



TER M1 INFORMATIQUE

# Analyse et conception d'algorithmes économes en énergie dans les réseaux de capteurs

*Auteurs :*

Chloé DESDOUITS  
Sofiane Zahir KALI  
Rabah LAOUADI  
Samuel ROUQUIE

*Encadrante :*

Anne-élisabeth BAERT

# Table des matières

<b>1</b>	<b>Introduction</b>	<b>3</b>
1.1	Capteurs . . . . .	3
1.2	Caractéristiques techniques des capteurs actuels . . . . .	5
1.3	Réseau de capteurs sans fil (Wireless sensors network) . . . . .	5
1.3.1	Architecture . . . . .	6
1.3.2	Caractéristiques . . . . .	7
1.3.3	Problématique et Défis . . . . .	7
<b>2</b>	<b>État de l’art</b>	<b>9</b>
2.1	Modélisation des Réseaux de Capteurs sans fil . . . . .	9
2.1.1	Modèle de base . . . . .	9
2.1.2	Modèle énergétique . . . . .	11
2.1.3	Initialisation et mise à jour d’un capteur . . . . .	11
2.1.4	Broadcast . . . . .	12
2.2	Etat de l’Art . . . . .	13
2.2.1	Algorithmes sans balisage . . . . .	13
2.2.2	Algorithmes globaux . . . . .	13
2.2.3	Algorithmes avec balisage . . . . .	14
2.2.4	Probalistics . . . . .	16
2.2.5	Protocoles . . . . .	17
2.2.6	Elements de classification . . . . .	17
2.2.7	Synthèse . . . . .	19

<b>3</b>	<b>Analye et reflexion</b>	<b>21</b>
3.1	Reflexions sur la duree de vie . . . . .	21
3.1.1	Durée de vie du reseau, problématique . . . . .	21
3.1.2	Differentes définitions . . . . .	21
3.1.3	Comparatif . . . . .	22
3.2	Nouvelle approche du broadcast . . . . .	22
<b>4</b>	<b>Simulations et résultats</b>	<b>23</b>
<b>5</b>	<b>Conclusion</b>	<b>24</b>

# Chapitre 1

## Introduction

Les réseaux sans fil font depuis plus d'une dizaine d'années partie intégrante de la vie quotidienne des entreprises, des particuliers, de l'industrie et d'autres organisations. Ils représentent aujourd'hui une des briques de base sur lesquelles vont se fonder les systèmes intelligents omniprésents qui vont constituer une des technologies de l'avenir. Cependant, la majeure partie de ces technologies sans fils, à commencer par le Wifi, elle est basée sur des infrastructures fixes, limitant la mobilité des utilisateurs. Pour faciliter cette mobilité, il existe un autre type de réseau, de plus en plus courant, qui permet aux nœuds du réseau de communiquer directement entre eux sans nécessiter d'infrastructure : ce sont les réseaux ad hoc.

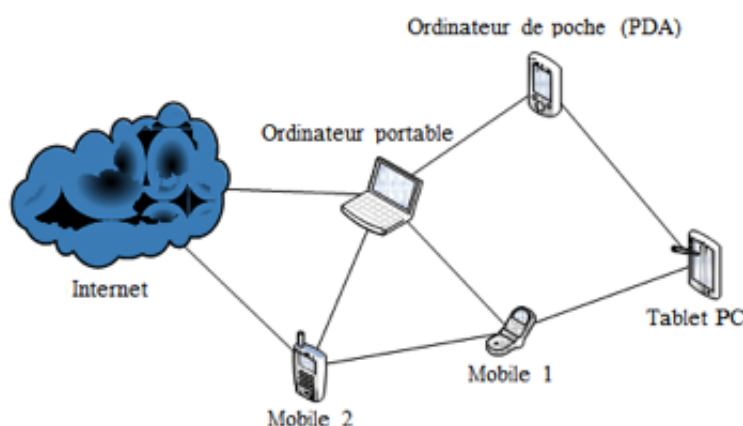


FIGURE 1.1 – Réseau ad hoc

On distingue donc deux principales classes de réseaux sans fils, les classiques structurés et les non structurés comme les réseaux ad hoc. Les réseaux ad hoc offrent la possibilité de connecter différents dispositifs sans avoir à préinstaller une infrastructure fixe comme dans les réseaux traditionnels. Dans les réseaux ad hoc, l'ensemble des nœuds communiquent directement entre eux (voir figure 1.1). Nous allons nous intéresser à un type particulier de réseau ad hoc : les réseaux de capteurs. Ces réseaux ont de nombreuses applications pratiques dans le médicale, la physique, la chimie, le multimédia, l'automobile, la climatologie...

### 1.1 Capteurs

Les capteurs sont des petites entités électroniques à faible coût qui ont pour but de récolter des informations dans leur environnement proche comme la température, la vitesse, le bruit, la pression,

le mouvement, la chaleur ou la lumière... La valeur mesurée est convertie dans une représentation analogique ou numérique.



FIGURE 1.2 – Capteur sans fil

Il existe des **capteurs intelligents** (Smart Sensors) dans lesquels coexistent le(s) capteur(s) et les circuits de traitement et de communication. Leurs relations avec des couches de traitement supérieures vont bien au-delà d'une simple « transduction de signal ». Les capteurs intelligents sont des « capteurs d'informations » et non pas simplement des capteurs et des circuits de traitement du signal juxtaposés. De plus, les « Smart Sensors » ne sont pas des dispositifs banalisés car chacun de leurs constituants a été conçu dans l'objectif d'une application bien spécifique.

Lorsque nous parlerons de capteur plus loin dans ce rapport, il s'agira d'un capteur intelligent. Un tel capteur contient quatre unités de base (voir Figure 1.3) :

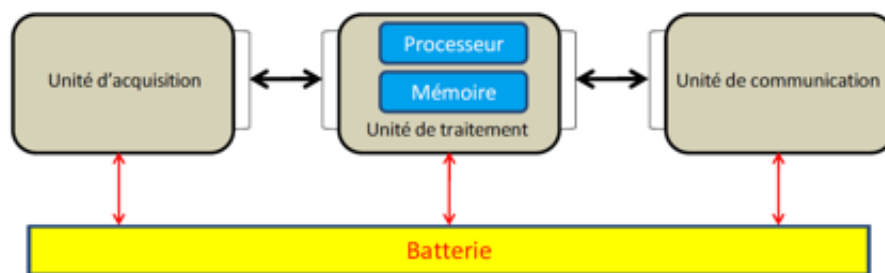


FIGURE 1.3 – Architecture d'un capteur

**L'unité d'acquisition** est composée d'un capteur qui obtient des mesures sur les paramètres environnementaux et d'un convertisseur Analogique/Numérique qui convertit l'information relevée et la transmet à l'unité de traitement. La perception d'un capteur est limitée par un rayon de sensation ( $R_s$ ). La Figure 1.4 illustre ce principe.

**L'unité de traitement** est composée d'un processeur et d'une mémoire intégrant un système d'exploitation spécifique. Cette unité possède deux interfaces, une interface pour l'unité d'acquisition et une interface pour l'unité de communication. Elle acquiert les informations en provenance de l'unité d'acquisition et les envoie à l'unité de communication. Cette unité est chargée aussi d'exécuter les protocoles de communications qui permettent de faire collaborer le capteur avec d'autres capteurs. Elle peut aussi analyser les données captées.

**L'unité de communication** est l'unité responsable de toutes les émissions et réceptions de données via un support de communication radio. Elle peut être de type optique, ou de type radiofréquence. Fonctionnellement chaque capteur possède un rayon de communication ( $R_c$ ). La figure 1.4 montre la zone dans laquelle le capteur peut communiquer. Certains capteurs peuvent moduler leur rayon de communication.

**L'unité de contrôle d'énergie (batterie)** sert à alimenter tous les composants. Cependant, à cause de la taille réduite du capteur, la batterie est limitée et généralement irremplaçable. Ainsi, l'énergie est la ressource la plus précieuse puisqu'elle influe directement sur la durée de vie des capteurs.

Selon son domaine d'application, un capteur peut contenir des modules supplémentaires comme le système de positionnement GPS (Global Positioning System) ou un système lui permettant de se déplacer.

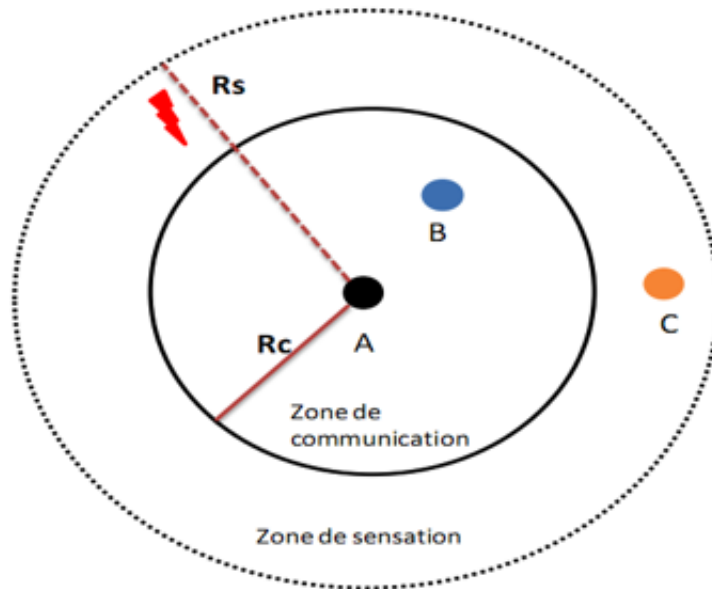


FIGURE 1.4 – Rayon de communication et de sensation

## 1.2 Caractéristiques techniques des capteurs actuels

**Une faible puissance de calcul** : quand les ordinateurs peuvent avoir jusqu'à 4 processeurs, chacun cadencé à 3GHz, ou quand les derniers Smartphones peuvent fonctionner jusqu'à 800MHz, un capteur actuel est à peine plus puissant qu'une calculatrice graphique produite dans les années 90.

**Un espace de stockage mémoire limité** à quelques kilo-octets ou quelques méga-octets impose l'utilisation d'algorithmes distribués, localisés et collaboratifs.

**Une puissance radio limitée** : l'ordre de grandeur des portées actuellement atteignables par les principaux capteurs est d'une centaine de mètres en extérieur et de quelques dizaines de mètres en intérieur. Cette portée est largement dépendante de la fréquence utilisée et de l'environnement. Elle nécessite un routage multi-saut pour l'acheminement des données vers une entité de collecte : le puits. Les capteurs ne peuvent communiquer qu'avec leur voisinage direct qui va relayer les communications.

**Un débit faible** : les composants radio d'un capteur sont limités à quelques centaines de kilo-octets par seconde.

**Une réserve d'énergie réduite** : même s'il existe des mécanismes de recharge d'énergie, la durée de vie d'un capteur reste directement liée au niveau de sa batterie. Cette réserve d'énergie est partagée par chaque unité d'un capteur mais l'unité de communication va en consommer près de 95% lors du fonctionnement actif du capteur. Les enjeux actuels portent donc sur :

- l'augmentation des capacités des batteries
- les dispositifs de transmission radio ultra-basse consommation
- les architectures basse consommation
- des mécanismes d'endormissement
- des protocoles de communication spécifiques

## 1.3 Réseau de capteurs sans fil (Wireless sensors network)

Les réseaux de capteurs sans fil sont un type particulier de réseau ad-hoc. Ces réseaux sont formés d'une multitude de capteurs, capables de s'auto-organiser et ainsi de travailler pour la collecte, le partage et le traitement coopératif des informations sur leur environnement ; le tout sans intervention humaine. Ces dispositifs sont peu coûteux, mais peu performants. Depuis quelques décennies, le besoin croissant

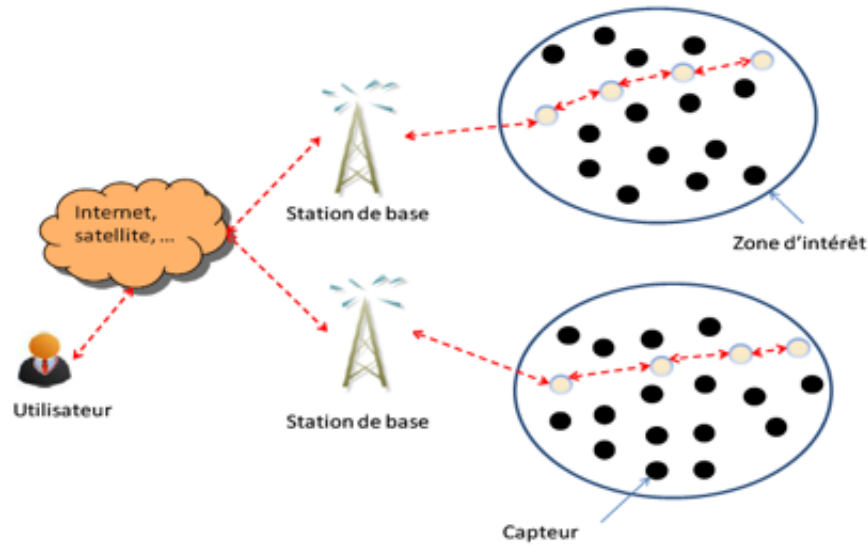


FIGURE 1.5 – Réseaux de capteurs

d'observer et de contrôler des phénomènes physiques tels que la température, la pression ou encore la luminosité a conduit au déploiement de nombreux réseaux de capteurs.

Dans l'exemple de la figure 1.5, les capteurs sont déployés d'une manière aléatoire dans une zone d'intérêt, et une station de base, située à l'extrémité de cette zone, est chargée de récupérer les données collectées par les capteurs. Lorsqu'un capteur détecte un événement pertinent, un message d'alerte est envoyé à la station de base par le biais d'une communication entre les capteurs. Les données collectées sont traitées et analysées par des machines puissantes.

### 1.3.1 Architecture

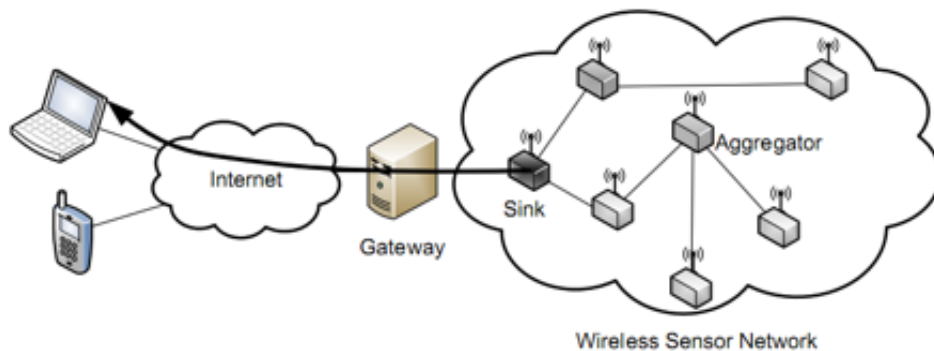


FIGURE 1.6 – architecture d'un réseau WSN

Les réseaux de capteurs sans fils sont construits autour des quatre principales entités suivantes :

**Les capteurs** décrits précédemment.

**L'agrégateur** est en charge d'agréger les messages qu'il reçoit de plusieurs capteurs puis de les envoyer en un seul message au puits. Cette opération a pour principal but de limiter le trafic sur le réseau et donc de prolonger la durée de vie globale du réseau de capteur.

**Le puits** est le nœud final du réseau. C'est à lui qu'est envoyé l'ensemble des valeurs mesurées par le réseau. Il peut arriver qu'il y ait plusieurs puits sur un même réseau de capteurs.

**La passerelle** est un dispositif qui a la particularité d'avoir deux interfaces réseau. Elle permet de relier le réseau de capteurs sans fils à un réseau plus traditionnel, typiquement l'internet. Habituellement le réseau de capteurs ne sert qu'à faire remonter les mesures, les applications traitant ces informations étant exécutées sur la machine de l'utilisateur final.

### 1.3.2 Caractéristiques

**L'absence d'infrastructure** préexistante et de tout genre d'administration centralisée.

**Des interférences** : les liens radio ne sont pas isolés, deux transmissions simultanées sur une même fréquence, ou utilisant des fréquences proches, peuvent interférer.

**Une taille importante** : un réseau de capteurs peut contenir des milliers de nœuds.

**L'hétérogénéité des nœuds** : plusieurs types de capteurs différents connectés entre eux.

**Une topologie dynamique** : les capteurs peuvent être attachés à des objets mobiles qui se déplacent d'une façon libre et arbitraire rendant ainsi la topologie du réseau fréquemment changeante.

**Des contraintes énergétiques** : la caractéristique la plus critique dans les réseaux de capteurs est la modestie de ses ressources énergétiques (batterie).

**La capacité de stockage et la puissance de calcul** sont limitées dans un capteur.

**Une bande passante limitée** en raison des caractéristiques techniques des radios.

**Le faible coût du matériel** qui facilite une redondance des liens pour assurer une connexité du réseau en cas de panne d'un ou plusieurs capteurs.

**L'impossibilité de remplacer manuellement les nœuds** dans le cas où leur position est inconnue (déploiement rapide dans des conditions difficiles ; absence de puce GPS...).

**Le caractère aléatoire de la topologie du réseau** : celui-ci est déployé en fonction des zones d'intérêt ou aléatoirement.

### 1.3.3 Problématique et Défis

Les caractéristiques particulières des réseaux de capteurs modifient les critères de performance par rapport aux réseaux sans fil traditionnels. Dans les réseaux locaux filaires ou les réseaux cellulaires, les critères les plus pertinents sont le débit, la latence et la qualité de service. En effet, les nouvelles activités telles que le transfert d'images, le transfert de vidéos, et la navigation sur Internet requièrent de bonnes performances selon ces trois critères.

En revanche, dans les réseaux de capteurs conçus pour surveiller une zone d'intérêt, la longévité du réseau est fondamentale. De ce fait, la conservation de l'énergie est devenue un critère de performance prépondérant et se pose en premier lieu tandis que les autres critères comme le débit ou l'utilisation de la bande passante sont devenus secondaires.

Les perspectives d'application des réseaux de capteurs sont enthousiasmantes mais les défis qu'elles posent n'en sont pas moins nombreux et complexes. Parmi les problématiques cruciales, nous pouvons citer :

**L'énergie** : cette contrainte impose de concevoir des protocoles économes en énergie.

**Le routage** : le problème de routage consiste à déterminer un acheminement optimal des paquets à travers le réseau au sens d'un certain critère de performance (l'énergie par exemple).

**La sécurité** : la puissance de calcul limitée d'un capteur ouvre de véritables défis pour concevoir des algorithmes de cryptages distribués et des politiques de confiance spécifiques.

**La collecte de données** : récupérer les données des capteurs et les assembler.



**L'auto-configuration** : une partie des applications visées appartient au domaine des applications domestiques. Il est donc important que le routage, l'intégration et l'adaptation à l'environnement soient transparents pour l'utilisateur.

**L'autoréparation** : les capteurs sont parfois inaccessibles (intégrés dans un mur, installés chez un particulier, déployés dans une zone dangereuse, etc.) et de conception peu sûre (faible coût de production). Une solution complète doit donc gérer efficacement la perte ou l'ajout d'un nœud dans le réseau.

**La localisation** : il s'agit de concevoir des mécanismes de localisation réalistes vis-à-vis des contraintes et des applications propres aux réseaux de capteurs. Les solutions actuellement proposées sont soit imprécises, soit coûteuses en énergie ou en matériel.

# Chapitre 2

## État de l'art

### 2.1 Modélisation des Réseaux de Capteurs sans fil

#### 2.1.1 Modèle de base

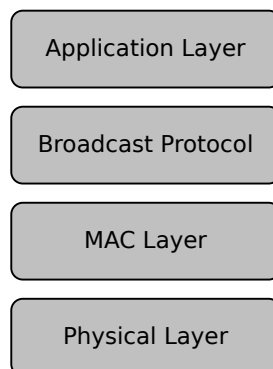


FIGURE 2.1 – Les couches dans les WSNs

Pour l'étude et la conception d'algorithmes dans les réseaux de capteurs sans fil (WSN), il est indispensable de définir le problème précis, d'établir un cadre rigoureux, formel et sans ambiguïtés. Au vue de la figure 2.1, notre étude se placera dans la couche 'Broadcast Protocols'. Nous ne développerons pas la couche MAC ni la couche application.

La modélisation de base des réseaux de capteurs sans fil  $M_1$  :

- Tout les capteurs sont identiques (batterie, portée, capacité de calcul..).
- le réseau est initialement connexe (chaque capteur est lié directement ou indirectement à tout les autres).
- Dans un plan euclidien à deux dimensions (distance euclidienne).
- Sans mobilité des capteurs : nous supposerons que les capteurs sont immobiles.
- Sans ajout de capteurs : le réseau comprend un nombre fixe de capteurs  $n$ . Aucun ajout de capteurs en cours de fonctionnement n'est possible.
- Dans des conditions de transmission de message idéale : aucune interférence entre les messages, pas de perturbations des ondes, système d'identifiants unique.

- Fiabilité des capteurs : aucune panne possible, énergie des capteurs infinie .
- Chaque site peut a tout moment debuter une procedure de broadcast. Une loi de probabilité modélise ce phénomène.
- (Chaque site connait sa position absolue. )

La modélisation  $M_2$  prendra de plus en compte :

- Chaque capteur a une énergie initiale  $E_i$  donnée. L'envoi de messages consomme une quantité d'énergie en fonction du modèle choisi. Un capteur est éliminé lorsqu'il n'a plus d'énergie ou que son énergie restante ne permet plus aucun envoi de message. Les capteurs ne tombent pas en panne en raison d'autres facteurs.
- L'ajout de capteurs : le réseau comprend un nombre variable de capteurs  $n$ . Les ajouts de capteurs en cours de fonctionnement sont possibles.

La modélisation  $M_3$  prendra de plus en compte :

- Les capteurs peuvent tomber en panne en raisons de divers facteurs. Une loi de probabilité modélise ce phénomène.
- La mobilité des capteurs.

Un WSN peut être représenté par un graphe  $G = (V, E, \gamma)$  ou  $V$  est un ensemble de noeuds (capteurs),  $\gamma$  le rayon d'émission maximum et  $E \subseteq V^2$  l'ensemble des arêtes représentant les communications possibles entre les capteurs :  $(u, v)$  appartient à  $E$  signifie que  $u$  peut envoyer un message à  $v$ . On note  $n = |V|$  la taille du WSN. En fait les éléments de  $E$  dépendent de la position des capteurs ainsi que de leur portée. Nous supposons que tous les capteurs ont la même portée maximale notée  $\gamma$ . Nous noterons  $ij$  l'arête allant de  $i$  à  $j$ . Nous noterons  $d_e(u, v)$  la distance euclidienne dans  $\mathbb{R}^2$  entre  $u$  et  $v$ .

$$E = \{(u, v) \in V^2 \mid d(u, v) \leq R\}$$

**Définition 1.** On appellera  $G = (V, E, \gamma)$  le *graphe unité* du WSN et  $\gamma$  son rayon de communication.

**Définition 2.** On note le 1-voisinage de  $u$  :  $N_1(u) = \{v \in V \mid (u, v) \in E\}$

On note le 2-voisinage de  $u$  :  $N_2(u) = \{v \in V \mid \exists w \in V : \{(u, w); (w, v)\} \in E^2\}$

On note le  $k$ -voisinage de  $u$ ,  $k \in \mathbb{N}$  :  $N_k(u) = \{v \in V \mid \exists \text{ un chemin } c(u, v) : |c| \leq k\}$  On parlera de 1- 2- et  $k$ -voisins de  $i$  pour désigner l'ensemble des noeuds appartenant respectivement à  $N_1(i)$ ,  $N_2(i)$ ,  $N_k(i)$ .

Soit  $A \subseteq V$ , on note  $N(A) = \{v \in V \setminus A \mid \forall u \in A, (u, v) \in E\}$

**Définition 3.** On appellera diamètre de  $G$   $diameter_G = \max_{i, j \in [1, n], i < j} (d_G(i, j))$ .

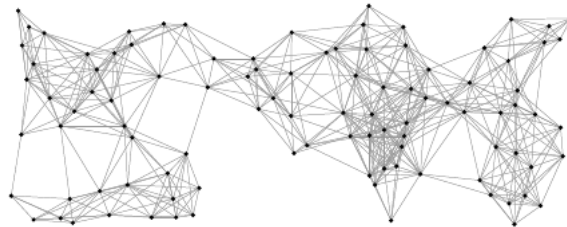


FIGURE 2.2 – Graphe unité

**Définition 4.** Le degré de  $u$  est le nombre  $N(u) = |N_1(u)|$ .

On note  $d(u, v)$  la distance entre  $u$  et  $v$  :  $d_G(u, v) = \min_{k \in \mathbb{N}} (k \mid v \in N_k(u))$

**Définition 5.** La densité de G est la moyenne des degrés :

$$N_G = \sum_{i=1}^n \frac{1}{n} N(i)$$

La distance de G est la moyenne des distances entre toutes paires de sommets :

$$d_G = \sum_{i,j \in [1,n], i < j} \frac{1}{n} d_G(i, j)$$

La distance euclidienne de G est la moyenne des distances euclidienne entre toutes paires de sommets :

$$d_e(G) = \sum_{i,j \in [1,n], i < j} \frac{1}{n} d_e(i, j)$$

### 2.1.2 Modèle énergétique

#### Energie d'un capteur

Tous les capteurs  $i$  offrant les mêmes caractéristiques, ils peuvent modifier leur rayon d'émission  $r_i$  entre  $r_i = 0$  et  $r_i = \gamma$ . Nous considérerons que chaque capteur  $i$  a une énergie initiale  $E_{init} = \beta$ . Nous noterons  $E_i$  l'énergie restante de  $i$ . L'envoi d'un message de  $i$  avec un rayon  $r$  coûte

$$E(r) = \begin{cases} r^\alpha + c & \text{si } i \neq j \\ 0 & \text{sinon} \end{cases}$$

L'envoi d'un message de  $i$  à  $j$  ( $d_e(i, j) \leq \gamma$ ) coûte  $E_{ij} = E(d_e(i, j))$ . L'énergie consommée lors de la réception de message, du traitement et des calculs propres aux capteurs est négligée.

#### Energie globale

**Définition 6.** L'énergie potentielle de G est la somme des énergies des capteurs :

$$E_G = \sum_{i=1}^n E_i$$

La consommation de G est :

$$C_G = n\beta - E_G$$

Le coût moyen de transmission de G est :

$$c_G = E(d_e(G))$$

### 2.1.3 Initialisation et mise à jour d'un capteur

Un capteur 'prend naissance', c'est à dire débute son activité dès qu'il est en place dans le réseaux et ce automatiquement. Deux choix sont possibles : soit le capteur est immédiatement opérationnel soit il nécessite une phase d'initialisation. Il existe donc deux type d'algorithmes : sans balisage (beaconless) et avec balisage. Les algorithmes 'beaconless' ne nécessitent pas de phase d'initialisation ni de mise à jour. Les nouveaux capteurs sont immédiatement opérationnels mais n'ont aucune connaissance de leur environnement et notamment de leur voisinage dans G. Dans les algorithmes avec balisage, tout capteur, lorsqu'il 'prend naissance' commence par une procédure d'initialisation et stocke en mémoire un certain nombre d'informations(voisins,groupe,topologie locale...). Au cour de l'algorithme, chaque noeud met

périodiquement à jour ces informations. Pour ce faire chaque site envoie régulièrement à ses voisins un message de type 'Hello' contenant par exemple son Id, sa position, sa dominante connexe, son degré, ses voisins, etc.

Quelle type d'informations est utilisée dans l'algorithme : informations globale du réseau ou informations locales ? La distinction entre global et local n'est pas toujours évidente. Des algorithmes centralisés peuvent être implémentés d'une manière distribuée en décidant par exemple d'un nœud possédant la connaissance globale du réseau. Sinon, au travers d'un échange séquentiel d'informations très localisées (1 ou 2-voisinage par exemple), chaque nœud peut combiner sa connaissance avec celle de ses voisins et ainsi obtenir une vision globale du réseau. Cependant, une telle phase de propagation coûte très cher en terme de nombre de messages échangés et de temps. Si le réseau est dynamique (mobilité des capteurs), maintenir une connaissance locale du réseau devient plus complexe tandis que tenir à jour la topologie globale de celui-ci devient impossible par le moyen cité précédemment. Ainsi, la quantité et la nature des informations nécessaires au déroulement de l'algorithme sont une bonne mesure de la capacité d'adaptation du protocole à un environnement dynamique.

1. **Global** : protocole de broadcast, centralisé ou distribué nécessitant une connaissance globale du réseau (ex : BIP).
2. **Quasi-global** : protocole distribué de broadcast nécessitant une connaissance quasi-globale du réseau.
3. **Quasi-local** : protocole distribué nécessitant une connaissance du réseau principalement locale et occasionnellement globale. (ex : Cluster networks : tandis que les groupes peuvent être construits de manière locale, des réactions en chaînes peuvent arriver).
4. **Local** : protocole distribué nécessitant une connaissance très locale du réseau. Tous les algorithmes de 1 ou 2-voisinage appartiennent à cette catégorie.

## 2.1.4 Broadcast

Dans un WSN  $G(V, E, \gamma)$ , deux capteurs peuvent communiquer directement uniquement si ils sont 1-voisins dans  $G$ . À cause de la perte de propagation des messages, le rayon de transmission est relativement limité, c'est pourquoi les communications doivent se faire par multi-sauts, parfois même si le destinataire est à distance 1 dans le graphe unité (pour des raisons énergétiques). Pour établir une connection entre deux nœuds non voisins, les messages doivent effectuer des sauts via des nœuds intermédiaires. Dans un large WSN, il est bien trop difficile pour un capteur voulant transmettre un message à un autre de trouver une route, à cause de l'absence d'infrastructures. La procédure de Broadcast est un mécanisme fondamentale pour la propagation des données ainsi que pour la découverte de route. C'est pourquoi, concevoir des algorithmes efficaces et économes en énergie est un problème primordial dans les WSN.

**Notations**

$G(V, E, \gamma)$	Graphe unité
$n$	Nombre de capteur
$\gamma$	Rayon d'émission maximum
$\alpha$	constante de consommation énergétique
$c$	constante de consommation énergétique
$\beta$	Energie initiale des capteur
$E_i$	Energie restante de $i$
$E_{ij}$	Coût d'envoi d'un message de $i$ à $j$
$N(u)$	Degré de $u$
$d_G(i, j)$	distance dans $G$ entre $i$ et $j$
$d_e(i, j)$	distance euclidienne entre $i$ et $j$
$N_k(u)$	nombre de $k$ -voisins de $u$
$N_G$	densité de $G$
$d_G$	distance de $G$
$d_e(G)$	distance euclidienne de $G$
$diameter_G$	diamètre de $G$

## 2.2 Etat de l'Art

### 2.2.1 Algorithmes sans balisage

**Bling Flooding** Le blind flooding ou broadcast aveugle est un algorithme glouton de broadcast. Lors de la réception d'un message par un noeud, si c'est la première fois qu'il le reçoit, il le broadcast à ses voisins (avec le rayon maximum), sinon il ne fait rien.

**Algorithme ABBA** 'Area-based beaconless reliable broadcasting in sensor networks'[15].

L'idée maîtresse de ABBA est assez simple. Nous supposons que les noeuds n'ont aucune connaissance de leur voisinage. Cependant ceux-ci connaissent leur position géographique (GPS par exemple). À la première réception d'un message, le noeud initialise un timer. Avant expiration du timer, le noeud peut recevoir d'autre copie du même message. Chaque capteur a un rayon d'émission fixe  $R$  et couvre une zone circulaire. Si un noeud  $u$  reçoit le même message de différentes sources et que ces sources couvrent sa zone, alors  $u$  ne retransmet pas le message. Cela signifie que chacun de ses voisins potentiels a déjà reçu le message. Si un noeud est couvert avant que son timer n'expire, il ne fait rien. Sinon il retransmet. Comme chaque couverture est un cercle de rayon  $R$ , le critère de couverture peut être simplifié : au lieu de s'intéresser au disque entier, un noeud vérifie uniquement si le périmètre  $\pi$  de sa zone est couvert par ses voisins lui ayant envoyé le message.

---

**Algorithm 1** ABBA

---

**Require:** : Un noeud source  $s$ , un message  $M$

Reception par  $u$  du message  $M$  venant de  $v$

Calculer la zone  $z$  couverte par  $v$  lors de la transmission ;

Déclancher le timer ;

**repeat**

Attendre la réception d'une autre copie de  $M$  ou que le timer expire ;

**if** Une copie de  $M$  est reçue **then**

Mettre à jour  $z$  ;

reinitialiser le timer ;

**end if**

**until** timeout expires ;

**if**  $\pi \not\subseteq z$  **then**

Retransmettre ;

**end if**

Ignorer les autres copies de  $M$  ;

---

### 2.2.2 Algorithmes globaux

**Algorithme BIP** Article [18]

Rappelons les désavantages des transmissions longue portée : interférences, coût énergétique (le modèle de consommation énergétique des noeuds est non linéaire par rapport au rayon à cause de l'atténuation du signal radio). C'est pourquoi il est nécessaire de trouver le bon compromis entre le rayon de transmission et le nombre de messages circulant : un large rayon de transmission coûte cher mais atteint beaucoup de noeuds ; un court rayon coûte très peu cher mais augmente le nombre de messages.

'Broadcast Incremental Protocol' est glouton et centralisé. Il se base sur l'algorithme de Prim : un algorithme permettant de construire un arbre couvrant minimal d'un graphe. Le principe de l'algorithme de Prim est de construire l'arbre couvrant minimal arête par arête : pour ajouter une arête à un arbre partiellement construit, il considère l'ensemble des arêtes dont une extrémité est connectée à l'arbre déjà construit, et l'autre extrémité ne l'est pas, et il choisit dans cet ensemble une arête de poids minimal qu'il

ajoute à l'arbre. L'algorithme commence avec un arbre couvrant contenant un noeud et zéro arêtes, et ajoute successivement  $n - 1$  arêtes.

La formation d'un arbre couvrant de poids minimum dans BIP suit le meme principe dans le sens ou les aretes sont une par une ajoutées à l'arbre. En fait, Bip utilise l'algorithme de Prim avec une difference fondamentale : au lieu d'utiliser des couts fixes  $P_{ij}$  sur les aretes (demeurant inchangés au court de la procédure), Bip actualise dynamiquement ces couts à chaque étapes (*i.e* à chaque ajout d'une arete), ce qui traduit le fait que le cout d'ajout d'une arete depends des noeuds déjà dans l'arbre :

$$\forall i \in BIP, \forall j \notin BIP, P'_{ij} = P_{ij} - P(i)$$

ou  $P_{ij}$  est le cout reel de transmission et  $P(i)$  le cout de broadcast de  $i$  dans l'arbre ( $P(i) = 0$  si  $i$  est une feuille,  $P(i) = \max_{j \in N_1(i) \cap BIP} (E_{ij})$  sinon ).  $P'_{ij}$  represente donc le cout d'ajout de  $j$  par un noeud appartenant au sous ensemble des feuilles directes de  $i$  (ou  $i$  lui meme si c'est une feuille). La paire  $\{i, j\}$  minimisant  $P'_{ij}$  est selectionnée et  $i$  transmet à  $j$ . Ainsi, une nouveau arete est ajoutée a chaque etape de l'algorithme.

---

**Algorithm 2** Procédure de construction du BIP-Tree

---

ENTREES  $G = (V, E)$  un graphe connexe,  $s$  une source  
 SORTIE Arbre BIP de racine  $s$   
 $B$  : ENSEMBLE des arêtes de l'arbre  
 $B \leftarrow \text{graphe}_{vide}$   
 Marquer  $s$   
 Creer les nouveaux poids :  $\forall j \notin BIP, P'_{sj} = P_{sj} - P(s) = P_{sj}$   
**while** il existe un sommet non marqué adjacent à un sommet marqué **do**  
   Mettre à jour les poids :  $\forall i \in BIP, \forall j \notin BIP, P'_{ij} = P'_{ij} - P(i)$   
   Sélectionner un sommet  $j$  non marqué adjacent à un sommet marqué  $i$  tel que  $(i, j)$  est l'arête sortante de plus faible poids  $P'_{ij}$   
    $B := B \cup (i, j)$   
   Marquer  $j$   
**end while**  
 Retourner  $B = (V, B)$

---

Contairement a Prim qui garantie l'optimalité de l'arbre couvrant en termes de cout total, Bip ne construit pas forcement un arbre de poids minimum. Un fois cet arbre construit, le broadcast se fait naturellement via celui-ci.

---

**Algorithm 3** BIP

---

ENTREES  $G = (V, E)$  un graphe connexe,  $s$  une source, un message  $M$   
 SORTIE BIP Broadcast  
 $s$  envoie  $M$  à ses fils dans son Bip-tree  
 Lors de reception de  $M$  par  $i$  :  
**if**  $i$  est un noeud **then**  
   Retransmettre  $M$  à ses fils sinon ne rien faire  
**end if**

---

### 2.2.3 Algorithmes avec balisage

Comme expliqué en 2.1.3, dans les algorithmes avec balisage, tout capteur, lorsqu'il 'prend naissance' commence par une procédure d'initialisation et stocke en mémoire un certain nombre d'informations(voisins,groupe,topologie locale...). Au cour de l'algorithme, chaque noeud met périodiquement à jour ces informations. Pour ce faire chaque site envoie regulierement a ses voisins un message de type 'Hello' contenant par exemple son Id, sa position, sa dominante connexe, son degres, ses voisins, etc.

**Decouverte 2-voisinage** La connaissance du 2-voisinage est un tres bon compromis conservant la localité du protocoles tout en minimisant le nombre de messages.

---

**Algorithm 4** Decouverte k-voisinage

---

```

for chaque noeud  $i$  do
  Broadcaster un message de type  $\langle \text{HELLO}, i \rangle$  avec un rayon de transmission  $\gamma$ 
end for
A la réception de  $\langle \text{HELLO}, j \rangle$  , ajouter  $j$  a son 1-voisinage.
Attendre  $\Delta$ 
for chaque noeud  $i$  do
  Broadcaster  $\langle \text{HELLO}, N_1(i), i \rangle$ , message contenant  $N_1(i)$  le 1-voisinage de  $i$ 
  A la réception de  $\langle \text{HELLO}, N_1(j), j \rangle$  , ajouter  $N_1(j)$  a son 2-voisinage.
end for

```

---

Complexité en messages :  $O(2n)$ .

**Algorithme LBIP** ‘Localized Broadcast Incremental Power Protocol for Wireless Ad Hoc Networks’ : article [10] LBIP est l’application local de BIP : au lieu de construire l’arbre de diffusion BIP de facon centralisé, il le construit de facon locale. Chaque site lorsqu’il recoit un message à retransmettre calcul son BIP-tree sur son 2-voisinage et diffuse le message par l’intermediaire de celui ci.

---

**Algorithm 5** LBIP

---

```

ENTREES  $G = (V, E)$  un graphe connexe,  $s$  une source, un message  $M$ 
SORTIE LBIP Broadcast
REQUIE Connaissance du 2-voisinage
 $s$  calcul son arbre et diffuse  $\langle \text{MSG}, s \rangle$  à ses fils
if  $u$  recoit  $\langle \text{MSG}, v \rangle$  : then
  if Le paquet contient des instruction pour  $u$  then
     $u$  construit son 2-BIP-tree est retransmet le message à ses fils
  end if
end if

```

---

**Algorithme DLBIP** ‘Dynamic Localized Broadcast Incremental Power Protocol for Wireless Ad Hoc Networks’ : article[5]. DLBIP est une amélioration de LBIP. Le principe est de repartir l’énergie consommée lors d’un broadcast en empreintant des chemins (arbre couvrants) différents en fonction de l’énergie restantes des noeuds qui vont servir de relais. A chaque broadcast, l’abre de diffusion est recalculé sur le meme graphe mais avec de nouveaux poids dépendants de l’énergie de communication entre les noeuds ainsi que de l’énergie propre à chaque noeud. Ce protocols permet que le niveau d’énergie propre à chaque noeud baisse globalement de façon identique et ainsi retarder un maximum la panne d’un capteur par manque d’énergie.

.....Algo a faire

---

**Algorithm 6** DLBIP

---

```

ENTREES  $G = (V, E)$  un graphe connexe,  $s$  une source, un message  $M$ 
SORTIE DLBIP Broadcast
REQUIE Connaissance du 2-voisinage
 $s$  calcul son arbre et diffuse  $\langle \text{MSG}, s \rangle$  à ses fils
if  $u$  recoit  $\langle \text{MSG}, v \rangle$  : then
  if Le paquet contient des instruction pour  $u$  then
     $u$  construit son 2-BIP-tree est retransmet le message à ses fils
  end if
end if

```

---



**Algorithme RRS** [3]

**Algorithme RBOP**

[2]

**Algorithme LBOP**

[2]

**Algorithme DLBOP**

[4]

**Algorithme TR-LBOP**

[10]

**Algorithme FONIA**

**Algorithme Dominating set**

In a *reliable* protocol, every node in the network is reached. The set of nodes that rebroadcast message in a reliable broadcasting scheme define a connected dominating set. A dominating set  $D(S)$  of a set  $S$  is a set of nodes such that each node from  $S$  either belongs to  $D(S)$  or has a neighboring node that belongs to  $D(S)$ . It is easy to observe that all nodes will receive the message if it is retransmitted only by nodes that belong to a connected dominating set. Connectivity provides propagation through the whole network, while domination assures reachability by all nodes. Broadcasting task can therefore be solved optimally by finding a connected dominating set of minimal size. Optimality here is measured by percentage of saved retransmissions in a reliable broadcasting scheme. Unfortunately, the problem of finding connected dominating set of minimal size is NP-complete, even if a node has global knowledge about the network [HH, LK, QVL]. Therefore one needs to apply heuristics to flood intelligently. Note also that a protocol that is reliable on the network layer may be very unreliable at the MAC layer, such as blind flooding. Excess messages in any protocol affect node's power and bandwidth available, thus the main goal is to describe a reliable broadcast protocol with minimal number of retransmissions, that is, to construct connected dominating set of minimal size. Note also that MAC layer cannot guarantee 100% reliability, due to the hidden terminal problem (a node simultaneously receiving messages from two other nodes that are not aware of each other transmission) and the probabilistic nature of protocols used.

## 2.2.4 Probabilistics

**Algorithme BWGOSSIP**

[12]

#### **Algorithme Duty-Cycle-Aware Broadcast in WSN**

[17],[13]

#### **Algorithme HEED**

[20]

#### **Algorithme WMH**

[1]

#### **Algorithme IRRIGATOR Firework**

[14]

#### **Algorithme ML2B**

[21]

#### **Algorithme MLE**

[7]

#### **Algorithme MPR**

#### **Algorithme Power Adaptive Broadcasting with Local Information**

[6]

### **2.2.5 Protocoles**

#### **NES**

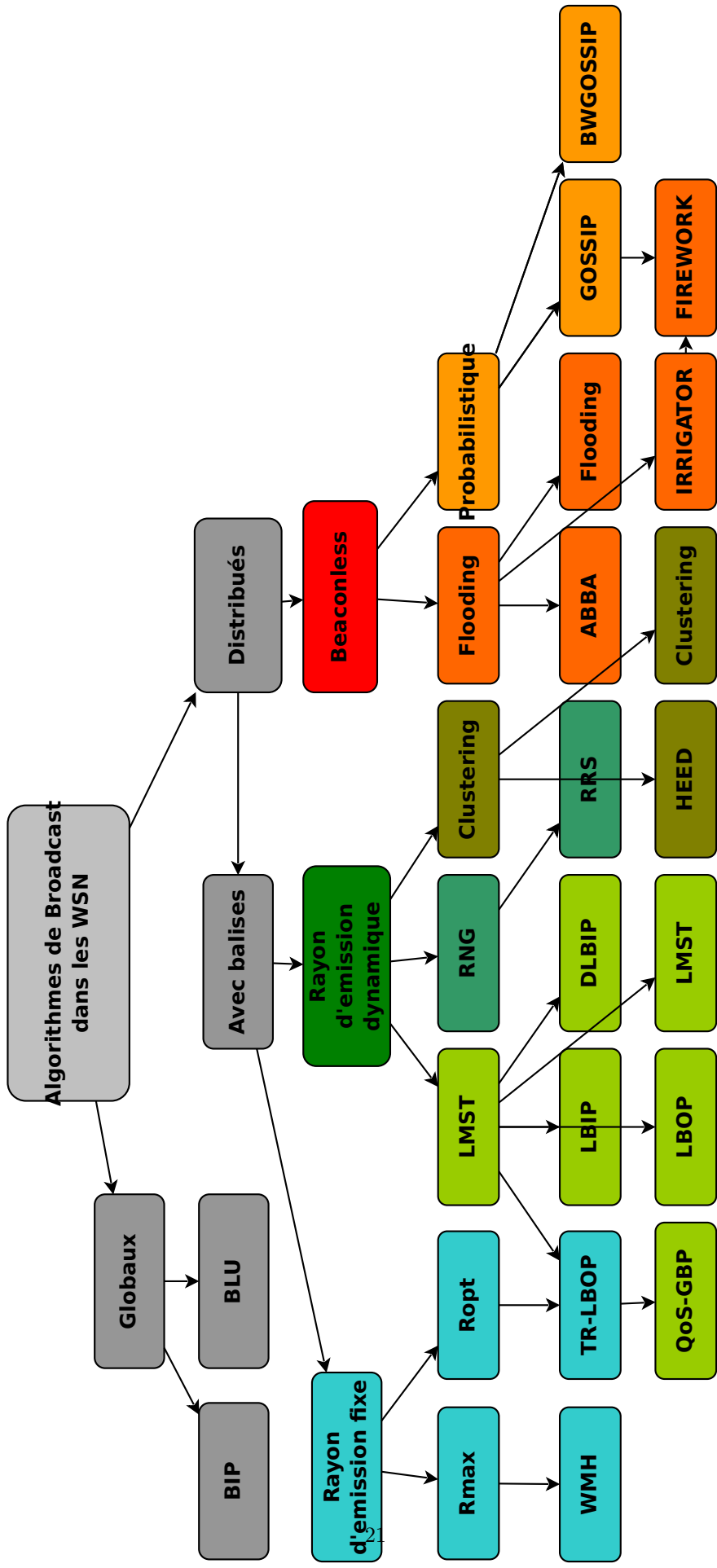
### **2.2.6 Elements de classification**

Les éléments de classification cités ci dessous sont inspirés des articles [16],[11],[19]

1. **Extensibilité** : Pour qu'un protocole soit extensible, il doit etre avant tout distribué et local : le comportement de chaque noeud bien qu'il nécessite qu'une connaissance locale du reseau, permet d'atteindre l'objectif global. Il est facile d'admettre que l'extensibilité d'un WSN est inversement proportionnelle à la localité du protocol.
2. **Connectivité** : La connectivité d'un algorithme mesure se capacité à atteindre tout les sites.

3. **Deterministe vs probabiliste** : Un protocol de broadcast peut utiliser des fonctions probabilistes pour prendre certaines descisions de maniere aléatoire.
4. **Rayon d'émission** : rayon fixe ou variable
5. **Avec ou sans balisage** :
6. :
7. basé sur les clusters
8. Grid Partitioning-Based Backbone
9. Connected Dominating Set (CDS)-Based Backbone
10. Activity scheduling and power-aware broadcasting
11. Broadcasting with directional antennas
12. Flooding scheme
13. Probabilistic flooding
14. 1-Hop Neighbor Knowledge Methods
15. Flooding with Self Pruning (FSP)
16. 2-Hop or More Neighbor Knowledge Methods
17. Multipoint Relaying (MPRs)
18. Using Connected Dominating Set (CDS)
19. Area Based Methods.....

### 2.2.7 Synthèse



# Chapitre 3

## Analyse et reflexion

### 3.1 Reflexions sur la duree de vie

#### 3.1.1 Durée de vie du reseau, problématique

#### 3.1.2 Differentes définitions

Les definitions et critères de durée de vie d'un WSN sont tirés des articles [8],[?],[9].

1. Energy-Efficiency (EE)
2. Time To First Fail (TTFF)
3. Operative Nodes Percentage (ONP)
4. Monitored Interest Points Percentage (MIPP)
5. The time there is at least a certain fraction  $\beta$  of surviving nodes in the network
6. The time until all nodes have been drained of their energy
7. K-coverage : the time the area of interest is covered by at least k nodes
8. 100 a. The time each target is covered by at least one node ; b. The time the whole area is covered by at least one node ;
9.  $\alpha$ -coverage a. The accumulated time during which at least  $\alpha$  portion of the region is covered by at least one node ; b. The time until the coverage drops below a predefined threshold  $\alpha$  (until last drop below threshold) ; c. The continuous operational time of the system before either the coverage or delivery ratio first drops below a predefined threshold ;
10. The number of successful data-gathering trips
11. The number of total transmitted messages
12. The percentage of nodes that have a path to the base station
13. Expectation of the entire interval during which the probability of guaranteeing connectivity and k-coverage simultaneously is at least  $\alpha$
14. The time until connectivity or coverage are lost
15. The time until the network no longer provides an acceptable event detection ratio
16. The time period during which the network continuously satisfies the application requirement
17.  $\min(t_1, t_2, t_3)$  with  $t_1$  : time for cardinality of largest connected component of communication graph to drop below  $c_1 \times n(t)$ ,  $t_2$  : time for  $n(t)$  to drop below  $c_2 \times n$ ,  $t_3$  : time for the covered volume to drop below  $c_3 \times l$  d .

### **3.1.3 Comparatif**

## **3.2 Nouvelle approche du broadcast**

## Chapitre 4

### Simulations et résultats



Chapitre 5

Conclusion

# Bibliographie

- [1] Manish Agarwal, Lixin Gao, Joon Ho Cho, and Jie Wu. Energy Efficient Broadcast in Wireless Ad hoc Networks with Hitch-hiking. *MONET*, 10(6) :897–910, 2005.
- [2] J. Cartigny, F. Ingelrest, D. Simplot-Ryl, and I. Stojmenovic. Localized LMST and RNG based minimum-energy broadcast protocols in ad hoc networks. *Ad Hoc Networks*, 3(1) :1–16, 2005.
- [3] Julien Cartigny, François Ingelrest, and David Simplot. RNG Relay Subset Flooding Protocols in Mobile Ad-Hoc Networks. *Int. J. Found. Comput. Sci.*, 14(2) :253–265, 2003.
- [4] Julien Cartigny, David Simplot-Ryl, and Ivan Stojmenovic. An Adaptive Localized Scheme for Energy-Efficient Broadcasting in Ad Hoc Networks with Directional Antennas. In Ignas G. Niemegeers and Sonia M. Heemstra de Groot, editors, *PWC*, volume 3260 of *Lecture Notes in Computer Science*, pages 399–413. Springer, 2004.
- [5] J. Champ, A.E. Baert, and V. Boudet. Dynamic localized broadcast incremental power protocol and lifetime in wireless ad hoc and sensor networks. *Wireless and Mobile Networking*, pages 286–296, 2009.
- [6] X. Chen, M. Faloutsos, and S.V. Krishnamurthy. Power adaptive broadcasting with local information in ad hoc networks. In *Network Protocols, 2003. Proceedings. 11th IEEE International Conference on*, pages 168–178. IEEE, 2003.
- [7] Maggie Xiaoyan Cheng, Jianhua Sun, Manki Min, Yingshu Li, and Weili Wu. Energy-efficient broadcast and multicast routing in multihop ad hoc wireless networks. *Wireless Communications and Mobile Computing*, 6 :213–223, 2006.
- [8] Isabel Dietrich and Falko Dressler. On the lifetime of wireless sensor networks. *TOSN*, 5(1), 2009.
- [9] Abdelrahman Elleithy and Gonhsin Liu. A simulation model for the lifetime of wireless sensor networks. *CoRR*, abs/1201.2237, 2012.
- [10] F. Ingelrest, D. Simplot-Ryl, and I. Stojmenovic. Target transmission radius over LMST for energy-efficient broadcast protocol in ad hoc networks. In *Communications, 2004 IEEE International Conference on*, volume 7, pages 4044–4049. IEEE, 2004.
- [11] F. Ingelrest, D. Simplot-Ryl, and I. Stojmenović. Energy-efficient broadcasting in wireless mobile ad hoc networks. *Resource Management in Wireless Networking*, pages 543–582, 2005.
- [12] F. Iutzeler, P. Ciblat, W. Hachem, and J. Jakubowicz. A NEW BROADCAST BASED DISTRIBUTED AVERAGING ALGORITHM OVER WIRELESS SENSOR NETWORKS.
- [13] Xianlong Jiao, Wei Lou, Junchao Ma, Jiannong Cao, Xiaodong Wang, and Xingming Zhou. Duty-Cycle-Aware Minimum Latency Broadcast Scheduling in Multi-hop Wireless Networks. In *ICDCS*, pages 754–763. IEEE Computer Society, 2010.
- [14] L. Orecchia, A. Panconesi, C. Petrioli, and A. Vitaletti. Localized techniques for broadcasting in wireless sensor networks. In *Proceedings of the 2004 joint workshop on Foundations of mobile computing*, pages 41–51. ACM, 2004.
- [15] Francisco Javier Ovalle-Martínez, Amiya Nayak, Ivan Stojmenovic, Jean Carle, and David Simplot-Ryl. Area-based beaconless reliable broadcasting in sensor networks. *IJSNet*, 1(1/2) :20–33, 2006.
- [16] I. Stojmenovic and J. Wu. Broadcasting and activity scheduling in ad hoc networks. *Mobile Ad Hoc Networking*, pages 205–229, 2004.
- [17] Feng Wang and Jiangchuan Liu. Duty-Cycle-Aware Broadcast in Wireless Sensor Networks. In *INFOCOM*, pages 468–476. IEEE, 2009.

- [18] Jeffrey E. Wieselthier, Gam D. Nguyen, and Anthony Ephremides. On the Construction of Energy-Efficient Broadcast and Multicast Trees in Wireless Networks. In *INFOCOM*, pages 585–594, 2000.
- [19] J. Wu and W. Lou. Forward-node-set-based broadcast in clustered mobile ad hoc networks. *Wireless Communications and Mobile Computing*, 3(2) :155–173, 2003.
- [20] O. Younis and S. Fahmy. HEED : a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks. *Mobile Computing, IEEE Transactions on*, 3(4) :366–379, 2004.
- [21] R. Zhao, X. Shen, and X. Zhang. Broadcast protocols for wireless sensor networks.