



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>
<b>3</b>	<b>Analyse et reflexion</b>	<b>10</b>
3.1	Modélisation des Réseaux de Capteurs sans fil . . . . .	10
3.1.1	Modèle de base . . . . .	10
3.1.2	Broadcast . . . . .	12
3.1.3	Modèle énergétique . . . . .	12
3.1.4	Initialisation et mise à jour d’un capteur . . . . .	13
3.1.5	Duree de vie . . . . .	14
3.2	Analyse d’algorithme existants . . . . .	15
3.2.1	Algorithmes sans balisage . . . . .	15
3.2.2	Algorithmes globaux . . . . .	16
3.2.3	Algorithmes avec balisage . . . . .	18
3.2.4	Elements de classification . . . . .	21
3.2.5	Synthèse . . . . .	22
3.3	Nos idées . . . . .	22
3.3.1	Application du rayon optimal à DLBIP . . . . .	22
3.3.2	. . . . .	23

<b>4</b>	<b>Simulations et résultats</b>	<b>24</b>
<b>5</b>	<b>Conclusion</b>	<b>25</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.

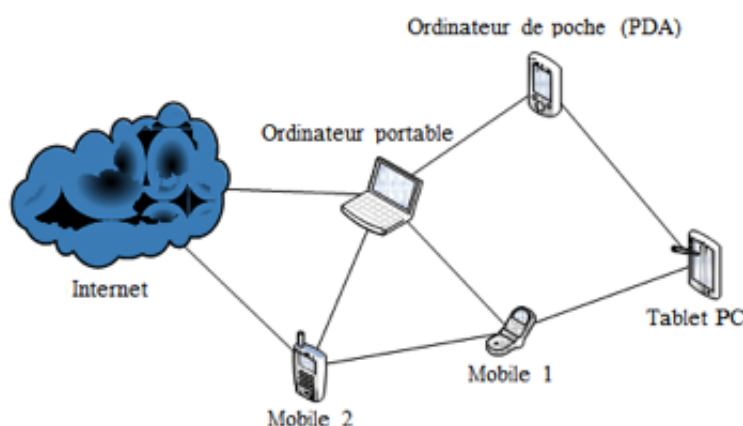


FIGURE 1.1 – Réseau ad hoc

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) :

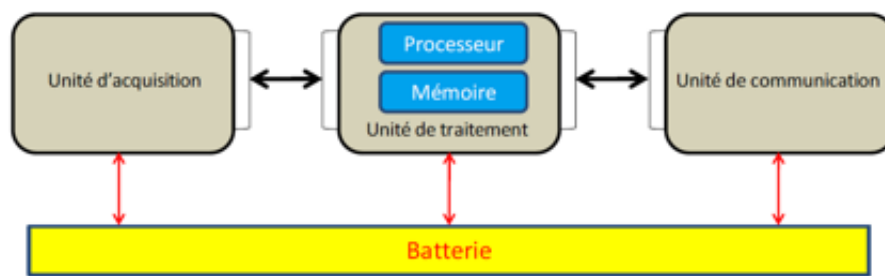


FIGURE 1.3 – Architecture d'un capteur

**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.

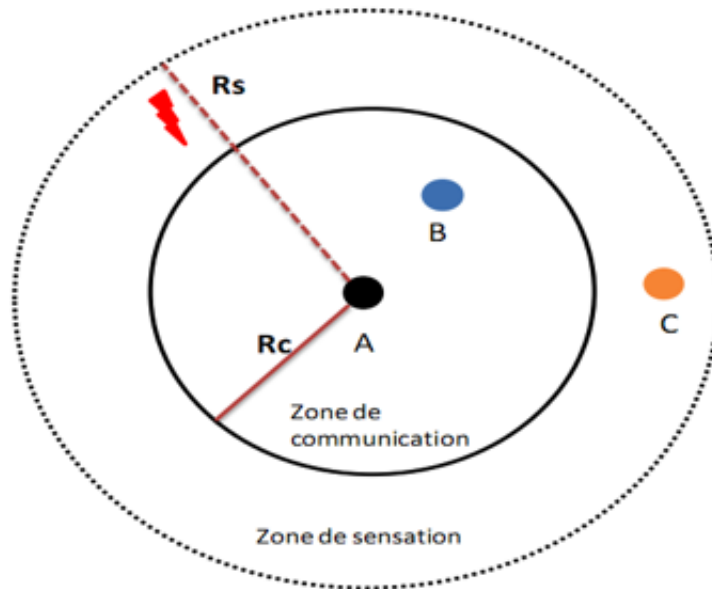


FIGURE 1.4 – Rayon de communication et de sensation

## 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

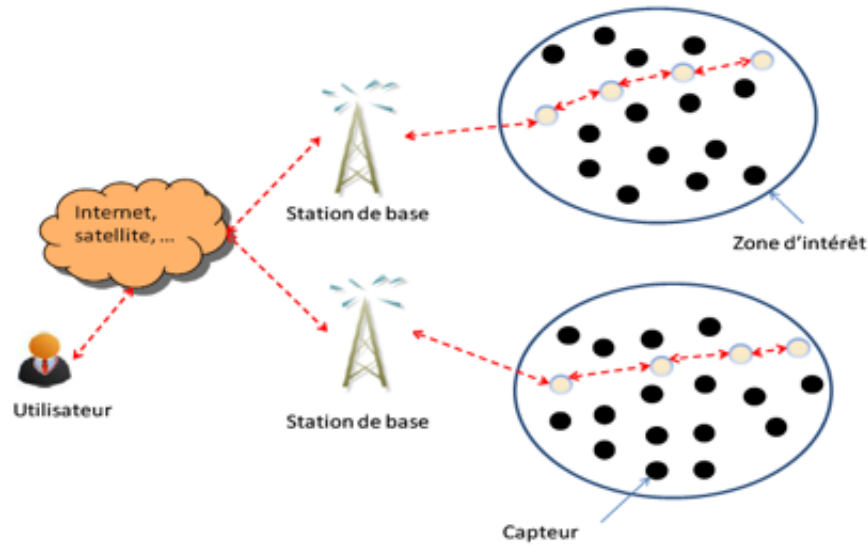


FIGURE 1.5 – Réseaux de capteurs

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

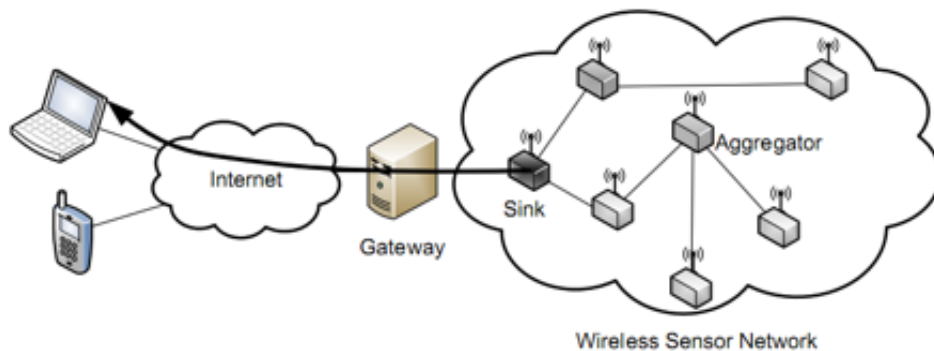


FIGURE 1.6 – architecture d'un réseau WSN

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.

Dans la Section suivante, nous faisons un bref Etat de l'art des algorithmes de broadcast dans les WSNs. Puis en Section 3, nous établissons un modèle et analysons quelques algorithmes existant. En Section 4, nous présentons des résultats de simulations sur la plateforme WSNET. Enfin nous concluons ce rapport.

## Chapitre 2

# État de l'art

Chloé????

Faire une jolie tartine sur les algo existant peut etre par ordre chronologique ou autre?

1-2 Pages

# Chapitre 3

## Analyse et reflexion

### 3.1 Modélisation des Réseaux de Capteurs sans fil

#### 3.1.1 Modèle de base

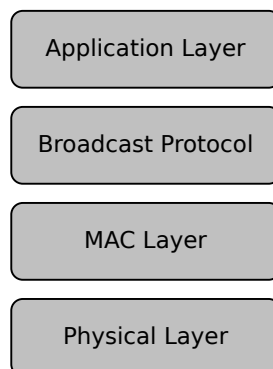


FIGURE 3.1 – Les couches dans les WSNs

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 3.1, notre étude se placera dans la couche 'Broadcast Protocols'. Nous ne développerons pas la couche MAC ni la couche application.

Differents modèles sont possibles :

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

- *Uniforme*. Tout les capteurs sont identiques (batterie, portée, capacité de calcul..).
- *Connexe*. Le réseau est initialement connexe (chaque capteur est lié directement ou indirectement à tout les autres).
- *Plan* Dans un plan euclidien à deux dimensions (distance euclidienne).
- *Statique*. 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.

- *Transmission*. Dans des conditions de transmission de message idéale : aucune interférence entre les messages, pas de perturbations des ondes, système d'identifiants unique.
- *Fiabilité*. Les capteurs sont fiables, aucunes pannes ne sont possibles.
- *Energie*. Chaque capteur a une énergie initiale  $\beta$  donnée. Un modèle décrit la consommation énergétique. Un capteur est éliminé lorsqu'il n'a plus d'énergie ou que son énergie restante ne permet plus aucun envoi de message.
- *Egalité*. Chaque site peut à tout moment débuter une procédure de broadcast. Une loi de probabilité modélise ce phénomène (loi de poisson).
- *Position*. Chaque site connaît sa position absolue.

D'autres modèles plus complexes mais plus réalistes prennent en compte :

- L'ajout de capteurs : le réseau comprend un nombre variable de capteurs  $n$ . Les ajouts de capteurs en cours de fonctionnement sont possibles.
- Les capteurs peuvent tomber en panne en raison de divers facteurs. Une loi de probabilité modélise ce phénomène.
- La mobilité des capteurs. La position de chaque capteur varie au cours du temps.
- Les capteurs ne sont pas forcément identiques (batterie, portée, capacité de calcul, connaissance du réseau...).
- Espace en 3 dimensions.
- Interférences radio.
- Modèle de consommation énergétique complexe.
- ...

Dans le cadre de notre travail d'étude et de recherche, nous resterons sauf mention du contraire dans la modélisation  $M_1$  avec pour modèle énergétique celui décrit ci-dessous.

Un WSN peut être représenté par un graphe  $G = (V, E, \gamma)$  où  $V$  est un ensemble de nœuds (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  $\gamma$ .

**Définition 1.** 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  $v$  :

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

Nous noterons  $d_G(u, v)$  la *distance* entre  $u$  et  $v$  :  $d_G(u, v) = \min_{k \in \mathbb{N}} (k \mid v \in N_k(u))$

**Définition 2.** On appellera  $G = (V, E, \gamma)$  le *graphe unité* du WSN et  $\gamma$  son rayon de communication.

**Définition 3.** Nous noterons le 1-voisinage de  $u$  :  $N_1(u) = \{v \in V \mid (u, v) \in E\}$

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

Nous noterons le  $k$ -voisinage de  $u$ ,  $k \in \mathbb{N}$  :  $N_k(u) = \{v \in V \mid \exists \text{ un chemin } c(u, v) : |c| \leq k\}$  Nous parlerons de 1- 2- et  $k$ -voisins de  $i$  pour désigner des nœuds appartenant respectivement à  $N_1(i)$ ,  $N_2(i)$ ,  $N_k(i)$ .

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

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

**Définition 4.** Nous noterons  $diameter_G = \max_{i, j \in [1, n], i < j} (d_G(i, j))$ .

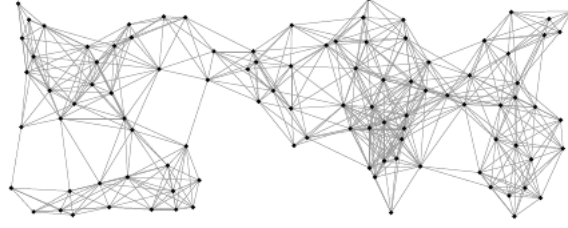


FIGURE 3.2 – Graphe unité

**Définition 5.** Le degré de  $G$  est la moyenne des degrés :

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

La densité de  $G$  est le nombre  $D_G = N_G / \text{diametre}_G$

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 :

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

### 3.1.2 Broadcast

Dans un WSN  $G(V, E, \gamma)$ , deux capteurs peuvent communiquer directement uniquement si ils sont 1-voisins dans  $G$ . A 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 meme si le destinataire est a distance 1 dans le graphe unité (pour des raison énergetiques). Pour établir une connection entre deux noeud non voisins, les messages doivent effectues des sauts via des noeuds intermediaires. 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 decouverte de route. C'est pourquoi, concevoir des algorithmes efficaces et economes en énergie est un probleme primordial dans les WSN.

### 3.1.3 Modèle énergétique

#### Energie d'un capteur

Dans [5], Dong présente les deux modèles de consommation énergétique communement utilisés.

**The Packet based model.** Nous utiliserons pour notre analyse le modele de connsomation d'énergie idéale suivant : Nous considérerons que chaque capteur  $i$  a une énergie initiale  $E_{init} = \beta$ . L'envoi de message est le seul facteur de perte d'énergie. L'énergie consommée lors de la reception de message, l'aquisition et traitement des informations sera considérée comme négligeable. 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 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  $E_{ij} = E(d_e(i, j))$ .

**The Time based model** Cependant, un autre modèle plus réaliste prend en compte l'énergie de réception des messages, de traitement ainsi que d'écoute inactive du réseau (mode veille). En effet, dans [9], Kasten souligne le fait que souvent, la réception, l'écoute et le traitement consomment en moyenne autant d'énergie que la transmission de message. Dans de nombreuses topologies, si la fréquence des broadcasts est faible, beaucoup de capteurs vont être à court d'énergie avant même d'avoir pu transmettre des messages. Dans beaucoup d'applications, la densité du réseau étant élevée, le coût propre aux transmissions de message est relativement faible tandis que le coût de réception des messages est élevé puisque chaque capteur traite les messages de son voisinage qui en l'occurrence est grand.

## 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 :

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

Le coût moyen d'un broadcast est  $C(1)$ .

Le coût moyen de  $k$  broadcasts est  $C(k)$

### 3.1.4 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 réseau et ce automatiquement. Deux choix sont possibles : soit le capteur est immédiatement opérationnel soit il nécessite une phase d'initialisation. Il existe donc deux types 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 cours de l'algorithme, chaque nœud 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 globales 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** : protocol distribué de broadcast nécessitant une connaissance quasi-globale du réseau.
3. **Quasi-local** : protocol distribué nécessitant une connaissance du réseau principalement locale et occasionnellement globale. (ex : Cluster networks : tandis que les groupes peuvent etres construits de maniere locale , des reactions en chaines peuvent arrivées).
4. **Local** : protocol distribué nécessitant une connaissance tres locale du réseau. Tout les algorithmes de 1 ou 2-voisinage appartiennent à cette catégorie.

### 3.1.5 Durée de vie

#### Durée de vie du réseau, problématique

Dans un réseau de capteur sans fil, la contrainte majeure est l'efficacité de l'algorithme utilisé en matière de consommation énergétique. En effet la principale caractéristique des capteurs est leur petite taille et leur micro-batterie. Dans la majeure partie des cas, remplacer les batteries est impossible. Cependant la durée de vie d'un WSN est difficile à définir et à mesurer. Il n'y a pas de définition absolue. Elle indique combien de temps le réseau sera 'efficace' par rapport à l'application donnée ( nombre de broadcast effectués, connectivité, pourcentage de noeuds vivants...) Dans la littérature, deux approches apparaissent clairement en matière de maximisation de la durée de vie des WSNs. Une approche indirecte consiste à minimiser la consommation d'énergie de façon locale tandis que l'autre a pour but de maximiser la directement la durée de vie du réseau de façon plus globale. Bien que l'approche indirecte peut améliorer la durée de vie du réseau, elle ne suffit pas à elle seule à appréhender le problème de la durée de vie. Parmi ces approches figurent par exemple le fait de maximiser le nombre de broadcasts effectués avant qu'un capteur ne meurt.

Dans [10], Liang prouve que le problème THE MINIMUM-ENERGY BROADCAST TREE PROBLEM est NP-complet par réduction à (3-CNF SAT) :

**Theoreme 1.** Soient un WSN  $G$  dans lequel chaque noeud a  $k$  rayons de transmission possibles, une source  $s$  et un entier positif  $w$ . Déterminer si il existe un arbre de broadcast de source  $s$  tel que la somme des coûts de transmission aux noeuds relais (qui ne sont pas des feuilles) soit inférieur à  $w$  est NP-Complet.

Ainsi, pour un broadcast donnée il n'existe pas d'algorithme centralisé polynomiale pour trouver l'arbre de diffusion optimale. Il est évident qu'il n'en existe pas de distribués. Dans [5], Dong prouve que le problème BROADCAST LIFETIME est NP-Complet par réduction de (3DM) :

**Theoreme 2.** Soient un WSN  $G$ , une source  $s$  et un entier positif  $k$ . Déterminer si  $G$  a assez d'énergie pour broadcaster  $k$  messages à partir de  $s$  est NP-Complet.

De façon formel, il est donc souvent impossible de calculer le nombre de broadcasts total réussis étant donné un réseau et un protocole (sauf quand le protocole est très simple cf 3.2.1). Nous analyserons donc *les performances moyennes* des algorithmes quand cela est possible. Sinon, les simulations nous permettront de mesurer les performances en fonction de différentes topologies.

#### Différentes définitions

Les définitions et critères de durée de vie d'un WSN sont tirés des articles [4],[3],[6].

**Définition 7.** Nous notons  $\lambda$ -broadcast un broadcast touchant  $\lambda$  pourcent des noeuds encore en vie. Nous écrirons broadcast pour parler de 1-broadcast.

**LIFETIME 1.** Nombre moyen de broadcast réussis avant qu'un capteur n'ai plus de batterie (TTFF).

**LIFETIME 2.** Nombre moyen de broadcast réussis avant que le réseau ne perde sa connectivité.

**LIFETIME 3.** Nombre moyen de broadcast reussis jusqu'à ce qu'il ne reste  $X$  pourcent de noeuds vivants.

**LIFETIME 4.** Le nombre

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 $i$
$E_{ij}$	Cout d'envoi d'un message de $i$ à $j$
$N(u)$	Degre 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 $u$
$N_G$	Degre de $G$
$d_G$	Distance de $G$
$d_e(G)$	Distance euclidienne de $G$
$diametre_G$	Diametre de $G$
$D_G$	Densité de $G$
$C(k)$	Cout de $k$ broadcasts

## 3.2 Analyse d'algorithmes existants

Les reseaux de capteurs sans fils constituent un domaine de recherche récent et actif. Une grande quantité d'articles ont été publiés cette dernière décennie. Nous n'aborderons que quelques uns.

### 3.2.1 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.

**Consommation et durée de vie :** Le cout global d'un broadcast est exactement le cout moyen puisque chaque noeud  $i$  envoie un unique message  $\langle M, \gamma \rangle$  par broadcast. Le cout global d'un broadcast est

$$C(1) = n \cdot E(\gamma) = n \cdot \gamma^\alpha + n \cdot c$$

Le cout global de  $k$  broadcast est

$$C(k) = k \cdot n \cdot E(\gamma) = k \cdot n \cdot \gamma^\alpha + k \cdot n \cdot c$$

La durée de vie du réseau est exactement de

$$LT(BlindFlooding) = \lfloor \frac{\beta}{n \cdot \gamma^\alpha + n \cdot c} \rfloor$$



**Probabilistic Flooding** Afin d'éviter le redondance et les collisions, une idée est que chaque noeud retransmet le message suivant une probabilité  $P$  lorsqu'il le reçoit pour la première fois. Si  $P = 1$  cela equivaut au blind flooding.

**Consommation et durée de vie** : Chaque noeud ayant une probabilité  $P$  de retransmettre le message  $\langle M, \gamma \rangle$ , en moyenne  $P \cdot n$  envoient ce message lors d'un broadcast. Le coût moyen d'un broadcast est

$$C(1) = P \cdot n \cdot E(\gamma)$$

Le coût global de  $k$  broadcast est

$$C(k) = P \cdot k \cdot n \cdot E(\gamma)$$

La durée de vie moyenne du réseau est

$$TTF(ProbabilisticFlooding) = \lfloor \frac{\beta}{P \cdot (n \cdot \gamma^\alpha + n \cdot c)} \rfloor$$

**Algorithme ABBA** 'Area-based beaconless reliable broadcasting in sensor networks'[11].

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.

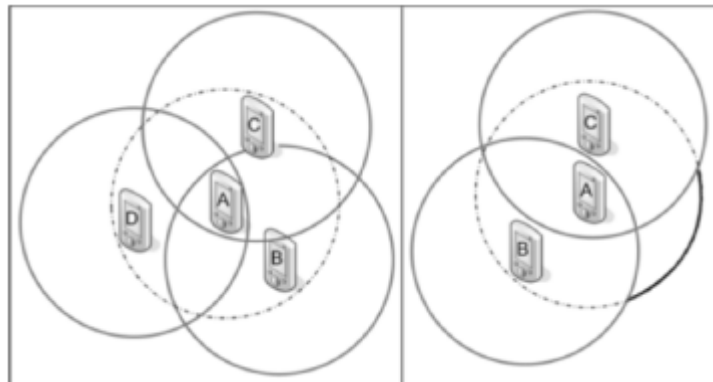


FIGURE 3.3 –

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.

### 3.2.2 Algorithmes globaux

**Algorithme BIP** Article [13]

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

---

**Algorithm 1** ABBA

---

**Require:** : Un noeud source  $s$ , un message  $M$   
 $s$  envoie  $\langle M, s, position_s, R \rangle$  ;  
Reception par  $u$  de  $\langle M, v, position_v, R \rangle$   
 $z_u \leftarrow ensemble_{vide}$  ;  
Calculer la zone  $z$  couverte par  $v$  lors de la transmission ;  
Déclancher le timer ;  
**repeat**  
  Attendre la reception d'un autre copie de  $M$  ou que le timer expire ;  
  **if** Une copie de  $M$  est reçue **then**  
    Mettre à jour  $z_u$  ;  
    reinitialiser le timer ;  
  **end if**  
**until** timer expire ;  
**if**  $\pi \not\subseteq z_u$  **then**  
  Retransmettre ;  
**end if**  
Ignorer les autres copies de  $M$  ;

---

et le nombre de messages circulant : un large rayon de transmission coute cher mais atteint beaucoup de noeuds ; un court rayon cout tres peu cher mais augmente le nombre de messages.

'Broadcast Incremental Protocol' et 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  $P'_{ij}$  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  $i$  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.

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. Cependant, contrairement à Prim, Bip exploite l'avantage notoire du mulicast dû aux transmissions radio. Un fois cet arbre construit, le broadcast se fait naturellement via celui-ci.

---

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

---

ENTREES  $G = (V, E)$  un graphe connexe,  $s$  une source, un cout de transmission  $P_{ij}$   
SORTIE Arbre BIP de racine  $s$   
 $B$  : ENSEMBLE des arêtes de l'arbre  
 $B \leftarrow arbre_{vide}$   
Marquer  $s$   
Créer 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  $B = (V, B)$

---

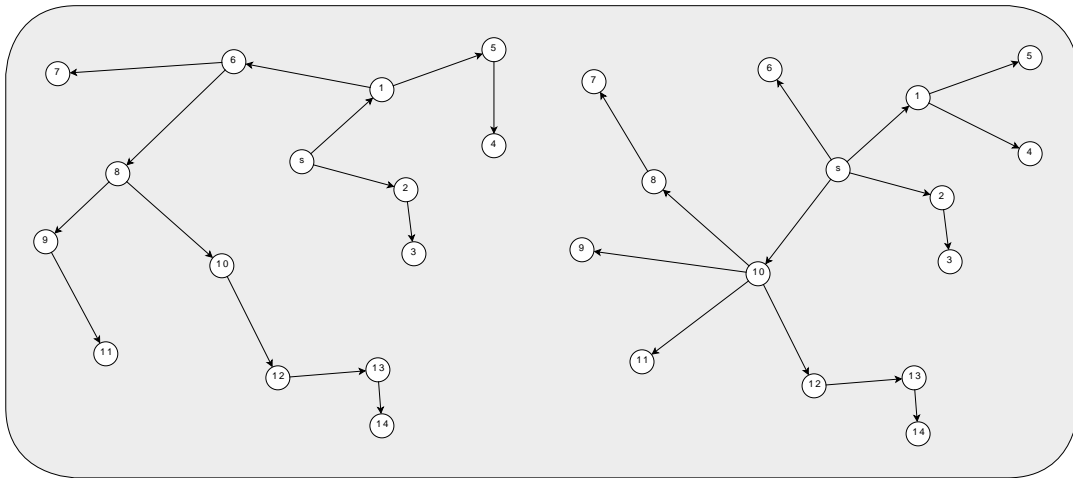


FIGURE 3.4 – Arbre Prim(gauche) ; Arbre Bip(droite)

---

**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**

---

### 3.2.3 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 1,2-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 [7] LBIP est l’application local de BIP : au lieu de construire l’arbre de diffusion BIP de façon centralisé, il le construit de façon locale. Chaque site lorsqu’il reçoit un message à retransmettre calcule son BIP-tree sur son 2-voisinage et diffuse le message par l’intermédiaire de celui-ci.

---

**Algorithm 5** LBIP

---

```
ENTREES  $G = (V, E)$  un graphe connexe,  $s$  une source, un message  $M$ 
SORTIE LBIP Broadcast
REQUIRE Connaissance du 2-voisinage
 $s$  calcule son arbre BIP( $N_2(s), s, E_{ij}$ ) et diffuse  $\langle M, s \rangle$  à ses fils
if  $u$  reçoit  $\langle M, v \rangle$  : then
  if Le paquet contient des instructions pour  $u$  then
     $u$  construit BIP( $N_2(u), u, M$ ) et retransmet le message à ses fils
  end if
end if
```

---

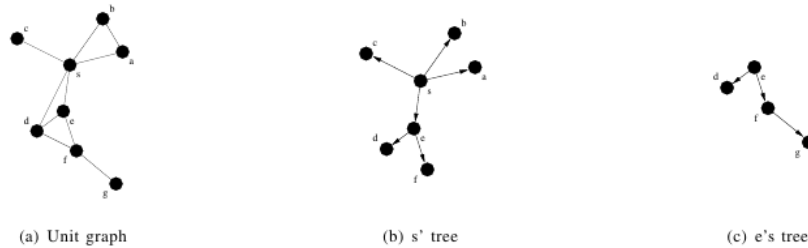


Fig. 1. Applying LBIP.

FIGURE 3.5 –

**Algorithme DLBIP** ‘Dynamic Localized Broadcast Incremental Power Protocol for Wireless Ad Hoc Networks’ : article [2]. DLBIP est une amélioration de LBIP. Le principe est de répartir 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’arbre de diffusion est recalculé sur le même 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 protocole 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. DLBIP actualise dynamiquement ces coûts  $P'_{ij}$  à chaque broadcast :

$$\forall i, j \in V, P'_{ij} = \frac{E_{ij}}{E_i}$$

ou  $E_{ij}$  est le coût réel de transmission et  $E_i$  l’énergie restante de  $i$ .

---

**Algorithm 6** DLBIP

---

ENTREES  $G = (V, E)$  un graphe connexe,  $s$  une source, un message  $M$   
SORTIE DLBIP Broadcast  
REQUIRE Connaissance du 2-voisinage  
 $s$  calcul son BIP( $N_2(s), s, P'_{ij}$ ) et diffuse  $\langle M, s \rangle$  à ses fils  
**if**  $u$  recoit  $\langle M, v \rangle$  : **then**  
    **if** Le paquet contient des instruction pour  $u$  **then**  
         $u$  construit BIP( $N_2(u), u, P'_{ij}$ ) et retransmet le message à ses fils  
    **end if**  
**end if**

---

**Algorithme RRS** [1]**Algorithmes hexagonaux****Rayon Optimal**

**Theoreme 3.** Sans chevauchement et sans vide, un plan ne peut etre découpé de facon uniforme que par des polygones reguliers de type triangle, carrés ou hexagones.

*Démonstration.* Soit  $m$  le nombre de sommet d'un  $m$ -polygone et  $n$  le nombre de  $m$ -polygones nécessaire pour couvrir  $2\pi$  degrés Nous avons :

$$\frac{(m-2)n\pi}{m} = 2\pi \Leftrightarrow (m-2)(n-2) = 4$$

Comme  $n$  et  $m$  sont des entiers, les seuls solutions sont :

$$(m-2; n-2) \in \{(1, 4), (2, 2), (4, 1)\} \Leftrightarrow m \in \{3, 4, 6\}$$

□

Nous admettons :

**Theoreme 4.** Le quadrillage en hexagone est celui offrant le moins de chevauchement du point de vu WSN.

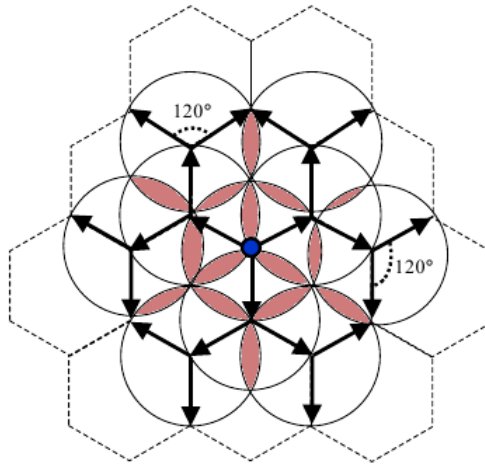


FIGURE 3.6 – Chevauchement d'un maillage hexagonale

Soit  $P$  un plan sur lequel  $n$  capteurs sont placés. Chaque capteur peut émettre des message avec un rayon compris entre 0 et  $\gamma$ . Dans ce protocole, tout les noeuds auront un meme rayon d'émission fixe  $R$ . Etant donnée une source  $s$  et un message  $M$  à broadcaster, nous devons placer les noeuds relais de facon a minimiser leur nombre  $m$ . Biensur leur nombre dépend directement du rayon  $R$ .

Dans un reseau hexagonale, nous avons les resultats suivants : Soit  $S$  l'aire du plan rectangulaire  $P$ . Connaissant  $R$ , nous pouvons calculer facilement le nombre de sommet pour couvrir le plan. Soit  $h$  le nombre d'hexagones couvrant  $P$  : Knowing  $r$  is the exact distance between two emitting nodes, we can easily compute the necessary quantity of them to cover the entire area. To do this, we just have to find how many hexagons, denoted by  $h$ , fit on our area of surface  $S$  :

$$h \simeq \frac{Surface(P)}{Surface(hexagone)} = \frac{2S}{3R^2\sqrt{3}}$$

Comme il faut deux noeuds par hexagone,

$$n = 2h = \frac{4S}{3R^2\sqrt{3}} = \frac{k}{R^2}$$

où  $k = \frac{4S}{3\sqrt{3}}$

Ainsi, le cout d'un broadcast vaut :

$$C(1)(R) = n \cdot E(R) = \frac{k}{R^2} \cdot E(R) = \frac{k}{R^2} \cdot (R^\alpha + c)$$

Nous cherchons le rayon optimal minimisant  $C(1)(R)$ . Comme  $\alpha \geq 2$ ,  $c \geq 0$  et  $R > 0$ , nous avons 4 cas possibles :

- $\alpha = 2$ ,  $c = 0$  : nous avons  $C(1)(R) = k$ .  $R_{opt}$  ne depend pas de  $r$ .
- $\alpha = 2$ ,  $c \neq 0$  : nous avons  $C(1)(R) = k(1 + cR^{-2})$ . Au plus le rayon doit etre le plus gran possible :  $R_{opt} = \gamma$
- $\alpha > 2$ ,  $c = 0$  : nous avons  $C(1)(R) = kR^{\alpha-2}$ .  $R_{opt} = 0$
- $\alpha > 2$ ,  $c \neq 0$  : nous avons  $C(1)(R) = k(R^{\alpha-2} + cR^{-2})$ .  
En derivant, nous obtenons :  $C'(1)(R) = k((\alpha-2)R^{\alpha-3} - 2cR^{-3})$ .  
D'ou :

$$R_{opt} = \sqrt[\alpha]{\frac{2c}{\alpha-2}}$$

Biensur, en pratique, il est impossible sur un topologie quelconques d'extraire un tel maillage hexagonale. Cependant, l'idée des algorithmes de broadcast est de choisir les noeuds relais de façon à ce que leur ensemble se rapproche au maximum d'un tel maillage.

**TR-LBOP** 'Target Transmission Radius over LMST for Energy-Efficient Broadcast Protocol in Ad Hoc Networks' :[7].

**DMTB** 'Distributed Mobility Transparent Broadcast in Mobile Ad Hoc Networks' : [?].

### 3.2.4 Elements de classification

Les éléments de classification cités ci dessous sont inspirés des articles [12],[8],[14]

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. **Deterministe vs probabiliste** : Un protocol de broadcast peut utiliser des fonctions probabilistes pour prendre certaines descisions de maniere aléatoire.
3. **Rayon d'émission** : rayon fixe ou variable
4. Avec ou sans balisage :
5. hello msg
6. Principe

### 3.2.5 Synthèse

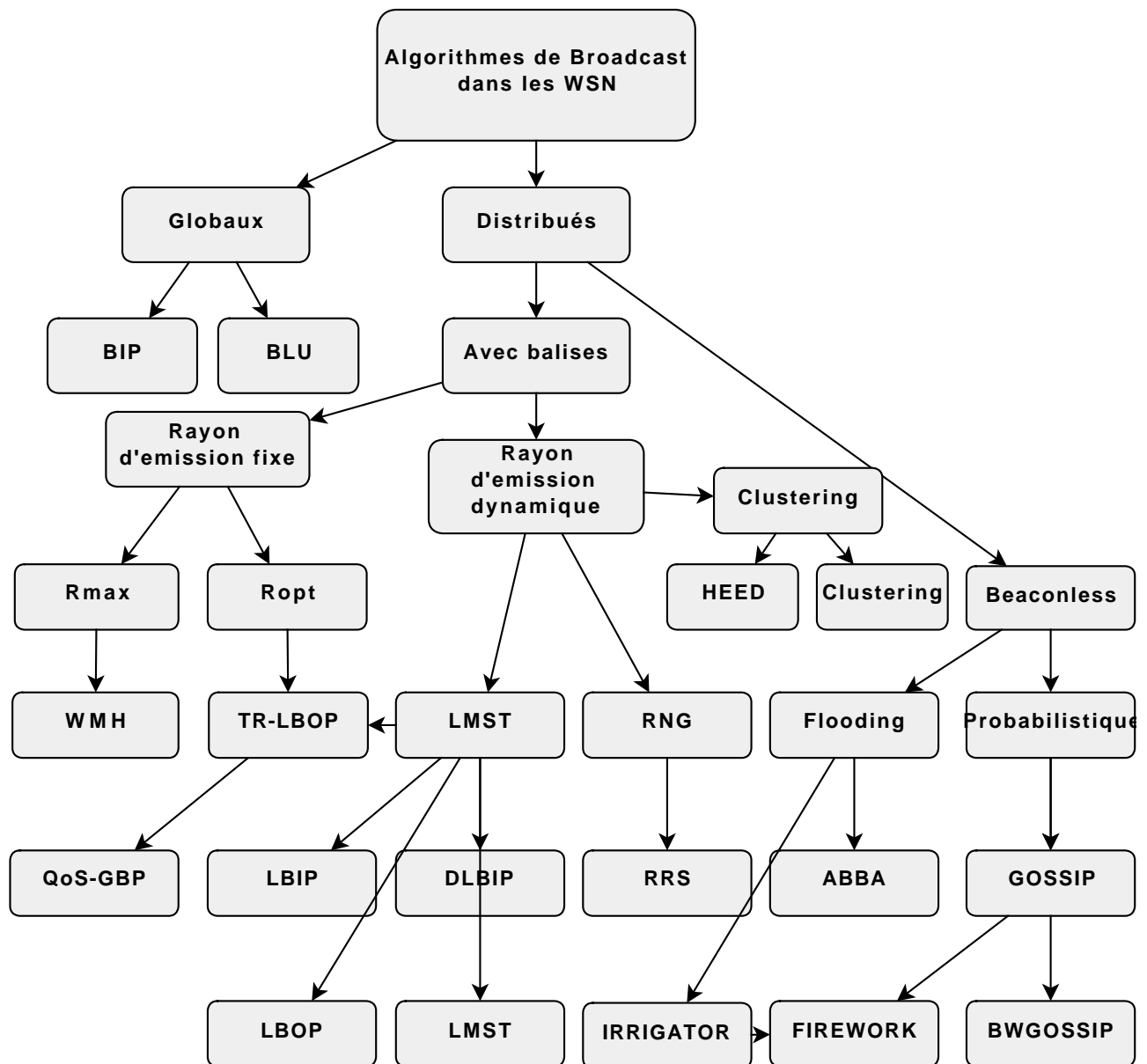


FIGURE 3.7 – Graphe de synthèse

## 3.3 Nos idées

### 3.3.1 Application du rayon optimal à DLBIP

---

**Algorithm 7** DLBIP avec rayon optimal

---

ENTREES  $G = (V, E)$  un graphe connexe,  $s$  une source, un message  $M$

SORTIE DLBIP Broadcast

REQUIRE Connaissance du 2-voisinage

$$R_{opt} = \sqrt[\alpha]{\frac{2c}{\alpha-2}}$$

$s$  calcul son BIP( $N_2(s), s, P'_{ij}$ ) et diffuse  $\langle M, s \rangle$  à ses fils

**if**  $u$  recoit  $\langle M, v \rangle$  : **then**

**if** Le paquet contient des instruction pour  $u$  **then**

$u$  construit BIP( $N_2(u), u, P'_{ij}$ ) est retransmet le message à ses fils avec pour rayon :

        -  $R_{opt}$  si  $\max_{v \in N_u(1) \cap BIP(N_2(u), u, P'_{ij})} (d_e(u, v)) < R_{opt}$

        -  $\max_{v \in N_u(1) \cap BIP(N_2(u), u, P'_{ij})} (d_e(u, v))$  sinon.

**end if**

**end if**

---

### 3.3.2



## Chapitre 4

### Simulations et résultats

Chapitre 5

Conclusion

# Bibliographie

- [1] 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.
- [2] 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.
- [3] J. Champ, C. Saad, and A.E. Baert. Lifetime in wireless sensor networks. In *Complex, Intelligent and Software Intensive Systems, 2009. CISIS'09. International Conference on*, pages 293–298. IEEE, 2009.
- [4] Isabel Dietrich and Falko Dressler. On the lifetime of wireless sensor networks. *TOSN*, 5(1), 2009.
- [5] Q. Dong. Maximizing system lifetime in wireless sensor networks. *IPSN '05 : Proceedings of the 4th international symposium on Information processing in sensor networks*, page 3, 2005.
- [6] Abdelrahman Elleithy and Gonhsin Liu. A simulation model for the lifetime of wireless sensor networks. *CoRR*, abs/1201.2237, 2012.
- [7] 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.
- [8] 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.
- [9] O. Kasten. Energy consumption, 2001.
- [10] W. Liang. Constructing minimum-energy broadcast trees in wireless ad hoc networks. In *Proceedings of the 3rd ACM international symposium on Mobile ad hoc networking & computing*, pages 112–122. ACM, 2002.
- [11] 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.
- [12] I. Stojmenovic and J. Wu. Broadcasting and activity scheduling in ad hoc networks. *Mobile Ad Hoc Networking*, pages 205–229, 2004.
- [13] 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.
- [14] 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.
- [15] Hua Yang, Fengji Ye, and Biplab Sikdar. Distributed Mobility Transparent Broadcast in Mobile Ad Hoc Networks. In *GLOBECOM*. IEEE, 2006.