

République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

FACULTE DE TECHNOLOGIE
DEPARTEMENT DE GENIE ELECTRIQUE ET ELECTRONIQUE
(GEE)



Mémoire

Pour l'obtention du diplôme de
MASTER en Télécommunication

Option : Réseaux et Systèmes de Télécommunications(RST)

Présenté par

MERABET Siham

ZIANI Djamila

THEME

*Etude et simulation des techniques de
minimisation de la consommation
d'énergie dans les RCSFs*

Soutenu en juin 2013 devant un jury composé de

Mr S.KAMECHE

Mr A.DJEMAI

Mr D.MOUSSAOUI

Mr R.MERZOUGUI

Mme S.MAMCHAOU

M.C.A à l'Université de Tlemcen

M.A.A à l'université de Tlemcen

M.A.A à l'université de Tlemcen

M.C.A à l'université de Tlemcen

Doctorante à l'université de Tlemcen

Président

Examinateur

Examinateur

Encadreur

Co-encadreur

Remerciements

Tout d'abord, nous remercions Dieu le tout Puissant qui nous a donné la force et la volonté pour réaliser ce modeste travail.

Nous tenons à exprimer notre grande gratitude à notre encadreur Mr R.MERZOUGUI, pour avoir accepté de nous encadrer tout au long de ce travail, pour sa disponibilité, son amabilité, ses conseils et suggestions et pour toute l'aide morale qu'il n'a cessé de nous prodiguer.

Nous tenons à remercier vivement la doctorante Mme S. MAMCHAOUI pour son implication totale et son aide importante pendant la rédaction de cette thèse.

Nous tenons également à remercier Mr.S. KAMECHE.S pour l'honneur qu'il nous fait de présider notre jury de soutenance, nous lui exprimons notre gratitude profonde.

Nos remerciements s'adressent ensuite à Mr D.MOUSSAOUI et à Mr A.DJEMAI qui ont aimablement accepté d'examiner et de juger notre modeste travail.

Nous remercions profondément les doctorants: Mr H.HACHEMI, et Mlle B.BOUKNADIL pour leur collaboration et leurs conseils.

Enfin, nous remercions toutes les personnes, qui de près ou de loin ont contribué à l'élaboration de ce mémoire.

Merci à tous et à toutes.

DÉDICACES

C'est grâce à ALLAH le tout puissant que j'ai pu achever ce travail, que je dédie à:

A mes très chers parents, 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. Qu'ALLAH vous accorder santé, bonheur et longue vie et faire en sorte que jamais je ne vous déçoive.

A mes très chers frères : MOHAMED, HICHAM et AKRAM, *je vous souhaite une vie pleine de bonheur et de succès et que Dieu, le tout puissant, vous protège et vous garde.*

Particulièrement, à ma tante ZAHRA qui m'a toujours supporté moralement par ses encouragements pendant toutes mes longues années d'étude.

A mon adorable binôme SIHAM ainsi que sa noble famille, En souvenir de notre sincère et profonde amitié et des moments agréables que nous avons passés ensemble. Veuillez trouver dans ce travail l'expression de mon respect le plus profond et mon affection la plus sincère.

A tous mes amis qui me sont très chers et qui m'ont aidé du près de près ou de loin
A tous ceux qui j'ai omis de citer et qui ne sont pas moins chers.

Djamila

DÉDICACES

Avec l'aide du Dieu le tout puissant, j'ai pu achever ce modeste travail que je dédie.

A ma très chère mère pour toute sa tendresse, son appui, son amour, sa compréhension, son aide, et pour ses nombreux sacrifices. Que Dieu la garde et la préserve une bonne santé et longue vie.

A mon très cher père pour son amour, sa patience, son soutien et son encouragement afin de m'assurer cette formation dans les meilleures conditions, et pour tout ce qu'il a fait pour moi. Que Dieu le garde et le préserve une bonne santé et longue vie.

A mon très cher frère: FETHEDDINE et sa femme NADIA ;

A mon adorable sœur: HADJER et son mari BENAMMAR ;

A la petite de ma famille: IMENE

Particulièrement à ma très proche cousine HANAN

Surtout à mon très cher binôme et amie DJAMILA et sa famille ; pour la merveilleuse ambiance qui caractérise notre amitié. Qu'elle soit heureuse sur les plans personnels, professionnels et sociaux.

A tous ce que je n'ai pas cités et qui ne me sont pas moins chers ;

A tous ceux que j'aime et tous ceux qu'ils m'aiment.

Siham

Table des matières

Remerciement.....	2
Résumé.....	5
Table des matières	6
Table des figures.....	10
Liste des tableaux.....	11
Glossaire	12

Introduction générale	13
-----------------------------	----

Premier Chapitre:
Concepts de base sur les RCSFs

I.1 – Introduction	15
I.2 – Présentation des réseaux de capteurs sans fil.....	15
I.2.1 – Architecture d’un capteur sans fil	16
– Unité de captage (Sensing unit)	16
– Unité de traitement (Processing unit).....	16
– Unité de communication (Tranceiver unit)	17
– Unité d’énergie (Power unit)	17
– Unité de stockage (Mémoire)	17
I.2.2 – Caractéristiques principales d’un capteur.....	17
I.2.3 – Différents nœuds capteurs.....	17
I.2.3.1 – MICA2	18
I.2.3.2 – TelosB	19
I.2.3.3 – Imote2	19
I.2.3.4 –TinyNode.....	20
I.2.4 – Les états d’un nœud capteur.....	20
I.3 – Réseaux de capteurs	21
I.3.1 – Définition d’un réseau de capteurs sans fil	21
I.3.2 – Architecture de communication d’un RCSF.....	21
I.3.3 – Architecture protocolaire.....	22
I.3.4 – Types d’architecture des RCSFs	23
I.3.4.1 – Architecture plate ou horizontale.....	23
I.3.4.2 – Architecture hiérarchique.....	24
I.4 – Contraintes de conception des RCSFs.....	25
I.4.1 – La tolérance aux fautes.....	25
I.4.2 – Le facteur d’échelle (Scalabilité)	25
I.4.3 – Coût de production	25

I.4.4 – Environnement	26
I.4.5 – Médias de transmission	26
I.4.6 – Système d’exploitation	26
I.4.7 – Sécurité physique limitée.....	26
I.4.8 – Topologie de réseau	26
I.4.9 – Connectivité.....	26
I.4.10 – Consommation d’énergie.....	27
I.4.11 – Agrégation de données.....	27
I.5 – Réseau de capteurs vs réseaux ad hoc.....	27
I.6 – Technologies des réseaux de capteurs	29
I.6.1 – Bluetooth.....	29
I.6.2 – Zegbee.....	29
I.6.3 – UWB.....	29
I.6.4 – Infrarouge.....	29
I.7 – Applications des RCSFs.....	30
I.7.1 – Applications militaires	30
I.7.2 – Applications médicales	30
I.7.3 – Applications domestiques	31
I.7.4 – Applications environnementales	31
I.7.5 – Applications industrielles	31
I.7.6 – Applications écologiques.....	32
I.7.7 – Agriculture de précision	32
I.8 – Conclusion	33



Second Chapitre:
Consommation d’énergie dans les RCSFs

II.1 – Introduction	35
II.2 – Problème de l’énergie dans les RCSFs.....	35
– L’énergie de capture	35
– L’énergie de communication.....	35
– L’énergie de traitement.....	36
II.3 – Facteurs intervenants dans la consommation d’énergie	36
II.3.1 – Technologie de communication.....	36
II.3.2 – Etat du module radio	36
II.3.3 – Accès au médium de transmission.....	37
II.3.3.A – La retransmission	37
II.3.3.B – L’écoute active.....	37
II.3.3.C – La surécoute	37
II.3.3.D – La surcharge	37
II.3.3.E – La surémission	37
II.3.3.F – La taille des paquets	37
II.3.4 – Modèle de propagation radio	38
II.4 – Techniques de minimisation de la consommation d’énergie	38
II.4.1 – Routage	38
II.4.1.1 – Classification des protocoles de routage pour les RCSFs	39
A – Selon la topologie du réseau	40

1) – Topologie plate	40
2) – Topologie hiérarchique.....	40
B – Selon le mode d'établissement des routes	41
1) – Protocoles de routage proactifs	41
2) – Protocoles de routage réactifs.....	41
3) – Protocoles de routage hybrides	41
II.4.2 – Contrôle de la topologie.....	41
II.4.2.1 – Construction de topologie.....	42
A – Contrôle de la puissance de transmission.....	42
B – Hiérarchisation.....	42
II.4.2.2 – Maintenance de la topologie.....	42
A – Techniques statiques globales de maintenance de la topologie.....	43
B – Techniques dynamiques de maintenance de la topologie	43
C – Techniques hybrides de maintenance de la topologie	44
II.5 – Conclusion.....	44

Troisième Chapitre:

Protocoles de routage et de contrôle de la topologie

III.1 – Introduction	46
III.2 – Les protocoles de routage dans RSCF minimisant la consommation des batteries.....	46
III.2.1 – Protocole DSDV	46
III.2.2 – Protocole DSR.....	47
III.2.3 – Protocole AODV	47
III.2.4 – Protocole TORA	48
III.2.5 – Protocole LEACH.....	48
III.2.6 – Protocole AOMDV.....	49
III.2.7 – Comparaison entre les différents protocoles	49
III.3 – Protocoles de contrôle de topologie dans RSCF minimisant la consommation d'énergie.....	50
III.3.1 – Construction de la topologie.....	50
III.3.1.a – A3: connectivité.....	50
III.3.1.b – A3Lite: connectivité	51
III.3.1.c – A3Cov: couverture	52
III.3.1.d – A3CovLite: couverture	54
III.3.1.e – EECDs	54
III.3.1.f – CDS-règle-k	55
III.3.2 – Maintenance de la topologie.....	55
III.3.2.a – SGTTRot	55
III.3.2.b – SGETRot.....	56
III.3.2.c – DGTTRec	56
III.3.2.d – DGETRec	56
III.3.2.e – HGTTRec-Rot.....	56
III.3.2.f – HGETRec-Rot.....	56
III.3.3 – Comparaison entre les différents protocoles.....	56
III.3.3.a – Construction de topologie.....	56
III.3.3.b – Maintenance de topologie	57
III.4 – Conclusion.....	57

Quatrième Chapitre:

**Simulation des techniques:
Routage et contrôle de topologie**

IV.1 – But	59
IV.2 – Simulation NS2 : Partie Routage	59
IV.2.1 – Présentation du simulateur	59
IV.2.2 – Résultats de simulations.....	60
a) – 1 ^{er} scénario : 3 nœuds.....	60
b) – 2 ^{eme} scénario : 10 nœuds.....	63
IV.3 – Simulation par Atarraya : Partie Contrôle de la topologie	67
IV.3.1 – Présentation du simulateur	67
IV.3.2 – Résultats de simulations.....	67
IV.3.2.A – Simulation I	67
IV.3.2.B – Simulation II	71
a) – 1 ^{er} cas: $R_{comm}=R_{dét}$	72
b) – 2 ^{ème} cas: $R_{comm} > R_{dét}$ et $\alpha=1$	74
IV.3.2.C – Simulation III	77
IV.4 – Conclusion	80
<hr/>	
Conclusion générale	81
Bibliographie.....	82

Introduction générale

Actuellement, les réseaux de capteurs sans fil suscitent un intérêt croissant pour les technologies de l'information et de communication (TIC). Ces réseaux ont des atouts manifestes: ils peuvent être déployés très rapidement, couvrir de très grands espaces géographiques et fonctionner sans intervention humaine avec une grande tolérance aux défaillances. En raison de leur flexibilité, de leur faible coût et de la facilité de déploiement, les RCSFs promettent de révolutionner notre vie à travers plusieurs domaines d'applications tels que la détection et la surveillance des désastres, le contrôle de l'environnement et la cartographie de la biodiversité, le bâtiment intelligent, l'agriculture de précision, la surveillance et la maintenance préventive des machines, la médecine et la santé, la logistique et les transports intelligents.

En effet, ce type des réseaux présente des contraintes dont la plus importante est leur faible capacité d'énergie; puisque les nœuds capteurs sont dotés des batteries qui ne sont pas généralement rechargeables. Pour prolonger la durée de vie d'un réseau de capteur sans fil tout en assurant les trois tâches principales d'un nœud capteur: capture, traitement et communication, il faut bien conserver l'énergie des nœuds capteurs. Parmi ces trois tâches, la communication est la tâche qui consomme la plus grande partie de l'énergie. Ceci a motivé notre travail à se focaliser sur deux techniques de conservation d'énergie dans ce niveau.

L'objectif principal de notre travail est de s'initier au domaine des réseaux de capteurs sans fil: étudier et simuler le problème d'énergie, parcourir les causes de pertes d'énergie au niveau de la communication, et enfin proposer deux techniques qui minimisent la consommation d'énergie: le routage et le contrôle de topologie.

Le présent projet est scindé en 4 chapitres:

Dans le premier chapitre, nous allons présenter les réseaux de capteurs sans fil y compris leurs architectures de communication, leurs principaux facteurs et contraintes qui influencent la conception de ce type de réseaux ainsi que les différentes applications qui peuvent être réalisés grâce à ces réseaux.

Dans le deuxième chapitre, nous allons exposer le problème d'énergie au niveau communication. D'abord, nous allons présenter les facteurs intervenants dans la consommation d'énergie, ensuite nous allons citer d'une manière générale les techniques de conservation de cette énergie. Par la suite, nous allons détailler les deux techniques pour minimiser l'énergie de communication: Routage et Contrôle de la topologie exploités pour la génération des résultats de simulation présentés au dernier chapitre.

Le troisième chapitre sera divisé en deux parties, la première partie sera consacrée aux protocoles de routage, tandis que dans la deuxième partie nous allons citer les protocoles de contrôle de la topologie. A la fin de chaque partie, nous allons faire une comparaison entre les différents algorithmes de la technique étudiée.

Dans le dernier chapitre, nous allons présenter les résultats obtenus par les deux simulateurs NS2 et Atarraya pour la conservation d'énergie dans les réseaux de capteurs sans fil, en exploitant pour cela les deux techniques: routage et contrôle de la topologie.

I.1. Introduction

La forte demande dans le domaine des réseaux de communication sans fil, permet actuellement d'envisager l'évolution des réseaux de c

apteurs pour un grand nombre d'applications. Actuellement, le thème de réseau de capteurs sans fil (RCSF) provoque un intérêt croissant en termes de leur facilité, leur souplesse de déploiement, et leur faible coût. Ce qui permet à ces types de réseaux d'être plus exploités dans de nombreux domaines tels que le domaine militaire, médical, et la domotique...

Grâce à ces technologies, il est maintenant possible de capter, recueillir et traiter des données en provenance de plusieurs unités portatives à distance.

Dans ce chapitre, nous allons présenter l'infrastructure des réseaux de capteurs sans fil en précisant leurs caractéristiques, leurs fonctionnalités de base ainsi que leurs domaines d'applications dans la vie quotidienne.

I.2. Présentation des réseaux de capteurs sans fil

Par définition, un capteur est un dispositif réagissant à un stimulus, comme la chaleur, l'humidité ou la lumière. Son principe est de transformer l'état d'une grandeur physique observée (la pression, etc.) en une grandeur utilisable [1] (intensité électrique, position d'un flotteur) en générant un signal qui peut être mesuré ou interprété. Pour cela, il possède au moins un transducteur dont le rôle est de convertir une grandeur physique en une autre.

On distingue deux types de capteurs, les capteurs « traditionnels » (figure I.1 (a)) et « les micro-capteurs » (figure I.1 (b)) [2]. La différence entre ces deux types réside principalement au niveau de leurs fonctionnements (niveau d'intelligence) et de leurs dimensions. La figure I.1 illustre les photos de ces unités technologiques sans fil:



Figure I.1(a): Les capteurs traditionnels

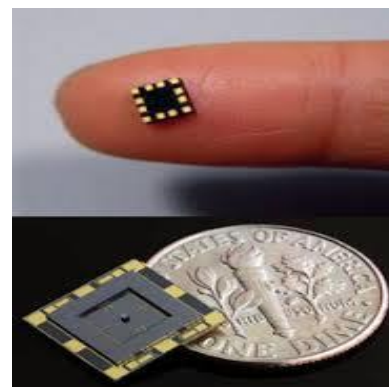


Figure I.1(b): Les micro-capteurs

Figure I.1: Les types de capteurs.

Ces capteurs, nommés aussi « motes », représentent les nœuds qui constituent un réseau de capteurs sans fil (RCSF).

I.2.1. Architecture d'un capteur sans fil

Suivant le type d'application il existe une multitude de capteurs sur le marché. Cependant malgré cette diversité apparente, ils restent dotés d'une architecture matérielle similaire. Un capteur est composé principalement d'une unité de captage, traitement, stockage, communication et énergie. Des composants additionnels peuvent être ajoutés selon le domaine d'application [3] comme par exemple un système de localisation tel qu'un GPS, un générateur d'énergie (exemple: cellules solaires) ou un mobilisateur lui permettant de se déplacer. Ces éléments principaux et optionnels (représentés par les traits discontinus) sont visibles sur la figure I.2:

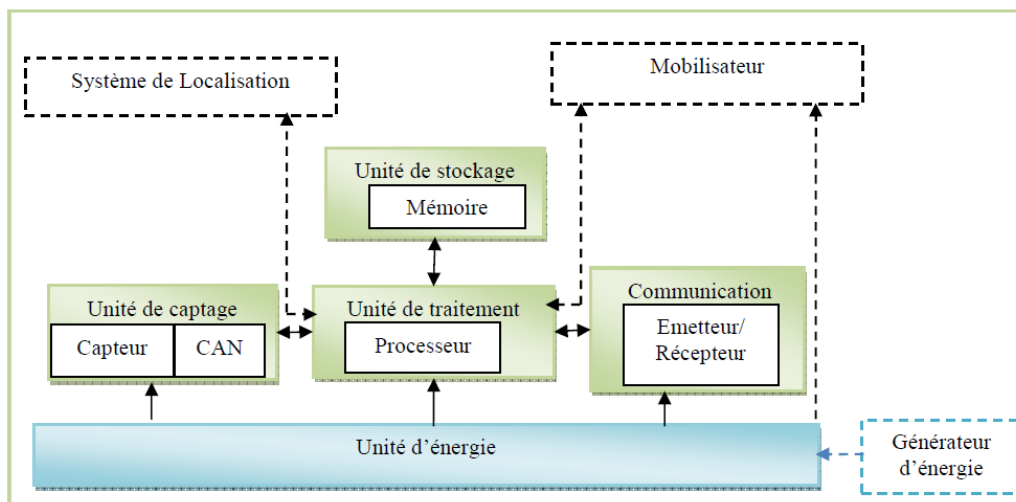


Figure I.2: Architecture matérielle d'un capteur.

Donc un capteur regroupe principalement les unités suivantes:

❖ Unité de captage (*Sensing unit*)

La fonction principale de l'unité de captage est de capturer ou mesurer les données physiques à partir de l'objet cible. Elle est composée de deux sous-unités: le récepteur (reconnaissant la grandeur physique à capter) et le transducteur (convertissant le signal du récepteur en signal électrique). Le capteur fournit des signaux analogiques, basés sur le phénomène observé, au convertisseur Analogique/Numérique(CAN). Ce dernier transforme ces signaux en données numériques et les transmet à l'unité de traitement. Un capteur peut avoir une ou plusieurs unités de captage [4].

❖ Unité de traitement (*Processing unit*)

Elle recueille des données de l'unité de captage ou d'autres capteurs, effectue un traitement sur ces données (si nécessaire) et décide quand et où les envoyer. Elle doit exécuter des programmes et des protocoles de communication différents. Les types de processeurs qui peuvent être utilisés dans un capteur incluant le Microcontrôleur, les DSP, les FPGA et les ASIC [4]. Parmi toutes ces alternatives, le microcontrôleur a été le processeur le plus utilisé pour les capteurs à cause de sa flexibilité à être relié à d'autres composants (comme par exemple l'unité de communication), à son bon prix et sa faible consommation énergétique [4, 5, 6].

❖ Unité de communication (*Transceiver unit*)

Cette unité est responsable de toutes les émissions et réceptions de données via un support de communication sans fil. Les différents choix de média de transmission incluent la Radiofréquence (RF), Laser et l'Infrarouge [4].

❖ Unité d'énergie (*Power unit*)

Un capteur est muni d'une source d'énergie, généralement une batterie [7], pour alimenter tous ses composants. Les batteries utilisées sont soit rechargeables ou non. Souvent, dans les environnements sensibles, il est impossible de recharger ou changer une batterie. Pour cela, l'énergie est la ressource la plus précieuse puisqu'elle influe directement sur la durée de vie des capteurs et donc d'un réseau de capteurs.

❖ Unité de stockage (Mémoire)

Elle inclut la mémoire de programme (dont les instructions sont exécutées par le processeur) et la mémoire de données (pour conserver des données fournies par l'unité de captage et d'autres données locales). La taille de cette mémoire est souvent limitée essentiellement par les considérations économiques et s'améliorera aussi probablement au fil des années [8].

1.2.2. Caractéristiques principales d'un capteur

Le fonctionnement d'un capteur est basé sur deux entités fondamentales, l'unité d'acquisition qui représente le cœur physique permettant la prise de mesure et l'unité de communication qui réalise le transfert de celle-ci, vers d'autres dispositifs électroniques. Ainsi fonctionnellement, chaque capteur possède un rayon de communication (R_c) et un rayon de sensation (R_s) [9].

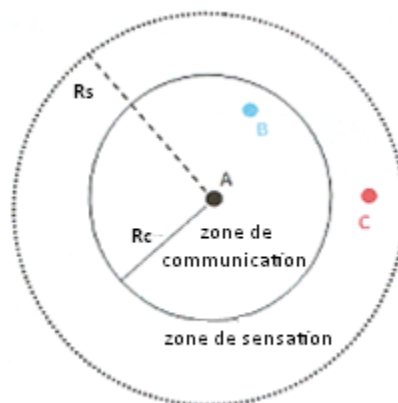


Figure I.3: Rayon de communication et de sensation d'un capteur [10, 11].

La figure I.3 montre les zones définies par ces deux rayons pour le capteur A. La zone de communication est la zone où le capteur A peut communiquer avec les autres capteurs (le capteur B). D'autre part, la zone de sensation est la zone où le capteur peut canaliser (capter) l'événement.

1.2.3. Différents nœuds capteurs

Les réseaux de capteurs offrent une grande diversité [12]:

-Diversité dans le matériel: MicaZ, Telos, Sky-Mote, équipés de microcontrôleurs **ARM7**, **AVR** ou **TI MSP430** ;

-Diversité dans l'interface radio et la couche physique: **Chipcon CC1100**, **Chipcon CC2420**, **868 MHz**, **2.4 GHz**, **Zigbee**, **modulation**, **frequency hopping**, etc.

-Diversité de point de vue du système d'exploitation: **TinyOS**, **MantisOS**, **SOS**, **Contiki**, **FreeRTOS**, **JITS**.

-Diversité dans les contraintes: énergie, mémoire, calcul.

-Diversité des applications: du domaine militaire au domaine civil, de la détection et suivi de cible à la mesure des ondes sismiques.

Comme on a mentionné auparavant les nœuds capteurs se déclinent en une multitude de modèles en fonction de l'application à laquelle ils sont destinés. Parmi les modèles les plus courants, on trouve les capteurs **MICA** développés par l'université de Berkeley et commercialisés par Crossbow, les capteurs **Imote** commercialisés par Crossbow, ainsi que les capteurs **TinyNode** développés pour des applications réelles liées à l'industrie par Shockfish SA [13].

Bien qu'ils soient différents, ces modèles ont en commun les mêmes composants de base. Les différents composants de chaque modèle ainsi que leurs caractéristiques sont donnés dans le tableau I.1:

	Modèle	Miro- contrôleur	Type radio	Radio (kbps)	RAM	Flash	EEPROM	Batterie
Crossbow Technology	MICA2	Atmega 128L MPR400(8- bit)	ChipCon CC1000	38	4 KB	128 KB	4 KB	2xAA
	MICAz	Atmega 128L MPR2400(8- bit)	ChipCon CC2420	250	4 KB	128 KB	4 KB	2xAA
	TelosB	TI MSP 430 (16-bit)	ChipCon CC2420	250	10 KB	48 KB	16 KB	2xAA
	Imote2	Intel PXA271 (32-bit)	ChipCon CC2420	250	32 MB	32 KB		3xAA
Shockfish SA	Tanynode	TI MSP 430 (16-bit)	Semtech XE1205	153	10 KB	48 KB	16 KB	2/3xAA

Tableau I.1: Les différentes technologies des nœuds capteurs [14].

I.2.3.1. MICA2 [15]

Le capteur MICA2 est un capteur de 3^{ème} génération utilisé pour les réseaux de capteurs sans-fil et à faible consommation, (voir la figure 1.4). Ce type de capteurs a été développé par l'université de Berkeley et est utilisé dans les applications suivantes:

- Contrôles environnementaux.
- Surveillance et sécurité.
- Réseaux de capteurs de grande capacité (+1000 nœuds).



Figure I.4: Capteur «MICA2».

I.2.3.2. TelosB

La plate-forme TelosB a été élaborée et publiée à la communauté scientifique par l'université de Berkeley. Cette plate-forme offre une faible consommation d'énergie permettant une longue autonomie de la batterie ainsi qu'un éveil rapide de l'état de veille. Le microcontrôleur TPR2420 utilisé dans TelosB, est compatible avec la distribution open-source de TinyOS. Ce type de nœud peut être utilisé dans les applications suivantes [15]:

- Plate-forme à faible puissance pour le développement de la recherche.
- Expérimentation des réseaux de capteurs sans-fil.



Figure I.5: Capteur «TelosB».

I.2.3.3. Imote2

L'Imote2 est une plate-forme avancée de nœuds capteurs sans-fil. Il est construit autour d'un processeur XScale PXA271 à faible puissance et intègre également une radio compatible 802.15.4. Ce type de nœud développé par Crossbow est utilisé dans les applications suivantes [15]:

- Traitement des images numériques.
- Contrôle et analyse industriels.
- Surveillance des séismes et des vibrations.



Figure I.6: Capteur «Imote2».

I.2.3.4. TinyNode

Le TinyNode est un nœud capteur développé par Shockfish SA. Il est optimisé de telle sorte à supporter TinyOS. Ce type de nœud, on le retrouve dans les applications suivantes [13]:

- Surveillance de l'environnement.
- Agriculture de précision.
- Gestion de stationnement.

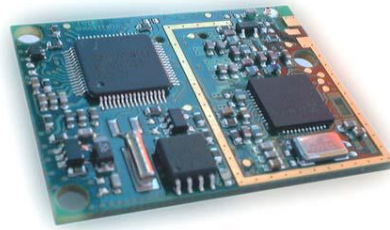


Figure I.7: Capteur «TinyNode 584».

I.2.4. Les états d'un nœud capteur [9]

Un capteur est initialisé avant toute utilisation (c'est l'état Initiale). Puis, il fonctionne selon trois modes:

- ❖ **Mode Actif**: où il exécute une fonction ou transmet un message.
- ❖ **Mode écoute (Idle)**: où il est seulement à l'écoute d'éventuels messages à recevoir.
- ❖ **Mode veille (Sleep)**: où il est en veille.
- ❖ Enfin, il se met dans l'état Stop lorsque la batterie est vide où qu'il est mis hors tension (bouton OFF).

Lorsqu'il est **actif**, le capteur peut envoyer et recevoir des messages (et seulement recevoir s'il est dans l'état **Idle**). La transmission des messages connaît trois états. Avant toute transmission, le module de communication du capteur (la radio) doit être dans l'état **Idle**, c'est-à-dire qu'aucune transmission n'est en cours sur le réseau. Le réseau doit être vide de message.

Un capteur ne peut pas envoyer et recevoir des messages simultanément. Il fonctionne en *half duplex*, c'est-à-dire qu'il ne fait qu'une transmission à la fois.

La figure I.8 présente les différents états d'un capteur:

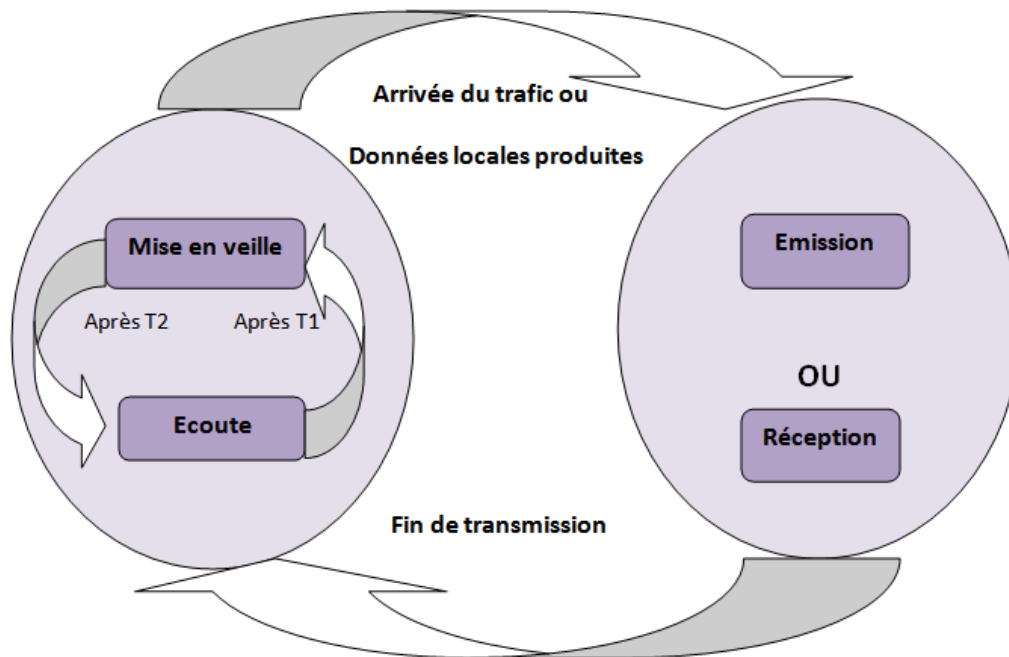


Figure I.8: Diagramme de transition d'état du modèle du nœud capteur.

Une interface radio engendre un coût énorme pour la consommation d'énergie. Pour réduire cette dernière, nous avons choisi de travailler avec des capteurs dont ils peuvent alterner entre le mode **sleep** et le mode **actif**, tout en gardant la connectivité et la couverture du réseau.

I.3. Réseaux de capteurs

I.3.1. Définition d'un réseau de capteurs sans fil [16]

Un réseau de capteurs sans fil ou WSN (Wireless Sensor Network) est un réseau composé d'un grand nombre de nœuds capteurs, avec une communication sans fil entre ces nœuds. Ces nœuds capteurs sont dispersés dans l'environnement (à surveiller) situé loin de l'utilisateur. Les entités principales qui constituent un réseau de capteurs sont:

- Les nœuds capteurs qui forment le réseau.
- La station de base qui communique avec l'utilisateur par l'intermédiaire d'Internet ou par une communication par satellite.
- Le phénomène faisant l'objet des mesures présentant de l'intérêt pour l'utilisateur.

I.3.2. Architecture de communication d'un RCSF

Les nœuds capteurs sont habituellement dispersés dans une zone de capture. Chacun de ces nœuds a la possibilité de collecter les données et de les router vers une ou plusieurs stations de base (*sink nodes*). Ce dernier est un point de collecte de données capturées. Il peut communiquer les données collectées à l'utilisateur final à travers un réseau de communication, éventuellement l'Internet. L'utilisateur peut à son tour utiliser la station de base comme passerelle, afin de transmettre ses requêtes au réseau. Cette architecture est illustrée dans la figure I.9.

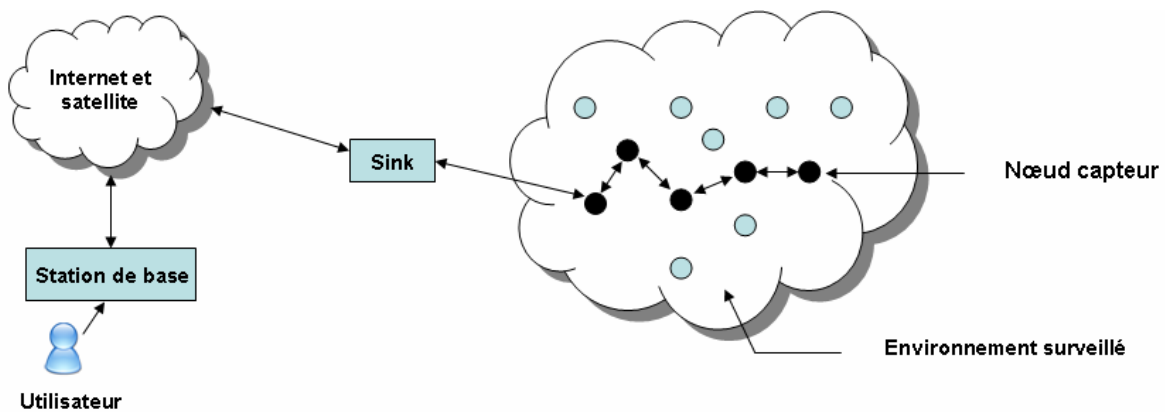


Figure I.9: Architecture de communication d'un RCSF [3].

I.3.3. Architecture protocolaire

La pile protocolaire [3] utilisée par la station de base ainsi que tous les autres capteurs du réseau est illustrée par la figure I.10. Ce modèle comprend 5 couches (une couche application, une couche transport, une couche réseau, une couche liaison de données, une couche physique) qui ont les mêmes fonctions que celles du modèle OSI, ainsi que trois plans qui sont: un plan de gestion d'énergie, un plan de gestion de mobilité, et un plan de gestion des tâches.

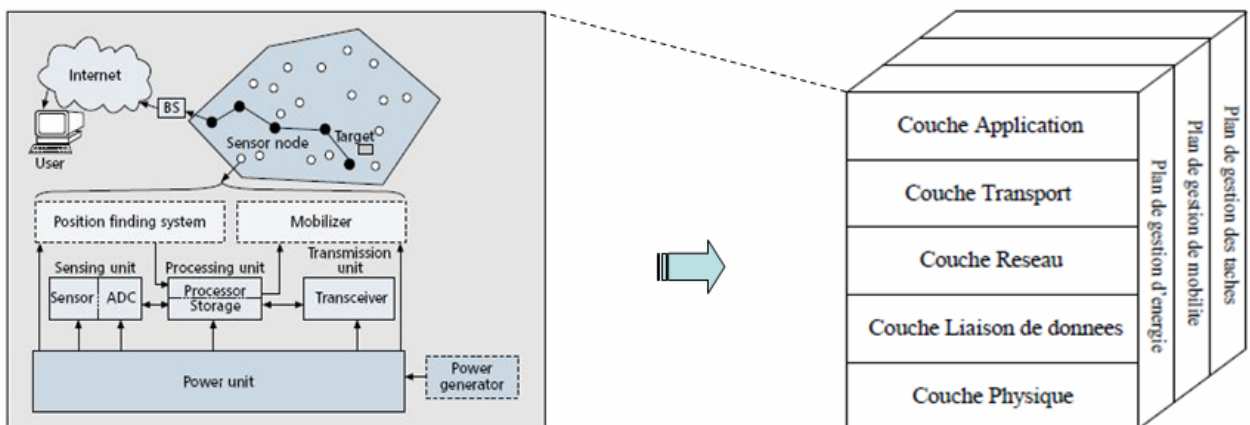


Figure I.10: Pile protocolaire.

❖ Couche physique

La couche physique est responsable de:

- ✓ La sélection des fréquences porteuses.
- ✓ La détection du signal.
- ✓ La modulation.

❖ Couche liaison de données

En général, cette couche est responsable du multiplexage du flux de données, du contrôle d'accès aux média (MAC Media Acces Control), et du contrôle des erreurs. Elle assure une connexion fiable (point-à-point et point-à-multipoints) selon la topologie du réseau de capteurs.

❖ Couche réseau

Le but principal de la couche réseau est de trouver une route et une transmission fiable des données captées des nœuds capteurs vers le puits en optimisant l'utilisation de l'énergie des capteurs. Les caractéristiques spécifiques aux RCSFs exigent que leurs protocoles de routage diffèrent de ceux des réseaux ad hoc traditionnels. Le routage dans les RCSFs sera étudié en détails dans le deuxième chapitre de ce mémoire.

❖ Couche transport

Cette couche est chargée du transport de données, de leur découpage en paquets, du contrôle de flux, de la conservation de l'ordre des paquets et de la gestion des éventuelles erreurs de transmission.

❖ Couche application

Elle constitue l'ensemble des applications implémentées sur un réseau de capteurs. Ces applications doivent fournir des mécanismes permettant à l'utilisateur d'interagir avec le RCSF à travers différentes interfaces, et éventuellement par l'intermédiaire d'un réseau étendu (par exemple: internet). Il s'agit donc du niveau le plus proche des utilisateurs, géré directement par les logiciels.

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

➤ Plan de gestion d'énergie

Un nœud capteur sans fil, nécessite seulement une source d'énergie limitée (< 0.5 Ah, 1.2V). La vie du nœud montre une dépendance forte à l'égard de la vie de la batterie. Le plan de gestion d'énergie doit gérer la manière dont les nœuds utilisent leurs énergies. Par exemple le nœud doit se mettre en sommeil après la réception d'un message à partir d'un voisin afin d'éviter la duplication des messages déjà reçus.

➤ Plan de gestion de mobilité

Puisque les nœuds peuvent être mobiles, un système de gestion de mobilité doit exister. Un tel système doit être capable d'enregistrer les mouvements du nœud afin de l'aider à se localiser.

➤ Plan de gestion de tâches

Lors d'une opération de captage dans une région donnée, les nœuds composants le réseau ne doivent pas obligatoirement travailler avec le même rythme, cela dépend essentiellement de la nature du capteur, son niveau d'énergie et la région dans laquelle il a été déployé. Pour cela, le niveau de gestion des tâches assure l'équilibrage et la distribution des tâches sur les différents nœuds du réseau, afin d'assurer un travail coopératif et efficace en matière de consommation d'énergie, et par conséquent, prolonger la durée de vie du réseau.

I.3.4. Types d'architecture des RCSFs

Selon [17], il existe deux types d'architectures pour les réseaux de capteurs sans-fil:

I.3.4.1. Architecture plate ou horizontale

Dans ce type de configuration, tous les nœuds ont un même niveau et peuvent communiquer avec tous les autres nœuds (figure I.11).

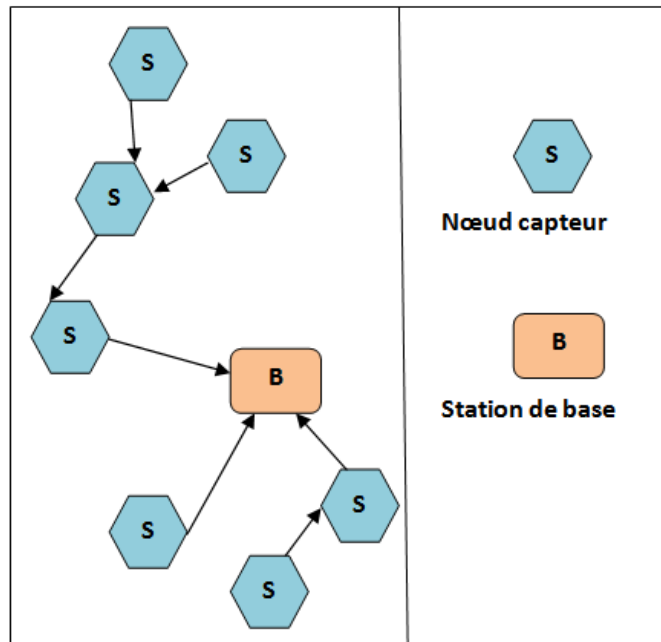


Figure I.11: Architecture plate [18].

On peut distinguer deux schémas:

Centralisé: dans lequel, toutes les données capturées sont envoyées vers un nœud central qui fait le traitement et la fusion des données pour les transmettre à la station de base. Ce schéma est très simple et il est utilisé seulement pour des réseaux de petite densité.

Distribué: il est plus compliqué. Plusieurs nœuds de traitement de données existent et peuvent communiquer entre eux. Un groupe de nœuds récoltent chacun de leur côté les données du réseau et les communiquent entre eux jusqu'à acheminement à la station de base.

I.3.4.2. Architecture hiérarchique

Une architecture hiérarchique a été proposée pour réduire la complexité de la plupart des nœuds capteurs et leur déploiement, en introduisant un ensemble de nœuds capteurs plus puissants. L'architecture hiérarchique est composée de plusieurs couches : une couche de capteurs, une couche de transmission et une couche de point d'accès.

Donc, le réseau est découpé en clusters ; dans chaque cluster, un Cluster-Head est élu pour gérer les communications inter et intra cluster. Toutes les données reçues d'un niveau inférieur sont traitées et agrégées par les cluster-heads de ce niveau avant d'être transmises vers le niveau supérieur.

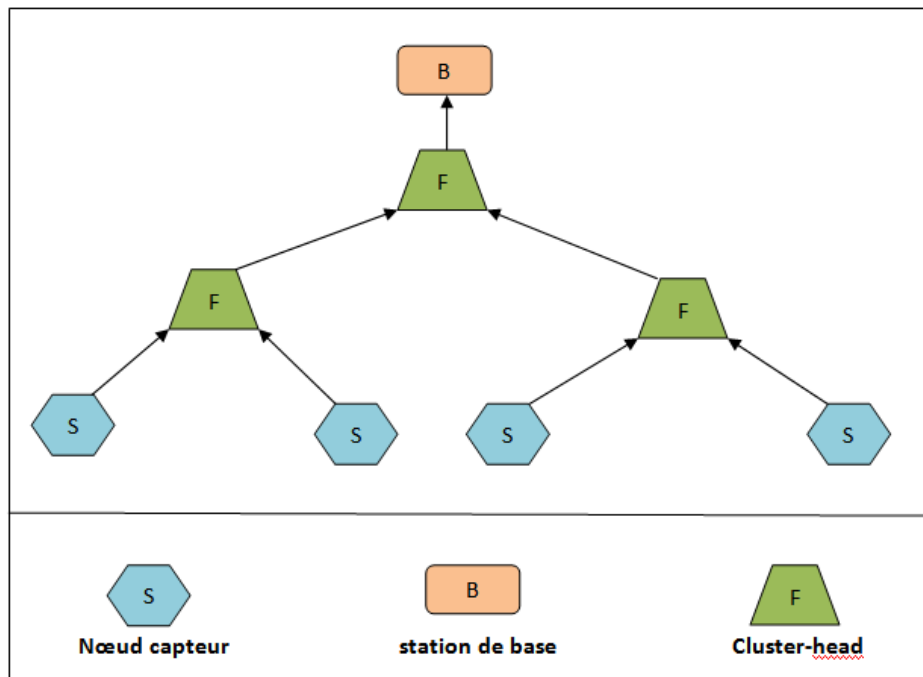


Figure I.12: Architecture hiérarchique [18].

I.4. Contraintes de conception des RCSF

Les principaux facteurs et contraintes influençant l'architecture des réseaux de capteurs peuvent être résumés comme suit [3]:

I.4.1. La tolérance aux fautes

La tolérance aux fautes est la capacité de maintenir les fonctionnalités du réseau en présence de fautes. La fiabilité des réseaux de capteurs sans fil est affectée par des défauts qui se produisent à cause de diverses raisons telles que le mauvais fonctionnement du matériel ou à cause d'un manque d'énergie. Ces problèmes n'affectent pas le reste du réseau [19, 20].

I.4.2. Facteur d'échelle (Scalabilité)

L'une des caractéristiques des RCSFs est qu'ils peuvent contenir des centaines voir des milliers de nœuds capteurs. Suivant l'application, ce nombre peut encore augmenter jusqu'à des millions de capteurs. Les nouveaux schémas doivent pouvoir garantir un bon fonctionnement avec ce nombre élevé de capteurs. Ils doivent aussi exploiter la nature fortement dense des réseaux de capteurs [21].

I.4.3. Coût de production

Le coût de production d'un seul micro-capteur est très important pour l'évaluation du coût global du réseau, si ce dernier est supérieur à celui nécessaire pour le déploiement des capteurs classiques, l'utilisation de cette nouvelle technologie ne serait pas financièrement justifiée. Par conséquent, réduire le coût de production jusqu'à moins de 1 dollar par nœud est un objectif important pour la faisabilité de la solution des réseaux de capteurs sans fil [3].

I.4.4. Environnement

Les nœuds capteurs doivent être conçus d'une manière à résister aux différentes et sévères conditions de l'environnement : forte chaleur, pluie et humidité [3].

I.4.5. Médias de transmission

Dans un réseau de capteurs, les nœuds sont reliés par une architecture sans fil. Pour permettre des opérations sur ces réseaux dans le monde entier, le média de transmission doit être standardisé. On utilise le plus souvent l'infrarouge, le Bluetooth [18] et les communications radio ZigBee [22].

I.4.6. Système d'exploitation

TinyOS est parmi les systèmes d'exploitation open-source pour les réseaux de capteurs, développé par l'université américaine de BERKELEY. Le caractère open-source permet à ce système d'être régulièrement enrichie par une multitude d'utilisateurs. Sa conception a été entièrement réalisée en NesC, langage orienté composant syntaxiquement proche du C. Il respecte une architecture basée sur une association de composants, réduisant ainsi la taille du code nécessaire à sa mise en place. Cela s'inscrit dans le respect des contraintes de mémoires qu'observent les capteurs pourvus de ressources très limitées dues à leur miniaturisation. Pour autant, la bibliothèque de composants de TinyOS est particulièrement complète puisqu'on y retrouve des protocoles réseaux, des pilotes de capteurs et des outils d'acquisition de données.

Un programme s'exécutant sur TinyOS est constitué d'une sélection de composants systèmes et de composants développés spécifiquement pour l'application à laquelle il sera destiné (mesure de température, du taux d'humidité...). TinyOS s'appuie sur un fonctionnement évènementiel, c'est-à-dire qu'il ne devient actif qu'à l'apparition de certains évènements, par exemple l'arrivée d'un message radio. Le reste du temps, le capteur se trouve en état de veille, garantissant une durée de vie maximale connaissant les faibles ressources énergétiques des capteurs [3].

I.4.7. Sécurité physique limitée

Les réseaux de capteurs sans-fil sont plus touchés par le paramètre de sécurité que les réseaux filaires classiques. Cela se justifie par les contraintes et limitations physiques qui font que le contrôle des données transférées doit être minimisé [3].

I.4.8. Topologie de réseau

Le déploiement d'un grand nombre de nœuds nécessite une maintenance de la topologie. Cette maintenance consiste en trois phases : déploiement, post-déploiement (les capteurs peuvent bouger, ne plus fonctionner,...) et redéploiement de nœuds additionnels [23].

I.4.9. Connectivité

Un réseau de capteurs est dit connecté si et seulement si, il existe au moins une route entre chaque paire de nœuds [24]. La connectivité dépend essentiellement de l'existence des routes. Elle est affectée par les changements de topologie dus à la mobilité, la défaillance des nœuds, attaques, etc. Ce qui a pour conséquences : la perte de liens, l'isolement des nœuds, le partitionnement du réseau, la mise à jours des routes (le routage), etc.

Un graphe G est dit k -connecté, s'il y a au moins k chemins disjoints entre deux nœuds quelconques. La connectivité est une mesure de tolérance aux fautes ou de diversité de chemin dans le réseau. Le 1-connectivité est une condition fondamentale pour que le réseau soit opérationnel. En effet la connectivité d'un réseau s'exprime de la façon suivante [25]:

$$\mu(R) = \frac{N \cdot \pi \cdot R^2}{A}$$

Où:

- **R**: est le rayon de transmission d'un nœud.
- **A**: l'aire de calcul.
- **N**: le nombre de nœuds situés dans l'aire A .

I.4.10. Consommation d'énergie

Un capteur, par sa taille, est limité en énergie (<1.2V). Dans la plupart des cas le remplacement de la batterie est impossible. Ce qui veut dire que la durée de vie d'un capteur dépend grandement de la durée de vie de la batterie. Dans un réseau de capteurs (multi-sauts) chaque nœuds collecte des données et envoie/transmet des valeurs [26]. Le dysfonctionnement de quelques nœuds nécessite un changement de la topologie du réseau et un re-routage des paquets. Toutes ces opérations sont gourmandes en énergie, c'est pour cette raison que les recherches actuelles se concentrent principalement sur les moyens de réduire cette consommation [27].

I.4.11. Agrégation de données

Dans les RCSFs, les données produites par les nœuds voisins sont très corrélés spatialement et temporellement. Ceci peut engendrer la réception par la station de base d'informations redondantes. Réduire la quantité d'informations redondantes transmises par les capteurs permet de réduire la consommation d'énergie dans le réseau et ainsi d'améliorer sa durée de vie. L'une des techniques utilisées pour réduire la transmission d'informations redondantes est l'agrégation de données, appelée aussi fusion de données [3].

I.5. Réseaux de capteurs vs réseaux ad hoc

Les MANETs et les RCSFs partagent quelques points communs mais quelques fois avec un poids différent. Le tableau I.2 résume les similitudes et les différences entre ces deux types de réseaux:

<i>Réseau de capteur sans fil</i>	<i>Réseau MANET</i>
➤ Utilisation d'un médium sans fil.	➤ Utilisation d'un médium sans fil.
➤ Déploiement ad hoc.	➤ Déploiement ad hoc.
➤ Robuste aux pannes des nœuds (auto-organisation).	➤ Robuste aux pannes des nœuds (auto-organisation).
➤ Routage multi-saut.	➤ Routage multi-saut.
➤ Les nœuds collaborent pour remplir un objectif.	➤ Chaque nœud a son propre objectif.
➤ Flot de données « many-to-one ».	➤ Flot « any-to-any ».
➤ L'énergie est un facteur déterminant.	➤ Le débit est majeur.
➤ Utilisation du broadcast.	➤ Communication point à point.
➤ La mobilité des nœuds est restreinte.	➤ La mobilité des nœuds.
➤ Grand nombre de nœuds (de l'ordre de mille).	➤ Nombre de nœuds moyen (de l'ordre de cents).
➤ Les nœuds ont une basse capacité de traitement et de stockage.	➤ Les nœuds ont une grande capacité de traitement et de stockage.
➤ Un objectif ciblé.	➤ Générique/ communication.
➤ Data-centric: souvent pas d'adresses uniques, les requêtes sont envoyés à tous les nœuds.	➤ Adress-centric: une adresse unique pour chaque nœud utilisée pour réaliser la communication entre les nœuds.

Tableau I.2: Comparaison entre les réseaux de capteurs et les réseaux MANET [28].

I.6. Technologies des réseaux de capteurs [29]

Parmi les technologies radio utilisées pour permettre les communications entre capteurs, nous pouvons citer : Bluetooth (IEEE 802.15.1), UWB (ou IEEE 802.15.3) et Zigbee (IEEE 802.15.4)...

I.6.1. Bluetooth

La technologie Bluetooth, dont Ericsson a initié le projet en 1994, a pour but principal de remplacer les câbles sur les petites distances. Elle est utilisée dans la plupart des téléphones portables comme interface de connexion pour accéder à un PC. Malheureusement, le grand défaut de cette technologie est sa trop grande consommation d'énergie. Donc elle ne peut pas être utilisée par des capteurs qui sont alimentés par une batterie et qui, idéalement devraient fonctionner durant plusieurs années [30].

I.6.2. Zegbee

Le standard Zigbee, combiné avec IEEE 802.15.4, offre des caractéristiques qui répondent encore mieux aux besoins des réseaux de capteurs. Zegbee offre des débits de données moindres, mais il consomme également nettement moins que Bluetooth. Un petit débit de données (250Kbit/s max) n'est pas handicapant pour un réseau de capteurs où les fréquences de transmission ne sont pas soutenues et conséquentes [30].

I.6.3. UWB

L'UWB peut être utilisé en tant que technique de communication sans fil, qui fournit des taux de transfert réseaux très élevés sur des distances relativement courtes et à faible puissance. De plus, l'atténuation du signal engendré par des obstacles est moindre qu'avec les systèmes radio à bande étroite conventionnels [31] [32].

I.6.4. Infrarouge

Les ondes infrarouges sont utilisées pour la communication à courte distance, ainsi elles s'adaptent au mode de communication entre les nœuds d'un réseau de capteurs [33]. Les ondes infrarouges sont préférées aux ondes radio, car elles n'interfèrent pas avec les autres signaux électromagnétiques.

Les émetteurs-récepteurs utilisant la technologie infrarouge ne sont pas chers, et ils sont faciles à fabriquer [33]. Les composants matériels utilisent ce type de communication sont généralement conformes aux standards publiés par l'Irfrared Data Association (IrDA).

Enfin, le tableau ci-dessous récapitule la différence entre les technologies ZigBee, Bluetooth et Wi-Fi.

Protocole	Zigbee	Bluetooth	Wifi
IEEE	802.15.4	802.15.1	802.11a/b/g
Besoin mémoire	4-32Ko	250 Ko	1 Mo
Energie nécessaire	Faible	Moyenne	Elevée
Nombre de nœuds	Non limité	7	32
Vitesse de transfert	250 Kb/s	720 Kb/s	11-54-108 Mb/s
Portée	100 m	1-100m	300m
Temps de démarrage	Court	Moyen	Long

Tableau I.3: Technologies sans fil et leurs caractéristiques.

I.7. Les applications des RCSF

La taille de plus en plus réduite des micro-capteurs, leur coût de plus en plus faible, la large gamme des types de capteurs disponibles (thermique, optique, de vibrations, etc.) ainsi que le support de communication sans fil utilisé, permettant aux réseaux de capteurs d'envahir rapidement plusieurs domaines d'applications [34]. En effet, ces réseaux se révèlent très utiles dans de nombreuses applications lorsqu'il s'agit de collecter et de traiter des informations provenant de l'environnement. Parmi ces domaines, nous citons les suivants: militaire, environnemental, écologique, domestique, de santé, de sécurité, etc.

I.7.1. Applications militaires

Le déploiement rapide, l'auto-configuration et la tolérance aux pannes des réseaux de capteurs sont des caractéristiques qui font de ce type de réseau un outil appréciable dans un tel domaine. Déploiement sur un endroit stratégique ou difficile d'accès, afin de surveiller toutes les activités des forces ennemies ou d'analyser le terrain avant d'y envoyer des troupes: par exemple, la détection d'agents chimiques, biologiques ou de radiations [35].



Figure I.13: Applications militaires des RCSFs.

I.7.2. Applications médicales

Le champ de contrôle de santé représente un grand marché pour les réseaux de capteurs sans fil qui a tendance à croître très rapidement. Ces réseaux de capteurs peuvent être utilisés pour assurer une surveillance permanente des organes vitaux de l'être humain (surveillance de la glycémie, détection de cancer ...) grâce à des micro-capteurs qui pourront être avalés ou implantés sous la peau. Ils peuvent aussi faciliter le diagnostic de quelques maladies en effectuant des mesures physiologiques telles que: la tension artérielle, le rythme cardiaque,... à l'aide des capteurs ayant chacun une tâche bien particulière [36].

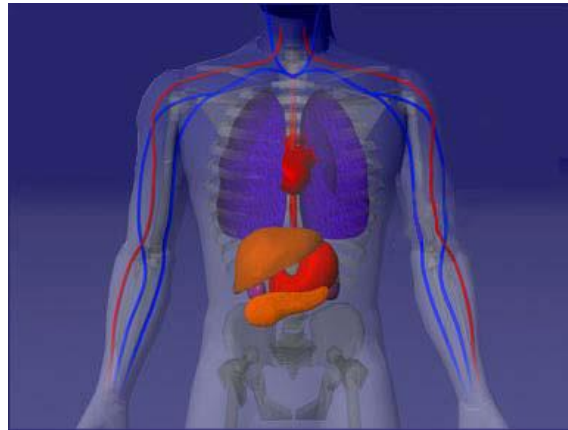


Figure I.14: Applications médicales des RCSFs.

I.7.3. Applications domestiques

Avec le développement technologique, les capteurs peuvent être embarqués dans des appareils, tels que les aspirateurs, les fours à micro-onde, les réfrigérateurs, les magnétoscopes, etc. Ces capteurs embarqués peuvent interagir entre eux et avec un externe via internet pour permettre à un utilisateur de contrôler les appareils domestiques localement ou à distance. En plaçant des capteurs sur le plafond ou dans le mur, on peut économiser l'énergie en gérant l'éclairage ou le chauffage en fonction de la localisation des personnes et seulement si c'est nécessaire [37].

I.7.4. Applications environnementales

Les micro-capteurs dispersés à partir d'un avion dans une zone difficile d'accès peuvent permettre de détecter des incendies, surveiller des catastrophes naturelles (inondations, séismes, éruptions volcaniques), surveiller des phénomènes météorologiques, de détecter de la pollution (qualité des eaux, taux d'ensoleillement, taux de radioactivité, fuite du pétrole, taux de CO₂...) [38].



Figure I.15: Applications environnementales des RCSFs.

I.7.5. Applications industrielles

L'intégration des micro-capteurs dans un processus de stockage et de livraison de marchandises peut être utilisée pour connaître la position, l'état et la position de direction d'un paquet ou d'une charge. Pour les entreprises manufacturières, les réseaux de capteurs permettront de suivre le procédé de production à partir des matières premières jusqu'au produit final livré [35].



Figure I.16: *Applications industrielles des RCSFs.*

I.7.6. Applications écologiques

L'utilisation des réseaux de capteurs pour optimiser la consommation des ressources énergétique peut avoir une conséquence environnementale positive. Un exemple de ce type d'application est l'intégration de plusieurs micro-capteurs dans le système de climatisation et le chauffage des immeubles. Ainsi, la climatisation ou le chauffage ne sont déclenchés qu'aux endroits où il ya des personnes présentes et seulement si c'est nécessaire. Le système distribué peut aussi maintenir une température une température homogène dans les pièces. Utilisée à grande échelle, une telle application permettrait probablement de réduire la demande mondiale en énergie [35].

I.7.7. Agriculture de précision

Les réseaux de capteurs sont capables d'apporter des bénéfices considérables dans le domaine d'agriculture, grâce à leur habilité de surveiller les taux de pesticides dans l'eau, le degré d'érosion du sol, détection des parasites et le niveau de pollution de l'air en temps réel [35].



Figure I.17: *Applications d'agriculture des RCSFs.*

I.8. Conclusion

Les réseaux de capteurs restent une nouvelle technologie peu accessible au grand public. Elle est principalement répandue dans les laboratoires de recherches. Des progrès sont encore à réaliser dans ce domaine. Néanmoins ils correspondent à une certaine vision du futur et permettront des améliorations dans d'innombrables domaines de la vie quotidienne.

À cette politique, nous avons vu dans ce chapitre les concepts généraux liés aux réseaux de capteurs sans fil. Parmi ces concepts nous avons identifié les contraintes de conceptions des réseaux de capteurs sans fil. Ensuite nous avons entamé leurs fonctionnements, les technologies utilisées dans un tel domaine ; enfin leurs domaines d'application.

Dans le chapitre suivant, nous allons présenter la problématique de la consommation d'énergie dans les réseaux de capteurs sans fil.

II.1. Introduction

Dans les réseaux de capteurs sans fil, la consommation d'énergie est très importante puisque généralement les capteurs sont déployés dans des zones inaccessibles. Ainsi, il est difficile voire impossible de remplacer les batteries après leur épuisement. De ce fait, la consommation d'énergie au niveau des capteurs a une grande influence sur la durée de vie du réseau.

Dans ce chapitre, nous allons étudier la consommation d'énergie dans ce type de réseaux sans fil. Pour cela, nous avons organisé notre travail comme suit: nous allons évoquer tout d'abord le problème d'énergie au sein d'un réseau de capteurs. Nous allons présenter, ensuite les différents facteurs intervenants dans la dissipation d'énergie des capteurs. Puis, nous allons montrer les différentes techniques proposées et créées afin de minimiser cette consommation et qui peuvent être appliquées au niveau de la couche liaison.

Finalement, nous allons présenter une étude des deux techniques de conservation d'énergie de communication (le routage et le contrôle de la topologie).

II.2. Problème de l'énergie dans les RCSFs

La consommation d'énergie est un problème fondamental lorsque les capteurs sont déployés dans des zones inaccessibles ou encore déployés sur de grands espaces. Pour cela, les nœuds limités en termes d'énergie vont avoir un impact sur la durée de vie du réseau tout entier. Augmenter la durée de vie du réseau sans augmenter la capacité des batteries signifie réduire la dépense d'énergie des nœuds.

Pour le cas des applications de surveillance par exemple exigeant que le réseau de capteurs soit opérationnel pendant une très longue durée, économiser l'énergie est un véritable challenge. La littérature scientifique traitant l'efficacité énergétique dans les réseaux de capteurs est abondante. Avant de présenter les techniques permettant de rendre les réseaux de capteurs sans fil moins gourmands en énergie, nous commençons par une analyse des principales sources de dissipation d'énergie [38].

➤ **Energie de capture**

L'énergie de capture est dissipée pour accomplir les tâches suivantes: Échantillonnage, traitement de signal, conversion analogique/numérique et activation de la sonde de capture.

Cette énergie consommée au moment de la mesure varie suivant la nature du capteur. En effet, pour la capture des données scalaires simples, comme la température, vibrations, humidité ou la pression, quelques octets est largement suffisante pour leur codage (2 ou 3 octets). Tandis que pour une image il est généralement représenté sur plusieurs milliers d'octets (16384 octets pour une image de taille 128 × 128 codée sur 8 bits par pixel).

Malgré ces variétés, l'énergie d'acquisition représente un faible pourcentage de l'énergie totale consommée [38].

➤ **Energie de communication**

L'énergie de communication est déterminée par la quantité des données à communiquer et la distance de transmission, ainsi que par les propriétés physiques du module radio.

Cette énergie se décline en deux parties : l'énergie de réception et l'énergie de l'émission. Le coût de réception est fixe mais le coût en émission dépend de la puissance d'émission du signal radio, de débit binaire et de la taille des données à transmettre...

Quand la puissance d'émission est élevée, le signal aura une grande portée et l'énergie consommée sera plus élevée.

Notons que cette partie d'énergie dépensée représente la portion la plus grande de l'énergie totale consommée par un nœud capteur. C'est pour cette raison que nous allons étudier par la suite les techniques de minimisation de la consommation de cette énergie [39].

➤ **Energie de traitement**

L'énergie de traitement se divise en deux parties: l'énergie de commutation et l'énergie de fuite. L'énergie de commutation est déterminée par la tension d'alimentation et la capacité totale commutée au niveau logiciel (en exécutant un logiciel). Par contre, l'énergie de fuite correspond à l'énergie consommée lorsque l'unité de calcul n'effectue aucun traitement.

En général, l'énergie de traitement est faible par rapport à celle nécessaire pour la communication. Ce qui justifie notre choix de minimiser l'énergie au niveau de l'unité de communication [39].

II.3. Facteurs intervenants dans la consommation d'énergie

La consommation d'énergie dépend de plusieurs facteurs qui sont expliqués ci-dessous:

II.3.1. Technologie de communication

L'unité de communication met en œuvre des protocoles de communication dépendant de la technologie utilisée (par exemple Wi-Fi (802.11), Bluetooth (802.15.1), ZigBee (802.15.4), UWB (802.15.3) pour les technologies sans fil) et ils n'ont pas le même coût d'énergie.

II.3.2. Etat du module radio

Le module radio est le composant du nœud capteur qui consomme le plus d'énergie, puisque c'est lui qui assure la communication entre les nœuds. On distingue quatre états des composants radio (transmetteur et récepteur) : actif, réception, transmission et sommeil [40].

- **Etat actif:** la radio est allumée, mais elle n'est pas employée. En d'autres termes, le nœud capteur n'est ni en train de recevoir ni de transmettre. Cet état provoque une perte de l'énergie suite à l'écoute inutile du canal de transmission.
- **Etat sommeil:** la radio est mise hors tension.
- **Etat transmission:** la radio transmet un paquet.
- **Etat réception:** la radio reçoit un paquet.

Il est aussi à noter que le passage fréquent de l'état actif à l'état sommeil peut avoir comme conséquence une consommation d'énergie plus importante que de laisser le module radio en mode actif. Ceci est dû à la puissance nécessaire pour la mise sous tension du module radio. Cette énergie est appelée l'énergie de transition. Il est ainsi souhaitable d'arrêter complètement la radio plutôt que de transiter dans le mode sommeil. Le changement d'état du module radio doit être géré par un protocole de la couche MAC.

II.3.3. Accès au medium de transmission

La couche MAC joue un rôle important pour la coordination entre les nœuds et la minimisation de la consommation d'énergie. Dans cette section, nous allons analyser les principales causes de consommation d'énergie au niveau de la couche MAC [41].

II.3.3.A. Retransmission

Les nœuds capteurs possèdent en général une seule antenne radio et partagent le même canal de transmission. Par ailleurs, la transmission simultanée des données provenant de plusieurs capteurs peut produire des collisions et ainsi une perte de l'information transmise.

La retransmission des paquets perdus peut engendrer une perte significative de l'énergie.

II.3.3.B. Ecoute active

L'écoute active (*idle listening*) du canal pour une éventuelle réception de paquet qui ne sera pas reçu peut engendrer une perte importante de la capacité des nœuds en énergie. Pour éviter ce problème, il faut basculer les nœuds dans le mode sommeil le plus longtemps possible.

II.3.3.C. Surécoute

Le phénomène de surécoute (*overhearing*) se produit quand un nœud reçoit des paquets qui ne lui sont pas destinés (figure II.1).

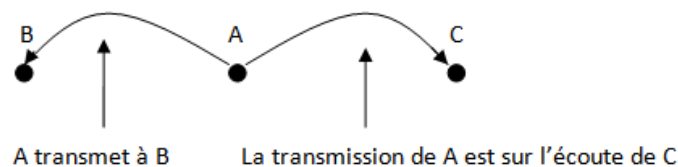


Figure II.1: La surécoute dans une transmission.

La surécoute conduit à une perte d'énergie additionnelle à cause de l'implication des autres capteurs dans la réception des données.

II.3.3.D. Surcharge

Plusieurs protocoles de la couche MAC fonctionnent par échange de messages de contrôle (*overhead*) pour assurer différentes fonctionnalités: signalisation, connectivité, établissement de plan d'accès et évitement de collisions. Tous ces messages nécessitent une énergie additionnelle.

II.3.3.E. Surémission

Le phénomène de surémission (*overemitting*) se produit quand un nœud capteur envoie les données à un destinataire qui n'est pas prêt à les recevoir. En effet, les messages envoyés sont considérés inutiles et consomment une énergie additionnelle.

II.3.3.F. Taille des paquets

La taille des messages échangés dans le réseau a un effet sur la consommation d'énergie des nœuds émetteurs et récepteurs. Ainsi, la taille des paquets ne doit être ni trop élevée ni trop faible. En effet, si elle est petite, le nombre de paquets de contrôle (acquiescement) générés augmente l'*overhead*. Dans le cas contraire, une grande puissance de transmission est nécessaire pour des paquets de grande taille.

II.3.4. Modèle de propagation radio

Le modèle de propagation représente une estimation de la puissance moyenne reçue du signal radio à une distance donnée d'un émetteur. La propagation du signal radio est généralement soumise à différents phénomènes: la réflexion, la diffraction et la dispersion par divers objets. Généralement, la puissance du signal reçue est de l'ordre de $(1/d^n)$, où d est la distance entre l'émetteur et le récepteur, n un exposant de perte d'un chemin (Exemple : $n=2$ dans le vide, de 4 à 6 dans un immeuble) [42].

II.4. Techniques de minimisation de la consommation d'énergie

Après la description des principales causes de consommation d'énergie dans les RCSFs, nous présentons dans ce qui suit les différentes techniques utilisées pour minimiser cette consommation. Ces techniques sont appliquées soit au niveau de la couche liaison soit au niveau de la couche réseau. Le schéma suivant (Figure II.2) donne un aperçu global de ces mécanismes:

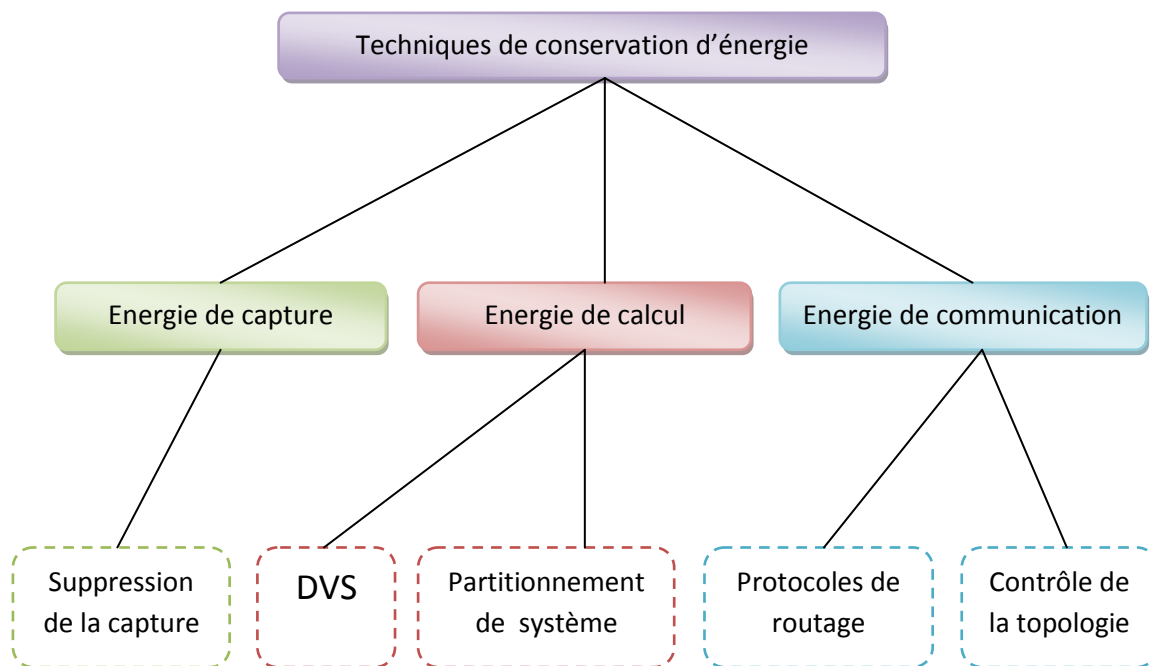


Figure II.2: Techniques de conservation d'énergie existantes [43].

Dans le cadre du projet « Etude et simulation des techniques de minimisation de la consommation d'énergie dans les RCSFs », nous proposons de minimiser la consommation d'énergie au cours de la communication afin d'augmenter la durée de vie des nœuds sans fil dans le réseau. Donc nous allons orienter tous nos travaux vers les techniques basées sur le routage et le contrôle de la topologie. Les présentations de ces deux mécanismes seront décrits ci-après.

II.4.1. Routage

Le rôle propre d'un capteur est de détecter un événement et d'en avertir une station de base. Alors le problème est l'acheminement du message émis par un capteur vers la station de base.

En effet, si un capteur n'atteint pas dans sa portée d'émission la station de base, il doit alors effectuer un routage dit multi-sauts. En d'autres termes, chaque capteur intermédiaire fera suivre le message jusqu'à la destination. La figure ci-dessous (figure II.3) illustre le schéma de routage vers la station de base lorsqu'un capteur détecte un événement.

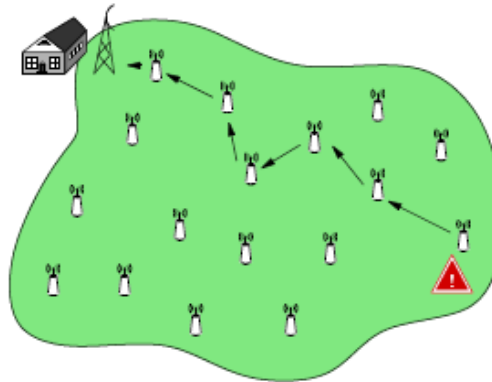


Figure II.3: Transmission d'un message vers la station de base lors de la détection d'un événement.

Par conséquent, la fonction de routage peut se définir comme la façon d'acheminer de l'information d'une source à un destinataire à travers un réseau de connexions donné.

Dans les réseaux de capteurs, les algorithmes de routage doivent garantir un fort taux d'acheminement des messages vers la station de base tout en limitant la consommation d'énergie par le billet de réduire le nombre de communications coûteuses en énergie. La résolution de ce problème d'optimisation bi-critère nécessite des protocoles de routage efficaces pour la gestion des contraintes des RCSFs citées dans le premier chapitre [44].

En résumé pour un bon routage il faut:

- Faible latence.
- Utilisation des meilleures liaisons de données.
- Minimisation de la charge du réseau, afin d'éviter le problème des collisions.
- Tolérance aux pannes et fiabilité.

III.4.1.1. Classification des protocoles de routages pour les RCSF

Le développement des protocoles de routage doit prendre en compte plusieurs facteurs dont la mobilité, changement brusque de la topologie, maintenir dynamiquement des routes et la gestion de la consommation d'énergie lors du routage des paquets.

Le routage dans les réseaux de capteurs sans fil présente des défis plus complexes en comparaison avec le routage dans les réseaux filaires traditionnels. En effet, une stratégie intelligente de routage est nécessaire pour supporter la nature et les paramètres du réseau (la mobilité, le nombre de nœuds, la densité du trafic, la qualité du service et la superficie du réseau) [46]. Les protocoles de routage dans les RCSFs peuvent être classés selon les concepts suivants:

- La topologie du réseau.
- Le mode d'établissement de la route.
- Le mode de fonctionnement du protocole.
- Les paradigmes de communication.

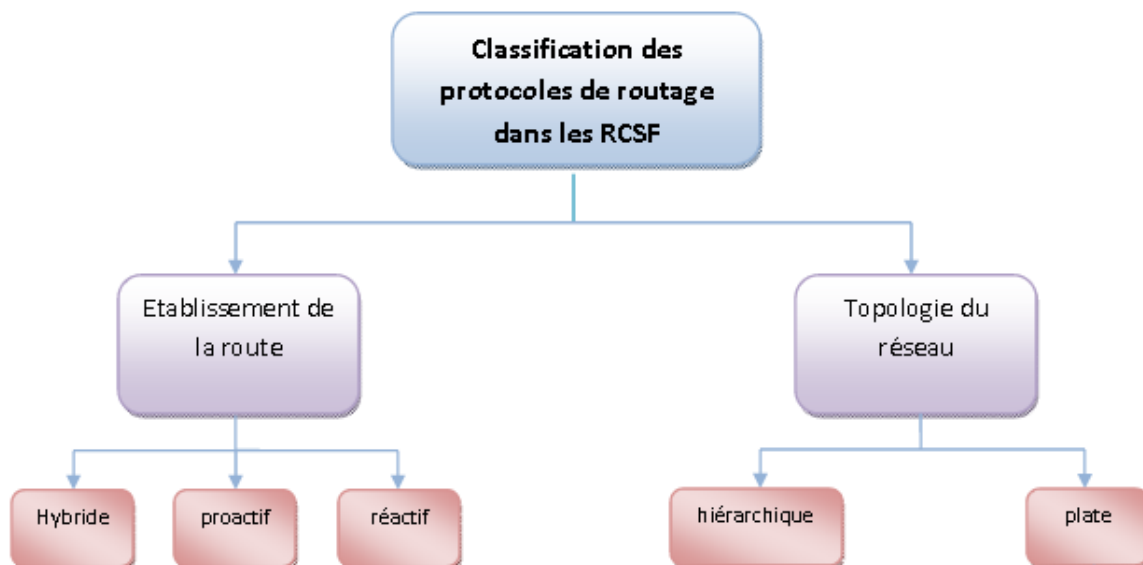


Figure II.4: Classification des protocoles de routage dans les RCSFs.

Notre étude s'intéresse beaucoup plus aux deux premiers concepts dont:

A. Selon la topologie du réseau

La topologie détermine l'organisation des capteurs dans le réseau. Globalement, il existe deux topologies pour les RCSF, topologie plate et hiérarchique [47].

1) Topologie plate

Dans ce type de topologie, tous les nœuds capteurs possèdent le même rôle et collaborent entre eux pour accomplir la tâche de routage. Les réseaux plats sont caractérisés par : la simplicité des protocoles de routage, un coup de maintien réduit, une grande tolérance aux pannes ainsi qu'une habilité à construire de nouveaux chemins suite aux changements de topologie. Cependant, les nœuds proches du puits participent plus que les autres aux tâches de routage. De plus, ces réseaux présentent une faible scalabilité due au fonctionnement identique des nœuds et d'une manière distribuée nécessitant ainsi un grand un grand nombre de messages de contrôle.

2) Topologie hiérarchique

Dans une topologie hiérarchique, les nœuds ont des différents rôles. En effet, certain nœuds sont sélectionnés pour exécuter des fonctions particulières. Une des méthodes les plus utilisées dans cette topologie est le clustering. Il consiste en un partitionnement du réseau en groupes appelés cluster. Un cluster est constitué d'un chef (clusterhead) et de ces membres.

Cette topologie présente beaucoup d'avantages tels que l'agrégation des données collectées ainsi qu'une grande scalabilité. Son inconvénient majeur est la surcharge des clusterheads qui induit à un déséquilibre de la consommation d'énergie dans le réseau. Pour remédier à

ce problème, les

clusterhead peuvent être des capteurs spécifiques avec plus de ressources énergétiques et plus de capacités de traitements ou bien ils peuvent être élus dynamiquement ainsi garantir un équilibre de la consommation d'énergie et augmenter la tolérance aux pannes. Un exemple de cette topologie est donné dans la figure ci-dessous, pour que les paquets générés par le nœud F atteignent le nœud L, ils doivent passer par les passerelles P, S et R.

B. Selon le mode d'établissement des routes

Suivant la manière dont les routes sont créées et maintenues pendant le routage, nous distinguons trois catégories de protocoles de routage, les protocoles réactifs, les protocoles proactifs et les protocoles hybrides [48].

1) Protocoles de routage proactifs

Les protocoles de routage proactif essaient de maintenir les meilleurs chemins existants vers toutes les destinations possibles (qui peuvent représenter l'ensemble de tous les nœuds du réseau), dans ce type de routage, les nœuds établissent les routes avant la demande, cette catégorie donne la plus grande importance à l'optimisation des chemins.

Au niveau de chaque nœud du réseau les routes sont sauvegardées même si elles ne sont pas utilisées. La sauvegarde permanente des chemins de routage est assurée par un échange continu des messages de mise à jour des chemins [49].

2) Protocoles de routage réactifs

Dits aussi protocoles de routage à la demande, créent et maintiennent les routes selon les besoins. Lorsqu'un nœud a besoin d'une route, une procédure de découverte globale est lancée. Cette procédure s'achève par la découverte de la route ou lorsque toutes les permutations de routes possibles ont été examinées. La route trouvée est maintenue par une procédure de maintenance de routes jusqu'à ce que la destination soit inaccessible à partir du nœud source ou que le nœud source n'aura plus besoin de cette route.

Comparés aux protocoles proactifs qui conservent les routes vers l'ensemble des stations du réseau dans leur table de routage, les protocoles réactifs ne conservent que les routes qui ont une utilité. Par conséquent, la taille des tables de routage contenues en mémoire est moins importante que pour les protocoles proactifs [49].

3) Protocoles de routage hybrides

Ce type de protocoles combine les mécanismes des protocoles proactifs et réactifs. Dans cette approche, les protocoles hybrides utilisent les méthodes proactives (messages périodiques de contrôle) pour découvrir les routes dans un voisinage prédéfini [46]. Au-delà de la zone de voisinage, le protocole hybride fait appel à un protocole réactif pour obtenir les routes vers les nœuds destinations [48].

II.4.2. Contrôle de la topologie

Le contrôle de topologie est une technique utilisée dans les réseaux de capteurs sans fil. Son principe est de modifier, réorganiser et gérer certains paramètres et modes de fonctionnalités des nœuds dans la topologie initiale du réseau. Cette technique consiste à éliminer du réseau les nœuds inutiles (par la mise en sommeil des nœuds redondants) et les liens inutiles (par l'ajustement de la

puissance de l'émetteur radio, et donc de la portée de communication), dans le but de diminuer la dépense d'énergie dans le réseau et prolonger sa durée de vie.

Le contrôle de topologie regroupe deux contraintes: la construction et la maintenance. La première contrainte est responsable de la réduction de topologie initiale, tandis que la seconde s'occupe de l'entretien de la topologie réduite tout en préservant les caractéristiques importantes du réseau comme la connectivité et la couverture. Cette topologie réduite doit être maintenue à jour de temps en temps car le réseau a besoin d'évoluer au fur et à mesure que des nœuds actifs arrivent à épuisement. Les détails de ces deux grandes contraintes seront présentés par la suite [53].

II.4.2.1. Construction de topologie

Cette phase comporte trois principaux mécanismes, le contrôle de la puissance de transmission, les approches hiérarchiques et hybrides. Notre cas s'intéresse beaucoup plus au contrôle de la topologie qui permet l'ajustement de la puissance de transmission et le regroupement des nœuds capteurs (hiérarchisation) [54].

A. Contrôle de la puissance de transmission

Ce mécanisme n'a pas seulement un effet sur la durée de vie de la batterie d'un nœud capteur, mais aussi sur la capacité de charge du trafic qui est caractérisée par le nombre de paquets transmis avec succès vers une destination. En outre, il influe sur la connectivité et la gestion de la densité (le nombre de nœuds voisins).

Ainsi, il peut conserver l'énergie à deux niveaux: explicitement par l'application de puissances faibles d'émissions et implicitement en réduisant la contention avec d'autres nœuds transmetteurs. Le contrôle de la puissance de transmission dans le cas homogène (c'est-à-dire le cas où tous les nœuds du réseau ont les mêmes caractéristiques), de manière centralisée est peut-être la technique la plus mature de la construction de topologie.

Parmi les approches les bien connues dans ce processus, nous citons la technique qui consiste à trouver la portée de communication minimale pour tous les nœuds du réseau et qui préserve sa connectivité, cette approche est mieux représentée par le problème de la chaîne de transmission critique CTR [54].

B. Hiérarchisation

L'approche hiérarchique de construction de topologie s'adresse au problème de l'évolutivité et facilite l'agrégation d'informations pour des économies d'énergie supplémentaires. Elle consiste à organiser le réseau en structure à plusieurs niveaux, où un sous ensemble de nœuds est sélectionné selon leurs fonctionnalités et activités par rapport aux autres nœuds du réseau.

C'est le cas, par exemple, des algorithmes de groupement (*clustering*), qui organisent le réseau en groupes (*clusters*) avec des chefs de groupe (*cluster head*) et des nœuds membres.

La classification des mécanismes de construction de topologie montre que les mécanismes basés sur la technique hiérarchique peuvent être classifiés comme: backbone-based, cluster-based et technique d'adaptation [55].

II.4.2.2. Maintenance de la topologie

Comme nous avons expliqué dans les sections précédentes, pour un certain nombre de nœuds distribués dans une zone spécifique, la construction de topologie vise à construire une topologie

réduite qui économise l'énergie tout en préservant la connectivité du réseau et la couverture de zone. Une fois que la topologie initiale créée a été réduite, le réseau commence à effectuer les tâches qu'il a été conçu pour elle.

Chaque activité impliquée dans la mission du réseau, comme la détection, le traitement et la distribution d'information à un coût associé qui consomme l'énergie des nœuds actifs.

Par conséquent, dans le but d'augmenter la durée de vie du réseau, l'ensemble des nœuds actifs ceux de la topologie réduite, ne peuvent pas être actifs tout le temps. Par conséquent, des mécanismes de maintenance de la topologie doivent être mis en place pour reconstruire une nouvelle topologie réduite avec la collaboration des nœuds inactifs de sorte que tous les nœuds appartenant au réseau consomment leur énergie de façon équitable et donc entraîne l'augmentation de la durée de vie du réseau.

La maintenance de la topologie est définie comme étant le processus qui restaure, fait pivoter, ou recrée la topologie du réseau de temps en temps lorsque le réseau réduit existant n'est plus optimal. Il consiste de faire tourner le rôle des nœuds autant que possible pour prolonger la durée de vie du réseau [55].

A. Techniques statiques globales de maintenance de la topologie

Ces techniques calculent toutes les différentes topologies pendant le premier processus de construction de la topologie. Ces topologies sont établies et enregistrées dans la mémoire, et elles seront commutées une fois qu'il est nécessaire. Donc les techniques statiques ont des topologies « pré-planifiées ».

Le meilleur exemple pour décrire le principe de ces techniques fait l'analogie avec les lumières d'un arbre de Noël: l'ensemble entier de lumières qui s'allument et s'éteignent périodiquement sont dans notre cas les topologies pré-planifiées qui couvrent l'arbre entier. C'est pour cette raison que ce type de techniques est appelé technique de l'arbre de Noël. Chaque fois qu'un nœud atteint un seuil d'énergie ou de temps, la technique statique globale de maintenance de la topologie fait tourner l'arbre de Noël.

Les techniques statiques de maintenance de topologie peuvent avoir quelques inconvénients. Par exemple, il est difficile de savoir a priori quelle est la topologie que ses nœuds consomment plus d'énergie. En plus, le mécanisme peut choisir d'employer quelques nœuds dans plus d'une topologie. [55].

B. Techniques dynamiques de maintenance de la topologie

Contrairement aux techniques statiques de maintenance de la topologie, les techniques dynamiques ne faisant pas un calcul à priori de déterminer la topologie qui devient active quand l'actuelle ne convient plus. Les techniques dynamiques nécessitent plus de temps et consomment plus d'énergie car elles doivent lancer le processus de construction de topologie plusieurs fois. Puisque les techniques dynamiques prennent en considération les états actuels du réseau lors de nouveaux calculs, le processus de construction de topologie choisit habituellement une topologie optimale à chaque fois qu'il est exécuté, conduisant à des meilleures topologies subséquentes comparativement avec les techniques statiques. Pour cela les techniques dynamiques globales et locales de maintenance de la topologie sont décrites et évaluées dans les réseaux clairsemés et denses [55].

Les techniques dynamiques globales de maintenance de la topologie changent entièrement la topologie par l'exécution de nouveau mécanisme de construction de la topologie à chaque fois selon la nécessité.

C. Techniques hybrides de maintenance de la topologie

La technique hybride de maintenance de la topologie est une combinaison d'une technique globale statique et d'une autre dynamique.

La technique est statique globale lors du premier processus de construction de la topologie, elle calcule tous les nœuds disjoints possibles ou partage les topologies disjointes et commence à travailler comme une technique statique globale régulière qui fait tourner l'arbre de Noël chaque fois qu'un nœud atteint son seuil d'énergie ou de temps. Au fil du temps les différents arbres sont entraînés en rotation, le nouvel arbre désigné ne peut pas être en mesure de fournir son service, car il n'a aucune communication avec le nœud sink.

Dans la technique statique globale normale, cet arbre serait éliminé de la liste de rotation et sauté, et la technique continuera à travailler tant qu'il y aura ses valables arbres restants. Dans la technique hybride, lorsque le sous-arbre ne peut plus fournir son service, elle déclenchera une technique dynamique globale pour trouver une nouvelle topologie réduite qui doit contenir certains nœuds qui n'avaient jamais été utilisés avant par un arbre. Le nouveau arbre créé de façon dynamique travaille jusqu'à ce que le seuil d'énergie ou du temps de déclenchement est atteint, à ce moment la technique statique prendra le relais et poursuivra les topologies de commutation pré-planifiées. Cette procédure est répétée jusqu'à ce qu'il n'ya plus d'arbres statiques ou dynamiques qui peuvent être établis dans le réseau.

Les algorithmes spécifiques pour chaque technique de maintenance de la topologie seront bien détaillés dans le chapitre suivant [55].

II.5. Conclusion

Contrairement aux réseaux traditionnels qui se préoccupent de garantir une bonne qualité de service, les réseaux de capteurs sans fil doivent, en plus, prendre en compte la conservation d'énergie.

Ils doivent intégrer des mécanismes qui permettent aux utilisateurs de prolonger la durée de vie du réseau en entier, car chaque nœud est alimenté par une source d'énergie limitée et généralement irremplaçable. Dans un nœud capteur, l'énergie est consommée en assurant les fonctions suivantes: la capture, le calcul (traitement) et la communication. Cette dernière représente une grande portion de l'énergie totale consommée. De ce fait, la communauté de recherche a développé deux techniques de conservation d'énergie à ce niveau qui sont: le routage et le contrôle de la topologie.

Dans le prochain chapitre, nous examinerons en détail les protocoles de conservation d'énergie appartenant à ces deux techniques proposées pour la minimisation de la consommation d'énergie de communication.

III.1. Introduction

Dans les réseaux de capteurs sans fil, chaque nœud joue le rôle de source et de relais. De ce fait, la défaillance énergétique d'un capteur peut changer significativement la topologie du réseau et imposer une réorganisation coûteuse de ce dernier.

Dans ce type de réseaux, les protocoles visent à conserver le maximum d'énergie. Plusieurs travaux de recherche récents sont entrepris dans le contexte d'économie d'énergie en RCSF. Pour cela, des études ont été effectuées dans la communication des nœuds qui regroupe les algorithmes de routage et les protocoles de contrôle de la topologie. Ce chapitre a pour but d'analyser ces différents protocoles en effectuant une étude comparative pour chaque technique (routage et contrôle de la topologie).

III.2. Protocoles de routage minimisant la consommation des batteries dans les RCSFs

Dans cette section, nous allons citer quelques protocoles de routage qui vont être simulés dans le prochain chapitre pour étudier leur efficacité énergétique.

III.2.1. Protocole DSDV

Le protocole DSDV est un protocole de routage proactif. Il a été conçu spécialement pour les réseaux mobiles. Cet algorithme est basé sur la technique du vecteur de distance. Le protocole à vecteur de distance permet de limiter l'échange des messages de contrôle de la topologie uniquement aux voisins d'un nœud. Ce point est extrêmement important pour préserver la bande passante disponible sur le réseau [56]. Chaque nœud maintient une table de routage qui contient:

- toutes les destinations possibles.
- le nombre de nœuds (ou de sauts) pour atteindre une destination.
- le numéro de séquence SN qui correspond à un certain nœud destination: qui permet de distinguer les nouvelles routes des anciennes pour bien éviter les boucles de routage dans le réseau [57].

Afin d'établir ces tables de routages dans une topologie extrêmement variée, chaque nœud envoie périodiquement sa table de routage à ses voisins directs. Le nœud peut aussi transmettre sa table de routage si le contenu de cette dernière subit des changements par rapport au dernier contenu envoyé. Mais toutes ces mises à jour génèrent un trafic important qu'il faut limiter.

A la réception de ces informations, les voisins mettent à jour leur table de routage. Toute entrée de la table de routage est mise à jour, seulement, si l'information reçue existe déjà. En ce terme, le protocole DSDV fournit pour chaque destination, la route qui possède le plus faible nombre de nœuds. Le protocole DSDV doit maintenir l'état des chemins aussi pour cela les nœuds détectent les ruptures de lien. Chaque nœud émet, périodiquement, ses informations de routage à l'ensemble de ses voisins. Si pendant un certain temps, un nœud ne reçoit plus les informations de routage d'un nœud voisin c'est que ce dernier ne fait plus partie de son voisinage. Un lien coupé affecte l'ensemble des routes utilisant ce lien. Un nœud, apercevant une coupure, diffuse un paquet contenant les destinations qui ne peuvent plus être atteintes à travers ce lien. Tout nœud, recevant un tel paquet, le propage immédiatement pour faire connaître au plus vite le changement de topologie.

La mise à jour dépend donc de deux paramètres: le temps, c'est-à-dire la période de transmission, et les événements comme l'apparition d'un nœud, détection d'un nouveau voisin.

La mise à jour est soit :

- Incrémentale.
- Complète.

Un des problèmes de ce protocole est qu'il réagit trop lentement aux nouvelles, c'est-à-dire un nœud doit attendre jusqu'à ce qu'il reçoive la prochaine mise à jour initiée par la destination

III.2.2. Protocole DSR

Le protocole DSR est basé sur l'utilisation de la technique "routage source". Dans cette technique, la source des données détermine la séquence complète des nœuds à travers lesquelles les paquets de données seront envoyés. Un site initiateur de l'opération de « découverte de routes » diffuse un paquet requête de route. Si l'opération de découverte est réussie, l'initiateur reçoit un paquet réponse de route qui liste la séquence de nœuds à travers lesquelles la destination peut être atteinte. Le paquet requête de route contient donc la route enregistrée dans un champ où sera accumulée la séquence des nœuds visités durant la propagation de la requête dans le réseau. L'utilisation de la technique "routage source" fait que les nœuds de transit n'aient pas besoin de maintenir les informations de mise à jour pour envoyer les paquets de données, puisque ces derniers contiennent toutes les décisions de routage. Dans ce protocole, il y a une absence totale de boucle de routage car le chemin source/destination fait partie des paquets de données envoyés [58].

III.2.3. Protocole AODV

Il est utilisé pour des routages unicast et multicast en utilisant des requêtes de type (**route request / route reply**). Lorsque la source a des données à envoyer vers une destination, elle diffuse une requête de type RREQ. Lorsque RREQ arrive à la destination, celle-ci génère une réponse de type RREP envoyée vers la source comme le montre la figure III.1.

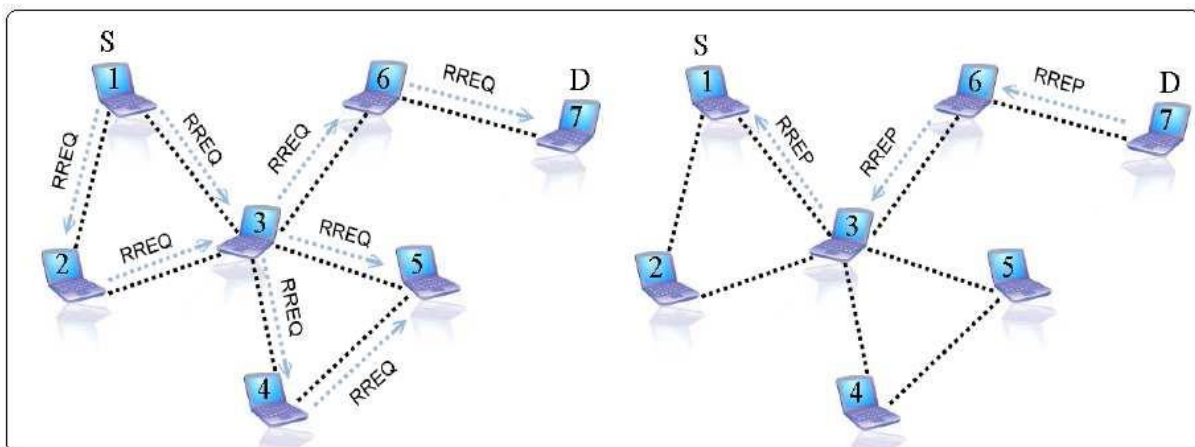


Figure III.1: Les deux requêtes RREQ et RREP en AODV

Avec AODV, s'il existe plusieurs routes possibles de la source vers la destination, AODV choisit la route la plus courte (la route où il y a un minimum de sauts). Si un tour de routage échoue, la source relance un nouveau RREQ avec un temps T plus important. Si plusieurs séries de Route Request échouent, alors aucune route ne peut être trouvée [59].

III.2.4. Protocole TORA

Le protocole TORA s'attaque aux problèmes d'économie de la bande passante en tentant de minimiser l'effet des changements fréquents de la topologie, qui due à la mobilité des nœuds.

Afin d'y parvenir, la recherche du meilleur chemin est délaissée non pas en terme de calcul mais en terme de procédure. De cette manière un protocole pourra choisir un plus long chemin entre la source et le nœud destination dans le but d'éviter le processus, coûteux, de découverte de nouveau voisin. De plus, TORA conserve plusieurs chemins vers une même destination et non plus seulement le meilleur chemin ce qui a pour conséquence de limiter les effets induits par une modification de la topologie sur le routage des données.

Le protocole est aussi caractérisé par la limitation des messages de contrôle à l'ensemble des nœuds proches de l'évènement.

TORA est basé sur l'utilisation de la propriété appelée "orientation destination" des graphes acycliques orientés. Un graphe est orienté si les liens qui le composent ont une direction, c'est à dire qu'un lien n'est pas forcément bidirectionnel. Un graphe acyclique signifie que le graphe ne possède aucune boucle. Un graphe acyclique orienté est dit orienté destination s'il y a toujours un chemin possible vers une destination spécifiée. Lorsque le graphe perd un ou plusieurs arcs de manière à devenir non orienté destination, alors les algorithmes utilisent le concept d'inversement de lien pour permettre de retrouver un graphe orienté destination. Pour réaliser ceci, TORA utilise le concept de taille des nœuds, la destination possède une taille nulle, et chaque nœud a pour taille, celle de son voisin possédant la plus petite taille incrémenté de un [60].

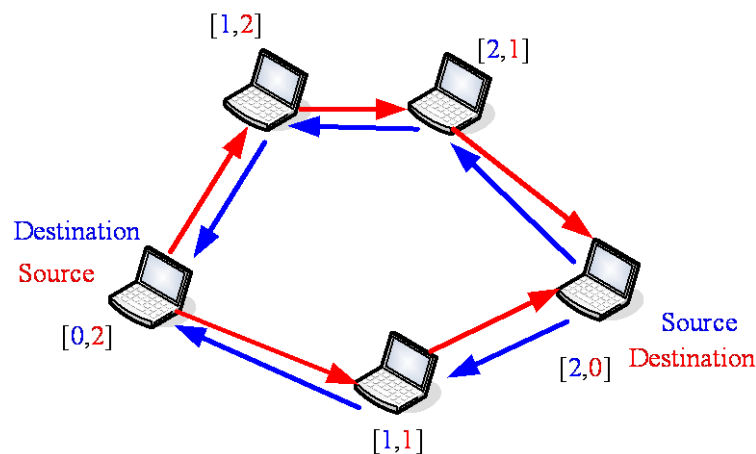


Figure III.2: Taille des nœuds avec TORA.

III.2.5. Protocole LEACH

LEACH, est le premier et le plus populaire des protocoles de routage hiérarchique avec une efficacité énergétique, qui a été proposé pour les RCSFs. Ce protocole emploie la technique de clustering qui divise le réseau en deux niveaux: les clusterheads et les nœuds membres. Dans le but de réduire la quantité d'informations transmises à la station de base, les clusterheads agrègent les données capturées par les nœuds membres qui appartiennent à leur propre cluster, et envoient un

paquet agrégé à la station de base. Cela permet d'économiser de l'énergie car les transmissions sont effectuées que par les clusterhead plutôt que par tous les nœuds du réseau [61].

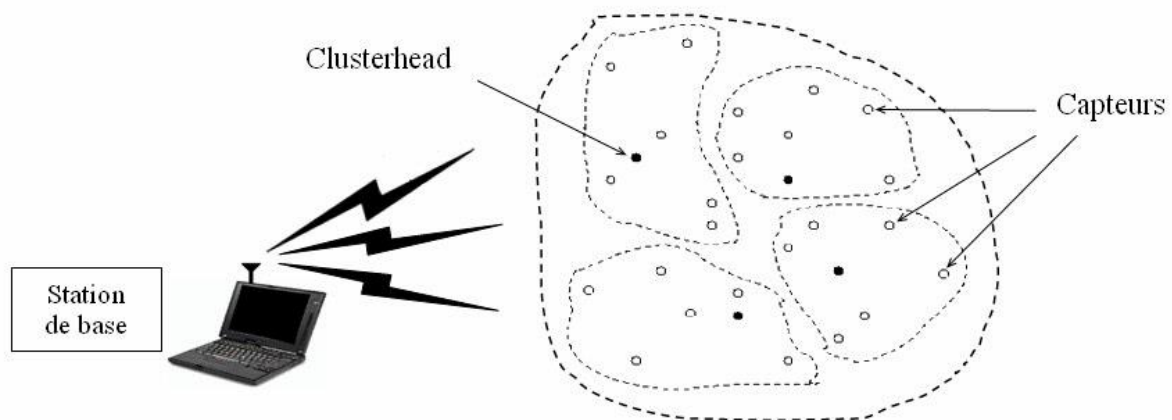


Figure III.3: Le Clustering dans un réseau de capteurs

III.2.6. Protocole AOMDV

Il enrichit son protocole unipath de base (AODV) en leur ajoutant deux notions supplémentaires: nombre de sauts annoncé et dernier nœud. Le champ nombre de sauts annoncé d'un nœud i pour une destination D représente le nombre de sauts de la route la plus longue entre les deux nœuds.

L'objectif primaire du protocole réactif AOMDV est de fournir une tolérance efficace aux fautes, dans le sens où le rétablissement en cas d'échec d'une route sera plus rapide. Pour réaliser cet objectif, AOMDV construit plusieurs routes sans boucles de routage au niveau des sources de trafic et des nœuds intermédiaires ; Dans ce qui suit, nous allons décrire la découverte et le maintien de la route dans AOMDV [62].

III.2.7. Comparaison entre les différents protocoles

Le protocole DSDV utilise les propriétés de la diffusion pour transmettre les informations de routage. En effet, le grand avantage de la diffusion est qu'une trame émise par une station est entendue par l'ensemble de ses voisins. Donc DSDV fournit à tout moment des routes valables vers toutes les destinations du réseau, mais l'inondation des paquets de mise à jour (périodique et en cas de changement de topologie) cause une charge de contrôle importante au réseau.

Le protocole AODV représente essentiellement une amélioration de l'algorithme DSDV, dans le contexte réactif. Il réduit le nombre de diffusions de messages, et cela en créant les routes lors du besoin, contrairement au DSDV, qui maintient la totalité des routes.

Le protocole AODV ne présente pas de boucle de routage. Par contre si le réseau est très utilisé il peut générer un trafic de contrôle énorme. Aussi l'AODV a un délai initial important avant de commencer la transmission des paquets.

L'AOMDV est une extension du protocole AODV où il a le même principe que celui d'AODV. Il a été conçu pour améliorer les performances du AODV en terme de consommation d'énergie.

Le protocole DSR utilise l'inondation pour découvrir les routes ce qui génère un trafic de contrôle énorme quand le réseau est très utilisé. La taille des paquets de données dans DSR devient

très grande quand le nombre de nœuds dans réseau est grand, puisque les paquets doivent porter les adresses de chaque nœud dans la route traversée.

Contrairement à AODV qui construit et maintient une seule route seulement pour chaque couple (source, destination). Avec TORA, une nouvelle découverte de route n'est nécessaire, que si toutes les routes deviennent non valides. Ceci réduit considérablement les latences et la surcharge de la procédure de découverte.

Le protocole LEACH emploie la technique de clustering qui divise le réseau en deux niveaux: les clusterheads et les nœuds membres, donc les communications se fait seulement entre les nœuds clusterheads ce qui réduit efficacement la consommation d'énergie.

III.3. Protocoles de contrôle de la topologie minimisant la consommation des batteries dans les RCSFs

Cette partie décrit les protocoles des deux mécanismes, construction et maintenance constituant le processus de contrôle de la topologie.

III.3.1. Construction de la topologie

Sachant que les critères nécessaires qui doivent être fournis par le contrôle de topologie sont la connectivité et la couverture. Nous allons exposer les différents protocoles assurant ces deux dernières.

III.3.1.a. A3: connectivité

L'algorithme A3 [63], dont le nom vient de «A Tree» est le premier membre de la famille des protocoles de construction de la topologie. Cet algorithme crée un arbre qui couvre chaque nœud dans la topologie et fournit une dorsale de communication pour le réseau en utilisant la technique «pousser un arbre». Le principe de fonctionnement du protocole A3 est basé sur les processus suivants:

➤ **Processus de découverte de voisinage:**

1. Tous les nœuds commencent par un état initial (non-visité), sauf le nœud «sink» qui commence avec un état candidat actif. Le sink est le nœud A dans la figure III.4 (a).
2. Il envoie un message «bonjour» à tous ses voisins, lui permettant de reconnaître leurs positions dominantes au nœud parent. Dans la figure III.4 (a), les nœuds B, C, D, E reçoivent le message, tandis que les nœuds F et G sont hors portée du nœud A.
3. Chaque nœud voisin recevant le message «bonjour» renvoie un message de «reconnaissance des parents» comme le montre la figure III.4 (b). Ce message inclut son ID et sa propre métrique de sélection, qui est une combinaison convexe du rapport de l'énergie restante dans le récepteur, et du rapport de la distance entre le nœud et son nœud parent. Puis il adopte aussi l'expéditeur comme son nœud parent, et il change son état en nœud enfant.

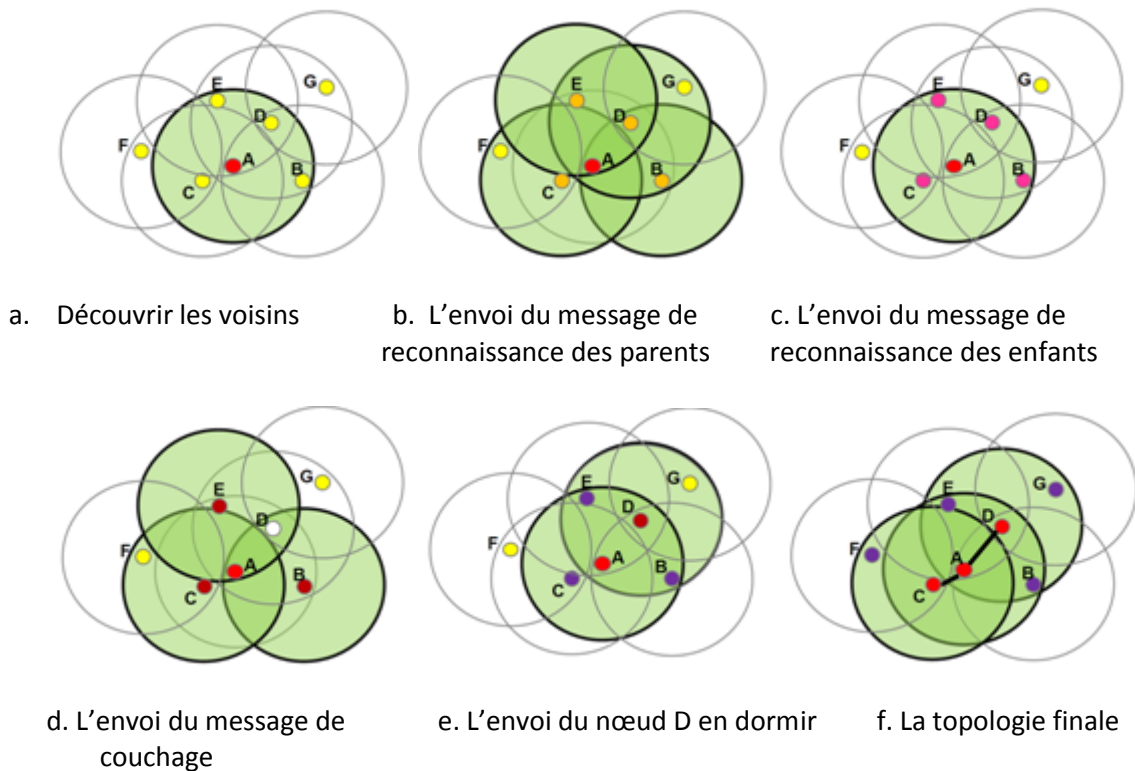


Figure III.4: Les étapes du protocole A3

➤ **Processus de sélection des enfants:**

1. Juste après l'envoi du message «bonjour», le nœud sink fixe un délai pour attendre les réponses de ses nœuds voisins.
2. Si le nœud candidat actif n'a reçu aucune réponse, il changera son état en sommeil et il arrêtera ses composants ; c'est le cas des nœuds E et B dans la topologie finale qu'ils n'ont pas d'enfants. Sinon, il changera son état en actif.
3. Le nœud parent envoie à tous ses nœuds dominés un message «reconnaissance des enfants», qui comprend la liste complète des candidats triés en ordre décroissant en fonction de la métrique de sélection, (figure III.4 (c)).
4. Chaque nœud recevant la liste des candidats change son état au nœud candidat. Puis, il fixe un délai proportionnel à sa position dans la liste pour attendre un message de couchage.
5. Si ce nœud reçoit un message de sommeil au cours de ce délai, il s'éteint tout seul pendant un certain temps et change son état pour un candidat de couchage, ce qui signifie que l'un de ses frères est mieux qualifié pour faire partie de l'arbre. Un exemple est le cas du nœud D dans la figure III.4 (d), qui est envoyé à dormir par le nœud E, car E envoie le message de couchage en premier.
6. Un nœud candidat qui ne reçoit pas un message de couchage, change son état au candidat actif et commence le processus de découverte de voisinage.

III.3.1.b. A3 Lite: connectivité

A3Lite [63], est un autre protocole qui se base sur A3 avec beaucoup de simplicité en créant un arbre semblable ou meilleur que celui d'A3.

➤ **Processus de découverte de voisinage:**

1. Semblable à A3, un nœud présélectionné, disant le nœud x démarre le processus de création de l'arbre par l'envoi d'un message «bonjour». Chaque message «bonjour» inclut l'ID du nœud parent.
2. Quand les voisins de x reçoivent le message «bonjour», ils changent d'état pour actif en attente. Et calculent le processus de découverte du voisinage qui semble à ce d'A3.
3. Les nœuds enregistrent leurs nœuds parents. Puis, chaque nœud définit un délai A pour envoyer un message «bonjour», et un délai B d'attente fixé à la moitié de la valeur de A, pour envoyer un message «reconnaissance des parents».

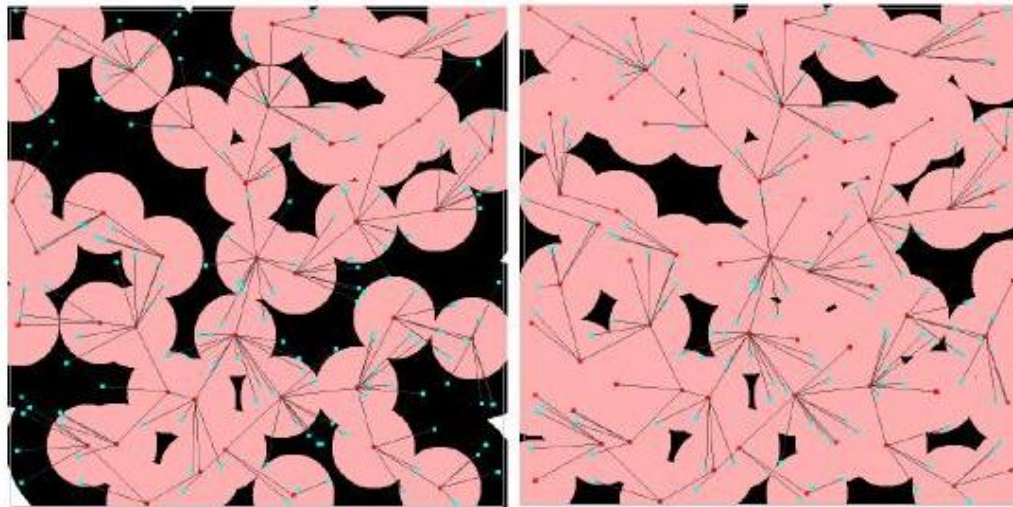
➤ **Processus de sélection des enfants:**

A ce stade, pendant l'état actif en attente plusieurs événements peuvent se produire, nous citons:

1. Le nœud peut recevoir un message «reconnaissance des parents» de l'un de ses frères (tous les nœuds sous la zone de couverture du même nœud parent). Dans ce cas, le nœud récepteur annule le délai d'attente B et reste à l'état actif en attente.
2. Le nœud peut recevoir un message «bonjour» à partir d'un nœud non frère. Dans ce cas, le nœud réinitialise le délai A et reste à l'état actif en attente, cela signifie qu'un nœud non frère a commencé un processus de découverte de voisinage.
3. Le délai A peut expirer. Lorsque cela se produit, le nœud envoie un message «bonjour» et passe à l'état candidat actif.
4. Le délai B expire. Dans ce cas, le nœud envoie un message de «reconnaissance des parents» et reste à l'état actif en attente.

III.3.1.c. A3Cov: couverture

Créant une épine dorsale pour la connectivité n'est pas suffisant pour garantir la couverture des événements. C'est pour cette raison, que le protocole A3Cov crée une épine dorsale qui fournit la connectivité au réseau, de même que fait A3. Mais en sélectionnant des nœuds supplémentaires qui prolongent la couverture de détection du réseau.



Couverture d'A3

Couverture d'A3Cov

Figure III.5: La comparaison en termes de couverture entre A3 et A3Cov.

Afin de faire ceci, A3Cov définit une nouvelle variable dans les nœuds appelée α -couverture de détection, dont son principe est: un nœud x est détecté couvert par un nœud y , si le nœud x est à l'intérieur de la portée de détection de y et ce dernier est un nœud actif ($d(x, y) < R_{\text{dét}}$).

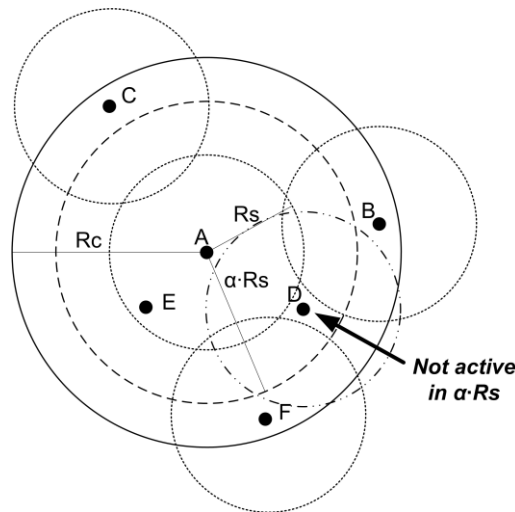


Figure III.6: Exemple de α couverture.

En résumé le protocole A3Cov fonctionne de la même façon qu'A3, mais avec des changements importants dans certaines de ses étapes [64]:

1. Pendant l'étape 3 (processus de découverte de voisinage pour A3), s'il y a un nœud qui n'a pas reçu aucun message de « reconnaissance des parents », il signifie qu'il n'y a aucun nœud qui dépend de lui pour la communication ; cependant, il peut être encore utile afin de prolonger la couverture de détection du réseau.
2. Si le nœud a été détecté couvert par n'importe quel autre nœud (compris son nœud parent), il place un délai court pour attendre les messages de détection de couverture à partir de ses voisins actifs.

3. Si le délai expire et le nœud n'est pas encore couvert, il s'allumera, et il changera son état en actif, et envoie un message de détection de couverture et un message de sommeil. Si n'importe quel nœud dans sa portée reçoit le message de couverture de détection, il évaluera s'il a été couvert par l'expéditeur, dans ce cas il mettra à jour la valeur de la variable de couverture de détection.
4. Si pendant le délai réglé dans l'étape 4 du processus de sélection des enfants d'A3, le nœud recevait un message de détection de couverture de n'importe quel autre nœud, il arrêtera le délai, en changeant son état en sommeil et tournant ses composants «off» jusqu'au prochain processus de maintenance de la topologie.

III.3.1.d. A3CovLite: couverture

A3CovLite est l'extension respectueuse d'A3Lite afin de fournir une meilleure couverture [64]. Cet algorithme fonctionne exactement comme A3Lite, sauf pour les deux moments suivants:

1. Quand un nœud reçoit le premier message bonjour et quand le délai fixé par un nœud à état candidat actif pour attendre le message de « reconnaissance des parents » est expiré. Dans le premier cas, quand un nœud non visité reçoit un message bonjour, il évalue s'il est couvert par détection par son nœud parent. Dans le second cas, le délai est réglé pour écouter les messages de reconnaissance des parents à partir des nœuds enfants.
2. Si le nœud ne reçoit pas tous les messages de ce genre, cela signifie qu'il ne fournit pas de connectivité à n'importe quel nœud. Dans A3Lite, ceci aurait été suffisant pour envoyer le nœud à l'état de veille. Toutefois, dans ce protocole, la décision dépend de savoir si le nœud a été détecté couvert par un autre nœud. Si c'est le cas, le nœud sera envoyé en mode de couchage; sinon, le nœud va changer son état à un nœud actif.
3. Chaque nœud qui atteint l'état du nœud actif envoie un message de détection de couverture à ses voisins dans le but de les informer qu'il a été choisi pour être actif.

III.3.1.e. EECDs

L'algorithme EECDs est un exemple important dans la deuxième approche de construction d'un arbre CDS qui consiste à créer des ensembles indépendants d'abord, et de les connecter par la suite.

Cet algorithme crée un MIS au cours de la première phase, après il sélectionne des nœuds de passage pour connecter les ensembles indépendants pendant la deuxième phase [66]. EECDs emploie également une approche de coloration pour construire le MIS. Au début tous les nœuds étant blancs. Un nœud d'initiateur s'élit en tant qu'élément de la coloration du MIS lui-même en noir et envoie un message noir pour annoncer ses voisins que c'est la partie de MIS. Lors de la réception de ce message, chaque voisin blanc se colorie comme gris et envoie un message gris pour informer ses propres voisins blancs qu'il a été converti en gris.

Par conséquent, tous les nœuds blancs qui reçoivent un message gris sont des voisins d'un nœud qui ne fait pas partie de MIS. Ces nœuds doivent se faire de la concurrence pour devenir des nœuds noirs. La concurrence se compose d'envoyer un message d'enquête à ses voisins pour savoir leur état et poids et attendre leurs réponses pour un laps de temps spécifique. Si pendant ce temps, il

ne reçoit aucun message noir dans la réponse, et il a le poids le plus élevé, il devient un nœud noir, et les débuts de processus se recommence. Sinon, il reste comme un nœud blanc.

Le poids est une métrique calculée par chaque nœud basé sur la capacité de la batterie et le degré effectif du nœud. L'ensemble des nœuds noirs produit par l'algorithme ci-dessus forme un MIS. Le but de la deuxième partie de l'algorithme est de former un CDS, dont les nœuds MIS envoient des messages «invite» aux nœuds non-MIS pour les inviter à devenir des connecteurs.

En réponse aux messages d'invitation, les nœuds non-MIS calculent leurs poids et envoient des messages de mise à jour. Et comme conclusion, le nœud non-MIS avec le poids le plus élevé devient une partie du CDS et envoie un message bleu pour notifier ses voisins.

III.3.1.f. CDS-règle-k

Dans l'approche à base d'élagage, les algorithmes calculent une topologie qui garantit la connectivité en incluant la plupart des nœuds, puis en taillant les nœuds inutiles dehors.

Cette approche est employée dans CDS sous l'algorithme règle k [67]. L'algorithme fonctionne en deux phases. La première phase consiste à créer un arbre CDS initial en utilisant un processus de marquage.

Dans cette première phase, les nœuds échangent des messages « Bonjour » pour prendre à connaître leurs voisins, et échanger leurs listes de voisinage. Une fois qu'un nœud reçoit les listes de ses voisins, il croise les listes avec sa propre liste de voisins et il compte le nombre d'éléments dans les deux listes. Si ce nombre est inférieur au montant des voisins de son propre liste, puis ce nœud va se marquer en tant qu'élément dans la liste initiale ; sinon, le nœud s'éteindra.

Pour la deuxième phase, l'algorithme élague les nœuds inutiles en appliquant l'une des trois règles d'élagage: Un nœud temporairement marqué décide de se démarquer s'il détermine que tous ses voisins sont couverts par des nœuds marqués avec une priorité plus élevée. Puis il annonce un message, pour que les autres nœuds puissent mettre à jour leurs états de voisins couverts.

Cette priorité a été définie à l'exécution du protocole en fonction du niveau de nœud dans l'arbre, dont un nœud qui a un niveau plus bas signifie une priorité plus élevée.

Si le nœud a au moins un nœud qui est couvert exclusivement par lui-même, il se marque alors comme élément permanent de l'arbre.

L'arbre final est une version taillée de l'arbre initial avec tous les nœuds redondants qui ont été couverts par d'autres nœuds qui ont une priorité similaire ou plus élevée.

III.3.2. Maintenance de la topologie

Dans cette section, nous allons présenter les algorithmes de maintenance de la topologie, qui consistent à prolonger la durée de vie du réseau. Et cela en exploitant l'énergie réservée dans la phase de construction de la topologie lorsque les nœuds commencent à échouer [68]:

III.3.2.a. SGTTRot:

L'algorithme SGTTRot de maintenance de la topologie est un protocole de technique statique basé sur le temps de rotation de la topologie. A chaque intervalle de temps prédéterminé, cet algorithme tourne la topologie réduite active pour une topologie des topologies pré-planifiées [69].

III.3.2.b. SGETRot:

L'algorithme SGETRot de maintenance de la topologie est un protocole de technique statique basé sur l'énergie de rotation de la topologie. A chaque fois qu'un nœud atteint un seuil d'énergie critique, ce protocole tourne la topologie réduite active pour une topologie des topologies pré-planifiées [69].

III.3.2.c. DGTTRec:

L'algorithme DGTTRec de maintenance de la topologie est un protocole de technique dynamique basé sur le temps de recréation de la topologie. A chaque intervalle de temps prédéterminé, ce protocole termine la topologie réduite précédente et invoque l'algorithme de construction de la topologie de créer une nouvelle topologie [70].

III.3.2.d. DGETRec:

L'algorithme DGETRec de maintenance de la topologie est un protocole de technique dynamique basé sur l'énergie de recréation de la topologie. A chaque fois qu'un nœud atteint un seuil d'énergie critique, ce protocole termine la topologie réduite précédente ce protocole et invoque l'algorithme de construction de la topologie de créer une nouvelle topologie [70].

III.3.2.e. DGTTRec-Rot:

L'algorithme DGTTRec-Rot de maintenance de la topologie est un protocole de technique hybride basé sur le temps de la recréation de rotation de la topologie. A chaque intervalle de temps, ce protocole tourne la topologie réduite active pour une de celles pré-planifiées. Si la nouvelle topologie pré-planifiée ne peut pas fournir le service attendu (n'a aucun lien avec le sink), le DGTTRec-Rot invoque l'algorithme de construction de la topologie de créer une nouvelle topologie réduite [71].

III.3.2.f. DGETRec-Rot:

L'algorithme DGETRec-Rot de maintenance de la topologie est un protocole de technique hybride basé sur l'énergie de la recréation de rotation de la topologie. A chaque fois qu'un nœud atteint un seuil d'énergie critique, ce protocole tourne la topologie réduite active pour une de celles pré-planifiées. Si la nouvelle topologie pré-planifiée ne peut pas fournir le service attendu (n'a aucun lien avec le sink), le DGETRec-Rot invoque l'algorithme de construction de la topologie de créer une nouvelle topologie réduite [71].

III.3.3. Comparaison entre les protocoles**➤ Construction de la topologie**

Les protocoles de construction de la topologie présentés en ce mémoire, sont les protocoles: A3, A3Lite, A3Cov et A3CovLite, qui emploient la technique «pousser un arbre», le protocole EECDs utilisant la technique reliant des ensembles indépendants, et l'algorithme CDS-règle-k qui se base sur les techniques d'élagage.

Tous ces algorithmes appartiennent à la catégorie **Backbone-based** des régimes hiérarchiques qui créent un ensemble connecté dominant (CDS) pour réduire la topologie. Leur avantage est que les nœuds sélectionnés pour faire partie des CDS sont suffisants pour fournir la connectivité au

réseau. Cela donne la possibilité d'éteindre tous les nœuds non sélectionnés. En outre, l'énergie conservée sera utilisée pour l'entretien futur du réseau.

Les protocoles A3Cov et A3CovLite sont mis en place comme des prolongements d'A3 et d'A3Lite respectivement. Ces protocoles utilisent la structure connectée produite par les algorithmes de construction comme point de départ, ensuite ils incluent d'autres nœuds qui s'étendent la couverture. Ces protocoles n'ont pas besoin d'informations de localisation, et travaillent aussi dans des scénarios où $R_{comm} \neq R_{dét.}$ Dont ils essaient d'augmenter la région couverte par la topologie réduite sans augmenter radicalement le nombre de nœuds actifs [63].

➤ Maintenance de la topologie

La plupart des protocoles de maintenance de la topologie font un nouvel appel à la construction de topologie lorsque les nœuds commencent à échouer. Ceci prévient que la phase de maintenance de la topologie consiste à exploiter l'énergie réservée dans la partie de construction de topologie.

Comparativement aux techniques statiques (SGTTRot, SGETRot), les techniques dynamiques (DGTTRec, DGETRec) nécessitent plus de temps et consomment plus d'énergie car elles doivent lancer le processus de construction de topologie plusieurs fois. A chaque fois que ce dernier sera exécuté, il choisit habituellement une topologie optimale et par conséquent ces protocoles produisent de meilleures topologies. En plus, pour toutes les techniques les protocoles basés sur l'énergie sont meilleures que ceux basés sur le temps.

Comparé avec ses homologues statiques et dynamiques, les protocoles hybrides de maintenance de la topologie (DGTTRec-Rot, DGETRec-Rot) présentent de meilleures performances. Cela revient à leur principe de fonctionnement qui combine les deux techniques statique et dynamique. Et qui consiste à déclencher une technique dynamique à chaque fois que la partie statique épuise l'énergie des arbres pré-planifiés. Et donc trouver de nouveaux arbres, et gérer le réseau plus longtemps dans le but de prolonger sa durée de vie.

III.4. Conclusion

Malgré les progrès technologiques perçus dans le domaine des RCSFs, l'énergie reste encore une ressource critique puisque chaque nœud est alimenté par une source d'énergie limitée.

Dans le but d'atteindre le défi de conservation d'énergie des nœuds au niveau communication, les RCSFs intègrent deux techniques: routage et contrôle de la topologie.

Dans ce chapitre nous avons mis l'accent sur les principaux protocoles de routage et de contrôle de topologie en étudiant leurs caractéristiques et principes de fonctionnement.

Dans le prochain chapitre, nous allons simuler ces différents protocoles, tout en faisant une étude comparative entre les protocoles de chaque technique, pour pouvoir déduire parmi lesquels est le moins consommant en terme d'énergie ou qui a de bons performances par rapport à ses homologues.

IV.1. But

Le but de ce chapitre est de simuler les mécanismes de minimisation de la consommation d'énergie au niveau de l'unité de communication. Nous exploitons pour cela les deux techniques de base: les mécanismes du contrôle de topologie et les protocoles de routage.

Ce chapitre est composé de deux grandes parties, la première partie regroupe les résultats obtenus de la simulation concernant les protocoles de routage par NS2 et la seconde présente la simulation des mécanismes de contrôle de topologie par Atarraya.

IV.2. Simulation par NS2: Partie routage

IV.2.1. Présentation du simulateur NS2

Notre choix s'est porté sur le simulateur du réseau NS2 et ceci pour les raisons suivantes:

- ❖ NS2 est un outil de recherche très utile pour le design et la compréhension des protocoles. Il sert aussi bien dans l'étude des protocoles de routage qu'à l'étude des réseaux mobiles ou les communications par satellites. Il permet à l'utilisateur de définir un réseau et de simuler les communications entre les nœuds.
- ❖ Son modèle libre permet l'ajout rapide de modèles correspondant à des technologies émergentes. Ceci le rend le meilleur simulateur par événements discrets d'après les spécialistes de télécommunications.

NS2 est aujourd'hui le simulateur de réseau probablement le plus utilisé par la communauté scientifique des réseaux. Il s'agit d'un simulateur à événement discret, fruit de la collaboration entre l'université Berkeley USC et le centre de recherche Xerox PARC dans le cadre du projet VINT2. Ce projet est soutenu par le DARPA3.

NS2 est un outil logiciel de simulation des réseaux informatiques. Il permet la simulation d'un grand nombre de réseaux locaux et/ou étendus, filaires ou sans fil. C'est un moyen de modélisation qui est à la portée de tous le monde (open source). Le logiciel est exécutable sous Linux et Windows.

Le simulateur utilise le langage orienté objet TCL pour la description des conditions de simulation sous forme d'un script. Dans le script l'utilisateur fournit la topologie du réseau, les caractéristiques des liens physiques, les protocoles utilisés, le type de trafic généré par les sources, les événements, etc.

NS2 est en réalité un programme relativement complexe écrit en C++ et interfacé via TCL. Pour modifier le comportement d'objets existants, il est donc nécessaire de modifier le code C++ qui en réalise l'implémentation.

❖ Fichier trace

Le fichier trace est un fichier qui retrace tous les événements qui sont produits pendant la simulation à savoir émission, réception, énergies des nœuds, temps simulé auquel chaque événement est arrivé.

❖ Modèle de réseau sous NS

Un modèle de réseau sous NS est constitué:

- Nœuds du réseau: présente les différentes machines du réseau, qui peuvent être des postes fixes, des routeurs, des nœuds mobiles, etc.
- Liens: représentent les connexions entre les nœuds. Ils modélisent le système de transmission. On peut régler le type de lien d'être filaire ou sans fil.
- Agents : rattachés à des nœuds ils définissent les producteurs et les consommateurs de paquets IP comme par exemple : TCP, UDP.
- Applications : rattachées à un agent producteur, elles permettent de générer du trafic comme par exemple: CBR.
- Les paramètres d'énergie : ils permettent de donner l'énergie initiale des nœuds et de définir les énergies d'émission, de réception, d'écoute et de sommeil.

IV.2.2. Résultats de simulation

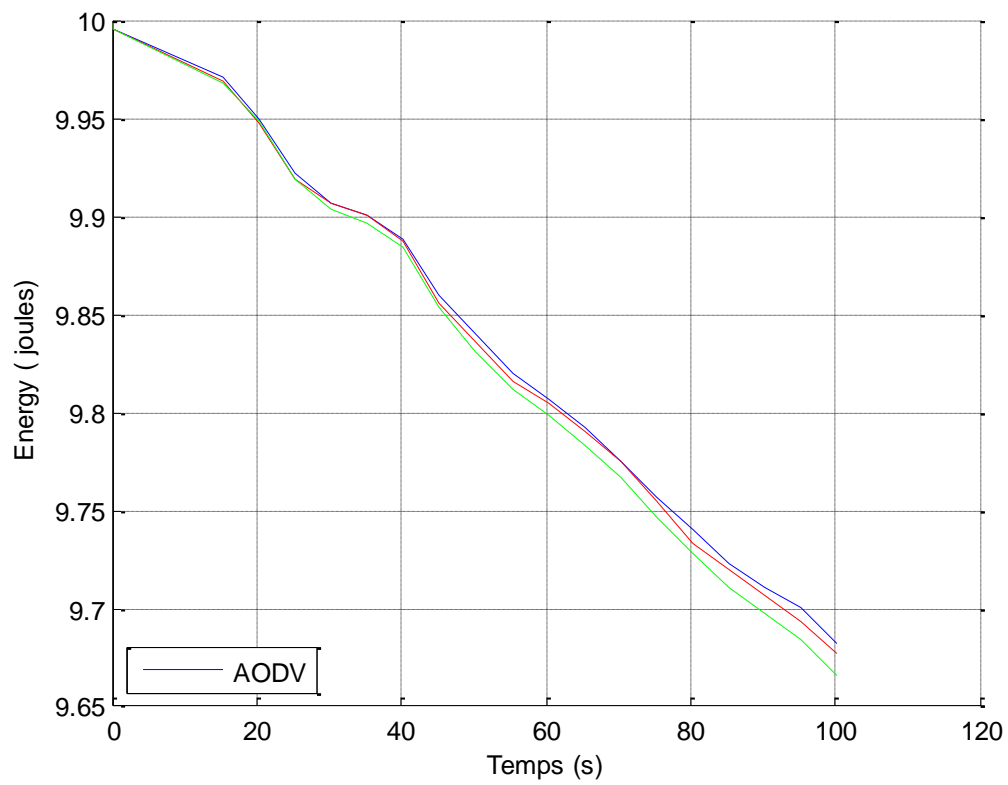
Le tableau ci-dessous (**Tableau IV.1**) représente les paramètres de simulation d'énergie sous NS 2 tout en variant les différents protocoles de routage.

Paramètre	Valeur
channel	Channel/WirelessChannel
MAc	Mac/SMAC
Antenne	Antenna/OmniAntenna
Nombre de nœuds	3/10
Protocoles de routage	AODV/AOMDV/DSDV/DSR
Temps de simulations	100
Surface	500x500
Modèle d'énergie	EnergyModel
Energie initiale (joules)	10
Energie d'émission (joules)	0.6e-1
Energie de réception (joules)	0.67e-1
Energie d'écoute (joules)	9.2e-4
Energie de sommeil (joules)	3e-7

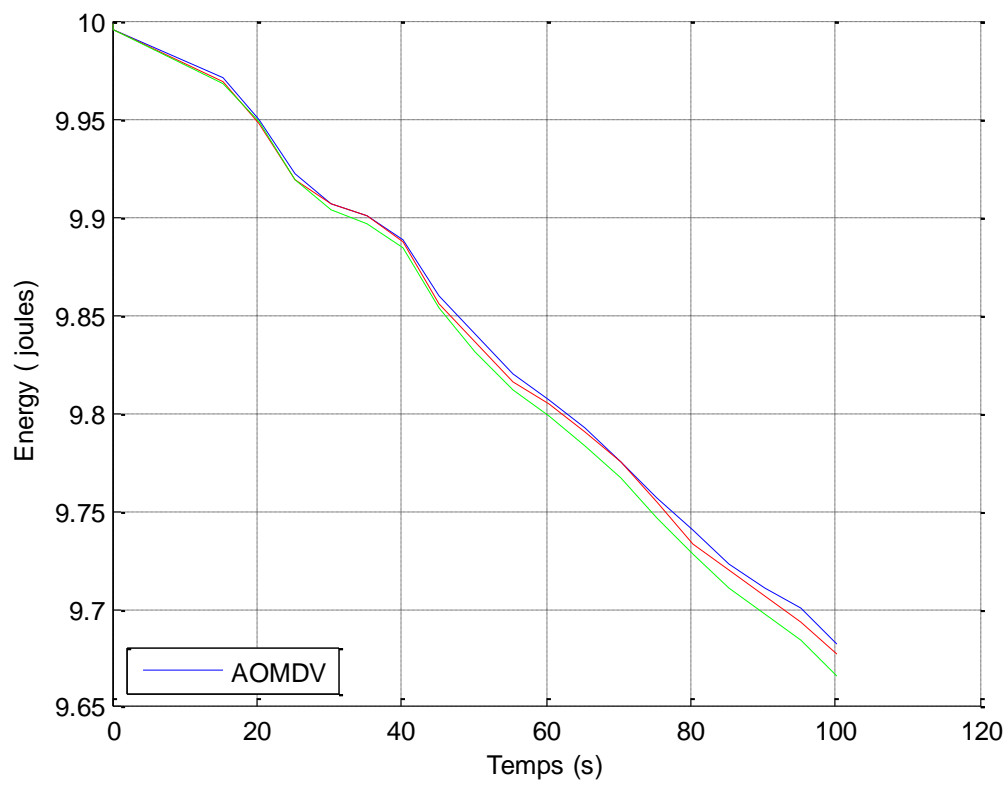
Tableau IV.1: Les paramètres de la simulation d'énergie de chaque protocole.

a) 1^{er} scénario: 3 nœuds

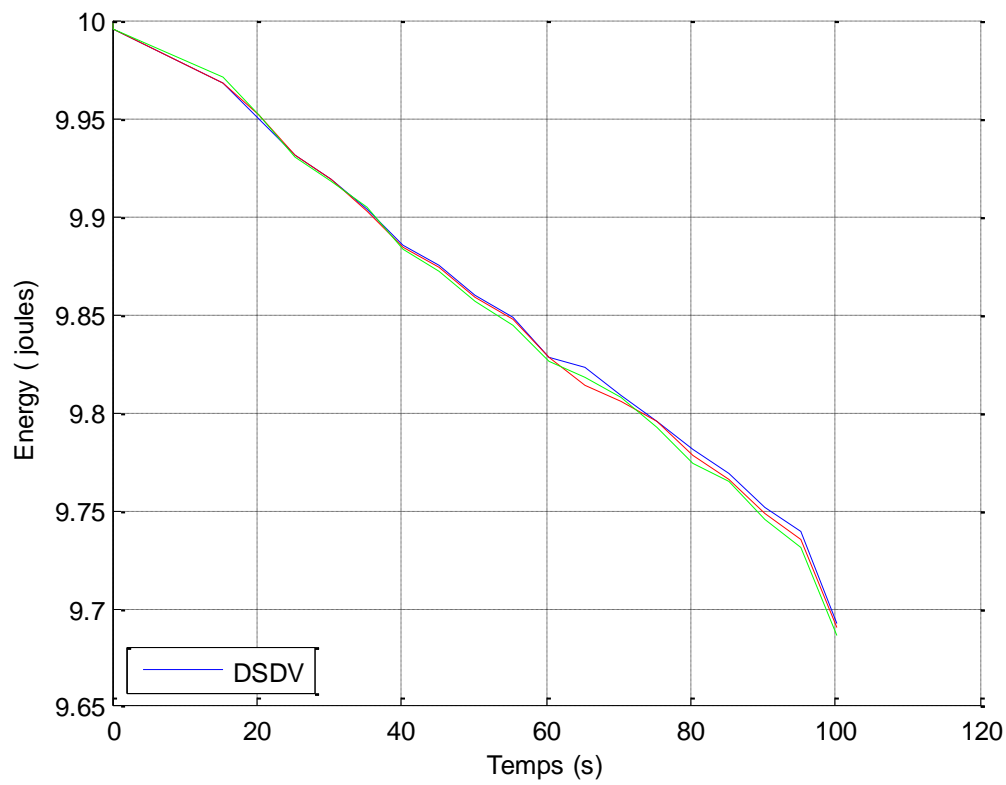
Dans un premier temps nous fixons le nombre de nœuds à trois et nous traçons l'énergie de chaque nœud en fonction du temps: (Figure IV.1)



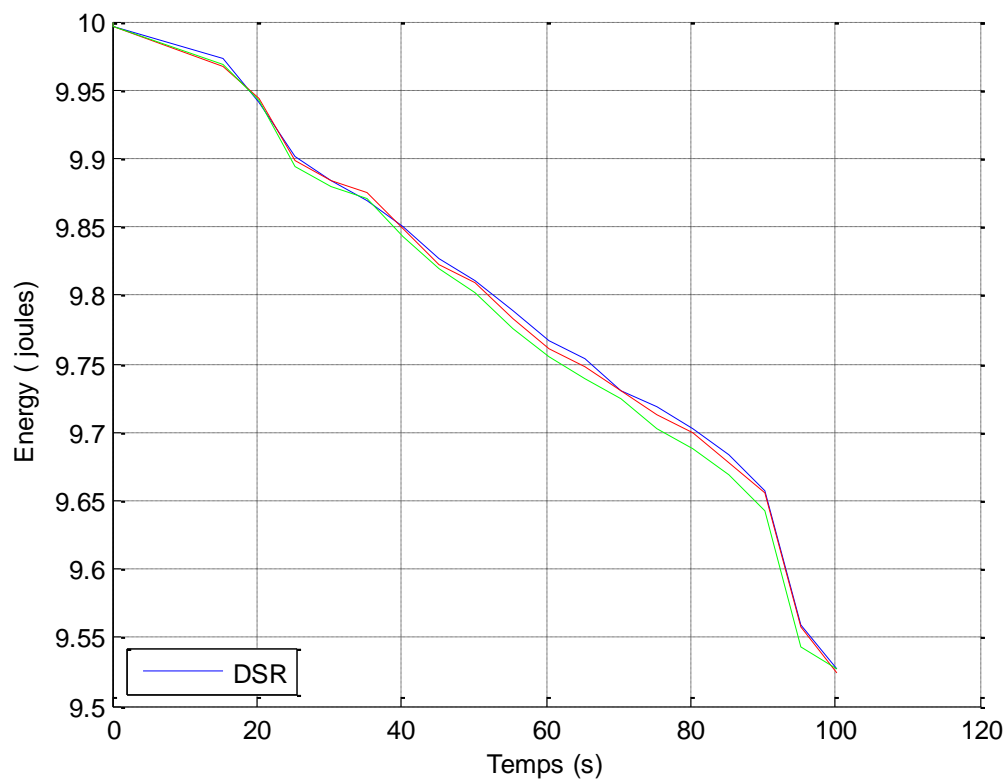
(a) Simulation d'énergie pour le protocole AODV.



(b) Simulation d'énergie pour le protocole AOMDV.



(c) Simulation d'énergie pour le protocole DSDV.



(d) Simulation d'énergie pour le protocole DSR.

Figure IV.1: Energie des 3 nœuds du réseau en fonction du temps.

La consommation d'énergie est principalement due à la transmission et la réception des paquets de données, ce qui inclut les paquets de mise à jour. La figure (IV.1) montre la diminution de cette énergie avec différents protocoles dans (a), (b), (c) et (d).

Dans la deuxième étape de ce scénario, nous avons calculé l'énergie moyenne de réseau (3 nœuds) avec les quatre protocoles de routage.

La troisième étape permet de comparer l'énergie des différents protocoles en fonction du temps en prenant en considération nos paramètres de simulation cités sur le tableau IV.1 afin de sélectionner le plus efficace et le mieux adapté aux caractéristiques de ce réseau.

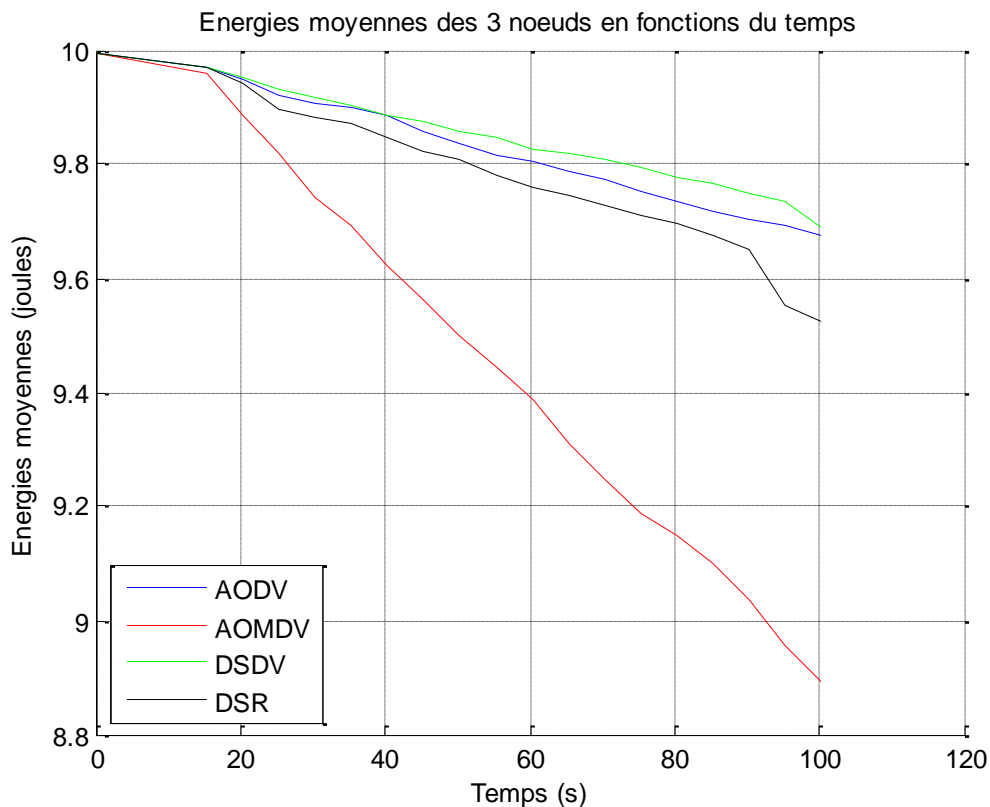


Figure IV.2: Comparaison : Energies moyennes du réseau en fonction du temps des quatre protocoles

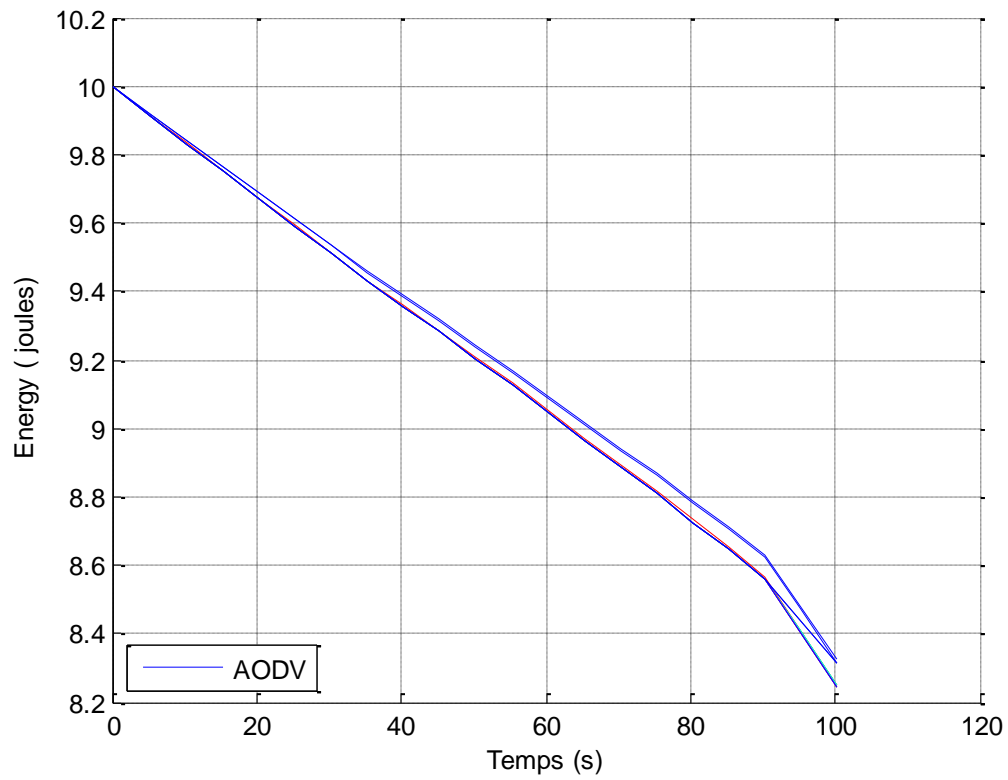
➤ Interprétation des résultats

L'énergie du réseau nécessairement décroît avec la diminution de l'énergie de chaque nœud au fil du temps pour les quatre protocoles.

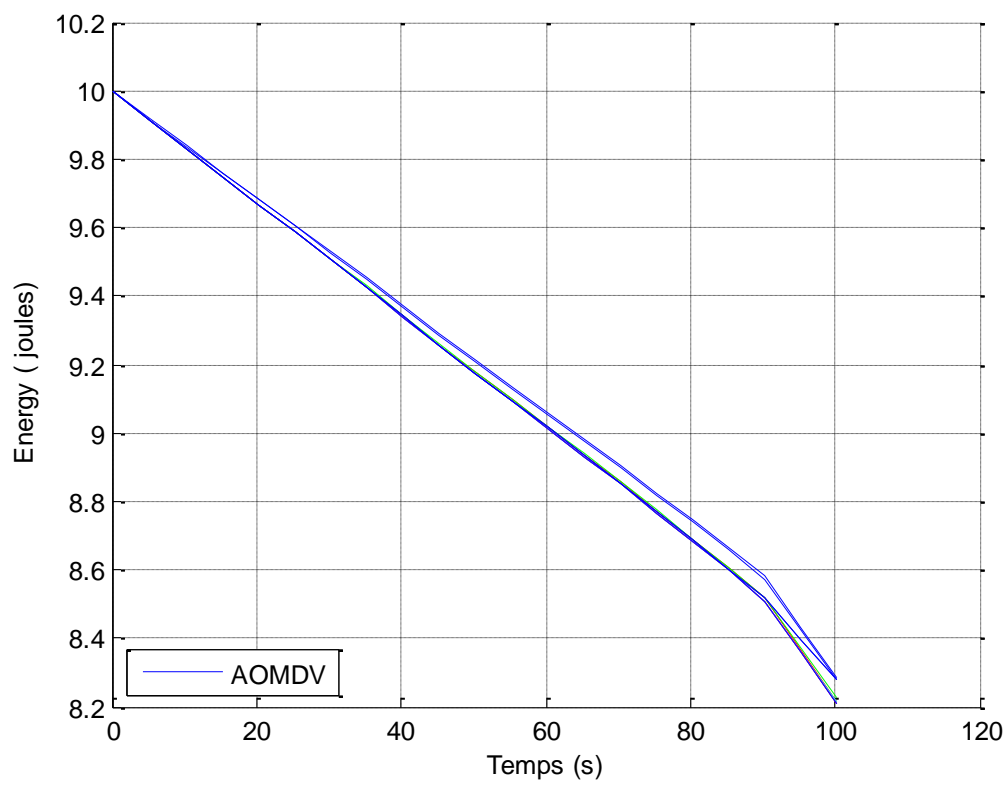
D'après la figure (IV.2) nous remarquons que le protocole DSDV a consommé moins d'énergie que les autres protocoles. Même que AODV et DSR consomme un peu plus d'énergie que DSDV. Par contre, AOMDV représente une forte consommation.

b) 2^{ème} scénario: 10 nœuds

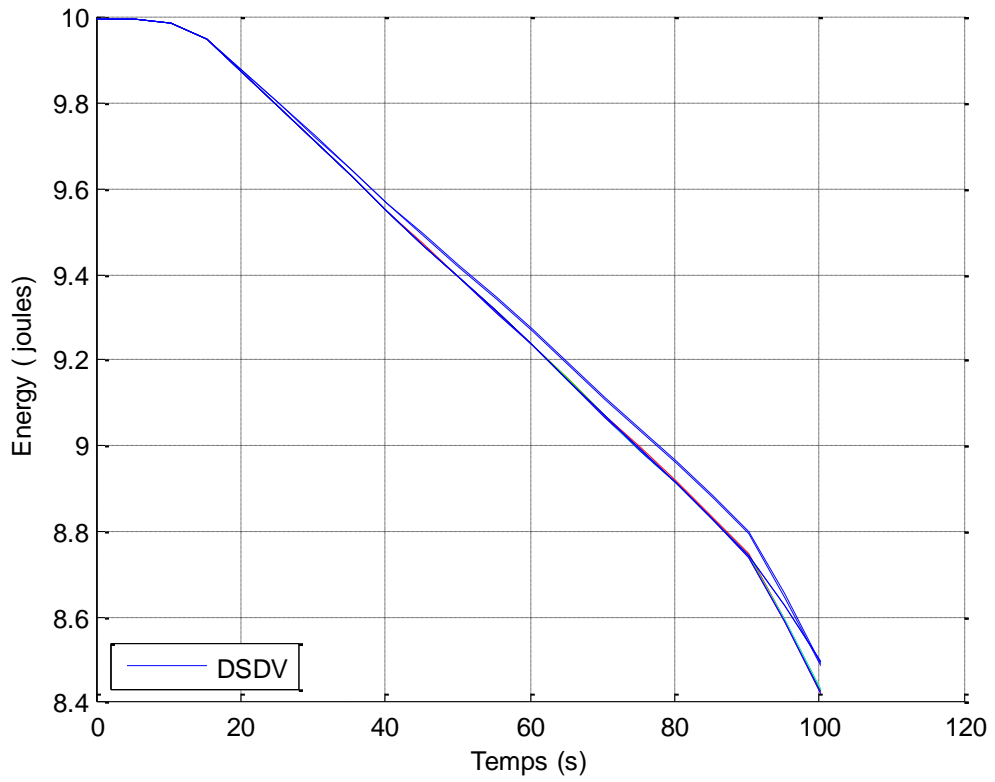
Le deuxième scénario simule un réseau de dix nœuds. Il s'agit d'étudier l'énergie de chaque nœud en fonction du temps (Figure IV.3).



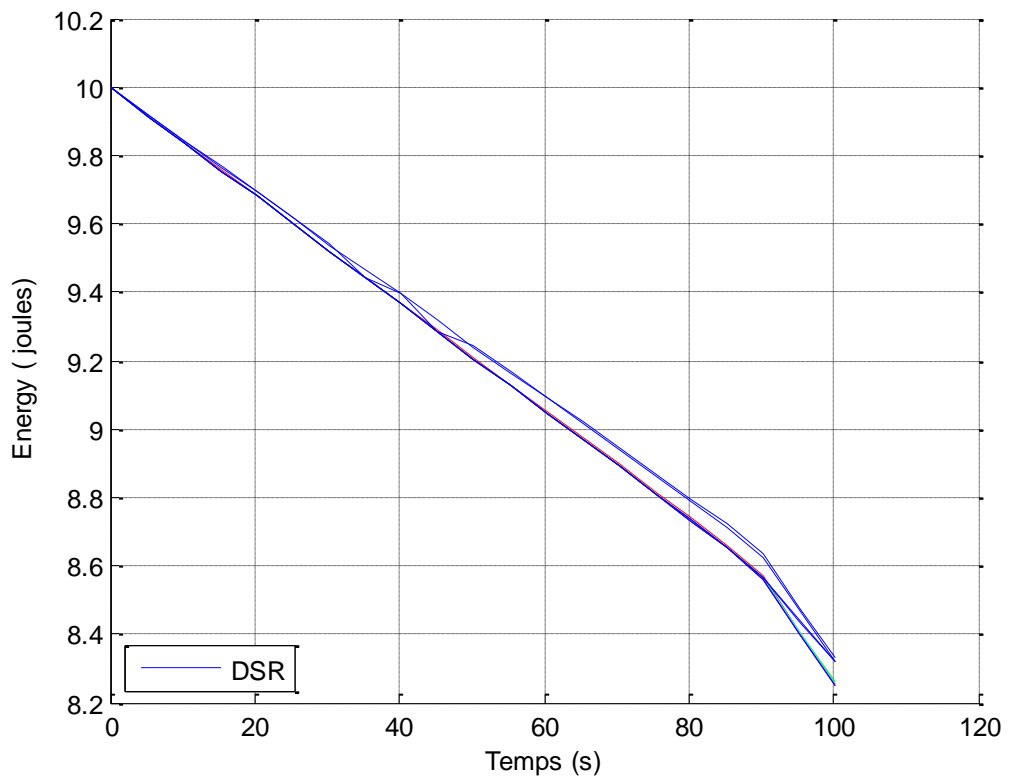
(a) Simulation d'énergie pour le protocole AODV.



(b) Simulation d'énergie pour le protocole AOMDV.



(c) Simulation d'énergie pour le protocole DSDV.



(d) Simulation d'énergie pour le protocole DSR.

Figure IV.3 : Energie des 10 nœuds du réseau en fonction du temps.

➤ Interprétation des résultats

Les courbes obtenues sur la figure IV.3, nous montre que l'énergie de chaque nœud est inversement proportionnelle au temps de simulation d'où les nœuds se déchargent au cours du temps pour tous les protocoles.

Ensuite et de la même façon que précédemment, nous avons calculé l'énergie moyenne du réseau (10 nœuds) en fonction du temps simulé des quatre protocoles de routage avec les même paramètres pour que nous pouvant faire une comparaison afin de sélectionner le plus efficace et le mieux adapté pour ce réseau (tableau IV.1).

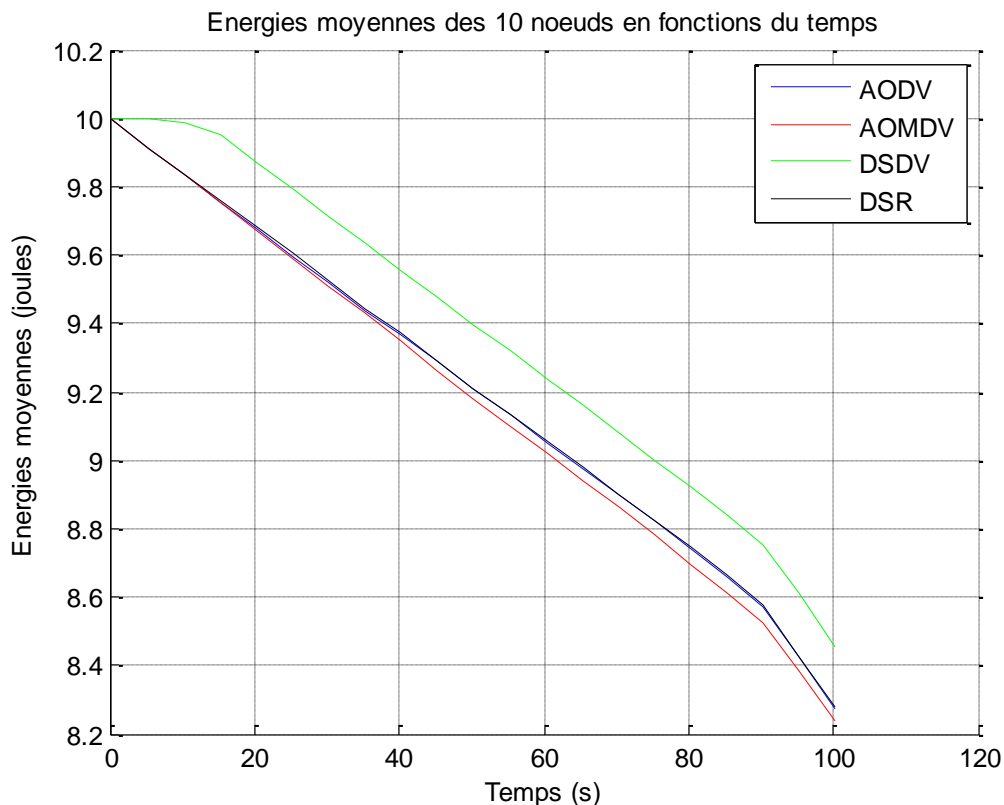


Figure IV.4: Comparaison : Energie du réseau en fonction du temps des quatre protocoles.

D'après les résultats de la figure IV.4, nous concluons que le protocole DSDV est toujours le meilleur en termes de la consommation d'énergie par rapport aux autres. Etant donné que le niveau d'énergie moyenne du réseau à un temps $t=40$ s est de 9.55 joules pour DSDV, 9.37 pour DSR, 9.36 pour AODV et de 9.34 pour AOMDV, ce qui justifie nos résultats.

Les résultats des simulations des deux scénarios indiquent que toute fois que nous changeons la taille du réseau, l'énergie se diminue successivement dans un moyen de temps.

NB

- Les performances des protocoles de routage changent non seulement avec la taille et la densité du réseau mais aussi selon les paramètres de simulations fixés par l'utilisateur final.
- Nous avons récupéré les résultats de simulation du fichier trace (fichier.tr) et nous avons tracé les courbes à l'aide du logiciel MATLAB.
- Nous avons exploité pour notre projet la version ns-allinone-2.35.

IV.3. Simulation par Atarraya: Partie contrôle de topologie

IV.3.1. Présentation du simulateur

Atarraya est un outil de simulation développé par le département de l'informatique et d'ingénierie à l'université de Floride du Sud ; pour l'enseignement et la recherche des protocoles de contrôle de la topologie pour les RSCFs. L'idée principale de ce simulateur est de fournir un outil simple pour créer et tester des protocoles anciens et nouveaux de contrôle de la topologie.

Atarraya comprend à la fois des algorithmes de construction et de maintenance de la topologie. L'inclusion de maintenance de la topologie est un aspect unique d'Atarraya. La modularité d'Atarraya le rend très facile à créer de nouveaux protocoles.

En plus, ce simulateur est capable de maintenir certaines métriques telles que: l'énergie produite et celle consommée, c'est pour cette raison que nous avons l'utiliser dans notre étude en se basant sur la métrique d'énergie consommée afin d'atteindre notre objectif de cette partie. Pendant que le simulateur peut déployer aléatoirement des nœuds, il est un peu plus difficile de placer statiquement des nœuds dans des emplacements stratégiques.

En résumé, Atarraya présente les caractéristiques suivantes:

- Plusieurs protocoles de construction et de maintenance de la topologie.
- Génération de déploiement en utilisant différentes distributions.
- Agent de simulation pour la construction de topologie initiale basée sur un algorithme disponible (l'utilisateur peut créer et ajouter son propre).
- Générateur de message qui peut être utilisé pour tester la résistance du réseau après que l'algorithme TC a été exécuté.
- Les résultats de simulation peuvent être stockés dans un fichier CSV qui peut être ouvert dans de nombreux logiciels d'analyse mathématique. En plus, des simulations peuvent être exécutées en groupe.
- La mise en œuvre d'une version du protocole de construction topologie KNEIGH comme un exemple de protocoles TC à base Tx.
- Prise en charge des modèles de mobilité simples, et la mise en œuvre d'un modèle de mobilité basée marche aléatoire.
- Prise en charge de différents modèles énergétiques.

IV.3.2. Résultats de simulation

IV.3.2.A. Simulation I

Dans cette partie nous avons simulé les algorithmes de construction de la topologie qui bâtissent un CDS: A3, A3Lite, EECDs et CDS-règle-k en fonction de la densité du réseau.

Le tableau suivant (tableau IV.2) regroupe les paramètres caractérisant le déploiement des nœuds exploités à notre simulation:

Paramètre	Valeur
Surface de déploiement	600m × 600m
Nombre total de nœuds	10, 20, 40, 60, 80, 100,120
Distribution des nœuds	Uniforme
E_{max}	60000 mJoules
Rayon de communication	100m
Rayon de détection	20m
Protocole	A3, A3Lite, EECDs, CDS-règle-k

Tableau IV.2: Paramètres de la simulation des protocoles de construction de la topologie.

➤ **Génération de déploiement:**

La figure (IV.5) présente les informations remplis par l'utilisateur sur la fenêtre de déploiement de simulateur Aarraya pour la génération de la topologie initiale:

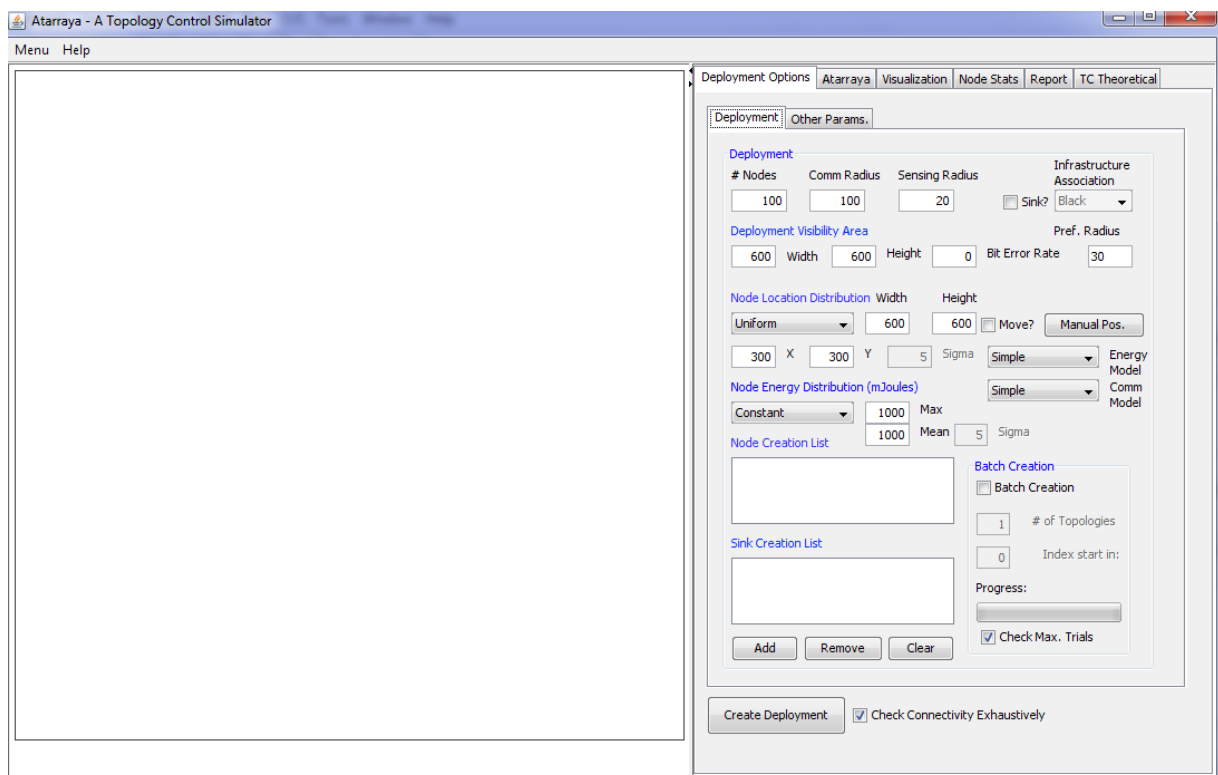


Figure IV.5: La fenêtre des options de déploiement du simulateur.

Par un simple click sur le bouton « Create deployment» nous obtenons l'interface de la figure (IV.6):

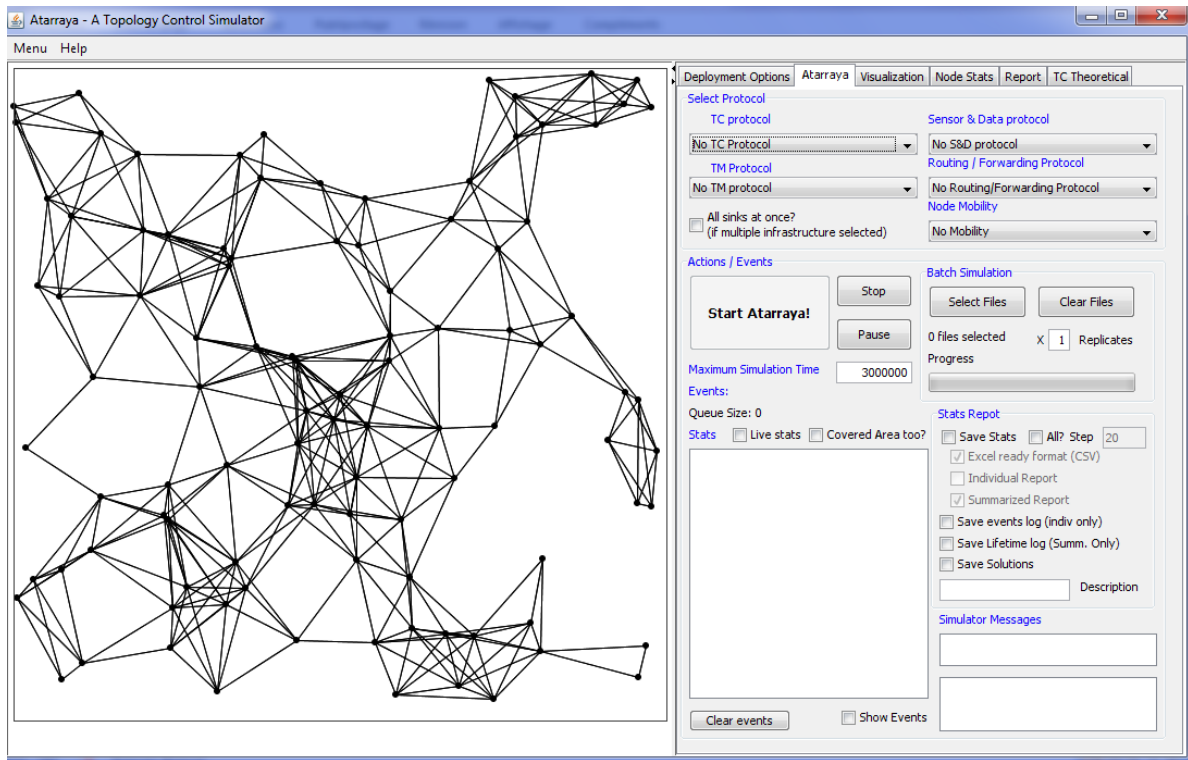
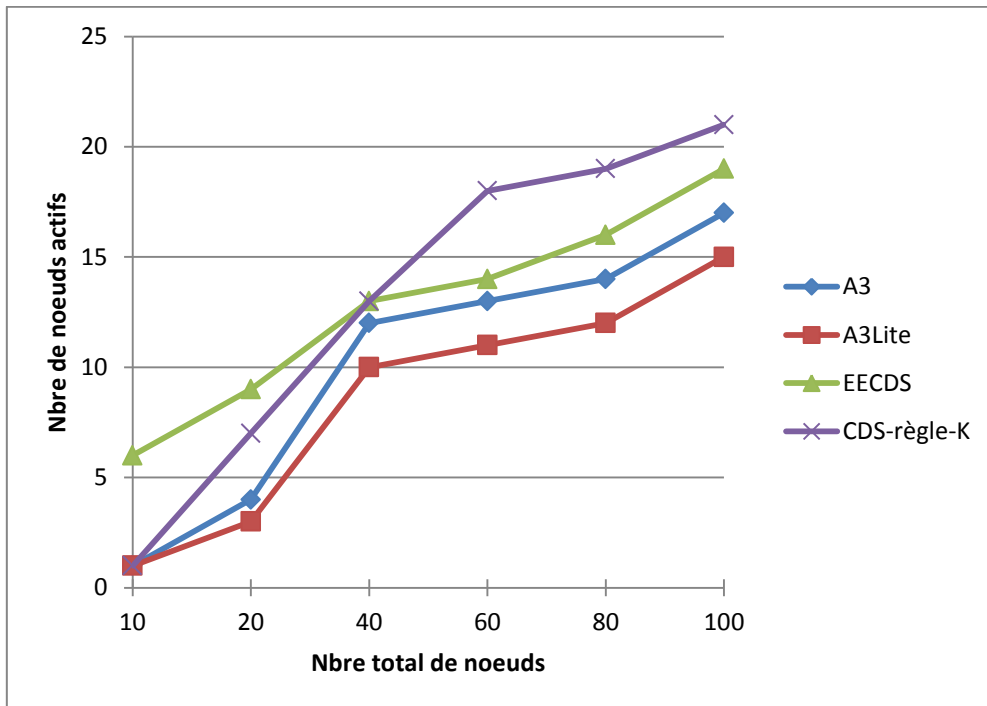
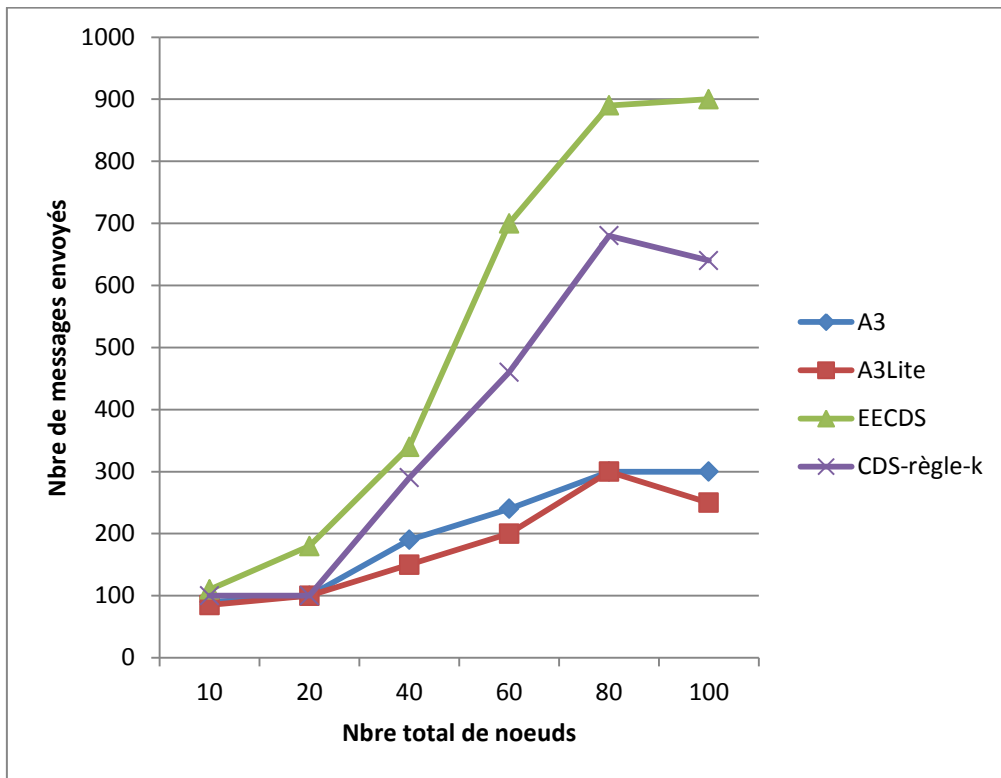


Figure IV.6: La fenêtre de l'onglette Atarraya.

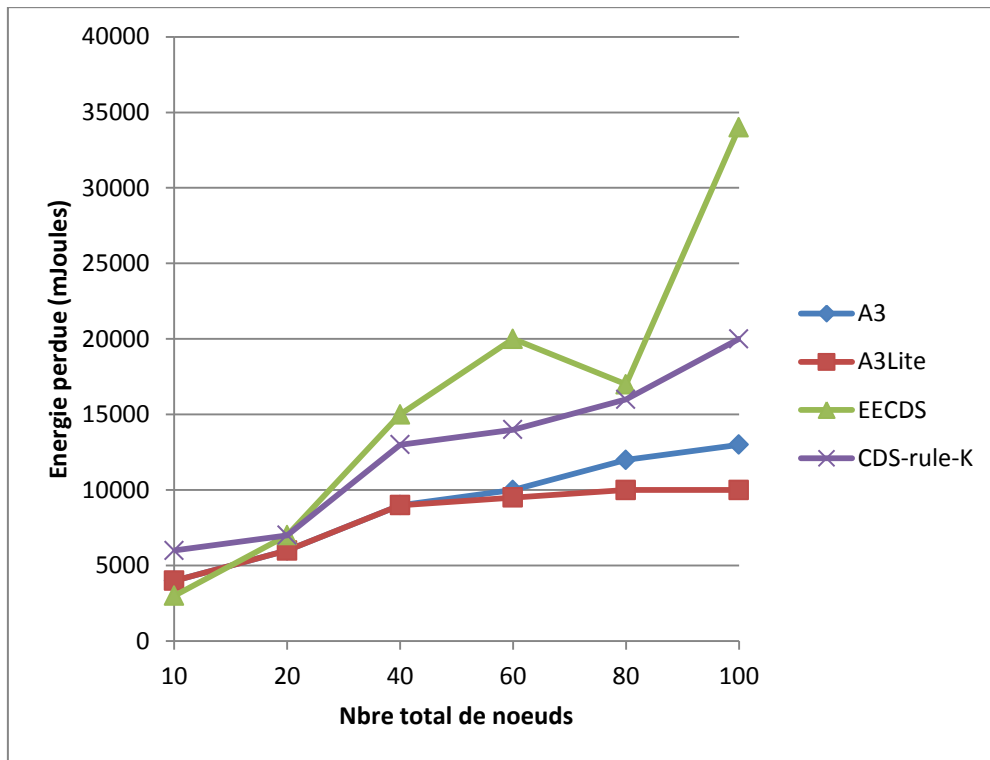
Nous avons exploité le logiciel Excel pour tracer les résultats générés par la fenêtre d'Atarraya, montrés dans la figure (IV.7):



(a) Nombre de nœuds actifs.



Figure(b): Nombre de messages envoyés.



Figure(c): Energie utilisée.

Figure IV.7: Caractéristiques des protocoles de construction.

➤ **Interprétation des résultats:**

A partir de la figure IV.7 (a), nous pouvons remarquer que malgré le changement de la densité du réseau, le protocole A3Lite produit le minimum de nœuds actifs ainsi qu'A3 (légèrement croissant par rapport à A3Lite). Le CDS-règle-K donne des résultats meilleurs dans les réseaux clairsemés jusqu'au scénario de 40 nœuds, où il commence à produire plus de nœuds actifs qu'EECDs qui apparaît d'avoir de bonnes performances en ce terme dans les réseaux denses.

En termes de complexité de message et efficacité énergétique (figures IV.7 (b) et (c)), les protocoles A3 et A3Lite montrent leurs performances supérieures avec A3Lite a le moins de messages envoyés, ce qui implique moins d'énergie utilisée dans le procédé de construction de la topologie. Le CDS-règle-K présente une augmentation presque exponentielle, ceci peut être expliqué par la taille du processus dans lequel chaque fois qu'un nœud décide de se démarquer il envoie des requêtes et reçoit des réponses individuelles. L'EECDs montre une augmentation linéaire rapide, le facteur qui augmente la quantité d'énergie est lié à la concurrence utilisée pour déterminer les meilleurs candidats pour être inclus dans les ensembles indépendants et l'arbre final. Ainsi ce processus est très coûteux en termes de messages, parce que chaque nœud doit consulter ses voisins pour leurs états afin de calculer sa propre métrique utilisée dans les deux phases de l'algorithme. Ceci prouve que les algorithmes A3 et A3Lite sont évolutifs et ne sont pas fortement affectés par le nombre de nœuds déployés ou le degré de nœuds dans le réseau.

IV.3.2.B. Simulation II

La deuxième simulation est consacrée pour une étude comparative entre les algorithmes de construction de la topologie orientés connectivité et ceux orientés couverture respectivement: A3 et A3Lite, A3Cov et A3CovLite. En basant sur les deux cas de $R_{comm} = R_{dét}$ et $R_{comm} > R_{dét}$.

Les paramètres caractérisant le déploiement des nœuds exploités à notre simulation sont résumés dans les tableaux suivants (tableau IV. 3, tableau IV. 4):

Paramètre	Valeur
Surface de déploiement	600m×600m
Nombre total de nœuds	Varié selon le type du réseau comme il est montré dans le tableau IV.3
Distribution des nœuds	Uniforme
E_{max}	1000mJoules
R_{comm} et $R_{dét}$	Variés selon le tableau IV.4
Protocole	A3, A3Lite, A3Cov, A3CovLite

Tableau IV.3: Paramètres de simulation des protocoles orientés couverture.

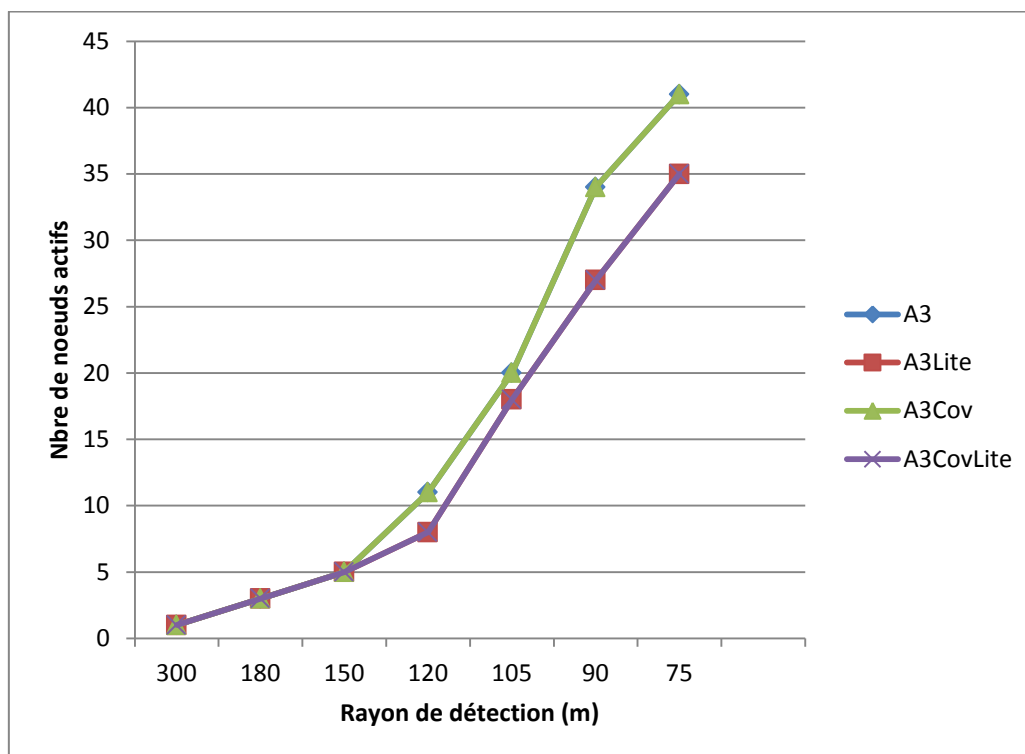
$R_{dét}$	R_{comm}	N_{dense}	$N_{clairsemé}$
300	520	21	5
180	312	83	12
150	260	129	21
120	208	225	39
105	182	311	56
90	156	445	83
75	130	683	131

Tableau IV.4: Rayons et nombres de nœuds des topologies des protocoles orientés couverture.

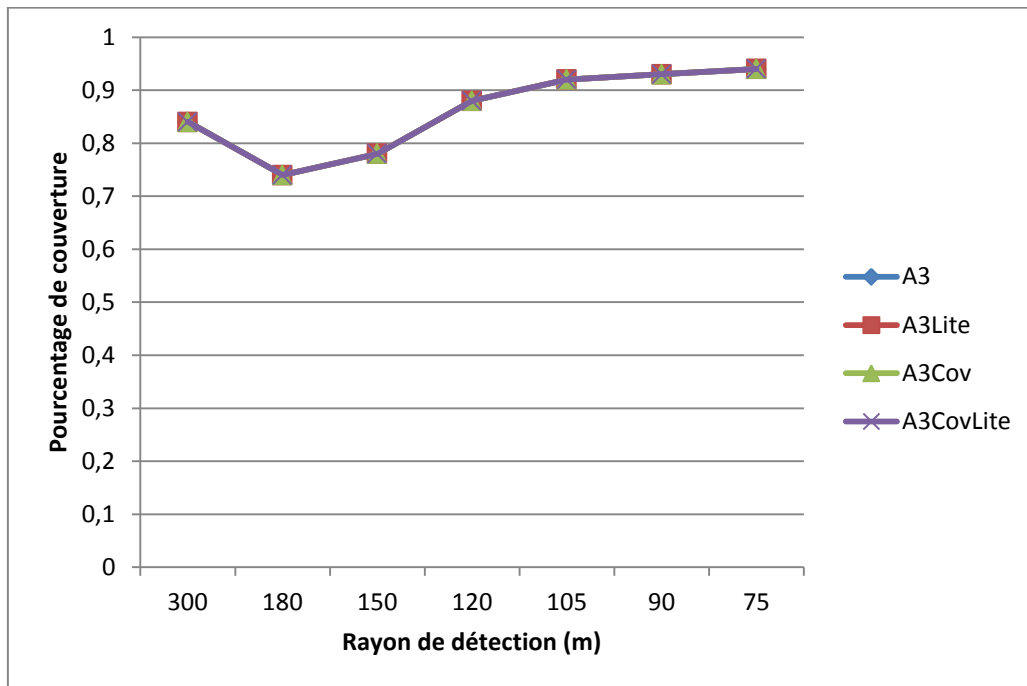
Dans cette partie, nous allons simuler les cas suivants:

a) 1^{er} cas: $R_{comm}=R_{dét}$

Dans ce cas nous supposons que $R_{comm}=R_{dét}$. Cette hypothèse crée un scénario dans lequel la connectivité et la couverture deviennent le même problème.

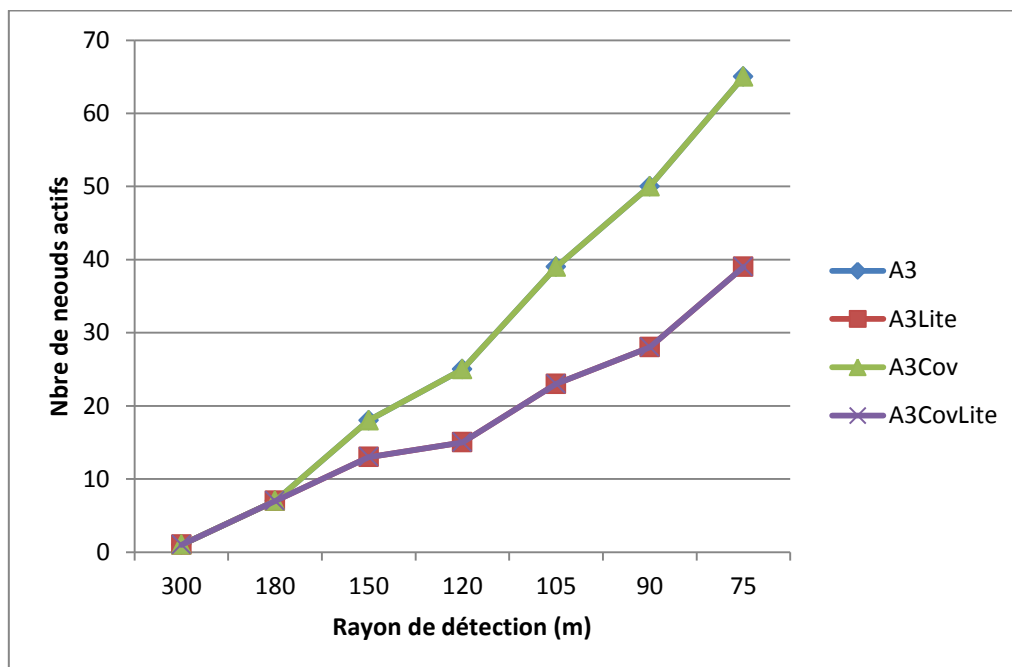


(a) Nombre de nœuds actifs dans un réseau clairsemé.

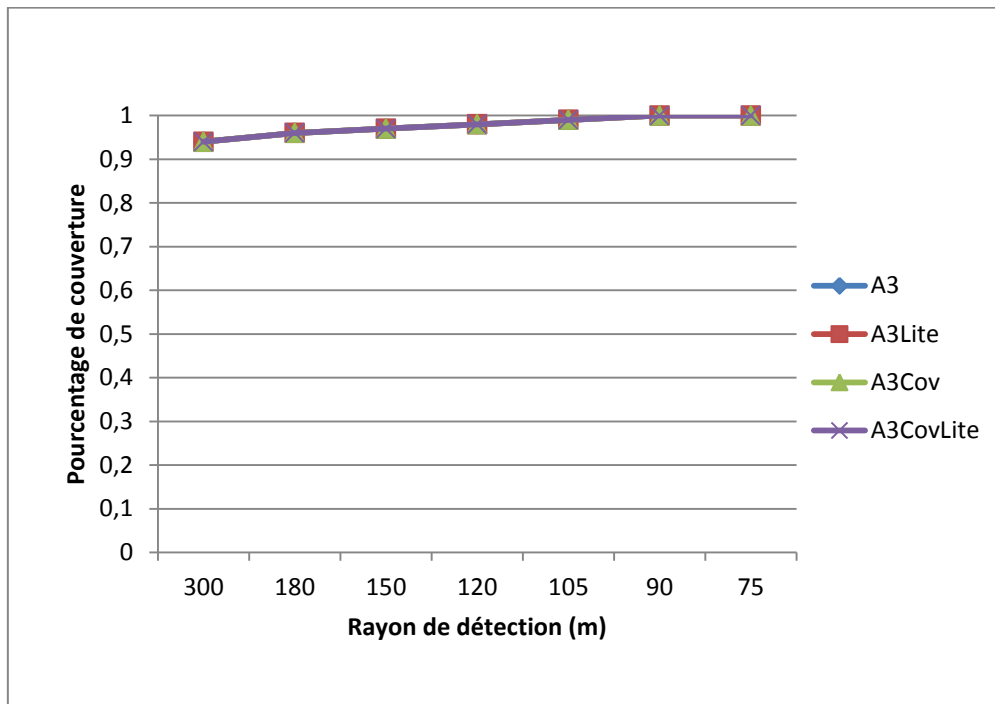


(b) Rapport de couverture dans réseau clairsemé.

Figure IV.8: Performances dans un réseau clairsemé quand $R_{comm}=R_{det}$.



(a) Nombre de nœuds actifs dans un réseau dense.



(b) Rapport de couverture dans un réseau dense.

Figure IV.9: Performances dans un réseau dense quand $R_{comm} = R_{dét}$.

➤ **Interprétation des résultats:**

A partir des résultats montrés sur les figures (IV, 8) et (IV, 9), nous remarquons que pour les deux scénarios (clairsemé et dense), les protocoles de la famille A3 montrent des comportements semblables en termes de nombre de nœuds actifs.

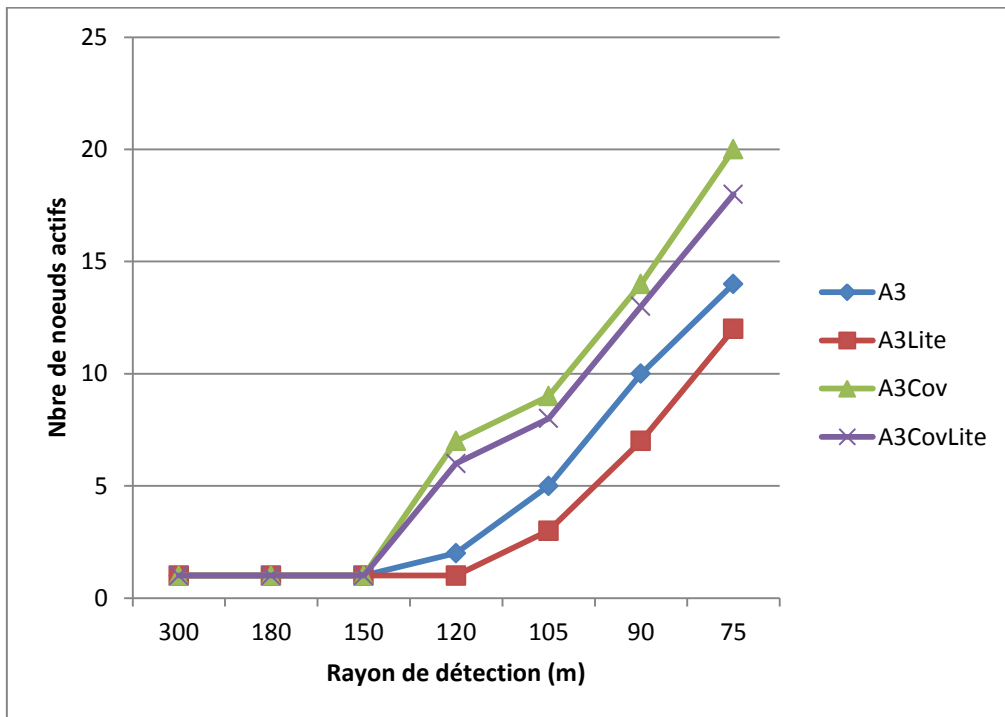
Ceci est expliqué par le fait que ces protocoles partagent les mêmes mécanismes internes, sauf pour le problème de la couverture de détection. Mais dans ce cas, ce point ne représente pas un souci, puisque un nœud couvert par la portée de communication sera également couvert par la portée de détection.

En termes de couverture, dans les deux types de réseaux les algorithmes A3 améliorent la couverture considérablement et montrent un comportement presque semblable (grande couverture).

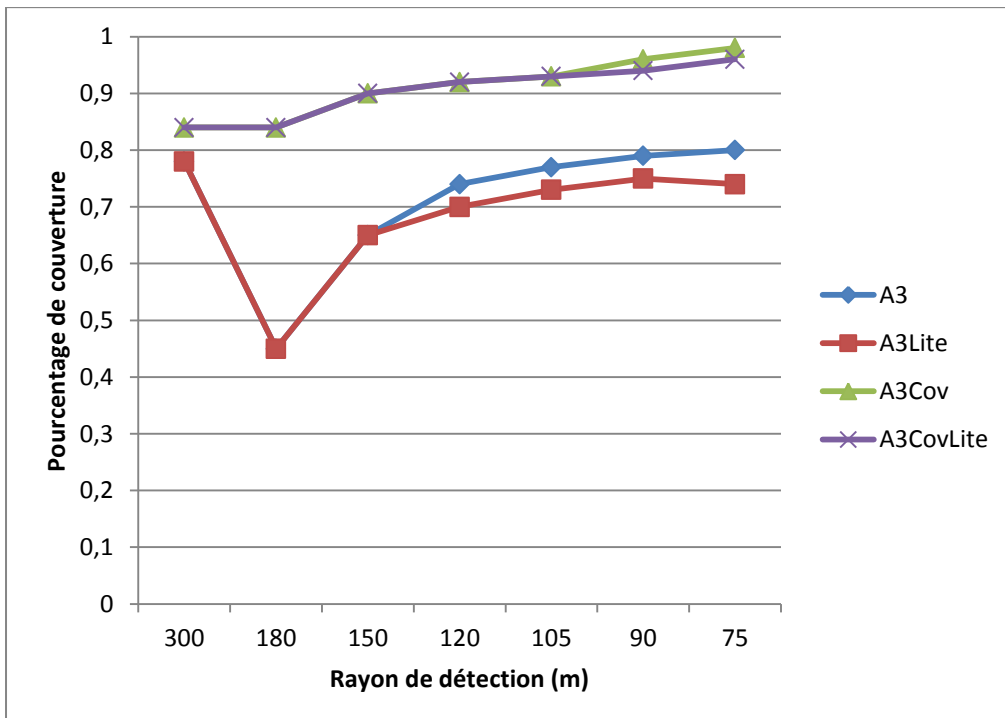
En résumé, que ce soit dans un réseau dense ou clairsemé quand $R_{comm} = R_{dét}$, l'algorithme A3Lite est le meilleur choix, parce qu'il a besoin moins de nœuds actifs que les autres protocoles, tout en fournissant la même couverture et en employant moins de messages.

b) 2^{ème} cas: $R_{comm} > R_{dét}$ et $\alpha=1$

Dans ce cas nous augmentons la valeur de R_{comm} par rapport à celle de $R_{dét}$ tout en gardant la valeur initiale du paramètre α ($\alpha = 1$):

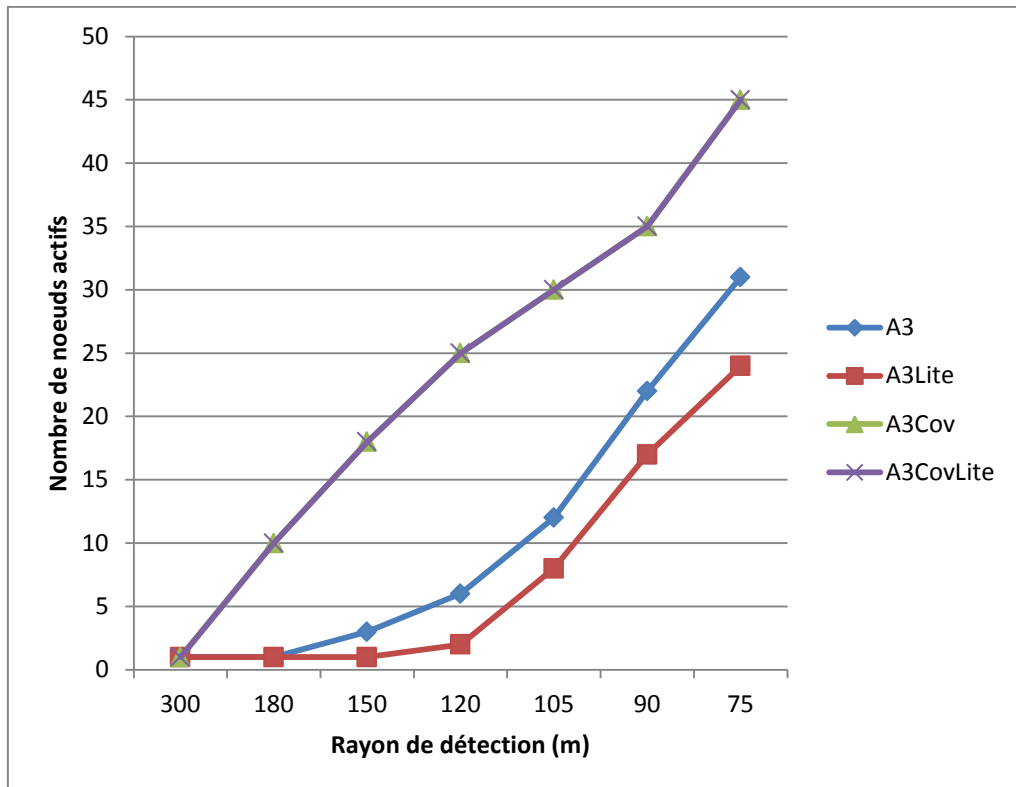


(a) Nombre de nœuds actifs dans un réseau clairsemé.

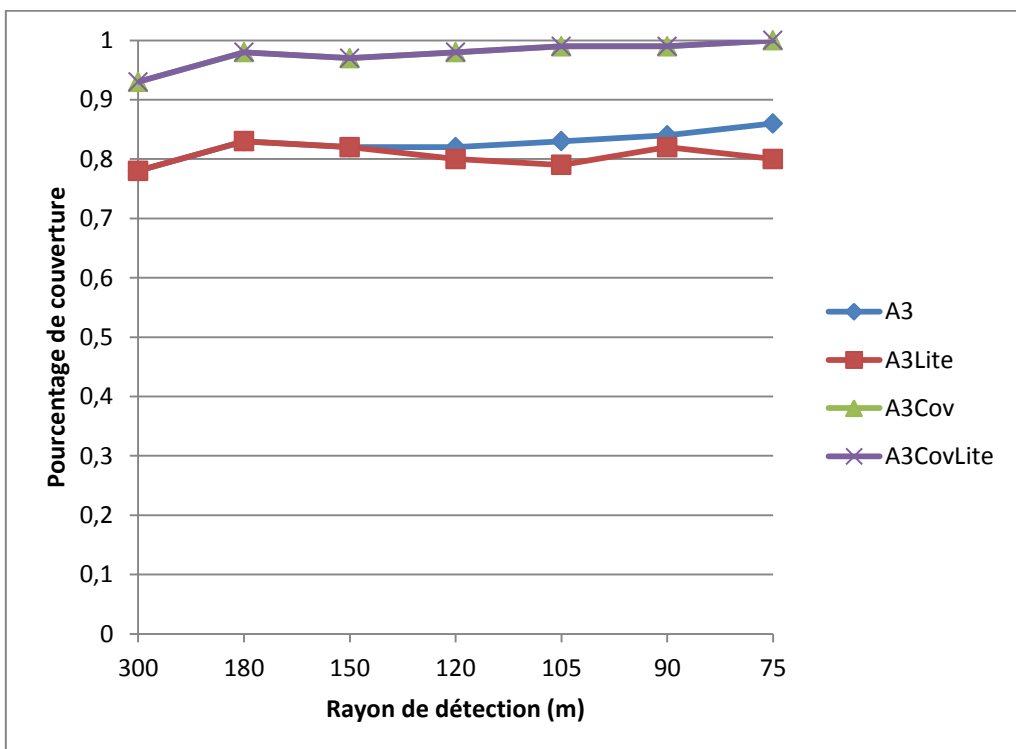


(b) Rapport de couverture dans un réseau clairsemé.

Figure IV.10: Performances dans un réseau clairsemé quand $R_{comm} > R_{dét}$.



(a) Nombre de nœuds actifs dans un réseau dense.



(b) Rapport de couverture dans un réseau dense.

Figure IV.11: Performances dans un réseau dense quand $R_{comm} > R_{dét}$.

➤ **Interprétation des résultats:**

Nous pouvons voir à partir des figures IV, 10(a) et IV, 11(a) que dans les deux scénarios (clairsemé et dense), les protocoles Cov (A3Cov et A3CovLite) utilisent plus de nœuds actifs que ceux Non-Cov pour construire une topologie réduite. Ceci est expliqué par le fait que les protocoles Non-Cov n'incluent pas tous les nœuds supplémentaires pour la couverture, parce que la connectivité est leur principal but.

Pour le cas de la couverture, les figures IV, 10(b) et IV, 11(b) montrent que les protocoles Cov s'exécutent considérablement meilleurs que ceux Non-Cov dans les deux scénarios: clairsemés et denses. Naturellement, ceci revient à la dépense d'avoir plus de nœuds actifs.

En résumé, dans le cas des systèmes où $R_{comm} \neq R_{dét}$. Si la couverture n'est pas aussi critique que l'énergie, il est préférable d'employer A3Lite, qui a besoin moins de nœuds et de messages et donc une basse énergie pour établir une topologie réduite. Par contre, si la couverture est très importante pour l'application, alors l'algorithme A3CovLite est le meilleur choix en grande partie en raison de son inférieure complexité de message.

IV.3.2.C. Simulation III

Dans cette partie nous allons voir les performances des algorithmes de maintenance de la topologie basés sur l'énergie: SGETRot, DGETRec. Afin de faire une comparaison entre ces protocoles en termes de consommation d'énergie, nous allons réaliser trois simulations dont chacune est testée pour les deux cas du réseau clairsemé et dense. Pour cela, nous employons trois protocoles de construction de la topologie: A3, EECDs et CDS-règle-K.

Les données utilisées dans les simulations sont résumées dans le tableau (IV.5):

Paramètre	Valeur	
	Clairsemé	Dense
Surface de déploiement	600m × 600m	
Nombre de nœuds	50	100
Nombre de sink	1 sink	
Distribution des nœuds	Uniforme (200,200)	
E_{max}	1000mJoules	

Tableau IV.5: Paramètres de simulations des protocoles de maintenance de la topologie.

Les résultats donnés par la suite sont tirés à partir des fichiers «lifetime» générés par le simulateur sous le format CSV:

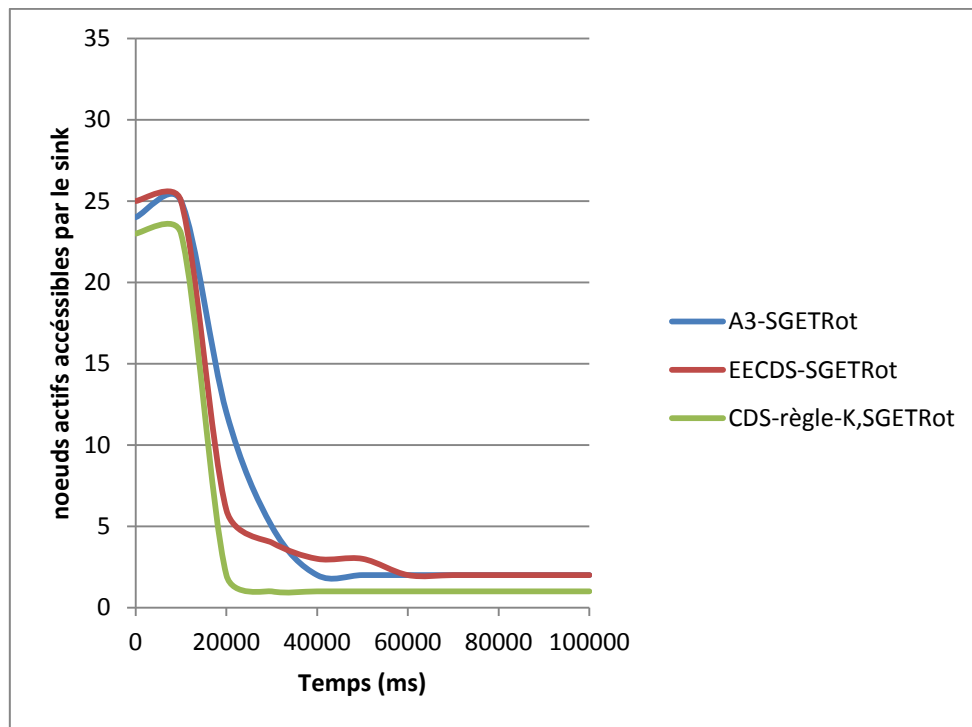


Figure IV.12: Evaluation des performances de SGETRot dans un réseau clairsemé.

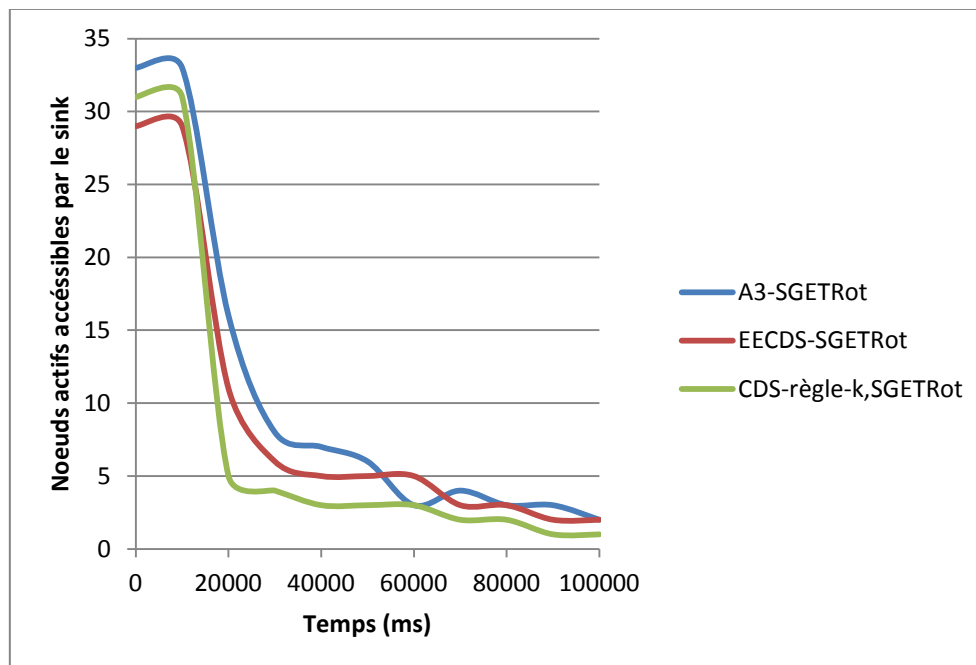


Figure IV.13: Evaluation des performances de SGETRot dans un réseau dense.

➤ **Interprétation des résultats:**

Nous observons à partir de la figure (IV.12) que dans un réseau clairsemé, parmi les trois techniques (A3-SGETRot, EECDs-SGETRot et CDS-règle-k/SGETRot), celle basée sur l'algorithme A3 est

la plus performante en termes de consommation d'énergie. C'est parce qu'A3 est plus efficace en terme d'énergie qu'EECDs et CDS-règle-k.

Pour le cas d'un réseau dense, le SGETRot augmente la possibilité d'avoir plus de topologies décousues. Cette amélioration est montrée sur la figure (IV.13) qui nous explique que malgré le progrès des techniques basées sur EECDs et CDS-règle-k par rapport au premier cas (clairsemé), celle basée sur A3 reste surpassée que les deux autres en termes de consommation d'énergie. Ceci revient à la manière de sélectionner les arbres de disjonction de chaque protocole sous-jacent de construction de topologie.

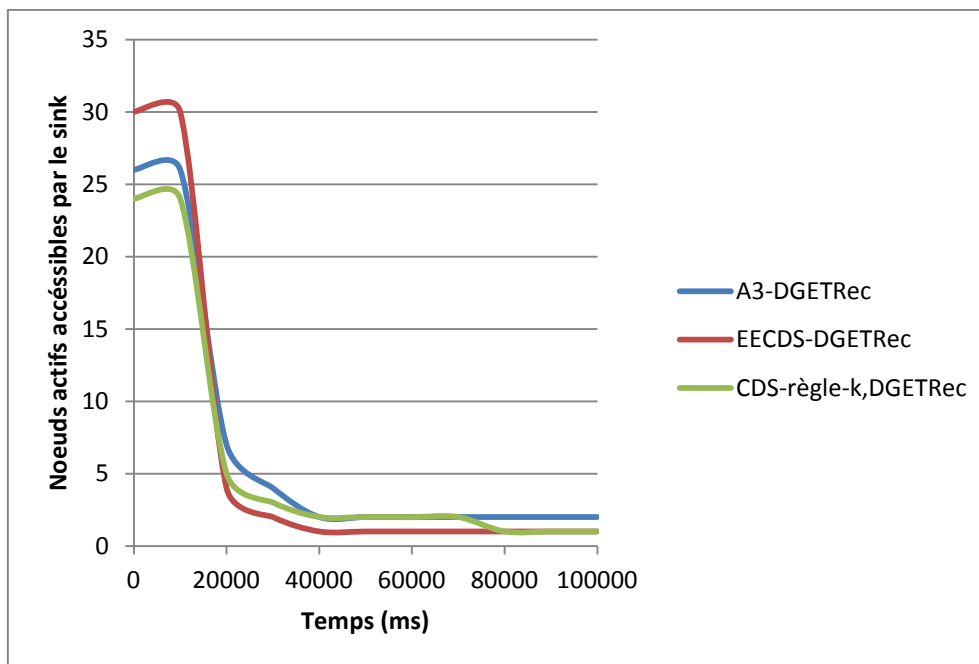


Figure IV.14: Evaluation des performances de DGETRec dans un réseau clairsemé.

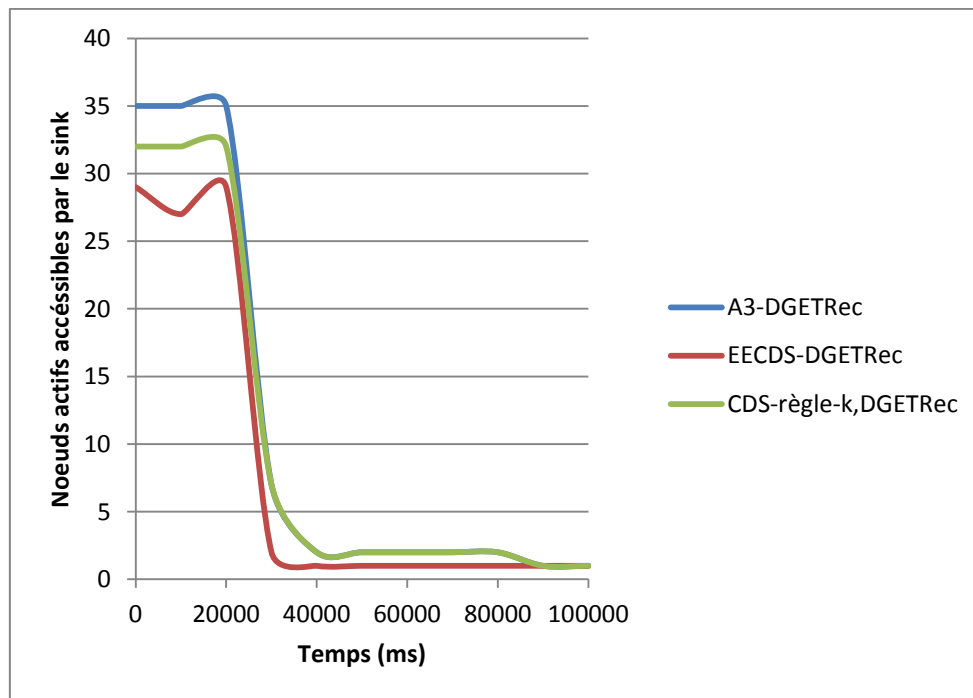


Figure IV.15: Evaluation des performances de DGETRec dans un réseau dense.

➤ **Interprétation des résultats:**

Comparant les figures (IV.12) et (IV.14), nous remarquons que dans un réseau clairsemé les techniques dynamiques fournissent plus et meilleures topologies que ceux statiques, ceci prouve que les protocoles dynamiques consomment plus d'énergie que leurs homologues statiques.

La figure (IV.15) nous montre que dans un réseau dense, la technique dynamique utilisant l'algorithme de construction de la topologie A3 (DGETRec-A3) est la plus performante, bien que pas par une marge importante. Ceci due au fait qu'A3 a une meilleure complexité de message qu'EECDs et CDS-règle-k.

IV.4. Conclusion

Dans ce chapitre, nous avons simulé les protocoles des deux techniques de minimisation de la consommation d'énergie au niveau communication: routage et contrôle de la topologie. Pour cela, nous avons utilisé respectivement les simulateurs NS2 et Atarraya.

Suite à ces simulations, nous avons comparé les différents protocoles étudiés dans le but de constater l'optimalité et l'efficacité de chaque protocole en termes de consommation d'énergie.

Conclusion générale

Les réseaux de capteurs constituent un axe de recherche très fertile et peuvent être appliqués dans plusieurs domaines différents. Cependant, il reste encore de nombreuses recherches à résoudre dans ce domaine afin de pouvoir les utiliser dans les conditions réelles.

Nous nous sommes intéressés à la problématique de l'économie d'énergie et l'amélioration de la durée de vie des réseaux de capteurs. Ce problème est causé par le fait que les réseaux de capteurs sont déployés dans des environnements inaccessibles et que leurs batteries à faible puissance ne peuvent pas être rechargées.

Dans ce contexte, nous avons exploité deux techniques efficaces pour la minimisation de la consommation de l'énergie telle que le routage et le contrôle de topologie. Il s'agit d'étudier et de simuler les performances des mécanismes de ces derniers en termes d'énergie sous deux simulateurs respectivement, NS2 et Attaraya. Les résultats obtenus de ces deux simulations nous montrent la fiabilité et l'efficacité au niveau de la conservation d'énergie dans le réseau RCSF selon le choix judicieux des approches proposées pour les deux techniques qui nous donne une meilleure optimisation.

Ce travail nous a permis d'identifier les processus clés d'un réseau de capteur sans fil et de bien saisir leurs différentes fonctionnalités de base pour pouvoir résoudre sa contrainte majeure ainsi qu'une bonne familiarisation des outils de simulation réseau.

Perspectives

L'une des préoccupations majeures qui reste à développer comme une suite logique à notre travail consiste à étendre cette étude vers des réseaux denses pour mieux évaluer le changement du comportement des approches proposées pour la partie routage.

Glossaire

AODV: *Ad Hoc On demand Distance Vector.*

AOMDV: *Ad Hoc On demand Multipath Distance Vector.*

ASIC: *Application Specific Integrated Circuit.*

CBR: *Constant Bit Rate.*

CDS: *Connected Dominating Set.*

CTR: *Critical Transmission Range.*

DARPA3: *Agence Avancée de Projet de la Défense.*

DSDV: *Dynamic destination Sequenced Distance Vector.*

DSP: *Digital Signal Processors.*

DSR: *Dynamic Source Routing.*

EECDs: *Energy Efficient Connected Dominating Set.*

FPGA: *Field Programmable Gate Array.*

GPS: *Global Positioning System.*

LEACH: *Low Energy Adaptive Clustering hierarchy.*

MAC: *Medium Acces Control.*

MANET: *Mobile Ad Hoc NETworks.*

MIS: *Maximal Indepent Set.*

MST: *Minimal Spanning Tree.*

NS: *Network Simulator.*

QoS: *Quality of Service.*

RCSF: *Réseaux de Capteurs Sans Fil.*

RREP: *Route REPlay.*

RREQ: *Route REQuest.*

SN: *Sequence Number.*

Tcl: *Tools Command Language.*

TCP: *Transmission Control Protocol.*

TinyOs: *Tiny Microthreading Operating System.*

TORA: *Temporary Ordering Routing Algorithm.*

UDP: *User Datagram Protocol.*

UWB: *Ultra Wide Based.*

WSN: *Wireless Sensor Network.*

Résumé

Les RCSF trouvent leur application dans diverses activités de la société, tels les processus industriels, les applications militaires de surveillance, l'observation et le suivi d'habitat, etc. À cause de la nature intrinsèque de leur fonctionnalité, les capteurs ont une contrainte principale: leur source d'énergie est limitée et presque jamais renouvelable.

Dans ce contexte, nous proposons l'étude et l'analyse du problème de la minimisation de la consommation d'énergie dans RCSF en exploitant deux techniques « le routage » et « le contrôle de la topologie ». Les résultats de ces deux techniques sont obtenus respectivement par deux simulateurs: *NS2* et *Atarraya*.

Mots clés: RCSF, Consommation d'énergie, Routage, Contrôle de la topologie, NS2, Atarraya.

Abstract

The WSN find their application in various activities of the society, such the industrial processes, the military monitoring applications, house automation, etc. Because of the intrinsic nature of their functionality, the sensors have a principal constraint: their energy source is limited and almost never renewable.

In this context, we propose the study and analysis of the minimization of energy consumption problem in WSN by exploiting two techniques "the routing" and "the topology control". The results of these two techniques are respectively carried out by two simulators: NS2 and Atarraya.

Keywords: Wireless sensor networks, Energy consumption, Routing, Topology control, NS2, Atarraya.

تلخيص

لقد حققت شبكات الاستشعار اللاسلكية تطبيقاتها في مختلف مجالات المجتمع مثل: المجال الصناعي، المجال العسكري، المراقبة و غيرها. بسبب طبيعة اشتغالها الجوهرية، لعقد الاستشعار ميزة خاصة: و هي أن مصدر طاقتها محدود و قريب من غير القابل للتجديد. في هذا المشروع، قمنا بطرح مشكل التقليل من الطاقة المستهلكة في شبكات الاستشعار اللاسلكية باستعمال تقنيتين "routage" و "controle de topologie". حصلت نتائج هاتين التقنيتين على الترتيب باستعمال جهازي محاكاة: "NS2" و "Atarraya".

Références & Bibliographie

- [1] F. Baudoï, «Capteurs : principes et utilisations», Édition Casteilla, Etats-Unis, 2007.
ISBN: 978-2-7135-2749-4.
- [2] E.P. Freitas, T. Heimfarth, I. F. Netto, «Enhanced Wireless Sensor Network Setup Strategy Supported by Intelligent Software Agents», Magister theses, Computer Science Department, Federal University of Lavras, Lavras, Brazil.
- [3] AKYILDIZ, I.F, W. SU, Y. SANKARASUBRAMANIAM, E.CLAYIRCI, «A Survey on Sensor Networks», Magazine communications IEEE, Vol. 40, no. 8, pp. 102-114, technology Institut, Georgia, USA, August 2002.
- [4] QINGHUAWANG, H. BALASINGHAM Ba, «Wireless Sensor Networks An introduction», Application centric design: Wireless Sensor Networks, InTech, 2010.
ISBN: 978-953-307-321-7.
- [5] L.TITZER, J.PALSBERG, «Nonintrusive Precision Instrumentation of Microcontroller Software», ACM, New York, ETATS-UNIS, Volume 40 issue 7, Pages 59-68, Juillet 2005.
- [6] H. KARL, A.WILLIG, «Protocols and architectures for Wireless Sensor Networks», Edition: John Wiley & Sons, USA, 2005.
- [7] MONTOYA, D.C, RESTREPO, D.A.OVALLE, «Artificial Intelligence for Wireless Sensor Networks Enhancement», InTech, 2010.
- [8] B.KRISHNAMACHARI, «Networking Wireless Sensors», Cambridge University press, 2005.
- [9] K. BEYDOUN, «Conception d'un protocole de routage hiérarchique pour les réseaux de capteurs», Thèse de Doctorat en Informatique de l'université de FRANCHE-COMTE, 16 décembre 2009.
- [10] R. ZHANG, «Sink localization and topology control in large scale heterogeneous wireless sensor networks», thesis of doctorate, university of South Florida, USA, 2007.
<http://scholarcommons.usf.edu/etd/2430.html>
- [11] GALLAIS, François INGELREST, Jean CARLE, David SIMPLOT-RYL, «Maintenance de la couverture de surface dans les réseaux de capteurs avec une couche physique non idéale», CIFP, Colloque Francophone sur l'ingénierie des Protocoles, 2006.
- [12] N. LASLA, «La gestion de clés dans les réseaux de capteurs sans fil», Thèse présentée pour obtenir le grade de Magister en Informatique de l'Institut National de formation en Informatique (I.N.I) Oued-Smar, Alger, 2006-2007.

- [13] N. Mitton, «Réseaux de capteurs, un état de l'art des algorithmes de routage», thèse pour l'obtention de grade de Magister, Institut national de recherche en informatique et en automatique, Lille, France, 2009.
- [14] K. CHAKRABARTY, S.S. IYENGAR, «Scalable Infrastructure for Distributed Sensor Networks», Edition Springer, USA, 2005.
ISBN: 978-1852339517
- [15] C. Buratti, A. Conti, D. Dardari, R. Verdone, «An Overview on Wireless Sensor Networks Technology and Evolution», Technical report, University of Bologna, Bologna, Italy, 2009.
ISSN: 1424-8220
- [16] G. JOLLY, M.C. KURSU, P. KOKATE, M. YOUNIS, «A Low- Energy Key Management Protocol for Wireless Sensor Networks», Magazine communications IEEE, 2003.
- [17] CHLAMTAC, L. CARRERAS, Hagen WOESNER, «From Internets to bionets: biological kinetic service oriented networks», Sensor Networks, Italy 2005.
- [18] L. Khelladi, N. Badache, «Réseaux de capteurs: état de l'art», Rapport de recherche, 2004.
- [19] F. KOUSHANFAR, M. POTKONJAK, A. SANGIOVANNI-VINCENTELLI, «Fault Tolerance in Wireless Ad hoc Sensor Networks». Proceeding of Networks, IEEE, June 2002.
- [20] L. PARADIS, Q. Han, «A Survey of Fault Management in Wireless Sensor Networks». Press New York, NY, USA, 2007.
- [21] F. NEKOOGAR, F. DOWLA, «Spiridon, Self organization of wireless sensor networks using ultra-wideband radios», Technical report, Atlanta, GA, United States, September 2004.
- [22] F. L. LEWIS, «Wireless Sensor Networks», Edition: D.J. Cook and S.K. Das, John Wiley, Automation a Robotics Research Institute, New York, USA, 2004.
- [23] T. ZHAO, W. D. CAI, Y. J. Li, «A Sensor Network Topology Inference Algorithm, computational Intelligence and Security», Conference international, Harbin, Chine, January 2008.
E-ISBN 978-0-7695-3072-7
- [24] R. MERAIHI, «Gestion de la qualité de service et contrôle de topologie dans les réseaux ad hoc», Thèse de doctorat, Ecole nationale supérieure des télécommunications, Paris, 2003.
- [25] D. CULLER, D. ESTRIN, M. SRIVASTAVA, «Overview of Sensor Networks», Scientific Research Report, Aston University, August 2004.
- [26] TEIXEIRA, J. F. REZENDE, A. CASTRO, A. C. P. PEDROZA, «Wireless Sensor Network: Improving the Network Energy Consumption», Telecommunications Institute, Brazil, September 2004.

- [27] E. SOUTO, R. GOMES, D. SADOK, J. KELNER, «*Sampling Energy Consumption in Wireless Sensor Networks*», International Conference of sensors networks IEEE, June 2006.
- [28] Carlos POMALAZA-REZ, «*Wireless Ad Hoc & Sensor Networks*», Technical report, Oulu University, Finland, 2004.
http://www.inets.rwth-aachen.de/pub/Cross_Layer_Energy_Analysis.pdf
- [29] H. CHAOUCHI, M. LAURENT-MAKNAVICIUS, «*Les réseaux sans fil et la sécurité*», Rapport Technique, Institut national de télécommunications, France, Octobre 2007.
- [30] C.KARLOF, D.WAGNER, «*Secure routing in wireless sensor networks: Attacks and countermeasures*», proceeding of the first international project for protocols and applications of Sensors Networks, May 2003.
- [31] K. BAUMGARTNER, «*Réseaux de capteurs sans fil*», Rapport PFE, Université de Rennes, Bretagne, Décembre 2005.
- [32] Y. CHALLAL, «*Réseau de capteurs sans fil*», Université de technologie Compiègne, France, Novembre 2008.
<http://www.stephan-robert.ch/attachments/File/Travaux-etudiants/coexistence.pdf>.
- [33] K. FELLAH, «*Techniques D'optimisation Pour L'économie D'énergie Dans Les Réseaux de Capteurs Sans Fil*», Mémoire de Magister en Informatique, Faculté des Sciences, Université d'Oran Es-Senia, 02/07/2008.
- [34] K. ROMER, F. MATTERN, «*The design space of wireless sensor networks*», Institute of calcul ETH Zurich, Suisse, December 2006.
[http://www.vs.inf.ethz.ch/publ/papers/wsn-design space.pdf](http://www.vs.inf.ethz.ch/publ/papers/wsn-design%20space.pdf)
- [35] S.SENTILLES, «*Architecture logicielle pour capteurs sans fil en réseau*», Rapport de recherche, Université de Pau et des pays de l'Adour, Juin 2006.
- [36] MILENKOVIC et Al, «*Wireless sensor networks for personal health monitoring: Issues and an implementation*», Rapport technique, Electrical Department, Alabama University of Huntsville, USA, August 2006.
<http://ebookbrowse.com/wireless-sensor-networks-for-personal-health-monitoring-issues-and-an-implementation-pdf-d256541147>
- [37] (E.M). PETRIU, (N.D). GEORGANAS, (D.C). PETRIU, D. MAKRAKIS, (V.Z). GROZA, «*Sensor-based information appliances*», Magazine of instrumentation measure IEEE, vol.3, no.4, p.31-35, December 2000.
- [38] L. MEKKAOUI, «*Compression d'images dans les réseaux de capteurs sans fil*», Thèse de doctorat en sciences, Spécialité Automatique, Traitement du Signal et Génie Informatique, Université de Lorraine, 26 Novembre 2012.

- [39] W. HEINZELMAN, A. CHANDRAKASAN, H. BALAKRISHNAN, «*Energy-Efficient Communication Protocol for Wireless Micro sensor Networks*», International Conference of scientific systems, vol. 8, pp. 8020, Hawaï, January 2000.
- [40] M. YOUNIS, T. NADEEM, «*Energy efficient MAC protocols for wireless sensor networks*», Rapport technique, Mryland Baltimore pays University, USA, 2004.
- [41] M. ILYAS et I. MAHGOUB. «*Handbook of sensor networks Compact wireless and wired Sensing Systems*», Technical report, CRC PRESS LLS, USA, 2005.
ISBN: 08493196864
- [42] H. NAMGOOG, D. LEE, D. NAM, «*Energy efficient topology for wireless micro-sensor networks*», ACM, PE-WASUN, October 2005.
- [43] S. ZIANE et A. MELLOUK, «*A swarm intelligent scheme for routing in mobile ad networks*». Communications Systems IEEE, August 2005.
- [44] Akyildiz, W. Su, Y. Sankarasubramaniam, et E. Cayirci, «*Wireless sensor networks: a survey*». Computer Networks, Amsterdam, Netherlands, 1999.
- [45] T. LEMLOUMA, «*Le Routage dans les Réseaux Mobiles Ad-Hoc*», PFE, Université des Sciences et de la Technologie Houari Boumediene, Département d'Informatique, Oran, Septembre 2000.
- [46] R. ABDELLAOUI, «*SU-OLSR une nouvelle solution pour la sécurité du protocole OLSR*», Mémoire soutenu pour l'obtention de la maîtrise en génie concentration Réseaux de Télécommunications, le 05 Mai 2009.
- [47] J.N AL-KARAKI, A.E KAMEL, «*Routing Techniques in Wireless Sensor Networks: A Survey*», PFE, Jordan University of Science and Technology, December 2004.
- [48] R. JURDAK, «*Wireless Ad Hoc and Sensor Networks: A cross Layer Design perspective*» Edition: Springer, Dublin, Irlande, 2007.
- [49] S. HAGGAR BECHAR, «*Les protocoles de routage dans les réseaux Ad Hoc*», Rapport de stage, Université de Reims –UFR sciences, France, 21 Juin 2007.
- [50] D.NICULESCU, «*Topics in Ad Hoc Networks: Communication Paradigms for Sensor Networks*», American laboratory NEC, Magazine Communications IEEE, March 2005.
- [51] G. CHALHOUB, «*Les réseaux de capteurs sans fil*», Rapport technique, Dpt R&T, Université Clermont, France, Janvier 2011.
<http://sancy.univ-bpclermont.fr/~chalhoub/workshopIUT10.pdf>
- [52] G. CHALHOUB, «*Les réseaux de capteurs sans fil*», Rapport technique, Dpt R&T, Université Clermont, France, Janvier 2011.
<http://sancy.univ-bpclermont.fr/~chalhoub/workshopIUT10.pdf>

- [53] C. TOWNSEND, S. ARMS, «*Wireless Sensor Networks: Principles and Applications*», Technical report, Savoie University, 2005.
<http://www.engpaper.com/wireless-sensors-research-papers-102.html>
- [54] C. LIASKOS, Dr. G. KOUTITAS, «*Topology Control in Wireless Sensor Networks*», Technical report, Hellenic International University, Greece, September 2010.
http://vlabs.ihu.edu.gr/fileadmin/labsfiles/wireless_communications/DOCUMENTATION/doc_vlab4.pdf
- [55] P. SANTI, «*Topology Control in Wireless Ad Hoc and Sensor Networks*», Edition: Wiley, New York, USA, 2005.
 ISBN-13 978-0-470-09453-2 (HB)
 ISBN-10 0-470-09453-2 (HB)
- [56] K. HOLGER, A. WILLIG, «*Protocols and Architectures for Wireless Sensor Networks*», Edition: Wiley, New York, USA, 2005.
- [57] G. MALKIN, RFC 2453, «*RIP Version 2*», Technical report, Jean Monnet University, French, 1998.
- [58] N. BOUKHECHEM, «*Routage dans les réseaux mobiles Ad Hoc par une approche à base d'agents*», Mémoire pour l'obtention du diplôme de Magister en Informatique, Université de Constantine, 25 Juin 2008.
- [59] D. B. JOHNSON, D. A. MALTZ, and J. BROCH, «*DSR: the Dynamic Source Routing Protocol for Multihop Wireless Ad Hoc Networks*», Ad Hoc networks, Charles E. Perkins, 2001.
- [60] C. E. PERKINS, E. M. ROYER, S. R. DAS, «*Ad hoc on demand distance vector (aodv) routing*», IETF, Internet Draft, draft-ietf-manet-aodv-05.txt, [En ligne] 2000.
- [61] F. RISSON, «*Le Routage au sein des réseaux Ad Hoc*», PFE en Informatique, Université de Lille, France, Décembre 2008.
- [62] M. ALI and S. K. RAVULA, «*Real-time support and energy efficiency in wireless sensor networks*», Technical report, IDE0805, January 2008.
- [63] (M. K). MARINA, (S. R). DAS, «*Ad hoc on-demand multipath distance vector routing*», ACM SIGMOBILE, July 2002.
- [64] P.WIGHTMAN, M.A. LABRADOR, «*A3: a topology control algorithm for wireless sensor networks*», IEEE Proceeding Globecom, 2008.
- [65] (R.P) Mario WIGHTMAN, (M.A). LABRADOR, «*Reducing the communication range or turning nodes off? An initial evaluation of topology control strategies for wireless sensor networks*», University of Nord Colombia, 1 October 2010.

- [66] Li. HALPERN, J. BAHL, P. WANGA, Y.M. WATTENHOFER, «*Analysis of distributed topology control algorithm for wireless multi-hop networks*», En: Proceeding Symposium ACM of principal computing , pp. 264–273, 2001.
- [67] Li. N. HOU, J.C. FLSS, «*A fault-tolerant topology control algorithm for wireless networks*», En: ACM Proceeding Mobicom, pp. 275–286, 2004.
- [68] Z. YUANYUAN, X.JIA, H. YANXIANG, «*Energy efficient distributed connected dominating sets construction in wireless sensor networks*». ACM Proceeding conference International on Communications and Mobile Computing, pp. 797–802, 2006.
- [69] J. WU, H. LI, «*On calculating connected dominating set for efficient routing in ad hoc wireless Networks*», Third International Proceeding ACM Workshop on Discrete Algorithm sand Methods for Mobile Computing and Communications, pp. 7–14, 1999.
- [70] Ji. WU, F. DAI, «*An extended localized algorithm for connected dominating set formation in ad hoc wireless networks*». IEEE. Parallel Distributed Syst. 15(10), 908–920, 2004.
- [71] P.M). Wightman ROJAS, «*Topology control in wireless sensor networks*», Graduate School Theses and Dissertations, University of South Florida, 6-1-2010.

Table des figures

FIG. I.1 – Les types de capteurs	15
FIG. I.2 – Architecture matérielle d'un capteur	16
FIG. I.3 – Rayon de communication et de sensation d'un capteur	17
FIG. I.4 – Capteur « MICA2 »	19
FIG. I.5 – Capteur « TelosB »	19
FIG. I.6 – Capteur « Imote2 »	19
FIG. I.7 – Capteur « TinyNode 584 »	20
FIG. I.8 – Diagramme de transition d'état du modèle du nœud capteur	21
FIG. I.9 – Architecture de communication d'un RCSF	22
FIG. I.10 – Pile protocolaire	22
FIG. I.11 – Architecture plate	24
FIG. I.12 – Architecture hiérarchique	25
FIG. I.13 – Applications militaires des RCSFs	30
FIG. I.14 – Applications médicales des RCSFs	31
FIG. I.15 – Applications environnementales des RCSFs	31
FIG. I.16 – Applications industrielles des RCSFs	32
FIG. I.17 – Applications d'agriculture des RCSFs	32
FIG. II.1 – La surécoute dans une transmission	37
FIG. II.2 – Techniques de conservation d'énergie existantes	38
FIG. II.3 – Transmission d'un message vers la station de base	39
FIG. II.4 – Classification des protocoles de routage dans les RCSFs	40
FIG. III.1 – Les deux requêtes RREQ et RREP en AODV	47
FIG. III.2 – Taille des nœuds avec TORA	48
FIG. III.3 – Le clustering dans un réseau de capteurs	49
FIG. III.4 – Les étapes du protocole A3	51
FIG. III.5 – La comparaison en terme de couverture entre A3 et A3Cov	53
FIG. III.6 – Exemple de α couverture	53
FIG. IV.1 – Energie des 3 nœuds du réseau en fonction du temps	62
FIG. IV.2 – Comparaison : Energies moyennes du réseau en fonction du temps des 4 protocoles	63
FIG. IV.3 – Energie des 10 nœuds du réseau en fonction du temps	65
FIG. IV.4 – Comparaison : Energies moyennes du réseau en fonction du temps des 4 protocoles	66
FIG. IV.5 – La fenêtre des options de déploiement du simulateur	68
FIG. IV.6 – La fenêtre de l'onglette Aarraya	69
FIG. IV.7 – Caractéristiques des protocoles de construction	70
FIG. IV.8 – Performances dans un réseau clairsemé quand $R_{comm}=R_{det}$	73
FIG. IV.9 – Performances dans un réseau dense quand $R_{comm}=R_{det}$	74
FIG. IV.10 – Performances dans un réseau clairsemé quand $R_{comm}>R_{det}$	75
FIG. IV.11 – Performances dans un réseau dense quand $R_{comm}>R_{det}$	76
FIG. IV.12 – Evaluation des performances de SGETRot dans un réseau clairsemé	78
FIG. IV.13 – Evaluation des performances de SGETRot dans un réseau dense	78
FIG. IV.14 – Evaluation des performances de DGETRec dans un réseau clairsemé	79
FIG. IV.15 – Evaluation des performances de DGETRec dans un réseau dense	80