation flexible.....	38
VII.	Comparaison entre les RCSFs et les réseaux Ad-hoc classiques	38
VIII.	Facteurs et contraintes de conception d'un RCSF.....	40
8.1.	Contraintes conceptuelles.....	40
8.2.	Contraintes matérielles	42
IX.	Domaines d'application des RCSF.....	42
X.	Communication dans les RCSFs	44
XI.	Modélisation pour les réseaux de capteurs.....	50
11.1.	Simulateurs de réseaux généraux.....	50
11.2.	Simulateurs dédiés aux réseaux de capteurs	53
11.3.	Simulateurs prenant en compte un modèle d'environnement.....	55
XII.	Protocoles de routage	56
12.1.	Protocoles centralisés.....	56
12.2.	Routage géographique	59
XIII.	Conclusion.....	63
Chapitre 3		64
Description et Etude de la performance de la norme IEEE 802.15.4		64
I.	Introduction	64
II.	Dispositifs du réseau	65
III.	Topologie de réseau	65
3.1.	Topologie en étoile (star)	66
3.2.	Topologie point à point (peer to peer).....	66
3.3.	Topologie cluster –tree (arborescence)	66
IV.	Principales couches du standard IEEE 802.15.4	67
4.1.	Couche physique	67
4.2.	Couche d'accès au médium (MAC).....	68
V.	Modèle de simulation d'IEEE 802.15.4	71
VI.	Etapas d'installation sous OPNET Modeler.....	72
VII.	Description du système	74
VIII.	Résultat.....	74
8.1.	Charge de sortie du réseau (Network Output Load).....	74
8.1.1.	Coordinateur du PAN (PAN_Coordinator).....	74
8.1.2.	GTS End Device	76
8.1.3.	CAP End Device (Non GTS End Device)	77
8.2.	Energie consommée	77
8.3.	Puissance reçue par le récepteur radio	78
IX.	Conclusion.....	80

Chapitre 4.....	81
Localisation dans les réseaux de capteur sans fil.....	81
I. Introduction.....	81
II. Systèmes de localisation.....	81
2.1. Les principaux systèmes de localisation.....	81
2.2. Principe de localisation GPS.....	82
III. Localisation dans les Réseaux de Capteurs Sans fil.....	84
3.1. Estimation des distances.....	84
3.2. Dérivation des positions.....	84
IV. Caractérisations des méthodes.....	84
4.1. Utilisation d'estimations de distances.....	85
4.1.1. Méthodes range-free (Les méthodes libres de mesure).....	85
4.1.2. Méthodes range-based (Les méthodes basées mesures).....	85
4.2. Nécessité de connaître la position d'ancres.....	85
4.2.1. Méthodes anchor-based.....	85
4.2.2. Méthodes anchor-free.....	85
4.3. Forme d'implémentation.....	86
4.3.1. Méthodes centralisées.....	86
4.3.2. Méthodes distribuées.....	86
V. Technologies de signaux utilisés dans la localisation.....	86
5.1. Infrarouges.....	86
5.2. (Ultra) sons.....	87
5.3. Radio frequencies.....	87
5.3.1. GPS : Global positioning system.....	87
5.3.2. RFID : Radio Frequency Identification.....	87
5.3.3. RF-UWB : Ultra Wide Band.....	88
5.3.4. RF-WIFI, RF-Bluetooth.....	88
5.4. Image.....	88
VI. Technologies de mesure.....	89
6.1. Différence des temps d'arrivée.....	89
6.2. Temps d'arrivée.....	90
6.3. Puissance du signal.....	90
6.4. Angle d'arrivée.....	90
VII. Algorithmes de localisation.....	91
7.1. Algorithmes basés sur les méthodes Range-based.....	91
7.1.1. SumDistMinMax.....	91
7.1.2. DV-Distance.....	92

7.1.3.	DV-Euclidean	92
7.2.	Algorithmes basés sur les méthodes Range-free	93
7.2.1.	Centroid	93
7.2.2.	Weighted Centroid Localization (WCL).....	96
7.2.3.	DV-Hop	97
7.2.4.	Amorphous.....	100
VIII.	Simulation et Analyse	102
8.1.	Topologie du réseau	102
8.2.	Plate-forme de simulation et distribution de nœuds.....	102
8.3.	Définition de l'erreur de localisation.....	102
8.4.	Simulation des algorithmes	103
8.5.	Temps de calcul d'algorithme amélioré est comparé à d'autres algorithmes.....	105
8.6.	Consommation d'énergie des algorithmes.....	107
IX.	Conclusion.....	109
Chapitre 5	110
Routeage et Localisation Conjointes dans les Réseaux WSNs: Nouvelle Approche.....		110
I.	Introduction	110
II.	Protocoles de routage géographique.....	111
III.	Services de Localisation.....	112
IV.	Combinaison entre le protocole GPSR et les services de localisation	117
4.2.	Algorithme	118
V.	Environnement de travail pour les expérimentations	121
VI.	Résultats expérimentaux	121
6.1.	Résultats de performance de localisation	124
6.1.1.	Taux de Requêtes Réussites ou Répondues (TRR).....	124
6.1.2.	Taux de Requêtes Réussites (TRR) par rapport au nombre de nœuds.....	124
6.1.3.	Temps de déplacement de requête : Request Travel Time(RTT)	125
6.2.	Résultats des Coûts des Services de Localisation	127
6.3.	Impact de la fraîcheur des informations de localisation.....	127
6.3.1.	Impact sur le coût de localisation.....	127
6.3.2.	Impact sur les performances de routage.....	128
6.4.	Résultats des Statistiques de localisation	129
6.5.	Résultats du coût de localisation	130
6.6.	Résultats des Performances du Routage.....	131
VII.	Conclusion.....	133
Chapitre 6	134
Approche optimisée pour prolonger la durée de vie du réseau WSN.....		134

I.	Introduction	134
II.	Protocoles de routage hiérarchique	136
2.1.	Protocole LEACH	137
2.2.	Avantages et inconvénients du protocole LEACH.....	138
2.3.	Descendants du protocole LEACH	139
2.4.	Comparaison entre le protocole LEACH et ses descendants pour les réseaux de capteurs sans fil.....	148
III.	Modèle du réseau	150
3.1.	Modèle de consommation d'énergie dans les réseaux de capteurs	150
IV.	Déclaration de problème et solution proposée	151
V.	Résultats de simulation et discussions.....	154
VI.	Conclusion.....	165
	Conclusion générale et Travaux Futures	166
I.	Travaux de thèse.....	166
II.	Travaux futures	168
	Annexe A : NS-2 : Tutoriel d'utilisation	170
I.	Introduction	170
II.	Notions pour l'interpréteur.....	171
2.1.	Tcl	171
2.2.	OTcl.....	172
2.3.	Lien C++ et Tcl.....	172
III.	Installation	173
	Annexe B : Descriptions des procédures de simulation NS-2	175
I.	Génération du Trafic du réseau	175
II.	Exécution sous NS2	177
III.	Tracés d'évaluations.....	179
	Références	181
	Liste des publications	196
	Liste des communications	198

Liste des Acronymes

AMILC :Age Maximal des Informations de Localisation dans le Cache

ASIC :Application Specific Integrated Circuit

BI :Beacon Interval

BPSK : Binary Phase Shift Keying

BS: Base Station

CAP : Contention Access Period

CBR: Constant Bit Rate

CCA : Clear Channel Assessment

CFP: Contention Free Period

CH : Cluster Head

CSMA/CA : Carrier Sense Multiple Access with Collision Avoidance

DREAM : Distance Routing Effect Algorithm for Mobility

DSP: Digital Signal Processors

DSSS: Direct Sequence Spread Spectrum

ED: Energy Detection

FFD : Full Fonction Device

FPGA :Field Programmable Gate Array

GDOP :Geometric Dilution Of Precision

GLS : Grid Location Service

GPS :Global Position system

GPSR: Greedy perimeter stateless routing

GEAR: Geographic Energy Aware Routing

GAF: Geographic Adaptive Fidelity

GTS: Guaranteed time slots)

GTNetS :Georgia Tech Network Simulator

GF: Greedy Forwarding

GG : Gabriel Graph

HLS: Hierarchical Location Service

IEEE: The Institute of Electrical and Electronics Engineers

IFF : Identification Friend or Foe

ISO : International Standardization Organization

LAN :Local Area Network

LAR : Location-Aided Routing

LCMA: Location Cache Maximum Age

LEACH: Low Energy Adaptive Clustering Hierarchy

LP-LR-WPAN: Low-Power Low-Rate Wireless Personal Area Network

LQI : Link Quality Indication

MAC: Media Access Control

MANET: Mobile Ad hoc NETwork

MEO : Medium Earth Orbit

MEMS :Microelectromechanical Systems

NAM : Network ANimator

NAVSTAR : Navigation Satellite Timing and Ranging

NS-2 : Network Simulator 2

OMNET++ : Objective Modular Network Testbed in C++

OPNET : Optimized Network Engineering Tool

OSI : Open Systems Interconnexion

PDA : Personal Digital Assistant

PDR : Packet Delivery Rate

PEGASIS : Power Efficient Gathering in Sensr Information Systems

PF : Perimeter Forwarding

PSSS: Parallel Sequence Spread Spectrum

QLS: Quorum-based location service

QoS :Quality of Service

QPSK : Quadrature Phase-Shift Keying

QSR: Query Success Ratio

RLS :Reactive Location Service

RFD : Reduced Fonction Device

RFID : Radio Frequency Identification

RNG : Relative Neighborhood Graph

RREP: Route REPlY

RREQ : Route REQuest SN

RSSI : Received Signal Strength Indicator

RTT: Request Travel Time

SD : Super-frame Duration

SO : Super-frame Order

SNFG: Forward Stateless Statistical Non-Deterministic Forwarding

SPEED: Stateless Protocol for Real-Time Communication in Sensor Networks

SPIN : Sensor Protocol for Information via Negotiation

TCP: Transport Control Protocol

TDoA :Time Difference of Arrival

TEEN : Threshold sensitive Energy Efficient sensor Network protocol

TPD: Taux de Requêtes Réussites

TRR : Taux de Requêtes Réussites

ToA : Time of Arrival

UART: Universal Asynchronous Receive and Transmit

UDP: User Datagram Protocol

UERE: User Equivalent Range Error

UWB: Ultra Wide Band

WPAN: Wireless Personal Area Network

WSN : Wireless Sensor Network

Liste des Figures

Figure 2.1. Architecture Matérielle d'un capteur	31
Figure 2.2. Architecture d'un RCSF	34
Figure 2.3. Quelques domaines d'application pour les RCSFs	45
Figure 2.4. Pile protocolaire pour les RCSFs	46
Figure 2.5. Présentation de NS2	51
Figure 2.6. y est le voisin de x le plus proche de la destination D	61
Figure 2.7. X est plus proche de D que ses voisins y, w	61
Figure 2.8. Principe des graphes RNG et GG.....	62
Figure 2.9. Perimeter forwarding	63
Figure 3.1. Topologies supportées dans IEEE 802.15.4.....	66
Figure 3.2. Modes d'exécution de la sous couche MAC de la norme IEEE 802.15.4.....	69
Figure 3.3. Structure de la super-trame IEEE 802.15.4.....	70
Figure 3.4. Structure du modèle de simulation d'IEEE 802.15.4 [7]	72
Figure 3.5. Ajout du répertoire dans les répertoires du modèle d'OPNET	73
Figure 3.6. Scénario par défaut.....	73
Figure 3.7. Palette de wpan_ IEEE 802.15.4.....	73
Figure 3.8. Scénario de réseau pour BPSK et QPSK	74
Figure 3.9. Charge de sortie de réseau pour le PAN_Coordinator	76
Figure 3.10. Charge de sortie de réseau pour le GTS End Device	76
Figure 3.11. Charge de sortie de réseau pour le Non GTS End Device	77
Figure 3.12. Energie Consumé par le PAN_Coordinator	78
Figure 3.13. Energie Consumé par le GTS Device	78
Figure 3.14. Energie Consumé par le Non GTS End Device	79
Figure 3.15. Puissance reçue par le PAN Coordinator	79
Figure 3.16. Puissance reçue par le GTS End Device	80
Figure 3.17. Puissance reçue par le Non GTS End Device	80
Figure 4.1. Principe des méthodes de localisation par satellite (en 2D).....	82
Figure 4.2. Positionnement simple sur le globe.....	83
Figure 4.3. Algorithme SumDistMinMax	91
Figure 4.4. Algorithme DV-Distance	92
Figure 4.5. Algorithme DV-Euclidean	93
Figure 4.6. Exemple de localisation avec Centroid	94
Figure 4.7. Exemple de localisation avec Centroid [168]	95
Figure 4.8. Algorithme DV-HOP [23].....	99
Figure 4.9. Topologie isotropique du réseau	102
Figure 4.10. Répartition des nœuds (a) Centroid (b) DV-Hop (c) Amorphous.....	103
Figure 4.11. Erreur de localisation figures: (a) Centroid (b) DV-Hop ET (c) Amorphous	104
Figure 4.12. Proportion des ancrs et l'erreur de localisation	105
Figure 4.13. Portée radio et l'erreur de localisation	106
Figure 4.14. Nombre des nœuds et l'erreur de localisation	106
Figure 4.15. Consommation d'énergie pour de différentes proportions des ancrs	108
Figure 4.16. Consommation d'énergie pour de différentes portées sans fil	108
Figure 4.17. Consommation d'énergie pour de différents nombres des nœuds.....	108
Figure 5.1. Taxonomie des services basés sur la localisation.....	113
Figure 5.2. Exemple de partition hiérarchique du réseau dans GLS et HLS	115

Figure 5.3. Présentation schématique d’algorithme de Combinaison	118
Figure 5.4. Fichier NAM généré du réseau proposé.....	121
Figure 5.5.Résultats de performance de localisation : Taux de Requêtes Réussites	124
Figure 5.6 .Résultats de performance de localisation : Taux de Requêtes Réussites (TRR).....	125
Figure 5.7. Résultats de performance de localisation: Taux de requêtes réussites.....	125
Figure 5.8.Résultats de performance de localisation : Request Travel Time (RTT).....	126
Figure 5.9.Résultats de performance de localisation: Request Travel Time (100 nœuds).....	126
Figure 5.10.Résultats du coût de localisation : Coûts des Services de Localisation	127
Figure 5.11. Impact de la fraîcheur des informations de localisation sur les requêtes	128
Figure 5.12 . Impact de la fraîcheur des informations de localisation sur le coût.....	128
Figure 5.13 .Impact de la fraîcheur des informations de localisation sur le TPD	129
Figure 5.14. Impact de la fraîcheur des informations de localisation sur la latence.....	129
Figure 5.15. Résultats des Statistiques de localisation dans le cache.....	130
Figure 5.16.Résultats des Statistiques des Paquets de localisation.....	131
Figure 5.17. Coût de localisation: Bande passante consommée pour la localisation	132
Figure 5.18. Statistiques de routage.....	133
Figure 6.1.Modèle du routage hiérarchique.....	137
Figure 6.2 .Modèle de consommation d’énergie pour la communication.....	150
Figure 6.3 . Flow chart du protocole LEACH-C amélioré	154
Figure 6.4 . Energie de dissipation de LEACH-C	157
Figure 6.5 . Cycle de la mort du premier nœud comme valeur de n.....	158
Figure 6.6. First Node Dies (FND).....	159
Figure 6.7. Half of the Nodes Die (HND)	159
Figure 6.8. Last Node Dies (LND)	159
Figure 6.9 . Nœuds vivants de différents protocoles pour un réseau de 100 nœuds.....	159
Figure 6.10. Nœuds vivants de différents protocoles pour un réseau de 200 nœuds.....	160
Figure 6.11 . Nœuds vivants de différents protocoles pour un réseau de 300 nœuds.....	160
Figure 6.12 . Nœuds Vivants avec différentes valeurs d’énergie initiale.....	160
Figure 6.13 . Variance d’énergie de différents protocoles.....	161
Figure 6.14 . Paquets envoyés à la BS pour différents protocoles pour un réseau de 100 nœuds.....	161
Figure 6.15 . Paquets envoyés à la BS pour différents protocoles pour un réseau de 200 nœuds.....	161
Figure 6.16 . Paquets envoyés à la BS pour différents protocoles pour un réseau de 300 nœuds.....	162
Figure 6.17 . Consommation d’énergie des différents protocoles pour un réseau de 100 nœuds	162
Figure 6.18. : Consommation d’énergie des différents protocoles pour un réseau de 200 nœuds	163
Figure 6.19 . Consommation d’énergie des différents protocoles pour un réseau de 300 nœuds	163
Figure 6.20 . Nombre de cluster-heads.....	163
Figure 6.21 . Equilibre de la charge (Load balance).....	164
Figure 6.22 . Nœuds morts de différents protocoles pour un réseau de 100 nœuds	164
Figure 6.23 . Nœuds morts de différents protocoles pour un réseau de 200 nœuds	164
Figure 6.24. Nœuds morts de différents protocoles pour un réseau de 300 nœuds	165
Figure 6.25 . Paquets au CH.....	165
Figure 8.1. NAM: Network ANimator tool.....	179

Liste des Tableaux

Tableau 2.1. Caractéristiques de MicaZ et TelosB	33
Tableau 3.1 : Récapitulatif de la couche physique dans IEEE 802.15.4	68
Tableau 3.2. Comparaison selon les modulations BPSK et QPSK	75
Tableau 4.1. Technologies de localisation [167]	89
Tableau 4.2. Temps moyen d'exécution pour de localiser un seul nœud et d'erreurs de localisation	107
Tableau 5.1. Paramètres de simulation NS2	122
Tableau 6.1. Ligne de temps de LEACH et ses descendants	148
Tableau 6.2: LEACH et ses protocoles Descendants	149
Tableau 6.3: Paramètres de simulation	156

Chapitre 1

Contexte Général

I. Introduction

Les progrès récents dans la microélectronique, la communication sans fil et les technologies de capteurs à faible coût ont permis l'émergence et l'évolution des réseaux de capteurs sans fil (WSN) comme un nouveau paradigme de réseau informatique [1]. Un réseau de capteurs sans fil se compose d'un nombre de nœuds capteurs peu coûteux et capables de détecter, traiter des données et de communiquer sans fil à courte portée [2,3]. Les nœuds de capteurs sont déployés dans des domaines d'intérêt pour surveiller de manière coopérative les conditions physiques ou environnementales, telles que le son, les vibrations, la température, la pression, le mouvement, les perturbations électromagnétiques, etc. Les réseaux de capteurs sans fil ont montré de plus en plus de popularité pour les deux tâches militaires [4, 5 6, 7, 8, 9] et les applications civiles, y compris la surveillance et le contrôle des procédés industriels [10, 11, 12, 13, 14], structure de la surveillance de la santé [15, 16, 17, 18, 19], surveillance de l'habitat et de l'environnement [20, 21, 22, 23, 24, 25], applications de soins de santé [26, 27, 28, 29, 30, 31], domotique [32, 33, 34, 35, 36], réseaux de véhicules et les systèmes intelligents de transport [37, 38, 39, 40, 41].

II. Motivation

Un réseau de capteurs sans fil (WSN) est composé d'un grand nombre de nœuds de capteurs, qui ont la capacité de détecter, calculer et de communiquer sans fil et peuvent surveiller et acquérir des informations physiques dans la zone de détection et de distribution en temps réel.

Dans de nombreuses applications susmentionnées, les informations de localisation de chaque nœud capteur du réseau sont essentielles pour le service. Cela revient du fait que les

utilisateurs doivent normalement savoir non seulement ce qui se passe, mais aussi où les événements intéressés se produisent et l'origine de la cible. Par exemple, dans la surveillance des champs de bataille [6, 7, 8], la connaissance du lieu d'origine de l'ennemi peut être beaucoup plus importante que de ne connaître que l'apparence de l'ennemi; dans une opération de secours en cas de catastrophe en utilisant WSN pour localiser les survivants dans un bâtiment effondré, il est essentiel que les capteurs signalent des informations de surveillance ainsi que leur emplacement [23, 48,49, 52, 56]. Un autre exemple, dans la surveillance hospitalière, la connaissance de l'endroit du patient peut aider les médecins à arriver au bon endroit le plus rapidement possible en cas d'urgence [2] [3]. D'autre part, les paramètres de position des nœuds de capteurs sont supposés disponibles dans de nombreuses opérations pour la gestion du réseau, comme le routage où une famille d'algorithmes géographiques a été proposée [42, 43, 44], un contrôle de topologie qui utilise des informations de localisation comme une connaissance a priori pour ajuster la connectivité réseau pour l'économie d'énergie [45, 46, 47] et la maintenance de la sécurité où les informations de localisation peuvent être utilisées pour prévenir les attaques malveillantes [50,51, 52].

La localisation, comme l'un des services de middleware les plus fondamentaux et largement appliqués dans les réseaux de capteurs sans fil, permet généralement à chaque nœud du réseau d'obtenir ses informations de localisation, soit les coordonnées géographiques absolues, soit une position relative qui peut être transformée en contrepartie absolue lorsque cela est nécessaire. La localisation joue un rôle clé dans de nombreuses applications de réseau de capteurs, mais elle est elle-même un problème difficile [53, 54, 55], en raison des exigences exigeantes pour un faible coût, une efficacité énergétique élevée et un faible encombrement sur le côté du noyau du capteur à contrainte de ressources, ainsi que des problèmes pratiques associés aux déploiements réseau. Nous énumérons des difficultés majeures qui remettent en question un positionnement précis et efficace dans les réseaux de capteurs sans fil dans les éléments suivants :

- ✓ **Contraintes de coûts et d'énergie pour chaque nœud de capteur.** :L'exigence d'un design peu coûteux et peu énergétique à chaque nœud capteur interdit la localisation avec un support matériel supplémentaire. Par exemple, le GPS (Global Position System [56]), qui est la technique la plus utilisée dans la localisation, peut difficilement s'appliquer à tous les nœuds capteurs du réseau [57]. De même, les modules supplémentaires, tels que les antennes directionnelles, les boussoles électroniques, les rangiers à laser, les caméras vidéo, etc., sont très limités en raison de leur taille incompatible, de leur coût considérable ou de leur consommation

d'énergie excessive [2, 51, 55, 57]. Cela indique qu'une solution de localisation doit être compatible avec un nœud de capteur, où des caractéristiques d'une empreinte économique, peu coûteuse sont nécessaires.

- ✓ **Scalabilité du réseau** : Un réseau de capteurs sans fil peut être composé d'un grand nombre de nœuds [1, 2, 8, 53]. Par exemple, ExScal [61] et GreenOrbs [24] ont utilisé plus de mille nœuds de capteurs dans leurs réseaux déployés. On prévoit également que les futurs réseaux de capteurs sans fil peuvent inclure des milliers ou même des millions de nœuds [62, 63, 64]. Dans tous ces réseaux, la configuration traditionnelle des paramètres de localisation par nœud [58, 59] pourrait être extrêmement coûteuse, sinon impossible. Par conséquent, une conception de localisation doit être évolutive en réseau, ce qui signifie qu'il devrait être rentable avec les systèmes à petite et grande échelle.
- ✓ **Des environnements de travail difficiles** : Les réseaux de capteurs sans fil sont susceptibles d'être déployés au hasard dans des terrains et des environnements inaccessibles [1, 56, 60], tels que le champ de bataille et la zone de conflit [6, 7], ainsi que les zones habitables [23, 65, 66], etc. ,il n'y a généralement pas d'infrastructure (p. ex., les signaux radio provenant des tours AP sans fil ou des rayonnements de lignes électriques comme références de coordonnées) qui peuvent être utilisés à des fins de localisation. Dans ce cas, la localisation auto-organisée sans interférence humaine et étalonnage est essentielle. En d'autres termes, le mécanisme de localisation est hautement préféré à fonctionner comme un système autonome libre des profils étendus d'environnement.

Les nœuds de capteurs dans un WSN doivent être conscients de leurs positions afin de collaborer efficacement. Les informations de position de chaque nœud obtenues avec les données requises peuvent aider les êtres humains de diverses façons. Ils peuvent permettre une meilleure interprétation des données détectées en identifiant d'où proviennent les données. Cela peut aider à déterminer la qualité de la couverture du réseau et peut aider dans des applications telles que le routage géographique ou la surveillance du mouvement cible.

Cependant, comme le réseau comprend un grand nombre de nœuds, le processus de localisation doit être effectué et distribué par les nœuds eux-mêmes au lieu d'être exécuté sur un système central et devrait commencer automatiquement et immédiatement après le déploiement du réseau. Les défis lors de l'exécution de cette tâche sont que le processus de localisation devrait converger, il devrait converger aussi rapidement que possible, les nœuds devraient utiliser autant

d'énergie que possible et les estimations de position de nœud résultantes devraient être d'une précision souhaitable.

Les données recueillies par un nœud capteur sont habituellement signalées au nœud collecteur (sink) pour les mesures nécessaires. Pour lancer une action rapide, le nœud collecteur doit être au courant des informations de localisation du nœud de rapport. Par exemple, supposons que le feu a eu lieu dans une partie de la forêt et un capteur proche indique ces informations au nœud collecteur. Pour une réponse rapide, le capteur de rapport devrait inclure son emplacement avec d'autres informations. Le marquage du cache d'emplacement avec les informations détectées n'est possible que lorsque le nœud de rapport est localisé. Cela signifie l'importance de localiser un nœud avant son processus de collecte de données. Quelques applications indiquant l'importance de la localisation dans les WSN sont listées ci-dessous:

- ✓ Les capteurs rassemblent des paramètres vitaux liés à la sécurité tels que la communication par radio, les mouvements vigoureux dans une zone de surveillance et les signalent au système de sécurité back-end (un nœud collecteur). Une action rapide du personnel de sécurité n'est possible que si les informations de localisation sont fournies avec les informations détectées [67].
- ✓ Dans certains cas, certains nœuds peuvent mourir en raison du drainage de la batterie ou des forces physiques. Dans de tels cas, les nouveaux nœuds à injecter ou les remplacements de batterie peuvent être réalisés efficacement en adoptant un routage géographique plutôt que des schémas de routage physique [67]. Le routage géographique facilite la tâche de localisation d'un nœud défectueux par rapport au routage physique.
- ✓ Les informations de localisation sont également utilisées pour diviser le WSN en différents clusters afin de faciliter le traitement collaboratif et le routage hiérarchique. Pour chaque cluster, un nœud est choisi comme chef de cluster (Cluster-Head) qui reste responsable de l'inter-connectivité du cluster et de la maintenance de l'état.
- ✓ Les réseaux de capteurs sont comme une base de données distribuée pour les utilisateurs afin d'interroger le monde physique pour des informations utiles. Avec la localisation, l'interrogation spatiale efficace par un nœud collecteur (sink) ou un nœud de passerelle est répondue uniquement par le nœud capteur prévu.
- ✓ Le routage basé sur l'emplacement permet d'économiser de l'énergie en éliminant le besoin de découverte de route et d'améliorer le comportement de mise en cache pour les applications où les requêtes dépendent de l'emplacement.
- ✓ Déterminer la qualité de couverture de tous les capteurs actifs en utilisant leur position.

Une grande majorité des applications dans les réseaux de capteurs utilise un déploiement aléatoire d'un grand nombre de capteurs, en raison soit de l'hostilité de la zone à surveiller, soit de son immensité. La phase de localisation est donc nécessaire non seulement au fonctionnement du réseau (routage géographique par exemple), mais également à l'exploitation des données. Il est donc nécessaire de localiser, avec la meilleure précision possible, tous les nœuds du réseau. Cette problématique, malgré les nombreux travaux de recherche qui s'y étaient attachés, reste une problématique ouverte.

III. Objectifs de la thèse

Un réseau de capteurs sans fil (WSN) est une collection de nœuds de capteurs, fortement déployés dans un environnement physique et organisés en un réseau coopératif. Chaque nœud de capteur a généralement plusieurs parties (capteurs, processeur, émetteur-récepteur, mémoire et batterie). Un nœud capteur est généralement un petit appareil électronique équipé d'une batterie pour une source d'énergie. Il dispose de capteurs physiques pour détecter des conditions environnementales telles que la température, le son, les vibrations, etc. Un émetteur-récepteur sans fil est équipé pour les communications bidirectionnelles avec d'autres capteurs. Dans ces grands systèmes de réseaux de capteurs, nous avons besoin de nœuds pour pouvoir se localiser dans de différents environnements. Les données détectées et recueillies n'auraient aucun sens sans connaître le nœud de localisation dans certaines applications particulières. Ce problème, auquel nous nous référons, est connu sous le nom de localisation des capteurs.

Pour pouvoir localiser les nœuds (déployés aléatoirement) dans un réseau de capteurs, la connaissance des positions des capteurs dans l'environnement surveillé est souvent indispensable pour une grande majorité des applications (militaires, suivis des animaux, ...), afin de pouvoir déterminer l'origine des événements détectés. « Où ? » est la question qui suit immédiatement la détection d'un événement (par exemple, où est le feu ?). En outre, la localisation peut être utilisée dans les protocoles de routage géographique dans les réseaux à grande échelle, en transmettant les données seulement dans la direction de la destination. Comme déjà mentionné, les nœuds dans un réseau de capteurs ne possèdent pas des identifiants globaux. Dans la plupart des cas, les positions des nœuds sont utilisées pour les identifier. De plus, la connaissance des positions des nœuds conditionne de manière directe la qualité de la couverture de la zone d'intérêt.

Dans le cadre de cette étude, on s'est intéressé aux problématiques de localisation dans les réseaux de capteurs sans fil. L'objectif principal consiste à proposer des mécanismes de

localisation adaptés aux caractéristiques des réseaux WSNs et à leurs applications afin de minimiser la consommation d'énergie et prolonger la durée de vie du réseau WSN.

Dans un premier temps on a effectué un état de l'art sur les principaux concepts, spécificités et challenges liés aux réseaux WSNs. Les différents protocoles de routages dans les réseaux de capteurs ont été détaillés. On a pu apporter une vue sur la communication dans les réseaux de capteurs sans fil en décrivant le standard de communication 802.15.4 qu'on a ensuite intégré sous OPNET [68]. Rappelons qu'OPNET est un simulateur qui permet la modélisation et la simulation de réseaux de communication grâce à des 19 bibliothèques de modèles (de routeurs, de commutateurs, de nœuds fixes ou mobiles, serveurs...). Dans cette simulation sous OPNET, on a représenté le réseau WSN comme étant un ensemble de nœuds mobiles et on a examiné les statistiques moyennes de la charge du réseau, la consommation d'énergie et la réception de puissance pour l'ensemble du réseau WSN en utilisant ce standard afin d'évaluer ses performances. Ceci a permis d'avoir un aperçu sur le fonctionnement de la norme IEEE 802.15.4 et de la choisir comme système de communication comme étant le meilleur pour les WSNs.

La deuxième étape de notre étude consiste à présenter les systèmes et les algorithmes de localisation dans les réseaux de capteurs. En effet la localisation des nœuds est l'un des principaux problèmes du réseau de capteurs sans fil (WSN) qui a attiré beaucoup d'attention ces dernières années. L'emplacement des nœuds capteurs n'est pas prédéterminé ou conçu. Cela permet à ces nœuds de capteurs de se déployer au hasard dans des terrains inaccessibles ou des opérations de secours en cas de catastrophe. D'autre part, cela signifie également que les protocoles et les algorithmes du réseau de capteurs doivent posséder des capacités auto-organisées. Dans les WSNs, le problème de localisation peut être intercepté que, dans un réseau de capteurs, l'emplacement de plusieurs nœuds est connu (nœud de balise) et l'emplacement du nœud cible (nœuds inconnus) est obtenu par l'information du capteur et l'algorithme de localisation efficace.

Dans cette partie de notre travail on a choisi pour localiser les nœuds capteurs d'utiliser l'estimation de distance grâce aux méthodes rang-free (libres de mesure). Ces méthodes ne calculent jamais de distances entre voisins. Elles utilisent d'autres informations telles que la connectivité pour identifier la position des nœuds. Dans un objectif de simplicité et de réduction du coût, ces méthodes supposent que le déploiement des nœuds respecte certaines contraintes et propose des calculs plus ou moins complexes pour évaluer la position. Elles semblent donner de bons résultats dans les réseaux denses et réguliers. On a choisi d'étudier trois types d'algorithmes de localisation basés sur les méthodes rang-free (Centroid, Dv-Hop et Amorphous) pour

comparer l'erreur de localisation et la consommation d'énergie de chacun et montrer l'algorithme le plus performant en termes de précision de localisation.

La troisième partie présente notre protocole hybride combinant les services de localisation GLS et HLS avec le protocole de routage géographique GPSR. Les protocoles de routage géographiques ont été conçus pour offrir de meilleures performances pour ces réseaux. Chaque nœud doit prendre en considération sa position géographique réelle et la position du nœud cible à atteindre. Le service basé sur la localisation est nécessaire pour savoir la position de destination. La combinaison de ce service avec le routage est tout à fait naturelle afin d'assurer de meilleures performances. Si un nœud doit envoyer des données à un autre nœud du réseau, GPSR demande au service de localisation s'il possède une information fraîche sur l'emplacement de la destination. Si c'est le cas, GPSR utilise cette position pour transmettre les paquets à la destination. Autrement, le service de localisation lance une demande de localisation pour trouver la nouvelle position de la destination. Lorsque la réponse de localisation est reçue, le service de localisation informe GPSR, qui sera en mesure de transmettre les données vers le nœud destination en utilisant la position reçue. On a utilisé comme critère de performance le coût de localisation, l'impact de la fraîcheur des informations de localisation sur le réseau, la performance de localisation et les performances du réseau.

Notre approche hybride a montré que la combinaison entre le service de localisation et le protocole de routage permet de réduire les coûts de localisation et d'augmenter non seulement les performances de localisation, mais aussi les performances du routage dans les réseaux de capteurs sans fils (WSNs).

Dans les WSNs, la consommation d'énergie est l'une des questions les plus importantes. Les protocoles de routage traditionnels pour WSN peuvent ne pas être optimaux en termes de consommation d'énergie. Les protocoles de routage hiérarchique (HRP) sont plus efficaces en énergie que d'autres protocoles [70]. HRP suit les mécanismes de regroupement; Les techniques de regroupement peuvent être efficaces en termes d'énergie et d'évolutivité [71]. En utilisant une technique de regroupement, ils minimisent considérablement la consommation d'énergie dans la collecte et la diffusion (fusion et agrégation) des données. HRP minimise la consommation d'énergie en divisant les nœuds en différents clusters. Dans chaque cluster, des nœuds d'énergie plus élevés, c'est-à-dire les CH (Cluster-Head) peuvent être utilisés pour traiter et envoyer l'information à la BS (Station de base) alors que les nœuds à faible énergie, c'est-à-dire (les membres du cluster) peuvent être utilisés pour effectuer la détection à proximité de la cible et envoyer à son CH. Cela signifie que la création de clusters et l'attribution de tâches spéciales au

CH peuvent contribuer grandement à l'évolutivité globale, à la durée de vie et à l'efficacité énergétique, réduisant la taille de la table de routage en localisant la configuration de l'itinéraire dans les clusters et en conservant la bande passante de communication du réseau [71]. Le premier protocole hiérarchique est la hiérarchie de regroupement adaptatif à faible énergie (LEACH) qui a été introduite dans [69]. De nombreux protocoles hiérarchiques ont émergé de LEACH.

Notre dernière contribution consiste à proposer un protocole LEACH-C centralisé optimisé en étendant LEACH-C avec deux modifications.

- ✓ La première modification détermine la portée de la sélection du chef du cluster (Cluster-Head) en fonction du niveau d'énergie des nœuds.
- ✓ La seconde est l'attribution des nœuds au cluster-head le plus proche, uniquement si les nœuds déjà attribués à ce cluster-head sont inférieurs à un nombre N.

Dans La première approche, les cluster-heads ne sont pas sélectionnés de façon aléatoire, et la seconde approche permet à chaque cluster d'avoir un nombre égal de nœuds qui améliore la durée de vie du réseau, la région stable et le débit du réseau de capteurs. Le schéma proposé équilibre la répartition de l'énergie dans le réseau.

IV. Organisation de la thèse

Cette thèse traite la localisation dans les réseaux WSNs. Elle est organisée comme suit :

Dans le chapitre 2, on introduit les concepts généraux liés aux réseaux de capteurs sans fil (WSN). Parmi ces concepts on identifie les contraintes de conception des réseaux de capteurs sans fil. Parmi ces contraintes on va voir les contraintes matérielles et conceptuelles. On décrit aussi les architectures et les technologies de communication sans fil pour ce type de réseaux. Enfin on a analysé les caractéristiques, les applications, ainsi que les divers protocoles de routage et les simulateurs utilisés dans ces réseaux.

Le chapitre 3 est consacré principalement à donner un aperçu détaillé sur le standard IEEE 802.15.4 largement utilisé et adapté aux applications des réseaux de capteurs sans fil ,en présentant ses principaux dispositifs fonctionnant selon la topologie adaptée .Ce chapitre permet aussi de décrire les deux principales couches du standard IEEE 802.15.4 en expliquant les deux modes de fonctionnement de cette dernière Un modèle de simulation avec ses différents blocs fonctionnels est aussi présenté. Enfin on présente notre comparaison de la performance de la norme IEEE 802.15.4 sous OPNET MODELER afin de valider le choix de cette norme.

Dans le chapitre 4, on présente les systèmes de localisation dans les réseaux de capteurs sans fil en précisant les caractéristiques des méthodes utilisées pour résoudre ce problème et les technologies de signaux et de mesure utilisées dans la localisation. Enfin on étudie trois types d'algorithmes de localisation pour comparer l'erreur de localisation et la consommation d'énergie de chacun et montrer l'algorithme le plus performant en termes de précision de localisation.

Le chapitre 5 présente les protocoles de routage géographiques et les services de localisation. Ce chapitre décrit l'algorithme hybride utilisé en combinant le protocole de routage géographique GPSR et les services de localisation GLS et HLS afin de construire un système puissant permettant de réduire les coûts de localisation et d'augmenter non seulement les performances de localisation, mais aussi les performances du routage.

Le chapitre 6 permet de montrer les concepts de base du protocole de routage hiérarchique en présentant des détails sur le protocole LEACH et ses différents descendants. On présente le modèle réseau de la consommation d'énergie dans les réseaux de capteurs. Enfin on décrit notre protocole LEACH-C optimisé qui améliore l'efficacité énergétique et prolonge le temps de survie du réseau.

Enfin, toutes les contributions importantes de ce travail de recherche sont présentées dans une synthèse générale. De plus, un ensemble de perspectives sont identifiées et discutées.

Réseaux de Capteurs sans fils

I. Introduction

Les années récentes ont connu d'énormes progrès dans la conception et les applications des réseaux sans fil et les capteurs intégrés. Les nœuds de capteurs sans fil sont généralement des petits appareils peu coûteux et de faible puissance, équipés de capacités limitées de détection, de traitement des données et de communication sans fil, ainsi que des alimentations de puissance. Ils utilisent le concept des réseaux de capteurs sans fil (RCSF, WSN), dans lesquels un nombre important (éventuellement énorme) de nœuds de capteurs collaboratifs pourrait être déployé. À la suite de la convergence de la technologie des systèmes micro-électromécaniques (MEMS), des communications sans fil et de l'électronique numérique, les WSNs représentent une amélioration significative par rapport aux capteurs traditionnels. En fait, l'évolution rapide de la technologie WSN a accéléré le développement et le déploiement de divers nouveaux types de capteurs sans fil, par exemple, des capteurs multimédia. Les capteurs sans fil deviennent plus petits et moins coûteux, et en même temps plus puissants et omniprésents.

Grâce à ces caractéristiques très variées et son faible coût de production, les RCSFs couvrent plusieurs domaines, allant du domaine militaire au domaine médical, en passant par l'industrie, l'écologie, la domotique, l'agriculture de précision, etc [72-75]. Ils permettent à l'utilisateur une meilleure compréhension de l'environnement. De nos jours, les réseaux de capteurs sans fils font partie intégrante de notre vie. Pour cela, ils forment un domaine de recherche très vaste et en pleine croissance. Ainsi, mener un travail de recherche dans ce domaine nécessite la connaissance d'un certain nombre de concepts généraux.

Dans la suite de ce chapitre, on présente ces concepts à travers un certain nombre de points que nous avons adopté. On commence par donner quelques définitions d'un capteur, l'architecture matérielle et voir comment ces capteurs sont déployés pour former des réseaux de capteurs sans fil, qui seront classifiés et comparés aux réseaux ad hoc classiques. Ensuite, les facteurs et contraintes de conception des RCSFs ainsi que les domaines d'applications de ces

réseaux seront étudiés. Avant de conclure ce chapitre, on présente une étude détaillée sur les protocoles de communication destinés aux réseaux de capteurs sans fil.

II. Définition

2.1. Capteur

Un capteur sans fil est un petit dispositif électronique autonome capable de transformer une grandeur physique observée (température, pression, humidité, etc.) en une grandeur utilisable et de la communiquer à un centre de contrôle via une station de base.

Un nœud capteur sans fil est un composant physique, de petite taille, capable d'accomplir trois tâches complémentaires :

- le relevé d'une grandeur physique
- le traitement de l'information
- la communication avec d'autres nœuds capteurs.

2.2. Capteur intelligent

Le terme capteur intelligent (smart sensor ou intelligent sensor) a été utilisé dans l'industrie des capteurs pour désigner des capteurs qui ne fournissent pas seulement des mesures, mais aussi une fonctionnalité aux mesures spécifiques[76]. Par rapport à un capteur classique, un capteur intelligent intègre de nombreux éléments électroniques additionnels ,ainsi que des unités programmables et des aspects logiciels nécessaires au traitement des données, aux calculs, à la communication numérique [77]. Il est donc caractérisé par [76] sa capacité à effectuer une collecte des mesures, les traiter et à les communiquer au monde extérieur.

III. Architecture matérielle d'un capteur

Suivant le type d'application, il existe une multitude de capteurs sur le marché : les capteurs de température, d'humidité, de pression, etc. Cependant, malgré cette diversité apparente, ils restent dotés d'une architecture matérielle similaire. Un nœud de capteur est constitué principalement d'une unité de : captage, traitement, stockage, communication, et énergie illustrée dans la figure 2.1. Il comporte également des composants supplémentaires tels qu'un système de localisation GPS (Global Positioning System) par exemple, un générateur d'énergie (exemple : cellules solaires) ou un mobilisateur. L'unité de traitement est généralement associée à une petite

unité de stockage et elle peut gérer les procédures qui permettent au nœud de capteur de collaborer avec les autres nœuds pour exécuter les tâches de détection affectées. Une unité de communication connecte le nœud au réseau. L'un des composants les plus importants d'un nœud de capteur est l'unité d'énergie. Les unités d'énergie peuvent être supportées par une unité de balayage d'énergie telle que des cellules solaires. Les autres sous-unités du nœud dépendent de l'application [72].

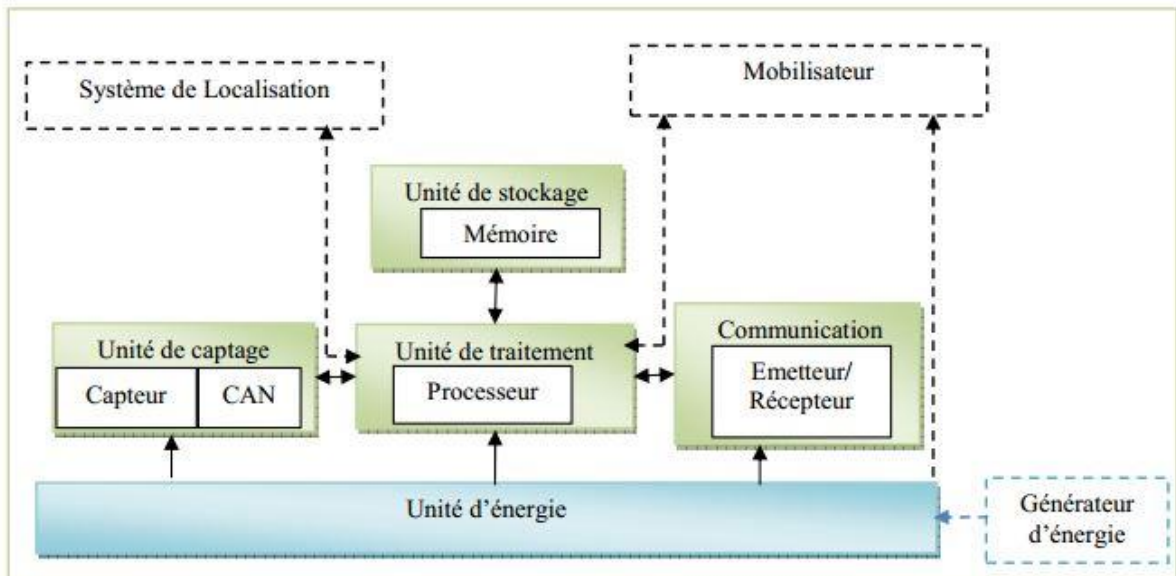


Figure 2.1. Architecture Matérielle d'un capteur

a. Unité d'énergie

Un nœud capteur est constitué d'une unité d'énergie qui est responsable du calcul et de la transmission et délivre l'énergie à toutes ses unités. La consommation de base du nœud est due au calcul et à la transmission qui est l'activité la plus coûteuse. , les nœuds de capteurs fonctionnent sur batterie [78], mais peuvent également récupérer de l'énergie de l'environnement via des cellules solaires.

b. Unité de captage

L'unité de captage est généralement composée de deux sous-unités: des capteurs et des convertisseurs analogique-numérique (CAN). Les signaux analogiques produits par les capteurs sont convertis en signaux numériques par l'CAN, puis introduits dans l'unité de traitement. Un capteur peut avoir une ou plusieurs unités de captage [79].

c. Unité de traitements (processeur)

L'unité de traitement est responsable de l'acquisition des données, du traitement des informations entrantes et sortantes, de la mise en œuvre et du réglage des informations de routage en fonction des conditions de performance du transmetteur. Le nœud capteur est un microcontrôleur composé d'une unité de traitement, mémoire, convertisseurs (Analogique Numérique CAN) et des interfaces de réception et de transmission asynchrones universelles (Universal Asynchronous Receive and Transmit UART) pour effectuer les tâches de traitement [79][80][81]..

d. Unité de communication

Dans l'unité de communication, un émetteur-récepteur commun agit comme une unité de communication et il est principalement utilisé pour transmettre et recevoir l'information entre les nœuds et la station de base et vice versa. Il y a quatre états dans l'unité de communication: transmettre, recevoir, inactif et dormir (transmit, receive, idle and sleep) [79].

e. Unité de stockage (Mémoire)

L'unité de stockage du nœud capteur est utilisée pour stocker à la fois les données et le code du programme. Pour les paquets de données stockés à partir de nœuds voisins (autres), une mémoire aléatoire (RAM) est normalement utilisée et pour mémoriser le code de programme, la mémoire flash ou la mémoire EEPROM (mémoire morte programmable effaçable électriquement) est utilisée [74].

3.1. Classification des capteurs.

Selon la conception du capteur et le mode de fonctionnement, les capteurs sans fil peuvent être catégorisés comme actifs et passifs.

a. Capteurs actifs

Ce type de capteurs a une alimentation électrique intégrée (par exemple une batterie, un supercondensateur ou un récupérateur d'énergie) et peuvent transmettre des signaux jusqu'à plusieurs centaines de mètres. Ce groupe de capteurs sondent activement l'environnement, par exemple un capteur solaire ou radar ou un capteur sismique, qui génère des ondes de choc par petites explosions.

b. Capteurs passifs

Ces capteurs n'ont pas d'alimentation embarquée et reçoivent leur puissance du champ électromagnétique généré par le lecteur du capteur.

Bien que les capteurs sans fil actifs communiquent sur de plus grandes distances que les capteurs passifs, leur plus grande taille et la nécessité de maintenir la source d'alimentation sont des limitations possibles dans certaines applications. Sans la nécessité d'une batterie, la durée de vie d'un capteur passif est limitée principalement par la stabilité du film de détection et du transducteur. Ainsi, alors que la gamme de lecture des capteurs passifs de communication est plus courte que celle des capteurs actifs, les capteurs passifs sont attrayants car ils sont durables, rentables et peu visibles. Les capteurs passifs peuvent être alimentés par couplage inductif ou capacitif avec la puissance délivrée en fonction de la fréquence de fonctionnement, de la taille de l'antenne du capteur, de la taille de la bobine, des conditions d'impédance et de la puissance du lecteur. La communication sans fil à courte portée avec de petits capteurs et des lecteurs de faible puissance (communication de proximité) peut être mise en œuvre lorsqu'il est important de minimiser les effets du fouillis ambiant.

3.2. Caractéristiques des capteurs

Le tableau 2.1 présente les caractéristiques matérielles des capteurs, nous présentons deux types de capteurs : MicaZ et TelosB.

Caractéristiques	TelosB	MicaZ
Flash	48Kb	512Kb
RAM	10Kb	4Kb
MCU	1MHz T1 MPS430	8MHz A Tmega128
Consommation d'énergie	Radio+cpu :75mW Sleep mode : 140μW	Radio+cpu :63mW Sleep mode :30μW
External flash	512Kb	512kB

Tableau 2.1. Caractéristiques de MicaZ et TelosB

IV. Architecture d'un réseau de capteurs sans fil

Un réseau de capteur sans fil(RCSF), ou Wireless Sensor Network (WSN) en anglais, est constitué d'un ensemble de nœuds capteurs variant de quelques dizaines d'éléments à plusieurs milliers, placés de manière plus ou moins aléatoire (par exemple par largage depuis un hélicoptère),dans une zone géographique appelée zone de captage, ou zone d'intérêt, afin de surveiller un phénomène physique et de récolter leurs données d'une manière autonome. Les

nœuds capteurs utilisent une communication sans fil pour acheminer les données captées vers un nœud collecteur appelé nœud puits (*sink* en anglais), ou station de base (base station BS). Le puits transmet ensuite ces données par Internet ou par satellite à l'ordinateur central « Gestionnaire de tâches » pour analyser ces données et prendre des décisions. Ainsi, l'utilisateur peut adresser des requêtes aux autres nœuds du réseau, précisant le type de données requises, puis récolter les données environnementales captées par le biais des nœuds puits.

En plus des nœuds capteurs, le modèle peut introduire les super-nœuds, appelés des passerelles (Gateways) [82]. Ces derniers possèdent une source d'énergie importante, la capacité de traitement et de stockage plus élevées comparativement aux nœuds capteurs [82]. Ils peuvent ainsi être utilisés pour exécuter les tâches les plus complexes comme la fusion des données issues des capteurs d'une même zone. Dans le cas le plus simple, les nœuds capteurs seront dans le voisinage direct du puits (communication à un saut). Cependant, dans le cas d'un réseau à grande échelle, ils ne sont pas tous dans le voisinage du puits et les données seront acheminées du nœud source vers le puits en transitant par plusieurs nœuds, selon un mode de communication multi-sauts comme l'illustre la figure 2.2 ci-après.

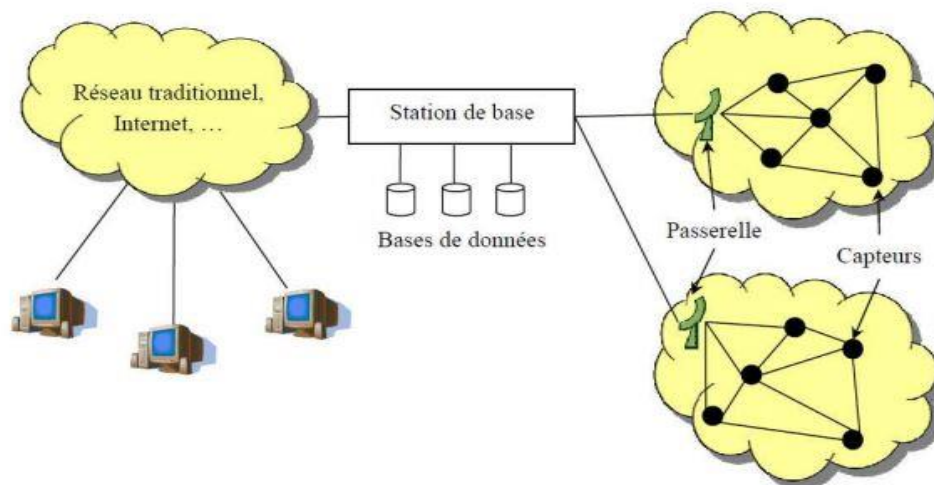


Figure 2.2. Architecture d'un RCSF

V. Classification des RCSFs

Il existe plusieurs critères pour classifier les réseaux de capteurs [83][84][85]. En effet, pour chaque type d'application, ces réseaux ont des caractéristiques différentes. Ils se distinguent par le mode d'acquisition et de livraison des données au puits, la distance entre les nœuds capteurs et le puits, le modèle de mobilité dans le réseau, les capacités des nœuds du réseau, etc.

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

Dans les réseaux de capteurs, le modèle d'acquisition et de livraison des données au puits dépend de l'application de ses exigences, et de la hronologie des données. Il peut être : continu (*time-driven*), événementiel (*event-driven*), à base de requête (*query-driven*), ou hybride.

Les rapports de données peuvent être catégorisés en fonction du temps (continu), des événements, des requêtes et des hybridations [84]. Le modèle continu, est adapté aux applications nécessitant un suivi périodique des données (intervalle de temps constant). Les nœuds capteurs allument périodiquement leurs capteurs et les transmetteurs, détectent l'environnement et transmettent les données d'intérêt à des intervalles de temps périodiques constants. Dans les modèles orientés événements et requêtes, les nœuds de capteurs réagissent immédiatement aux changements soudains et drastiques de la valeur d'un attribut détecté en raison de l'occurrence d'un certain événement ou d'une requête générée par la station de base. Ils sont bien adaptés aux applications critiques dans le temps. Une combinaison des modèles précédents est également possible. Le protocole de routage est fortement influencé par le modèle de rapport de données en ce qui concerne la consommation d'énergie et la stabilité de la route.

5.2. Selon la distance entre les nœuds capteurs et le puits

Dans cette classification, on distingue les réseaux multi-sauts (multi-hop WSN) des réseaux à un seul saut (Single-hop WSN). Les deux sont conçus sur la base de la conservation d'énergie des capteurs. Dans un réseau de capteur à un simple saut, tous les capteurs envoient les informations collectées directement au puits sans passer par aucun autre nœud intermédiaire. Dans le cas du réseau de capteur multi-sauts, les capteurs envoient les données à un nœud voisin (appelé nœud d'agrégation) [85].

5.3. Selon le modèle de mobilité dans le réseau

Cette classification consiste en une combinaison entre la mobilité des nœuds capteurs et celle du puits. Par cette combinaison, nous pouvons distinguer deux grandes catégories de réseaux : *réseaux statiques* et *réseaux dynamiques ou mobiles* (static and mobile networks). La plupart des architectures de réseau supposent que les nœuds de capteurs sont stationnaires. Cependant, la mobilité des puits ou des capteurs est parfois nécessaire dans de nombreuses applications. Le routage des messages depuis ou vers les nœuds mobiles est plus difficile car la stabilité de la route devient un problème important, en plus de l'énergie, de la bande passante, etc. Le réseau peut être dynamique ou statique selon l'application, par exemple dynamique dans une cible application de détection / suivi, alors qu'elle est statique dans la surveillance des forêts pour la

prévention précoce des incendies. La surveillance des événements statiques permet au réseau de fonctionner en mode réactif, générant simplement du trafic lors de la création de rapports. Les événements dynamiques dans la plupart des applications nécessitent des rapports périodiques et, par conséquent, génèrent un trafic important à acheminer vers la station de base.

5.4. Selon les capacités des nœuds du réseau

Dans cette classe, on distingue les réseaux homogènes des hétérogènes [85][86].

En réseau homogène, tous les nœuds sont identiques en termes d'énergie de la batterie et de complexité matérielle. Les réseaux homogènes ont un regroupement statique pur. Dans un type de réseau homogène, une topologie de réseau unique est utilisée et ce réseau n'est pas complexe. Ainsi, en bref, nous pouvons dire que, le réseau de capteurs dans lequel les nœuds de capteurs ayant la même complexité matérielle et la même énergie de la batterie est appelé réseau de capteur homogène.

Dans les réseaux hétérogènes, plus d'un et différents types de nœuds avec des fonctionnalités de batteries différentes sont utilisés. En réseau hétérogène, des différentes topologies sont utilisées, ce qui rend le réseau très complexe. Bref, on peut dire qu'en cas de réseau de capteurs hétérogènes, il existe deux ou plusieurs types de nœuds de réseau différents avec des fonctionnalités différentes et de différente énergie de la batterie.

VI. Caractéristiques des réseaux de capteurs sans fil

Un ensemble de caractéristiques est important pour l'accomplissement des tâches assignées aux applications. Les plus importantes sont :

6.1. Type de service

Le type de service rendu par un réseau de communication conventionnel est évident, il déplace des bits d'un endroit à un autre. Pour un RCSF, le déplacement d'un bit n'est qu'un moyen, mais pas le but réel. Un RCSF devrait fournir des informations utiles et / ou encore des actions sur une tâche donnée. En outre, des concepts tels que la portée des interactions vers des régions géographiques spécifiques ou des intervalles de temps deviendront importants. De nouveaux paradigmes d'utilisation d'un tel réseau sont nécessaires, ainsi qu'une nouvelle interface et de nouvelles façons de penser au service d'un réseau.

6.2. Qualité de service (QoS)

La qualité de service est étroitement liée au type de service d'un réseau. Les exigences de qualité de service traditionnelles -provenant généralement d'applications multimédias - comme le délai de transmission ou la bande passante minimale sont sans importance lorsque les applications tolèrent la latence ou la bande passante. Dans certains cas, seule la livraison occasionnelle d'un paquet peut être plus que suffisante, dans d'autres cas, des exigences de fiabilité très élevées disparaissent. Dans d'autres cas encore, le retard est important lorsque les actionneurs doivent être contrôlés. Le taux de livraison de paquets est une mesure insuffisante, ce qui est pertinent est la quantité et la qualité des informations qui peuvent être extraites à des puits donnés sur les objets observés ou sur la zone. Par conséquent, des concepts de qualité adaptés comme la fiable détection d'événements, la qualité d'approximation ou la carte de température sont importants.

6.3. Tolérance aux fautes

Chaque nœud du réseau est sujet à une défaillance imprévue. La tolérance aux pannes est la capacité à maintenir les fonctionnalités du réseau de capteurs sans interruption due aux défaillances des nœuds de capteurs.

6.4. Durée de vie

Dans de nombreux scénarios, les nœuds devront compter sur une quantité limitée d'énergie (en utilisant des batteries). Il n'est généralement pas possible de remplacer ces sources d'énergie sur le terrain, et simultanément. Un WSN doit fonctionner au moins pour une mission donnée ou aussi longtemps que possible. Donc, la durée de vie d'un WSN devient une valeur très importante. Evidemment, un mode d'opération écoénergétique du WSN est nécessaire.

6.5. Scalabilité

Le nombre de nœuds de capteurs dans un réseau de capteurs peut être de l'ordre de centaines ou même de milliers. Par conséquent, les réseaux de capteurs sont supposés être hautement évolutifs.

6.6. Maintenance

Comme l'environnement d'un WSN et le WSN lui-même changent (batteries épuisées, nœuds défaillants, nouvelles tâches), le système doit s'adapter. Il doit surveiller son propre statut pour changer les paramètres opérationnels ou choisir des différents compromis (par ex. pour fournir une qualité inférieure lorsque les ressources énergétiques deviennent rares). Dans ce sens, le

réseau doit se maintenir, il pourrait également être en mesure d'interagir avec des mécanismes de maintenance externes pour assurer son fonctionnement étendu à une qualité requise.

6.7. Programmation flexible

L'option de reprogrammation ou de reconfiguration devrait être disponible pour que le WSN s'adapte aux changements dynamiques dans le réseau.

VII. Comparaison entre les RCSFs et les réseaux Ad-hoc classiques

Les RCSF sont souvent comparés aux réseaux ad hoc (ou MANET, pour Mobile Ad hoc NETWORK [88]) traditionnels car ces deux types de réseaux partagent beaucoup de points communs [72][89]:

- ✓ Les deux sont des réseaux sans fil distribués qui ne disposent pas d'une infrastructure de réseau significative.
- ✓ Le routage entre deux noeuds peut impliquer l'utilisation de noeuds de relais intermédiaires également connus comme routage multi-sauts.
- ✓ Les noeuds Ad hoc et les capteurs sont généralement alimentés par batterie et il y a donc une grande préoccupation à minimiser la consommation d'énergie.
- ✓ Les deux réseaux utilisent un canal sans fil placé dans un spectre sans licence qui est susceptible d'être perturbé par d'autres technologies radio fonctionnant sur la même fréquence.
- ✓ L'autogestion est nécessaire en raison de la nature distribuée des deux réseaux

Malgré les points sur lesquels ces deux types de réseaux convergent, ils diffèrent sur plusieurs aspects. Parmi les points sur lesquels ils divergent, on cite [72][89][90] :

- ✓ Les MANETs sont généralement «proches» des humains, la plupart des noeuds dans le réseau sont des dispositifs destinés à être utilisés par des êtres humains (par exemple, ordinateurs portables, PDA, terminaux radio mobiles, etc.); inversement, les réseaux de capteurs ne se concentrent pas sur l'interaction humaine mais se concentrent plutôt sur l'interaction avec l'environnement.
- ✓ Le nombre de noeuds dans les réseaux de capteurs, ainsi que la densité de déploiement,

peuvent être supérieurs à ceux des réseaux ad hoc car les nœuds d'un réseau de capteurs sont généralement intégrés dans l'environnement pour détecter un phénomène et éventuellement l'actionner.

- ✓ Les termes tels que unicast et multicast commun dans les MANETs, ne sont guère applicables dans les WSN où nous trouvons d'autres formes de routage comme un-à-plusieurs, plusieurs-à-un, plusieurs-à-plusieurs, etc.
- ✓ La taille du réseau dans le cas du MANET dépend des utilisateurs actifs présents dans la zone de déploiement. Mais dans les WSNs les nœuds dépendent de l'extension de la zone d'observation, des caractéristiques des nœuds et de la redondance requise.
- ✓ Le trafic dans les MANETs sera généralement plus élevé en raison de l'utilisation de services bien connus tels que le Web, le courrier, la vidéo, etc. Mais dans le cas des WSNs, le débit de données est faible sur une période de temps importante, car les réseaux interactifs interagissent avec l'environnement.
- ✓ Dans le cas des réseaux WSNs, les nœuds de capteurs sont des dispositifs de calcul de pointe (Ordinateur portable ou PDA) ou comme dans le cas des WSNs, les nœuds sont des éléments simples et un bon marché implémentant des algorithmes efficaces.
- ✓ Les WSNs sont concevables avec des densités de réseau différentes allant des déploiements très clairsemés à denses qui nécessiteront des protocoles différents ou moins adaptatifs. Cette diversité, bien que présente, n'est pas aussi importante dans les MANETs.
- ✓ Dans les MANETs l'énergie est un recours rare. Mais les WSNs ont des exigences plus strictes sur la durée de vie du réseau. La recharge ou le remplacement des batteries de nœuds WSN est une option inférieure dans les MANET. En raison de cela, l'impact des considérations énergétiques sur l'ensemble de l'architecture du système est beaucoup plus profond dans les WSN que dans les MANETs.
- ✓ Les services QoS dans un MANET sont traditionnellement dictés par les applications

traditionnelles (lowjitter pour les applications vocales) mais pour les WSNs, une QoS entièrement nouvelle est requise, ce qui prend également en compte explicitement la consommation d'énergie.

- ✓ Le déploiement redondant rend le protocole centré sur les données très important dans le cas de WSN qui est hors de propos et étranger dans le cas de MANETs.

VIII. Facteurs et contraintes de conception d'un RCSF

La conception de réseaux de capteurs sans fil (RCSF) nécessite une connaissance approfondie d'une grande variété de domaines de recherche, y compris la communication sans fil, la mise en réseau, le traitement du signal numérique et l'ingénierie logicielle. Par conséquent, il existe plusieurs facteurs qui influencent de manière significative la conception de ces réseaux. Cette partie décrit les principales contraintes conceptuelles et matérielles. Ces facteurs sont importants puisqu'ils servent de guide à la conception et directives pour le développement des algorithmes et protocoles utilisés dans les réseaux de capteurs.

8.1. Contraintes conceptuelles

La conception des RCSFs, leurs protocoles et algorithmes sont guidés par plusieurs facteurs:

8.1.1. Tolérance aux pannes

Les nœuds de capteurs sont vulnérables et fréquemment déployés dans un environnement dangereux. Les nœuds peuvent échouer en raison de problèmes de matériel ou de dommages physiques ou en épuisant leur approvisionnement en énergie. Nous nous attendons à ce que les défaillances de nœud soient beaucoup plus élevées que celles normalement prises en compte dans les réseaux sans fil câblés ou basés sur l'infrastructure. Les protocoles déployés dans un réseau de capteurs devraient être capables de détecter ces défaillances dès que possible et être suffisamment robustes pour gérer un nombre relativement important de défaillances tout en maintenant la fonctionnalité globale du réseau. Ceci est particulièrement pertinent pour la conception du protocole de routage, qui doit garantir que des chemins alternatifs sont disponibles pour le réacheminement des paquets. Des différents environnements de déploiement posent différentes exigences de tolérance aux pannes.

8.1.2. Extensibilité (passage à l'échelle)

Les réseaux de capteurs varient en échelle de plusieurs nœuds à potentiellement plusieurs centaines de milliers. De plus, la densité de déploiement est également variable. Pour collecter des données à haute résolution, la densité de nœuds peut atteindre le niveau où un nœud a plusieurs milliers de voisins dans sa plage de transmission. Les protocoles déployés dans les réseaux de capteurs doivent être évolutifs à ces niveaux et être capables de maintenir des performances adéquates.

8.1.3. Coût de production

Étant donné que de nombreux modèles de déploiement considèrent les nœuds de capteurs comme des dispositifs jetables, les réseaux de capteurs ne peuvent rivaliser avec les approches traditionnelles de collecte d'informations que si les nœuds de capteurs individuels peuvent être produits à très bas prix. Le prix cible envisagé pour un nœud de capteur devrait idéalement être inférieur à 1 dollar [72].

8.1.4. Environnement

Les nœuds capteurs doivent être conçus d'une manière à résister aux différentes et sévères conditions de l'environnement : forte chaleur, pluie, humidité...

8.1.5. Média de transmission

La communication entre les nœuds est normalement mise en œuvre en utilisant la communication radio sur les bandes ISM populaires. Cependant, certains réseaux de capteurs utilisent une communication optique ou infrarouge, cette dernière ayant l'avantage d'être robuste et pratiquement sans interférences.

8.1.6. Consommation énergétique

Comme nous l'avons déjà vu, de nombreux défis des réseaux de capteurs tournent autour des ressources énergétiques limitées. La taille des nœuds limite la taille de la batterie. La conception du logiciel et du matériel doit examiner attentivement les problèmes d'utilisation efficace de l'énergie. Par exemple, la compression de données peut réduire la quantité d'énergie utilisée pour la transmission radio, mais utilise de l'énergie supplémentaire pour le calcul et / ou le filtrage. La politique énergétique dépend également de l'application; dans certaines applications, il peut être acceptable de désactiver un sous-ensemble de nœuds afin de conserver l'énergie alors que d'autres

applications nécessitent que tous les nœuds fonctionnent simultanément.

8.1.7. Agrégation de données

Puisque les nœuds de capteurs peuvent générer des données redondantes significatives, des paquets similaires provenant de plusieurs nœuds peuvent être agrégés de sorte que le nombre de transmissions est réduit. L'agrégation de données est la combinaison de données provenant de différentes sources selon une certaine fonction d'agrégation, par exemple, suppression de duplication, minima, maxima et moyenne. Cette technique a été utilisée pour optimiser l'efficacité énergétique et le transfert de données dans un certain nombre de protocoles de routage. Les méthodes de traitement du signal peuvent également être utilisées pour l'agrégation de données. Dans ce cas, on parle de fusion de données où un nœud est capable de produire un signal de sortie plus précis en utilisant certaines techniques telles que la formation de faisceau pour combiner les signaux entrants et réduire le bruit dans ces signaux.

8.2. Contraintes matérielles

Au minimum, chaque nœud de capteur doit avoir une unité de détection, une unité de traitement, une unité de transmission et une alimentation électrique. En option, les nœuds peuvent avoir plusieurs capteurs intégrés ou des dispositifs supplémentaires tels qu'un système de localisation pour permettre un routage sensible à l'emplacement. Cependant, chaque fonctionnalité supplémentaire entraîne des coûts supplémentaires et augmente la consommation d'énergie et la taille physique du nœud. Ainsi, les fonctionnalités supplémentaires doivent toujours être équilibrées par rapport aux coûts et aux besoins de faible puissance.

IX. Domaines d'application des RCSF

Les réseaux de capteurs sans fil ont acquis une popularité considérable en raison de leur flexibilité dans la résolution de problèmes dans différents domaines d'application et ont le potentiel de changer nos vies de différentes manières. Les WSNs ont été appliqués avec succès dans différents domaines d'application [91].

9.1. Applications militaires [92]

Comme les WSNs peuvent être déployés rapidement et sont auto-organisés, ils sont très utiles dans les opérations militaires pour détecter et suivre les mouvements amicaux ou hostiles. La surveillance sur le champ de bataille peut être effectuée à travers les nœuds de détection afin de

vérifier tout, au cas où plusieurs équipements ou forces seront nécessaires sur le champ de bataille. Les attaques chimiques, nucléaires et biologiques peuvent également être détectées par les nœuds de capteurs. Un exemple de ceci est le «système de détection de sniper» qui peut détecter le feu entrant à travers des capteurs acoustiques et la position du tireur peut également être estimée en traitant l'audio détecté à partir du microphone.

9.2. Application médicales (santé) [93]

Certaines applications médicales pour les réseaux de capteurs prennent en charge les interfaces pour les personnes handicapées, la surveillance intégrée des patients, les diagnostics et l'administration de médicaments dans les hôpitaux ainsi que la télésurveillance des données physiologiques humaines et le suivi et la surveillance des médecins ou des patients. Ils peuvent être utilisés pour la surveillance de la santé des patients (comme la surveillance du taux de glycémie, la détection du rythme cardiaque ou du cancer, etc.) et dans les hôpitaux pour surveiller les signes vitaux et enregistrer les anomalies.

9.3. Applications domestiques [94]

Au fur et à mesure que la technologie progresse, elle fait également son chemin dans nos appareils électroménagers pour leur bon fonctionnement et leurs performances satisfaisantes. Ces capteurs peuvent être trouvés dans les réfrigérateurs, les fours à micro-ondes, les aspirateurs, les systèmes de sécurité et également dans les systèmes de surveillance de l'eau. L'utilisateur peut contrôler les périphériques localement et à distance à l'aide des WSNs.

9.4. Applications environnementales [95]

Ces réseaux de capteurs ont un très grand nombre d'applications dans l'environnement. Ils peuvent être utilisés pour suivre le mouvement des animaux, des oiseaux et les enregistrer. La surveillance de la terre, du sol, du contexte de l'atmosphère, de l'irrigation et de l'agriculture de précision peut être réalisée grâce à ces capteurs. Ils peuvent également être utilisés pour la détection d'incendies, d'inondations, de tremblements de terre et d'épidémies chimiques / biologiques, etc. Un exemple courant est de 'Zebra Net'. Le but de ce système est de suivre et de surveiller les mouvements et les interactions des zèbres en eux-mêmes et avec d'autres espèces.

9.5. Transport [92]

Des informations sur le trafic en temps réel sont collectées par les WSNs pour alimenter ultérieurement les modèles de transport et alerter les conducteurs des problèmes de congestion et

de trafic.

9.6. Agriculture de précision

L'utilisation d'un réseau sans fil libère l'agriculteur de l'entretien du câblage dans un environnement difficile. L'automatisation de l'irrigation permet une utilisation plus efficace de l'eau et réduit les déchets.

9.7. Application industrielles

Les WSNs peuvent également être utilisés pour surveiller la bonne exécution des étapes du processus dans les déploiements industriels en fournissant à l'opérateur des moyens d'adapter les paramètres du processus à la demande [94]. La surveillance et la maintenance des machines constituent un autre domaine d'application important dans ce contexte. Ici, les WSNs peuvent être utilisés pour la maintenance conditionnelle de machines exploitant la capacité de traitement de données locales et fournissant des données de surveillance en temps réel afin d'améliorer l'utilisation et la durée de vie de l'équipement surveillé..

La figure 2.3 montre quelques domaines d'application cités précédemment.

X. Communication dans les RCSFs

10.1. Pile protocolaire

La pile de protocoles [72] utilisée par le puits et tous les nœuds de capteurs est illustrée dans la Figure 2.4. Cette pile de protocoles combine la connaissance de l'alimentation et du routage, intègre les données avec les protocoles réseau et communique efficacement l'énergie via le support sans fil. La pile protocolaire comprend la couche physique, la couche de liaison de données, la couche de transport, la couche application, ainsi que le plan de synchronisation, le plan de localisation, le plan de gestion topologique, le plan de gestion de la mobilité et le plan de gestion des tâches. La couche physique répond aux besoins de techniques de modulation, de

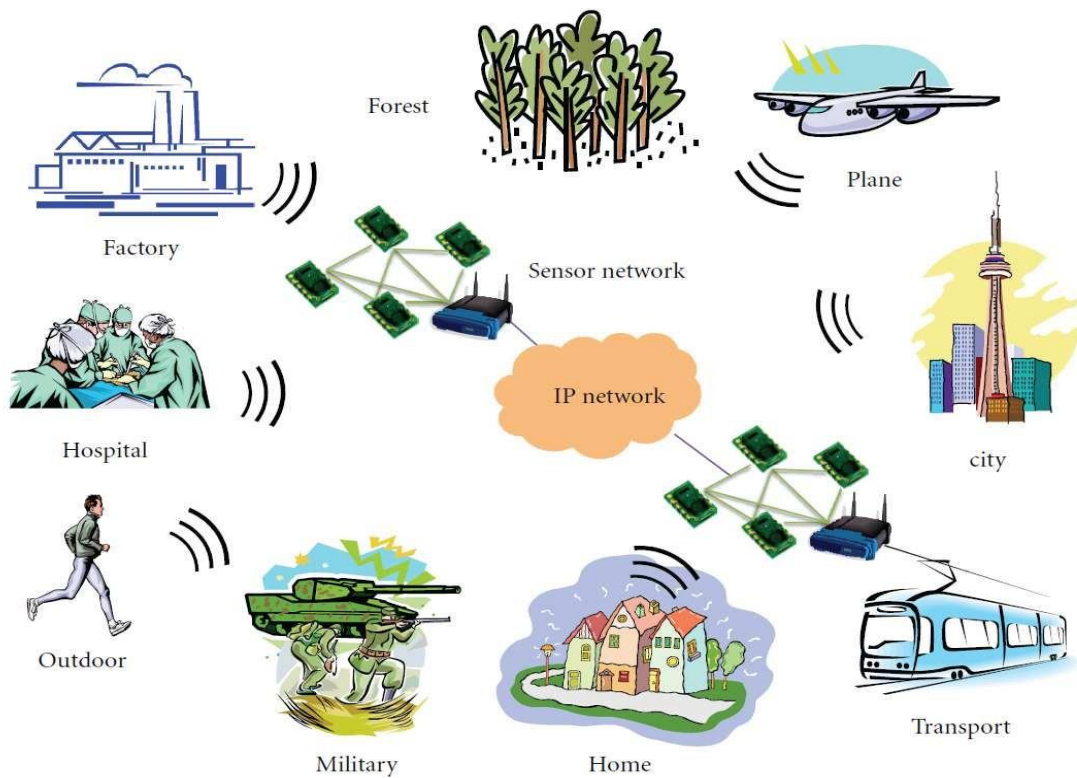


Figure 2.3. Quelques domaines d'application pour les RCSFs

transmission et de réception simples mais robustes. Comme l'environnement est bruyant et que les nœuds de capteurs peuvent être mobiles, la couche de liaison est responsable d'assurer une communication fiable grâce à des techniques de contrôle des erreurs et de gérer l'accès au canal par le protocole MAC, afin de minimiser les collisions avec les voisins. Selon les tâches de détection, différents types de logiciels d'application peuvent être créés et utilisés sur la couche d'application. La couche réseau prend en charge le routage des données fournies par la couche de transport. La couche de transport aide à maintenir le flux de données si l'application de réseau de capteurs l'exige. En outre, les plans de gestion de l'alimentation, de la mobilité et des tâches surveillent la puissance, le mouvement et la répartition des tâches entre les nœuds de capteurs. Ces plans aident les nœuds de capteurs à coordonner la tâche de détection et à réduire la consommation d'énergie globale.

a. Couche physique

- ✓ La couche physique est responsable de la sélection des fréquences, la génération des fréquences porteuses, la détection du signal, la modulation et le cryptage des données.

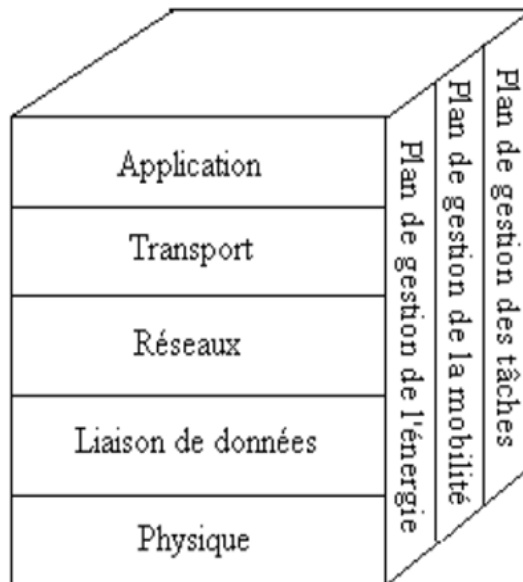


Figure 2.4. Pile protocolaire pour les RCSFs

- ✓ La bande ISM 915 MHz a été largement suggérée. "Ultra large bande (UWB)" est un candidat intéressant en raison de la faible puissance de transmission, des circuits simples, de l'utilisation de la modulation de position d'impulsion, de la résilience aux trajets multiples.
- ✓ La modulation dépend des contraintes de conception de l'émetteur-récepteur et du matériel, qui visent la simplicité, la faible consommation d'énergie et le faible coût par unité. Les schémas de modulation binaire sont plus simples à mettre en œuvre et sont considérés comme plus économes en énergie.

La couche physique répond aux besoins de techniques de modulation, de transmission et de réception simples mais robustes.

b. Couche liaison de données

La couche de liaison de données est responsable du multiplexage des flux de données, de la détection des trames de données et de l'accès au support et du contrôle des erreurs. Elle assure des connexions point-à-point et point-multipoint fiables dans un réseau de communication. Plus spécifiquement, nous discutons des stratégies d'accès au support et de contrôle des erreurs pour les réseaux de capteurs.

c. Couche réseau

La couche réseau prend en charge le routage des données fournies par la couche de transport.

- ✓ La couche réseau est conçue selon certains principes.
- ✓ L'efficacité de la puissance est toujours un facteur important: des itinéraires efficaces sur le plan énergétique sont trouvés comme la route avec une puissance maximale disponible (PA); itinéraire avec consommation d'énergie minimale (ME); itinéraire avec un minimum de sauts (MH); etc.
- ✓ Les réseaux de capteurs sont principalement centrés sur les données: les données sont données uniquement au nœud qui exécutera la tâche. C'est ce que l'on appelle la diffusion d'intérêts, le récepteur diffuse la disponibilité des données et la publicité diffusée par les nœuds de capteurs. L'événement attend une demande des nœuds intéressés.
- ✓ L'agrégation de données n'est utile que lorsqu'elle n'entrave pas l'effort de collaboration des nœuds de capteurs: l'agrégation de données combine les données de nombreux nœuds dans une forme plus compacte avant de les transmettre pour le traitement.
- ✓ Un réseau de capteurs idéal a un adressage basé sur les attributs et une connaissance de la localisation: l'utilisateur souhaite interroger un attribut des phénomènes; plutôt que d'interroger un nœud individuel.
- ✓ Les autres caractéristiques comprennent la synchronisation de l'horloge, la tolérance aux pannes, etc.

d. Couche transport

- ✓ La couche de transport aide à maintenir le flux de données si l'application de réseau de capteurs le nécessite.
- ✓ Il est nécessaire lorsqu'un système doit être accessible via Internet ou d'autres réseaux externes.
- ✓ Le «protocole de contrôle de transmission (TCP)» doit être divisé en deux parties. L'un connecte le réseau de capteurs à l'autre réseau comme Internet et l'autre connecte le nœud récepteur aux nœuds capteurs. Le schéma d'adressage ici n'est pas basé sur l'adressage global. Il est basé sur la dénomination basée sur les attributs. Des facteurs tels que la consommation d'énergie, l'évolutivité, etc sont pris en compte.

e. Couche application

- ✓ Selon les tâches de détection, différents types de logiciels d'application peuvent être créés et utilisés sur la couche d'application.

- ✓ Ces protocoles rendent le matériel et les logiciels des couches inférieures transparents afin que les administrateurs système puissent modifier les paramètres.
- ✓ Trois de ces protocoles sont: «protocole de gestion des capteurs (SMP)», protocole d'attribution des tâches et de publication des données (TADAP) », « requête de sonde et protocole de diffusion des données »(SQDDP).
- ✓ SMP fournit les opérations logicielles nécessaires à l'exécution de tâches telles que l'introduction de règles liées à l'agrégation de données, à la dénomination basée sur des attributs, à la mise en cluster de nœuds de capteurs; synchronisation temporelle des nœuds de capteurs; déplacement des nœuds de capteurs, etc.

Quant aux niveaux (plans) intégrés dans la pile protocolaire, ils ont les fonctions suivantes :

f. Plan de gestion d'énergie

Lorsque le niveau d'énergie d'un nœud capteur est faible, il transmet à ses voisins qu'il est faible et ne peut pas participer aux messages de routage. Ceci est géré par un plan de gestion d'énergie.

g. Plan de gestion de mobilité

Le plan de gestion de la mobilité détecte et enregistre le mouvement des nœuds de capteurs. Les nœuds de capteurs peuvent garder une trace de leurs voisins.

h. Plan de gestion de tâche

Le plan de gestion des tâches équilibre et planifie les tâches de détection attribuées à une région spécifique. Tous les nœuds de capteurs n'effectuent pas la détection en même temps.

10.2. Standards de communication pour les RCSFs

Il y a une multitude de normes sans fil comme le Wi-Fi (le standard IEEE 802.11) et le WiMax (le standard IEEE 802.16) qui s'adressent au transport des données à haut débit. Certains dispositifs comme les capteurs n'ont pas besoin d'une largeur de bande très élevée, mais plutôt d'un temps de latence faible ainsi qu'une consommation d'énergie très basse, pour une longue durée de vie sur batterie et un grand nombre de dispositifs. D'où la nécessité de concevoir d'autres normes sans fil capables de répondre à ces exigences. Parmi les standards les plus adaptés à être exploités dans les réseaux de capteurs sans fil on retrouve les standards *Bluetooth* et *ZigBee* présentés ci-après.

10.2.1. Bluetooth

Le Bluetooth (IEEE 802.15.1) [96] a été développé comme un protocole sans fil pour la communication à courte portée dans les réseaux personnels sans fil (PAN) en tant que remplacement de câble pour les appareils mobiles. Il utilise les bandes radio 868 et 915 MHz et 2,4 GHz pour communiquer à 1 Mb par seconde entre sept appareils maximum. Le Bluetooth est principalement conçu pour optimiser les fonctionnalités de mise en réseau ad hoc. Certaines de ses fonctions communes transmettent et synchronisent des données, par ex. entre un PDA (assistant numérique personnel) et un ordinateur, l'accès sans fil aux réseaux locaux et la connexion à Internet. Il utilise une communication à spectre étalé à saut de fréquence (FHSS), qui transmet des données sur différentes fréquences à différents intervalles de temps. Le Bluetooth utilise un protocole MAC (Medium Access Control) basé sur un maître esclave [96].

10.2.2. ZigBee

La norme ZigBee [97] est basée sur la norme IEEE 802.15.4 [98]. La norme IEEE 802.15.4 définit les couches physique et MAC (Medium Access Control) pour les réseaux personnels sans fil à faible débit (WPAN) [99][100]. La couche physique prend en charge trois bandes de fréquences avec des débits bruts différents: 2 450 MHz (250 kbs-1), 915 MHz (40 kbs-1) et 868 MHz (20 kbs-1). Elle prend également en charge les fonctionnalités de sélection des canaux, d'estimation de la qualité de la liaison, de mesure de l'énergie et d'évaluation claire des canaux. ZigBee standardise à la fois le réseau et la couche application. La couche réseau est chargée de l'organisation et du routage sur un réseau multi-sauts, en spécifiant les différentes topologies de réseau: étoile, arbre, pair-à-pair et maillage. La couche application fournit un cadre pour le développement d'applications distribuées et la communication.

Mis à part l'industrie agricole et alimentaire, il est largement utilisé dans le contrôle de la construction de maisons, l'automatisation, la sécurité, l'électronique grand public, les périphériques d'ordinateurs personnels, la surveillance médicale et les jouets. Ces applications nécessitent une technologie qui offre une longue durée de vie, une fiabilité, une installation automatique ou semi-automatique, la possibilité d'ajouter ou de supprimer facilement des nœuds de réseau, des signaux pouvant traverser les murs et les plafonds et un coût de système réduit [101].

XI. Modélisation pour les réseaux de capteurs

La méthode de modélisation des réseaux de capteurs consiste à créer un modèle du système que l'on veut développer. Une fois ce modèle est créé, on obtient un prototype du système. Ce prototype virtuel est ensuite analysé et modifié jusqu'à ce qu'il respecte les spécifications du problème. On peut alors implémenter le prototype. Le prototypage virtuel permet de développer rapidement des systèmes complexes. Dans les méthodes de prototypage, on retrouve les simulateurs de réseaux. Nous proposons dans cette section un état de l'art des simulateurs de réseaux dans un premier temps, puis les simulateurs spécifiques des réseaux de capteurs puis les simulateurs prenant en compte une modélisation de l'environnement du réseau afin d'obtenir des simulations plus réalistes.

11.1. Simulateurs de réseaux généraux

Parmi les simulateurs de réseaux on trouve les simulateurs de réseaux classiques, ces simulateurs ont été conçus pour modéliser et simuler des réseaux classiques tels que les réseaux filaires, les réseaux sans fil voire les réseaux ad hoc. Dans ces réseaux, le facteur d'énergie n'était pas encore le plus important. On retrouve un des simulateurs de réseaux les plus utilisés : NS-2 (The Network Simulator) [102] codé en C++. Le projet VINT correspond à l'origine de NS-2. Breslau et al. [103] estiment qu'il faut un seul simulateur de réseaux pour la communauté scientifique. Ce souhait donne naissance au projet VINT. L'intérêt d'un simulateur unique correspond surtout à la facilité à comparer différentes solutions. NS-2 est un simulateur à événement discret. NS-2 propose quatre niveaux d'abstraction ce qui permet d'adapter le simulateur aux différents intérêts. En effet, certains souhaiteront des informations bas-niveau, pour étudier par exemple l'effet de la propagation de l'onde radio dans un environnement alors que d'autres étudient les protocoles de routages et ne souhaitent que des informations au niveau réseau. NS-2 était d'abord destiné aux réseaux filaires ce qui explique certainement la simplicité des modèles de propagations radio de ce simulateur. Malgré les différents niveaux d'abstraction, NS-2 est essentiellement utilisé par les chercheurs qui s'intéressent aux protocoles de routage et/ ou aux protocoles d'accès au médium. Le code NS-2 étant ouvert, ceci permet à chacun d'ajouter sa pierre à l'édifice. Il existe donc une large bibliothèque d'écrivant toutes les informations dont on a besoin de la configuration physique d'un nœud jusqu'à l'application déployée sur le réseau. Ceci permet donc de mieux comparer les dernières avancées avec l'état de l'art. Dans le monde de la recherche NS-2 reste

un simulateur très utilisé. L'outil Nam [104][105] permet de visualiser l'exécution d'une simulation. NS-2 génère des traces qui détaillent tous les événements de la simulation. Grâce à l'une de ces traces, Nam génère le "film" de la simulation. Cet outil permet surtout d'avoir une vue de l'exécution de la simulation beaucoup plus simple que la vue du fichier trace qui dans certains cas fait largement plus de 100 000 lignes.

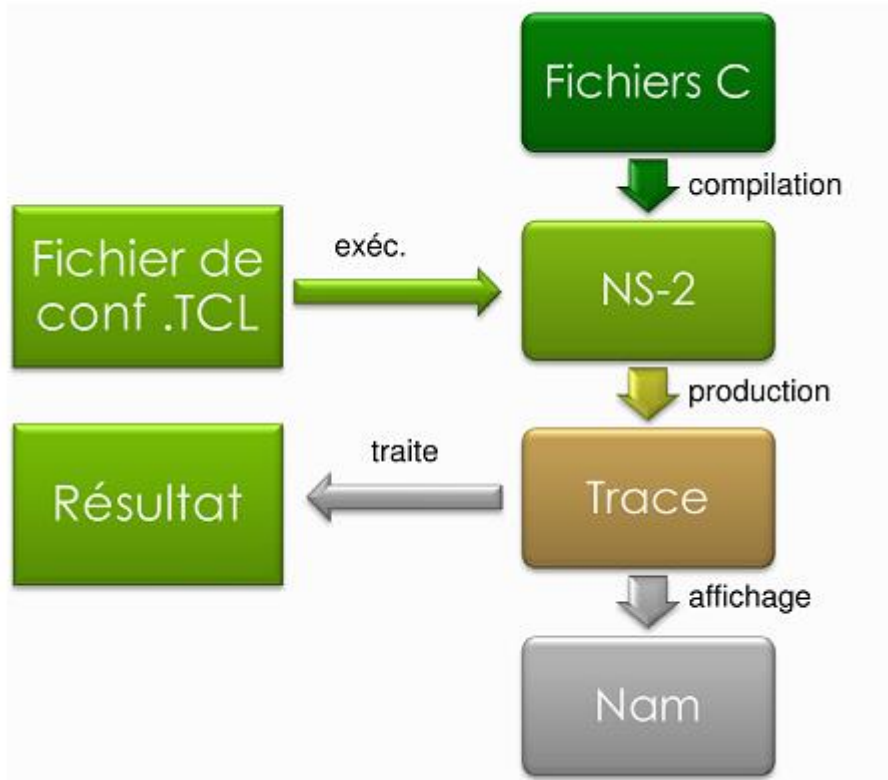


Figure 2.5. Présentation de NS2

Cependant, NS-2 n'est pas le simulateur parfait. Tout d'abord, l'utilisation n'est pas optimisée, il est nécessaire d'effectuer toutes ces tâches pour chaque simulation : modifier le fichier TCL, lancer la simulation, analyser les résultats, produire les graphiques. Ce qui est un point faible pour l'utilisateur ayant beaucoup de simulations à exécuter. D'autre part à cause des multiples contributions qui enrichissent NS-2, ce simulateur est devenu très complexe. En effet, les composants qui constituent un réseau, ou les nœuds d'un réseau ne sont pas toujours bien séparés dans NS-2. De ce fait pour l'implémentation de protocoles qui contiennent des optimisations inter-couches, il est nécessaire de modifier des parties du code à différents niveaux, et dans NS-2 les couches n'étant pas clairement séparées, l'implémentation devient vite illisible. Enfin NS-2 ne propose pas de modèle d'environnement, mais plutôt des outils pour essayer de reproduire un environnement. Concernant la communication on peut choisir soit l'utilisation d'un flux de

type TCP/IP, UDP, etc. ou bien considérer une loi de Poisson qui modélise bien l'émission de paquets au niveau d'un nœud. Pour les réseaux de capteurs, ces deux approches sont tout à fait discutables.

OMNeT++ [106] est un environnement de simulation d'événements discrets. Son domaine d'application principal est la simulation des réseaux de communication, mais en raison de son architecture générique et flexible, il est utilisé avec succès dans d'autres domaines tels que la simulation des systèmes informatiques complexes, de files d'attente ou d'architectures matérielles. OMNeT ++ fournit une architecture de composants pour les modèles. Les composants (modules) sont programmés en C ++, puis assemblés en composants plus grands et en modèles utilisant un langage de haut niveau (NED). La réutilisabilité des modèles est gratuite. OMNeT ++ a une prise en charge étendue de l'interface graphique et, grâce à son architecture modulaire, le noyau de simulation (et les modèles) peuvent être facilement intégrés dans vos applications. Bien qu'OMNeT ++ ne soit pas un simulateur de réseau en soi, il gagne actuellement en popularité en tant que plate-forme de simulation de réseau dans la communauté scientifique ainsi que dans les environnements industriels, et en créant une large communauté d'utilisateurs. Les composants d'OMNeT ++ sont les suivants: compilateur de bibliothèque de noyau de simulation pour le langage de description de topologie NED, OMNeT ++ IDE basé sur l'interface graphique de la plateforme, Eclipse pour l'exécution de la simulation, liens vers l'interface utilisateur de la ligne de commande pour les simulations (Cmdenv) makefile creation tool, etc.), documentation et simulations d'échantillons. Pour la simulation de réseaux de grandes tailles, l'utilisation de simulateurs distribués n'est pas négligeable. On retrouve dans ce cadre le simulateur GTNetS [107] (Georgia Tech Network Simulator) qui permet de distribuer les calculs du simulateur sur plusieurs machines. GTNetS n'est pas non plus un simulateur dédié aux réseaux de capteurs.

OPNET (Optimized Network Engineering Tools) [108], est un simulateur de réseau à événement discret. OPNET a été initialement développé au Massachusetts Institute of Technology (MIT) et depuis 1987 est devenu un logiciel commercial. OPNET est un outil de simulation de niveau réseau basé sur des événements de haut niveau. La simulation fonctionne au niveau «paquet». Elle fournit un environnement de développement complet supportant la modélisation des réseaux de communication et des systèmes distribués. Le comportement et la performance des systèmes modélisés peuvent être analysés en effectuant des simulations d'événements discrets. Les auteurs dans [109] ont simulé un processus de diffusion en utilisant les simulateurs OPNET, ns-2. On a montré que les résultats obtenus par OPNET sont à peine comparables à ceux obtenus par les deux autres simulateurs, ce qui explique la différence dans

leur comportement. Les divergences n'étaient pas seulement quantitatives mais aussi qualitatifs.

11.2. Simulateurs dédiés aux réseaux de capteurs

Avrora [110] est un simulateur spécialement conçu pour les WSN construit en Java. Avrora peut également simuler des nœuds de capteurs MICA2 à microcontrôleur AVR. Ce simulateur a été développé par l'Université de Californie, Los Angeles Compilers Group. Avrora fournit une large gamme d'outils qui peuvent être utilisés pour simuler des WSN. Ce simulateur combine les mérites de TOSSIM [112] et d'ATEMU [111], et limite leurs inconvénients. Avrora ne fournit pas d'interface graphique. Avrora prend également en charge la simulation de consommation d'énergie. Ce simulateur fournit des sources ouvertes et des documents en ligne. Cependant, ce simulateur a quelques inconvénients. Il n'a pas d'interface graphique. En outre, Avrora ne peut pas simuler les algorithmes de gestion de réseau car il ne fournit pas d'outils de communication réseau.

Atemu [111] est aussi un simulateur pour réseaux de capteurs très précis. C'est un émulateur d'un processeur AVR pour WSN construit en C; AVR est un microcontrôleur monopuce couramment utilisé dans la plate-forme MICA. ATEMU fournit une interface graphique, Xatdb; les utilisateurs peuvent utiliser cette interface graphique pour exécuter des codes sur des nœuds de capteurs, des codes de débogage et des exécutions de programmes de surveillance. Les gens peuvent lancer ATEMU sur le système d'exploitation Solaris et Linux. ATEMU est un émulateur spécifique pour les WSN; il peut aider les utilisateurs à exécuter TinyOS sur le matériel MICA2. ATEMU peut imiter non seulement la communication entre les capteurs, mais aussi toutes les instructions mises en œuvre dans chaque capteur. Cet émulateur fournit des sources ouvertes et des documents en ligne.

TOSSIM [112] est un émulateur spécifiquement conçu pour un WSN fonctionnant sur TinyOS, qui est un système d'exploitation open source ciblant le système d'exploitation embarqué. En 2003, TOSSIM a été développé par l'équipe de projet TinyOS [113][114] de UC Berkeley. TOSSIM est un émulateur de réseau à événements discrets au niveau de bit construit en Python [Python], un langage de programmation de haut niveau mettant l'accent sur la lisibilité du code, et C ++. Les utilisateurs peuvent exécuter TOSSIM sur les systèmes d'exploitation Linux ou sur Cygwin sous Windows. TOSSIM fournit également des sources ouvertes et des documents en ligne.

PowerTOSSIM [114] est l'extension de TOSSIM qui contient un modèle de

consommation d'énergie. Pour les valeurs de consommation, les auteurs se sont basés sur les nœuds de type Mica2 (cousin du MicaZ [115]). Les auteurs connaissent les consommations des différents composants de ce nœud suivant leurs états. Il faut donc connaître l'état de chaque composant d'un nœud pendant la simulation. Grâce au modèle de simulation basé sur TinyOS, on connaît immédiatement l'état des composants autres que le microcontrôleur puisque les changements d'état correspondent à des événements dans TinyOS et donc dans TOSSIM. Néanmoins plusieurs composants du nœud sont parfois abstraits dans TOSSIM par un seul composant. L'estimation de la consommation du microcontrôleur est plus délicate. En effet il faut analyser le code pour être capable de compter le nombre d'exécutions de chaque bloc d'instruction, et il faut faire correspondre chaque bloc d'instructions avec son code en assembleur. De ce fait pendant la simulation, on note le nombre de passages, d'exécution, de chaque bloc d'instructions. Sachant combien d'instructions élémentaires contiennent chaque bloc de base, on en déduit le nombre d'instructions effectuées par le microcontrôleur. Et ceci nous permet d'obtenir sa consommation énergétique. L'inconvénient majeur de PowerTOSSIM et TOSSIM est que les applications doivent obligatoirement être écrites en NesC [116].

Les auteurs de **Worldsens** [117] proposent un autre concept qui consiste à relier deux simulateurs WSN et WSim. **WSNet** est un simulateur à événements discrets à base de composants comme le simulateur NS-2. Il est composé de nombreux blocs qui modélisent les propriétés des nœuds capteurs et du support radio. Le modèle de nœud capteur comprend l'abstraction matérielle et logicielle et la modélisation du comportement de nœud (par exemple, la mobilité). WSN peut être utilisé pour évaluer le haut niveau de conception, comme le modèle de trafic, le dimensionnement des applications et le réglage des paramètres de protocole. C'est l'un des deux simulateurs de Worldsens [117], un environnement intégré pour le développement et le prototypage rapide d'applications de réseaux de capteurs sans fil. Worldsens inclut également un simulateur de bas niveau pour permettre le perfectionnement du développement d'applications WSN. Le simulateur de cycle précis, **WSim** [118]. Il est basé sur une simulation de plate-forme complète à cycle précis utilisant des minuteries commandées par un microprocesseur. Il offre de nombreuses descriptions de blocs matériels de composants au niveau de la puce. Une plate-forme de nœud capteur peut être construite en décrivant l'interconnexion physique entre ces composants. WSim peut également gérer la simulation et le débogage de code binaire de cible réel. La résolution temporelle peut être au niveau de la nanoseconde. En combinant WSim et WSN, Worldsens peut fournir un flux de conception complet d'application WSN, depuis les choix de conception de haut niveau

jusqu'à l'implémentation de code cible, le débogage et l'analyse des performances.

11.3. Simulateurs prenant en compte un modèle d'environnement

Ces simulateurs dédiés aux réseaux de capteurs permettent d'estimer et d'étudier assez finement les réseaux de capteurs comme on le souhaite. Mais l'environnement reste un facteur permettant d'obtenir des données plus réalistes. En effet, dans un réseau de capteurs, le trafic dépend des capteurs et donc de l'environnement. On retrouve donc avec les simulateurs présentés précédemment, des incertitudes concernant la circulation du trafic lié à l'environnement. Nous allons décrire quelques simulateurs prenant en compte un modèle d'environnement [119][120].

SensorSim [121] est un simulateur à événements discrets pour les réseaux de capteurs. Il est basé sur le simulateur NS-2 avec de diverses fonctionnalités supplémentaires : Canaux de détection et modèles de capteurs, modèles de batteries, piles de protocoles légères pour microcapteurs sans fil, scénario de génération et la simulation hybride. L'objectif fondamental derrière le développement de SensorSim est de fournir une compréhension plus approfondie des WSNs et un cadre profond pour le développement de nouveaux protocoles avec des applications et des techniques d'évaluation des performances pour les réseaux de capteurs. SensorSim a hérité de diverses fonctionnalités d'autres simulateurs pilotés par événements ayant un mode de simulation hybride qui facilite de nouvelles fonctionnalités comme l'interaction de nœuds réels et simulés, de nouveaux protocoles de communication et une interaction utilisateur en temps réel avec l'interface GUI.

Outre divers avantages et de nouvelles fonctionnalités pour simuler le réseau WSN, SensorSim a un problème d'évolutivité. Et en raison de l'incapacité des développeurs à fournir un support pour la diffusion publique, SensorSim aujourd'hui n'est pas actuellement disponible pour le public.

J-Sim [122] s'inspire de SensorSim pour la modélisation de l'environnement. J-Sim est un simulateur de réseau à événements discrets construit en Java. Ce simulateur fournit une bibliothèque GUI, qui permet aux utilisateurs de modéliser ou de compiler le langage de modélisation mathématique, un «langage basé sur le texte» écrit sur les modèles J-Sim. J-Sim fournit des modèles open sources et des documents en ligne. Ce simulateur est couramment utilisé dans les domaines de la physiologie et de la biomédecine, mais il peut également être utilisé dans la simulation du WSN. De plus, J-Sim peut simuler des processus en temps réel [123].

XII. Protocoles de routage

Le routage est un processus d'acheminement des messages d'un nœud source du réseau vers un nœud destinataire à travers un réseau de connexion donnée. Le routage dans les réseaux de capteurs sans fil est une tâche exigeante. Cette demande conduit à un certain nombre de protocoles de routage qui utilisent efficacement les ressources limitées disponibles sur les nœuds de capteurs. Tous ces protocoles trouvent généralement le chemin d'énergie minimum. Des protocoles de routage différents sont proposés pour les WSN en tenant compte des défis qui affectent les performances de routage.

Le problème qui se pose dans le contexte des réseaux ad hoc est l'adaptation de la méthode d'acheminement utilisée avec le grand nombre d'unités existant dans un environnement caractérisé par de modestes capacités de calcul et de sauvegarde et de changements rapides de topologies. Il semble donc important que toute conception de protocole de routage doive étudier les problèmes suivants :

- ✓ La minimisation de la charge du réseau
- ✓ Offrir un support pour pouvoir effectuer des communications multipoints fiables
- ✓ Assurer un routage optimal
- ✓ Offrir une bonne qualité concernant le temps de latence

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

- ✓ Les protocoles "centralisés" ou dédiés
- ✓ Les protocoles proactive
- ✓ Les protocoles réactifs

12.1. Protocoles centralisés

Les protocoles dits "centralisés" fonctionnent sur des réseaux de petite taille. Or, même si les réseaux de capteurs sans fil sont apparentés aux réseaux ad hoc, les spécificités, les objectifs et les besoins sont différents. Les protocoles dédiés pour les réseaux de capteurs, sont peu nombreux, les implémentations également. Les nœuds demandent une gestion soigneuse des ressources. Ces algorithmes dédiés sont répartis en quatre catégories [124] :

- ✓ Protocoles hiérarchiques
- ✓ Protocoles basés sur la localisation

- ✓ Protocoles "Data centric"
- ✓ Protocoles basé sur la qualité de service (QoS)

12.1.1. Protocoles hiérarchiques

Dans l'architecture hiérarchique, certains nœuds d'énergie plus élevée peuvent être utilisés pour traiter et envoyer les informations à la station de base tandis que les nœuds d'énergie inférieure peuvent effectuer la détection dans la zone cible. En d'autres termes, le réseau est partitionné en plusieurs clusters. Dans chaque cluster, un nœud est sélectionné en tant que chef de cluster avec certains membres de cluster. Une hiérarchie à deux niveaux est formée dans laquelle les chefs de clusters sont dans le niveau supérieur tandis que les membres de cluster sont créés dans un niveau inférieur. Les membres de cluster détectent les données de l'environnement physique et les envoient à leurs chefs de clusters respectifs. Les chefs de clusters traitent les données et les transmettent au collecteur directement ou de manière multi-hop.

On retrouve notamment le protocole LEACH [125], qui choisit aléatoirement les nœuds cluster-heads et attribue ce rôle aux différents nœuds selon la politique de gestion Round-Robin [126] pour garantir une dissipation équitable d'énergie entre les nœuds. Dans LEACH, le rôle du chef du cluster est transféré périodiquement entre les nœuds du réseau afin de répartir la consommation d'énergie. La performance de LEACH est basée sur des tours. Ensuite, un chef de cluster est élu à chaque tour. Pour cette élection, le nombre de nœuds qui n'ont pas été des chefs de clusters et le pourcentage des chefs de clusters sont utilisés. Une fois le chef de cluster est défini dans la phase d'installation, il établit une planification TDMA pour les transmissions dans son cluster [126]. Cette planification permet aux nœuds d'éteindre leurs interfaces lorsqu'ils ne seront pas utilisés. Le chef de cluster permet de router les informations vers la station de base et il est également responsable de l'agrégation des données. Comme le chef de cluster contrôle les capteurs situés dans une zone proche, l'agrégation de données effectuée par ce chef permet d'éliminer la redondance. On retrouve également d'autres protocoles tels que : PEGASIS [127], TEEN [128] ou encore DirQ [129].

12.1.2. Protocoles basés sur la localisation

Dans les protocoles basés sur la localisation, les nœuds de capteurs sont dispersés de manière aléatoire dans une zone d'intérêt et sont principalement connus par la position

géographique où ils sont déployés. Ils sont situés principalement au moyen du GPS. La distance entre les nœuds est estimée par la force du signal reçu de ces nœuds et les coordonnées sont calculées en échangeant des informations entre les nœuds voisins. Cependant, étant donné que la prise en charge de la localisation nécessite des composants matériels spécifiques et impose un surcoût de calcul significatif aux nœuds de capteurs, cette approche ne peut pas être facilement utilisée dans les réseaux de capteurs sans fil soumis à des contraintes de ressources.

On retrouve notamment le protocole GEAR [130] qui est un protocole de routage écoénergétique proposé pour le routage des requêtes afin de cibler les régions dans le domaine des capteurs. Les capteurs seront équipés de matériel de localisation comme le système de localisation GPS. Avec l'aide de cela, les capteurs peuvent connaître leurs positions actuelles. Les capteurs peuvent connaître leur emplacement, leur énergie résiduelle ainsi que celles de ses voisins. Afin de sélectionner les capteurs pour acheminer le paquet vers la destination, ils utilisent des méthodes conscientes de l'énergie en utilisant des informations géographiques. À ce stade, GEAR utilise l'acheminement géographique récursif pour répartir les paquets dans la région cible. Il existe aussi GAF [131] qui lui aussi est un protocole basé sur la localisation.

12.1.3. Protocoles Data centric

Dans les protocoles centrés sur les données, le récepteur envoie des requêtes à certaines régions et attend des données provenant des capteurs situés dans les régions sélectionnées. Puisque les données sont demandées par le biais de requêtes, la dénomination basée sur les attributs est nécessaire pour spécifier les propriétés des données. L'idée principale de ces protocoles est de combiner les données provenant de différentes sources en route (agrégation dans le réseau) en éliminant la redondance, minimisant le nombre de transmissions; économisant ainsi l'énergie du réseau et prolongeant sa durée de vie. Contrairement au routage de bout en bout traditionnel, le routage data centric trouve des itinéraires de plusieurs sources vers une destination unique qui permet la consolidation de données redondantes dans le réseau.

On retrouve notamment le protocole SPIN [132] qui est une famille de protocoles adaptatifs qui utilisent la négociation de données et des algorithmes adaptatifs aux ressources. Puisque SPIN est un protocole de routage centré sur les données, cela suppose que tous les nœuds du réseau sont des stations de base et les nœuds à proximité ont des données similaires. L'idée principale derrière SPIN est de nommer les données en utilisant des descripteurs de haut niveau ou des métadonnées. Comme tous les nœuds peuvent être considérés comme des stations

de base, toutes les informations sont diffusées à chaque nœud du réseau. Ainsi, l'utilisateur peut interroger n'importe quel nœud et peut obtenir l'information immédiatement. Les nœuds de ce réseau utilisent un nom de haut niveau pour décrire leurs données collectées appelées méta-données. Il existe beaucoup d'autres protocoles tels que : Direct Diffusion [133], Rumor [134], COUGAR [135] ou encore ACQUIRE [136].

12.1.4. Protocoles basés sur la qualité de service

En plus de minimiser la consommation d'énergie, il est également important de considérer les exigences de qualité de service (QoS) en termes de délai, de fiabilité et de tolérance aux pannes dans les WSN. Ils existent plusieurs protocoles de routage basé sur la QoS qui aide à trouver un équilibre entre la consommation d'énergie et les exigences de QoS.

On retrouve notamment le protocole SAR [137] qui est un protocole de routage d'assignation séquentielle. Dans le protocole SAR, trois points ou facteurs sont très importants. Ce sont: les ressources d'énergie, la QoS sur chaque chemin et le niveau de priorité de chaque paquet. Le protocole SAR crée un arbre et, par là, il crée plusieurs chemins entre l'évier et les capteurs. Le but du protocole SAR est d'atteindre l'efficacité énergétique et la tolérance aux pannes. L'objectif du protocole SAR est de minimiser la mesure QoS moyenne pondérée tout au long de la durée de vie du réseau. Le protocole SPEED [138] est un autre protocole de routage basé sur la QoS. Ce protocole exige que chaque nœud conserve des informations sur son voisin. Le protocole de vitesse évite l'encombrement lorsque le réseau est encombré. Le module de routage dans Speed est appelé Forward Stateless Statistical Non-Deterministic Forwarding (SNFG) et fonctionne avec quatre autres modules de la couche réseau.

12.2. Routage géographique

Le routage géographique [139][140] est une technique qui permet de délivrer un paquet à un nœud dans un réseau WSN via plusieurs sauts à l'aide de la position géographique. Dans ce type les décisions de routage ne sont pas basées ni sur les adresses réseau ni sur les tables de routage, au contraire, les messages sont acheminés vers l'emplacement de destination. Avec la connaissance de l'emplacement des voisins, chaque nœud peut sélectionner le voisin qui est plus proche de la destination, et ainsi avancer vers la destination à chaque saut. Le fait que ni les tables de routage, ni les activités de découverte de route ne sont nécessaires rend le routage géographique très attractif pour les réseaux dynamiques, tels que les réseaux de capteurs sans fil.

Pour effectuer un routage géographique dans un réseau ad hoc, il est nécessaire : (i) que tous les nœuds soient muni d'un moyen de localisation comme GPS [141], (ii) qu'un nœud source doit connaître la position du nœud destinataire. Pour ce faire les nœuds peuvent utiliser un service de localisation tels que GLS (Greedy Location Service) [142], QLS (Quorum-based location service) [143] ou encore Homezone [144].

Greedy Perimeter Stateless Routing

Greedy Perimeter Stateless Routing, GPSR [145][146], est un Protocole de Routage Géographique réactif transmet les paquets au voisin le plus proche du noeud cible jusqu'à atteindre la destination (approche Gloutonne ou Greedy Forwarding) . Par conséquent, il s'adapte mieux que les Protocoles de Routage Topologiques aux réseaux à larges échelles dynamiques comme les réseaux ad hoc de capteurs sans fils.

Dans un réseau WSN, les nœuds sont susceptibles de se déplacer. Il est donc nécessaire d'utiliser un mécanisme qui permet à chaque nœud de connaître la position de ses voisins, afin de signaler leur présence et leur localisation, les nœuds inondent le réseau en envoyant un paquet de signalement (messages « beacon ») contenant la position et un identifiant (par exemple, son adresse IP). L'échange périodique de ces paquets permet aux nœuds de construire leur table de position. La période d'émission des messages « beacon » dépend du taux de mobilité dans le réseau ainsi que de la portée radio des nœuds. En effet, lorsqu'un nœud ne reçoit pas de message « beacon » d'un voisin après un temps T, il considère que le voisin en question n'est plus dans sa zone de couverture et l'efface de sa table de position. Un des avantages des messages « beacon » est qu'un nœud n'a pas besoin que des informations sur ses voisins directs, ce qui nécessite peu de mémoire. Alternativement, le protocole GPSR permet au nœud d'encapsuler sur quelques bits leur position dans les paquets de données qu'il envoie, « We encode position as two four-byte floating point quantities, for x and y coordinate values. »[145]. Dans ce cas, toutes les interfaces des noeuds doivent être en mode promiscuité afin de recevoir les paquets s'ils se trouvent dans la zone de couverture de l'émetteur.

L'acheminement des paquets par GPSR se fait selon deux modes suivant la densité du réseau : le « Greedy Forwarding » et le « Perimeter Forwarding » (appelés respectivement GF et PF dans la suite).

Greedy Forwarding

Le GF construit un chemin parcourant les nœuds de la source à la destination où chaque

nœud qui reçoit un paquet l'achemine en faisant un saut vers le nœud intermédiaire le plus proche de la destination dans sa zone de couverture. La figure 2.6 montre un exemple de ce mode d'acheminement.

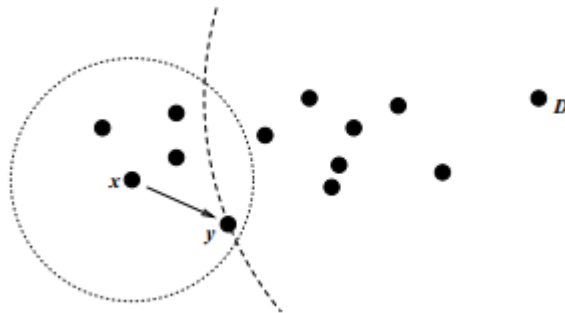


Figure 2.6. y est le voisin de x le plus proche de la destination D

La méthode « Perimeter Forwarding », est utilisé lorsqu' un nœud ne trouve aucun voisin plus proche que lui de la destination ou la destination ne se trouve pas à la portée de celui-ci. La figure 2.7 montre un exemple de ce problème d'acheminement.

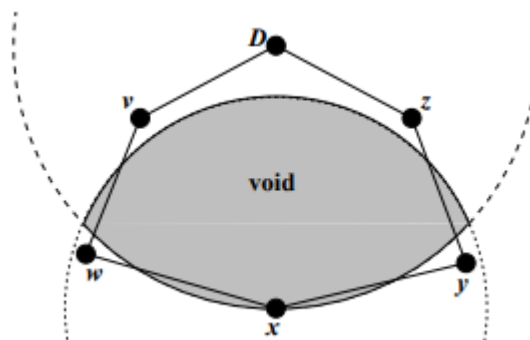


Figure 2.7. X est plus proche de D que ses voisins y, w

Perimeter Forwarding

La méthode « perimeter Forwarding » consiste à transformer la topologie du réseau en un graphe planaire (ne contenant pas des arrêtes qui se croisent). Ce graphe peut être de type RNG (Relative Neighborhood Graph) ou GG (Gabriel Graph) (figure 2.8).

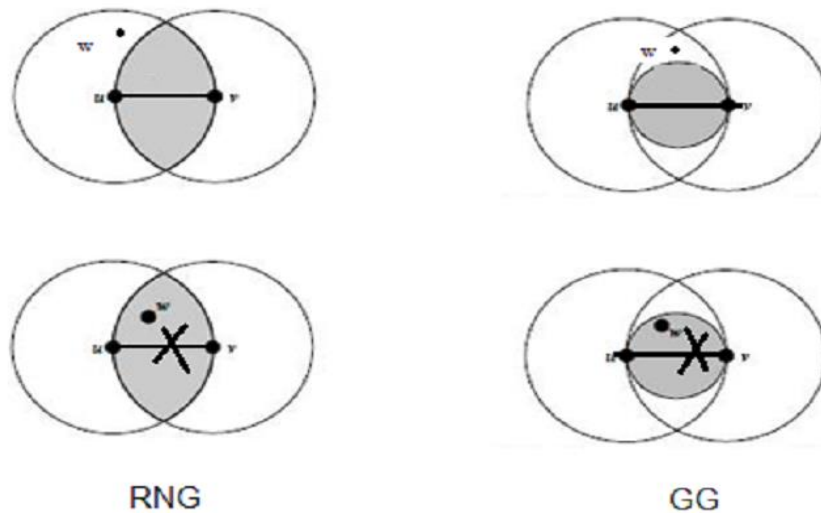


Figure 2.8. Principe des graphes RNG et GG

Les graphes Gabriel graph et Relative Neighborhood graph sont deux cas des graphes de proximité qui visent à représenter la disposition des points d'un ensemble dans l'espace. Dans un tel graphique, deux points sont reliés par une arête s'il n'y a pas d'autres points dans une certaine «zone interdite» définie mathématiquement par :

Relative Neighborhood graph

$$\forall w \neq u, v : d(u, v) \leq \max[d(u, w), d(v, w)] \quad (1)$$

Gabriel graph

$$\forall w \neq u, v : d^2(u, v) < [d^2(u, w), d^2(v, w)] \quad (2)$$

Ensuite le paquet traverse le graphe jusqu'à la destination en utilisant la règle de la main droite « Right-Hand Rule » définie comme suit : Lorsqu'un paquet arrive à un nœud x du nœud y, le chemin à suivre est le prochain qui se trouve dans le sens inverse des aiguilles d'une montre en partant de x et par rapport au segment [xy] tout en évitant les « crossing links » (route déjà parcourue). Pour conclure nous allons décrire le protocole GPSR en combinant entre les deux méthodes de routage: Un paquet GPSR contient dans son en-tête un champ pour le mode de routage. Ce champ contient « Greedy » lorsque le routage est « greedy forwarding » et « Perimeter » lorsque le routage est « Perimeter forwarding ». Un nœud x recevant un paquet en mode « Greedy » examine sa table de voisins. S'il trouve le voisin le plus proche de la destination alors il lui transmet le paquet. Dans le cas contraire, le nœud va modifier le champ mode de l'en-tête du paquet par « Perimeter » et enregistre sa localisation. Ensuite, il construit un graphe planaire à partir de ses voisins et transmet son paquet à travers ce graphe. La figure 2.9 montre un exemple de ce mode d'acheminement.

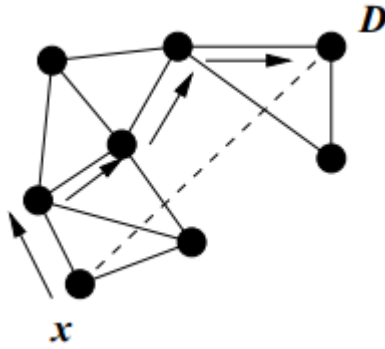


Figure 2.9. Perimeter forwarding

Où D est la destination ; x est le nœud où le paquet entre en mode Perimeter

XIII. Conclusion

Dans ce chapitre on a présenté les concepts généraux liés aux réseaux de capteurs sans fil. Parmi ces concepts nous avons identifié les contraintes de conception des réseaux de capteurs sans fil. Parmi ces contraintes nous avons vu les contraintes matérielles et conceptuelles. Un point sur les protocoles de communication sans fil dont Zigbee a été aussi abordé. La compréhension de ces concepts est nécessaire pour les concepteurs des protocoles et applications destinés aux réseaux de capteurs sans fil.

Description et Etude de la performance de la norme IEEE 802.15.4

I. Introduction

La norme IEEE 802.15.4 a été spécialement définie en fonction des caractéristiques des réseaux de capteurs, un faible débit et une faible consommation. Elle est caractérisée par une portée maximum de quelques centaines de mètres et un débit faible de (250kbit/s max). Elle décrit le fonctionnement de la couche physique et la couche MAC. Plusieurs types de topologies sont supportés par l'IEEE 802.15.4 selon l'usage des dispositifs du réseau : topologie en étoile, topologie point à point et la topologie en arborescence. Le standard IEEE 802.15.4 décrit le fonctionnement de la couche physique et la couche MAC, il fonctionne soit en mode sans balise (non-beacon-enabled) ou avec balise (beacon-enabled). Le mode de fonctionnement est déterminé par un seul contrôleur central appelé PAN coordinateur. Pour la gestion de la communication entre les dispositifs au sein d'un même PAN, le coordinateur utilise des super-frames. Une super-trame est délimitée par deux trames successives, et peut avoir une partie active suivie optionnellement d'une autre partie inactive. La partie active est composée de la période d'accès par contention CAP (Contention Access Period) et la période de libre collision CFP (Contention Free Period). Pour communiquer durant les CAPs chaque nœud doit concurrencer les autres nœuds en utilisant le protocole CSMA/CA slotté.

Dans ce chapitre les différents blocs fonctionnels du modèle de simulation du standard IEEE 802.15.4 sont présentés avec les étapes d'installation de ce standard sous OPNET Modeler.

L'objectif de ce chapitre est d'étudier et déterminer la performance des techniques de modulation BPSK et QPSK. Des enquêtes ont été signalées pour comparer les performances des deux schémas de modulation et trouver le schéma de modulation qui donne une

amélioration significative des performances en fonction de la charge du réseau, la consommation d'énergie et la réception de puissance au niveau des dispositifs WPAN. Les résultats ont été présentés par les trois dispositifs : Pan_Coordinator, GTS et Non GTS (CAP device) du réseau de capteurs sans fil (WSN).

II. Dispositifs du réseau

Un WPAN qu'on appellera par la suite PAN, est composé d'un ensemble de nœuds ayant pour rôle de collecter et de transmettre l'information vers un organe central du réseau, qui sera en charge de traiter ces informations ou de jouer le rôle de passerelle (gateway). Ce dernier a pour nom puits (PAN coordinateur) dans la littérature. Le standard IEEE 802.15.4 définit deux types de nœuds: FFD (Full Fonction Device) et RFD (Reduced Fonction Device) pouvant jouer le rôle de [147]:

- PAN Coordinateur qui est le nœud initiateur du réseau. Il a comme attribution d'initier, de gérer l'ensemble du réseau et de jouer le rôle de passerelle vers l'extérieur.
- Coordinateur qui est par définition un FFD et qui aura pour tâche de gérer une zone donnée du réseau (cluster) et de router les paquets de ses fils vers le PAN coordinateur.
- Nœuds terminaux qui sont de simples nœuds périphériques (actionneurs, capteurs)

On peut définir les FFD et RFD comme suit:

- ✓ **Les FFD (Full-Function Devices)** peuvent communiquer avec des FFD et des RFD, et peuvent ainsi jouer le rôle de coordinateur de PAN (puits), de routeur (Coordinateur) ou de simple périphérique (capteur, interrupteur).
- ✓ **Les RFD (Reduced-Function Devices)** sont des nœuds terminaux utilisés dans des contextes simples comme les interrupteurs ou de simples capteurs. Ces nœuds ne peuvent communiquer qu'avec des FFD (coordinateur) et ne peuvent participer au routage dans le réseau. Ils ne sont ainsi que des nœuds terminaux dans le réseau avec des ressources moindre en comparaison avec des FFD [147].

III. Topologie de réseau

Le standard IEEE 802.15.4 supporte plusieurs types de topologies (figure 3.1): la topologie en étoile (star), la topologie point à point (peer-to-peer) et la topologie en arbre (cluster-tree). Il est possible d'avoir plusieurs réseaux PAN opérant dans la même zone mais

fonctionnant indépendamment les uns des autres, chacun utilisant un identifiant de PAN différent [148].

3.1. Topologie en étoile (star)

La topologie en étoile impose des communications s'établissant directement et uniquement entre le nœud qui gère le réseau défini précédemment (PAN coordinateur) et les nœuds se trouvant dans sa portée radio. Le PAN coordinateur sera le nœud initiateur du réseau et tout le trafic devra passer par ce nœud [147][148].

3.2. Topologie point à point (peer to peer)

Dans la topologie point à point, chaque nœud pourra communiquer avec n'importe quel autre nœud du réseau grâce à la collaboration des nœuds intermédiaires (Coordinateur) sollicités afin de relayer les paquets jusqu'à la destination. Cette topologie permet de réaliser des réseaux beaucoup plus étendus, mais il faudra pour cela ajouter un protocole de routage [147][148].

3.3. Topologie cluster –tree (arborescence)

La topologie cluster-tree n'est qu'un cas particulier de la topologie peer-to-peer. La seule différence avec cette dernière est que le réseau sera hiérarchisé avec un PAN coordinateur, des coordinateurs (têtes de cluster) gérant les nœuds qui leur sont connectés, ainsi que des nœuds terminaux communiquant uniquement avec leur coordinateur [147-148].

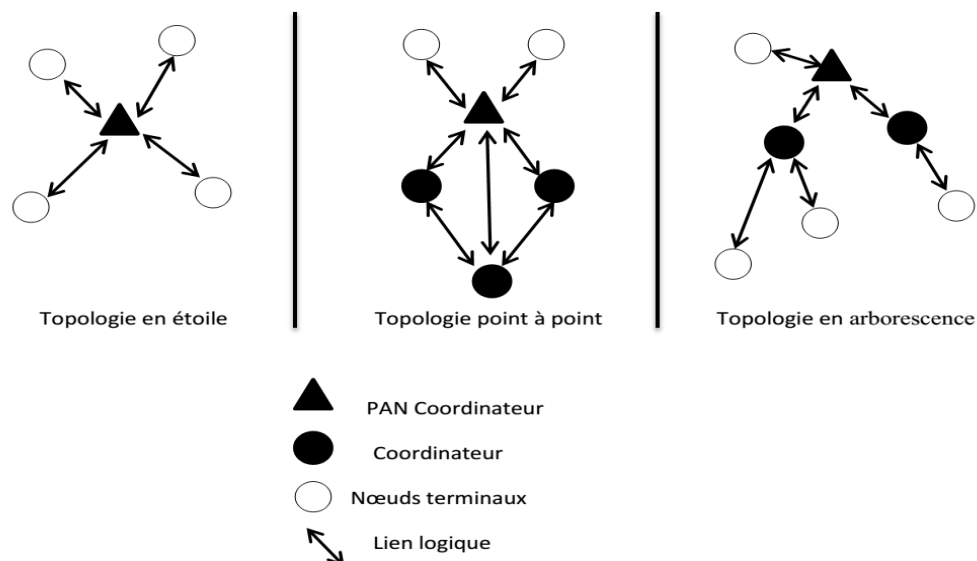


Figure 3.1. Topologies supportées dans IEEE 802.15.4

IV. Principales couches du standard IEEE 802.15.4

4.1. Couche physique

La couche physique (PHY) est la couche initiale du modèle de référence OSI utilisé dans le monde entier. La couche physique (PHY) fournit le service de transmission de données, ainsi que l'interface avec l'entité de gestion de couche physique, qui offre un accès à chaque fonction de gestion de couche et maintient une base de données d'informations sur des réseaux personnels connexes. Ainsi, cette couche gère l'émetteur-récepteur radio (RF) physique et effectue des fonctions de sélection de canal et de gestion de l'énergie et du signal [149][150].

La couche physique a pour rôle de gérer le support physique sur lequel seront faites les transmissions. Elle définit les techniques par lesquelles les bits seront transformés en signaux analogiques et inversement.

La couche physique dans IEEE 802.15.4 offre les fonctionnalités suivantes :

- Gestion de l'activation et de la désactivation du module radio, la radio pourra prendre trois états différents: réception, émission et éteinte.
- La détection de la puissance du signal sur le canal (Energy Detection (ED))
- La possibilité de remonter à la couche supérieure des informations sur la qualité du lien radio après réception d'une trame : LQI (Link Quality Indication)
- La détection de l'occupation ou non du médium : CCA (Clear Channel Assessment)
- La sélection d'un canal de transmission parmi les différents canaux disponibles

Le standard IEEE 802.15.4 [151] propose quatre différentes couches physiques exploitant chacune des techniques de modulation, d'étalement de spectre ou des bandes de fréquences différentes comme présenté ci-dessous :

- 868/915 MHz utilisant Direct Sequence Spread Spectrum (DSSS) comme technique d'étalement de spectre et la modulation Binary Phase-Shift Keying (BPSK)
- 868/915 MHz utilisant aussi la technique d'étalement de spectre DSSS mais combinée avec une modulation Offset Quadrature Phase-Shift Keying (O-QPSK)
- 868/915 MHz utilisant l'étalement de spectre Parallel Sequence Spread Spectrum (PSSS) et combinant les deux techniques de modulation BPSK et Amplitude Shift Keying (ASK)

- 2450 MHz utilisant DSSS pour l'étalement de spectre et l'O-QPSK comme modulation.

Le standard offre au total 27 canaux de communication. Selon le choix de la bande de fréquence et de la couche physique on aura des débits allant de 20 à 250 kbit/s (T 3.1) [147].

Bande de fréquence	Chiprate(kchip/s)	Modulation	Kbit/s	Ksymboles/s	Symboles
868-868.6	300	BPSK		20	Binary
902-928	600	BPSK		40	Binary
868-868.6	400	ASK		12.50	5-bit PSSS
902-928	1600	ASK		50	20-bit PSSS
868-868.6	400	Q-QPSK		25	16-ary Orthogonal
902-928	1000	Q-QPSK		62.5	16-ary Orthogonal
2400-2483.5	2000	Q-QPSK		62.5	16-ary Orthogonal

Tableau 3.1 : Récapitulatif de la couche physique dans IEEE 802.15.4

4.2. Couche d'accès au médium (MAC)

La couche MAC (Medium Acces Control), comme son nom l'indique, aura pour rôle de gérer l'accès au canal avec le mécanisme CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance). Cette couche offrira un mécanisme d'économie d'énergie, une des fonctionnalités les plus importantes dans les réseaux de capteurs sans fil. Nous pouvons résumer ce mécanisme d'une manière très simple: l'utilisation de la radio représentant le plus grand facteur de dépense énergétique doit être limité au strict minimum en la gardant éteinte la majorité du temps [147].

La couche MAC offre aussi d'autres fonctionnalités de contrôle liées à [149][150]:

- La gestion du réseau
- La gestion des acquittements
- La gestion de l'adressage (adresse IEEE 802.15.4 codé sur 64 et 16 bits)
- L'allocation et la gestion des Slots dédiés

- Le mécanisme de découverte de voisinage

4.2.1. Modes d'accès au support

Le standard IEEE 802.15.4-2006 propose deux modes pour l'accès au support (Voir Figure 3.2) :

a. Mode avec synchronisation ou balisé "beacon enabled"

Ce mode est utilisé pour les réseaux à contrainte de synchronisation et à faible délai de latence. Selon ce mode, le coordinateur envoie périodiquement des trames balises pour synchroniser les nœuds du réseau. Tout membre du réseau qui entend cette balise peut ainsi se synchroniser et se servir de ce coordinateur comme relais. L'espace temporel entre deux trames balises consécutives est appelé super-trame. Le choix de ce mode impose l'utilisation de la super trame pour la gestion de la communication entre les dispositifs au sein d'un même PAN [152].

b. Mode sans synchronisation ou non balisé "non-beacon enabled"

Ce mode n'utilise ni de trames balises ni de super-trames. A l'exception des trames d'acquiescement et certaines trames de données transmises immédiatement après un acquiescement, tous les messages sont transmis selon le mécanisme CSMA/CA non discrétisé (slotté) [152].

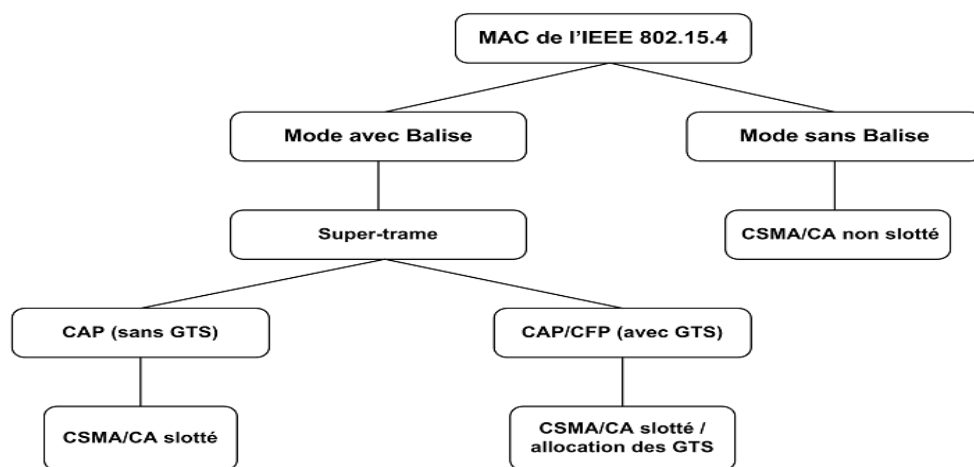


Figure 3.2. Modes d'exécution de la sous couche MAC de la norme IEEE 802.15.4

4.2.2. Structure d'une super-trame IEEE 802.15.4

Une super-trame est composée d'une partie active où les nœuds peuvent communiquer entre eux et optionnellement suivie d'une autre partie inactive où les nœuds se mettent en mode veille. Le paramètre, désigné par le terme BI intervalle (Beacon Interval), permet de fixer la durée des intervalles des balises. Le paramètre désigné par le terme SD (Super-frame Duration) définit la longueur de la portion active de la super-trame. Les durées de l'intervalle inter-balise BI et de la super-trame SD sont déterminées respectivement par l'ordre de balise BO (Beacon Order) (Equation.1) et l'ordre de la super-trame SO (Super-frame Order) (Equation.2) [152]

$$\frac{BI = aBaseSuperFrameDuration * 2^{BO} [symboles]}{0 \leq BO \leq 14} \quad (1)$$

$$\frac{SD = aBaseSuperFrameDuration * 2^{SO} [symboles]}{0 \leq SO \leq BO \leq 14} \quad (2)$$

Comme illustrée par la Figure 3.3, la partie active de la super-trame est divisée en 16 créneaux de temps de durées égales où le premier créneau de temps est occupé par la trame balise servant à la synchronisation du réseau. Cette partie active est composée de trois sous parties : La balise, la période d'accès par contention CAP (Contention Access Period) et la période de libre collision CFP (Contention Free Period). L'accès au médium durant la période CAP se fait selon le protocole CSMA/CA slotté. Dans le cas où la super-trame inclut une période CFP, le coordinateur garantit l'accès au médium à certains nœuds. Pour cela, la période CFP est divisée en plusieurs parties composées de créneaux de temps garantis ou GTS (Guaranteed time slots). Notons que les demandes de réservation des GTS ainsi que les demandes d'association au réseau ne peuvent se faire que durant la période CAP. Chaque GTS consiste en un multiple entier de créneaux de temps CFP et il peut y avoir jusqu'à 7 GTS dans une CFP [152][153].

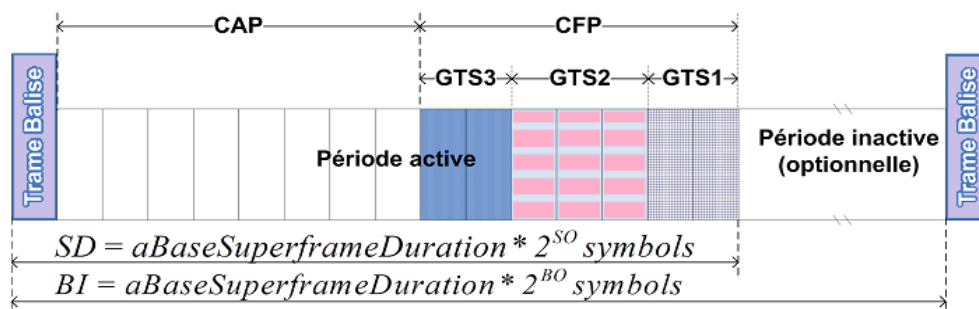


Figure 3.3. Structure de la super-trame IEEE 802.15.4

V. Modèle de simulation d'IEEE 802.15.4

Dans ce travail nous avons utilisé OPNET qui est un environnement de simulation et modélisation de réseau à évènements discrets.

Le modèle de simulation implémente la Couche physique et la Couche d'accès au médium(MAC) définies par le standard IEEE 802.15.4.OPNET Modeler est utilisé pour le développement, notamment en raison de sa précision et son interface utilisateur graphique sophistiquée. OPNET est un environnement de simulation et modélisation de réseau à évènements discrets [154].

La structure de la norme IEEE 802.15.4, nœuds capteurs (wpan_sensor_node) utilisés sans le modèle de simulation est composé de quatre blocs fonctionnels :

Couche physique : se compose d'un émetteur radio sans fil (tx) et un récepteur (rx) conforme au spécification de la norme IEEE 802.15.4, fonctionnant avec une bande de fréquence de 2,4 GHz et un taux de données de 250 Kbits/s. La puissance d'émission est de 1mW et la technique de modulation utilisée est la Modulation en quadrature de phase (Quadrature Phase Shift Keying (QPSK)) [154][155].

Couche MAC : implémente le CSMA/CA à fentes et les mécanismes de la GTS. Le trafic de données GTS (trafic temporellement critiques (time-critical traffic)), provenant de la couche application est stocké dans une mémoire tampon avec une capacité spécifiée et distribuée sur le réseau lorsque le GTS correspondant est actif. Cette couche est également responsable de générer des trames avec envoi de balise et la synchronisation du réseau lorsqu'un nœud donné agit comme coordonnateur de PAN [154][155].

Couche Application : comprend deux générateurs de trafic de données (Le trafic source et le trafic source GTS) et un trafic de réception. Le trafic source génère des trames de données non reconnue et reconnue transmises au cours du CAP (utilise la CSMA/CA à fentes).Le trafic source GTS peut produire des trames de données temporellement critiques reconnues ou non reconnues en utilisant le mécanisme de la GTS. Le module de trafic de réception reçoit les trames transmises de couches inférieures et effectue les statistiques du réseau [154][155].

Module de batterie : Calcule les niveaux d'énergie consommés et restants. Les valeurs par défaut des consommations énergétiques sont réglées pour ceux des spécifications de Modèle MICAz.[154][155].

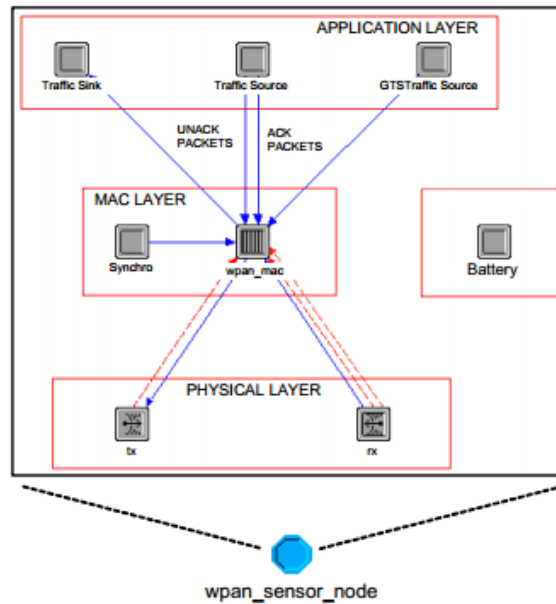


Figure 3.4. Structure du modèle de simulation d'IEEE 802.15.4 [7]

VI. Etapes d'installation sous OPNET Modeler

Pour installer et exécuter ce modèle de simulation sur votre ordinateur, il faut suivre les étapes suivantes (pour la version de simulation 2.1 et OPNET Modeler version 14,5) [156][157] :

1. Installer la bonne version d'OPNET Modeler Wireless Suite et configurer l'environnement.
2. Télécharger le fichier de modèle de simulation et décompressez-le dans un répertoire défini par l'utilisateur.
3. Ajouter ce répertoire dans les répertoires du modèle d'OPNET. File > Manage Model Files > Add Model Directory (Figure 3.5). Le nom du répertoire est automatiquement ajouté dans le fichier de base de données d'environnement : Op_admin\env_db14.5
4. Après cela, le nom du répertoire est trouvé dans la boîte de dialogue : Open file dialog : File > Open. Sélectionnez le répertoire du modèle approprié et le fichier puis appuyer sur ouvrir.

Le scénario par défaut contient un seul nœud analyseur (wpan_analyzer_node) et trois nœuds de capteurs (wpan_sensor_node) (Figure 3.6) organisés dans une topologie en étoile. Le PAN_coordinator est l'en-tête de l'étoile qui contrôle le réseau (par exemple activer et désactiver le mécanisme de la GTS définissant des BO, SO et les paramètres ID du PAN pour l'ensemble du réseau). Node_GTS génère uniquement le temps critique du trafic envoyé durant la GTS. De l'autre côté, node_CAP génère un trafic non reconnu et reconnu

envoyé durant le CAP.Des nœuds supplémentaires peuvent être ajoutés au référentiel de modèle : Topology > Open Object Palette > Node Models > Fixed Node Models > By Name > wpan. (Figure 3.7).

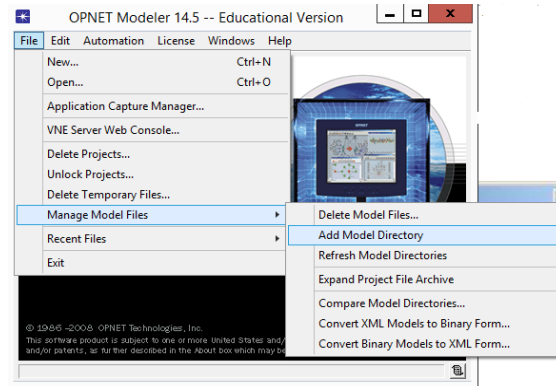


Figure 3.5.Ajout du répertoire dans les répertoires du modèle d’OPNET

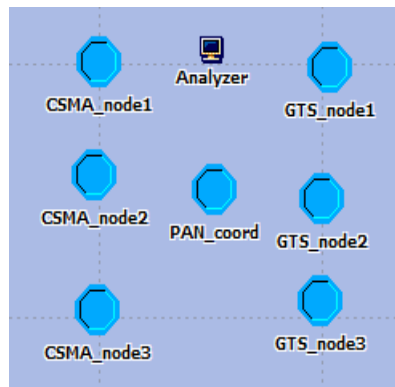


Figure 3.6.Scénario par défaut

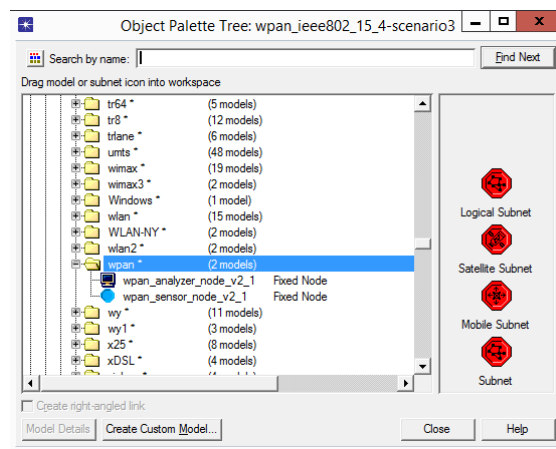


Figure 3.7.Palette de wpan_ IEEE 802.15.4

VII. Description du système

Le modèle de simulation met en œuvre la couche d'accès physiques et la couche MAC définies dans la norme IEEE 802.15.4. OPNET Modeler 14.5 est utilisé pour développer la norme 802.15.4 du réseau de capteurs sans fil.

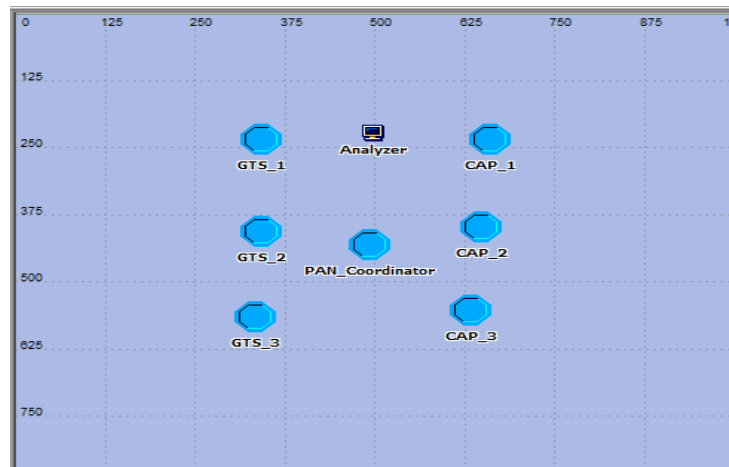


Figure 3.8.Scénario de réseau pour BPSK et QPSK

La figure 3.8 montre le scénario de QPSK et BPSK qui contient un coordinateur de PAN, un analyseur et six nœuds terminaux dans lesquels trois sont des dispositifs de créneaux de temps garantis (GTS) et trois sont des dispositifs non GTS.

Les deux scénarios différents ont été créés avec deux types de modulation : BPSK et QPSK. Les paramètres indiqués dans le tableau 3.2 ont été fixés pour ces scénarios.

VIII. Résultat

La simulation a été effectuée pour les deux scénarios différents de WPAN: La modulation en quadrature de phase QPSK et la modulation binaire t de phase BPSK. Dans cette section, les résultats pour la consommation d'énergie par la batterie et la puissance de réception du récepteur radio ont été présentés et discutées pour les différents types de dispositifs dans les réseaux de capteurs sans fil.

8.1. Charge de sortie du réseau (Network Output Load)

8.1.1. Coordinateur du PAN (PAN_Coordinator)

La figure 3.9 indique que la valeur moyenne de la charge de sortie du réseau pour le PAN_Coordinator est de 456, 19 863 bits /s et 451,559322bits /s pour QPSK et BPSK

respectivement, et après 19, 8 secondes, la valeur diminue à 51, 25 bits /se QPSK et 38, 05 bits /s pour BPSK.

Paramètres			PAN Coordinateur	GTS Enabled End Device	Non GTS End Device(CAP)
Modulation			QPSK/BPSK	QPSK/BPSK	QPSK/BPSK
Traffic source	Acknowledged Traffic Source	Destination MAC Address	Broadcast		
		MSDU Interarrival Time(sec)	Exponentiel(1.0)	Constant(1.0)	Exponentiel(1.0)
		MSDU Size(bits)	Exponentiel(100)	Constant(0.0)	Exponentiel(100)
		Start time(sec)	0.0	Infinity	1.0
		Stop time(sec)	Infinity		
	Unacknowledged Traffic Source	MSDU Interarrival Time(sec)	Exponentiel(1.0)	Constant(0.0)	Exponentiel(1.0)
		MSDU Size(bits)	Exponentiel(100)	Constant(0.0)	Exponentiel(100)
		Start time(sec)	0.1	Infinity	1.1
Stop time(sec)		Infinity			
CSMA/CA Parameters	Maximum Back- off Number	4			
	Minimum Back- off Exponent	3			
IEEE 802.15.4	Device node		PAN Coordinator	End Device	
	MAC Address		Auto Assigned		
	WPAN Settings	Beacon Order	12	9	
		Superframe Order	8		
PAN ID		0			
Logging	Enable Logging		Enabled		
GTS	GTS Settings	GTS Permit	Enabled		
		Start Time	0.0	0.1	Infinity
		Stop Time	Infinity		
		Lenght(slots)	1	0	
		Direction	Receive	Transmit	
		Buffer Capacity (bits)	1000	1000	
	GTS Traffic Parameters	MSDU Interarrival Time(sec)	Exponentiel(1.0)	Constant(1.0)	
		MSDU Size(bits)	Exponentiel(100)	Constant(0.0)	
Acknowledgement		Enabled	Disabled		

Tableau 3.2. Comparaison selon les modulations BPSK et QPSK

On observe que la charge de sortie du réseau est plus élevée en cas de QPSK, car en utilisant cette modulation nous pouvons atteindre un rapport signal sur bruit SNR (Signal to

Noise Ratio) élevé ainsi nous pouvons être sûr d’avoir moins de bruit dans le canal pendant la transmission. En outre, il est observé que la charge est minimale en cas du BPSK en raison de son très faible SNR qui fournit une haute immunité au bruit en comparant le signal flottant dans le moyen. Aussi BPSK et QPSK ont un taux d’erreur similaire (BER), mais la modulation BPSK est capable de moduler seulement 1 bit/s, mais en utilisant QPSK, nous pouvons transmettre deux fois plus des données sur le même canal avec la même bande passante.

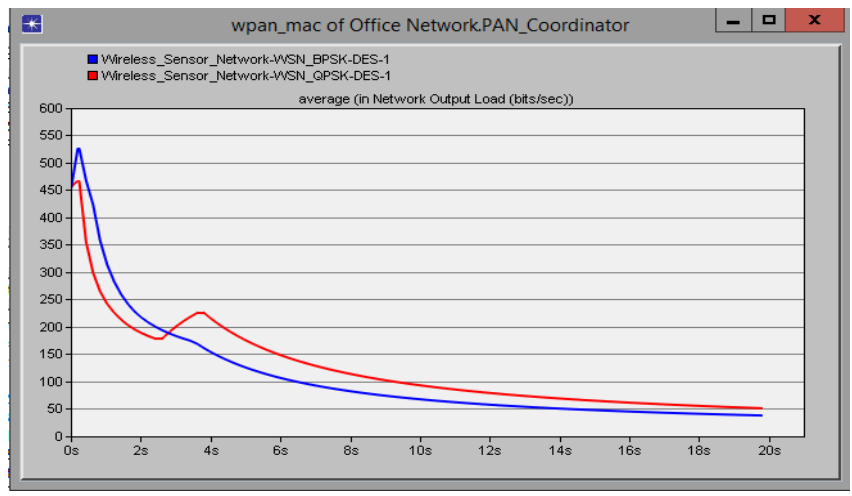


Figure 3.9. Charge de sortie de réseau pour le PAN_Coordinator

8.1.2. GTS End Device

La figure 3.10 montre que la charge de sortie de réseau pour le Dispositif GTS est: 602,789396bits /sec et 602,654867 bits /sec pour QPSK et BPSK respectivement au début de la simulation et après 19, 8 secondes la valeur diminue de 12 bits /sec pour QPSK et BPSK.

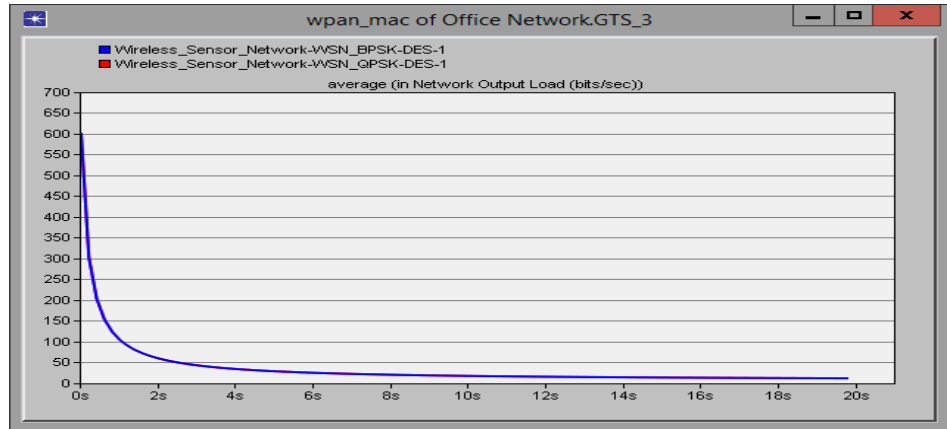


Figure 3.10. Charge de sortie de réseau pour le GTS End Device

8.1.3. CAP End Device (Non GTS End Device)

La figure 3.11 montre que la charge de sortie de réseau pour le dispositif non GTS: 328,634921 bits /s et 234,698158 bits /s pour QPSK et BPSK respectivement au début de la simulation et après 19, 8 secondes, la valeur décroît à 124,45 bits /s pour QPSK et 62,65bits /s pour BPSK.

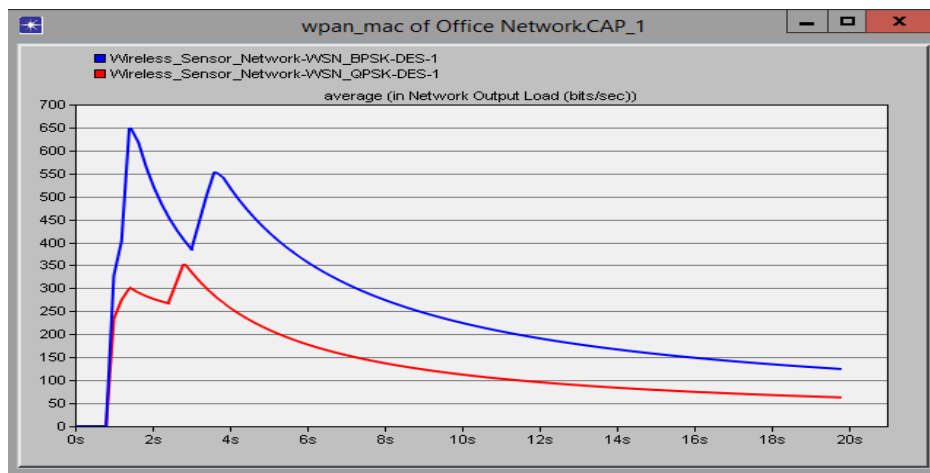


Figure 3.11. Charge de sortie de réseau pour le Non GTS End Device

8.2. Energie consommée

8.2.1. PAN Coordinateur

La Figure 3.12 ci-dessous indique que la valeur moyenne de l'énergie consommée par le coordinateur de PAN est 0.00147992 et 0.00166499 joules pour la QPSK et BPSK respectivement. On constate que la consommation d'énergie de la batterie est minimum en cas de QPSK en raison de son enveloppe constante qui permet d'utiliser des amplificateurs à leur meilleure zone de performance qui correspond souvent à un fonctionnement non linéaire, aussi la QPSK utilise quatre points sur le diagramme de constellation équidistants autour d'un cercle avec quatre phases qui est la raison pour laquelle l'énergie se propage sur un intervalle borné de quatre états (phases) .La consommation d'énergie avec la modulation BPSK dépend de la position du point de constellation.

8.2.2. GTS End Device

La figure 3.13 représente l'énergie consommée par le dispositif GTS. Il a été observé qu'elle est de 0.00129715 et 0.00150752 joules pour la QPSK et la BPSK respectivement. Il a été prouvé expérimentalement que l'énergie consommée est minimale en cas de QPSK alors qu'il est maximal en cas de BPSK.

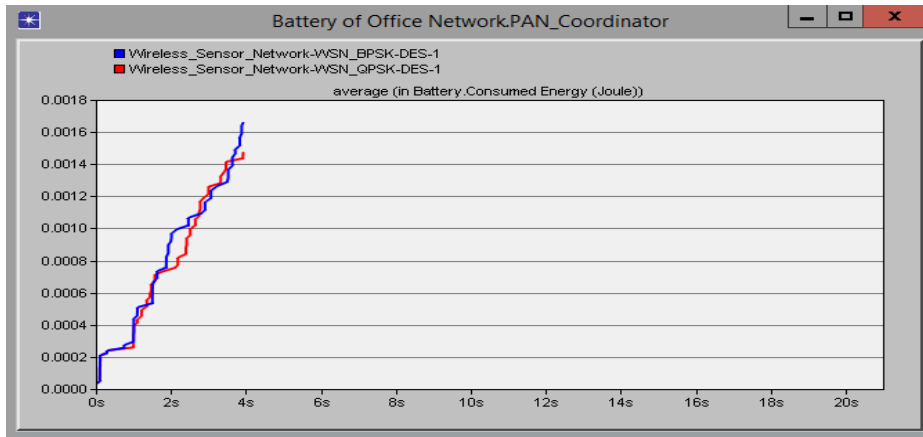


Figure 3.12. Energie Consumé par le PAN_Coordinator

8.2.3. CAP End Device

La figure 3.14 montre que l'énergie consommée par batterie pour le dispositif Non GTS est 0.00145471 et 0.00177974 joules pour la QPSK et la BPSK respectivement. On observe que l'énergie consommée est minimum pour la QPSK en comparaison avec celle du BPSK.

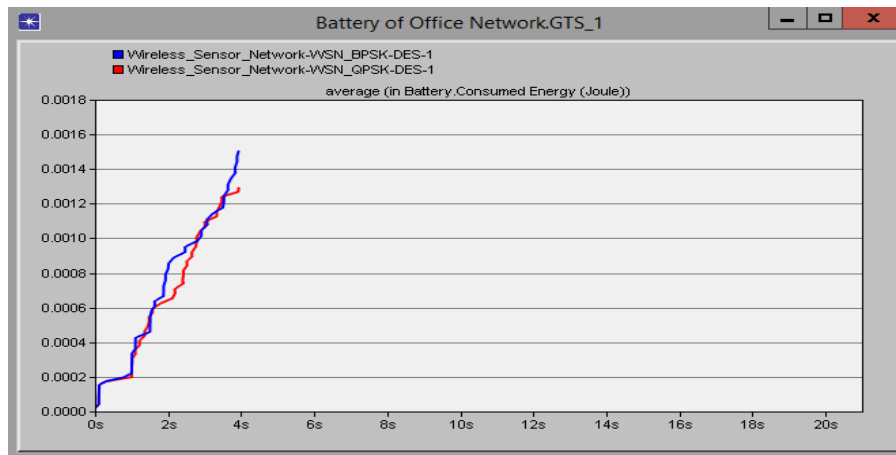


Figure 3.13. Energie Consumé par le GTS Device

8.3. Puissance reçue par le récepteur radio

8.3.1. PAN_Coordinator

La Figure 3.15 ci-dessous montre que la puissance reçue par le récepteur radio au coordinateur de PAN est 0.00023363 et 0.00023844 watts pour BPSK et QPSK respectivement. Ici on constate que la puissance reçue est à peu près la même dans le cas

de BPSK et QPSK parce que les deux schémas de modulation reçoivent de l'énergie selon $\frac{A^2}{2}$ où A est l'amplitude.

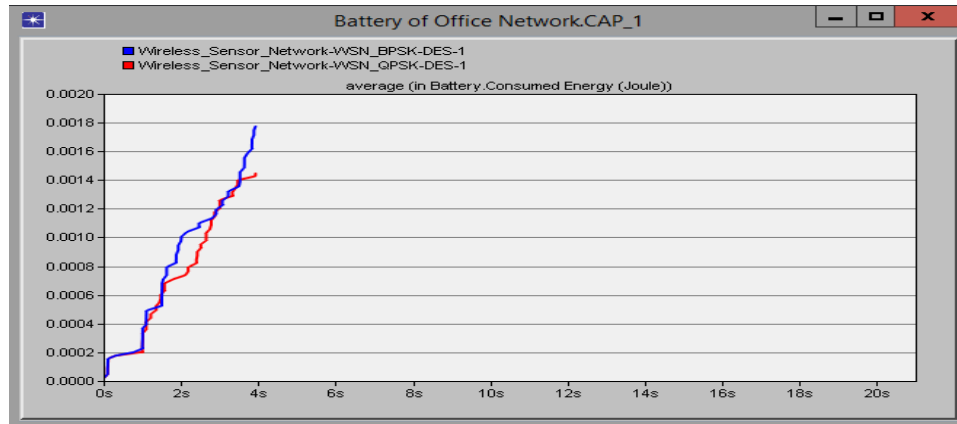


Figure 3.14. Energie Consumé par le Non GTS End Device

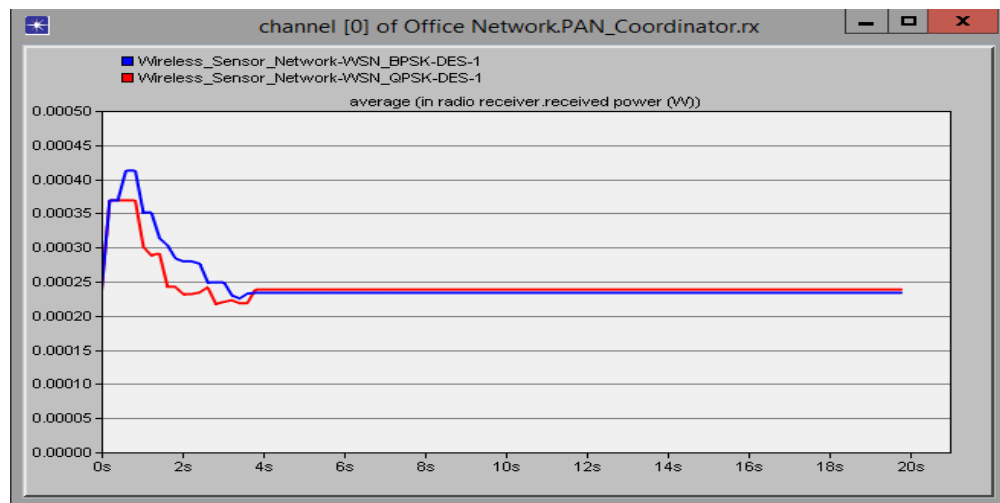


Figure 3.15. Puissance reçue par le PAN Coordinator

8.3.2. GTS End Device

La Figure 3.16 indique que la puissance reçue par le dispositif GTS est $2.9762 \cdot 10^{-6}$ et $3.4014 \cdot 10^{-6}$ watts pour BPSK et QPSK respectivement.

8.3.3. CAP End Device

La Figure 3.17 indique la puissance reçue par le dispositif Non GTS .Il a été observé qu'elle est de $9.2828 \cdot 10^{-5}$ et $7.594 \cdot 10^{-5}$ W pour BPSK et QPSK respectivement. Il a été étudié que la puissance reçue est maximale en cas de BPSK alors qu'elle est minimale en cas de QPSK.

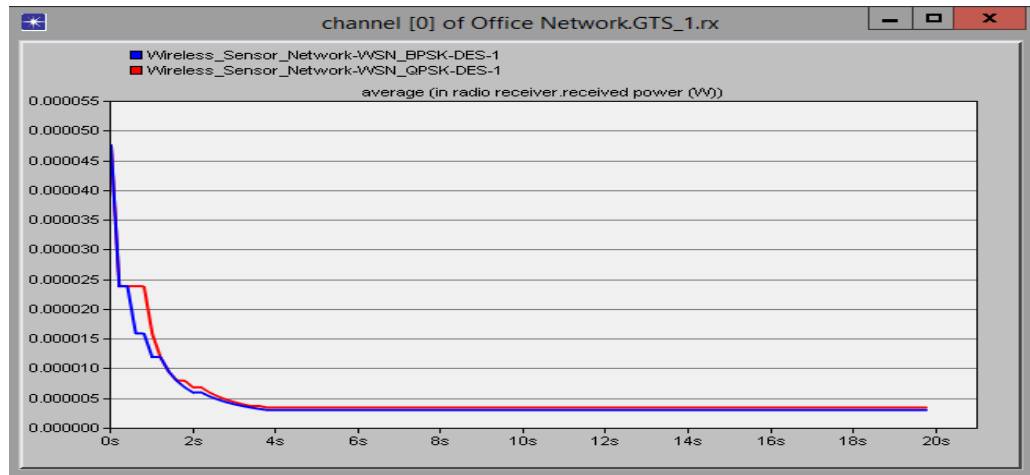


Figure 3.16. Puissance reçue par le GTS End Device

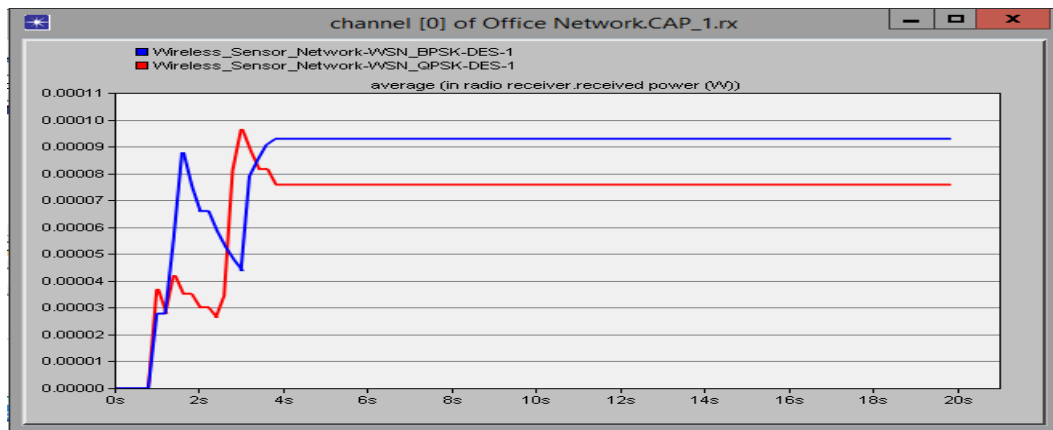


Figure 3.17. Puissance reçue par le Non GTS End Device

IX. Conclusion

Ce chapitre a permis de donner un aperçu détaillé sur le standard IEEE 802.15.4 largement utilisé et adapté aux applications des réseaux de capteurs sans fil, en présentant ses principaux dispositifs (FFD et RFD) fonctionnant selon la topologie adaptée (étoile, point à point ou arborescence). Ce chapitre a permis aussi de décrire les deux principales couches du standard IEEE 802.15.4 : la couche physique et la couche MAC en expliquant les deux modes de fonctionnement de cette dernière (avec et sans balise). Pour installer ce standard sous OPNET Modeler, on a présenté le modèle de simulation avec ses différents blocs fonctionnels ainsi que les différentes étapes d'installation de ce standard qui permettent de travailler avec la palette du « wpan » et réaliser plusieurs simulations pour les réseaux des capteurs sans fil WSNs, domaine en pleine expansion en raison de la multiplicité des applications auxquelles il peut conduire.

Localisation dans les réseaux de capteur sans fil

I. Introduction

Depuis quelques années, la localisation est devenue de plus en plus importante dans la vie quotidienne. Dans un premier temps, l'utilisation des systèmes de navigation par satellite (GPS), permettant une localisation relativement précise en environnements extérieurs dégagés avec une précision de l'ordre de quelques mètres), a connu une réelle expansion. Plus récemment, de nombreuses applications émergentes reposent sur la possibilité de poursuivre des dispositifs mobiles dans des environnements où les méthodes de navigation traditionnelles ne sont plus opérantes [158]. Avoir sa position exacte n'importe où sur la surface de la terre est rendu possible grâce aux constellations satellitaires de la géolocalisation. Il s'agit d'un ensemble de satellites placés de façon précise en orbite MEO (Medium Earth Orbit). Nous présentons dans ce qui suit le principe de localisation utilisé dans des systèmes tels que GPS et Galileo, puis nous abordons la problématique de la localisation dans les réseaux de capteurs sans fil (RCSF).

II. Systèmes de localisation

Si autrefois les systèmes de localisation étaient exclusivement réservés aux applications militaires, ils se sont ouverts depuis les années 90 au monde civil. Les grandes puissances ont toutes cernées les impacts politiques et économiques de ces systèmes [158].

2.1. Les principaux systèmes de localisation

- A. **Le GPS : (Global Positioning System)** est le système de localisation américain. Opérationnel depuis les années 1980, il a été développé pour fournir à l'armée américaine un système de repérage à couverture mondiale et de très grande précision. Son rôle consiste, par exemple, à guider un missile sur des centaines de kilomètres. Son nom officiel est NAVSTAR system (Navigation Satellite Timing

and Ranging). C'est à la fin de l'année 1993 que le département américain de la défense a ouvert l'accès gratuit au GPS pour les utilisateurs civils [158].

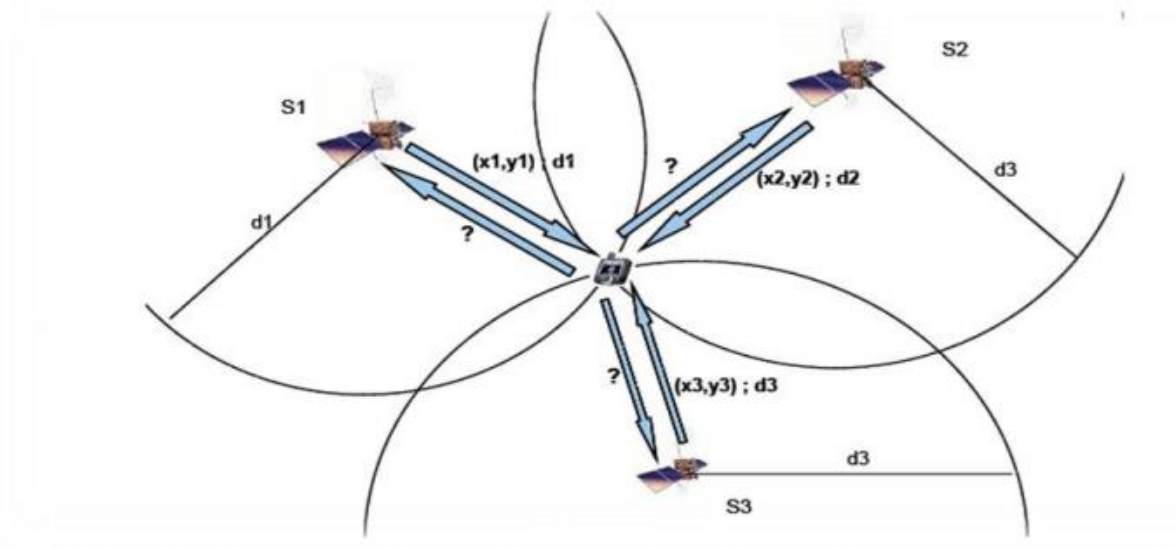


Figure 4.1.Principe des méthodes de localisation par satellite (en 2D)

- B. **Le système russe GLONASS (IAC)** : développé de 1976 à 1982, n'est plus pleinement opérationnel et ce dû aux conditions politiques et économiques du pays. Toutefois, la Russie a entrepris sa remise à niveau et GLONASS est désormais de nouveau fonctionnel depuis 2011.
- C. **Le système IRNSS (Indian Regional Navigational Satellite System)** : De son côté, l'Inde met en œuvre son système IRNSS. Il offrira une précision au sol inférieur à 20 mètres et il est devenu près depuis 2012.
- D. **Le système Beidou** : développé par la Chine en est à une version expérimentale (Beidou-1). Elle est composée de quatre satellites ayant des fonctionnalités limitées. La version Compass (ou Beidou-2) a comptée 35 satellites opérationnels depuis 2011.
- E. **Le système Galileo** : Enfin, Galileo (CNES et ESA) est le système de localisation européen. Il est composé de 27 satellites et atteindra une précision inférieure au mètre pour les applications du domaine civil [158].

2.2. Principe de localisation GPS

Le GPS est un système de positionnement global par satellite. Le principe de localisation est en lui-même très simple. Il suffit d'entrer en contact avec des satellites qui lui

communiquent leurs coordonnées. Au moins trois satellites sont nécessaires pour une localisation dans la deuxième dimension, alors qu'au moins quatre le sont pour la troisième dimension. Chaque satellite du réseau envoie un signal indiquant sa position et l'heure de la transmission du signal. Comme les signaux se déplacent à la vitesse de la lumière, le récepteur peut déterminer la distance radiale du satellite par rapport au récepteur.

Le boîtier récepteur, que possède le particulier, procède par mesures de distances; de ce point de vue, le GPS travaille en régime sphérique.

On peut construire une sphère de rayon D_1 autour du satellite S_1 qui contient l'emplacement désiré M . L'intersection avec le globe donne un premier cercle C_1 . En procédant de la même manière avec le satellite S_2 , on peut construire une sphère de rayon D_2 autour du satellite S_2 qui contient le point M . L'intersection avec le globe donne un deuxième cercle C_2 . Les cercles C_1 et C_2 se coupent donc en 2 points. On peut construire enfin une sphère de rayon D_3 autour du satellite S_3 qui contient le point M , l'intersection avec le globe donne un troisième cercle C_3 . C'est le troisième satellite "qui lève l'indétermination" et précise de manière unique le point M cherché (Figure 4.2).

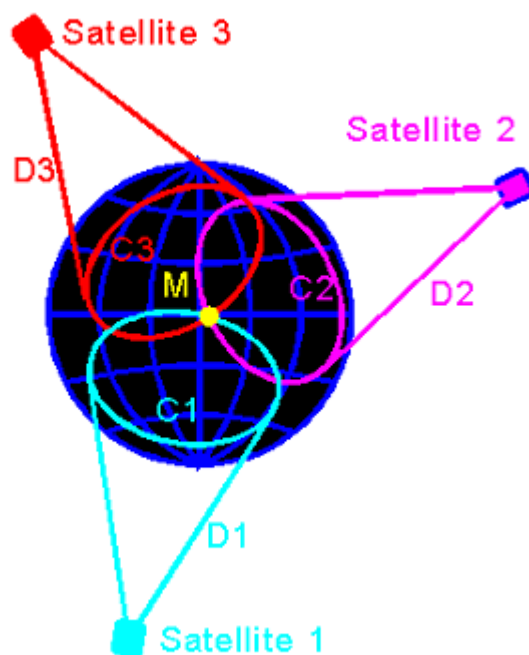


Figure 4.2. Positionnement simple sur le globe

Si certaines méthodes de localisation s'interdisent d'utiliser ces technologies à cause de leurs erreurs de mesure, il est indispensable, pour celles qui les utilisent, de considérer ces erreurs lors du calcul des positions [158].

III. Localisation dans les Réseaux de Capteurs Sans fil

Dans bon nombre d'applications, un événement détecté par un capteur n'est utile que si une information relative à sa localisation géographique est fournie. C'est le cas de la surveillance des feux de forêt ou de troupes ennemies dans un contexte militaire. Sans cette information, ces applications n'auraient aucun sens. Il s'agit donc de déterminer pour chacun des capteurs sa position. La localisation des capteurs est un des principaux problèmes dans ce type de réseaux et nombreuses sont les solutions qui ont été proposées pour le résoudre, chacune faisant des hypothèses diverses sur les capacités des capteurs. Les méthodes de localisation étudiées dans ce travail ont pour but d'estimer ces positions de manière automatique. Une méthode de localisation dans les réseaux de capteurs est composée de deux parties : estimation des distances et dérivation des positions.

3.1. Estimation des distances

Dans cette phase les nœuds communiquent entre eux et collectent différents indicateurs de qualité des communications radios. Le hardware radio peut rapporter diverses informations sur le signal radio entre deux nœuds. En effet le simple fait qu'ils communiquent entre eux nous indique qu'ils sont à portée radio l'un de l'autre. De plus le hardware radio de nos nœuds peut nous rapporter diverses caractéristiques à propos du signal radio entre les deux nœuds, à partir desquelles les distances séparant les nœuds peuvent être estimées.

3.2. Dérivation des positions

Le but de cette phase est de trouver les positions des nœuds qui respectent au mieux les distances inter-nœuds estimées. Si nous connaissons la position de quelques nœuds du réseau dans un certain système de coordonnées, les positions des autres nœuds dans ce système de coordonnées peuvent être trouvées.

IV. Caractérisations des méthodes

Il existe de nombreuses approches pour résoudre le problème de la localisation et chaque méthode a ses avantages et ses inconvénients.

4.1. Utilisation d'estimations de distances

4.1.1. Méthodes range-free (Les méthodes libres de mesure)

Ces méthodes ne calculent jamais de distances entre voisins. Elles utilisent d'autres informations telles que la connectivité pour identifier la position des nœuds. Dans un objectif de simplicité et de réduction du coût, ces méthodes supposent que le déploiement des nœuds respecte certaines contraintes et propose des calculs plus ou moins complexes pour évaluer la position. Elles semblent donner de bons résultats dans les réseaux denses et réguliers. Dans ce mémoire nous nous intéresserons aux méthodes range-free.

4.1.2. Méthodes range-based (Les méthodes basées mesures)

Ces méthodes estiment les distances entre les nœuds en utilisant une mesure de la distance inter-nœud obtenue grâce au signal radio, et ensuite dérivent de ces distances les positions des nœuds.

4.2. Nécessité de connaître la position d'ancres

Si une méthode requiert l'encodage au préalable de la position d'un certain nombre d'ancres, cela signifie qu'il faudra qu'une personne intervienne avant un déploiement pour mesurer la position d'un certain nombre de nœuds. Cela est parfois difficile, voire impossible dans certaines situations. Et donc, le fait que la méthode de localisation requiert ou non de connaître la position d'un certain nombre d'ancres est une caractéristique importante de la méthode.

4.2.1. Méthodes anchor-based

Sont celles qui ne fonctionnent pas sans connaître la position d'un certain nombre d'ancres à priori.

4.2.2. Méthodes anchor-free

Sont celles qui n'ont besoin de la position d'aucun nœud pour fonctionner ; elles créent donc une carte relative du réseau. Par relative, nous entendons une carte qui est à une translation, une rotation orthogonale, une réflexion et une dilatation près de la 'vraie' carte. Autrement dit, c'est une carte qui conserve les rapports entre les distances entre tous les points.

4.3. Forme d'implémentation

Nous distinguons plusieurs façons d'implémenter le processus de localisation :

4.3.1. Méthodes centralisées

Tous les nœuds communiquent avec leurs voisins et renvoient à l'ordinateur central soit des informations sur le signal, soit directement les distances. L'ordinateur central s'occupe si nécessaire d'estimer la distance à partir des informations sur le signal et ensuite de localiser les nœuds [159][160].

4.3.2. Méthodes distribuées

Ici tous les nœuds communiquent avec leurs voisins pour estimer les distances et échangent leurs informations de voisinage. Ils dérivent ensuite de façon distribuée la position de tous les nœuds dans le réseau. C'est-à-dire qu'à la fin du processus de localisation, chaque nœud doit connaître sa position ainsi que celles de ses voisins et ce sans l'aide d'un ordinateur central qui effectuerait les calculs. Pour les grands réseaux, on considère qu'une méthode distribuée est nécessaire car les méthodes centralisées demanderaient trop de communication pour l'acheminement des informations vers l'unité centrale et consommeraient donc trop d'énergie [161].

V. Technologies de signaux utilisés dans la localisation

On distingue plusieurs types de signaux utilisés dans la localisation selon le type de capteurs, l'environnement ou bien selon la technologie de mesure utilisée.

5.1. Infrarouges

On distingue plusieurs types d'utilisation de l'infrarouge pour la localisation. Premièrement on peut placer une série d'émetteurs (typiquement des LEDs) infrarouges autour d'une zone et alors un dispositif muni de détecteurs infrarouges peut, en utilisant les caractéristiques des divers signaux infrarouges émis par les différentes sources, se localiser et trouver son orientation dans la zone couverte par les émetteurs [162]. On utilise également les infrarouges pour permettre à des robots de se localiser dans un environnement (chaque robot a des émetteurs et des récepteurs infrarouges placés dans plusieurs directions et peut émettre des infrarouges. L'intensité de lumière reçue en retour permet d'estimer la distance entre eux [163].

5.2. (Ultra) sons

Il y a eu beaucoup de recherches sur la localisation à base de sons ou ultrasons. Le principal avantage de la technologie est que la vitesse de propagation du son est assez lente, en comparaison à celle des ondes. Cela permet de mesurer les temps de propagation précisément et ainsi d'obtenir des estimations de distances fiables. Le système cricket [164], par exemple, combine des communications par radio et des émetteurs/récepteurs de sons pour se localiser. Pour calculer les distances jusqu'à ses voisins, un 'cricket' envoie simultanément une onde radio et une onde sonore ayant une certaine forme caractéristique. Les nœuds qui reçoivent l'onde radio démarrent un minuteur et attendent l'arrivée de l'onde sonore. Lorsque celle-ci arrive ils consultent le minuteur et peuvent ainsi déduire le temps de propagation du son. A partir de cela ils calculent les distances de façon assez précise et en déduisent les positions. La seule restriction de ce système est qu'il ne peut être utilisé sur de trop longues distances et qu'il est parfois difficile à utiliser dans des environnements trop bruyants ou ayant des obstacles ou des murs [165].

5.3. Radio frequencies

5.3.1. GPS : Global positioning system

La technologie de localisation la plus répandue est sans doute le GPS. Il remplit très bien son rôle de système de localisation à l'échelle planétaire. Les satellites émettent en continu des signaux radios. Ces signaux radios contiennent une description des trajectoires de chaque satellite. Les récepteurs GPS reçoivent ces informations et connaissent ainsi les positions des satellites. Ils calculent également les temps que prends l'onde radio pour arriver de chaque satellite et en déduisent ainsi les distances jusqu'à chacun d'eux. Ils déduisent ensuite, en utilisant la multilatération leurs positions.

5.3.2. RFID : Radio Frequency Identification

C'est une technologie très en vogue ces derniers temps. Elle permet par exemple de localiser et identifier un objet dans un stock. Les premiers systèmes RFID furent inventés en 1939 par des anglais sous le nom d'IFF (Identification Friend or Foe) et furent utilisés durant la deuxième guerre mondiale par les pilotes alliés afin d'identifier les autres avions comme amis ou ennemis. RFID est un terme générique désignant les technologies qui utilisent les ondes radios pour identifier des gens ou des objets. Il y a plusieurs méthodes d'identification mais en général on stocke un numéro de série - qui désigne une personne ou un objet - et parfois d'autres informations sur une puce électronique attachée à une antenne (l'antenne et

la puce forment ce que l'on appelle un émetteur RFID ou tag RFID). Cela permet à un lecteur RFID de questionner les tags et de recevoir leurs informations. On distingue les tags passifs et les tags actifs. Les tags passifs n'ont pas de source de courant et utilisent donc l'énergie des ondes émises par le lecteur RFID (devant être à proximité) pour s'activer et répondre. Les tags actifs ont une source de courant propre et peuvent donc être plus actifs et indépendants [166].

5.3.3. RF-UWB : Ultra Wide Band

L'Ultra Wide Band est un nouveau type de radio sans fil. Il est caractérisé par une grande largeur de bande par rapport à la fréquence centrale des ondes émises. Il y a deux facteurs derrière le concept UWB : (1) Bande passante relativement large et (2) une fréquence centrale relativement petite. La grande largeur de bande permet des résolutions temporelles précises et dans des systèmes bien architecturés une meilleure confidentialité. La basse fréquence centrale quant à elle devrait permettre un meilleur passage des ondes à travers les différents matériaux. Divers systèmes de localisation ont été testés sur les radios UWB et semblent donner des résultats très prometteurs.

5.3.4. RF-WIFI, RF-Bluetooth

Les méthodes utilisées pour la localisation dans les réseaux WIFI et Bluetooth sont assez comparables à celles utilisées dans les réseaux de capteurs. Ces 3 types de réseaux utilisent des technologies radio assez comparables (le standard radio des RCSF est un sous-standard de Bluetooth).

5.4. Image

Les méthodes de localisation par la vision sont nombreuses, basées sur du traitement d'images et sont très différentes des méthodes de localisation habituelles. La difficulté consiste à extraire des informations pertinentes des pixels formant l'image. L'extraction d'informations depuis des images forme un domaine de recherche à part entière et qui sort du cadre de ce mémoire. Divers travaux de recherche sont en cours dans ce domaine.

Technologie	Remarque
Infrarouges	Doit être en ligne de vue Inutilisable en lumière de soleil Précision : 5-10m
Ultrasons	Pas d'obstacle Précision : 1-10m
Radio Fréquence	Très utile GPS précision : 5-10m RFID précision : 5cm-5m Cellulaire précision : 50-100m Wifi précision : 2-50m Bluetooth, ZigBee précision :2-10m
Optique	Doit être en ligne de vue Bonne précision
UWB Radio	Expérimentale Précision :6-10m

Tableau 4.1.Technologies de localisation [167]

VI. Technologies de mesure

Plusieurs technologies permettent à un capteur de mesurer la distance qui le sépare d'un capteur voisin (ToA, TDoA, RSSI) ou bien de mesurer l'angle qu'il forme avec celui-ci (AoA).

6.1. Différence des temps d'arrivée

La technologie TDoA (Time Difference of Arrival) se base sur la différence des dates d'arrivée d'un ou plusieurs signaux et suppose également que la vitesse de propagation des signaux est connue. Cette technologie s'applique dans les cas suivants :

- Un émetteur envoie des signaux de natures différentes (par exemple, l'ultrason, l'onde radio,..) à un récepteur.
- Un récepteur reçoit des signaux d'une même nature d'au moins trois émetteurs

- Un émetteur envoie un signal reçu par au moins trois récepteurs (dans ce dernier cas une vue globale des signaux sera connue)

Dans chacun des cas, les récepteurs mettent en corrélation leurs informations et en déduisent les distances qui les séparent des émetteurs. Il s'agit d'une simple résolution d'un système d'équations dont les distances sont les inconnues.

6.2. Temps d'arrivée

La technologie ToA (Time of Arrival) suppose que les nœuds du réseau sont synchrones. La distance qui sépare deux capteurs se déduit de la vitesse de propagation du signal et de la différence entre les dates d'émission et de réception du message. Cette technologie est celle utilisée par le système GPS. Lorsque les nœuds ne sont pas synchrones, l'envoi d'un message aller-retour est nécessaire. En fonction de son horloge, de la vitesse de propagation du signal et du temps de traitement du signal reçu, un capteur récepteur obtient la distance qui le sépare du capteur émetteur en calculant la différence entre les dates d'émission et de réception, en y soustrayant le temps de traitement du signal, puis en divisant le résultat par deux. Cela suppose que les nœuds du réseau ont un temps de traitement du signal identique.

6.3. Puissance du signal

La puissance d'émission et de réception d'un signal peut être également exploitée pour obtenir la distance entre deux capteurs. La technologie RSSI (Received Signal Strength Indicator) considère la perte de puissance d'un signal entre son émission et sa réception. Cette perte (path loss) varie en fonction de la distance entre les deux capteurs : plus les capteurs sont éloignés, plus la perte est importante. Cette perte sera alors traduite en une distance.

6.4. Angle d'arrivée

La technologie AoA (Angle of Arrival) calcule l'angle formé entre deux capteurs. Chaque capteur est doté d'antennes orientées de sorte à déduire l'angle qu'il forme avec un voisin lorsque ce dernier lui envoie un signal. Cet angle est reporté par rapport à un axe propre au capteur. Toutefois, un capteur peut être équipé d'une boussole et, dans ce cas, l'angle sera reporté sur un des axes nord, sud, est ou ouest.

VII. Algorithmes de localisation

7.1. Algorithmes basés sur les méthodes Range-based

Les méthodes Range-based utilisent les technologies ToA, RSSI, AoA et autres afin de mesurer les distances ou les angles entre deux capteurs voisins. Grâce à cette capacité de mesure, un capteur pourra, sous certaines conditions, obtenir sa position exacte. Autrement, une position estimée lui sera attribuée. Les méthodes Range-based sont les plus répandues.

7.1.1. SumDistMinMax

Dans la méthode SumDistMinMax, chaque ancre commence par diffuser sa position. Lorsqu'un capteur reçoit cette position, il estime la distance qui le sépare de cette ancre en appliquant la technique SumDist qui est la plus simple pour l'estimation de la distance entre un capteur et une ancre. Elle consiste à ajouter les distances mesurées entre chaque paire de capteurs voisins séparant l'ancre et le capteur qui cherche à estimer sa position. Lorsqu'une ancre envoie sa position, elle joint au message la distance qui la sépare d'elle-même, c'est à dire 0. A la réception de ce message, chaque capteur voisin mesure la distance avec l'émetteur, l'enregistre, ajoute cette distance à celle contenue dans le message et fait suivre la position de l'ancre avec la distance mise à jour. Le même processus est répété pour tous les capteurs.

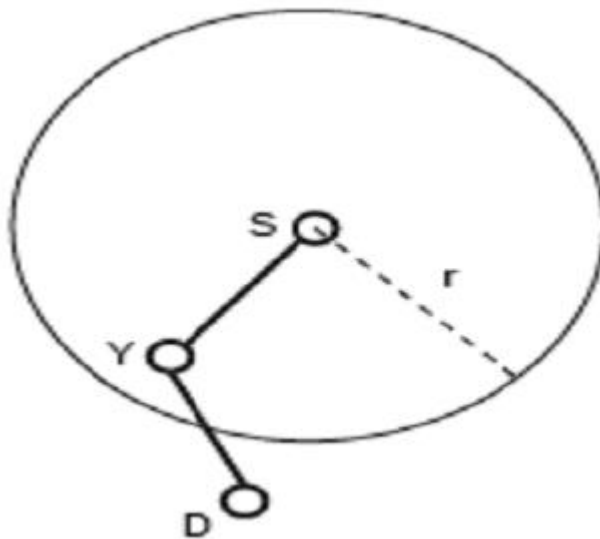


Figure 4.3. Algorithme SumDistMinMax

Au final, chaque capteur obtient la position de chacune des ancres et calcule une estimation de la distance qui le sépare de celle-ci. La figure 4.3 illustre le fonctionnement de

SumDist. Sur cette figure, la distance estimée entre les nœuds S et D est égale à $d_{SY} + d_{YD}$ sachant que d_{SD} est défini par l'inégalité triangulaire :

$$d_{SD} \leq d_{SY} + d_{YD} \quad (1)$$

Soit X_1, X_2, \dots, X_q , a étant un chemin d'un capteur X_1 vers une ancre a, l'estimation de la distance entre X_1 et a (notée dx_1a) est définie de façon récursive comme suit :

$$dx_1a = dx_1x_2 + dx_2a \quad (2)$$

Après cette phase d'estimation des distances avec les ancres, les capteurs calculent leurs positions estimées en utilisant la méthode MinMax. Le principe de cette méthode est de déterminer, pour chaque capteur, une « boîte » le contenant dont le centre de gravité correspond à sa position estimée.

7.1.2. DV-Distance

Cette méthode est similaire à la précédente sauf que la distance entre voisins est calculée en se référant à la puissance du signal reçu et elle est exprimée en mètres et non en nombre de sauts (Figure 4.4). Dans ce cas, l'algorithme à vecteur de distance utilise comme métrique la distance cumulative exprimée en unité de longueur.

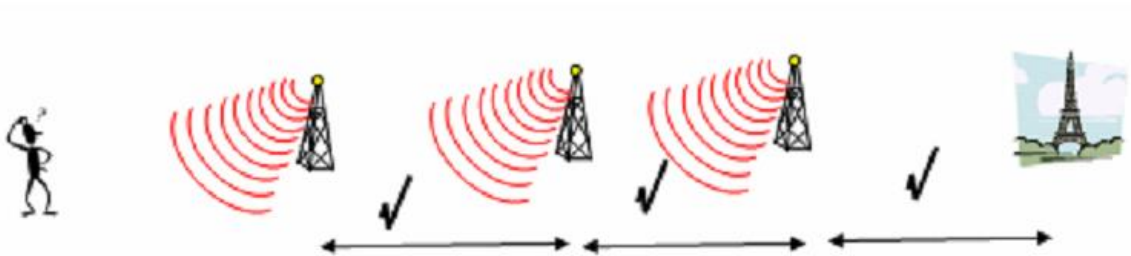


Figure 4.4. Algorithme DV-Distance

Cette méthode est plus concrète que la précédente car elle ne considère pas que tous les nœuds sont équidistants les uns des autres. Elle est, par contre, sensible à la propagation d'erreur de mesure.

7.1.3. DV-Euclidean

Cette troisième méthode repose sur la propagation de la distance euclidienne séparant le nœud de l'ancre.

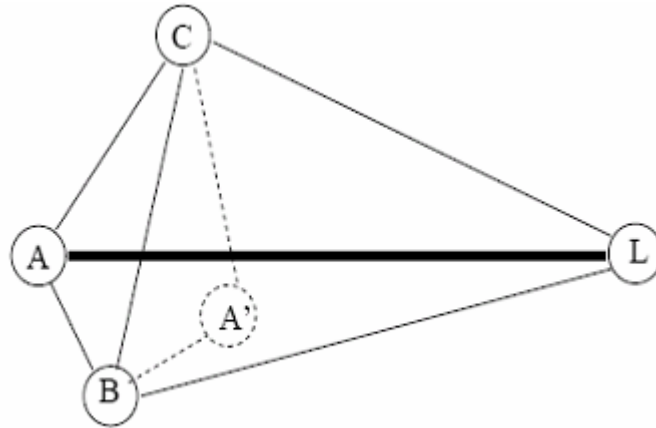


Figure 4.5. Algorithme DV-Euclidean

Soit un nœud A du réseau, pour déterminer sa position, il doit recevoir la position d'au moins deux de ses voisins : B et C. (figure 4.5). A doit également estimer les distances AB et AC par mesure de puissance du signal reçu, et BC qui peut être déduite à partir de la reconstitution du voisinage de A. Si on considère le quadrilatère ABCL, où les côtés sont tous connus, la diagonale BC l'est aussi, on peut donc calculer AL qui représente la distance euclidienne de A à l'ancre L. Cette méthode de propagation a l'avantage d'utiliser les mesures réelles, coordonnées GPS par exemple, ce qui évitera la propagation d'erreur de mesure.

7.2. Algorithmes basés sur les méthodes Range-free

Il existe plusieurs algorithmes appartenant aux méthodes Range-free nous citons :

7.2.1. Centroid

Les hypothèses de départ de *Centroid* [168] incluent une propagation sphérique isotrope ainsi qu'une portée identique pour toutes les stations. La méthode se base sur un quadrillage de nœuds de référence émettant des *beacons* à un intervalle T sans collision. Selon une métrique de connectivité calculée localement (équation 4) et un seuil *Tresh*, le mobile décide de tenir compte d'un sous-ensemble de balises détectées. Les facteurs de cette caractéristique sont : $N_{recv}(i, t)$: le nombre de *beacons* émis par la balise i et reçus durant le temps t et $N_{sent}(i, t)$: le nombre de *beacons* émis par la balise i durant le temps t.

Les coordonnées des balises retenues sont utilisées pour calculer le centre de gravité de la zone, position que le nœud considèrera comme sienne. Sur la figure 4.6, une réalisation de *Centroid* est illustrée avec uniquement les balises retenues représentées par un point. La position estimée est représentée par une croix.

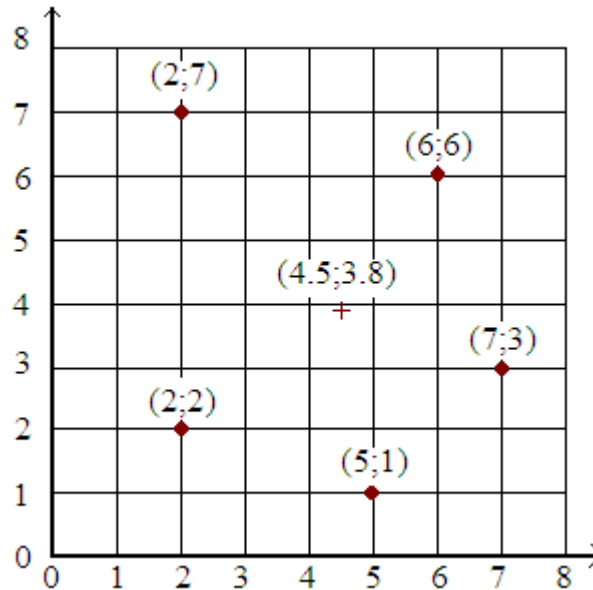


Figure 4.6.Exemple de localisation avec Centroid

La validation par un prototype en environnement extérieur a impliqué quatre balises placées aux coins d'un carré de 10 m X 10 m et a mené à une erreur moyenne de 2m.

Le scénario est montré dans la figure 4.7..Dans le réseau, il y a au total m ancrés situées à des positions connus, $A_1(x_1, y_1), A_2(x_2, y_2) \dots A_m(x_m, y_m)$. Toutes ces ancrés ont la même portée de communication notée R . Leurs zones de transmission ont une superposition, comme indiqué par la partie ombrée dans la figure. A l'intérieur de la superposition se localise le nœud normal N_x . Cela signifie que, toutes ces m ancrés sont les ancrés voisins de N_x .

La topologie du réseau est maillée. Chaque ancre périodiquement (période = T) transmet un signal de balise contenant la position. Il est supposé que tous les ancrés sont bien synchronisés sans collisions se produisant pendant les transmissions. Pour faciliter l'explication et avant d'introduire l'algorithme, l'auteur définit quelques termes énumérés ci-dessous :

R : Portée de transmission de noeud(range)

T : Intervalle de temps entre deux signaux de balise transmis par une ancre

t : Le Noeud normal N_x utilise ce laps de temps pour collecter des signaux de balise, $t > T$

$N_{sent}(i, t)$: Nombre des balises envoyé par l'ancre A pendant un temps t

$N_{recv}(i, t)$: Nombre des balises reçu par le noeud normal pendant un temps t (Les balises sont envoyées par l'ancre A_i)

CM_i : Le métrique de connectivité pour l'ancre A

CM_{thresh} : Seuil de CM

(x_{cen}, y_{cen}) : Position estimée du noeud normal par l'algorithme Centroid

(x_a, y_a) : Position réelle du noeud normal

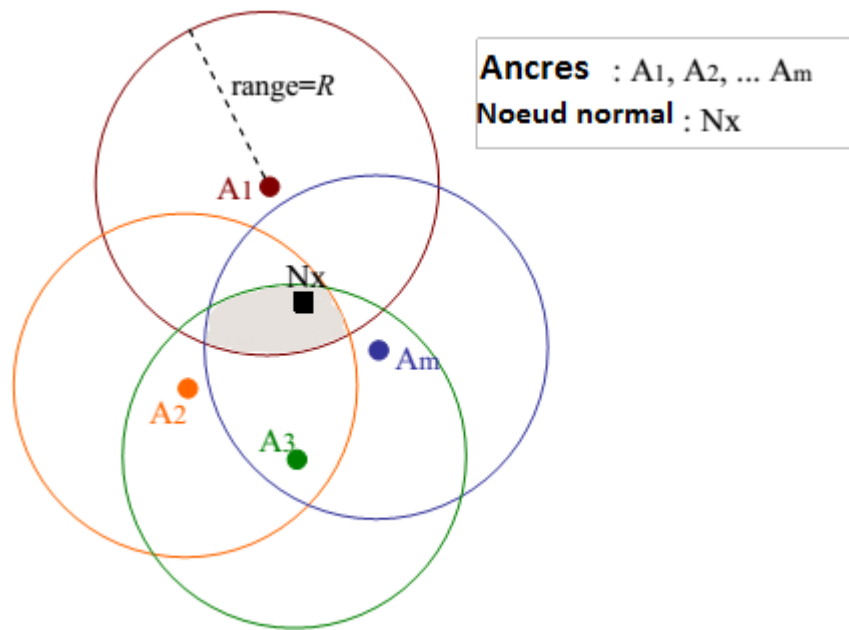


Figure 4.7. Exemple de localisation avec Centroid [168]

Pendant un temps fixe t , le nœud normal N_x écoute le canal et collecte tous les signaux de balise de diverses ancres. Bien que chaque ancre A_i a envoyé $N_{sent}(i, t)$ signaux, en raison de la propagation radio interférences, le noeud normal peut effectivement recevoir $N_{recv}(i, t)$ signaux de A_i . Notons que :

$$N_{recv}(i, t) \leq N_{sent}(i, t) \quad (3)$$

Afin de savoir si une ancre est vraiment dans la gamme radio du nœud normal, l'auteur définit la connectivité métrique pour chaque ancre A_i , noté CM_i :

$$CM_i = \frac{N_{recv}(i, t)}{N_{sent}(i, t)} \quad (4)$$

L'auteur définit également un seuil pour CM_i , noté CM_{thresh} . Si CM_i , est plus grand que CM_{thresh} , le nœud normal N_x considèrent l'ancre correspondant A_i comme voisin de N_x .

Par conséquent, lors du calcul de la position, N_x va prendre A_i en considération. Toutefois, si CM_j , est plus petit que CM_{thresh} N_x verra que l'ancre A_j n'est pas dans sa proximité, puis N_x va pas prendre en considération A_j lors de l'estimation de sa position.

Supposons que, finalement, N_x peut avoir k ancres dont leurs métriques de connectivité sont plus grandes que CM_{thresh} . Ces k ancres sont A_1, A_2, \dots, A_k . Alors N_x se localise au centre de gravité (Centroid) de ces k ancres:

$$x_{cen} = \frac{x_1 + x_2 + \dots + x_k}{k}$$

$$y_{cen} = \frac{y_1 + y_2 + \dots + y_k}{k} \quad (5)$$

La procédure de programme d'algorithme Centroid se résume comme suit :

1. Algorithme "Centroid":
2. Au cours d'une période t , un nœud normal N obtient les positions des k ancres A_1, A_2, \dots, A_k .
3. $x_{cen} \leftarrow 0; y_{cen} \leftarrow 0$
4. **For** $i \leftarrow 1$ **to** k
5. **do** $x_{cen} \leftarrow (x_{cen} + x_i); y_{cen} \leftarrow (y_{cen} + y_i)$ avec (x_i, y_i) est la position de A_i
6. $x_{cen} \leftarrow \frac{x_{cen}}{k}; y_{cen} \leftarrow \frac{y_{cen}}{k}$
7. **return** x_{cen} and y_{cen}

Procédure d'algorithme centroid

La précision de l'algorithme centroid est estimé par l'erreur de localisation défini comme suit [169] :

$$erreur\ de\ localisation = \sqrt{(x_{cen} - x_a)^2 + (y_{cen} - y_a)^2} \quad (6)$$

7.2.2. Weighted Centroid Localization (WCL)

Certaines améliorations ont été proposées ces dernières années. Cependant, la plupart d'entre eux doivent ajouter les informations RSSI. Par exemple, un algorithme de Weighted Centroid Localization (WCL) est proposé. WCL ajoute les informations RSSI à l'algorithme Centroid originale, en associant des poids pour les liens entre les nœuds mobiles et les ancres. La position estimée du nœud mobile est calculée comme suit:

$$x_{wcl} = \frac{\sum_{i=1}^m (w_i \times x_i)}{\sum_{i=1}^m w_i}, \quad y_{wcl} = \frac{\sum_{i=1}^m (w_i \times y_i)}{\sum_{i=1}^m w_i} \quad (7)$$

Dans l'équation (7), (x_i, y_i) sont les coordonnées de l'ancre A_i , et w_i est le poids associé à la liaison entre le noeud mobile et A_i .

La valeur de w_i est déterminée par $RSSI_i$ qui est la valeur RSSI du signal reçu (envoyé à partir de A_i arrive au noeud mobile).

7.2.3. DV-Hop

C'est le schéma le plus basique, il utilise un échange de vecteur de distance afin que tous les noeuds du réseau parviennent à calculer la distance les séparant du ancre [170]. Chaque ancre maintient une table $\{X_i, Y_i, h_i\}$ où $\{X_i, Y_i\}$ sont les coordonnées des autres ancres du réseau et h_i est le nombre de sauts séparant ce dernier du noeud en question. Chaque ancre calcule la distance le séparant des autres ancres dans le réseau, en utilisant les informations de localisation obtenues à partir d'un système de positionnement, il en déduit une approximation de la distance par saut. C'est la distance par saut qui va constituer l'information de correction pour tout le réseau.

Chaque noeud ancre calcule :

$$Hopsize_i = \frac{\sum_{i \neq j} \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}}{\sum_{i \neq j} h_{ij}} \quad (8)$$

Après que tous les noeuds inconnus reçoivent le hop-size des noeuds ancres qui ont le minimum saut entre eux, ils calculent la distance aux noeuds d'ancrage sur la base de deux facteurs: le hop-size et le minimum hop count (h_{id}) en utilisant la formule suivante :

$$d_i = h_{id} * HopSize_i \quad (9)$$

Dans la troisième étape, les noeuds inconnus calculent leur position par rapport à la distance pour chaque noeud d'ancrage obtenu dans la deuxième étape.

Soit (x, y) les coordonnées des noeuds inconnus et (x_i, y_i) les coordonnées des ancres i . Posant d_i est la distance entre l'ancre i et les autres noeuds inconnus et après on obtient les formules suivantes :

$$\left\{ \begin{array}{l} (x - x_1)^2 + (y - y_1)^2 = d_1^2 \\ (x - x_2)^2 + (y - y_2)^2 = d_2^2 \\ \vdots \\ (x - x_n)^2 + (y - y_n)^2 = d_n^2 \end{array} \right. \quad (10)$$

La formule (10) peut être écrite sous la forme d'une équation linéaire à la suite linéaire:

$$AP = B \quad (11)$$

Avec

$$P = \begin{pmatrix} x \\ y \end{pmatrix} \quad (12)$$

$$A = -2 * \begin{bmatrix} x_1 - x_n & y_1 - y_n \\ x_2 - x_n & y_2 - y_n \\ \dots & \dots \\ x_{n-1} - x_n & y_{n-1} - y_n \end{bmatrix} \quad (13)$$

$$B = \begin{bmatrix} d_1^2 - d_n^2 - x_1^2 + x_n^2 - y_1^2 + y_n^2 \\ d_1^2 - d_n^2 - x_2^2 + x_n^2 - y_2^2 + y_n^2 \\ \vdots \\ d_{n-1}^2 - d_n^2 - x_{n-1}^2 + x_n^2 - y_{n-1}^2 + y_n^2 \end{bmatrix} \quad (14)$$

La position du nœud inconnue est obtenue en utilisant la méthode des moindres carrés qui peut être exprimée comme suit:

$$P = (A^T A)^{-1} A^T B \quad (15)$$

❖ Illustration du DV-Hop

L'exemple suivant illustre le calcul de DV-Hop

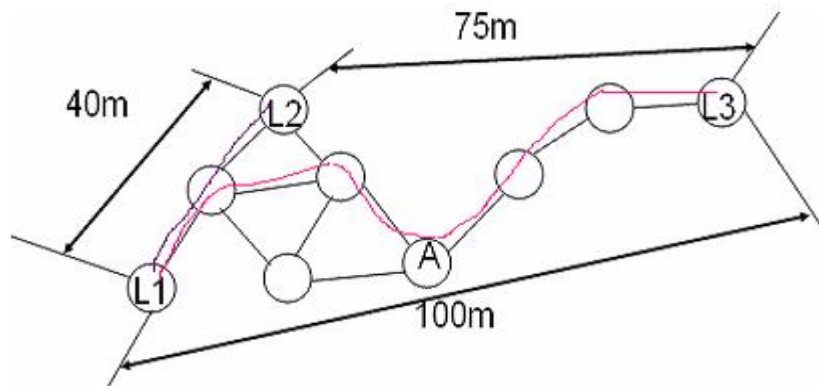


Figure 4.8. Algorithme DV-HOP [23]

Soit dans la figure 4.8, les nœuds L1, L2 et L3 représentent des ancrs. Et A le nœud voulant calculer sa position. Chaque ancre calcule la correction (hop size) et la diffuse.

$$L1 \rightarrow \frac{(100+40)}{(6+2)} = 17.50$$

De même pour L2 et L3

$$L2 \rightarrow \frac{(75+40)}{(5+2)} = 16.42$$

$$L3 \rightarrow \frac{(100+75)}{(6+5)} = 15.90$$

Un nœud quelconque du réseau obtient une mise à jour de l'ancre le plus proche. La diffusion des corrections à travers le réseau est contrôlée : quand un nœud reçoit ou émet une mise à jour, il doit tout d'abord supprimer les anciennes. Lorsque le réseau est étendu, un champ TTL (Time To Leave) est utilisé pour localiser les mises à jour au voisinage de l'ancre.

Dans l'exemple précédent, le nœud A peut estimer la distance le séparant à L1, L2, et L3 à travers la mise à jour de correction reçue de L2 ainsi A calcule la distance qui le sépare des ancrs L1, L2 et L3:

$$A \rightarrow L1 = 16.42 * 3$$

$$A \rightarrow L2 = 16.42 * 2$$

$$A \rightarrow L3 = 16.42 * 3$$

A partir de ces trois valeurs, le nœud A peut déterminer sa position relative en appliquant le même principe qu'en GPS. Pour obtenir leurs positions, les capteurs utilisent

ensuite la multilatération. En reprenant l'exemple de la figure 4.8, X obtiendra sa position en résolvant le système suivant :

$$\begin{cases} d_{L_1L_2}^2 = (X_A - X_{L_1})^2 + (Y_A - Y_{L_1})^2 \\ d_{L_2L_3}^2 = (X_A - X_{L_2})^2 + (Y_A - Y_{L_2})^2 \\ d_{L_1L_3}^2 = (X_A - X_{L_3})^2 + (Y_A - Y_{L_3})^2 \end{cases} \quad (16)$$

L'avantage du DV-Hop est sa simplicité. L'inconvénient est qu'il n'est adapté qu'au réseau isotrope où les propriétés du graphe sont indépendantes de la direction.

7.2.4. Amorphous

L'Algorithme amorphous est similaire à l'algorithme DV-Hop, et l'idée est de calculer la distance de saut (hop distance) entre deux nœuds au lieu de calculer la distance linéaire entre eux. L'algorithme Amorphous est constitué de trois étapes suivantes [171] :

1. Calcul du saut minimum (*Minimum Hop*) entre le nœud inconnu et le nœud ancre (beacon) :

Chaque nœud ancre (de balise) envoie des messages aux nœuds inconnus par la méthode d'inondation. La formule (17) est utilisée pour calculer le saut minimum à partir du nœud i au nœud k .

$$S_{(i,k)} = \frac{\sum_{j \in nbrs(i)} h_{(j,k)} + h_{(i,k)}}{|nbrs(i)| + 1} - 0.5 \quad (17)$$

Avec :

$S_{(i,k)}$: est le saut minimum (*Minimum Hop*) du nœud inconnu i au nœud ancre k

$h_{(j,k)}$: est le saut entier du nœud inconnu j au nœud ancre k

$h_{(i,k)}$: est le saut entier du nœud inconnu i au nœud ancre k

$nbrs(i)$: sont les nœuds voisins autour du nœud inconnu i

$|nbrs(i)|$: Le nombre des nœuds voisins autour du nœud inconnu i

2. Calcul de la distance du nœud inconnu au nœud ancre (beacon)

La Formule (18) est utilisé pour calculer la distance moyenne d'un hop:

$$HopSize = r \left(1 + e^{-n_{local}} - \int_{-1}^1 e^{-(n_{local}/\pi)(\arccos t - t\sqrt{1-t^2})} dt \right) \quad (18)$$

Avec r est la portée sans fil du nœud et n_{local} est la connectivité moyenne du réseau.

La distance d du nœud inconnu au nœud ancre est calculée en se basant sur la distance moyenne d'un saut et le saut minimum du nœud inconnu au nœud ancre [172]. Il peut être exprimé par la formule suivante :

$$d = HopSize_i \times S_{(i,k)} \quad (19)$$

3. Adoption de la méthode des moindres carrés pour localiser

Lorsque les distances estimées à partir du nœud inconnu aux trois ou plus de trois nœuds ancrés ont été obtenus, la localisation du nœud inconnu peut être calculée. Il est montré que :

$$\begin{aligned} (x_1 - x)^2 + (y_1 - y)^2 &= d_1^2 \\ &\vdots \\ (x_n - x)^2 + (y_n - y)^2 &= d_n^2 \end{aligned} \quad (20)$$

La formule ci-dessus peut être résolue par la méthode des moindres carrés, la localisation du nœud inconnu peut être obtenue:

$$X = (A^T A)^{-1} A^T b \quad (21)$$

Avec $(x_1, y_1), (x_2, y_2), (x_3, y_3) \dots \dots (x_n, y_n)$ sont les coordonnées de n nœuds ancrés, et $d_1, d_2, d_3, \dots, d_n$ sont les distances à partir du nœud inconnu aux nœuds ancrés.

$$A = 2 * \begin{bmatrix} x_1 - x_n & y_1 - y_n \\ \dots & \dots \\ x_{n-1} - x_n & y_{n-1} - y_n \end{bmatrix} \quad (22)$$

$$b = \begin{bmatrix} x_1^2 - x_n^2 + y_1^2 - y_n^2 + d_n^2 - d_1^2 \\ \vdots \\ x_{n-1}^2 - x_n^2 + y_{n-1}^2 - y_n^2 + d_n^2 - d_{n-1}^2 \end{bmatrix} \quad (23)$$

La localisation du nœud inconnu peut être calculée sur la base de (22) et (23).

VIII. Simulation et Analyse

8.1. Topologie du réseau

La topologie du réseau est une topologie isotropique :

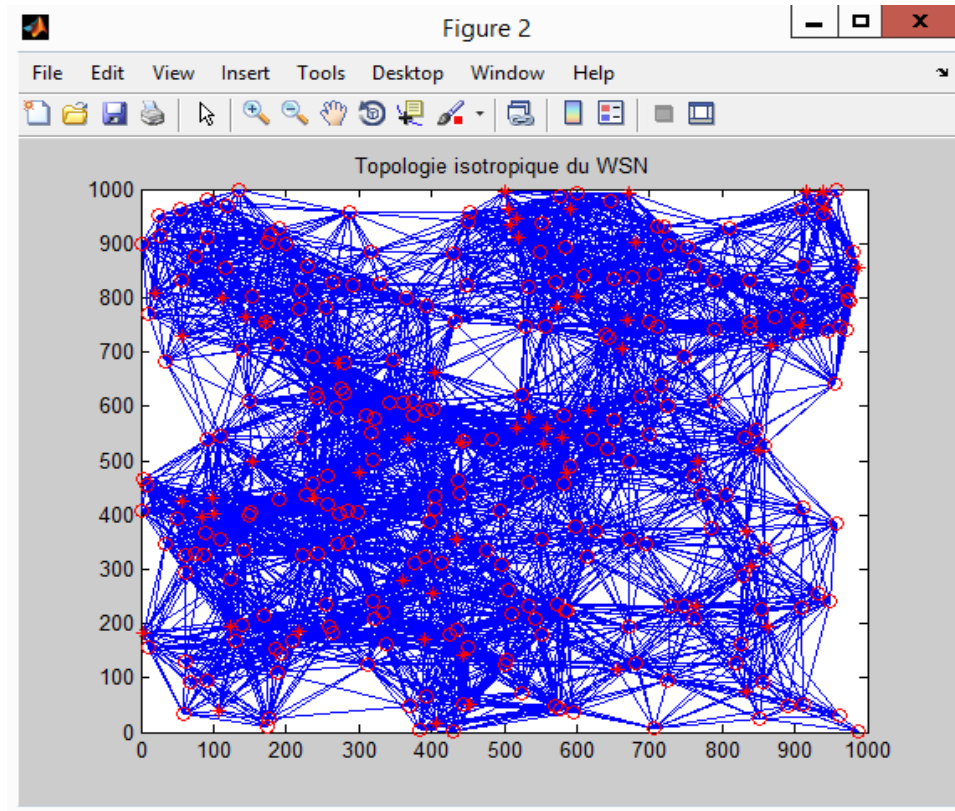


Figure 4.9. Topologie isotropique du réseau

8.2. Plate-forme de simulation et distribution de nœuds

MATLAB simulation software can be used to verify the feasibility of the proposed algorithm. The deployment area of the network is $1000\text{ m} \times 1000\text{ m}$, the coordinates of the nodes are generated randomly, the number is 300, there are 60 beacon nodes, the proportion of beacons is 20%, the wireless range is 300 m, and the communication model is the regular model. The distribution of nodes is represented in Figure 4.9.

8.3. Définition de l'erreur de localisation

The definition of localization error. Localization error is a very important indicator to evaluate localization performance. The following formula is used to calculate localization error:

$$\text{Localization error} = \frac{\sqrt{(x_1-x_2)^2+(y_1-y_2)^2}}{R} \quad (24)$$

Avec $(x_1, y_1)^2$ correspond à l'emplacement réel du nœud inconnu et $(x_2, y_2)^2$ est la position estimée, où R est la portée sans fil.

8.4. Simulation des algorithmes

On représente ici la simulation des algorithmes centroid, DV-Hop et amorphous .

La figure 4.10 représente la répartition des nœuds pour les trois algorithmes.

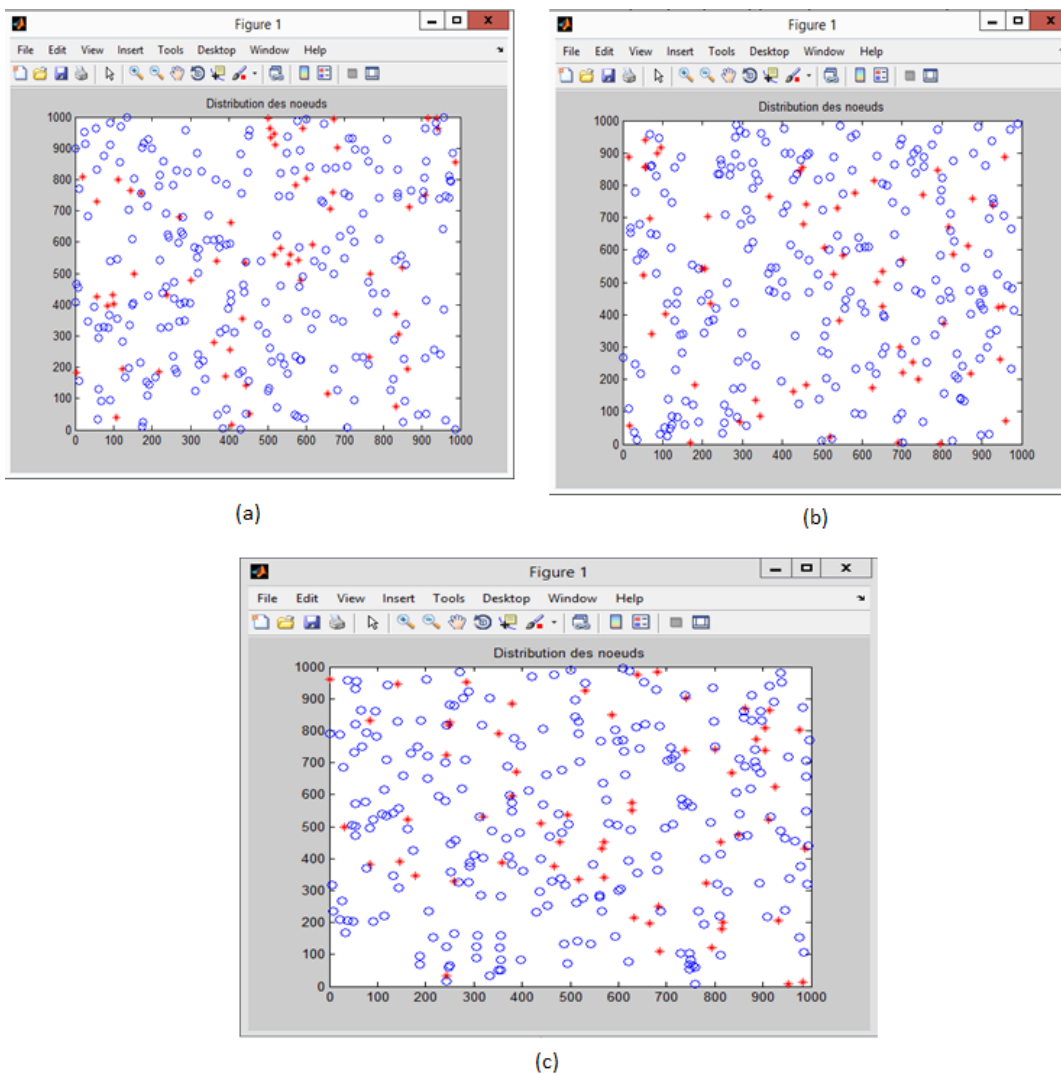


Figure 4.10. Répartition des nœuds (a) Centroid (b) DV-Hop (c) Amorphous

Les points étoiles rouges représentent les ancres et les petits cercles bleus représentent les nœuds inconnus. Tout d'abord, les algorithmes centroid, DV-Hop et amorphous peuvent être simulés pour obtenir la figure 4.11 qui représente l'erreur de localisation.

Sur la figure 4.11, le petit cercle bleu représente l'emplacement estimé du nœud inconnu et la ligne bleue représente l'erreur de localisation du nœud inconnu.

L'erreur de localisation de l'algorithme centroid est 0,2937. Sur la figure 4.11(b), l'erreur de localisation est 0,3022, la précision de localisation de l'algorithme Centroid est supérieure à celle de l'algorithme DV-hop. Sur la figure 4.11(c), l'erreur de localisation de l'algorithme amorphous est 0,2361. On voit que la précision de la localisation de l'algorithme amorphous est supérieure à celle de l'algorithme centroid et l'algorithme DV-Hop évidemment.

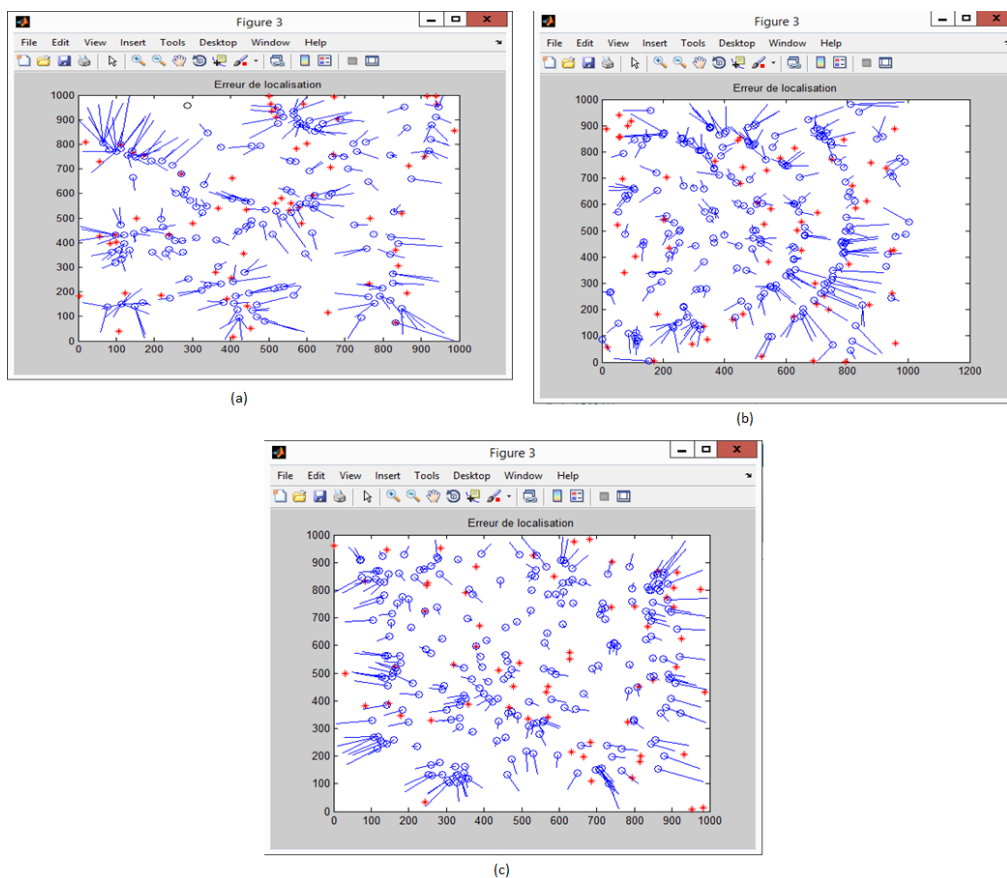


Figure 4.11. Erreur de localisation figures: (a) Centroid (b) DV-Hop ET (c) Amorphous

❖ Les Simulations dans différentes conditions sont remplies pour les trois algorithmes

Pour connaître la performance des différents algorithmes proposés dans ce travail, les simulations dans des conditions différentes sont effectuées pour l'algorithme Centroïde, DV-HOP et Amorphous. Tous les nœuds de la simulation sont répartis de façon aléatoire dans la zone de $1000\text{ m} \times 1000\text{ m}$.

Chaque cas de condition est exécuté pour 1000 fois afin que les faits reflètent, l'erreur de localisation de différents algorithmes. La valeur moyenne de l'erreur de localisation est utilisée pour la comparaison.

Dans le cas d'utilisation des différentes proportions de balises (des ancrs), il y a 300 nœuds au total avec une portée sans fil étant définies comme 300m. D'après la figure 4.12, il est clair que l'erreur de localisation des trois algorithmes diminue à mesure que la proportion de balises augmente.

Dans la simulation de différents portés sans fil, la proportion des ancrs est définie par 20% et le nombre total de nœuds est définie par 300. La figure 4.13 montre le résultat de la simulation. Il est clair que l'erreur de localisation des algorithmes diminue à mesure que la portée sans fil augmente. L'erreur de localisation de l'algorithme amorphous augmente lorsque la portée sans fil est plus de 300m.

Basé sur des nombres différents de nœuds, la portée sans fil est définie comme 300m, et la proportion de balises (ancres) est définie comme 20%. Le résultat est montré sur la figure 4.14. Il est clair que l'erreur de localisation des trois algorithmes diminue à mesure que le nombre de nœuds augmente.

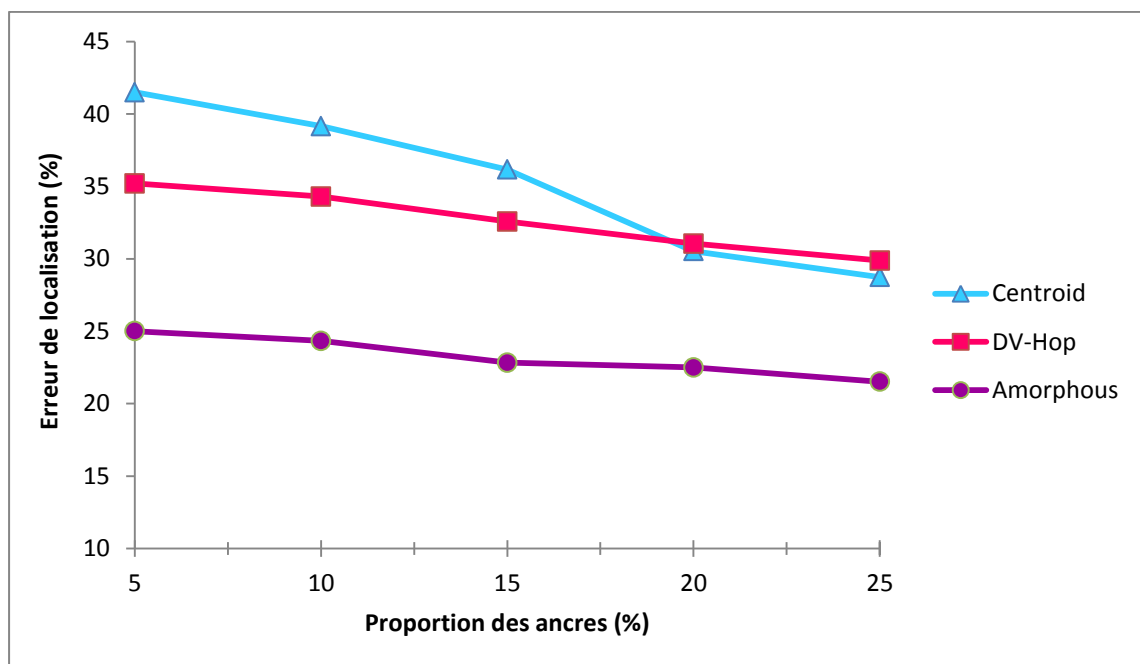


Figure 4.12. Proportion des ancrs et l'erreur de localisation

8.5. Temps de calcul d'algorithme amélioré est comparé à d'autres algorithmes

Selon l'analyse de l'article [173], les coûts informatiques ne représentent qu'une petite partie de la consommation d'énergie dans les WSN; la plupart de la consommation d'énergie est la communication. Donc, la diminution de la consommation d'énergie de la communication est la clé pour étendre le cycle de vie des réseaux. Afin de diminuer la consommation d'énergie de la communication, tous les nœuds ne peuvent pas envoyer les

informations à un nœud central pour calculer leur emplacement, parce que la consommation d'énergie de la communication est trop grande. Les algorithmes de localisation doivent adopter un calcul distribué. Le tableau 4.2 compare le temps de calcul moyen nécessaire par les algorithmes pour localiser un nœud unique et l'erreur de localisation.

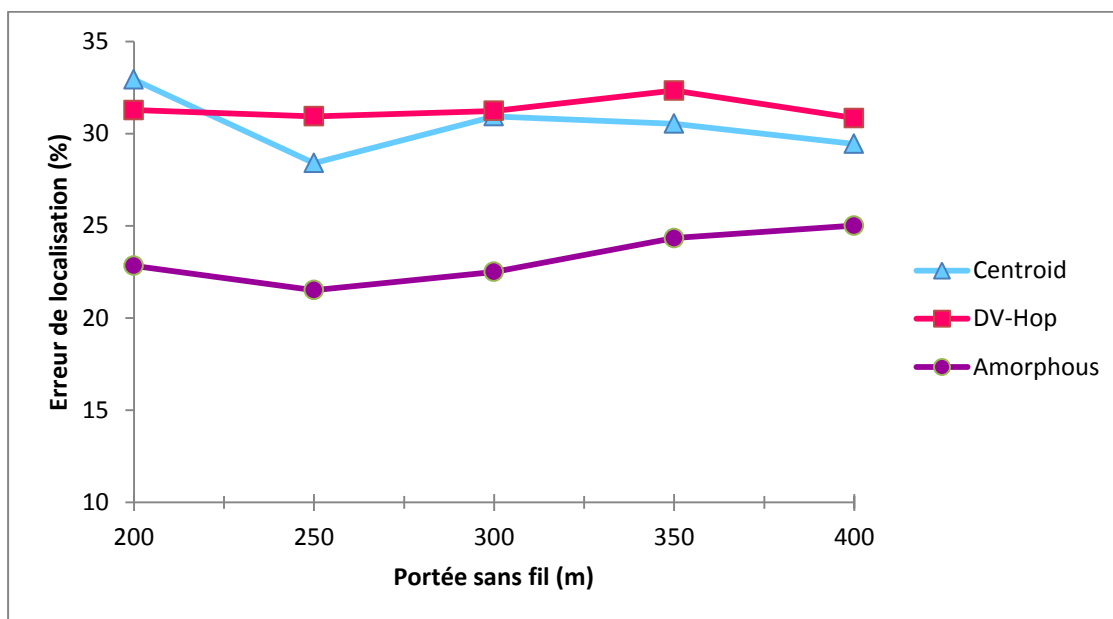


Figure 4.13. Portée radio et l'erreur de localisation

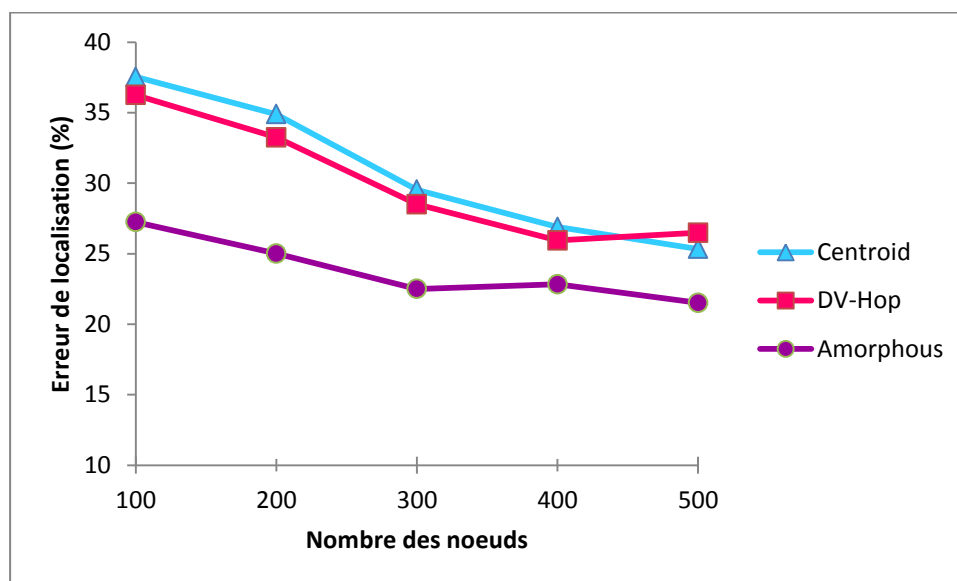


Figure 4.14. Nombre des nœuds et l'erreur de localisation

Toutes les expériences sont réalisées sur le même ordinateur. La Zone de déploiement du réseau est de 1000 m × 1000 m, les coordonnées des nœuds sont générées de façon aléatoire, le nombre est de 300, la proportion d'ancres est de 20%, la portée sans fil est de 300 m, et le modèle de communication utilisé est le modèle régulier. Grâce à son simple

calcul, l'algorithme Centroid nécessite généralement moins de temps de calcul que les autres algorithmes. L'Algorithme amorphous est similaire à l'algorithme DV-Hop.

Algorithme	Erreur de localisation	Temps d'exécution(s)
Centroid	0.2937	0.383
DV-Hop	0.3022	0.512
Amorphous	0.2361	0.536

Tableau 4.2. Temps moyen d'exécution pour de localiser un seul nœud et d'erreurs de localisation

8.6. Consommation d'énergie des algorithmes

Pour tester la performance des algorithmes proposés, les simulations de la consommation d'énergie dans des conditions différentes sont présentées pour les algorithmes Centroid, DV-HOP et amorphous.

Tous les nœuds de la simulation sont distribués au hasard dans la zone $1000 \text{ m} \times 1000 \text{ m}$. Chaque simulation est exécuté pour 1000 fois afin de déterminer la consommation d'énergie de différents algorithmes.

Dans le cas d'utilisation des différentes proportions de balises (des ancres), il y a 300 nœuds au total avec une portée sans fil étant définies comme 300m. D'après la figure 4.15 il est clair que l'erreur localisation des trois algorithmes diminue à mesure que la proportion de balises augmente.

Le résultat de la consommation d'énergie est représenté sur la figure 4.15. Il est clair que la consommation d'énergie des trois algorithmes augmente à mesure que la proportion des balises (des ancres) augmente. Sous la même proportion de balises, la consommation d'énergie de l'algorithme centroïde est la plus faible, parce qu'il diffuse qu'une seule fois. L'Algorithme DV-hop a besoin de diffuser deux fois, de sorte que la consommation d'énergie de la communication est importante [173]. La consommation d'énergie de l'algorithme Amorphous adopte un calcul distribué.

Dans le cas de différentes portée sans fil, il y a 300 nœuds au total avec une proportion de balises étant définie comme 20%, et dans le cas de différents nombres de nœuds, la portée sans fil est réglée à 300 m, la proportion de balises est définie comme 20 %. Les différents résultats sont présentés dans les figures 4.16 et 4.17. Il est clair que la localisation précise apportera plus de consommation d'énergie. Donc, les algorithmes de localisation doivent être conçus en fonction des différentes applications.

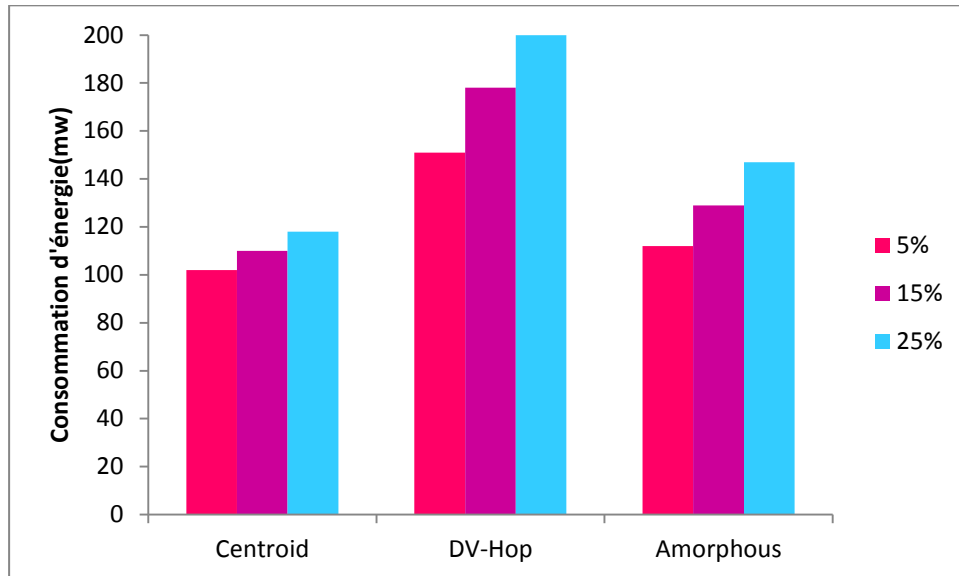


Figure 4.15 Consommation d'énergie pour de différentes proportions des ancres

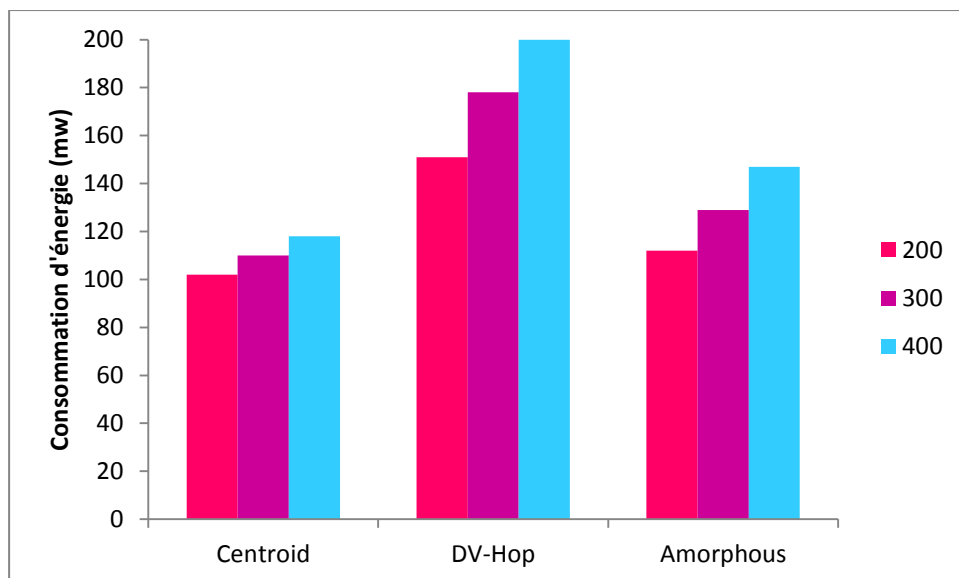


Figure 4.16. Consommation d'énergie pour de différentes portées sans fil

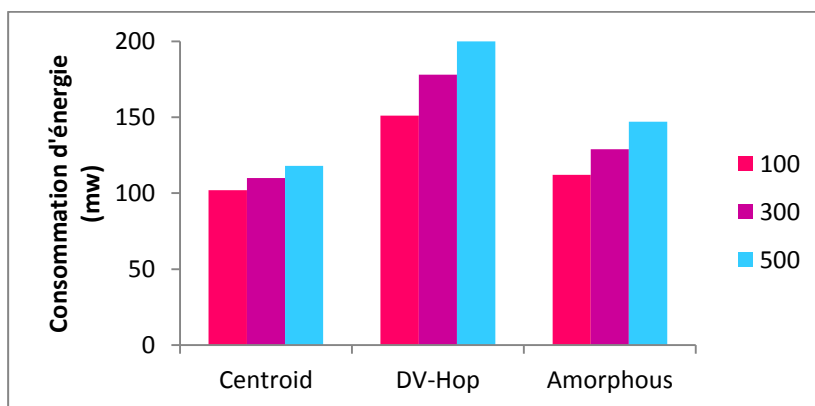


Figure 4.17. Consommation d'énergie pour de différents nombres des nœuds

IX. Conclusion

La précision de localisation est un indicateur très important pour évaluer la performance de localisation. Plus la précision de localisation n'est élevée, la performance de localisation est meilleure. En outre, la précision de la localisation de l'algorithme Amorphous est supérieure à celle des autres algorithmes et il y a pas une grande augmentation de la consommation d'énergie, c'est pourquoi il est adapté à la localisation des nœuds de réseau de grande échelle.

Routage et Localisation Conjointes dans les Réseaux WSNs: Nouvelle Approche

I. Introduction

Les réseaux de capteurs sans fils (Wireless Sensor Networks) sont un cas particulier des réseaux MANET (Réseaux Ad-hoc mobiles). Les protocoles de routage basés sur les topologies habituelles ont des performances limitées dans tels réseaux en raison de la haute dynamique du réseau. Les protocoles de routage géographiques ont été conçus pour offrir de meilleures performances pour ces réseaux. Chaque nœud doit prendre en considération sa position géographique réelle et la position du nœud cible à atteindre. Le service basé sur la localisation est nécessaire pour savoir la position de destination. La combinaison de ce service avec le routage est tout à fait naturelle afin d'assurer de meilleures performances. Si un nœud doit envoyer des données à un autre nœud du réseau, GPSR demande au service de localisation s'il possède une information fraîche sur l'emplacement de la destination. Si c'est le cas, GPSR utilise cette position pour transmettre les paquets à la destination. Autrement, le service de localisation lance une demande de localisation pour trouver la nouvelle position de la destination. Lorsque la réponse de localisation est reçue, le service de localisation informe GPSR, qui sera en mesure de transmettre les données vers le nœud destination en utilisant la position reçue. A cet effet, nous avons proposé un patch sur le simulateur NS-2 qui combine le protocole GPSR et les services GLS et HLS selon notre proposition. Nous avons entrepris une série d'expérimentations et nous avons examiné quatre critères de performance, le Cout de localisation (nombre des requêtes de localisation envoyées et la bande passante consommée par la localisation), l'impact de la fraîcheur des informations de localisation sur le réseau, la performance de localisation (Taux de Requêtes Réussites et Temps de déplacement de requête) et les performances du réseau (le Taux de Paquets Délivrés TPD et la latence moyenne).

Le chapitre est organisé de la manière suivante. La première partie est consacrée aux travaux connexes. La section III détaille notre algorithme de combinaison à propos de

GPSR, GLS et HLS. La section IV parle de nos expérimentations et les résultats obtenus et la section V présente la procédure et la démarche suivie pour faire l'exécution sous NS2 avec les différents scripts utilisés. Enfin, la dernière section conclut l'étude.

II. Protocoles de routage géographique

L'introduction de la technologie GPS a permis la proposition de nouveaux protocoles de routage plus performant. Les informations de géolocalisation peuvent être exploitées dans le routage des paquets de données où uniquement la position du nœud destination ainsi que les positions des voisins du nœud qui veut envoyer le paquet de données sont utilisés [174]. En effet, les protocoles de routage basés sur ce principe de routage ne nécessitent pas l'établissement ou la maintenance des chemins (pas de tables de routage). De plus, la taille des paquets contenant les informations de localisation est inférieure aux paquets généralement échangés dans les protocoles basés topologie ou dans ceux basés destination [174]. Tout cela permet une meilleure mise à l'échelle. En outre, les informations de géolocalisation peuvent être utilisées pour améliorer les performances des protocoles de routage réactifs en limitant la zone de propagation des RREQs lors de la phase de découverte de chemins [175].

Les algorithmes de protocoles de routage doivent choisir certains critères pour prendre des décisions de routage, par exemple le nombre de sauts, la latence, la puissance de transmission, bande passante, etc. Les Protocoles de Routage Topologiques sont pénalisés par des phases de découverte et de maintenance lourdes et coûteuses, ce qui conduit à des problèmes de mise à l'échelle. Cela est dû à la haute mobilité, ce qui engendre des liens courts et interrompus. C'est pourquoi les Protocoles de Routage Géographique ont été pensés afin de traiter ces problématiques.

LAR (The Location-Aided Routing) est le premier protocole de routage réactif qui exploite les informations de géolocalisation pour limiter la zone de propagation des paquets RREQs. Dans LAR [175], les nœuds décident d'acheminer ou de supprimer les paquets RREQs selon deux schémas "LAR1" et "LAR2" [175]. Dans LAR1, le nœud source inclut dans le paquet RREQ les coordonnées des sommets de la zone de requêtes. Cette dernière est définie comme la plus petite zone rectangulaire qui contient le nœud source et la zone dans laquelle il est probable de trouver le nœud destination. Un nœud accepte d'acheminer un paquet RREQ, si et seulement si, il se trouve dans la zone des requêtes. Dans LAR2, le nœud source inclut dans le paquet RREQ la distance qui le sépare du nœud destination avec les coordonnées de ce dernier. À la réception d'un RREQ, un nœud intermédiaire calcule la distance qui le sépare du nœud destination. Si elle est inférieure de la distance qui sépare la

source de la destination, il transmet le RREQ, sinon il le supprime. Dans LAR, un nœud destination répond à un paquet RREQ en envoyant un paquet RREP qui contient sa position courante, l'instant qui correspond à cette position et optionnellement sa vitesse. Le paquet RREP va suivre le chemin inverse que celui enregistré dans le paquet RREQ.

Un autre protocole de routage est le DREAM (Distance Routing Effect Algorithm for Mobility) .DREAM [174] est un protocole de routage proactif où chaque nœud maintient une table de localisation qui contient les coordonnées géographiques de toutes les destinations obtenues par un système de positionnement comme GPS. L'utilisation de ces informations de localisation permet de calculer la direction de chaque destination ainsi que la distance vers chacune d'elles. Chaque nœud broadcast des paquets contenant sa position courante dans tout le réseau, dont la fréquence du broadcast est déterminée en considérant :

- ✓ L'effet de la distance : plus importante la distance entre deux nœuds, moins rapidement ils se déplacent l'un par rapport à l'autre. De ce fait, les nœuds éloignés échangent leurs informations de localisation les uns avec les autres moins fréquemment que les nœuds qui sont proches.
- ✓ Le taux de la mobilité du nœud : un nœud qui se déplace rapidement doit informer fréquemment les autres nœuds de sa localisation et vice versa [176].

Quand un nœud veut envoyer un paquet de données vers une destination, il consulte sa table de localisation pour extraire la direction de cette dernière. Ensuite, il choisit ses sauts prochains parmi ses voisins qui se trouvent dans la même direction que le nœud destination. Cela fait que plusieurs copies d'un même paquet de données arrivent à destination, ce qui assure une meilleure fiabilité [177].

Le premier protocole réel est le Greedy Perimeter Stateless Routing (GPSR) .GPSR [178] est un Protocole de Routage Géographique réactif qui transmet les paquets au voisin le plus proche du nœud cible jusqu'à atteindre la destination (approche Gloutonne ou Greedy Forwarding). Par conséquent, il s'adapte mieux que les Protocoles de Routage Topologiques aux réseaux à larges échelles dynamiques.

Le protocole de routage géographique utilisé dans cette étude est le protocole GPSR. Cependant, ce travail reste compatible avec les autres Protocoles de Routage Géographique.

III. Services de Localisation

Les protocoles de Routage Géographique utilisent les informations de localisation des nœuds quand ils ont besoin d'acheminer des paquets. Cependant, ces informations sont

maintenues par des services distribués connus sous le nom de Services de Localisation. Par conséquent, les Protocoles de Routage Géographique et les Services de Localisation sont très liés, mais généralement sont considérés séparément dans la littérature. Les Services de Localisation (dans le contexte des réseaux de Capteurs WSNs) sont des services distribués sans infrastructure. Ils ont pour but de répondre à des requêtes de localisation telles que : « Où est le nœud X ? ».

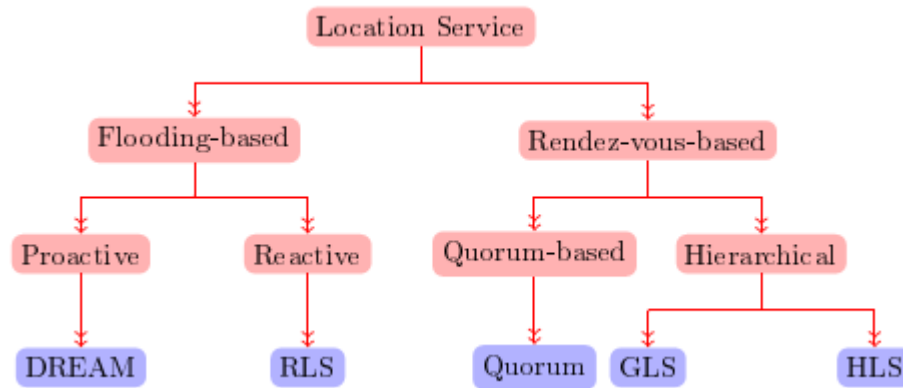


Figure 5.1. Taxonomie des services basés sur la localisation

Il existe principalement deux types de Services de Localisation selon la figure 5.1 : les services basés sur l’inondation du réseau et ceux qui sont basés sur la hiérarchie du réseau.

La première classe est composée de services réactifs et proactifs. Dans les services de localisation proactifs basés sur les inondations, chaque nœud inonde ses informations géographiques sur l'ensemble du réseau périodiquement. Ainsi, tous les nœuds peuvent mettre à jour leurs tables de localisation. Un exemple de service de localisation proactif basé sur les inondations est l'Algorithme (DREAM) [174]. Derrière DREAM il y a deux idées, la première est: si un nœud est loin, il apparaît comme s'il se déplace plus lentement qu'un voisin ayant la même vitesse. Par conséquent, la fréquence de mise à jour est réduite lorsque la distance au nœud augmente. La deuxième idée est: un nœud avec une mobilité élevée envoie plus de paquets de localisation de mise à jour. En conséquence, il y a moins de paquets qu'un schéma simple sans affecter les performances du réseau.

Pour le deuxième groupe (c.-à-d. le service de localisation réactif basé sur l’inondation), la réponse à la localisation (Location response) est envoyée après avoir reçu une demande de localisation. Cela évite les frais généraux. Un exemple parmi les Services de Localisation basé sur l’inondation est le RLS [179].

Dans ce service, chaque nœud transmet sa demande de localisation à tout le réseau grâce à l'inondation (broadcast). Une fois que cette requête a atteint un nœud ayant l'information, ce dernier répond directement.

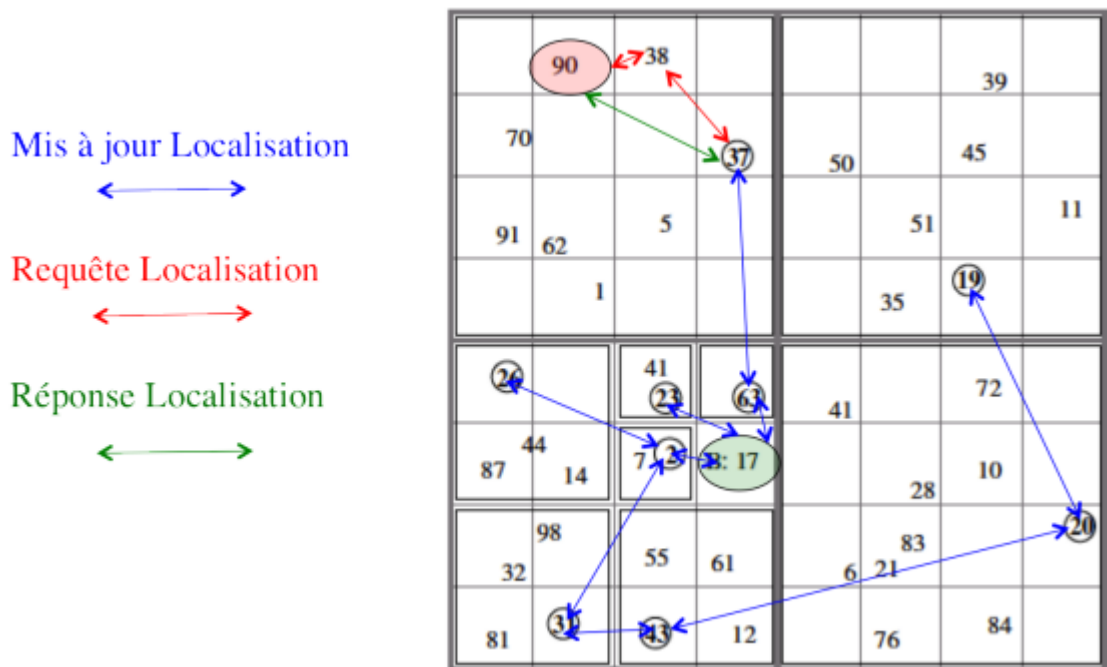
Dans la deuxième classe (c'est-à-dire Rendez-vous location-based services), tous les nœuds s'accordent sur un mappage unique d'un nœud vers d'autres nœuds spécifiques. L'information géographique est diffusée à travers certains nœuds élus appelés les serveurs de localisation. Ainsi, le service basé sur la localisation comprend deux composantes:

1. **Mises à jour de localisation (Location Updates)** : un nœud doit recruter des serveurs de localisation et, par conséquent, il a besoin de mettre à jour sa localisation via ces serveurs. Les serveurs de localisation sont responsables de stocker les données géographiques de ces nœuds.
2. **Demandes de localisation (Location Requests)**: un nœud, qui recherche l'information de localisation d'un autre nœud, diffuse une demande de localisation. Les serveurs de localisation répondent dès qu'ils reçoivent cette demande.

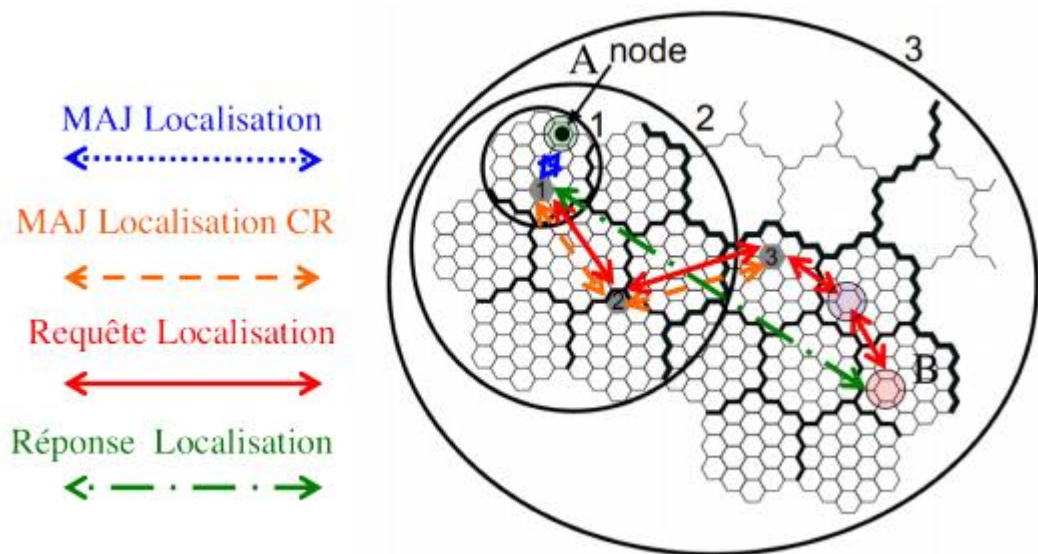
Il existe deux approches majeures pour les services de localisation (Rendez-vous location-based service). Dans l'approche basée sur le Quorum [179], la mise à jour de localisation est envoyée à un groupe de nœuds (quorum de mise à jour, Update quorum). La requête de localisation est envoyée à un autre groupe (quorum de requête, Query quorum). Ces deux groupes ne doivent pas être disjoints. Le défi est alors de choisir comment générer le système de requête tel que décrit dans [179].

La deuxième approche est l'approche hiérarchique. Le réseau est divisé en plusieurs niveaux. À chaque niveau, un nœud recrute des serveurs de localisation. La requête de localisation est transférée vers le haut et vers le bas dans la hiérarchie. Cela limite le nombre de paquets transférés et évite les inondations.

Les Services de Localisation Hiérarchiques les plus connus sont GLS [180] et Hierarchical Location Service (HLS) [181]. La région est partagée dans ces réseaux en niveaux hiérarchiques et les requêtes sont transmises du haut vers le bas dans cette hiérarchie (Figure 5.2).



a. Partition du réseau dans GLS



b. Partition du réseau dans HLS

Figure 5.2. Exemple de partition hiérarchique du réseau dans GLS et HLS

Les nœuds dans les Services de Localisation Hiérarchiques doivent élire des serveurs de localisation dans les différents niveaux afin de maintenir à jour leurs positions auprès de ces serveurs. Quand un autre nœud a besoin de cette position, il envoie une requête vers ces serveurs qui répondront après l'avoir reçu.

Un exemple de mises à jour et de requêtes de localisation dans GLS est représenté par la Figure 5.2 (a). GLS subdivise la région en quatre carrés qui sont divisés en quatre carrés plus petits et ainsi de suite. Les quatre plus petits carrés sont appelés le niveau de l'ordre 1 dans la hiérarchie. Cette hiérarchie est la même pour tous les nœuds. Figure 5.2(a) montre une hiérarchie à trois niveaux. À chaque carré dans n'importe quel niveau, un nœud sélectionne un serveur de localisation avec le plus petit ID supérieur à l'ID du nœud (s'il existe). Dans notre exemple, le nœud B avec ID :17 choisit les nœuds avec ID :23 et 63 à l'ordre-1; les nœuds avec ID :26, 31 et 43 à l'ordre-2; et les nœuds avec ID :19, 20 et 37 à l'ordre-3; en tant que serveurs de localisation. En connaissant ses serveurs de localisation, un nœud partage ses informations de localisation via des messages de mise à jour de localisation. Lorsque le nœud 90, par exemple, doit envoyer des données au nœud B, il diffuse une requête de localisation dans l'ordre 1 si le nœud ciblé n'est pas dans le même carré. L'emplacement est transmis dans l'ordre suivant au nœud avec le plus petit ID supérieur à l'ID du nœud ciblé (c.-à-d. 17) jusqu'à atteindre un serveur de localisation (dans notre exemple le nœud 37), qui répond avec un paquet de réponse de localisation indiquant les informations de localisation reçues de la dernière mise à jour.

HLS divise également le réseau en régions, qui sont subdivisées en cellules hexagonales. Les régions voisines sont regroupées en niveaux régionaux. Cette partition est définie et connue par tous les nœuds participants. Un exemple de partition de réseau dans trois niveaux régionaux est illustré à la Figure 5.2 (b). Les dimensions de la cellule doivent être inférieures à la transmission de la portée. Ainsi, un nœud peut être capable de diffuser un message à tous les nœuds dans la même cellule. Contrairement à GLS, une fonction de hachage dépend non seulement de l'ID mais aussi de la position actuelle du nœud est utilisée pour choisir la cellule responsable (la cellule où un nœud doit sélectionner ses serveurs de localisation) à chaque niveau de région. Il existe deux méthodes de mise à jour différentes:

1. **La méthode directe:** pour mettre à jour ses informations de localisation, un nœud envoie souvent des paquets à la cellule responsable au premier niveau. Une cellule responsable peut contenir plus d'un serveur de localisation. Ce n'est le cas que pour les serveurs de localisation voisins (à la même région que le niveau). Pour les autres serveurs de localisation, la deuxième méthode est utilisée.
2. **La méthode indirecte:** pour réduire le trafic sur le réseau, seul l'emplacement de la cellule responsable du niveau N-1 est envoyé à la région de niveau N. Au lieu d'envoyer les informations géographiques du nœud, les coordonnées de la cellule responsable sont envoyées lorsque le nœud se déplace d'une région à l'autre. Par

conséquent, la congestion du trafic générée par le mouvement du nœud est locale au premier niveau et quelques paquets longue distance multi-hop sont envoyés aux niveaux supérieurs.

Si un nœud B, par exemple, doit envoyer des données vers un nœud A, B diffuse un emplacement de requête. Étant donné que cet emplacement atteint la cellule responsable de A, et si A et B sont dans le même niveau de région, un serveur de localisation peut envoyer directement une réponse de localisation. Si elles ne sont pas dans le même niveau de région, le serveur de localisation transmet le paquet de demande à la cellule responsable de A dans le niveau inférieur et ainsi de suite jusqu'à ce que la cellule responsable du premier niveau soit atteinte. Ce dernier génère un paquet de réponse et l'envoie en mode unicast au nœud B. En connaissant la position du nœud A, B pourra envoyer des données en utilisant n'importe quel protocole de routage géographique.

La différence remarquable entre GLS et HLS illustrée par la Figure 5.2, se fait au niveau de la mise à jour. Dans GLS, chaque nœud envoie une mise à jour périodique de sa position vers tous les serveurs de localisation à tous les niveaux. Alors que dans HLS, cette mise à jour se fait qu'au premier niveau et que seule la mise à jour de la position de la cellule responsable (la cellule où l'élection des serveurs de localisation doit se faire) est envoyée du niveau i vers la cellule responsable du niveau $i+1$.

Plusieurs expérimentations ont été effectuées sur le simulateur réseau NS-2. Ces expérimentations montrent que les combinaisons efficaces entre les protocoles de routage géographiques et les services de localisation permettent d'améliorer les performances du réseau tout en réduisant les coûts de localisation.

IV. Combinaison entre le protocole GPSR et les services de localisation

4.1. Description de l'approche utilisée

Afin de réduire le coût de localisation et d'améliorer les performances de routage, nous combinons GLS et HLS basés sur la localisation avec le protocole de routage géographique GPSR, ensuite, nous comparons les coûts générés par le service basé sur la localisation (nombre de demandes de localisation envoyées, bande passante consommée par la localisation, etc.) avec la combinaison. En outre, nous comparons les performances de routage (Taux de Paquets Délivrés, la latence, nombre de sauts, etc.).

Dans le nouvel algorithme, si un nœud doit envoyer des données à un autre nœud du réseau, GPSR demande au service de localisation s'il possède une information fraîche sur

l'emplacement de la destination. Si c'est le cas, GPSR utilise cette position pour transmettre les paquets à la destination. Autrement, le service de localisation lance une demande de localisation pour trouver la nouvelle position de la destination. Lorsque la réponse de localisation est reçue, le service de localisation informe GPSR, qui sera en mesure de transmettre les données vers le nœud destination en utilisant la position reçue. L'algorithme de localisation du GLS et HLS est présenté sur la figure 5.3.

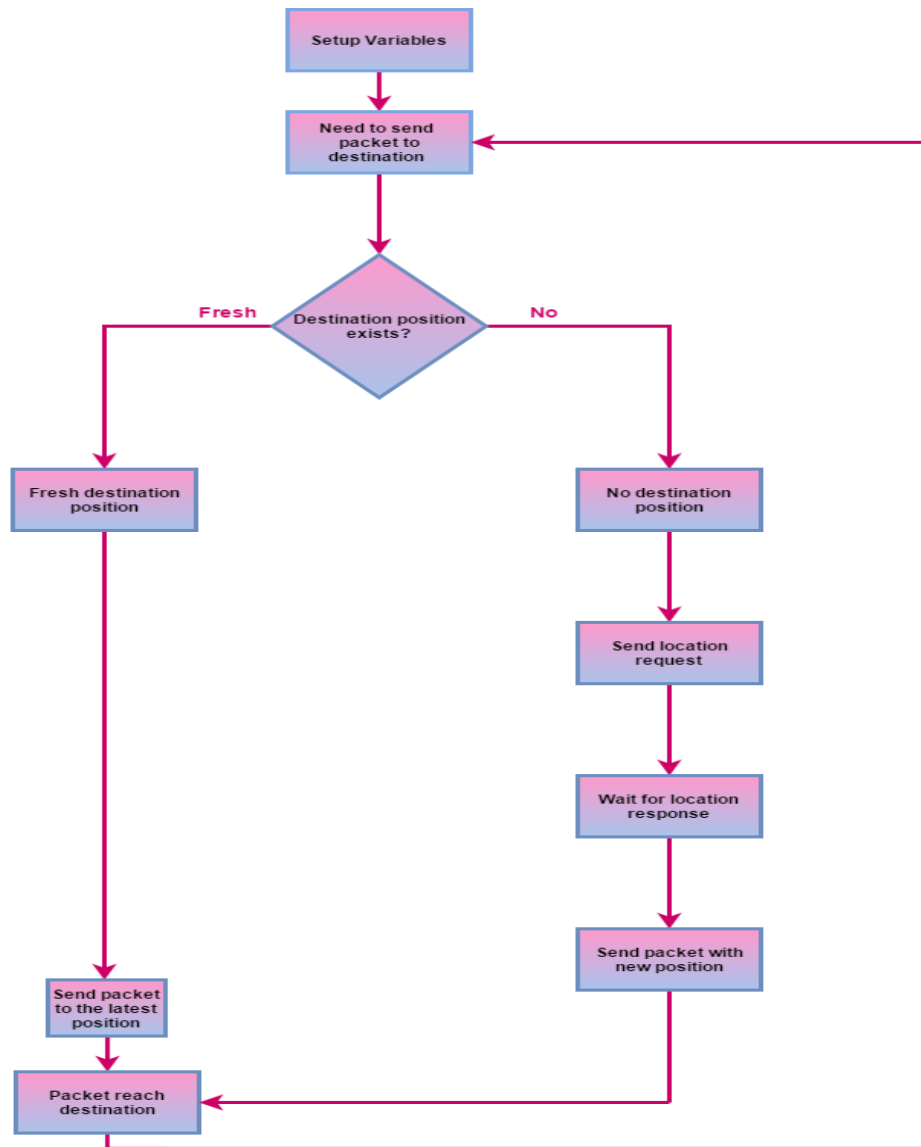


Figure 5.3. Présentation schématique d'algorithme de Combinaison

4.2. Algorithme

La première procédure modifiée dans le service basé sur la localisation (GLS ou HLS) est la procédure FindPosition (définie dans la ligne 8 de l'Algorithme 1). Il gère l'interrogation de la position de la destination, il se trouve dans la mémoire cache locale du nœud actuel et met à jour les informations du paquet avec la position de destination.

Algorithm 1 GLS or HLS :: FindPosition

```
1: cacheThreshold ← LocationCacheMaxAge;
2: Struct Position{
3: int T; .                               Time Stamp
4: int id; .                               Node ID
5: double x ,y                             Position
6: }Position
7: enum freshness {NoPosition = 0, Fresh = 1};
8: procedure FindPosition(packetToSend)
9: if (! Location information) then
10: freshness ← NoPosition;
11: PreparePacket(packetToSend);
12: else
13: if (Destination Location age < cacheThreshold) then
14: freshness ← Fresh;
15: PreparePacket(packetToSend);
16: end if
17: end if
18: end procedure                               End of FindPosition
```

La procédure GPSREmit (définie dans la ligne 1 de l'Algorithme 2) dans le protocole de routage géographique GPSR gère la création et l'envoi de nouveaux paquets, il vérifie d'abord si l'émetteur a des informations fraîches sur la position de destination puis commence le routage des paquets. S'il contient des informations de localisation, la fonction lance une nouvelle requête de position et insère le paquet dans un tampon pendant que la requête se déroule.

La fonction forwardPacket (définie dans la ligne 11 de l'Algorithme 2) gère le renvoi des paquets, elle est appelée à chaque fois qu'un paquet atteint un nœud intermédiaire, il vérifie si ce nœud a une position plus fraîche de la cible et peut éventuellement mettre à jour les informations sur les paquets avec ce nœud. Sinon, une nouvelle requête est lancée pour récupérer la nouvelle position exacte de la cible.

Algorithme 2: Geographic Routing Protocol GPSR

```
1: procedure GPSREmit(packetToSend)
2: PreparePacket(packetToSend);
3: FindPosition(packetToSend);
4: if (freshness == Fresh) then
5: ForwardPacket(packetToSend);
6: else
7: LaunchPositionQuery(destination);
8: StickToBuffer(packetToSend);
9: end if
10: end procedure .                               End of GPSREmit
11: procedure forwardPacket(packetToSend)
12: if (freshness == Fresh) then
13: ChooseNextBestHop(packetToSend);
14: ForwardNextHop(packetToSend);
15: else
16: if (Current Node freshness == Fresh) then
17: UpdatePacketDestination(packetToSend);
18: ChooseNextBestHop(packetToSend);
19: ForwardNextHop(packetToSend);
20: else
21: if (Current Node Region == Destination Node Region) then
22: LaunchPositionQuery(destination);
23: StickToBuffer(packetToSend);
24: else
25: ChooseNextBestHop(packetToSend);
26: ForwardNextHop(packetToSend);
27: end if
28: end if
29: end if
30: end procedure                               End of forwardPacket
```

V. Environnement de travail pour les expérimentations

Les simulations ont été réalisées à l'aide du Simulateur NS-2 2.35 [182]. Le protocole de routage géographique utilisé est le Greedy Perimeter Stateless Routing (GPSR) [178]. La région choisie est une surface de $2*2 \text{ km}^2$. La topologie proposée est présentée sur la figure 5.4. La couche MAC utilisée est l'implémentation NS-2 du IEEE 802.15.4. La simulation a été lancée 10 fois et les résultats sont calculés sur la moyenne pour plus de précisions. Les paramètres utilisés dans les simulations sous NS-2 sont résumés dans le Tableau 5.1.

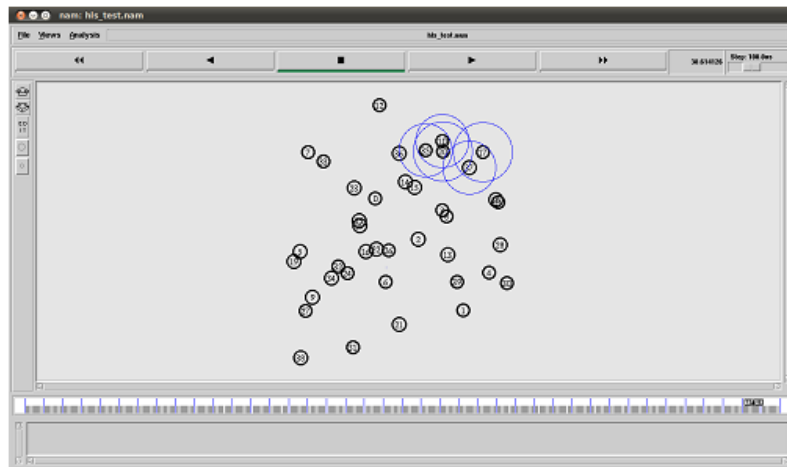


Figure 5.4. Fichier NAM généré du réseau proposé

Pour chaque simulation, plusieurs requêtes de test aléatoire sont lancées. Une requête de test est le nœud à la recherche d'une information de localisation d'un autre nœud. La requête peut être répondue par un cache dynamique ou envoyée comme une requête de localisation, qui peut être répondue par le nœud ciblé ou un serveur de localisation. Afin d'éviter les effets secondaires des échanges du paquet d'application, juste les requêtes de test sont échangées au cours des simulations.

VI. Résultats expérimentaux

Les expérimentations ont fourni des différents résultats. Pour chaque résultat, le comportement de réseau avec HLS et GLS est présenté.

Six critères ont été pris en compte lors du processus de simulation:

✓ Performance de localisation

Ce critère est estimé par le Taux de Requêtes Réussites (TRR) présenté dans les figures (5.5-5.7) et le Temps de déplacement de requête (RTT) présenté dans les figures 5.8 et 5.9.

Paramètres	Valeurs
Channel type	Channel/WirelessChannel
Propagation model	Propagation/TwoRayGround
Network interface	Phy/WirelessPhyExt
MAC layer	802.15.4
Interface queue type	Queue/DropTail/PriQueue
Link layer	LL
Antenna model	Antenna/OmniAntenna
Interface queue length	512 packets
Ad-hoc routing protocol	GPSR
Location-based service	GLS or HLS
Location cache maximum age	4,8,12,16 and 22s
Number of nodes	20,40,50,60,80,100,120,140 and 160
Simulation time	300s
GPSR beacon interval	0.5s
Average speed	21.3m/s
CBR traffic	4*20 packets/node
CBR packet size	128 B
CBR send interval	1s

Tableau 5.1.Paramètres de simulation NS2

Le TRR représente le taux de requêtes qui sont répondues avec des informations de localisation précises (une marge de 250m, de la taille de la portée de transmission, est autorisée). En fait, la requête de localisation doit être non seulement répondue positivement, mais elle doit aussi être reçue par le nœud cible dès possible afin d'être rapidement traitée (à savoir répondue).En effet, les réponses dans les approches GLS et HLS sont transmises de la même manière en utilisant l'approche greedy forwarding du GPSR. Ainsi, le RTT est considéré comme un indicateur efficace du temps de réponse. Les requêtes les plus rapides sont reçues, les réponses les plus rapides sont retournées.

✓ **Coût de localisation**

Le coût est mesuré par le nombre de paquets envoyés et transmis au cours des mises à jour de localisation, des requêtes et des réponses (Figure 5.10).

✓ **Fraîcheur des informations de localisation**

Probablement, le paramètre le plus crucial et important à étudier est l'Age Maximal des Informations de Localisation dans le Cache (AMILC). Ce paramètre définit la limite d'âge au cours de laquelle les informations de localisation sont considérées comme fraîches et peuvent être utilisées lors de l'acheminement des paquets. Au-delà, une demande de localisation est lancée pour connaître une position plus fraîche de la destination. En d'autres termes, il définit la fraîcheur de la position sauvegardée dans la mémoire cache. Dans ce contexte, nous avons réalisé des expérimentations dans lesquels nous avons fait varier l'AMILC. Les résultats sont présentés dans les figures (5.11-5.14).

✓ **Statistiques de localisation**

Cela inclut toutes les statistiques que nous avons pu extraire des simulations concernant la localisation tels que l'accès aux informations de localisation dans le cache, l'âge moyen du cache, le nombre des requêtes et des mises à jour envoyées (figures 5.15 et 5.16).

✓ **Coût de localisation et routage**

Le coût est représenté comme la bande passante de localisation consommée dans la couche MAC afin d'envoyer des mises à jour de localisation, transmettre les requêtes de localisation et répondre par des réponses de localisation (figure 5.17).

✓ **Performance de routage**

Le service de localisation est considéré ainsi comme un support pour le protocole de routage géographique (GPSR ici). Prendre des décisions de routage dépend alors directement des informations de localisation maintenues par le service de localisation. Par conséquent, le suivi des performances du routage est aussi important que les performances de localisation. La performance de routage est évaluée par le Taux de Paquets Délivrés (TPD) illustré par la figure 5.18(a). Le TPD est le pourcentage des paquets CBR reçus parmi tous ceux envoyés. Le deuxième indicateur des performances du routage représenté par la Figure 5.18(b) est la latence moyenne. La Latence des paquets CBR est évaluée comme étant la différence entre l'heure d'arrivée du paquet CBR vers la destination et le moment où le paquet est envoyé depuis la source.

6.1. Résultats de performance de localisation

6.1.1. Taux de Requêtes Réussites ou Répondues (TRR)

La figure 5.5 représente le QSR ou le Taux de Requêtes Réussites (TRR). Comme le montre la figure 5.5, HLS performe plus de quatre fois mieux que GLS. Ceci montre que le mécanisme qui maintient les informations de localisation dans les cellules responsables fonctionne bien. Le faible taux réussis de GLS est partiellement affecté par sa mise en cache agressive. Les informations de localisation par le cache deviennent rapidement non précises dans un réseau très dynamique.

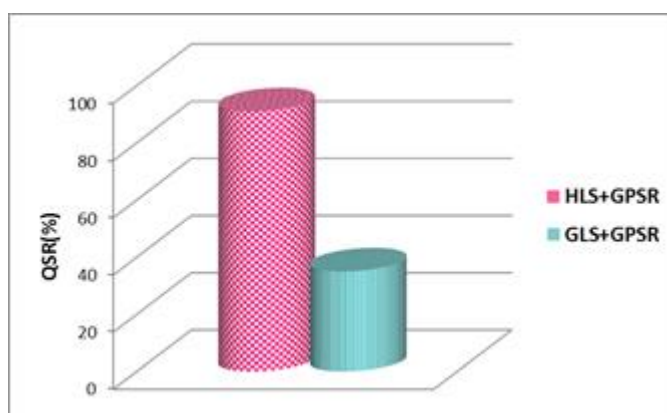


Figure 5.5. Résultats de performance de localisation : Taux de Requêtes Réussites

6.1.2. Taux de Requêtes Réussites (TRR) par rapport au nombre de nœuds

Par la suite, on fait varier le nombre de nœuds (50, 100, 200 et 400 nœuds). L'objectif est d'observer son impact sur les performances des services de localisation. Les résultats sont présentés dans la figure 5.6. La courbe représente le QSR ou le Taux de Requêtes Réussites (TRR). Ce paramètre représente le taux de requêtes qui sont répondues avec des informations de localisation précises (une marge de 250m, de la taille de la portée de transmission, est autorisée). Le TRR diminue en raison du nombre de nœuds élevé, donc le nombre de requêtes envoyées augmente en conséquence. Ceci surcharge le réseau, surtout si les mises à jour de localisation sont fréquentes comme dans GLS. La diminution du TRR est seulement d'environ 11% pour HLS, lorsque le nombre de nœuds augmente de 50 à 400 nœuds, alors qu'il diminue de 23% pour GLS.

La figure 5.7 représente le QSR obtenu. En augmentant le nombre de nœuds le QSR diminue, donc le nombre de demandes envoyées augmente par conséquent. Le QSR en HLS saute d'environ 25% en moyenne pour atteindre plus de 98% par rapport à GLS.

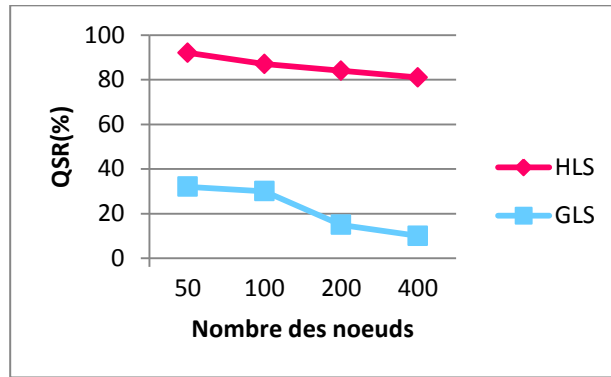


Figure 5.6 .Résultats de performance de localisation : Taux de Requêtes Réussites (TRR)

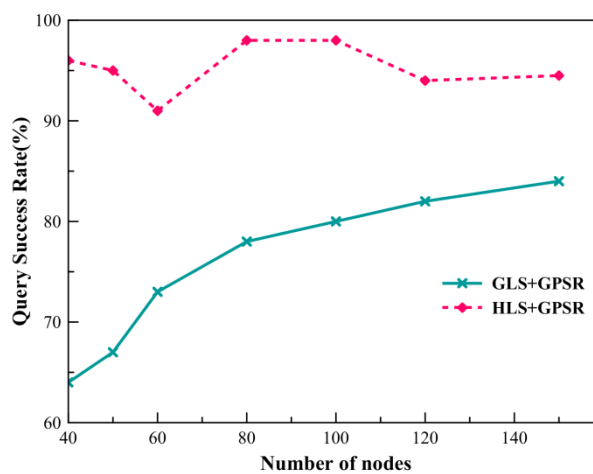


Figure 5.7. Résultats de performance de localisation: Taux de requêtes réussites

6.1.3. Temps de déplacement de requête : Request Travel Time(RTT)

Une autre caractéristique importante dans les services de localisation est le temps de déplacement de requête (Request Travel Time(RTT)) montré sur la figure 5.8. En fait, la requête de localisation doit être non seulement répondue positivement, mais elle doit aussi être reçue par le nœud cible dès que possible afin d'être rapidement traitée (à savoir répondue).

En effet, les réponses dans les approches GLS et HLS sont transmises de la même manière en utilisant l'approche greedy forwarding du GPSR. Ainsi, le RTT est considéré comme un indicateur efficace du temps de réponse. Les requêtes les plus rapides sont reçues, les réponses les plus rapides sont retournées.

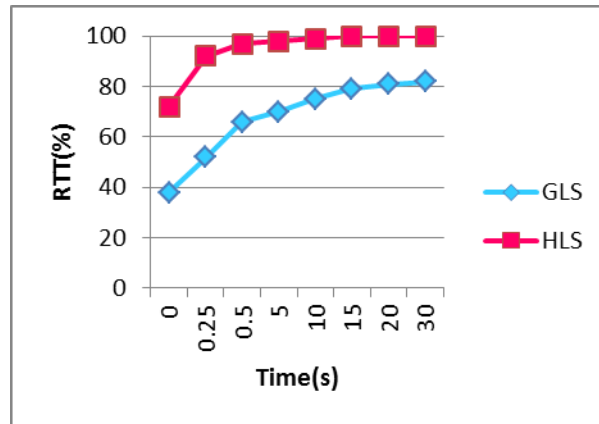


Figure 5.8. Résultats de performance de localisation : Request Travel Time (RTT)

La Figure 5.8 montre que lorsqu'on utilise le GLS, seulement 68% des requêtes arrivent au nœud cible après une demi-seconde de leur envoi tandis que la plupart d'entre eux (97%) sont reçues pendant le même intervalle de temps pour HLS. En outre, plus de 15% de requêtes GLS arrivent après 15s contre 2% en HLS. Par conséquent, le mécanisme de la cellule responsable dans le HLS aide dans la transmission des requêtes avec de faibles latences.

Le transfert des requêtes à travers des régions en HLS prend moins de temps que de les envoyer par emplacement serveurs à chaque niveau de GLS.

Le RTT avec 100 nœuds est représenté sur la figure 5.9. Il montre que HLS a les demandes les plus rapides suivi par GLS. Ainsi, le mécanisme de la cellule responsable en HLS offre des faibles latences dans la livraison de la requête. Pour transporter des demandes à travers des régions, HLS prend moins de temps que de les envoyer à travers des serveurs de localisation à chaque niveau de GLS.

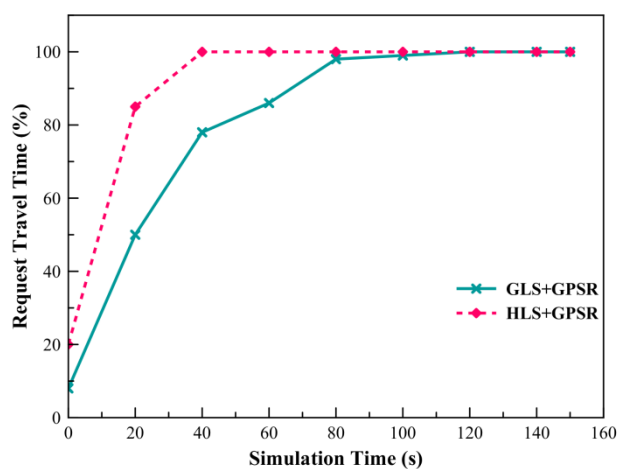


Figure 5.9. Résultats de performance de localisation: Request Travel Time (100 nœuds)

On conclut donc que le mécanisme basé sur la hiérarchie dans les services de localisation est mieux adapté aux larges réseaux dynamiques. En plus de son extensibilité, HLS est plus rapide et plus performant que GLS.

6.2. Résultats des Coûts des Services de Localisation

Nous avons étudié le coût induit par les services de localisation en ce qui concerne les requêtes, la mise à jour et les réponses (Figure 5.10). GLS a un mécanisme de maintenance beaucoup plus coûteux que celui de HLS (+326% de paquets de mise à jour envoyés dans GLS par rapport à HLS).

Cela est dû au fait que la mise à jour de localisation dans GLS est globale pour tous les serveurs dans le réseau. Cependant, elle est principalement locale dans HLS à la cellule responsable du premier niveau. Par conséquent, nous nous attendions à avoir plus de mises à jour dans GLS que dans HLS, ce qui est le cas dans la Figure 5.10. Donc, HLS réduit cette surcharge de 56% par rapport à GLS en moyenne.

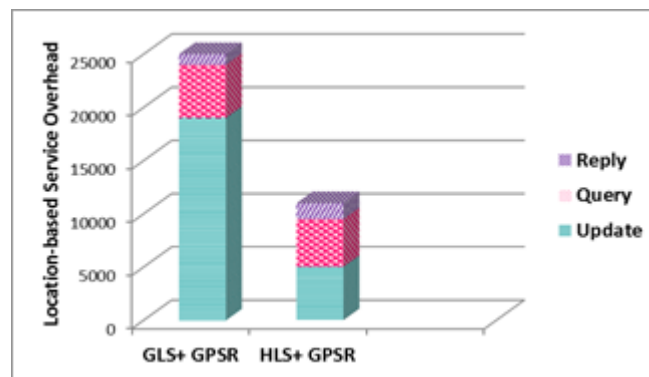


Figure 5.10. Résultats du coût de localisation : Coûts des Services de Localisation

6.3. Impact de la fraîcheur des informations de localisation

6.3.1. Impact sur le coût de localisation

La figure 5.11 montre que lorsque l'AMILC augmente, le nombre de requêtes de localisation envoyées diminue et le système HLS présente les plus faibles valeurs que le système GLS. On observe que lorsque l'AMILC est inférieure à 16s, le nombre de requêtes de localisation envoyées dans le système HLS est plus faible que dans le système GLS. Dans les autres situations, c'est presque pareil. Ceci est dû au fait que le HLS provoque moins de requêtes de localisation.

La figure 5.12 montre que le coût moyen dans le système HLS est plus faible que celui du GLS.

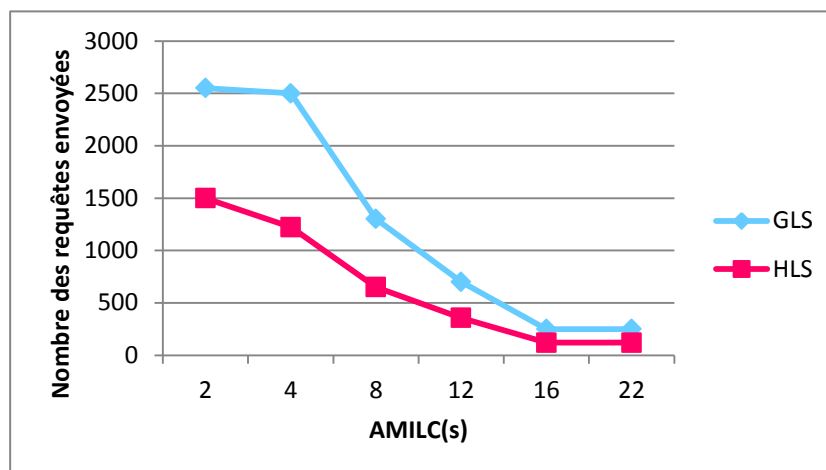


Figure 5.11. Impact de la fraîcheur des informations de localisation sur les requêtes

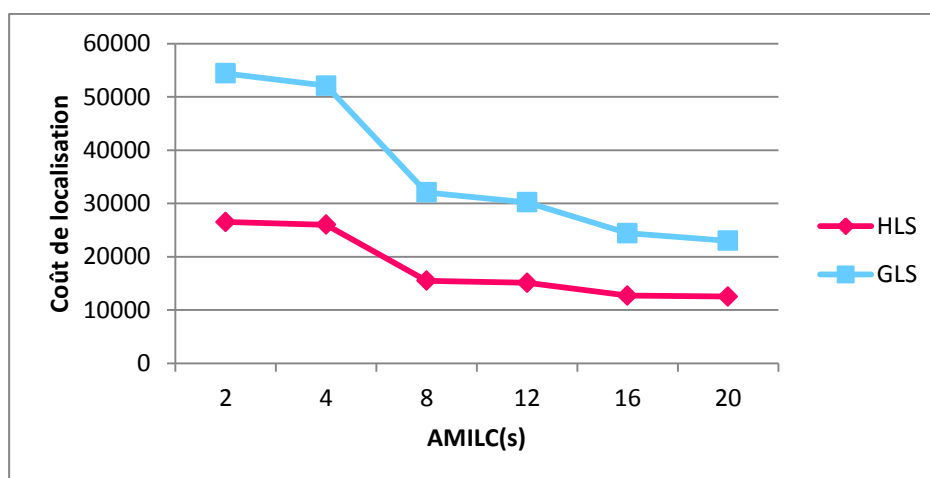


Figure 5.12 . Impact de la fraîcheur des informations de localisation sur le coût

6.3.2. Impact sur les performances de routage

La figure 5.13 présente le taux de paquets (TPD). Le TPD est défini comme le rapport entre les paquets CBR reçus et émis. Cette figure montre que le TPD dans le HLS est supérieur au TPD du GLS. En outre, quelque soit la valeur d'AMILC, le TPD est toujours amélioré.

La latence est mesurée comme le temps entre le moment de l'expédition et le moment de la réception de paquets CBR. Dans la figure 5.14, on observe que lorsque la valeur AMILC est inférieure à 12s, la latence moyenne dans le schéma HLS est inférieure à la latence moyenne du schéma GLS. Dans les autres situations, c'est presque la même chose. Cela signifie que les paquets atteignent plus vite leur destination avec une valeur AMILC inférieure à 12s.

On conclut enfin que les résultats observés sur le système HLS sont meilleurs que le schéma GLS, surtout lorsque l'AMILC est inférieure à 12s. Cela est dû au fait que les requêtes de localisation HLS ne sont pas lancés de manière cohérente, mais le paquet est envoyé directement avec la position la plus fraîche. Cela réduit les requêtes de localisation et le coût de localisation sans affecter les performances du réseau comme la latence et le taux de paquets délivrés, au contraire, il les améliore même. Cependant, lorsque la valeur AMILC est supérieure à 16 secondes, il n'y a pas de différence majeure entre les deux schémas. C'est principalement parce qu'il n'y a pas assez d'informations de localisation pour faire la différence. Par conséquent, l'AMILC a été fixée à 8s tout au long des expériences.

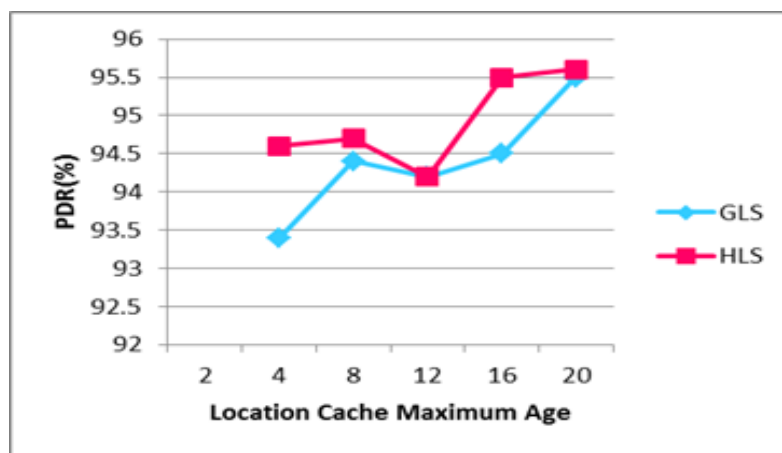


Figure 5.13 .Impact de la fraîcheur des informations de localisation sur le TPD

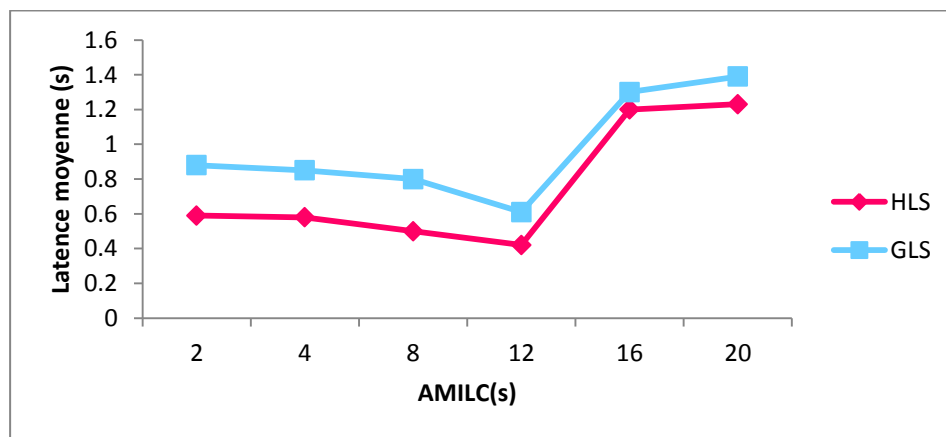


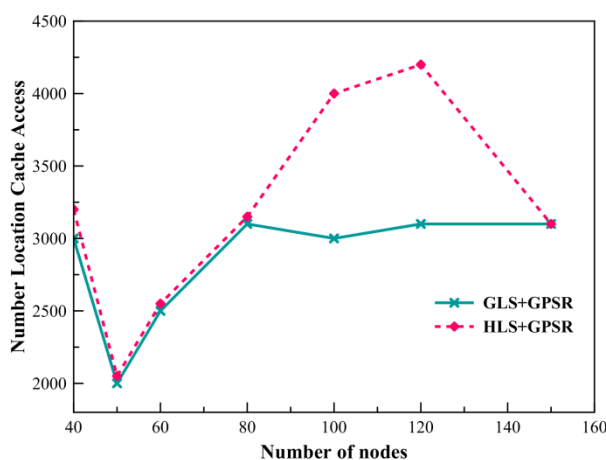
Figure 5.14. Impact de la fraîcheur des informations de localisation sur la latence

6.4. Résultats des Statistiques de localisation

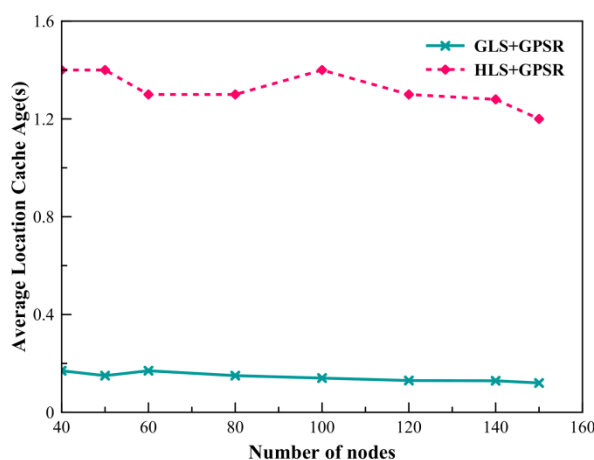
La figure 5.15 (a) montre le nombre d'accès à la mémoire de localisation dans le cache .Le HLS utilise plus de mémoire de cache en comparaison avec GLS. En conséquence directe, l'âge moyen du cache de localisation des entrées augmente de 0.17s pour GLS par rapport à 1.4s pour le HLS comme représenté dans la figure 5.15 (b).

La figure 5.16 (a) montre que le HLS envoie plus de requêtes par rapport au GLS. En outre, il y a moins de la moitié des requêtes de localisation en GLS que dans HLS.

Un autre résultat, moins prévisible, est le nombre de mises à jour de localisation très bas présenté par HLS par rapport à GLS (Figure 5.16(b)). En fait, le HLS se concentre sur la réduction des requêtes lancées avant le routage des paquets. Cependant, la figure 5.16 (b) révèle que ce mécanisme aussi réduit les mises à jour de localisation envoyées en particulier lorsque le nombre de nœuds augmente.



a. Location Cache Lookups Access



b. Location Cache Average

Figure 5.15. Résultats des Statistiques de localisation dans le cache

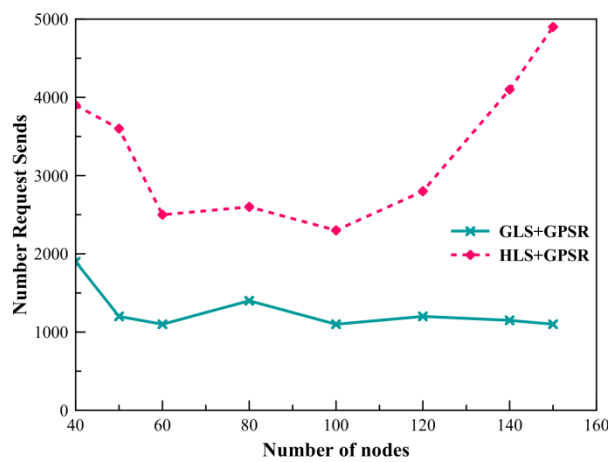
6.5. Résultats du coût de localisation

Comme les requêtes de localisation baissent, la bande passante consommée pour la localisation dans les couches MAC et de routage est réduite. Les Figure 5.17(a) et 5.17(b)

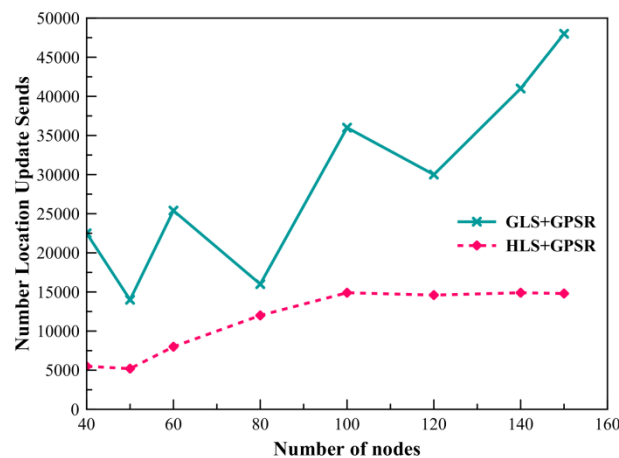
montrent que le service de localisation qui consomme en moyenne plus de bande passante est GLS, puis HLS.

6.6. Résultats des Performances du Routage

La Figure 5.18(a) illustre le Taux de Paquets Délivrés (TPD). On remarque que le HLS présente un taux élevé par rapport à GLS mais le TPD est généralement au-dessus de 80%. La seule exception est lorsque le nombre de nœuds augmente considérablement, en raison de la congestion du réseau.



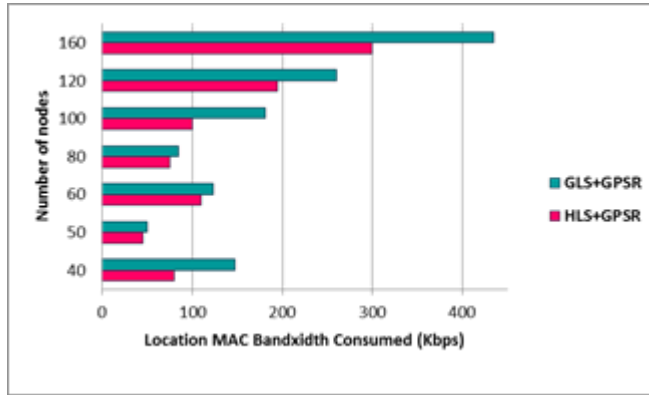
a. Location Request Sends Number



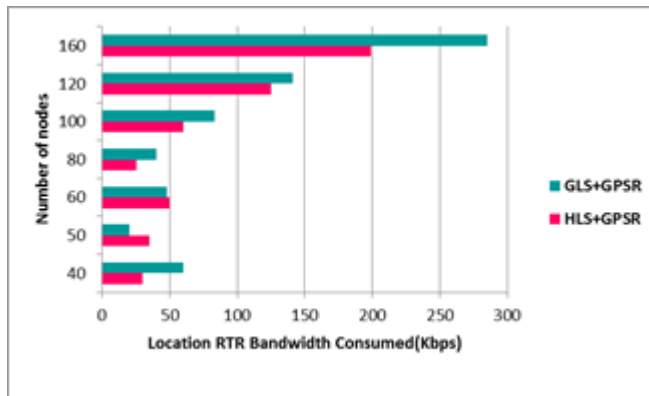
b. Location Update Sends Number

Figure 5.16. Résultats des Statistiques des Paquets de localisation

Le deuxième indicateur des performances du routage représenté par la Figure 5.18(b) est la latence moyenne. Ce temps de latence moyen est plus faible dans HLS alors qu'il est maximal pour GLS.

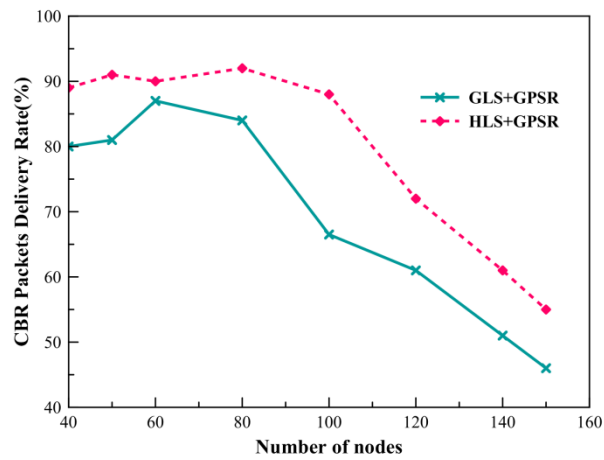


a. Bande passante MAC consommée pour la localisation

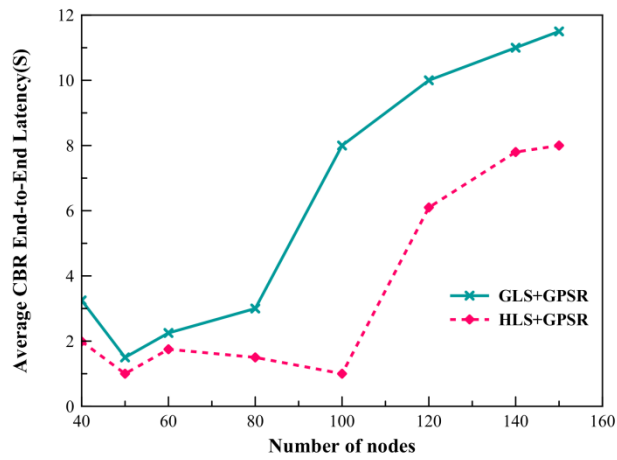


b. Bande passante du routage consommée pour la localisation

Figure 5.17. Coût de localisation: Bande passante consommée pour la localisation



a. CBR Taux de Paquets Délivrés (TPD)



b. Latence Moyenne

Figure 5.18. Statistiques de routage

VII. Conclusion

En conclusion, on peut remarquer que la combinaison entre le service de localisation et le protocole de routage permet de réduire les coûts de localisation et d'augmenter non seulement les performances de localisation, mais aussi les performances du routage. Un autre résultat intéressant de l'étude de simulation est que le système HLS combiné au GPSR présente les meilleures performances par rapport au service GLS.

Approche optimisée pour prolonger la durée de vie du réseau WSN

I. Introduction

Au cours des dernières années, une nouvelle vague de communications sans fil appelée, Wireless Sensor Network (WSN) a attiré beaucoup d'attention des chercheurs des communautés universitaires et industrielles. Un WSN se compose d'une collection de nœuds de capteurs et d'une station de base (BS) connectée par des canaux sans fil avec détection, communications sans fil et capacités de calcul. En raison de la nature adhoc de WSN, il peut être utilisé pour de nombreuses applications telles que l'application militaire, le contrôle du trafic aérien, l'observation de la circulation, la sécurité physique, la surveillance vidéo, l'automatisation industrielle et de fabrication, la surveillance de l'environnement, la surveillance du bâtiment, la surveillance des hôpitaux et des soins de santé [183] [184], [185], [186], [187]. Les avantages de WSN sont la facilité du déploiement, le faible coût d'installation, la distribution sur une vaste région et une tolérance de panne élevée [188]. Un WSN est déployé pour collecter des données environnementales dans la région d'intérêt (ROI) et pour l'envoyer à une BS. Dans un WSN, la façon dont les nœuds seront déployés, est spécifique à l'application et dépend totalement de l'environnement. L'option de déploiement de nœud affecte les performances du protocole de routage en termes de consommations d'énergie. Il existe trois façons de déployer de petits nœuds de capteurs dans un environnement WSN: Le déploiement régulier- les nœuds de capteurs peuvent être déployés de manière fixe et les données sont acheminées par un chemin prédéfini. Ce déploiement est utilisé dans le secteur médical, industriel, les réseaux domestiques, etc. Déploiement aléatoire - Les nœuds de capteurs se déploient sur une surface finie de manière dispersée. Ce déploiement est généralement utilisé dans les opérations de sauvetage, comme la surveillance environnementale et habituelle. Nœuds de capteurs avec mobilité - le nœud peut se déplacer pour compenser les défauts de déploiement; peut-être passivement déplacé

par une force externe (vent, eau et véhicule). Ce déploiement est utilisé dans les surveillances de terrain de combat, les situations d'urgence (incendie, volcan et tsunami), etc. Dans les catégories ci-dessus, une fois que les nœuds de capteurs sont déployés, le WSN sera en mesure d'effectuer un travail spécifique. Dans WSN, l'énergie est un grand défi car les nœuds sont alimentés par des batteries. Cette énergie peut être très coûteuse, difficile ou même impossible à renouveler. Ainsi, économiser de l'énergie pour maximiser la durée de vie du réseau est l'un des problèmes critiques dans les WSN.

Dans les WSN, les nœuds dissipent l'énergie dans le traitement et la transmission / réception de messages. En plus de cette énergie, il existe une grande quantité d'énergie nécessaire dans les états qui sont inutiles du point de vue de l'application, tels que: l'écoute, les interférences et la collision. Une autre contrainte qui réduit l'efficacité des nœuds déployés est la redondance des données car les nœuds dans la plupart des cas sont fortement déployés et cela provoque des données redondantes à partir des nœuds voisins de la région. Sur la base de ces contraintes énergétiques des nœuds sans fil, de nombreuses techniques énergétiques sont conçues pour minimiser la consommation d'énergie et maximiser la durée de vie du réseau [189]. L'une des techniques intéressantes est le routage hiérarchique, qui introduit le concept de création de cluster et l'attribution de tâches spéciales au nœud capteur sélectionné dans le cluster appelé chef de groupe ou cluster-head (CH). Les chercheurs ont convenu que le regroupement de nœuds dans un WSN est un programme efficace de conservation de l'énergie [190]. Le regroupement est défini comme le regroupement d'objets ou de données similaires [191]. Dans WSN, il est utilisé pour minimiser le nombre de nœuds qui participent à la transmission de données longue distance à une BS, ce qui entraîne une réduction de la consommation totale d'énergie du système [192]. Le regroupement réduit la quantité de données transmises en regroupant des nœuds similaires et en élisant un nœud en tant que CH, où l'agrégation des données est effectuée pour éviter la redondance et la charge de communication causée par plusieurs nœuds adjacents, puis envoyer les données agrégées au CH suivant ou à la BS, où il est traité, stocké et récupéré. La principale cible du routage hiérarchique ou du routage basé sur un cluster est de maintenir efficacement l'utilisation d'énergie des nœuds capteurs en les impliquant dans une communication multi-hop dans un cluster particulier. Le premier protocole hiérarchique est la hiérarchie de regroupement adaptatif à faible énergie (LEACH) qui a été introduite dans [193]. De nombreux protocoles hiérarchiques ont émergé de LEACH.

Ce chapitre est organisé comme suit. La section 2 présente les concepts de base du protocole de routage hiérarchique en présentant des détails sur le protocole LEACH et ses

différents descendants en donnant leur calendrier de protocoles descendants. La section 3 montre le modèle réseau de la consommation d'énergie dans les réseaux de capteurs. La section 4 décrit notre protocole LEACH-C amélioré. La section 5 présente les résultats des simulations. Enfin, la section 6 conclut ce travail.

II. Protocoles de routage hiérarchique

L'objectif de tout protocole de routage est d'acheminer les informations d'une source vers une destination d'une manière efficace [194]. La contrainte majeure du routage dans les WSN est d'acheminer les informations capturées en minimisant la consommation d'énergie des nœuds. Plusieurs protocoles de routage ont été proposés dans la littérature. L'approche utilisée par les protocoles hiérarchiques basés sur les clusters permet de regrouper les nœuds du réseau en clusters (groupes). Un nœud est élu cluster-head ou chef de groupe. Son rôle est de regrouper les données capturées par ses nœuds membres et les envoyer à la station de base (Figure 6.1). LEACH, TEEN [194] sont des exemples de protocoles hiérarchiques basés sur les clusters.

Dans les WSN, la consommation d'énergie est l'une des questions les plus importantes. Les protocoles de routage traditionnels pour les WSN peuvent ne pas être optimaux en termes de consommation d'énergie. Les protocoles de routage hiérarchique (PRH) sont plus efficaces en énergie que d'autres protocoles [194]. Ils suivent les mécanismes de regroupement; les techniques de regroupement peuvent être efficaces en termes d'énergie et d'évolutivité [195]. En utilisant une technique de regroupement, la consommation d'énergie est considérablement minimisée dans la collecte et la diffusion (fusion et agrégation) des données. PRH minimisent la consommation d'énergie en divisant les nœuds en différents clusters. Dans chaque cluster, des nœuds d'énergie plus élevés, c'est-à-dire les CH peuvent être utilisés pour traiter et envoyer l'information à la BS alors que les nœuds à faible énergie, c'est-à-dire les membres du cluster, peuvent être utilisés pour effectuer la détection à proximité de la cible et l'envoyer à son CH. Cela signifie que la création de clusters et l'attribution de tâches spéciales au CH peuvent contribuer grandement à l'évolutivité globale, à la durée de vie et à l'efficacité énergétique. Cela réduit aussi la taille de la table de routage en localisant la configuration de l'itinéraire dans les clusters en conservant la bande passante de communication du réseau [195]. La figure 6.1 montre une approche hiérarchique basée sur un cluster qui divise le réseau en différentes couches en cluster. Les différents nœuds de capteurs sont regroupés en clusters avec un CH qui a la responsabilité du routage.

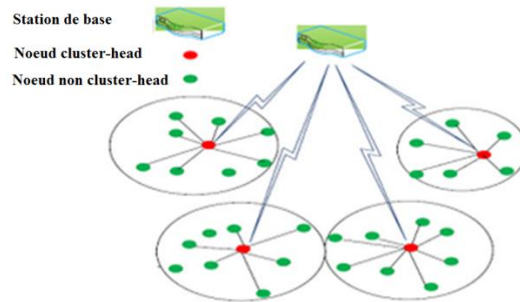


Figure 6.1. Modèle du routage hiérarchique

2.1. Protocole LEACH

LEACH est l'un des protocoles de routage hiérarchique les plus proéminents utilisés dans les réseaux de capteurs sans fil (WSN). Le protocole LEACH utilise une structure de réseau hiérarchique. Dans l'algorithme LEACH, les nœuds sont auto-organisés en différents clusters, et chaque cluster a juste un seul cluster Head. Chaque nœud décide indépendamment s'il doit s'agir de son propre cluster-head selon une certaine probabilité. Afin d'éviter la consommation excessive d'énergie des nœuds cluster-head et d'influencer la durée de vie du réseau, il est nécessaire de procéder périodiquement à l'élection des chefs de groupe (Cluster-Heads) et au processus de restructuration du réseau.

Dans le processus de cycle de fonctionnement de LEACH, la reconstruction de cluster s'exécute circulairement. Le processus de reconstruction peut être décrit comme le concept de ronde (tour). Chaque cycle peut être divisé en deux phases: phase de configuration et phase de transmission.

LEACH choisit aléatoirement des nœuds dans le réseau et leur attribue le rôle de cluster-head. Par la suite d'autres nœuds rejoignent les cluster-heads et forment ainsi les clusters. Dans le but de réduire la quantité d'informations transmises à la station de base, les cluster-heads agrègent les données capturées par les autres nœuds du cluster en un seul paquet et l'envoient à la station de base. LEACH s'exécute en deux grandes phases : la phase de configuration, « set-up phase » et la phase de transmission « steady-phase ». Lors de la phase de configuration, les nœuds cluster-head sont élus et les clusters sont formés. L'élection des cluster-heads se déroule comme suit : chaque nœud choisit un nombre de manière aléatoire entre 0 et 1. Si cette valeur est inférieure à un seuil $T(n)$ le nœud devient cluster-head. Le seuil est défini comme suit :

$$T(n) = \begin{cases} \frac{P}{1 - p * \left(r \bmod \frac{1}{p} \right)}, n \in G \\ 0, \text{otherwise} \end{cases} \quad (1)$$

Où p est le pourcentage de clusters souhaité, r est le round courant, G représente l'ensemble des nœuds qui n'ont pas encore élu le cluster-head sur les $\frac{1}{p}$ derniers rounds. Après l'élection des cluster-heads et la formation des clusters, on retrouve la phase de transmission. Lors de la phase de transmission les nœuds envoient leurs données au cluster-head du cluster auquel ils appartiennent, ensuite le cluster-head agrège toutes les données et envoie le paquet de données à la station de base. Au début de la « steady phase », un ordonnancement TDMA (Time Division Multiple Access) permet d'assigner à chaque nœud un slot de temps pour la transmission de ses données. Dans le souci d'économiser l'énergie, les nœuds simples i.e. qui ne sont pas cluster-head sont actifs uniquement durant leur temps de transmission, le reste du temps ils mettent leur radio en veille. Afin de recevoir les données des autres membres du cluster, le cluster-head par contre est tout le temps actif. A la fin du tour, on démarre un autre tour avec une nouvelle « set-up phase ». Les nœuds qui étaient cluster-head durant le cycle précédent ne peuvent pas être réélus de nouveau.

2.2. Avantages et inconvénients du protocole LEACH

2.2.1. Avantages

Le protocole LEACH présente les avantages suivants :

- ✓ L'auto-configuration des clusters se fait indépendamment de la station de base (algorithme distribué).
- ✓ Les données sont fusionnées pour réduire la quantité d'informations transmise vers la station de base.
- ✓ La consommation d'énergie est partagée sur l'ensemble des nœuds prolongeant ainsi la durée de vie du réseau.
- ✓ L'utilisation des techniques TDMA/CDMA permet d'avoir une hiérarchie et de réaliser des clusterings sur plusieurs niveaux. Ces derniers permettent d'économiser d'avantage d'énergie.

2.2.2. Inconvénients

LEACH a les inconvénients suivants :

- ✓ Le pourcentage optimal de Cluster-Heads pour le réseau est fixé à 5% du nombre total des nœuds sans être justifié. Néanmoins, la topologie, la densité et le nombre de nœuds peuvent être différents dans d'autres réseaux.
- ✓ Aucune suggestion n'est faite à propos du temps de réélection des CHs (temps des itérations).
- ✓ Les CHs les plus éloignés de la station de base meurent rapidement par rapport à ceux qui sont proches de la station.

Pour éviter ces inconvénients, les auteurs proposent une extension de l'algorithme. Cette fois-ci, ils proposent un algorithme centralisé (LEACH-C) [195]. C'est un algorithme itératif, dans lequel la structure des clusters est calculée au niveau de la station de base en utilisant la méthode d'optimisation du "Recuit simulé" [196]. Cependant, la station de base affecte dans chaque itération des rôles pour les différents nœuds du réseau (CH ou capteur simple). Ensuite, le fonctionnement continue de la même manière que pour LEACH.

2.3.Descendants du protocole LEACH

Plusieurs améliorations et versions étendues du protocole LEACH ont vu le jour.

a. LEACH-A (ADVANCE LOW ENERGY ADAPTIVE CLUSTERING HIERARCHY)

Dans LEACH, normalement le cluster-head consomme plus d'énergie que les autres nœuds de capteurs. Pour pallier à ce problème le LEACH-A est utilisé pour augmenter la durée de vie des nœuds. Chaque capteur connaît le début de chaque tour. Soit n le nombre total de nœuds et m la fraction de n qui ont une énergie supérieure à celle des autres nœuds appelés nœuds CGA (nœuds sélectionnés comme passerelle ou cluster-head), le reste des nœuds $(1 - m) * n$ agissent comme des nœuds normaux [197].

b. LEACH-B (BALANCED LOW ENERGY ADAPTIVE CLUSTERING HIERARCHY)

LEACH équilibré est la version étendue de LEACH en trouvant le nombre de Cluster-Heads. Il y a une deuxième étape de sélection des chefs de groupe dans LEACH-B. L'énergie résiduelle des nœuds de cluster devient Cluster-Heads, ce qui modifie le nombre de Cluster-Heads dans la phase de configuration en considérant l'énergie résiduelle des nœuds. Le protocole LEACH-B économise la consommation d'énergie en s'assurant que les clusters sont équilibrés. Le travail est semblable à LEACH, mais LEACH-B introduit une autre étape de sélection: Tous les chefs de groupe qui sont élus seront ordonnés de manière

déscendante en fonction de leur énergie résiduelle, et seulement ($n * p$) parmi eux, (où n est le nombre total de nœuds clusters et p le pourcentage des clusters heads) seront considérés comme des clusters-head et les autres sont des nœuds clusters [198].

c. LEACH-C (CENTRALIZED LOW ENERGY ADAPTIVE CLUSTERING HIERARCHY)

LEACH-C [195], une version centralisée de LEACH, divise également chaque tour en deux phases, La phase de configuration et la phase de transmission. Pendant la phase de configuration de LEACH-C, chaque nœud du WSN envoie ses informations, y compris l'emplacement et le niveau d'énergie, à la station de base. Puis la station de base calcule la valeur énergétique moyenne de tous les nœuds. Seuls les nœuds avec plus d'énergie que la valeur moyenne ont la chance d'être des chefs de cluster. La station de base utilise l'algorithme de recuit simulé pour établir des clusters. Les groupes de clusters sont sélectionnés pour minimiser la consommation d'énergie nécessaire pour que les nœuds ordinaires transmettent des données à leur cluster heads respectives. Les autres opérations de LEACH-C sont identiques à celles de LEACH, mais le résultat de simulation montre que LEACH-C a une amélioration évidente par rapport à LEACH. Il y a plusieurs raisons pour l'amélioration:

- Dans LEACH-C, la station de base a une station globale de réseau pour former de meilleurs clusters qui nécessitent moins d'énergie pour la transmission des données.
- Il n'y a pas de communication entre les nœuds lors de la construction de clusters. D'où plus d'énergie peut être utilisée pour la transmission de données.
- Le nombre optimal des clusters-head a été déterminé à l'avance. Le réseau peut utiliser l'énergie plus efficacement.

d. LEACH-CC (LEACH-CENTRALIZED WITH CHAIN)

Le système LEACH-CC utilisant un algorithme de contrôle central pour former le cluster peut produire de meilleurs clusters en dispersant les nœuds cluster-head dans le réseau. Ensuite, un routage de chaîne entre les cluster-heads est établi pour réduire la quantité de nœuds de cluster, qui communiquent avec la station de base. Une amélioration supplémentaire du coût énergétique pour la collecte des données peut être obtenue si un seul cluster-head est transmis aux chefs de cluster voisins dans la phase de fusion des données [199].

e. LEACH-E (ENERGY LOW ENERGY ADAPTIVE CLUSTERING HIERARCHY)

Le protocole LEACH-E améliore le processus de sélection du cluster-head par rapport au protocole LEACH. Le LEACH-E est divisé en différent tours ce qui est identique au protocole LEACH. Au premier tour, tous les nœuds de capteurs auraient la même probabilité d'être un cluster-head du cluster. Après le premier cycle de transmission, l'énergie résiduelle de chaque nœud serait différente et sur cette base, le nœud qui aurait l'énergie résiduelle la plus élevée serait choisi comme un cluster-head du cluster et les autres nœuds dans le cluster deviendraient le membre du cluster qui aurait moins d'énergie [200]. Les auteurs donnent les résultats de simulation de l'amélioration d'E-LEACH sur LEACH et multi hop-LEACH avec la consommation d'énergie due à la manière améliorée du processus de sélection de CH.

f. LEACH-F (FIXED NUMBER OF CLUSTER LOW ENERGY ADAPTIVE CLUSTERING HIERARCHY)

Le concept de base de la formation du cluster dans LEACH-F est le début de la configuration du réseau après que cela est fixé. La position du cluster-head tourne parmi les nœuds du cluster ce qui est identique à LEACH. L'avantage de ce processus par rapport à LEACH c'est qu'il n'y a pas de surcharge au début de chaque cycle comme LEACH. Pour la formation de clusters, LEACH-F utilise l'algorithme de formation du cluster centralisée qui est identique à LEACH-C. L'inconvénient de ce protocole c'est que les clusters fixes dans LEACH-F ne permettent pas l'ajout de nouveaux nœuds au réseau et n'ajustent pas leur comportement lorsqu'un nœud meurt dans le réseau [201]. La surcharge de regroupement dans le LEACH de base est supprimée par le protocole LEACH-F une fois que le nombre fixe des clusters est formé; ils sont maintenus dans tout le réseau.

g. I-LEACH (IMPROVED LOW ENERGY ADAPTIVE CLUSTERING HIERARCHY)

Les auteurs ont proposé un algorithme connu sous le nom d'I-LEACH [202]. I-LEACH emploie l'approche de clustering distribué par rapport au protocole LEACH. Le champ du capteur total est divisé en sous-région égale. Le choix du CH de chaque sous-région est déterminé par l'approche de seuil comme dans le protocole LEACH. Dans la simulation, des auteurs analysent la performance de l'I-LEACH proposé en le comparant avec le protocole LEACH basique en termes de consommation d'énergie moyenne. I-LEACH donne de meilleures performances comparativement à LEACH en termes d'énergie.

h. K-LEACH (KMEDOIDS-LOW ENERGY ADAPTIVE CLUSTERING HIERARCHY)

Le protocole proposé K-LEACH [203] utilise l'algorithme de clustering K-medoids pour obtenir un regroupement très homogène de nœuds et de très bons choix des cluster-heads. Il est très bien connu et la rétention d'énergie d'un réseau de capteur sans fil dépend fortement du groupement ou regroupement des nœuds d'émission et de réception pour le premier cycle de communication, en phase de configuration. K-LEACH considère le moins distant du centre de cluster comme un critère pour un nœud à choisir comme cluster-head pendant la procédure de sélection du cluster-head (à partir du second tour), K-LEACH est divisé en plusieurs tours et chaque cycle contient la phase de formation du cluster et la phase d'état stationnaire. Dans [203] les résultats montrent que les nœuds commencent à mourir dans LEACH dès les premières tours alors que dans K-LEACH, le premier nœud est mort au tour 15 qui est après presque 50% de la durée de vie du réseau.

i. L-LEACH (ENERGY BALANCED CLUSTERING ALGORITHM BASED ON LEACH PROTOCOL)

Les auteurs de [204] ont proposé un protocole amélioré de LEACH appelé algorithme de clustering équilibré d'énergie (L-LEACH). Le concept de base pour concevoir ce protocole amélioré par rapport au protocole LEACH traditionnel est de proposer un algorithme d'équilibre énergétique optimisant l'élection du cluster head. Cet algorithme considère globalement les facteurs d'énergie résiduelle et de distance, améliore l'élection du cluster-head et la stratégie des nœuds non-cluster-head en sélectionnant le cluster-head optimal. Les auteurs [204] présentent un résultat de simulation, la consommation d'énergie dans L-LEACH est inférieure à l'algorithme LEACH à tout moment. Les causes sont les suivantes: l'algorithme amélioré sélectionne le cluster-head en fonction de l'énergie résiduelle et des informations de coordonnées de nœud, permettent à chaque nœud de consommer de l'énergie d'une manière relativement équilibrée, évite la mort prématurée de nœuds à basse énergie, économise la dissipation d'énergie du réseau et la transmission de données est considérablement améliorée.

j. LEACH-M (MOBILE) and LEACH-ME (MOBILE ENHANCED)

Les auteurs de [205] ont proposé une version améliorée de LEACH avec un scénario de mobilité appelé LEACH-M. LEACH-M utilise la même formule de seuil du LEACH original pour calculer le seuil, mais LEACH-M prend en considération la mobilité des nœuds pendant la phase de transfert de données, ce qui n'est pas le cas dans LEACH. La

mobilité elle-même est un défi car le nœud mobile peut quitter le cluster pendant qu'il transmet des données à un cluster-head. LEACH-M résout ce problème en vérifiant si un nœud mobile peut toujours communiquer avec le cluster-head ou non selon la technique TDMA. Au début de chaque slot TDMA, le cluster-head transmet le message req-data-transmission. Si le nœud mobile ne peut pas recevoir le message, le cluster-head attend la requête dans le prochain emplacement TDMA. Si le nœud manque deux trames TDMA successives, il se considère hors de portée et le cluster-head supprime les nœuds inaccessibles de sa liste de membres. Une amélioration ultérieure par rapport à LEACH-M est connue comme LEACH-ME [206] comme une version améliorée de LEACH-M. LEACH-ME a été proposé pour améliorer LEACH-M en sélectionnant les nœuds moins mobiles par rapport à ses voisins pour être cluster-heads. Chaque nœud contient des cluster-heads de transitions qu'il a effectué pendant la phase d'état stationnaire pour la transmission de données. Les nœuds transmettent un compte de transition à leur cluster-head pendant la fente TDMA. Le cluster-head calcule le nombre de transition moyen de ses membres pour les quelques derniers cycles. La quantité de dissipation d'énergie pour les paquets de données transmis pour les protocoles LEACH-M et LEACH-ME économise plus d'énergie que LEACH-M.

k. LEACH-P (LOW ENERGY ADAPTIVE CLUSTERING HIERARCHY PERFORMANCE)

Les auteurs [207] ont introduit un autre protocole amélioré appelé LEACH-P qui considère la probabilité de sélection de l'EAMR (Energy Aware Multipath Routing) au sein de l'algorithme LEACH et fait un meilleur choix de sélection des cluster-heads en optimisant les chances de reconstruction du cluster. Dans [207], le résultat de la simulation montre que LEACH-P a amélioré la durée de vie du réseau par rapport à LEACH. Cela signifie une durée de vie du réseau LEACH-P plus longue que LEACH.

l. LEACH-S (SOLAR AWARE CENTRALIZED & DISTRIBUTED LOW ENERGY ADAPTIVE CLUSTERING HIERARCHY)

Les auteurs [208] ont proposé une autre version de LEACH, appelée solar-aware LEACH (S-LEACH). L'utilisation de l'énergie solaire dans les réseaux de capteurs sans fil a augmenté la durée de vie des réseaux. Dans S-LEACH certains nœuds sont facilités par l'énergie solaire et ces nœuds agissent comme des cluster-heads principalement en fonction de leur état solaire. Le LEACH et le LEACH-C sont tous deux prolongés par le S-LEACH. Le concept de schéma solaire est appliqué au protocole LEACH centralisé et non centralisé.

Solar-aware centralized LEACH: Dans le LEACH-S centralisé, le nœud récepteur (sink node) sélectionnerait le cluster-head à l'aide d'un algorithme de contrôle central amélioré. Dans LEACH-S, l'état solaire ainsi que l'énergie des nœuds du capteur sont transmis au sink node et les nœuds ayant l'énergie la plus élevée sont sélectionnés comme cluster-head. Lorsque le nombre de nœuds solaires augmente, la performance du réseau de capteurs augmente également ainsi que les durées de vie du réseau. La durée du soleil augmente la durée de vie du réseau de capteurs. Le transfert du cluster-head se produit si la durée de soleil est plus petite [209].

Solar-aware distributed LEACH: Dans le LEACH-S distribué, le choix du cluster-head est donné aux nœuds solaires (solar-driven nodes). La probabilité des nœuds solaires est plus élevée que celle des nœuds de batterie (battery driven nodes), pour plus d'informations sur ce papier il est conseillé de voir [209].

m. T-LEACH (THRESHOLD-BASED LOW ENERGY ADAPTIVE CLUSTERING HIERARCHY)

Les auteurs [210] présentent le protocole T-LEACH, qui est un schéma de remplacement du cluster head basé sur un seuil pour les protocoles de regroupement des réseaux de capteurs sans fil. T-LEACH minimise le nombre de sélection des cluster-head en utilisant le seuil d'énergie résiduelle. La durée de vie de l'ensemble des réseaux peut être étendue par rapport aux protocoles de regroupement existants en réduisant le montant de la sélection du cluster-head et le coût de remplacement. Dans [210], le résultat de la simulation montre que T-LEACH a surperformé LEACH en termes de consommation d'énergie et de durée de vie du réseau.

n. V-LEACH (VICE LOW ENERGY ADAPTIVE CLUSTERING HIERARCHY)

Un autre type de protocole LEACH introduit est appelé protocole V-LEACH. Dans V-LEACH, la structure hiérarchique contient un vice-cluster head et des nœuds de cluster. Dans le LEACH original, le cluster-head reçoit toujours des données du nœud du cluster, après avoir reçu les données des nœuds du cluster, le CH agrège ces données puis les envoie à la station de base. Le cluster-head meurt plus tôt que le nœud de cluster en raison de la lourde charge de travail. Lorsque le cluster-head meurt le cluster deviendra isolé parce que les données recueillies par les nœuds de cluster ne vont jamais atteindre la station de base. Dans le protocole V-LEACH ayant un cluster-head dans le cluster, il y a un sous-cluster-head qui prend le rôle du cluster-head lorsque le cluster-head original meurt [211]. Le sous-cluster-head fonctionne de manière similaire à celle du cluster-head. Dans V-LEACH, il

n'est pas nécessaire de faire une élection pour un nouveau cluster-head parce que le sous-cluster est disponible pour exécuter son travail. Cela prolongera la durée de vie du réseau.

o. W-LEACH (WEIGHTED LOW ENERGY ADAPTIVE CLUSTERING* HIERARCHY)

L'agression de données est utilisée dans cette nouvelle technique, elle gère la répartition des nœuds de différents capteurs. Dans ce cas, chaque cluster-head est choisi sur la base du poids. Le poids est basé sur deux facteurs: le premier est l'énergie restante (après quelques tours) et le second est la densité (le rapport du nombre de capteurs vivants au nombre total de capteurs). Également des nœuds de capteurs qui ont plus de poids sont situés près des cluster-heads. Il choisit seulement $x\%$ de capteurs près des cluster-heads qui transmettent les données aux cluster-heads [212].

p. WD-LEACH (WEIGHTED DECENTRALIZED LOW ENERGY ADAPTIVE CLUSTERING HIERARCHY)

Les auteurs de [213] présentent un modèle amélioré de W-LEACH à un nouveau protocole appelé W-LEACH décentralisé pour augmenter la durée de vie du réseau sans l'utilisation de cartes. Le travail avec l'algorithme décentralisé W-LEACH est similaire à LEACH, l'algorithme décentralisé W-LEACH est divisé en tours, où chaque cycle commence avec une phase de configuration dans laquelle les cluster-heads sont choisis et les clusters sont formés. Puis vient la phase d'état stationnaire dans laquelle les données sont transmises à la station de base. Juste avant le début de la transmission des données, chaque nœud détermine ses voisins selon une distance bien déterminée qui organise la densité des nœuds pour que les nœuds voisins ne transmettent pas les mêmes données. Selon le nombre de voisins, les nœuds décident leur statut soit être dans un état actif ou rester dans un état de sommeil pendant cette ronde, donc un capteur avec de faibles densités (avec beaucoup de voisins) reste en vie aussi longtemps que possible. De cette manière, les allocations de capteurs envoyant des données de manière dense vers leurs cluster-heads ont probablement conduit à minimiser les données redondantes (cluster-head) et elles partagent toujours l'envoi de données avec leur cluster-head. Par la suite, la consommation d'énergie totale du réseau est minimisée, de sorte que nous augmentons la durée de vie du réseau. Au cours de chaque cycle du réseau, il existe des nœuds membres qui sont sélectionnés en deux groupes: nœuds endormis et nœuds actifs. Dans [213], W-LEACH décentralisé économise plus d'énergie que le protocole LEACH.

q. LEACH-SM

Il a introduit un autre concept de gestion des nœuds de cluster de rechange pour les chefs de cluster primaires. Le protocole LEACH comporte deux inconvénients. L'un est le problème de hotspot qui est survenu en raison de surcharge supplémentaire des Cluster Heads. Le deuxième démerite est la redondance des transmissions de données au Cluster-Heads. Ces deux problèmes sont identifiés lorsque des nœuds de cluster de rechange sont utilisés. Lorsque le chef de cluster primaire perd toute son énergie, le nœud de cluster de rechange assume la responsabilité et évite ainsi l'arrêt de la saisie des données. Le protocole LEACH-SM sélectionne également les nœuds de cluster de rechange appropriés et décide la durée pendant laquelle les nœuds de cluster de rechange doivent être en état d'inactivité. La durée de vie de LEACH peut être prolongée jusqu'à 50% en utilisant ce protocole [214].

r. AIR QUALITY MONITORING LEACH (AQM-LEACH)

C'est un autre concept développé dans le protocole LEACH qui s'appelle AQM-LEACH. Ce protocole spécifiquement pour la surveillance de la qualité de l'air [215]. Dans LEACH, les nœuds de cluster n'émettent pas de données en continu à la station de base à chaque tour. Il s'agit d'un protocole orienté événement, dans lequel la plupart des nœuds de cluster sont en mode veille. Lorsqu'une anomalie se produit dans la qualité de l'air, les nœuds du cluster envoient des détails à la station de base concernant l'anomalie. Ce protocole mesure la qualité de l'air pour indiquer le niveau de pollution dans l'environnement. Ce protocole prévoit également le niveau de danger pour la santé humaine. La valeur est différente selon les pays. Lorsque l'un des nœuds capteurs détecte l'événement qui est anormal, à ce moment les données sont recueillies. Puisque la plupart du temps le nœud cluster étant dans le stade de sommeil, la durée de vie du réseau est augmentée.

s. ENERGY EFFICIENT LEACH PROTOCOL

Il a introduit un autre protocole qui s'appelle EELP. Ce protocole est un protocole pratique, ce n'est pas conçu pour les réseaux ad-hoc de capteurs sans fil. Il est spécialement conçu pour le réseau de nœuds de capteurs dont les emplacements sont prédéterminés. L'application générale de l'EELP est d'identifier les gaz dangereux dans les bâtiments résidentiels ou commerciaux. Le protocole EELEACH comporte deux seuils 1). Valeur seuil inférieur (TL), 2). Valeur de seuil plus élevé (HT) [216]. Si les données détectées sont en dessous du LT (niveau dangereux), les données ne sont pas envoyées à la station de base. Sinon, les données détectées i.e (HT) sont envoyées directement à la station de base et il n'est pas nécessaire d'attendre le cluster-head. L'aspect de l'EELP utilise l'opération XOP pour éliminer les données en double détectées par les nœuds du cluster voisins.

t. MULTI-HOP LEACH

Dans Multi-Hop Leach, lorsque le diamètre du réseau est augmenté dans un certain niveau, alors LEACH protocole devient inefficace. Dans le protocole LEACH, la dissipation d'énergie du cluster-Head n'est pas abordable. Dans le protocole Multi-Hop LEACH, un nœud de cluster se choisit comme cluster-Head pour compléter les informations de cluster dans la phase de configuration. Dans la phase d'état stationnaire suivante, le cluster-head recueille des données de tous les nœuds du cluster et les envoie directement à la station de base après agrégation. Ce protocole permet deux types d'opération de communication [217].

(1) Communication Inter-Cluster.

(2) Communication intra-Cluster.

Dans le protocole LEACH multi-Hop, la communication intra-cluster dans laquelle tout le réseau est divisé en clusters multiples, chaque cluster a un cluster-Head. Ce cluster-Head est responsable de la communication entre tous les nœuds du cluster dans le cluster. Et il reçoit des données de tous les nœuds du cluster à une distance de Hop unique et agrège et transmet les données directement à la station de base, ou à travers des Cluster-Heads intermédiaires. Dans la communication inter-cluster Multi-hop, lorsque la distance entre les cluster-heads et la station de base est importante, le cluster-head utilise des cluster-heads intermédiaires pour communiquer avec la station de base [217].

u. LEACH-TLCH (LEACH with TWO LEVELs CLUSTER HEAD)

C'est mieux que le LEACH normale, car nous savons que la sélection du cluster-head et du cluster est la même dans le protocole LEACH. Si l'énergie du cluster-head est inférieure à l'énergie moyenne et si la distance entre le cluster-head et la station de base est supérieure à la distance moyenne pour la communication, le nœud commun avec l'énergie maximale dans le WSN sera sélectionné comme cluster dans ce protocole [218].

v. ENERGY EFFICIENT WEIGHT CLUSTERING BASED ON LEACH

Il utilise une métrique pour sélectionner un cluster-head telle que :

Premièrement: la distance entre les cluster-heads, les nœuds et la station de base. La distance minimale du cluster-head aux clusters et la station de base sera sélectionnée comme cluster head. Deuxièmement: le degré de nœud, les nœuds ayant plus de voisinage économisent l'énergie par le transfert de données des cluster-heads à la station de base. Troisièmement: l'énergie résiduelle, plus l'énergie résiduelle est élevée il y aura plus de

chance de construire des cluster-heads. Il utilise également des modes de canal tels que l'espace libre et la décoloration par trajets multiples avec respectivement la perte de puissance d_2 et d_4 [219]. La ligne de temps de LEACH et ses protocoles descendants est présentée dans le tableau suivant :

S.No.	Leach et ses descendants	Auteurs	Abréviation	Année
01	LEACH[193]	Heinzelman. W.R. et al	Low Energy Adaptive Clustering Hierarchy	2000
02	LEACH-B[198]	Depedri. A.et.al.	Balanced Low Energy Adaptive Clustering Hierarchy	2003
03	LEACH-C[195]	Heinzelman.W.et.al.	Centralized Low Energy Adaptive Clustering Hierarchy	2002
04	LEACH-E[200]	Fan. X. N.et.al.	Energy Low Energy Adaptive Clustering Hierarchy	2007
05	LEACH-F[201]	Manimala.P.et.al.	Fixed number of cluster Low Energy Adaptive Clustering Hierarchy	2013
06	LEACH-I[202]	Dembla.D.et.al.	Improved Low Energy Adaptive Clustering Hierarchy	2013
07	LEACH-K[203]	Bakaraniya.P.,et.al.	Kmedoids-Low Energy Adaptive Clustering Hierarchy	2013
08	LEACH-L[204]	Qian .L.,et.al.	Energy Balanced Clustering Algorithm Based on LEACH Protocol	2013
09	LEACH-M[205]	Kim. D. S.et.al.	Mobile Low Energy Adaptive Clustering Hierarchy	2006
10	LEACH-ME[206]	Kumar G. S.et.al.	Mobile Enhanced Low Energy Adaptive Clustering Hierarchy	2008
11	LEACH-P[207]	Zhu.D.et.al.	Low Energy Adaptive Clustering Hierarchy Performance	2010
12	LEACH-S[208]	Thiemo .V.et.al.	Solar aware Low Energy Adaptive Clustering Hierarchy	2004
13	LEACH-T[210]	Jiman .H.et.al.	Threshold-based Low Energy Adaptive Clustering Hierarchy	2007
14	LEACH-V[211]	Bani Y.M.et.al.	Vice Low Energy Adaptive Clustering Hierarchy	2009
15	LEACH-W[212]	Hanady. M.et.al.	Weighted Low Energy Adaptive Clustering Hierarchy Aggregation	2013
16	LEACH-WD[213]	Abdelhalim.H.et.al	Weighted Decentralized Low Energy Adaptive Clustering Hierarchy	2014

Tableau 6.1.Ligne de temps de LEACH et ses descendants

2.4.Comparaison entre le protocole LEACH et ses descendants pour les réseaux de capteurs sans fil

Le protocole LEACH améliore la durée de vie d'un WSN, enregistre l'énergie par rotation aléatoire de cluster-head et affecte la technique TDMA à chaque membre du cluster afin d'éviter une collision. La sélection de CH est aléatoire, même si LEACH améliore l'efficacité énergétique, mais elle ne fonctionne pas bien dans une zone de grande

couverture qui nécessite une transmission multi-sites, ne supportant pas la mobilité, la fiabilité, etc. Pour surmonter ces inconvénients, plusieurs descendants plus efficaces de LEACH ont été développés et sont résumés sur diverses hypothèses, comme le montre le tableau 6.2.

Descendant de LEACH	Méthode de Clustering	Aggregation de données	Mobility Type	Mise à l'échelle	Avantages	Inconvénients
LEACH[193]	Distribuée	Oui	Statique	Limitée	Charge la distribution dans le réseau	CH ne sont pas uniformes
LEACH-B[198]	Distribuée	Oui	Statique	Bonne	Améliore la durée de vie du réseau	Le coût augmente
LEACH-C[195]	Centralisée	Oui	Statique	Bonne	Atteint plus de tours en n/w	Le coût dans la BS
LEACH-E[200]	Distribuée	Oui	Statique	Très bonne	Améliore la sélection des CH	CH est toujours actif
LEACH-F[201]	Centralisée	Oui	Statique	Limitée	Le retard est petit	Couvre une plus grande région
LEACH-I[202]	Distribuée	Oui	Statique	Très bonne	Divise également le champ	Mises à jour périodiques
LEACH-K[203]	Distribuée	Oui	Statique	Bonne	Période de stabilité prolongée	besoin d'équilibrage de charge
LEACH-L[204]	Distribuée	Oui	Statique	Très bonne	Charge réseau équilibrée	Besoin de plus de capacité de stockage
LEACH-M[205]	Distribuée	Oui	Mobile	Bonne	Mobilité des nœuds CH	Augmentation du coût
LEACH-ME[206]	Distribuée	Oui	Mobile	Limité	Supporte la mobilité des nœuds	Coût supplémentaire
LEACH-P[207]	Centralisée	Oui	Statique	Bonne	Améliore la durée de vie du réseau	Coût supplémentaire introduit
LEACH-S[208]	Distribuée	Oui	Statique	Très bonne	Gain de puissance de l'énergie solaire	Centralement contrôlé
LEACH-T[210]	Distribuée	Oui	Statique	Bonne	Réduire la sélection des CH	CH basé sur le seuil
LEACH-V[211]	Distribuée	Oui	Statique	Très bonne	Introduit vice CH	Traitement supplémentaire pour vice CH
LEACH-W[212]	Centralisée	Oui	Statique	Bonne	Améliore la durée de vie du réseau	Sélection aléatoire des CH

Tableau 6.2: LEACH et ses protocoles Descendants

On peut conclure de cette enquête que pour rendre le WSN plus efficace et prolonger la durée de vie, il faut encore améliorer le protocole de clustering.

III. Modèle du réseau

Dans ce travail, on suppose un modèle de réseau de capteurs avec les propriétés suivantes:

- Il existe une station de base fixe éloignée des nœuds du capteur. Dans l'étude, nous ne considérons pas la consommation d'énergie de la BS et supposons qu'il a suffisamment d'approvisionnement en énergie.
- Tous les nœuds ont la même configuration et une énergie limitée.
- Chaque nœud a la même énergie de départ dans le réseau.
- Les nœuds peuvent percevoir l'énergie résiduelle et changer la puissance d'émission.

3.1. Modèle de consommation d'énergie dans les réseaux de capteurs

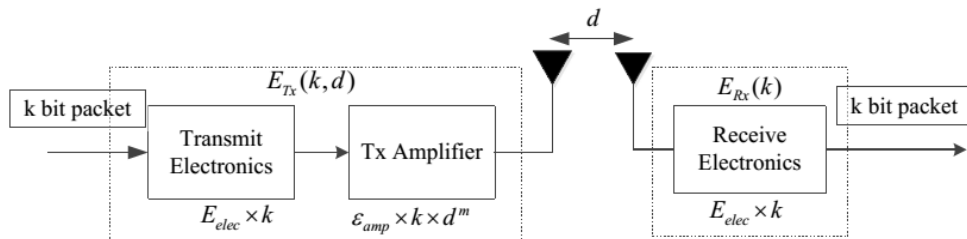


Figure 6.2 .Modèle de consommation d'énergie pour la communication

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

- Acquisition : L'énergie consommée pour effectuer l'acquisition n'est pas très importante. Néanmoins, elle varie en fonction du phénomène et du type de surveillance effectuée.

- Communication : Les communications consomment beaucoup plus d'énergie que les autres tâches. Elles couvrent les communications en émission et en réception. La figure 6.2 présente un modèle d'antenne et les règles de consommation d'énergie associées [220].

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

$$E_{Tx}(k, d) = E_{elec} \times k + \epsilon_{amp} \times k \times d^m \quad (2)$$

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

$$E_{Rx} = E_{elec} \times k \quad (3)$$

Avec E_{elec} : énergie de transmission/réception électronique, k : taille d'un message, ε_{amp} : facteur d'amplification et la distance d^m est une échelle. Si la distance de transmission est courte, $d < d_0$ et $m=0$, sinon $d \geq d_0$ et $m=4$. Alors la consommation d'énergie d'envoi de données de k bit est:

$$E_{Tx}(k, d) = \begin{cases} E_{elec} \times k + \varepsilon_{amp} \times k \times d^2, & d < d_0 \\ E_{elec} \times k + \varepsilon_{amp} \times k \times d^4, & d \geq d_0 \end{cases} \quad (4)$$

IV. Déclaration de problème et solution proposée

LEACH sélectionne le Cluster-Head complètement au hasard parmi tous les nœuds. Et LEACH-C définit une restriction simple pour les nœuds qui seraient sélectionnés en tant que cluster-Head, uniquement les nœuds qui ont plus d'énergie que la valeur moyenne peuvent participer à l'élection du cluster-head. Cependant, l'effet de cette méthode est encore limité. Surtout quand l'énergie moyenne des nœuds est faible, même si le niveau d'énergie résiduel des nœuds est supérieur à la valeur moyenne, il est encore possible que le nœud meurt après ce cycle.

Lorsque la distribution des nœuds est inégale, un cluster peut avoir plusieurs nœuds alors que d'autres peuvent en avoir très peu. Le cluster-Head d'un grand nombre de nœuds peut mourir plus rapidement. Ce phénomène affecte l'efficacité du réseau. LEACH et LEACH-C ne parviennent pas à résoudre ce problème.

Pour surmonter ces problèmes, nous proposons deux améliorations une dans la sélection du Cluster-Head et l'autre dans l'attribution des nœuds.

Le travail de ce chapitre propose un schéma amélioré de sélection du Cluster-Head. Ce schéma peut encore équilibrer la distribution d'énergie en redéfinissant la gamme électorale des Cluster-Heads en fonction du niveau d'énergie des nœuds.

L'autre stratégie principale consiste à allouer des nœuds au Cluster-Head le plus proche si le nombre de nœuds déjà attribués à ce Cluster-Head est inférieur à un nombre particulier N . Si le nombre est supérieur à N , un nœud est attribué au deuxième Cluster-Head le plus proche, et ainsi de suite. N est calculé en divisant le nombre total de nœuds par le nombre des Cluster-Heads.

Pendant la phase de configuration de chaque tour, comme le protocole LEACH-C, la BS recueille des informations sur l'état énergétique actuel de tous les nœuds. Après avoir reçue ces informations, la BS calcule le niveau d'énergie moyen et la différence entre l'énergie de chaque nœud et la valeur moyenne dans le réseau. Ensuite, BS choisit un ensemble de nœuds, désigné G, dont la différence satisfait une certaine relation avec la consommation d'énergie de chaque cycle.

Après la sélection des cluster-heads, la station de base alloue alors les nœuds au cluster-head le plus proche si le nombre de nœuds déjà attribué au cluster-head est inférieur à N. S'il est supérieur à N, la station de base calcule le deuxième cluster-head le plus proche. Si le nombre de nœuds du deuxième cluster-head est également supérieur au nombre N, la station de base trouve le troisième cluster-head le plus proche et le processus se poursuit. Une fois que les nœuds sont affectés aux cluster-heads, la station de base envoie à chaque nœud l'identifiant de son cluster-head. Le drain d'énergie égal sur les cluster-heads augmente également l'efficacité et la durée de vie du réseau.

La phase d'état stationnaire est similaire à celle du LEACH et LEACH-C. Les nœuds envoient des données au cluster-head. Le cluster-head agrège les données et les transmet à la station de base. Après un temps prédéfini, le prochain cycle commence et les phases de configuration et celle de l'état stationnaire continuent.

Procédure de LEACH-C amélioré

Le processus spécifique est le suivant:

Étape 1: Initialisation des paramètres de simulation comme indiqué dans le tableau 6.3

Étape 2: Répartition des nœuds de manière aléatoire et uniforme à l'aire totale

Étape 3: Supposons que tous les nœuds sont initialement normaux.

Création de clusters: initialiser le nombre de cluster-head à zéro

Étape 4: Sélection du cluster-head

Étape 5: la BS calcule le niveau d'énergie moyen de tous les nœuds du réseau

Étape 6: La BS choisit les nœuds qui peuvent participer à l'élection du cluster-head selon la formule (5)

$$E_{node_eng} - E_{node_avg_eng} \geq n \times E_{each_round}, n = 1;2;3.... \quad (5)$$

Étape 7: Comptez le nombre k des nœuds qui satisferont l'exigence ci-dessus. Si $k < k_{op}$, la station de base redéfinit l'ensemble G comme LEACH-C. Pour le reste des expériences, on a fixé k_{op} à 5.

Étape 8: Après validation, la BS sélectionne les cluster-heads selon la formule (1), puis trouve des clusters.

Étape 9: Phase de configuration

Trouver le nombre de nœuds pour chaque cluster

$$N_{nodes} = ((Nodes - cluster + 1) / (cluster - 1)) \quad (6)$$

Le nombre total de nœuds qui ne sont pas des CH est partagé également parmi les nœuds CH.

Si $d < d_{min}$ et $N_{nodes_allocated} < N_{nodes}$

$$d_{min} = d \quad \text{and} \quad d_{min_CH} = Current_{CH} \quad (7)$$

Sinon

Les nœuds sont affectés au CH le plus proche, s'il a moins que le nombre de nœuds attribué

S'il y a des nœuds restants, il choisira son CH voisin.

Étape 10: les cluster-heads créent une séquence TDMA pour chaque nœud au sein des clusters.

Étape 11: Dans la phase de communication, les nœuds qui ne sont pas des cluster-heads envoient leurs données collectées au cluster-head pendant leurs propres séquences TDMA. En autre temps, les nœuds restent en état de couchage pour économiser de l'énergie. Les cluster-heads traitent la fusion de données et les envoient à la BS. Après un certain temps, le réseau entrera dans la prochaine ronde.

L'organigramme de l'algorithme LEACH-C amélioré est présenté dans la figure 6.3.

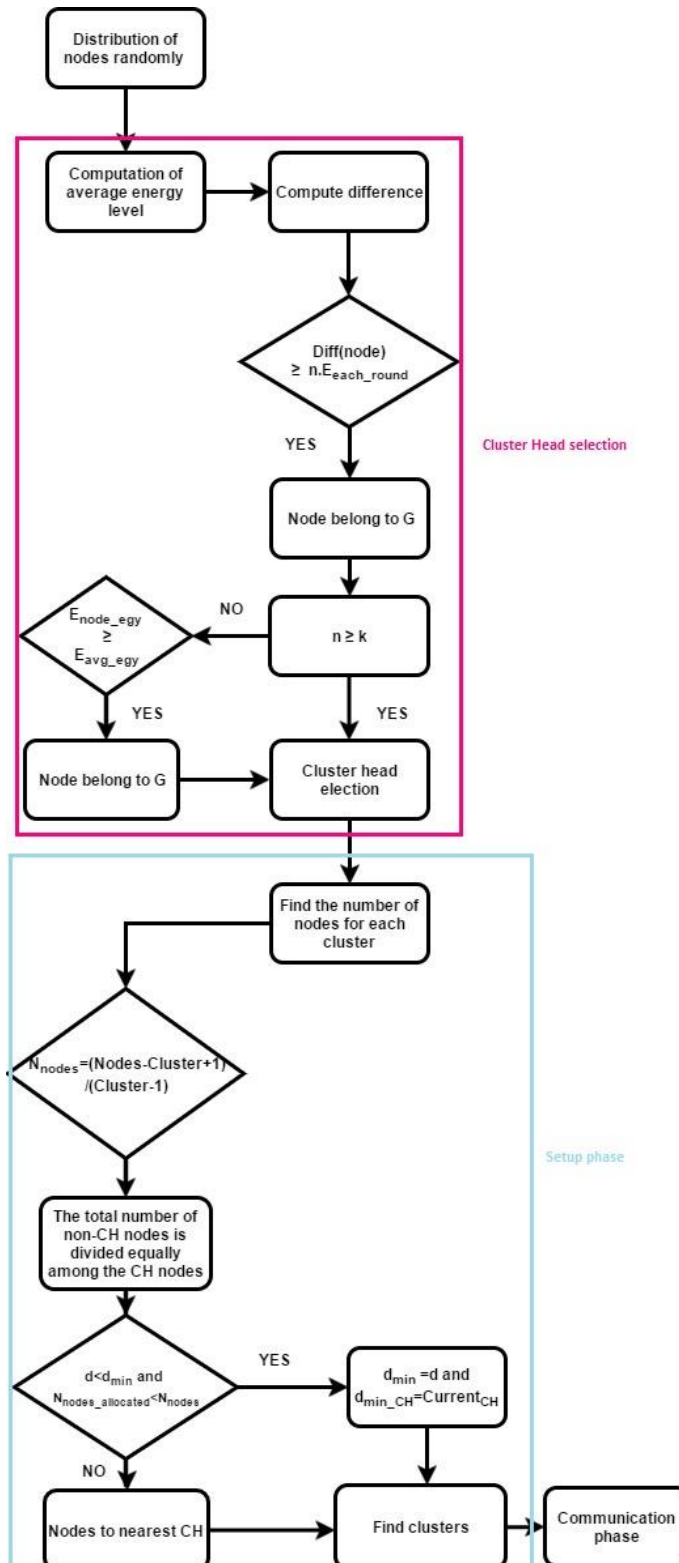


Figure 6.3 . Flow chart du protocole LEACH-C amélioré

V. Résultats de simulation et discussions

La simulation est exécutée à l'aide de MATLAB. Pour tester les performances de l'approche proposée, LEACH-C amélioré contre LEACH-C et LEACH, un réseau est créé avec 100 nœuds aléatoires d'abord, puis un réseau est créé avec un déploiement de nœud

aléatoire de densité de nœud variable entre 100 à 300 nœuds. Ces nœuds sont déployés dans une zone de $100m \times 100m$. Chaque nœud a une énergie initiale de $0.5J$. Les coordonnées du nœud de l'évier ont également été déterminées. Le nombre maximum de tours a été pris en charge en 5000. La taille des paquets a été considérée comme 4000. Les différents paramètres de simulation de réseau et leurs valeurs sont définis dans le Tableau 6.3.

Pour calculer la consommation d'énergie de chaque cluster-head $Energy_{CH}$, la distance a été comparée à la valeur do , cette valeur a été obtenue en utilisant l'équation suivante:

$$do = \text{sqrt}(Efs/Emp) \quad (8)$$

Où Efs , et Emp sont des renforceurs de transmission. Dans le cas de $distance > do$, $Energy_{CH}$ sera calculée par l'équation suivante:

$$Energy_{CH} = (ETX + EDA) \times (Packet) + Emp \times Packet \times (distance \times distance \times distance \times distance) \quad (9)$$

Sinon, si $distance \leq do$

$$Energy_{CH} = (ETX + EDA) \times (Packet) + Emp \times Packet \times (distance \times distance) \quad (10)$$

Où Emp , le renforceur de transmission et EDA et ETX sont les données d'agrégation d'énergie. Cependant, l'énergie consommée par chaque nœud a été calculée en comparant la distance la plus courte à do comme suivant:

Si $\min_{dist} > do$ alors:

$$Energy_{Mumber} = ETX \times (Packet) + Emp \times (Packet) \times (\min_{dist} \times \min_{dist} \times \min_{dist} \times \min_{dist}) - (1/4000) \times ((Emp \times Packet \times (\min_{dist}))^{(0.01)}) \quad (11)$$

Cela atteint les meilleurs résultats beaucoup mieux et plus rapidement que les méthodes précédentes.

Les paramètres de performance ci-dessous sont utilisés pour évaluer le protocole.

Paramètres de Simulation	Valeurs
Zone de déploiement des noeuds	100m×100m
Evier.x	0.5×x _m
Evier.y	0.5×y _m
Nombre total des noeuds (N)	100,200,300
Energie Initiale (E ₀)	0.5 J
T _x and R _x energy for each node	50×10 ⁻⁹ J
Energie d'agrégation de données (EDA)	50×10 ⁻⁹ J / bit / message
Nombre total des cycles r _{max}	5000
Taille des paquets	4000 bits
Coefficient d'amplification du noeud emetteur(d<d ₀) (Efs)	10pJ / bit / m ²
Coefficient d'amplification du noeud emetteur (d≥d ₀) (Emp)	0.0013pJ/bit/m ⁴

Tableau 6.3: Paramètres de simulation

- ✓ **Durée de vie du réseau (Network Life time):** c'est l'intervalle de temps entre le démarrage de l'opération du réseau jusqu'à la mort du dernier nœud. La durée de vie du réseau est calculée à l'aide de trois mesures: First Node Dies (FND)(La mort du premier nœud), Half Node Dies (HND)(la mort de la moitié des nœuds) et Last Node Dies (LND)(La mort du dernier nœudt). Le FND est également désigné comme une période stable. Le résultat pour FND, HND et LND est illustré dans les figures 6.6-6.8.
- ✓ **Période de stabilité:** c'est l'intervalle de temps entre le démarrage de l'opération du réseau jusqu'à la mort du premier nœud.
- ✓ **Nombre de nœuds vivants par tour:** cela mesurera le nombre de nœuds vivant dans chaque tour (Fig. 6.9-6.11).
- ✓ **Nombre de paquets envoyés à la station de base:** cela mesurera le nombre total de paquets envoyés à la station de base (Fig. 6.14-6.16).
- ✓ **Consommation d'énergie d'un nœud:** elle indique la consommation d'énergie par nœud.C'est un paramètre très important pour prolonger la durée de vie du réseau. La consommation d'énergie par nœud est représentée dans les Figures (6.17-6.19)
- ✓ **Nombre de cluster-heads par tour:** cela mesurera le nombre de cluster-heads formés à chaque tour (Fig. 6.20).
- ✓ **Nombre de nœuds morts par tour:** cela mesurera le nombre de nœuds morts dans chaque tour (fig. 6.22-6.24).

- ✓ **Le nombre de paquets envoyés au cluster-head:** cela mesurera le nombre total de paquets envoyés au cluster-heads (Fig. 6.25).

Dans la formule (5), E_{each_round} est un facteur clé dans notre nouveau schéma. La décision de sa valeur aura un effet direct sur la performance du protocole. De la présentation ci-dessus, par rapport à LEACH-C, nous pouvons constater que le nouveau schéma ne propose pas de suggestion d'économie d'énergie, sauf pour équilibrer la distribution d'énergie du réseau. La consommation d'énergie de chaque cycle devrait être égale à celle de LEACH-C. Par conséquent, il est juste que nous utilisions les données expérimentales de LEACH-C pour obtenir la valeur de E_{each_round} .

La figure 6.4 montre que la dissipation d'énergie du réseau est stable dans LEACH-C. Avant la mort du premier nœud, la consommation d'énergie est approximativement linéaire et s'accorde bien avec la ligne droite. La linéarité R^2 arrive jusqu'à 0.9888, ce qui indique que la droite présente la faisabilité et la validité des éclaboussures de la figure 6.4. La pente est directement liée à la consommation d'énergie de chaque tour. Par conséquent, selon cette ligne, on a fixé E_{each_round} à 0.03676J.

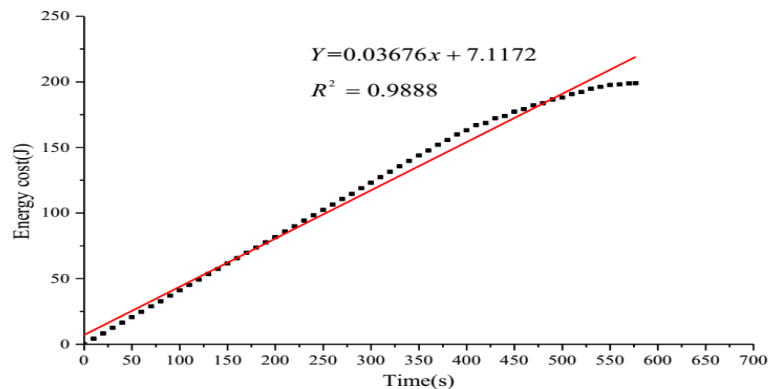


Figure 6.4 . Energie de dissipation de LEACH-C

Une autre contrainte est de savoir comment choisir la valeur de n . Si n est trop petite, la qualité des nœuds pour satisfaire la formule (5) est trop importante pour que l'effet d'un nouveau schéma ne soit pas évident, au contraire s'elle est trop grande, il n'aura pas assez de nœuds et le nouveau schéma sera similaire à LEACH-C. Dans ce travail, nous choisissons n par une simulation. La figure 6.5 montre le début de la mort du premier nœud lorsque l'énergie initiale est de 0,5 J. Selon cette figure, la simulation est conforme à l'analyse. Ce graphique montre que la valeur optimale est comprise entre 2 et 18 pour ce réseau. Par conséquent, nous choisissons de définir n à 8 pour le reste des expériences.

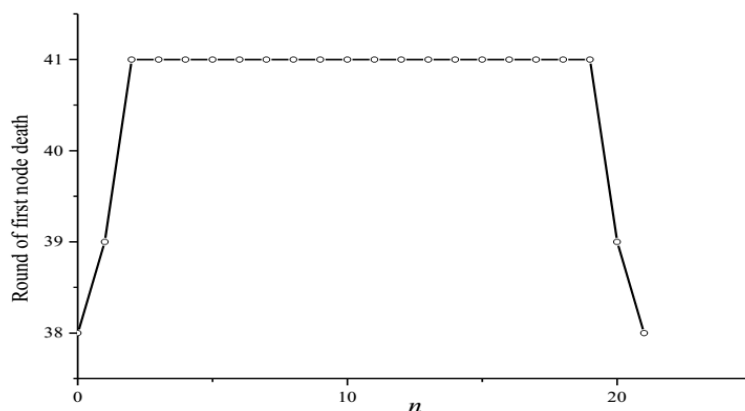


Figure 6.5 . Cycle de la mort du premier nœud comme valeur de n

On peut voir dans les figures 6.6-6.8 que l'approche proposée améliore la durée de vie du réseau, y compris la période de stabilité. Pour toutes les densités différentes de réseau, l'approche proposée présente les meilleures performances pour augmenter la durée de vie du réseau de capteurs.

Les figures 6.9 à 6.11 englobent le nombre de nœuds vivants par tour et ils sont plus dans LEACH-C amélioré par rapport à LEACH-C et LEACH. Selon ces chiffres, le temps de la mort (Dead time) du premier nœud dans l'algorithme LEACH-C amélioré est postérieur au protocole LEACH et LEACH-C. Le premier nœud découle au moment de 956s, 1262s et 1528s dans LEACH, LEACH-C et LEACH-C amélioré respectivement. C'est parce que, comme le protocole LEACH-C, la station de base connaît l'emplacement et le niveau d'énergie de tous les nœuds du réseau, ce qui permet d'établir des clusters plus efficaces nécessitant moins d'énergie pour transmettre des données. Et le LEACH-C amélioré a utilisé un nouvel algorithme pour choisir la gamme des nœuds qui participeront à la sélection des cluster-heads. Cela peut éviter une condition préalable que le nœud épuise son énergie après avoir été cluster-head, surtout lorsque l'énergie moyenne est faible dans le réseau. Donc, cette méthode peut forcer le nœud qui a plus d'énergie à consommer cette dernière en étant un cluster-head et à la distribuer de façon plus uniforme dans le réseau de capteurs (WSN).

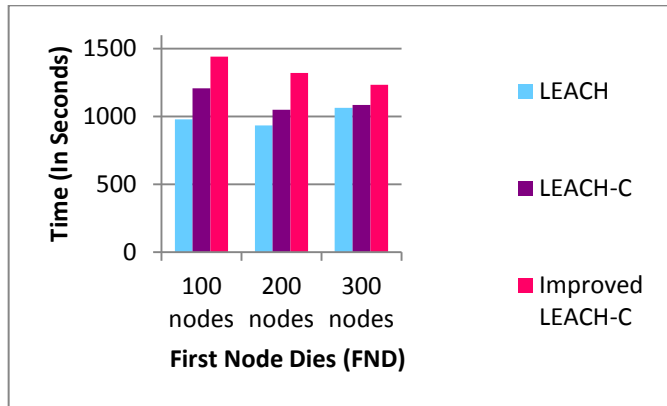


Figure 6.6. Mort du premier noeud (FND)

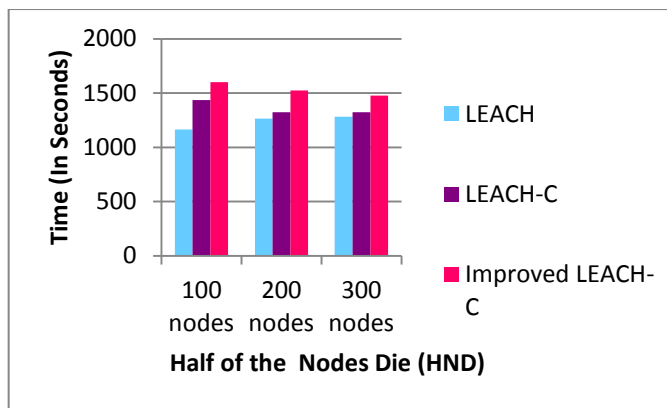


Figure 6.7. Mort de la moitié des noeuds (HND)

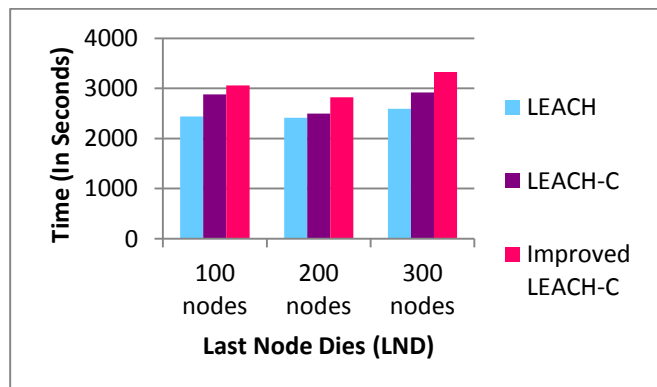


Figure 6.8. Mort du dernier noeud (LND)

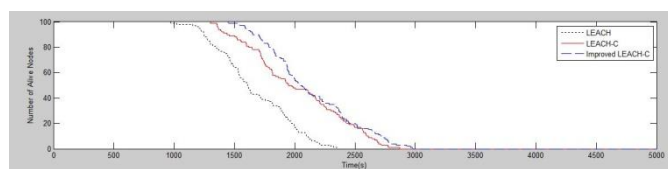


Figure 6.9 . Nœuds vivants de différents protocoles pour un réseau de 100 nœuds

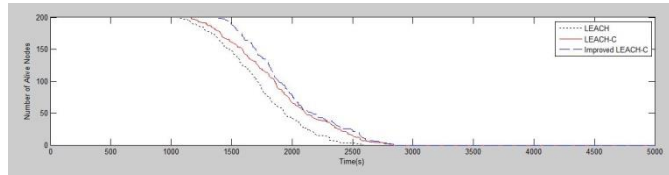


Figure 6.10. Nœuds vivants de différents protocoles pour un réseau de 200 nœuds

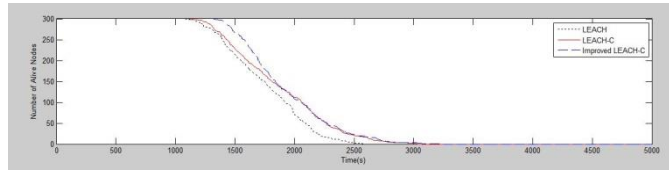
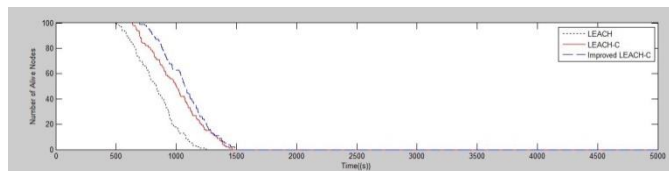
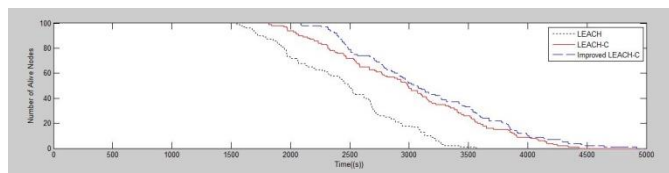


Figure 6.11 . Nœuds vivants de différents protocoles pour un réseau de 300 nœuds

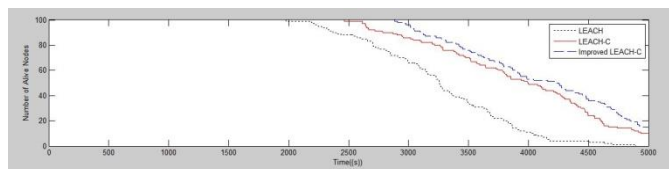
Trois autres simulations sont faites pour vérifier le nombre de nœuds vivants à différentes énergies initiales, 0.25J, 0.75J et 1J pour une zone de 100 nœuds. La figure 6.12- (a, b, c) montre que le protocole LEACH-C amélioré présente un avantage plus apparent lorsque l'énergie initiale est plus grande par rapport à LEACH-C et LEACH. Nous pouvons expliquer ce résultat comme suit: Plus d'énergie augmentera les chances de répartition d'énergie par déséquilibre et il y aura plus de probabilité que les nœuds arrivent à des décès prématurés. Le protocole LEACH-C amélioré peut éviter cette condition, respectivement.



(a) Nœuds vivants avec Energie Initiale de 0.25J



(b) Nœuds vivants avec Energie Initiale de 0.75J



(c) Nœuds vivants avec Energie Initiale de 1J

Figure 6.12 . Nœuds Vivants avec différentes valeurs d'énergie initiale

Afin de valider l'exactitude de notre analyse, nous calculons la variance de l'énergie résiduelle avant les 1000s de la mort du premier nœud. La figure 6.13 montre que la variance énergétique de LEACH-C amélioré est toujours inférieure à celle de LEACH-C et la distribution d'énergie est plus importante dans le LEACH-C amélioré. Le nouvel algorithme prolonge la durée de vie du réseau en éliminant les nœuds à faible énergie étant cluster-head.

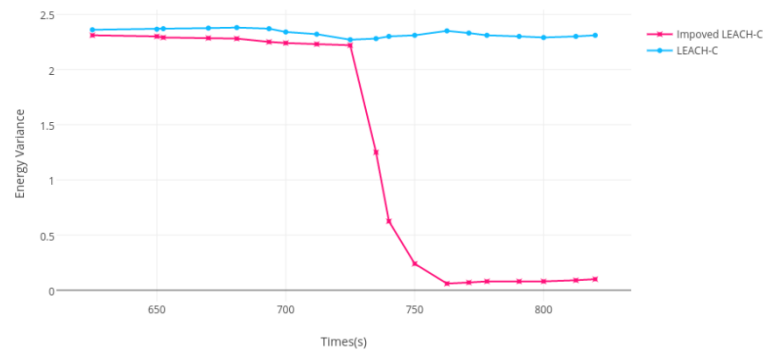


Figure 6.13 . Variance d'énergie de différents protocoles

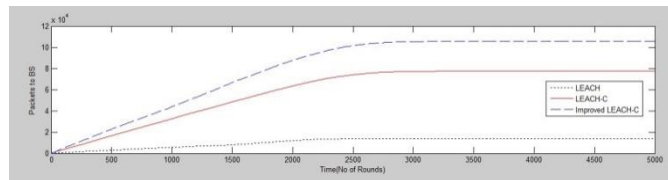


Figure 6.14 . Paquets envoyés à la BS pour différents protocoles pour un réseau de 100 nœuds

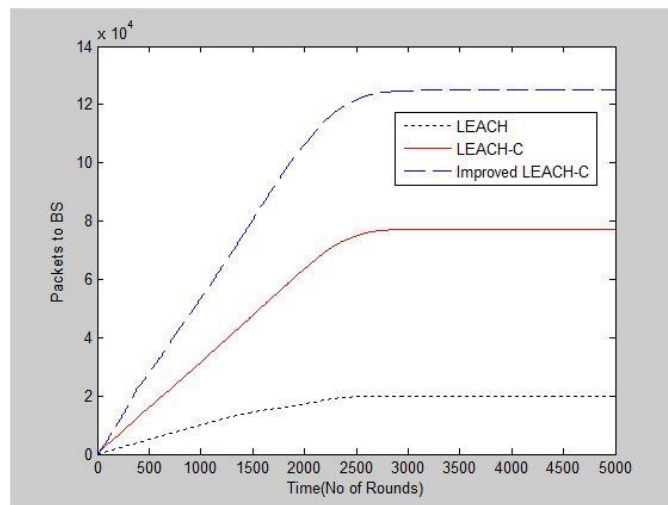


Figure 6.15 . Paquets envoyés à la BS pour différents protocoles pour un réseau de 200 nœuds

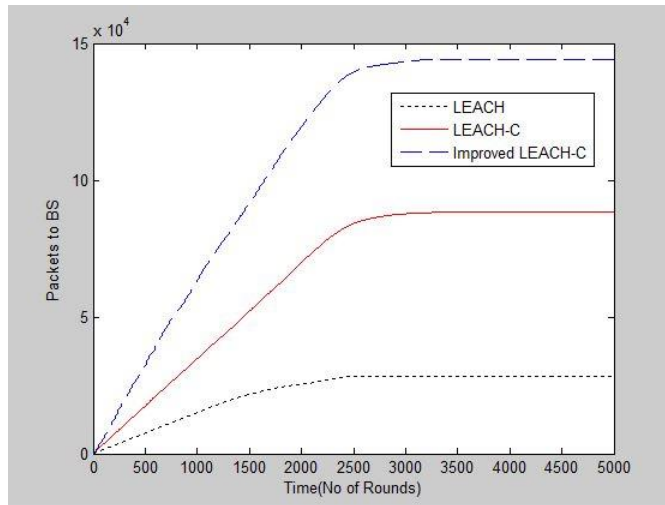


Figure 6.16 . Paquets envoyés à la BS pour différents protocoles pour un réseau de 300 nœuds

Les figures 6.14-6.16 montrent le nombre de paquets reçus par la BS du réseau. Elles confirment qu'un nombre plus élevé de paquets est reçu par la BS pour le protocole LEACH-C amélioré par rapport aux protocoles LEACH-C et LEACH. Le nombre élevé de paquets reçus indique un taux de mort des nœuds et une consommation d'énergie plus faibles.

Chaque fois que la densité du réseau augmente, le nombre de paquets reçus par la station de base du réseau augmente aussi et l'approche proposée montre toujours le nombre de paquets le plus élevé par rapport aux autres protocoles.

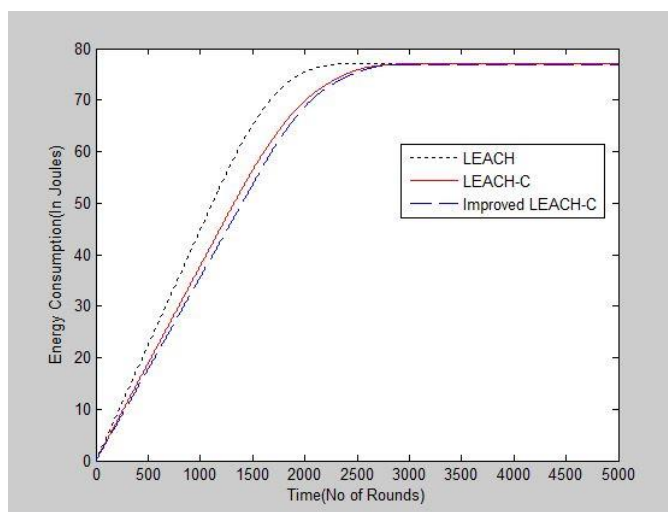


Figure 6.17 . Consommation d'énergie des différents protocoles pour un réseau de 100 nœuds

Les figures 6.17 à 6.19 tracent la consommation totale d'énergie par cycle de LEACH-C, LEACH-C et LEACH amélioré sur une période de temps pour un réseau avec différentes densités de nœuds. D'après les chiffres, il est clair que la consommation d'énergie totale de LEACH-C améliorée est inférieure à LEACH-C et LEACH. Le LEACH-C amélioré surpasse mieux à cet égard.

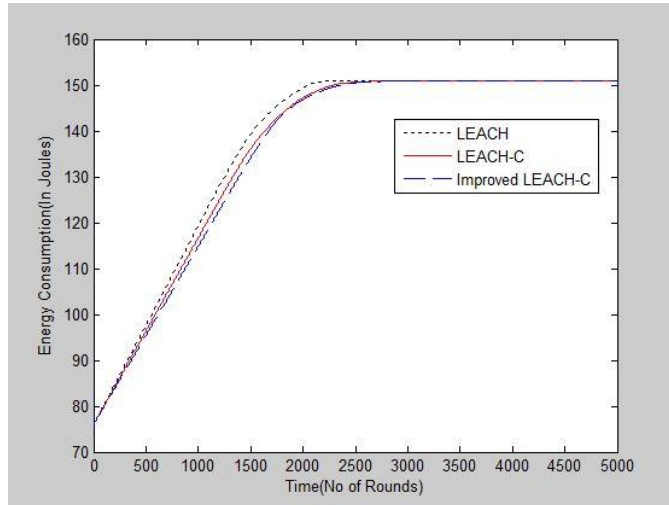


Figure 6.18. : Consommation d'énergie des différents protocoles pour un réseau de 200 nœuds

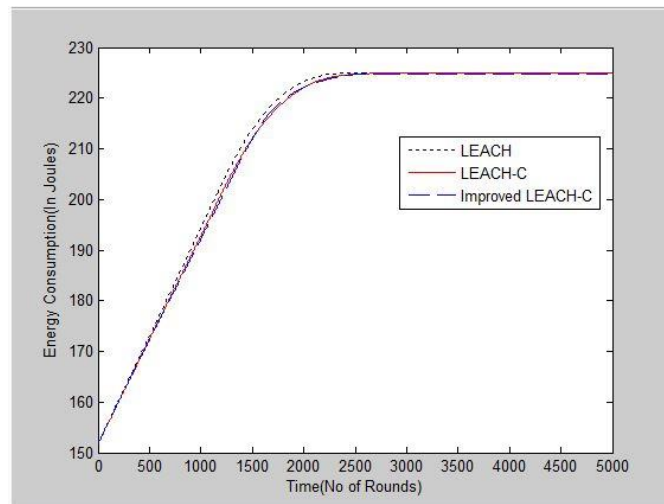


Figure 6.19 . Consommation d'énergie des différents protocoles pour un réseau de 300 nœuds

La figure 6.20 montre que le nombre total de cluster-heads par tour est plus élevé dans LEACH-C amélioré que LEACH et LEACH-C. Elle démontre que le protocole proposé offre un plus grand nombre de CH par rapport à LEACH.

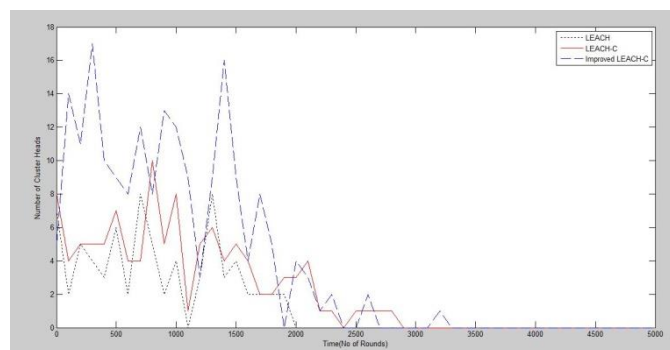


Figure 6.20 . Nombre de cluster-heads

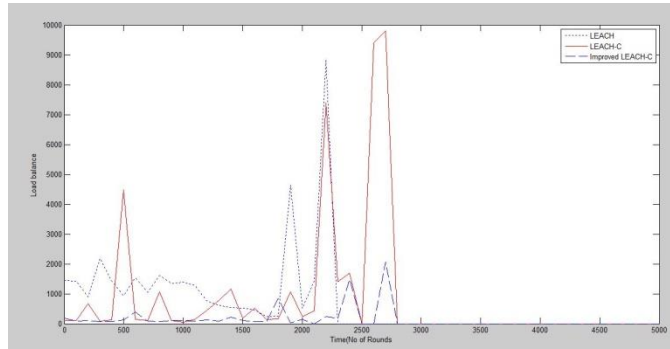


Figure 6.21 . Equilibre de la charge (Load balance)

La figure 6.21 correspond à l'équilibre de la charge, LEACH-C fait tant de surcharge lors de l'exploitation du cluster-head et c'est une faiblesse pour ce protocole. Le nouveau protocole LEACH-C amélioré a réussi à réduire le bilan de charge jusqu'à plus de 75% par rapport aux protocoles LEACH et LEACH-C.

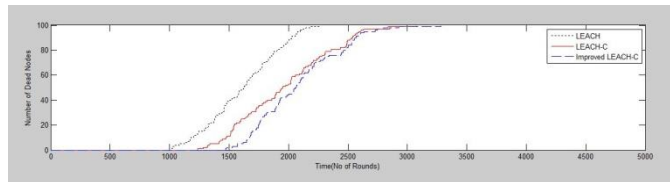


Figure 6.22 . Nœuds morts de différents protocoles pour un réseau de 100 nœuds

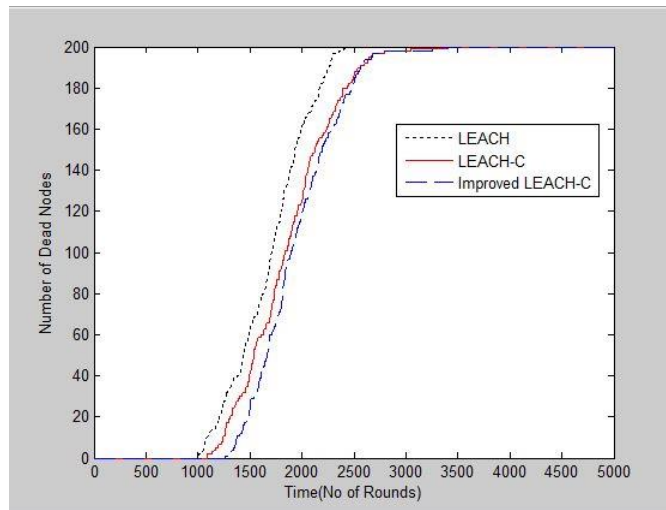


Figure 6.23 . Nœuds morts de différents protocoles pour un réseau de 200 nœuds

Les figures 6.22-6.24 présentent le nombre de nœuds morts par tour, ce qui montre que le nombre de nœuds morts est plus petit pour LEACH-C amélioré par rapport à LEACH-C et LEACH. À 2320s, tous les nœuds de LEACH sont morts, mais certains nœuds restent toujours en direct à 3250s dans LEACH-C amélioré. La durée de vie du réseau augmente dans LEACH-C amélioré.

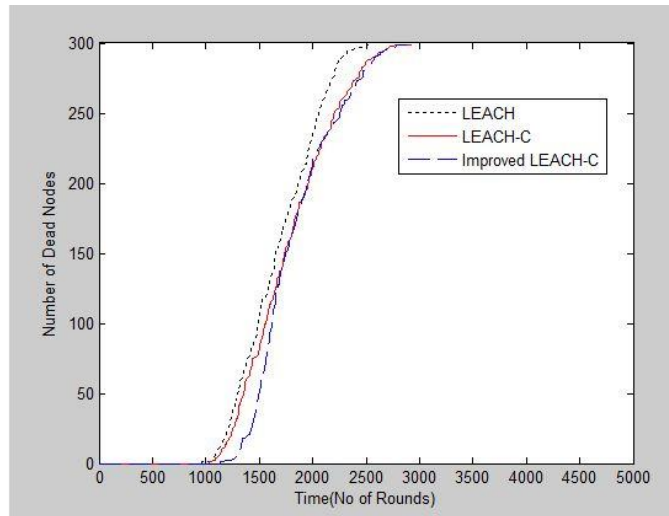


Figure 6.24. Nœuds morts de différents protocoles pour un réseau de 300 nœuds

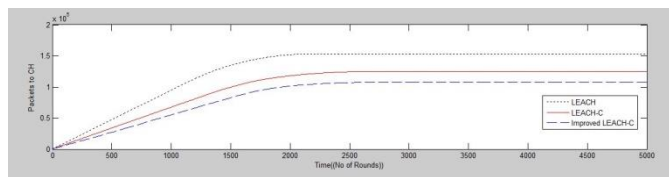


Figure 6.25 . Paquets au CH

La figure 6.25 montre que le nombre total de paquets envoyés au cluster-head par tour est moins élevé dans LEACH-C amélioré que LEACH et LEACH-C.

VI. Conclusion

Dans ce chapitre, on a proposé un protocole LEACH-C optimisé en étendant LEACH-C avec deux modifications. La première modification détermine la portée de la sélection du chef de cluster en fonction du niveau d'énergie des nœuds et la deuxième permet l'attribution des nœuds au chef de cluster le plus proche, uniquement si les nœuds déjà attribués à ce chef de cluster sont inférieurs à un nombre N . Dans la première approche, les chefs de cluster ne sont pas sélectionnés de façon aléatoire, et la seconde approche permet à chaque cluster d'avoir un nombre égal de nœuds qui améliore la durée de vie du réseau, la région stable et le débit du réseau de capteurs. Le schéma proposé équilibre la répartition de l'énergie dans le réseau. Par rapport à l'algorithme LEACH-C et LEACH, les résultats de la simulation démontrent qu'un nouveau schéma peut améliorer l'efficacité énergétique et prolonger le temps de survie du réseau. Notre recherche se concentre également sur l'amélioration des performances de LEACH et LEACH-C afin de minimiser le nombre de nœuds bloqués au fur et à mesure que les chefs de cluster et d'augmenter la durée de vie et le débit du réseau via l'équilibrage de la charge.

Conclusion générale et Travaux

Futures

I. Travaux de thèse

Avec le besoin toujours croissant de surveiller les différents phénomènes physiques, les réseaux de capteurs sans fil (WSNs) sont d'une grande utilité. Ces dernières années, ils ont acquis une recherche mondiale et un intérêt industriel, en particulier avec la prolifération des technologies de communication sans fil et de la technologie microélectronique, Micro-ElectroMechanical Systems (MEMS) qui a facilité le développement de capteurs intelligents. Ces capteurs sont des appareils microscopiques, à faible coût, à faible puissance et multifonctions qui empiètent certaines des propriétés du monde réel. L'objectif principal de WSN est d'observer, de collecter et de traiter la connaissance des nœuds capteurs dans la portée du réseau. Ainsi, l'élément fondamental le plus important dans un réseau WSN est le nœud capteur, qui est limité en termes de mémoire, d'énergie et de capacités de traitement. Précisément, chaque nœud capteur comprend la détection, le traitement, la transmission, le mobilisateur, le système de recherche de position et les unités de puissance (certains de ces composants sont facultatifs, comme le système de recherche de position). Ils sont généralement déployés de manière ad hoc dans un champ de capteur, qui est une zone où les nœuds de capteurs sont dispersés. Ces nœuds de capteurs dispersés ont la capacité de collecter et d'acheminer des données vers d'autres capteurs jusqu'à la destination, dite station de base (BS) ou Sink. Un nœud BS est capable de connecter le réseau de capteurs à une infrastructure de communication existante ou à Internet où un utilisateur peut avoir accès aux données signalées.

La nature Ad hoc et la vision «de déployer et de limiter» rendent les WSN la technologie la plus attrayante dans certaines applications. Les applications bien connues de WSN sont les applications militaires, les applications industrielles et les soins médicaux, les secours en cas de catastrophe, la surveillance de l'environnement, etc. Dans ces applications, nous avons besoin de nœuds pour pouvoir se localiser dans différents environnements. En effet, les données et les informations des nœuds des capteurs sont inutiles si ces nœuds ne connaissent pas leurs positions géographiques. Ainsi, donc, comment déterminer la localisation topographique des nœuds de capteurs alloués dans un réseau est souvent cité comme problème de localisation.

La localisation dans un WSN a captivé l'intérêt des chercheurs dans les quelques années. C'est parce que les applications WSN ne peuvent pas réussir si les utilisateurs ne parviennent pas à collecter l'information de position exacte des nœuds capteurs.

Dans cette thèse nous avons traité le problème de la localisation dans les réseaux ad-hoc de capteurs. Plus exactement, nos travaux permettent de présenter des protocoles et des algorithmes hybrides qui permettent de réduire les coûts de la localisation et augmenter les performances de localisation et de routage d'une part et de minimiser la consommation d'énergie et prolonger la durée de vie du réseau de capteurs d'autre part.

Pour atteindre cet objectif nous avons commencé par présenter un état d'art sur les principaux concepts et les protocoles de routage dédiés au réseau de capteur sans fils. On a pu porter une vue sur la communication dans les réseaux WSN en décrivant le standard 802.15.4. L'étude détaillée de cette norme sous OPNET a présenté des meilleurs performances pour un réseau à grande échelle comme les WSN. Ce qui a permis de choisir ce standard comme système de communication entre les nœuds capteurs.

Après avoir validé le choix du système de communication, on a étudié les différents protocoles de routage dédié aux réseaux de capteurs sans fil y compris les protocoles de routage géographique.

Les protocoles de routage géographiques sont ainsi mieux adaptés aux réseaux de capteur grâce à leur meilleure mise à l'échelle. En effet, les protocoles de routage topologiques souffrent d'une phase de découverte et de maintenance des chemins assez lourdes, alors que les protocoles de routage géographiques utilisent l'approche gloutonne pour transmettre les données graduellement de proche en proche jusqu'à atteindre la destination. Pour cela, ces protocoles nécessitent la position géographique de la destination. Cette dernière est obtenue grâce à des services distribués appelés services de localisation. Dans ces travaux, nous avons présenté une taxonomie des protocoles de routage géographiques qui sont utilisés dans les réseaux de capteurs sans fils. De plus, une classification des services de localisation est proposée.

Dans la littérature, les protocoles de routage géographiques sont généralement étudiés séparément des services de localisation même s'ils sont liés en pratique. Alors, nous avons proposé de les mixer dans des approches hybrides. L'approche hybride consiste à fusionner les protocoles de routage géographiques et les services de localisation afin de prendre en compte les problématiques de routage et de localisation simultanément, et par la suite de trouver le meilleur compromis entre le coût de la localisation et les performances du

roulage. Ainsi, au lieu de lancer des requêtes loin de la destination et attendre que les réponses traversent tout le réseau, les paquets de données sont transmis directement vers la dernière position connue de la destination. Une fois que l'on s'approche de cette position, une requête est initiée au plus près de la destination. Quand la position exacte est reçue, le protocole de routage est notifié et le paquet peut être dérouté vers la nouvelle position. Par conséquent, on évite l'inondation du réseau avec des requêtes et des réponses de localisation et donc la congestion du réseau. Les paquets sont transmis au plus tôt ce qui réduit les latences de bout-en-bout. Cette approche a été appliquée à GLS et Hierarchical Location Service (HLS) pour donner Hybrid Routing and Grid Location Service (GPSR+GLS) et Hybrid Routing and Hierarchical Location Service (GPSR+HLS). Ces services hybrides ont été comparés entre eux. La comparaison a démontré que l'approche hybride (GPSR+HLS) ne permettait pas que de réduire les coûts de la localisation, mais elle permettait aussi d'augmenter les performances de localisation et de routage.

On a aussi montré les concepts de base du protocole de routage hiérarchique en présentant des détails sur le protocole LEACH et ses différents descendants. On a présenté le modèle réseau de la consommation d'énergie dans les réseaux de capteurs. Enfin on a décrit notre protocole LEACH-C optimisé qui améliore l'efficacité énergétique et prolonge le temps de survie du réseau en proposant un protocole LEACH-C centralisé optimisé en étendant LEACH-C avec deux modifications.

- ✓ La première modification détermine la portée de la sélection du chef du cluster (Cluster-Head) en fonction du niveau d'énergie des nœuds.
- ✓ La seconde est l'attribution des nœuds au cluster-head le plus proche, uniquement si les nœuds déjà attribués à ce cluster-head sont inférieurs à un nombre N .

Dans La première approche, les cluster-heads ne sont pas sélectionnés de façon aléatoire, et la seconde approche permet à chaque cluster d'avoir un nombre égal de nœuds qui améliore la durée de vie du réseau, la région stable et le débit du réseau de capteurs. Le schéma proposé équilibre la répartition de l'énergie dans le réseau.

II. Travaux futures

Suite aux travaux effectués dans cette thèse, plusieurs pistes d'améliorations peuvent être envisagées. Premièrement, concernant les approches hybrides pour le routage et la localisation proposées GPSR+GLS et GPSR+HLS, une première piste d'amélioration serait

d'utiliser l'ancienne position d'un nœud pour envoyer les données. L'approche proposée repose essentiellement sur deux règles:

- L'ancienne position d'un nœud est utilisée pour envoyer les données même si elle n'est pas assez fraîche: Si un nœud a une ancienne position de la destination, cette position est utilisée pour transmettre les données dès que possible.
- En s'approchant de l'ancienne position de la destination, une requête de localisation locale est envoyée pour récupérer la nouvelle position exacte.

Actuellement, cette nouvelle approche hybride est en train d'être appliquée à GLS et HLS comme cela a été fait pour la première approche hybride dans le cadre de cette thèse. Cette approche dénommée sera comparée à la première dans des expérimentations similaires à celles réalisées pour comparer l'approche GPSR+GLS et GPSR+HLS.

En outre on peut utiliser la mobilité du nœud pour estimer sa position. L'approche hybride avec prédiction de mobilité, elle est basée sur l'approche hybride simple avec comme extension un algorithme d'estimation des positions futures d'un nœud grâce à des informations supplémentaires telles que la vitesse et la direction. Les effets de la mobilité peuvent ainsi être maîtrisés avec cette prédiction. En effet, le paquet de donnée est envoyé à la nouvelle position estimée et non plus à l'ancienne position. Ceci permet de réduire la distance sur laquelle le paquet est dérouté.

Finalement pour rendre le protocole LEACH-C optimisé plus performant, on est en train de travailler sur un algorithme qui permet de minimiser la consommation d'énergie dans la phase de communication de notre protocole optimisé surtout la phase dans laquelle les cluster-heads créent une séquence TDMA pour chaque nœud au sein des clusters. Dans cette phase, les nœuds qui ne sont pas des cluster-heads envoient leurs données collectées au cluster-head pendant leurs propres séquences TDMA. En autre temps, les nœuds restent en état de couchage pour économiser de l'énergie. Les cluster-heads traitent la fusion de données et les envoient à la BS. On essaie de trouver une amélioration qui permet de diminuer le temps de traitement et de fusion de données par les Cluster-heads afin de minimiser la consommation d'énergie et prolonger la durée de vie du réseau.

Annexe A : NS-2 : Tutoriel d'utilisation

I. Introduction

NS-2 comme présenté dans la thèse est un outil logiciel de simulation de réseaux informatiques à évènements discrets. Il est principalement bâti avec les idées de la conception par objets, de réutilisabilité du code et de modularité. C'est un logiciel libre qu'on retrouve facilement sur Internet. Le logiciel est exécutable sous Linux, MAC et Windows (en utilisant CygWin). Ici nous donnerons des astuces qui ne visent pas à remplacer la documentation de NS mais plutôt à présenter cet outil au lecteur avec un éclairage différent et plus général pour en faciliter la prise en main. NS-2 au départ a été conçu pour faciliter l'étude de l'interaction entre les protocoles et le comportement d'un réseau à différentes échelles. Pour cela, il contient des bibliothèques pour la génération de topologies réseaux, des trafics, ainsi qu'un outil de visualisation tel que l'animateur réseau NAM (network animator). Il est maintenant un outil bien adapté aux réseaux à communications de paquets et à la réalisation de simulations de petite taille. Il contient les fonctionnalités nécessaires à l'étude des algorithmes de routage uni point ou multipoint, des protocoles de transport, de session, de réservation, des services intégrés, des protocoles d'application comme HTTP. De plus le simulateur possède déjà une palette de systèmes de transmission, d'ordonnanceurs et de politiques de gestion de files d'attente pour effectuer des études de contrôle de congestion. On retrouve dans le tableau suivant les principaux composants disponibles dans NS par catégorie :

Application	Web, ftp, telnet, générateur de trafic (CBR, POISSON, ...)
Transport	TCP, UDP, RTP, SRM,...
Routage	DIFFUSION/RATE,DIFFUSION/-PROB,DSDV, DSR,FLOODING,OMNIMCAST,AODV,TORA AOMDV, PUMA
Gestion de file d'attente	RED, DropTail, Token bucket
Discipline de service	CBQ, SFQ, DRR, Fair queueing
Système de transmission	CSMA/CD, CSMA/CA, lien point à point

En combinant tous les composants, ces capacités ouvrent le champ à l'étude de nouveaux mécanismes au niveau des différentes couches de l'architecture réseau. NS est

devenu l'outil de référence pour les chercheurs du domaine.

II. Notions pour l'interpréteur

NS-2 est un langage écrit en C++, avec un interpréteur OTcl. Dans ce paragraphe nous allons donner les bases du langage Tcl, les principes de l'OTcl et les explications sur le mécanisme qui lie le C++ avec l'interpréteur Tcl.

2.1. Tcl

Tcl (Tool Command Language) est un langage de commande comme le shell UNIX mais qui sert à contrôler les applications. Il offre des structures de programmation telles que les boucles, les procédures ou les notions de variables. Il y a deux principales façons de se servir de Tcl : soit comme un langage autonome interprété ou comme une interface applicative d'un programme classique écrit en C ou C++. Toutes les applications qui utilisent Tcl créent et utilisent un interpréteur Tcl. Cet interpréteur est le point d'entrée standard de la bibliothèque. L'application telsh constitue une application minimale ayant pour but de familiariser un utilisateur au langage Tcl et ne comporte que l'interpréteur Tcl. On retrouve cet interpréteur dans l'application NS. Une fois la commande "ns" tapée, l'application effectue l'initialisation des objets puis passe en mode interactif où on peut alors commencer à entrer les commandes Tcl.

Concepts

Tcl est un langage non typé, chaque commande consiste en un ou plusieurs mots séparés par des espaces ou des tabulations. Tous les mots sont des chaînes de caractères. Le premier mot de la commande est le nom de la commande, les autres mots sont les arguments passés à la commande. Chaque commande Tcl retourne le résultat sous forme d'une chaîne de caractères. Le caractère de "retour à la ligne" termine une commande et lance son interprétation. Le caractère de séparation de plusieurs commandes sur une même ligne est ";".

A l'inverse du C ou C++, Tcl n'est pas un langage compilé, mais un langage interprété. Tcl évalue une commande en effectuant une analyse syntaxique et son exécution. L'analyse syntaxique consiste à identifier les mots et effectuer les substitutions. Durant cette étape, l'interpréteur ne fait que des manipulations de chaînes. Il ne traite pas la signification des mots. Pendant la phase d'exécution, l'aspect sémantique des mots est traité comme par exemple déduire du premier mot le nom de la commande, vérifier si la commande existe et appeler la procédure de cette commande

avec les arguments. Le backslash (\) et les groupages permettent d'insérer des caractères spéciaux dans les mots et d'écrire des commandes sur plusieurs lignes.

Exemple

```
set a 12          affecte la valeur 12 à la variable a
expr 2 + 3        calcule la valeur de l'expression 2 +
3 puts Coucou    affiche Coucou sur la sortie standard
```

Un tutoriel détaillé sur les commandes est disponible à cette adresse <http://wfr.tcl.tk/1140>.

2.2.OTcl

OTcl est une extension orientée objet de Tcl <ftp://ftp.tns.lcs.mit.edu/pub/otcl/>. Les commandes Tcl sont appelées pour un objet. En OTcl, les classes sont également des objets avec des possibilités d'héritage. Les correspondances avec le C++ sont :

- ✓ C++ a une unique déclaration de classe. En OTcl, les méthodes sont attachées à un objet ou à une classe.
- ✓ Les méthodes OTcl sont toujours appelées avec l'objet en préfixe.
- ✓ L'équivalent du constructeur et destructeur C++ en OTcl sont les méthodes `init{}` `destroy{}`.
- ✓ L'identification de l'objet lui-même : `this(C++)`, `$self (OTcl)`. `$self` s'utilise à l'intérieur d'une méthode pour référencer l'objet lui-même. A la différence de C++, il faut toujours utiliser `$self` pour appeler une autre méthode sur le même objet. C'est à dire "`$self xyz 5`" serait "`this->xyz(5)`" ou juste "`xyz(5)`" en C++.
- ✓ L'héritage multiple est possible dans les deux langages.

2.3.Lien C++ et Tcl

Construire une application avec un interpréteur Tcl revient à inclure une bibliothèque Tcl qui définit les commandes de bases de Tcl dans l'application. Comme nous l'avons dit, l'interpréteur effectue l'analyse syntaxique et appelle la fonction C correspondant à la commande Tcl. Ajouter une commande Tcl consiste à établir un lien entre un mot et une fonction C. Le mot sera le nom de la commande Tcl. La

fonction C est définie dans le code source de l'application. Au démarrage, l'application procède dans son main() aux initialisations nécessaires et passe la main à l'interpréteur. L'application passe en mode interactif : à chaque commande tapée par l'utilisateur, la fonction C correspondante est appelée afin de réaliser la commande demandée.

III. Installation

L'installation de NS-2 est ici détaillée pour une version d'Ubuntu 11.04. Selon moi, une installation via apt-get n'est pas adaptée si on compte modifier les fichiers C++. Il est plutôt conseillé de faire une installation via ns-allinone que l'on trouve facilement sur Google. La dernière version en date est la version **ns-allinone-2.35- RC10**. Elle contient l'essentiel pour faire fonctionner NS notamment :

- *ns*,
- *nam* (*visualisation*),
- *otcl-1.14*,
- *tcl8.5.8*,
- *tclcl-1.20*,
- *tk8.5.8*,
- *zlib-1.2.3*,
- *xgraph-12.2* (pour faire des graphiques).

Avant l'installation, il est nécessaire d'installer les dépendances nécessaires en effectuant cette commande.

sudo apt-get install build-essential autoconf automake libxmu-dev

Néanmoins à cause de l'évolution du compilateur gcc, plusieurs erreurs peuvent être observées pendant la compilation ou à l'exécution de NS. D'où il sera nécessaire dans certains cas d'installer une version antérieure de gcc et g++ ,la version 4.5.2 incluse dans la Ubuntu 11.04 fonctionne parfaitement sans erreur. N'oubliez pas d'installer g++... Si vous êtes amenés à changer la version du compilateur, il vous sera indispensable de modifier certains "Makefile.in" dans certains dossiers :

- **otcl-1.14/ : CC = gcc-4.5**
- **tclcl-1.20/ : CC = gcc-4.5**
- **tclcl-1.20/ : CPP = g++-4.5**
- **ns-2.35/ : CC = gcc-4.5**
- **ns-2.35/ : CPP = g++-4.5**

➤ **nam-1.15/ : CC = gcc-4.5**

➤ **nam-1.15/ : CPP = g++-4.5**

Il est conseillé d'installer NS dans le "home", pour cela décompressez le fichier ns-allinone-2.35-RC10 dans le home. D'autres bugs peuvent survenir pendant ou après l'installation de NS si c'est le cas il est nécessaire d'effectuer les changements pour enlever x : :X constructeur .

D'abord positionnez-vous dans le répertoire cd **\$HOME/ns-allinone-2.35-RC10**
puis :

➤ Editez le fichier ns-2.35/tools/ranvar.cc et changez la ligne 219 :

```
return GammaRandomV ariable :: GammaRandomV ariable(1.0+alpha , _beta  
.value()*pow (u, 1.0/alpha );
```

par

```
return GammaRandomV ariable(1.0+alpha_ , beta ).value()*pow (u, 1.0/alpha); _
```

➤ De même, changez les lignes 183 et 185 dans ns-2.35/mobile/nakagami.cc :

```
resultPower = ErlangRandomV ariable(P r/m, int m).value();
```

et

```
resultPower = GammaRandomV ariable(m, P r/m).value();
```

Une fois toutes ces modifications terminés, lancez l'installation en effectuant :

```
cd $HOME/ns-allinone-2.35-RC10  
sudo ./install
```

Annexe B : Descriptions des procédures de simulation NS-2

I. Génération du Trafic du réseau

Le trafic réseau est généré à l'aide de l'exécutable `./trafgen`. Il permet de créer différents types de traces réseau.

```
./trafgen -n <nodes> -a <active nodes>
        -t <simulation time> -c <connections>
        -m <mode: 0-query 1-cbr(old) 2-cbr(new)
        3-tcp(ftp) 4-ping 5-ping(memopt)>
        [-w <startup-wait time>]
        [-i <inverted timescale [1/0]>]
        [-r <send rate>] [-s <pkt size>]
        [-p <packets>] [-o <tcp window>]
```

Lancer des requêtes test :

Une requête de test est un nœud recherchant l'emplacement d'un autre. Si le nœud a une position fraîche, le cache de localisation est utilisé. Sinon, une demande de localisation est envoyée pour acquérir une nouvelle position. Un exemple utilisant l'exécutable `./trafgen` pour générer des requêtes de test peut être obtenu en exécutant cette commande:

```
./trafgen -n 100 -a 100 -t 105 -c 4 -m 0 -w 15 > QueriesT ests.tcl
```

The output le `QueriesT ests.tcl` will be as follow:

```
#
# nodes: 100, max conn: 4, send rate: 0.100000, seed: 1, active nodes: 100
#
```

```

$ns_ at 15.157052578469 "[$node_(65) set ragent_] test-query 5"
$ns_ at 15.161809181396 "[$node_(51) set ragent_] test-query 86"
.
$ns_ at 97.842091824604 "[$node_(26) set ragent_] test-query 12"
$ns_ at 97.924068041390 "[$node_(33) set ragent_] test-query 80"

```

Dans cet exemple, chaque nœud (parmi les 100) lance des requêtes de test pour 4 autres nœuds. La fonction de test-requête fonctionne comme suit :

"\$ns_ at T"[\$node_(S) set ragent_] test - query D" ", Où T est le temps, S est la source de la requête de test et D est le nœud de destination. Par exemple, la première commande signifie que le nœud 65 lance une requête de test au 15.157052578469s pour récupérer la position du nœud 5.

Envoi de paquets CBR :

Le trafic (CBR) peut être généré à l'aide de cette commande:

```

"./trafgen - n 100 - a 100 - t 100 - c 4 - m 2 - w 2 - i 1 - r 1 - s 128 - p 100 >
CBRT raffic.tcl".

```

Le fichier de sortie "*CBRT raffic.tcl*" apparaît comme suit:

```

#
# nodes: 100, max conn: 4, send rate: 1.000000, seed: 1, active nodes: 100
#
# 99.986179128699 7 -> 37
#
set udp_(143) [new Agent/UDP]
$ns_ attach-agent $node_(7) $udp_(143)
set null_(143) [new Agent/Null]
$ns_ attach-agent $node_(37) $null_(143)
set cbr_(143) [new Application/Traffic/CBR]

```



```
$cbr_(143) set packetSize_ 128  
$cbr_(143) set interval_ 1.000000  
$cbr_(143) set random_ 0  
$cbr_(143) set maxpkts_ 100  
$cbr_(143) attach-agent $udp_(143)  
$ns_ connect $udp_(143) $null_(143)  
$ns_ at 99.9861791286985806 "$cbr_(143) start"
```

Dans cet exemple, le nœud 7 lance un agent UDP 143 responsable de l'envoi des paquets CBR au nœud 37 à 99.986179128699 s. Les paquets de taille 128 octets sont envoyés chaque seconde pendant 100 fois.

II. Exécution sous NS2

L'exécution de la simulation sous NS-2 se fait avec le script run.tcl fourni avec le patch HLS. Ce script peut être utilisé avec les options suivantes:

Usage: ns run.tcl -out tracefile

NS Options:

- nn [number of nodes]
- stop [simulation duration in secs]
- x / -y [dimension in meters]
- adhocRouting [routing protocol to use]
- use_gk [radius for gridkeeper usage]
- zip [(0/1) should tracefiles be zipped on-the-fly]
- cc [alpha for congestion control ((MAC802_11 only))]
- ifqlen [max packets in interface queue]

File Options:

-cp [traffic pattern]
-sc [scenario file]
-nam [nam tracefile]
-on_off [wake/sleep pattern]
-lt [load trace file (MAC802_11 only)]
-pingLog [log file for ping statistics (Ping Traffic only)]

MAC Options:

-rr [radio range in meters]
-bw [link/dataRate bandwidth in bits/sec]
-bs [basicRate bandwidth in bits/sec]

GPSR Options:

-bint [beacon interval (and beacon expiry)]
-use_planar [(0/1) planarize graph]
-use_peri [(0/1) use perimeter mode]
-use_mac [(0/1) use mac callback]
-verbose [(0/1) be verbose]
-use_beacon [(0/1) use beacons at all (disable beacons with 0)]
-use_reactive [(0/1) use reactive beaconing]
-locs [locservice to use (0-Omni/1-RLS/2-GLS/3-HLS)]
-use_loop [(0/1) use loop detection]
-ed [topology file (edges)]
-ve [topology file (verteces)]

Voici un exemple de commande pour faire fonctionner HLS : `ns run.tcl -out hls_trace.txt -sc sc-x2000-y2000-n100-s30-t100-MRD -cp cp-n100-c400-15-98.tcl -nn 100 -locs 3 -use_peri 1 -x 2000 -y 2000 -mac_emu 0 -stop 100 -zip 0`

En fait, il suffit de changer l'option « `-locs` » pour choisir le service de localisation :

- 0 : Omniscient

- 1 : RLS

- 2 : GLS

- 3 : HLS

Ici on a choisi de travailler avec GLS et HLS.

III. Tracés d'évaluations

Il y a deux fichiers générés dans les simulations NS-2: le tracé de sortie (output trace) "hls_trace.tr" et le tracé (trace) "hls_namtrace.nam". Le tracé nam est utilisé avec l'outil d'Animateur réseau (NAM) (Figure 8.1) fourni avec NS-2 pour visualiser la simulation.

L'exécution de cet outil est lancé par: `nam hls_namtrace.nam &`

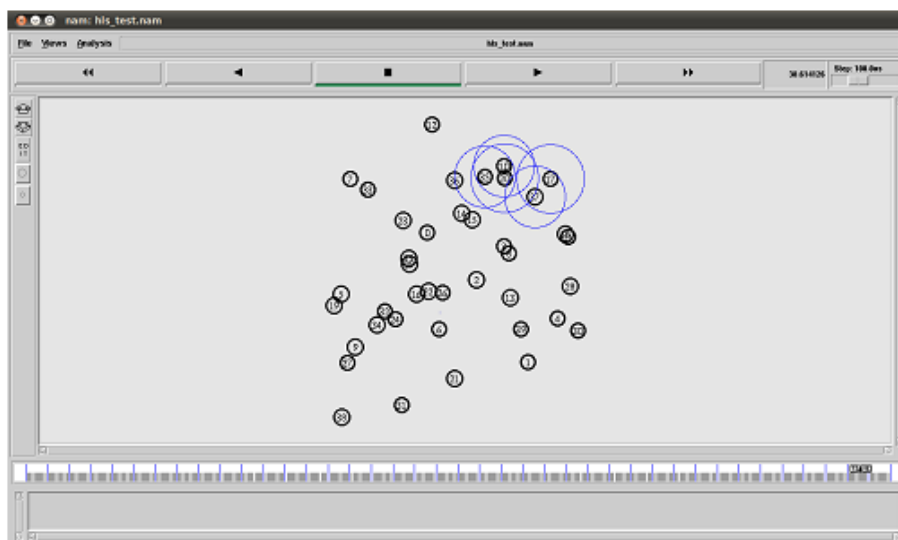


Figure 8.26. NAM: Network ANimator tool

Le script `evaluate.pl`, fourni avec le patch HLS, permet d'extraire de la sortie "hls_trace.tr" beaucoup de statistiques de localisation. Il peut être utilisé comme suit:

```
./evaluate.pl -f hls_trace.tr > hls-results.tr
```

Le script calcule le nombre de requêtes de localisation (location queries), les recherches de cache (cache lookups), les requêtes envoyées (sent requests), réponses reçues (received replies), les requêtes abandonnées (dropped requests), les réponses abandonnées (dropped replies), transferts, etc. Il donne également un grand nombre de statistique de livraison de paquets tels que le nombre des paquets envoyés, transmis, reçus ou perdus. En outre, il estime la bande passante consommée par le routage et les couches MAC. Le script a été étendu afin d'évaluer plus de paramètres de performances de routage tels que le taux de livraison de paquets (Packet Delivery Rate PDR), la latence moyenne de bout en bout (end-to-end latency) et le nombre moyen de sauts pour les paquets CBR.

La ligne suivante est extraite du fichier de sortie d' NS-2 "hls_trace.tr". La première lettre de la ligne spécifie le type de paquet (s: pour envoyer (sent), r: pour reçus (received), f: pour transmis (forwarded), et D: pour perdu (dropped)). Ainsi, le paquet dans cet exemple est un paquet envoyé. Le second champ représente la date de l'événement. Le troisième est le nœud où l'événement survient. Puis, le quatrième champ décrit la couche (ici dans cet exemple: la couche application AGT). Le cinquième champ permet de définir la raison de perte (ici le paquet n'a pas perdu).

```
s 5.000341235 _1_ AGT --- 540 cbr 128 [0 0 0 0] ----- [1:0 45:0 32 0]
```

L'ID du paquet est le suivant, et il est ici de 540. Le protocole utilisé est le CBR. La taille du paquet est 128 B. Ensuite, les quatre chiffres entre parenthèses sont liés aux adresses de la couche Media Access Control (MAC) (la durée du paquet dans l'en-tête de la couche MAC, l'adresse MAC de la destination, l'adresse MAC de la source et le type MAC de corps du paquet). Les autres sont le nœud IP de source et le numéro de port de source (1: 0), l'adresse IP du nœud de destination (-1 signifie diffusion) et le numéro de port de destination (45: 0), l'en-tête IP ttl (32) et l'adresse IP du prochain saut (0 signifie nœud 0).

Par conséquent, cette structure de ligne est utilisée pour calculer le nombre de paquets envoyés et reçus et donc le PDR. De plus, la latence moyenne de bout en bout et le nombre moyen de sauts sont calculés de la même manière.

Références

- [1] Akyildiz, I. F., Su, W., Sankarasubramaniam, Y., and Cayirci, E. "A survey on sensor networks". *IEEE Communication Magazine*, vol. 40, no. 8, Aug. 2002, pp. 102-114.
- [2] Culler, D., Estrin, D., and Srivastava, M. "Guest editors' introduction: overview of sensor networks". *Computer*, vol. 37, no. 8, Aug. 2004, pp. 41-49.
- [3] Hill, J., Horton, M., Kling, R., and Krishnamurthy, L. "The platforms enabling wireless sensor networks". *Communications of the ACM*, vol. 47, no. 6, Jun. 2004, pp. 41-46.
- [4] Pister, S. J. "Military Applications of Sensor Networks". Institute for Defense Analyses Paper P-3531, Defense Science Study Group. 2000.
- [5] Simon, G., Maróti, M., Lédecz, A., Balogh, G., Kusy, B., Nádás, A., Pap, G., Sallai, J., and Frampton, K. "Sensor network-based countersniper system". In *Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems (SenSys '04)*, Baltimore, MD, USA, Nov. 03-05, 2004. pp. 1-12.
- [6] Arora, A., Dutta, P., Bapat, S., Kulathumani, V., Zhang, H., Naik, V., Mittal, V., Cao, H., Demirbas, M., Gouda, M., Choi, Y., Herman, T., Kulkarni, S., Arumugam, U., Nesterenko, M., Vora, A., and Miyashita, M. "A line in the sand: a wireless sensor network for target detection, classification, and tracking". *Computer Networks*, vol. 46, no. 5, Dec. 2004, pp. 605-634.
- [7] Lédecz, A., Nádás, A., Volgyesi, P., Balogh, G., Kusy, B., Sallai, J., Pap, G., Dóra, S., Molnár, K., Maróti, M., and Simon, G. "Countersniper system for urban warfare". *ACM Transactions on Sensor Networks*, vol. 1, no.2, Nov. 2005, pp. 153-177.
- [8] He, T., Krishnamurthy, S., Luo, L., Yan, T., Gu, L., Stoleru, R., Zhou, G., Cao, Q., Vicaire, P., Stankovic, J. A., Abdelzaher, T. F., Hui, J., and Krogh, B. "Vigilnet: an integrated sensor network system for energy-efficient surveillance". *ACM Transactions on Sensor Networks*, vol. 2, no. 1, Feb. 2006, pp. 1-38.
- [9] Volgyesi, P., Balogh, G., Nadas, A., Nash, C. B., and Ledecz, A. "Shooter localization and weapon classification with soldier-wearable networked sensors". In *Proceedings of the 5th international Conference on Mobile Systems, Applications and Services (MobiSys '07)*, San Juan, Puerto Rico, Jun. 11-13, 2007, pp. 113-126.
- [10] Choi, S., Kim, B., Park, J., Kang, C., Eom, D. "An implementation of wireless sensor network". *IEEE Transactions on Consumer Electronics*, vol. 50, no. 1, Feb. 2004, pp. 236-244. 153.
- [11] Glidden, R., Bockorick, C., Cooper, S., Diorio, C., Dressler, D., Gutnik, V., Hagen, C., Hara, D., Hass, T., Humes, T., Hyde, J., Oliver, R., Onen, O., Pesavento, A., Sundstrom, K., Thomas, M. "Design of ultra-low-cost UHF RFID tags for supply chain applications", *IEEE Communications Magazine*, vol. 42, no. 8, Aug. 2004, pp. 140-151.
- [12] Morais, R., Fernandes, M. A., Matos, S. G., Serdio, C., Ferreira, P. J., and Reis, M. J. "A ZigBee multi-powered wireless acquisition device for remote sensing applications in precision viticulture". *Computers and Electronics in Agriculture*, vol. 62, no. 2, Jul. 2008, pp. 94-106.
- [13] Eriksson, J., Girod, L., Hull, B., Newton, R., Madden, S., and Balakrishnan, H. "The pothole patrol: using a mobile sensor network for road surface monitoring". In *Proceeding of the 6th international Conference on Mobile Systems, Applications, and Services (MobiSys '08)*, Breckenridge, CO, USA, Jun. 17-20, 2008, pp. 29-39.

- [14] Gungor, V. C., Hancke, G. P. "Industrial wireless sensor networks: challenges, design principles, and technical approaches". IEEE Transactions on Industrial Electronics, vol. 56, no. 10, Oct. 2009, pp. 4258-4265.
- [15] Xu, N., Rangwala, S., Chintalapudi, K. K., Ganesan, D., Broad, A., Govindan, R., and Estrin, D. "A wireless sensor network for structural monitoring". In Proceedings of the 2nd international conference on embedded networked sensor systems (SenSys '04), Baltimore, MD, USA, Nov. 03-05, 2004, pp. 13-24.
- [16] Chintalapudi, K., Fu, T., Paek, J., Kothari, N., Rangwala, S., Caffrey, J., Govindan, R., Johnson, E., and Masri, S. "Monitoring Civil Structures with a Wireless Sensor Network". IEEE Internet Computing, vol. 10, no. 2, Mar. 2006, pp. 26-34.
- [17] Kim, S., Pakzad, S., Culler, D., Demmel, J., Fenves, G., Glaser, S., and Turon, M. 2007. "Health monitoring of civil infrastructures using wireless sensor networks". In Proceedings of the 6th international Conference on information Processing in Sensor Networks (IPSN '07), Cambridge, Massachusetts, USA, Apr. 25-27, 2007, pp. 254-263.
- [18] Chebrolu, K., Raman, B., Mishra, N., Valiveti, P. K., and Kumar, R. "Brimon: a sensor network system for railway bridge monitoring". In Proceeding of the 6th international Conference on Mobile Systems, Applications, and Services (MobiSys '08), Breckenridge, CO, USA, Jun. 17-20, 2008, pp. 2-14.
- [19] Stajano, F., Hout, N., Wassell, I., Bennett, P., Middleton, C., and Soga, K. 2010. "Smart bridges, smart tunnels: transforming wireless sensor networks from research prototypes into robust engineering infrastructure". Ad Hoc Networks, vol. 8, no. 8, Nov. 2010, pp. 872-888.
- [20] Juang, P., Oki, H., Wang, Y., Martonosi, M., Peh, L. S., and Rubenstein, D. "Energyefficient computing for wildlife tracking: design tradeoffs and early experiences with ZebraNet". In Proceedings of the 10th international Conference on Architectural Support For Programming Languages and Operating Systems (ASPLOS-X), San Jose, California, Oct. 05-09, 2002, pp. 96-107.
- [21] Szewczyk, R., Osterweil, E., Polastre, J., Hamilton, M., Mainwaring, A., and Estrin, D. "Habitat monitoring with sensor networks". Communications of the ACM, vol. 47, no. 6, Jun. 2004, pp. 34-40. 154.
- [22] Tolle, G., Polastre, J., Szewczyk, R., Culler, D., Turner, N., Tu, K., Burgess, S., Dawson, T., Buonadonna, P., Gay, D., and Hong, W. "A microscope in the redwoods". In Proceedings of the 3rd international Conference on Embedded Networked Sensor Systems (SenSys '05), San Diego, California, USA, Nov. 02-04, 2005, pp. 51-63.
- [23] Werner-Allen, G., Lorincz, K., Welsh, M., Marcillo, O., Johnson, J., Ruiz, M., and Lees, J. "Deploying a wireless sensor network on an active volcano". IEEE Internet Computing, vol. 10, no. 2, Mar. 2006, pp. 18-25.
- [24] Mo, L., He, Y., Liu, Y., Zhao, J., Tang, S., Li, X., and Dai, G. "Canopy closure estimates with GreenOrbs: sustainable sensing in the forest". In Proceedings of the 7th International Conference on Embedded Networked Sensor Systems (SenSys '09), Berkeley, CA, USA, Nov. 04-06, 2009, pp. 99-112.
- [25] Tokekar, P., Bhaduria, D., Studenski, A., and Isler, V. "A robotic system for monitoring carp in Minnesota lakes". Journal of Field Robotics, 2010, to appear.
- [26] Schwiebert, L., Gupta, S. K., and Weinmann, J. "Research challenges in wireless networks of biomedical sensors". In Proceedings of the 7th Annual international Conference on Mobile Computing and Networking (MobiCom '01), Rome, Italy, Jul. 16-21, 2001, pp. 151-165.

- [27] Varshney, U. "Pervasive healthcare". *Computer*, vol. 36, no. 12, Dec. 2003, pp. 138- 140.
- [28] Efstratiou, C., Davies, N., Kortuem, G., Finney, J., Hooper, R., and Lowton, M. "Experiences of designing and deploying intelligent sensor nodes to monitor hand-arm vibrations in the field". In *Proceedings of the 5th international Conference on Mobile Systems, Applications and Services (MobiSys '07)*, San Juan, Puerto Rico, Jun. 11-13, 2007, pp. 127-138.
- [29] Iyengar, S., Bonda, F. T., Gravina, R., Guerrieri, A., Fortino, G., and SangiovanniVincentelli, A. "A framework for creating healthcare monitoring applications using wireless body sensor networks". In *Proceedings of the ICST 3rd international Conference on Body Area Networks*, Tempe, Arizona, Mar. 13-17, 2008, pp. 1-2.
- [30] Lorincz, K., Chen, B., Challen, G. W., Chowdhury, A. R., Patel, S., Bonato, P., and Welsh, M. 2009. "Mercury: a wearable sensor network platform for high-fidelity motion analysis". In *Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems (SenSys '09)*, Berkeley, California, USA, Nov. 04-06, 2009, pp. 183-196.
- [31] Shih, E. I., Shoeb, A. H., and Guttag, J. V. "Sensor selection for energy-efficient ambulatory medical monitoring". In *Proceedings of the 7th international Conference on Mobile Systems, Applications, and Services (MobiSys '09)*, Krak'ow, Poland, Jun. 22-25, 2009, pp. 347-358.
- [32] Kahn, J. M., Katz, R. H., and Pister, K. S. "Next century challenges: mobile networking for 'Smart Dust'". In *Proceedings of the 5th Annual ACM/IEEE international Conference on Mobile Computing and Networking (MobiCom '99)*, Seattle, Washington, USA, Aug. 15-19, 1999, pp. 271-278. 155.
- [33] Intille, S. S. "Designing a home of the future". *IEEE Pervasive Computing*, vol. 1, no. 2, Jun. 2002, pp. 76-82.
- [34] Yap, K., Srinivasan, V., and Motani, M. "MAX: Wide area human-centric search of the physical world". *ACM Transactions on Sensor Networks*, vol. 4, no. 4, Aug. 2008, pp. 1-34.
- [35] Lin, C., Wang, S., Kuo, S., and Chen, C. "Increasing service availability in a wireless home network environment". *The Computer Journal*, vol. 52, no. 8, Nov. 2009, pp. 851-860.
- [36] Hansen, E. K. "Denmark's Net-Zero-Energy Home". *IEEE Spectrum*, August 2010 issue, pp. 34-38.
- [37] Jeong, J., Guo, S., He, T., Du, D. "APL: autonomous passive localization for wireless sensors deployed in road networks". In *Proceedings of the 27th IEEE Conference on Computer Communications (INFOCOM '08)*, Phoenix, AZ, USA, Apr. 13-18, 2008, pp. 583-591.
- [38] Li, X., Shu, W., Li, M., Huang, H., Luo, P., Wu, M. "Performance evaluation of vehiclebased mobile sensor networks for traffic monitoring". *IEEE Transactions on Vehicular Technology*, vol. 58, no.4, May 2009, pp. 1647-1653.
- [39] Tubaishat M., Zhuang, P., Qi, Q., and Shang, Y. "Wireless sensor networks in intelligent transportation systems". *Wireless communications and mobile computing*, vol. 9, no. 3, Mar. 2009, pp. 287302.
- [40] Semertzidis, T., Dimitropoulos, K., Koutsia, A., Grammalidis, N. "Video sensor network for real-time traffic monitoring and surveillance". *IET Intelligent Transport Systems*, vol. 4, no. 2, Jun. 2010, pp.103-112.
- [41] Harrison, C., Eckman, B., Hamilton, R., Hartswick, P., Kalagnanam, J., Paraszczak, J., Williams, P. "Foundations for smarter cities". *IBM Journal of Research and Development*, vol. 54, no. 4, Aug. 2010, pp. 1-16.

- [42] Basagni, S., Chlamtac, I., Syrotiuk, V. R., and Woodward, B. A. "A distance routing effect algorithm for mobility (DREAM)". In Proceedings of the 4th Annual ACM/IEEE international Conference on Mobile Computing and Networking (MobiCom '98), Dallas, Texas, USA, Oct. 25-30, 1998, pp. 76-84.
- [43] Karp, B. and Kung, H. T. "GPSR: greedy perimeter stateless routing for wireless networks". In Proceedings of the 6th Annual international Conference on Mobile Computing and Networking (MobiCom '00), Boston, Massachusetts, USA, Aug. 06-11, 2000, pp. 243-254.
- [44] Kim, Y., Govindan, R., Karp, B., and Shenker, S. "Geographic routing made practical". In Proceedings of the 2nd Conference on Symposium on Networked Systems Design and Implementation (NSDI '05), May 02-04, 2005, Berkeley, CA, USA, pp. 217-230.
- [45] Alzoubi, K., Li, X., Wang, Y., Wan, P., and Frieder, O. "Geometric Spanners for Wireless Ad Hoc Networks". IEEE Transactions on Parallel and Distributed Systems, vol. 14, no. 4, Apr. 2003, pp. 408-421. 156.
- [46] Li, N. and Hou, J. C. "Localized topology control algorithms for heterogeneous wireless networks". IEEE/ACM Transactions on Networking, vol. 13, no.6, Dec. 2005, pp. 1313- 1324.
- [47] Xu, Y., Heidemann, J., and Estrin, D. "Geography-informed energy conservation for Ad Hoc routing". In Proceedings of the 7th Annual international Conference on Mobile Computing and Networking (MobiCom '01), Rome, Italy, Jul. 16-21, 2001, pp. 70-84.
- [48] Li, J. H. and Yu, M. "Sensor coverage in wireless ad hoc sensor networks". International Journal of Sensor Networks, vol. 2, no. 3/4, Jun. 2007, pp. 218-229.
- [49] Kalman, R. E. "A New Approach to Linear Filtering and Prediction Problems", Transactions of the ASME Journal of Basic Engineering, 82 (Series D), 1960, pp. 35-45.
- [50] Hu, Y., Perrig, A., Johnson, D. B. "Packet leashes: a defense against wormhole attacks in wireless networks". In the Proceeding of the 22nd Annual Joint Conference of the IEEE Computer and Communications (INFOCOM '03), San Francisco, CA, USA, Mar. 30-Apr. 3, 2003, pp. 1976-1986.
- [51] Zhong, Z., Hu, A., and Wang, D. "Physical layer design of wireless sensor network nodes", Journal of Southeast University, English Edition, vol. 22, no. 1, 2006, pp. 21-25.
- [52] Lazos, L. and Poovendran, R. "SeRLoc: Robust localization for wireless sensor networks". ACM Transactions on Sensor Networks, vol. 1, no. 1, Aug. 2005, pp. 73-100.
- [53] Estrin, D., Govindan, R., Heidemann, J., and Kumar, S. "Next century challenges: scalable coordination in sensor networks". In Proceedings of the 5th Annual ACM/IEEE international Conference on Mobile Computing and Networking (MobiCom '99), Seattle, Washington, USA, Aug. 15-19, 1999, pp. 263-270.
- [54] Patwari, N., Ash, J. N., Kyperountas, S., Hero, A. O. III, Moses, R. L., Correal, N. S. "Locating the nodes: cooperative localization in wireless sensor networks". IEEE Signal Processing Magazine, vol. 22, no. 4, Jul. 2005, pp. 54-69.
- [55] Sayed, A. H., Tarighat, A., Khajehnouri, N. "Network-based wireless location: challenges faced in developing techniques for accurate wireless location information". IEEE Signal Processing Magazine, vol. 22, no. 4, Jul. 2005, pp. 24-40.
- [56] Terzis, A., Anandarajah, A., Moore, K., and Wang, I. "Slip surface localization in wireless sensor networks for landslide prediction". In Proceedings of the 5th international Conference on information Processing in Sensor Networks (IPSN '06), Nashville, Tennessee, USA, April 19-21, 2006, pp. 109-116.

- [57] Benbadis, F., Friedman, T., de Amorim, M. D., Fdida, S. "GPS-free positioning system for wireless sensor networks". In Proceedings of the 2nd IFIP International Conference on Wireless and Optical Communications Networks (WOCN '05), Mar. 2005, pp. 541- 545.
- [58] Stoleru, R., He, T., and Stankovic, J. A. "Walking GPS: a practical solution for localization in manually deployed wireless sensor networks". In Proceedings of the 29th Annual IEEE international Conference on Local Computer Networks (LCN '04), Nov. 16-18, 2004, Washington, DC, USA, pp. 480-489.
- [59] Mi, Q., Stankovic, J. A., and Stoleru, R. "Secure walking GPS: a secure localization and key distribution scheme for wireless sensor networks". In Proceedings of the 3rd ACM Conference on Wireless Network Security (WiSec '10), Hoboken, New Jersey, USA, March 22-24, 2010, pp. 163-168.
- [60] Gandhi, S., Suri, S., and Welzl, E. "Catching elephants with mice: Sparse sampling for monitoring sensor networks". ACM Transactions on Sensor Networks, vol. 6, no. 1, Dec. 2009, pp. 1-27.
- [61] Arora, A., Ramnath, R., Ertin, E., Sinha, P., Bapat, S., Naik, V., Kulathumani, V., Hongwei Zhang., Hui Cao., Sridharan, M., Kumar, S., Seddon, N., Anderson, C., Herman, T., Trivedi, N., Nesterenko, M., Shah, R., Kulkarni, S., Aramugam, M., Limin Wang., Gouda, M., Choi, Y., Culler, D., Dutta, P., Sharp, C., Tolle, G., Grimmer, M., Ferriera, B., Parker, K. "ExScal: elements of an extreme scale wireless sensor network". In Proceedings of the 11th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications (RTCAS '05), Hong Kong, China, Aug. 17-19, 2005, pp. 102-108.
- [62] Estrin, D., Govindan, R., and Heidemann, J. "Embedding the Internet: introduction". Communications of the ACM, vol. 43, no. 5, May. 2000, pp. 38-41.
- [63] Min, R., Bhardwaj, M., Seong-Hwan Cho., Ickes, N., Shih, E., Sinha, A., Wang, A., Chandrakasan, A. "Energy-centric enabling technologies for wireless sensor networks". IEEE Wireless Communications, vol. 9, no. 4, Aug. 2002, pp. 28-39.
- [64] Kumar, V. 2003. "Sensor: the atomic computing particle". ACM SIGMOD Record, vol. 32, no. 4, Dec. 2003, pp. 16-21.
- [65] Hu, W., Bulusu, N., Chou, C. T., Jha, S., Taylor, A., and Tran, V. N. "Design and evaluation of a hybrid sensor network for cane toad monitoring". ACM Transactions on Sensor Networks, vol. 5, no. 1, Feb. 2009, pp. 1-28.
- [66] Song, W., Huang, R., Xu, M., Ma, A., Shirazi, B., and LaHusen, R. "Air-dropped sensor network for real-time high-fidelity volcano monitoring". In Proceedings of the 7th international Conference on Mobile Systems, Applications, and Services (MobiSys '09), (Krak'ow, Poland, Jun. 22-25, 2009, pp. 305-318.
- [67] S. K. Das, J. Wang, R. K. Ghosh, and R. Reiger. Theoretical Aspects of Distributed Computing in Sensor Networks, chapter 9: Algorithmic Aspects of Sensor Localization, pages 257–259. Springer, first edition, 2011.
- [68] Opnet, <http://www.opnet.com/>, 2012.
- [69] Heinzelman, W.R., Chandrakasan, A. "Energy efficient Communication Protocol for Wireless Microsensor Networks. In: IEEE Computer Society Proceedings of the 33rd Hawaii International Conference on System Sciences (HICSS '00), Vol. 8, pp. 8020, January, 2000.
- [70] Frey, Hannes, Stefan Rührup, and Ivan Stojmenović. "Routing in wireless sensor networks." Guide to Wireless Sensor Networks. Springer London, 2009. 81 -111.

- [71] Heinzelman, W.R., A.Chandrakasan,H.Balakrishnan.”An application Specific Protocol Architecture for Wireless Mincrosensor Networks.-IEEE Transactions on Wireless Communication, Vol.1, 2002,No 4,pp.660-670.
- [72]Su, Y. Sankarasubramaniam, E. Cayirci, Akyildiz, I.F.,W. "A Survey on Sensor Networks", IEEE Communications Magazine, August, 102-114(2002).
- [73] L. Gavrilovska; S. Krco; V.Milutinovic ; I. Stojmenovic ; R. Trobec “ Application and Multidisciplinary Aspects of Wireless Sensor Networks”, Springer, 2011.
- [74] B. Krishnamachari, “Networking Wireless Sensors”, Cambridge University Press, 2005.
- [75] Ning Xu, “A survey of sensor network applications”, IEEE communications magazine, vol 40, 2002.
- [76]Vernon S. Somerset, “Intelligent and Biosensors, Edited by Vernon S. Somerset”, Intech, January 2010.
- [77]F. Brissaud ,D. Charpentier, A. Barros et C. Bérenguer, « capteurs intelligents : nouvelles technologies et nouvelles problématiques pour la sureté de fonctionnement », Maîtrise des Risques et de Sûreté de Fonctionnement, Lambda-Mu 16, Avignon : France (2008).
- [78]A. Montoya, D. C. Restrepo et D.A. Ovalle, "Artificial Intelligence for Wireless Sensor Networks Enhancement”,InTech,2010.
- [79]QinghuaWang et Ilango Balasingham,”Wireless Sensor Networks – An Introduction” Wireless Sensor Networks: Application-Centric Design, Geoff V Merrett and Yen Kheng Tan (Ed.), ISBN: 978-953-307-321-7, InTech, 2010.
- [80]Ben L. Titzer et J. Palsberg , “Nonintrusive Precision Instrumentation of Microcontroller Software” , ACM, New York, NY, ETATS-UNIS , Volume 40 Issue 7, Pages 59 – 68, July 2005.
- [81]H. Karl et A. Willig, “Protocols and Architectures for Wireless Sensor Networks”, John Wiley & Sons, 2005
- [82]Gaurav Jolly, Mustafa C. Kuşçu, Pallavi Kokate et Mohamed Younis, “A Low- Energy Key Management Protocol for Wireless Sensor Networks”, ,Proceedings of the Eighth IEEE International Symposium on Computers and Communication ,(ISCC’03), IEEE COMPUTER SOCIETY,2003
- [83]S. Tilak, N. B. Abu-Ghazaleh et W. B. Heinzelman, « A taxonomy of wireless micro- sensor network model », ACM SIGMOBILE Mobile Computing and Communications Review, vol. 6,no. 2, pp. 28-36, April 2002.
- [84]H. Karl et A. Willig, “Protocols and Architectures for Wireless Sensor Networks”, John Wiley & Sons, 2005
- [85]Shio Kumar Singh et al, “Applications, Classifications, and Selections of Energy- Efficient Routing Protocols for Wireless Sensor Networks”, INTERNATIONAL JOURNAL OF ADVANCED ENGINEERING SCIENCES AND TECHNOLOGIES (IJAEST), Vol No.1, Issue No. 2, p85 – 95
- [86]V. R. Syrotiuk, B. Li, and A. M. Mielke, “Heterogeneous Wireless Sensor Networks” in Algorithms and Protocols for Wireless Ad hoc and Sensor Networks, A. Boukerche, ed., John Wiley & Sons, Inc., 2008, Chapter 2, pp. 21-50.
- [87]Boudjaadar Amina ; « Plateforme basée Agents pour l’aide à la conception et la simulation des réseaux de capteurs sans fil ». Thèse de Magistère ; Université de Skikda ; 2009/2010
- [88]Tseng et Al. “The Broadcast Storm Problem in a Mobile Ad Hoc Network”, in Wireless Networks, Kluwer Academic Publishers. 8, 153–167, 2002

- [89] Gérard CHALHOUB, “Les réseaux de capteurs sans fil ”
- [90] Y. Wang, G. Attebury et B. Ramamurthy, « A Survey of Security Issues in Wireless Sensor Networks » , 2005
- [91] K. Romer et F. Mattern, « The design space of wireless sensor networks», in IEEE Wireless Communications, 2006
- [92] S. Sentilles , « Architecture logicielle pour capteurs sans fils en réseau », rapport de recherche, Université de Pau et des pays de l’Adour, juin 2006
- [93] A. Milenkovic et al, « Wireless sensor networks for personal health monitoring: Issues and an implementation » Elsevier, 2006.
- [94] E.M. Petriu, N.D. Georganas, D.C. Petriu, D. Makrakis et V.Z. Groza , “ Sensor- based information appliances”, IEEE Instrumentation Measurement Magazine, vol.3, no.4, p.31-35, December 2000.
- [95] P. Corke ; T.Wark ; R.Jurdak ; Wen Hu ; P.Valencia ; D.Moore, « Environmental wireless sensor networks », Proceedings of the IEEE Volume: 98, Nov. 2010
- [96] Bluetooth SIG, “ Specification of the Bluetooth system, Core, Version 1.1”, Bluetooth SIG, www.Bluetooth.com .
- [97] ZigBee Standards Organization .”ZigBee Specification”, Document 053474r17, ZigBee Standards Organization, Janvier 2008
- [98] Institute of Electrical and Electronics Engineers Inc., "IEEE Std. 802.15.4-2003. Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low Rate Wireless Personal Area Networks (LR-WPANs)," IEEE Press, October 1, 2003.
- [99] LAN-MAN Standards Committee of the IEEE Computer Society, “802.15.4 IEEE Standard for Information technology”, Part 15.4 : Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs), IEEE Std 802.15.4-2003 (2003)
- [100] E. Callaway, P. Gorday, L. Hester, J.A. Gutierrez, M. Naeve, B.Heile et V. Bahl “Home Networking with IEEE 802.15.4 : a developing standard for low-rate wireless personal area Networks » ,IEEE Communications Magazine vol. 40, no. 8, p 70-77, Août. 2002
- [101] www.zigbee.org
- [102] The network simulator ns-2. <http://www.isi.edu/nsnam/ns/>, 2001
- [103] Lee Breslau, Deborah Estrin, Kevin Fall, Sally Floyd, John Heidemann, Ahmed Helmy, Polly Huang, Steven McCanne, Kannan Varadhan, Ya Xu, and Haobo Yu. Advances in network simulation. Computer, 33(5) :59–67, May 2000
- [104] Deborah Estrin, Mark Handley, John Heidemann, Steven McCanne, Ya Xu, and Haobo Yu. Network visualization with nam, the vint network animator. Computer, 33(11) :63–68, November 2000
- [105] Mobility framework for omnet++. <http://mobility-fw.sourceforge.net>.
- [106] Omnet++ discrete event simulator. <http://www.omnetpp.org>
- [107] George F. Riley. The georgia tech network simulator. In Proceedings of the ACM SIGCOMM workshop on Models, methods and tools for reproducible network research, MoMeTools ’03, pages 5–12, New York, NY, USA, 2003. ACM

- [108] Desbrandes, F., S. Bertolotti and L. Dunand, Opnet 2.4: An environment for communication network modeling and simulation, in: Proceedings of European Simulation Symposium. Society for Computer Simulation, 1993, pp. 64–74
- [109] Cavin, D., Y. Sasson and A. Schiper, On the accuracy of manet simulators, in: POMC '02: Proceedings of the second ACM international workshop on Principles of mobile computing (2002), pp. 38–43.
- [110] Ben L. Titzer, Daniel K. Lee, and Jens Palsberg. Avrora : scalable sensor network simulation with precise timing. In Proceedings of the 4th international symposium on Information processing in sensor networks, IPSN '05, Piscataway, NJ, USA, 2005. IEEE Press.
- [111] J. Polley, D. Blazakis, J. McGee, D. Rusk, and J. S. Baras. Atemu : a finegrained sensor network simulator. In IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks (SECON), pages 145–152. IEEE Press, October 2004.
- [112] Philip Levis, Nelson Lee, Matt Welsh, and David Culler. Tossim : accurate and scalable simulation of entire tinyos applications. In Proceedings of the 1st international conference on Embedded networked sensor systems, SenSys '03, pages 126–137, New York, NY, USA, 2003. ACM.
- [113] P. Levis, S. Madden, J. Polastre, R. Szewczyk, K. Whitehouse, A. Woo, D. Gay, J. Hill, M. Welsh, E. Brewer, and D. Culler. Tinyos : An operating system for sensor networks. In Werner Weber, Jan M. Rabaey, and Emile Aarts, editors, Ambient Intelligence, pages 115–148. Springer Berlin Heidelberg, 2005. Tinyos home page. <http://www.tinyos.net>.
- [114] Victor Shnayder, Mark Hempstead, Bor-rong Chen, Geoff Werner Allen, and Matt Welsh. Simulating the power consumption of large-scale sensor network applications. In Proceedings of the 2nd international conference on Embedded networked sensor systems, SenSys '04, pages 188–200, New York, NY, USA, 2004. ACM
- [115] Mica sensors specification. <http://www.xbow.com>
- [116] David Gay, Philip Levis, Robert von Behren, Matt Welsh, Eric Brewer, and David Culler. The nesc language : A holistic approach to networked embedded systems. SIGPLAN Not., 38(5) :1–11, May 2003
- [117] Antoine Fraboulet, Guillaume Chelius, and Eric Fleury. Worldsens : development and prototyping tools for application specific wireless sensors networks. In Proceedings of the 6th international conference on Information processing in sensor networks, IPSN '07, pages 176–185, New York, NY, USA, 2007. ACM.
- [118] E. Egea-Lopez, J. Vales-Alonso, A. S. Martinez-Sala, P. Pavon-Marino, and J. Garcia-Haro. Simulation tools for wireless sensor networks. In Summer Simulation Multiconference - SPECTS 2005, 2005.
- [119] Avinash Sridharan, Marco Zuniga, and Bhaskar Krishnamachari. Integrating environment simulators with network simulators, 2004.
- [120] The mathworks. matlab. <http://www.mathworks.com/products/matlab/>.
- [121] Sung Park, Andreas Savvides, and Mani B. Srivastava. Sensorsim : a simulation framework for sensor networks. In Proceedings of the 3rd ACM international workshop on Modeling, analysis and simulation of wireless and mobile systems, MSWIM '00, pages 104–111, New York, NY, USA, 2000. ACM
- [122] Ahmed Sobeih, Wei-Peng Chen, Jennifer C. Hou, Lu-Chuan Kung, Ning Li, Hyuk Lim, Hung-Ying Tyan, and Honghai Zhang. J-sim : A simulation environment for wireless sensor networks. In Proceedings of the 38th annual Symposium on Simulation, ANSS '05, pages 175–187, Washington, DC, USA, 2005. IEEE Computer Society.

- [123] Ian Downard and Washington Dc. Simulating sensor networks in ns-2. Proceedings of the IEEE, pages 1–9, 2004.
- [124] Luis Javier Garc'ia Villalba, Ana Lucila Sandoval Orozco, Alicia Triviño Cabrera, and Cl'audia Jacy Barenco Abbas. Routing protocols in wireless sensor networks. Sensors (Peterboroug, 9(11) :8399–8421, 2009
- [125] W.B. Heinzelman, A.P. Chandrakasan, and H. Balakrishnan. An applicationspecific protocol architecture for wireless microsensor networks. Wireless Communications, IEEE Transactions on, 1(4) :660–670, October 2002.
- [126] Wikipédia. Round-robin (informatique) — wikipédia, l'encyclopédie libre, 2012. [En ligne ; Page disponible le 11-mai-2012].
- [127] S. Lindsey and C. S. Raghavendra. PEGASIS : Power Efficient GATHERing in Sensor Information Systems. In Proceedings of IEEE Aerospace Conference, 2002.
- [128] Arati Manjeshwar and Dharma P. Agrawal. Teen : A routing protocol for enhanced efficiency in wireless sensor networks. In Proceedings of the 15th International Parallel & Distributed Processing Symposium, IPDPS '01, pages 189–, Washington, DC, USA, 2001. IEEE Computer Society.
- [129] Supriyo Chatterjea, Simone De Luigi, and Paul J. M. Havinga. Dirq : A directed query dissemination scheme for wireless sensor networks. In Abraham O. Fapojuwo and Bozena Kaminska, editors, Wireless and Optical Communications. IASTED/ACTA Press, 2006.
- [130] Y. Yu, R. Govindan, and D. Estrin. Geographical and energy aware routing :A recursive data dissemination protocol for wireless sensor networks. Technical report, University of California at Los Angeles, May 2001.
- [131] Ya Xu, John Heidemann, and Deborah Estrin. Geography-informed energy conservation for ad hoc routing. In Proceedings of the 7th annual international conference on Mobile computing and networking, MobiCom '01, pages 70–84, New York, NY, USA, 2001. ACM.
- [132] Joanna Kulik, Wendi Heinzelman, and Hari Balakrishnan. Negotiation-based protocols for disseminating information in wireless sensor networks. Wirel-Netw., 8(2/3) :169–185, March 2002.
- [133] Chalermek Intanagonwiwat, Ramesh Govindan, and Deborah Estrin. Directed diffusion : a scalable and robust communication paradigm for sensor networks. In Proceedings of the 6th annual international conference on Mobile computing and networking, MobiCom '00, pages 56–67, New York, NY, USA, 2000. ACM
- [134] David Braginsky and Deborah Estrin. Rumor routing algorithm for sensor networks. In Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications, WSNA '02, pages 22–31, New York, NY, USA, 2002. ACM.
- [135] Yong Yao and Johannes Gehrke. The cougar approach to in-network query processing in sensor networks. SIGMOD Rec., 31(3) :9–18, September 2002.
- [136] N. Sadagopan, B. Krishnamachari, and A.Helmy. The ACQUIRE mechanism for efficient querying in sensor networks. In Proc. 1st IEEE Intl. Workshop on Sensor Network Protocols and Applications (SNPA), Anchorage, AK, May 2003.
- [137] Jamal N. Al-karaki and Ahmed E. Kamal. Routing techniques in wireless sensor networks : A survey. IEEE Wireless Communications, 11 :6–28, 2004.

- [138] Tian He, John A. Stankovic, Chenyang Lu, and Tarek Abdelzaher. Speed : A stateless protocol for real-time communication in sensor networks. In Proc. 23rd International Conference on Distributed Computing Systems (ICDCS'03), Providence, Rhode Island, May 2003.
- [139] M. JERBI. "Protocoles pour les communications dans les réseaux de véhicules en environnement urbain : Routage et GeoCast basés sur les intersections". Thèse de doctorat, université d'Ivry, France, 2008
- [140] Giordano S., et al., "Position based routing algorithms for ad hoc networks: A taxonomy," in Ad Hoc Wireless Networking, X. Cheng, X. Huang and D.Z. Du (eds.), Kluwer, December 2003.
- [141] Nabil Drawil, "Improving the VANET Vehicles Localization Accuracy Using GPS Receiver in Multipath Environments", Thèse de doctorat, Waterloo, Ontario, Canada, 2007
- [142] J. Li, J. Jannotti, D. De Couto, D. Karger, and R. Morris, "A scalable location service for geographic ad hoc routing", In Proceedings of the ACM/IEEE International Conference on Mobile Computing and Networking (MOBICOM), p120.130, 2000.
- [143] Haas, Zygmunt J., and Ben Liang. "Ad hoc mobility management with uniform quorum systems". IEEE/ACM Transactions on Networking (TON), 1999, vol. 7, no 2, p. 228-240.
- [144] I. Stojmenovic, Home agent based location update and destination search schemes in ad hoc networks, Technical report, Department of Computer Science, University of Ottawa, TR-99-10, 1999.
- [145] Karp, B. and Kung, H.T., Greedy Perimeter Stateless Routing for Wireless Networks, in proceedings of the Sixth Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom 2000), Boston, MA, August, 2000.
- [146] Young Jin Kim and Ramesh Govidan ,Greedy Perimeter Stateless Routing (GPSR) Detailed-level Design Specification (DDS) Version 0.1 , Department of Computer Science University of Southern California, July 6, 2003.
- [147] Jurcik, P.; Koubaa, A.; Alves, M.; Tovar, E.; Hanzalek, Z.; "A Simulation Model for the IEEE 802.15.4 protocol: Delay/Throughput Evaluation of the GTS Mechanism", Modeling, Analysis, and Simulation of Computer and Telecommunication Systems, 2007. MASCOTS07. 15th International Symposium, 24-26 Oct. 2007.
- [148] Paul Daniel Mitchell; JianQiu; Hengguang Li; David Grace; " Use of aerial platforms for energy efficient medium access control in wireless sensor networks", Computer Communications, 2010 (Elsevier).
- [149] S.M. Mahdi Alavi; Michael J. Walsh; Martin J. Hayes; "Robust distributed active power control technique for IEEE 802.15.4 wireless sensor networks – A quantitative feedback theory approach", Control Engineering Practice, 2009 .
- [150] David Coudert; NapoleaoNepomuceno; HerveRivano; "Power-efficient radio configuration in fixed broadband wireless networks", Computer Communications, 2010.
- [151] Patrick Hosein, "Adaptive Algorithm for Mapping Channel Quality Information to Modulation and Coding Schemes"; IEEE, 2009.
- [152] Mohamed Nazim Abdeddaim. Analyse des performances d'un réseau de capteurs exploitant le standard IEEE 802.15.4, thèse doctorat octobre 2012.

- [153] MayyadaHammoshi, AwnySayed, An analysis for a cluster tree zigbee topology, Journal of Theoretical and Applied Information Technology, 30th June 2014. Vol. 64 No.3.
- [154] Prasad P. Netalkar¹, Yasha Kaushal², Dr. N. Shekar V. Shet³, Zigbee Based Wireless Sensor Networks for Smart Campus, International Journal Of Modern Engineering Research (IJMER), ISSN: 2249–6645, Vol. 4 Iss.7 July. 2014.
- [155] Devender, Chirag, Kuldeep Vats, DeepikaMadaan, WSN Based Performance Evaluation of Coordinator Fails in Different PAN Network using OPNET, International Journal of Computer Science and Mobile Computing, Vol. 3, Issue. 11, November 2014, pg.504 – 513.
- [156] M. M. Chandane, S. G. Bhirud, S.V. Bonde, Performance Analysis of IEEE 802.15.4, International Journal of Computer Applications (0975 – 8887) Volume 40– No.5, February 2012.
- [157] Singh K. (2014) Optimisation of IEEE 802.15.4 for suburban smart grid communication in hybrid LTE-WSN network. University of Oulu, Department of Communications Engineering, Master’s Degree program in Wireless Communications Engineering. Master’s thesis, 57 p
- [158] Mr.Clément SAAD, « Quelques contributions dans les réseaux de capteurs sans fil: Localisation et Routage »,thèse pour obtenir un diplôme de DOCTORAT, l’Université d’Avignon et des Pays de Vaucluse ,2009.
- [159] N. Bulusu. « Self-Con_guring Location Systems». PhD thesis, University of California, 2002
- [160] N. Bulusu, J. Heidemann, and D. Estrin. « GPS-less low-cost outdoor localization for very small devices». Personal Communications, IEEE, 7(5) :28_34, 2000
- [161] J.A. Costa, N. Patwari, and A.O. Hero III. «Distributed weighted multidimensional scaling for node localization in sensor networks». ACM Transactions on Sensor Networks (TOSN), 2(1) :39_64, 2006.
- [162] J. Pugh. «Local Range and Bearing Sensing Using Infrared Transceivers in Mobile Robotics». 2004.
- [163] Seattle Robotics Society. « Implementing infrared object detection»
- [164] N.B. Priyantha, A. Chakraborty, and H. Balakrishnan. «The Cricket locationsupport system. Proceedings of the 6th annual international conference on Mobile computing and networking, pages 32_43, 2000.
- [165] K. Whitehouse, C. Karlof, A. Woo, F. Jiang, and D. Culler. «The effects of ranging noise on multihop localisation : an empirical study». Proceedings of the fifth international conference on Information processing in sensor networks, 2005.
- [166] J.P. Curty, M. Declercq, C. Dehollain, and N. Joehl. « Design and Optimization of Passive UHF RFID Systems». Springer, 2007.
- [167] M. Kushwaha, K. Molnar, J. Sallai, P. Volgyesi, M. Maroti, and A. Ledeczi. , «Sensor Node Localization Using Mobile Acoustic Beacons». PhD thesis, Vanderbilt University, 2005.
- [168] Linqing GUI ,Improvement of Range-free Localization Systems in Wireless Sensor Networks these doctorat fevrier 2013
- [169] D. Niculescu and B. Nath, DV Based Positioning in Ad hoc Networks, In Journal of Telecommunication Systems, 2003.
- [170] SAHRAOUI Belkheyr, La Géo-localisation dans les Réseaux de Capteurs sans Fil, Mémoire de fin d’études pour l’obtention du diplôme Ingénieur s’Etat en Informatique 2011.

- [171] Y. Liu, "An adaptive multi-hop distance localization algorithm in WSN," *Manufacturing Automation*, vol. 33, pp. 161–163, 2011
- [172] R. Nagpal, H. Shrobe, J. Bachrach, Organizing a Global Coordinate System from Local Information on an Ad Hoc Sensor Network, In the 2nd International Workshop on Information Processing in Sensor Networks (IPSN '03), Palo Alto, April, 2003.
- [173] Y. Xu, J. Heidemann, D. Estrin, "Geography-informed energy conservation for Ad Hoc routing". In Proceedings of the 7th Annual international Conference on Mobile Computing and Networking (MobiCom '01), Rome, Italy, Jul. 16-21, 2001, pp. 70-84 166 173
- [174] S. Basagni, I. Chlamtac, V. R. Syrotiuk, and B. A. Woodward, "A distance routing effect algorithm for mobility (dream)," in Proceedings of the 4th annual ACM/IEEE international conference on Mobile computing and networking (MobiCom'98), New York, NY, USA, 1998, pp.76–84.
- [175] Y.-B. Ko and N. H. Vaidya, "Location-aided routing (lar) in mobile ad hoc networks," *Wirel. Netw.*, vol. 6, no. 4, pp. 307–321, Jul. 2000.
- [176] M. Ayaida, M. Barhoumi, Y. Ghamri-Doudane, L. Afilal. "Joint routing and location-based service in VANETs". *Journal of Parallel and Distributed Computing*, Volume 74, Issue 2, pp. 2077--2087, February ,2014.
- [177] R. Khadim, M. Erritali, A. Maaden, "Hierarchical Location-Based Services for Wireless Sensor Networks", in IEEE 13th International Conference on Computer Graphics, Imaging and Visualization, Beni-Mellal, Morocco, March 2016
- [178] B. Karp and H. T. Kung, "Gpsr: greedy perimeter stateless routing for wireless networks," in Proceedings of the 6th annual international conference on Mobile computing and networking (MobiCom'00), New York, NY, USA, 2000, pp. 243–254.
- [179] M. Kasemann, H. Hartenstein, and M. Mauve, "A reactive location service for mobile ad hoc networks," Department of Computer Science University of Mannheim Tech Rep TR02014, pp. 121–133, 2002.
- [180] J. Li, J. Jannotti, D. S. J. De Couto, D. R. Karger, and R. Morris, "A scalable location service for geographic ad hoc routing," in Proceedings of the 6th annual international conference on Mobile computing and networking (MobiCom'00), New York, NY, USA, 2000, pp. 120–130.
- [181] W. Kiess, H. Fussler, J. Widmer, and M. Mauve, "Hierarchical location service for mobile ad-hoc networks," *SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 8, pp. 47–58, October 2004.
- [182][Online]. Available, <http://nslam.isi.edu/nslam/>
- [183] Akyildiz.I. F., Weilian S, Sankarasubramaniam:A Survey on Sensor Networks.In: *IEEE Communications Magazine*, 40(8):102.114, 2002.
- [184] Mainwaring.A, Polastre.J, Szewczyk.J, Culler.D,J:Wireless Sensor Networks for Habitat Monitoring.*In: WSNA'02*, September, 2002.
- [185] Tubaishat.M, Madria.S: Sensor Networks: An Overview. *IEEE Potentials*, pp. 20-23, May, 2003.
- [186] Yu.L,Wang.N,Meng.X: Real-time forest re detection with wireless sensor networks. *Int. Conf. on Wireless Comm, N/w & Mobile Computing*.

- [187] Youssef.M, Yousif.A, El-Sheimy.N, Noureldin.A: A Novel Earthquake Warning System Based on Virtual MIMO-Wireless Sensor Networks. *Canadian Conference on Electrical and Computer Engineering*, April ,2007.
- [188] Loscri.V,Marano.S,Morabito.G:A:Two-Levels Hierarchy for Low-Energy Adaptative Clustering Hierarchy (TL-LEACH).In: *Proceedings ofVTC2005*, pp. 1809- 1813, Dallas USA, September, 2005.
- [189] Chaurasiya, Kumar.S,Pal.T:An Enhanced Energy-Efficient Protocol with Static Clustering for WSN. In: *International Conference on Information Networking (ICOIN)*, Kuala Lumpur, Malaysia, pp. 58 – 63, 2011.1014 *Rajendra Prasad Mahapatra and Rakesh Kumar Yadav / Procedia Computer Science 57 (2015) 1005 – 1014*
- [190] Eleburuike.I.O, Dekunle.S.S: Energy Efficient Wireless Sensor Network Using Hierarchical Routing Technique: *Blekinge Institute of Technology, Karlskrona, Blekinge, Sweden*, 2010.
- [191] Gongben.C., Shaorong.W: A Novel Node Deployment and Clustering Scheme in Wireless Sensor Networks. In: *Second International Symposium on Electronic Commerce and Security, ISECS '09*, Nanchang, China, pp.499 – 503, 2009.
- [192] Yang.J and Zhang.D:An Energy-Balancing Unequal Clustering Protocol for WSNs. In:*Information Technology Journal*, 57-63, 2009.
- [193] Heinzelman. W.R.,Chandrakasan: A, “Energy efficient Communication Protocol for Wireless Microsensor Networks. In: IEEE Computer Society Proceedings of the 33rd Hawaii International Conference on System Sciences (HICSS '00), Vol. 8, pp. 8020, Januarury, 2000
- [194] Frey, Hannes, Stefan Rührup, and Ivan Stojmenović. "Routing in wireless sensor networks." *Guide to Wireless Sensor Networks*. Springer London, 2009. 81-111.
- [195] Heinzelman, W.R., A.Chandrakasan,H.Balakrishnan.”An application Specific Protocol Architecture for Wireless Mincrosensor Networks.-*IEEE Transactions on Wireless Communication*, Vol.1, 2002,No 4,pp.660-670.
- [196] T.Murata and H. Ishibuchi, “Performance evaluation of genetic algorithms for flowshop scheduling problems” ,In *Proceedings of the 1st IEEE Conference Evolutionary Computation*, volume 2,pp. 812-817.(1994).
- [197] E.Abdellah, S. Benalla, A.B.Hsaane, and M. Lahcen, “Advanced Low Energy Adaptive Clustering Hierarchy,” (IJCSE) *International Journal on Computer Science and Engineering*, vol. 02, no. 07, pp. 2491-2497, 2010
- [198] Kumar, Vinay, Sanjeev Jain, and Sudarshan Tiwari. "Energy efficient clustering algorithms in wireless sensor networks: A survey." *IJCSInternational Journal of Computer Science Issues* 8.5 (2011)
- [199] Parmar, Bhakti, et al. "A Survey of routing protocol LEACH for WSN." *International Journal of Scientific and Research Publications* 4.1 (2014).
- [200] Fan. X. N, Song. Y. L: Improvement on LEACH protocol of wireless sensor network. In: *Proc. International Conference on Sensor Technologies and Applications*, Sensor Comm., pp. 260-264, 2007
- [201] Manimala.P, Senthamil.R: A Survey on Leach-Energy Based Routing Protocol. *International Journal of Emerging Technology and Advanced Engineering (IJETAE)*, Vol.3, Issue 12, pp. 657-660, December,2013

- [202] Dembla.D Shivam .H: Analysis and Implementation of Improved - LEACH protocol for Wireless Sensor Network (I-LEACH).In: *IJCSC,IJ*, Vol. 4, No. 2, pp.8-12, September, 2013
- [203] Bakaraniya.P, Sheetal. M: K-LEACH: An improved LEACH Protocol for Lifetime Improvement in WSN. In: *International Journal of Engineering Trends and Technology (IJETT)*, Vol-4, Issue 5, pp1521 - 1526, May, 2013
- [204] Qian .L, Zhu.H: An Energy Balanced Clustering Algorithm Based on LEACH Protocol. In: *Proceedings of the 2nd International Conference On Systems Engineering and Modeling (ICSEM-13)*, Paris, France, pp-72-73, 2013
- [205] Kim. D. S, Chung Y. J: Self-organization routing protocol supporting mobile nodes for wireless sensor network. In: *Proc. First International Multi-Symposiums on Computer and Computational Sciences*, Hangzhou, China, 2006.
- [206] Kumar G. S., Paul. M. V and Jacob. K. P.: Mobility metric based leach-mobile protocol. In: *Proc. 16th International Conference on Advanced Computing and Communications, ADCOM*, 2008
- [207] Zhu.D,Cai.D:Research and simulation of energy efficient protocol for wireless sensor network. In: *Proc. 2nd International Conference on Computer Engineering and Technology*, Quanzhou, China, 2010.
- [208] Gongben.C., Shaorong.W: A Novel Node Deployment and Clustering Scheme in Wireless Sensor Networks. In: *Second International Symposium on Electronic Commerce and Security, ISECS '09*, Nanchang, China, pp.499 – 503, 2009.
- [209] Thiemo .V, Hartmut R, Jochen. S, Adam .D, Juan .A: Solar-aware clustering in Wireless Sensor Networks. In: *Proceedings of the Ninth IEEE Symposium on Computers and Communications*, June, 2004.
- [210] Jiman .H and Joongjin .K: T-LEACH: The method of threshold-based cluster head replacement for wireless sensor networks,” published In *springer Inf Syst Front*, DOI 10.1007/s10796-008-9121 -4
- [211] Bani Y.M, Al-zou'bi.A, Khamayseh.Y and Mardini.W: Improvement on LEACH protocol of wireless sensor network (VLEACH). *International Journal of Digital Content Technology and its Applications*, Vol. 3, No. 2, pp. 132-136, 2009.
- [212] Hanady. M., Bader.A.:W-LEACH Based Dynamic Adaptive Data Aggregation Algorithm for Wireless Sensor Networks. *Hindawi Publishing Corporation International Journal of Distributed Sensor Networks*, Vol. 2013, Article ID 289527, pp.1 -11, 2013
- [213] Abdelhalim.H, Abdellah.E, Mohammed .F, Abdelmajid. H. W-LEACH: Decentralized: Weighted LEACH with a Decentralized way. *Published in: IOSR Journal of Computer Engineering (IOSR-JCE)*, Vol.16, Issue 2, Ver -VII, pp. 32-3, April, 2014
- [214] Bakr, Bilal Abu, and LeszekLilien. "A quantitative comparison of energy consumption and WSN lifetime for LEACH and LEACH-SM." *Distributed Computing Systems Workshops (ICDCSW)*, 2011 31st International Conference on. IEEE, 2011.
- [215] Yuvaraj, P., K. Vikram, and K. Venkata Lakshmi Narayana. "A Review on State of Art Variants of LEACH Protocol for Wireless Sensor Networks." *Sensors & Transducers (1726-5479)* 186.3 (2015).
- [216] Tumer, Abdullah Erdal, and Mesut Gunduz. "An improved leach protocol for indoor wireless sensor networks." *Signal Processing and Integrated Networks (SPIN)*, 2014 International Conference on. IEEE, 2014.

- [217] Xiangning, Fan, and Song Yulin. "Improvement on LEACH protocol of wireless sensor network." *Sensor Technologies and Applications, 2007. SensorComm 2007. International Conference on.* IEEE, 2007.
- [218] Fu, Chunyao, et al. "An Energy Balanced Algorithm of LEACH Protocol in WSN." *IJCSI International Journal of Computer Science Issues* 10.1 (2013): 354-359 LEACH-TLC
- [219] D. Q. Lu Cheng and W. Wu, "An energy efficient weight-clustering algorithm in wireless sensor networks," in *Proc. Japan-China Joint Workshop on Frontier of Computer Science and Technology*, IEEE Computer Society, 2008.
- [220] E. Cayirci, I.W. Su, Y. Sankarasubramaniam, "*A Survey on Sensor Networks*, *IEEE Communications Magazine*, Vol. 40, No. 8, pp. 102-114, (2002).220

Liste des publications

- 1. Rania KHADIM, Mohammed ERRITALI, Abdelhakim MAADEN, “Performance Study of IEEE 802.15.4 under OPNET Modeler for Wireless Sensor Networks”.**

TELKOMNIKA Indonesian Journal of Electrical Engineering .Vol 16, No 1: October 2015, pp: 98-105.

- 2. Rania KHADIM, Mohammed ERRITALI, Abdelhakim MAADEN, “Range-Free Localization Schemes for Wireless Sensor Networks”.**

TELKOMNIKA Indonesian Journal of Electrical Engineering .Vol 16, No.2, November 2015, pp: 323-332

- 3. Rania KHADIM, Mohammed ERRITALI, Abdelhakim MAADEN, “Performance Evaluation of Hierarchical Location-Based Services Coupled to the Greedy Perimeter Stateless Routing Protocol for Wireless Sensor Networks”.**

International Journal of Electronics and Communication Engineering Vol:10, No:8, 2016

- 4. Rania KHADIM, Mohammed ERRITALI, Abdelhakim MAADEN, “Hierarchical Location-Based Services for Wireless Sensor Networks”.**

In Proceedings of the IEEE 13th International Conference on Computer Graphics, Imaging and Visualization (CGIV), pp. 457—463.Beni-Mellal, Morocco, March 201

- 5. Rania KHADIM, Ansam ENNACIRI, Mohammed ERRITALI, Abdelhakim MAADEN, “Impact of Location Data Freshness on Routing in Wireless Sensor Networks”.**

Chapter: Europe and MENA Cooperation Advances in Information and Communication Technologies Volume 520 of the series Advances in Intelligent Systems and Computing pp 373-382.

En cours de publication :

6. Rania KHADIM, Mohammed ERRITALI, Abdelhakim MAADEN, "An Enhancement Approach for Extending the Network Lifetime in Wireless Sensor Network".

7. Rania KHADIM, Ansam ENNACIRI, Mohammed ERRITALI, Abdelhakim MAADEN. "An Energy-Efficient Clustering Algorithm for WSN Based on Cluster Head Selection Optimization to Prolong Network Lifetime".

Liste des communications

1. Rania KHADIM, Rachid El-AYACHI, Mohammed FAKIR Recognition of 3D Objects from 2D Views Features. *The Second International Conference on Business Intelligence (CBI'15) à FST, Béni Mellal, Maroc, 23-25 Avril 2015.*
2. Rania KHADIM, Mohammed ERRITALI, Abdelhakim MAADEN..Performance Study of IEEE 802.15.4 Under OPNET Modeler for Wireless Sensor. *The Second International Conference on Business Intelligence (CBI'15) à FST, Béni Mellal, Maroc, 23-25 Avril 2015.*
3. Rania KHADIM, Mohammed ERRITALI, Abdelhakim MAADEN. Rang-Free Localization Schemes for Wireless Sensor Networks. *International Conference of Advanced Information Technology Services and Systems (AIT2S-15).*
4. Rania KHADIM, Mohammed ERRITALI, Abdelhakim MAADEN. Hierarchical Location-Based Services for Wireless Sensor Networks. *13th International Conference on Computer Graphics, Imaging and Visualization (CGIV 2016).*
5. Rania KHADIM, Ansam ENNACIRI, Mohammed ERRITALI, Abdelhakim MAADEN. Impact of Location Data Freshness on Routing in Wireless Sensor Networks. *Europe Middle East and North Africa Conference on Technology and Security to Support Learning (EMENA-TSSL 2016).*

6. Rania KHADIM, Ansam ENNACIRI, Mohammed ERRITALI, Abdelhakim MAADEN. Coupling of geographic location-based service and routing for wireless sensor networks. *17th International Arab Conference on Information Technology (ACIT' 2016), Sultan Moulay Slimane University, Beni Mellal, Morocco.*