

République Algérienne Démocratique et Populaire  
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique  
Université ABDERRAHMANE MIRA de Béjaïa  
Faculté des Sciences Exactes  
Département D'Informatique



# Mémoire de fin de cycle

En vue de l'obtention du diplôme de master professionnel en informatique

**Option** : Administration et Sécurité des Réseaux

## Thème

---

ÉTUDE DES ALGORITHMES DE LOCALISATION DANS  
LES RÉSEAUX DE CAPTEURS SANS FIL

---

**Réalisé par :**

M<sup>r</sup> BELDJOUDI Fawzi

M<sup>r</sup> BOUKHANOUF Imad

**Devant le jury composé de :**

M <sup>r</sup> MOKATEFI Mohand	MAA	Univ. De Béjaïa	Président
M <sup>me</sup> HOUHA Amel	MAA	Univ. De Béjaïa	Examinatrice
M <sup>r</sup> BOUTEBEL Abdeldjalil	Doctorant	Univ. De Béjaïa	Examineur
M <sup>r</sup> LARBI Ali	MCB	Univ. De Béjaïa	Promoteur

Année Universitaire : 2017/2018

---

# Remerciement

---

En premier lieu, nous tenons à remercier le bon Dieu de nous avoir donné la santé, le courage et la volonté durant la préparation de ce modeste travail.

Nous tenons à remercier vivement Mr LARBI Ali, pour nous avoir honorés par son encadrement, pour sa disponibilité, ses orientations, ses précieux conseils et ses encouragements qui nous ont permis de mener à bien ce travail.

Nous poursuivons ces remerciements en saluant vivement les membres du jury pour l'honneur qu'ils nous ont fait en acceptant de juger ce travail.

Nous n'omettrons jamais d'exprimer toute notre gratitude à tous les membres du département d'informatique de l'université de Béjaïa, que ce soit enseignants ou cadres administratifs, qui de près ou de loin n'ont épargné aucun effort pour que notre formation et nos travaux se termine dans de bonnes conditions.

Un merci particulier à nos parents, pour leur amour, leurs sacrifices et leurs patiences.

Un énorme merci à nos familles et amis pour leurs éternel soutien et la confiance qu'ils ont en nos capacité

---

## Dédicaces

---

*Je dédie ce modeste travail à :*

- ✓ *Les êtres les plus chers à moi, mes parents qui m'ont toujours soutenu et qui croient toujours en moi : toutes les lettres ne sauraient trouver les mots pour vous exprimer mon amour, ma gratitude et reconnaissance . . . ;*
- ✓ *Mes chers frères Nacim, Youcef, Rabia, Daoud et leurs épouses ;*
- ✓ *Mes chers grands parents ;*
- ✓ *A toute ma famille ;*
- ✓ *A tous mes ami(e)s Mahdi, Abdel Alli(kholeyo), Tacfa, Adel... ;*
- ✓ *Mon promoteur Mr " LARBI Ali " qui nous a beaucoup aidé à la réalisation de ce travail ;*
- ✓ *Mon cher binôme Imad.*

*Fawzi BELDJOURI*

---

# Dédicaces

---

*À mes parents. Tout au long de mon cursus, ils m'ont toujours soutenu, encouragé et aidé. Ils ont su me donner toutes les chances pour réussir.*

*À ma famille et en particulier mes frères " Walid, Laid" et sœurs" Samia, Kenza, Amina" pour m'avoir fait partager leur joie de vivre et m'avoir ainsi soutenu dans mes efforts.*

*À toute ma famille : grand-père, grand-mère, mes cousins et cousines.*

*À mon binôme Zaki avec lui j'ai partagé de belles années d'études.*

*À mon encadreur Monsieur, LARBI Ali pour avoir accepté de nous encadrer, et qui a toujours été à notre écoute, et disponible tout au long de notre travail, malgré ses charges ainsi que, pour son aide et le temps qu'il a bien voulu nous consacré.*

*À tous mes professeurs ainsi que tous les étudiants de la promotion ASR, 2018.*

*À tous mes meilleurs amis " Zami Morad, Riad et sa petite fille celine, Zitouni, Larbi, Zoubir, Botabout, Kholeyo,...", dont la liste est longue.*

*À tous ceux que j'aime.*

*Imad BOUKHANOUF*

---

# Table des matières

---

<b>Table des matières</b> .....	<b>i</b>
<b>Liste des figures</b> .....	<b>iv</b>
<b>Liste des tableaux</b> .....	<b>v</b>
<b>Introduction générale</b> .....	<b>1</b>
<b>Chapitre 1</b>	
<b>1 Généralité sur les RCSFs</b> .....	<b>1</b>
1.1 Introduction .....	4
1.2 Le nœud capteur .....	4
1.2.1 Définition d'un nœud capteur .....	4
1.2.2 Architecture d'un nœud capteur .....	5
1.2.2.1 Architecture matérielle .....	5
1.2.2.2 Architecture logicielle .....	6
1.2.3 Systèmes d'exploitation pour capteurs .....	6
1.2.3.1 TinyOS .....	6
1.2.3.2 Contiki et Rime .....	7
1.2.3.3 MANTIS OS .....	8
1.2.3.4 SOS .....	8
1.3 Les réseaux de capteurs sans fil .....	9
1.3.1 Fonctionnement des RCSF .....	9
1.3.2 Architecture des RCSF .....	10
1.3.2.1 Architecture à plat .....	10
1.3.2.2 Architecture hiérarchique .....	11
1.4 Applications des RCSF .....	12
1.6 Contraintes de conception des RCSF .....	14
1.7 La synchronisation dans les réseaux de capteurs sans fil .....	15
1.7.1 Les méthodes de synchronisations .....	16
1.8 Conclusion .....	18
<b>Chapitre 2</b>	
<b>2 La localisation dans les réseaux de capteurs sans fil</b> .....	<b>19</b>
2.1 Introduction .....	19

---

2.2 Définition d'un système de localisation .....	19
2.3 Objectifs de localisation .....	19
2.4 Propriétés d'un système de localisation .....	20
2.5 Critères d'évaluations d'un système de la localisation .....	20
2.5.1 Précision de la localisation .....	20
2.5.2 Contraintes de ressources .....	21
2.5.3 Contraintes énergétiques.....	21
2.5.4 Techniques de mesure .....	21
2.6 Composition d'un système de localisation .....	21
2.6.1 Estimation de la distance .....	21
2.6.2 Calcul de position .....	22
2.6.3 Algorithme de localisation.....	22
2.7 Méthodes de localisation .....	22
2.7.1 Méthodes de localisation physique.....	22
2.7.1.1 Trilatération.....	23
2.7.1.2 Triangulation.....	23
2.7.1.3 Multilatération.....	24
2.7.1.4. Calcules des distances par calcul de temps de transit du signal.....	25
2.7.1.5 Calcules des distances par mesure de puissance du signal .....	28
2.7.2 Méthodes de localisation logique .....	30
2.8 Algorithmes de localisation.....	31
2.8.1 Algorithmes à ancrs statiques et nœuds statiques .....	31
2.8.2 Algorithmes à ancrs statiques et nœuds mobiles .....	31
2.8.3 Algorithmes à ancrs mobiles et nœuds statiques .....	31
2.8.4 Les algorithmes à ancrs mobiles et nœuds mobiles .....	32
2.9 Forme d'implémentation .....	32
2.9.1 Les méthodes centralisées .....	32
2.9.2 Les méthodes distribuées.....	33
2.10 Conclusion.....	33
<b>Chapitre 3</b>	
<b>3 Ancre mobile avec une trajectoire en Scan pour la localisation dans les RCSFs.....</b>	<b>34</b>
3.1 Introduction .....	34
3.2 Quelques définitions et terminologies .....	34

---

3.2.1 Ancre/Beacon .....	34
3.2.2 Ancre mobile .....	35
3.2.3 Trajectoire.....	36
3.3 Problématique.....	36
3.4 Élément de solution.....	37
3.5 Quelques trajectoires existantes dans la littérature.....	37
3.6 La Trajectoire .....	37
3.7 Les paquets et les points de diffusion.....	40
3.8 Architecture et fonctionnement .....	40
3.8.1 Le processus de localisation .....	40
3.8.2 Algorithme de localisation.....	42
3.9 Conclusion.....	43
<b>Chapitre 4</b>	
<b>4 Simulation et évaluations des performances .....</b>	<b>44</b>
4.1 Introduction .....	44
4.2 Environnement de la simulation.....	44
4.2.1 Choix du langage de programmation.....	44
4.2.2 Variables descriptives du système .....	45
4.2.2.1 Coordonnées d'un nœud capteur .....	45
4.2.2.2 Trajectoire d'une ancre mobile .....	45
4.2.2.3 Distance entre deux capteurs.....	45
4.3 Paramètres de simulation .....	45
4.4 Résultats de simulation.....	46
4.5 Conclusion.....	49
<b>Conclusion générale et perspectives .....</b>	<b>50</b>
Bibliographie .....	51

---

# Liste des figures

---

1.1 Architecture materielle d'un capteur .....	5
1.2 Exemple d'un RCSF.....	9
1.3 Exemple d'un RCSF fonctionnant selon un mode d'architecture à plat .....	11
1.4 Exemple d'un RCSF fonctionnant selon une architecture hiérarchique en clusters .....	12
1.5 Quelques exemples d'illustrations d'applications des RCSF.....	13
1.6 Synchronisation de type maître-esclave et point-à-point .....	16
2.1 Composition d'un système de localisation.....	22
2.2 Trilatération.....	23
2.3 Triangulation .....	24
2.4 Multilatération.....	25
2.5 Localisation par trilatération .....	26
2.6 Principe de ToA .....	26
2.7 Principe d'AoA. ....	28
2.8 Principe de l'utilisation des RSSI.....	29
3.1 Environnement de la solution de localisation proposée dans [31] .....	36
3.2 Trois différentes trajectoires.....	37
3.3 Trajectoire en scan .....	38
3.4 Trajectoire en double scan.....	39
3.5 Trajectoire en Hilbert .....	39
3.4 Architecture du réseau.....	40
3.5 Processus de la localisation en utilisant une ancre mobile.....	41
4.1 Environnement de la solution de localisation proposée .....	46
4.2 Nombre de nœuds localisés en fonction de nombre de nœuds sur la surface .....	47
4.3 Nombre de nœuds localisés en fonction de nombre de diffusion .....	47
4.4 Nombre de messages envoyés en fonction de nombre de nœuds .....	48
4.5 Nombre de messages reçus en fonction de nombre de nœuds .....	49

---

# Liste des tableaux

---

2.1 Liste des caractéristiques essentielles des méthodes décrites dans les paragraphes précédents. ....	30
4.1 Les paramètres de simulation .....	45

---

# Liste des abréviations

---

<b>ADC</b>	Anolog to Digital Converter
<b>AOA</b>	Angle of Arrival
<b>API</b>	Application Programming Interface
<b>CH</b>	Cluster Head
<b>GPS</b>	Global Positioning System
<b>GSM</b>	Global System for Mobile Communications
<b>ICMP</b>	Internet Control Message Protocol
<b>IEEE</b>	Institut of Electrical and Electronics Engineers
<b>IETF</b>	Internet Engineering Task Force
<b>IP</b>	Internet Protocol
<b>IPV4</b>	Internet Protocol version 4
<b>IPV6</b>	Internet Protocol version 6
<b>LTS</b>	Light weight Tree based Synchronization
<b>MAC</b>	Media Access Control
<b>MANTIS</b>	Multimod AI Networks of In-situ micro Sensor
<b>NTS</b>	Network wide Time Synchronization
<b>OMNeT++</b>	Objective Modular Network Test-bed in C++
<b>OSI</b>	Open Systems Interconnect
<b>RBS</b>	Referece Broadcast Synchronization
<b>RCSFs</b>	Réseaux de capteurs sans fils
<b>RSSI</b>	Signifie Received Signal Strength Indicator
<b>SOS</b>	Sensor Operating System
<b>TCP</b>	Tronsport control protocol
<b>TDOA</b>	Time Difference of Arrival
<b>Tinyos</b>	Tiny Operating System
<b>TOA</b>	Time of Arrival
<b>TS</b>	Time Synconization
<b>UDP</b>	User Datagram Protocol
<b>WPAN</b>	Wireless Personal Area Network
<b>WSN</b>	Wireless Sensor Networks

---

# Introduction générale

---

Les développements technologiques récents, en termes de miniaturisation des machines et des supports de communications conduit à l'émergence de nouvelle génération de micro-capteurs (miniaturisation, capacité de traitement (processeur), communication sans fil, diversité des capteurs (optiques, thermiques, multimédias, etc.), faible coût, etc.). En effet, grâce aux récents progrès des technologies sans fil, les capteurs peuvent communiquer non seulement de proche en proche mais aussi d'acheminer de l'information à tous les nœuds connectés au réseau.

Dans les domaines de recherche, d'analyse, d'observation et même de contrôle des phénomènes physiques, la phase de collection d'informations réelles tels que la température, la pression ou encore la luminosité est très importante. Cette tâche est déléguée aux capteurs dont la fonction est l'acquisition de l'information sur les phénomènes observés.

Les réseaux de capteurs sont considérés comme l'une des technologies clefs de l'avenir et ce en raison de l'incroyable potentiel applicatif qu'elle renferme. Le réseau de capteurs sans fil est composé d'un grand nombre de capteurs qui sont déployés de manière aléatoire à travers une zone géographique, appelée zone d'intérêt et travaillent de façon autonome et complètement transparente pour l'utilisateur. Les données récoltées par les capteurs sont acheminées grâce à des communications sans fil.

Les caractéristiques de RCSF (Réseau de Capteur Sans Fil) ont ouvert beaucoup de perspectives applicatives très larges et très variées pour ces réseaux dans de nombreux domaines, militaire, médical, domotiques, environnementales, etc.

Le domaine des réseaux de capteurs reste un « chantier » ouvert aux chercheurs de différentes communautés, sur tous les niveaux : initialisation, auto-configuration, localisation, couverture, déploiement, communication, topologie dynamique, collecte des données, requêtes et traitement, etc. Nous sommes intéressés dans ce mémoire au problème de la localisation.

Le déploiement d'un grand nombre de capteurs est généralement aléatoire, en raison soit de l'hostilité de la zone à surveiller, soit de son immensité ce qui pose **le problème de localisation des capteurs**. La localisation consiste à attribuer une position géographique aux capteurs. Elle est donc nécessaire non seulement au fonctionnement du réseau, mais également à l'exploitation des données récoltées (« où » est la question qui suit immédiatement l'avènement d'un évènement dans la zone surveillée). Il est donc nécessaire de localiser, avec la meilleure précision possible, tous les nœuds du réseau. Cette problématique, malgré les nombreux travaux de recherche qui s'y étaient attachés, reste une problématique ouverte. Cependant l'utilisation des solutions matérielles GPS (Global Positioning System.) pour la localisation est très chère. Pour pallier ce problème plusieurs algorithmes de localisation ont été proposés [1].

En effet l'utilisation d'une ancre mobile est l'une des solutions. Cette approche est très prometteuse vu que l'ancre mobile est équipée d'un GPS et n'a pas de contrainte d'énergie. Cette ancre mobile peut être un robot mobile, un opérateur humain ... En outre, le processus de localisation dans l'approche de l'ancre mobile dépend de deux principaux enjeux. Le premier enjeu est le choix de la trajectoire de l'ancre mobile. Le deuxième est le choix de l'algorithme de localisation. Dans ce mémoire, nous étudions une méthode de localisation basée sur une ancre mobile avec la Trilatération. Nous examinons deux différentes trajectoires de l'ancre mobile, Scan et Spiral. Afin d'analyser cet algorithme selon le nombre de nœuds localisés, nous avons utilisé un simulateur MATLAB pour notre simulation.

Pour mener à bien notre travail, nous l'avons organisé en quatre chapitres comme suit :

**Chapitre 1 :** Généralités sur les réseaux de capteurs sans fil : Donne une vue générale sur les réseaux de capteurs sans fil. Nous présentons les différents concepts tels que l'architecture, fonctionnement, domaines d'applications, et caractéristiques.

**Chapitre 2 :** La localisation dans les réseaux de capteurs sans fil : Nous avons présenté, une vue globale sur les différents points qui régissent les systèmes de localisation dans les RSCFs, les propriétés d'un système de localisation, ainsi que sa composition, et les méthodes de localisation physique et logique, et nous avons présenté et décrit de façon détaillée les algorithmes de localisation.

**Chapitre 3 :** Ancre mobile avec une trajectoire en Scan pour la localisation dans les RCSFs : Ce chapitre est consacré à notre amélioration pour la localisation des capteurs dans les RCSFs.

**Chapitre 4 :** Simulation et évaluation de performance : Dans ce chapitre, nous avons simulé le fonctionnement de notre algorithme afin d'extraire et évaluer les paramètres de performances.

En fin de ce mémoire, une conclusion est donnée pour résumer les apports essentiels de notre travail.

# Chapitre 1

# Généralités sur les réseaux de capteurs sans fil

## 1.1 Introduction

Les nœuds capteurs sont de très petits composants qui peuvent communiquer entre eux, ces composants coopèrent entre eux pour former une infrastructure de communication appelée réseau de capteurs. La position de ces nœuds n'est pas obligatoirement prédéterminée. Ils peuvent être aléatoirement dispersés dans une zone géographique. Ils sont utilisés essentiellement pour recueillir, traiter et acheminer les données environnementales de la région surveillée d'une manière autonome, vers des stations de collecte appelées nœuds, puits ou stations de base.

Dans le présent chapitre, nous ferons un aperçu de l'état de l'art sur les RCSF (réseaux de capteurs sans fil), en décrivant leur fonctionnement, leurs architectures, leurs différents domaines d'application ainsi que leurs contraintes de conception.

## 1.2 Le nœud capteur

### 1.2.1 Définition du nœud capteur

Les capteurs sont des dispositifs miniatures, multifonctionnels, capables de générer et de mesurer des données relatives à leur environnement physique (telles que température, vibrations et pression). Ces dispositifs embarquent un système de communication afin d'échanger des données formant ainsi un réseau implicite [2].

## 1.2.2 Architecture d'un nœud capteur

### 1.2.2.1 Architecture matérielle

L'architecture des capteurs se compose d'un module central communiquant avec les différentes interfaces d'entrée/sortie, de communication et d'alimentation. Le nœud d'un réseau de capteurs est formé des modules illustrés dans la figure suivante [2] [3] :

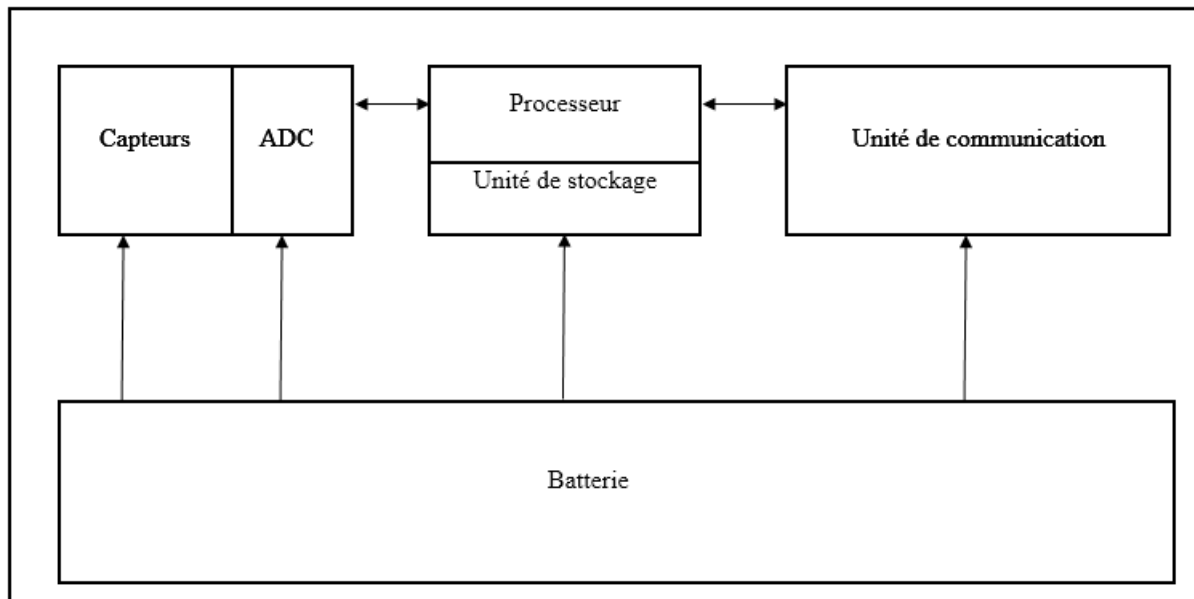


Figure 1.1 : Architecture matérielle d'un nœud capteur.

- **Unité de détection (Sensing unit)** : C'est un capteur qui relie le nœud au monde extérieur. Elle est composée de deux sous-unités, les capteurs et un ADC (Analog to Digital Converter) qui permet de convertir le signal produit par le capteur, en un signal numérique compréhensible par l'unité de traitement ;
- **Unité de traitement (MicroController Unit)** : Également appelée unité de calcul, dotée d'un processeur et d'une mémoire intégrant un système d'exploitation spécifique (TinyOS). Elle permet au nœud la gestion des procédures de collaboration avec les autres nœuds dans le but d'accomplir la tâche demandée. Il s'agit d'une carte physique (Mote) qui implémente le système d'exploitation et qui stocke le processeur et la mémoire ;
- **Unité de communication (Transceiver unit)** : Elle s'agit d'un système radio à courte portée, composé d'un émetteur/récepteur pouvant fonctionner en quatre modes : transmission, réception, libre et veille ;

- **Unité d'énergie (Power unit)** : Elle constitue la source d'énergie autonome du nœud capteur telle qu'une batterie ou un dispositif qui permet de la recharger à partir de l'énergie ambiante de l'environnement (convertissant l'énergie lumineuse en courant électrique) ;
- **Système de localisation (Location Finding System)** : Il fournit des informations sur la localisation dépendant du domaine d'application. Un capteur peut contenir des modules supplémentaires, tels qu'un système de localisation (GPS) ;
- **Mobilisateur (mobilizer)** : Il est appelé ainsi si le nœud capteur doit se déplacer pour satisfaire une requête à traiter.

### 1.2.2.2 Architecture logicielle

Le système d'exploitation conçu spécialement pour les capteurs sans fil est un système *Open Source TinyOS*. Il propose à l'utilisateur une gestion très précise de la consommation du capteur et permet de mieux s'adapter à la nature aléatoire de la communication sans fil entre interfaces physiques. Ainsi, lorsqu'aucune tâche n'est active, le capteur se met automatiquement en veille. Système, bibliothèques et applications de TinyOS ont été programmés en langage *NesC*, conçu spécialement pour capteurs sans fil et répond, donc, leurs exigences en terme de ressources. Il utilise un fonctionnement évènementiel différent du fonctionnement dit temporel où les actions du système sont gérées par une horloge. Plusieurs systèmes d'exploitation ont été développés pour répondre aux contraintes particulières des réseaux de capteurs sans fil [2].

## 1.2.3 Systèmes d'exploitation pour capteurs

Beaucoup de systèmes d'exploitation pour capteurs, plus généralement pour les systèmes embarqués, ont été mis en œuvre et implémentés dans les capteurs sans fil. Chacun de ces systèmes offre ses propres caractéristiques et fonctionnalités. Nous présentons dans les sections suivantes. Certains d'entre eux sont, actuellement, parmi les plus implémentés dans les nœuds capteurs sans fil.

### 1.2.3.1 TinyOS

Le système d'exploitation TinyOS (Tiny Operating System) [4], système intégré, a été développé par des chercheurs de l'Université de Berkeley. Il est un système d'exploitation intégré, modulaire et open source pour des nœuds capteurs sans fil. Il est transcrit en NesC. TinyOS est

conçu pour les systèmes embarqués et les systèmes à faibles ressources tels que les capteurs sans fil. Un programme s'exécutant sur TinyOS est constitué d'une sélection de composants systèmes et de composants développés spécifiquement pour l'application à laquelle il sera destiné, par exemple, l'acquisition de température, du taux d'humidité, de la pression, etc. TinyOS est basé sur les caractéristiques suivantes :

- *Évènementiel* : Fonctionnement basé sur la gestion d'évènements déclenchés par l'activation de tâches spécifiques ou de leur interruption ;
- *Non préemptif* : TinyOS ne gère pas le mécanisme de préemption entre les tâches en cours mais donne la priorité aux interruptions matérielles ;
- *Non temps réel* : TinyOS est un système à temps réel relâché, c'est-à-dire qu'il ne gère pas de niveau de priorité entre les différentes tâches, permettant ainsi de réduire les temps de traitements ;
- *Faible consommation énergétique* : TinyOS est conçu pour optimiser la consommation énergétique des nœuds capteurs. Ainsi, la consommation d'énergie de ces derniers est relativement faible.

### 1.2.3.2 Contiki et Rime

Contiki est un système d'exploitation *Open Source* [5], modulaire, léger et flexible, conçu pour capteurs miniatures et développé par des chercheurs suédois. Il est destiné à être embarqué dans des capteurs miniatures ayant généralement des ressources limitées (énergie, etc.).

Les principaux avantages sont surtout sa flexibilité, sa portabilité et le fait qu'il supporte des protocoles comme IPv6 et 6LoWPAN. Contiki implémente deux mécanismes de communication :

- *La couche Rime* : Elle fournit à la couche applicative des instructions de communication qui permettent des connexions avec les capteurs voisins. Les applications ou protocoles exécutés au-dessus de la couche Rime peuvent utiliser une ou plusieurs instructions de communication fournies par la couche Rime. Celle-ci peut être associée au mécanisme Chameleon afin de s'adapter aux protocoles de la couche MAC. Chameleon gère création, lecture et transformation des en-têtes des protocoles de la couche liaison des données du modèle OSI et communique avec la couche Rime en associant des attributs aux paquets ;

- *La couche uIP* : Elle implémente les protocoles uIPv4 et uIPv6. Cette couche supporte également les protocoles IP, TCP, UDP et ICMP. Notons qu'uIPv6 est la première implémentation d'IPv6 pour capteurs miniatures. Contiki implémente le protocole 6LoW-PAN prévu pour les communications radio utilisant la norme IEEE 802.15.4. En effet, lors de communications radio via la norme IEEE 802.15.4, la taille d'un paquet se limite à 127 octets, ce qui est insuffisant pour transmettre un paquet IPv6 dont la taille maximale est de 1280 octets. C'est dans ce cadre que l'IETF a mis en place cette technique.

Le système d'exploitation Contiki permet d'optimiser la consommation énergétique par rapport à d'autres systèmes d'exploitation traditionnels. Toutefois, notons que le manque d'étude comparative rend difficile l'évaluation et la comparaison des performances de ce système d'exploitation par rapport à d'autres.

### 1.2.3.3 MANTIS OS

MANTIS (Multimodal Networks of In-situ micro Sensor) [6] est un système d'exploitation léger, multitâche, conçu pour capteurs sans fil, développé par le groupe Mantis de l'Université du Colorado Boulder. Ce système, écrit en langage C, est capable de supporter un grand nombre de plateformes de RCSF. Trois composants principaux structurent MANTIS : la couche DEV, la couche NET et la couche COM. La première fournit accès aux périphériques d'entrée-sortie. La troisième (couche COM) permet d'accéder aux périphériques de communication tels que le port série et la radio. Cette couche interagit avec la couche MAC. La pile de communication est gérée au niveau utilisateur dans la couche NET. Par défaut, MANTIS utilise un mécanisme d'inondations (flooding) comme protocole de routage.

### 1.2.3.4 SOS

Le système d'exploitation SOS a été conçu pour des RCSFs dynamiques et modulaires [6]. La principale motivation de ce système est la reconfiguration dynamique, autrement dit sa capacité de modifier individuellement le code au niveau de chaque nœud capteur après déploiement du RCSF. Ainsi, mises à jour et ajouts de nouveaux modules peuvent être faits à tout moment après déploiement du réseau. SOS permet également de faire l'allocation dynamique de la mémoire, toutefois il ne possède pas de mécanisme de protection de celle-ci. Le noyau SOS comprend une API (Application Programming Interface) pour capteurs permettant gestion et interaction entre plusieurs nœuds capteurs et les modules utilisés par ces derniers, rendant ainsi une utilisation plus efficace des convertisseurs ADC.

## 1.3 Réseaux de capteurs sans fil

### 1.3.1 Fonctionnement des RCSF

Un RCSF est composé d'un grand nombre de capteurs placés dans une zone d'intérêt donnée. Ils coopèrent ensemble dans le but d'accomplir une tâche commune, très souvent dédié à la surveillance, celle des : champs de bataille dans les applications militaires, zones à risque ou difficilement accessibles, environnement, parkings, entrepôts, places publiques, forêts (afin de pouvoir détecter en temps réel les incendies), surveillance de troupeaux sur les pâturages, contrôle de la production industrielle, contrôle et suivi environnemental, etc. Chaque capteur déployé dans sa zone de surveillance doit rendre compte en permanence de l'état de l'environnement qui l'entoure afin de détecter et mesurer d'éventuels paramètres sur cet environnement et de les transmettre, le cas échéant, à une station de base via un mécanisme de routage donné, qui est souvent une communication multi-sauts. Les rayons de détection et de communication de chaque capteur étant limités, chaque nœud capteur doit ainsi accepter de relayer les données collectées par d'autres nœuds capteurs afin de les acheminer, de proche en proche, jusqu'au centre de traitement, soit le lieu où ces données doivent être traitées et éventuellement exploitées par un utilisateur final donné connecté au réseau via Internet ou un système satellite [6].

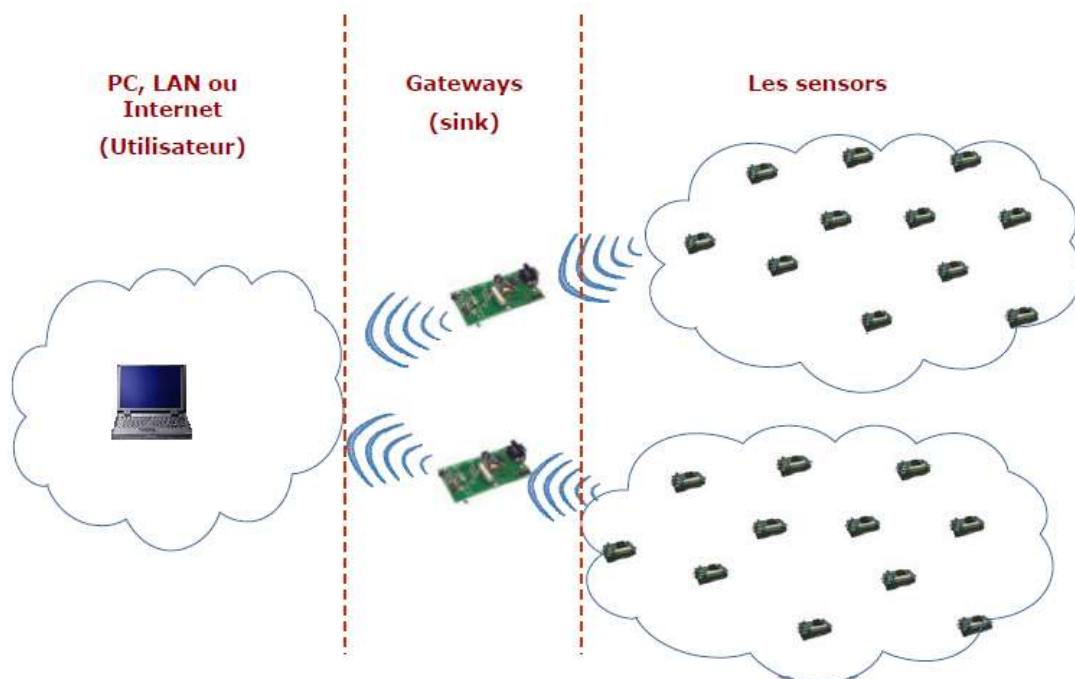


Figure 1.2 : Exemple de RCSF [6].

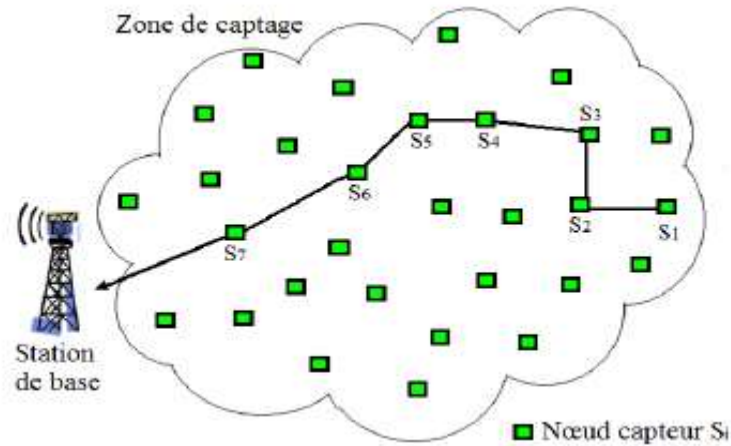
### 1.3.2 Architecture des RCSF

Les architectures des RCSF dépendent des applications et des techniques utilisées pour la transmission des données captées par les différents nœuds capteurs vers la station de base. On distingue principalement deux types d'architectures pour les RCSF : ce sont les architectures à plat et les architectures hiérarchiques. Nous présenterons dans la section suivante ces deux types d'architectures [7].

#### 1.3.2.1 Architecture à plat

Dans les architectures à plat, tout nœud capteur peut communiquer directement avec le centre de traitement en utilisant une forte puissance d'émission ou par l'intermédiaire d'un mode de communication multi-sauts, en utilisant une puissance d'émission beaucoup plus faible. Dans le premier cas, la consommation énergétique du nœud pour l'envoi au centre de traitement est plus importante à cause de la puissance élevée utilisée. Les nœuds peuvent ainsi épuiser assez rapidement leur énergie avec ce mode de transmission. Dans le cas, plus fréquent, de la communication multi-sauts, un nœud capteur qui voudrait transmettre ses données à un autre nœud (destinataire) se trouvant hors de sa portée de transmission (une station de base, par exemple), peut avoir recours à l'utilisation d'autres nœuds intermédiaires comme relais (routeurs). Ce mode offre comme principaux avantages la possibilité de passage à l'échelle, de la gestion des redondances et des tolérances aux pannes.

Cependant, des inconvénients existent, notamment une consommation d'énergie excessive dans l'ensemble du réseau. Cet excès de consommation est imputable à la communication multi-sauts dans laquelle d'autres nœuds capteurs vont participer au routage, dissipant ainsi leur énergie à relayer des paquets provenant d'autres nœuds. Toujours par rapport à ce mode de communication multi-saut, nous pouvons noter une latence plus importante causée par les passages des messages par plusieurs relais multi-sauts avant de parvenir à destination. La figure 1.3 illustre un exemple de RCSF fonctionnant selon un mode d'architecture à plat. Enfin, dans ce mode de communication, nous pouvons souligner des problèmes de sécurité dû au fait que des données qui ne sont pas destinées à des nœuds tiers intermédiaires (relais) peuvent être reçues, traitées et transmises par ces derniers.

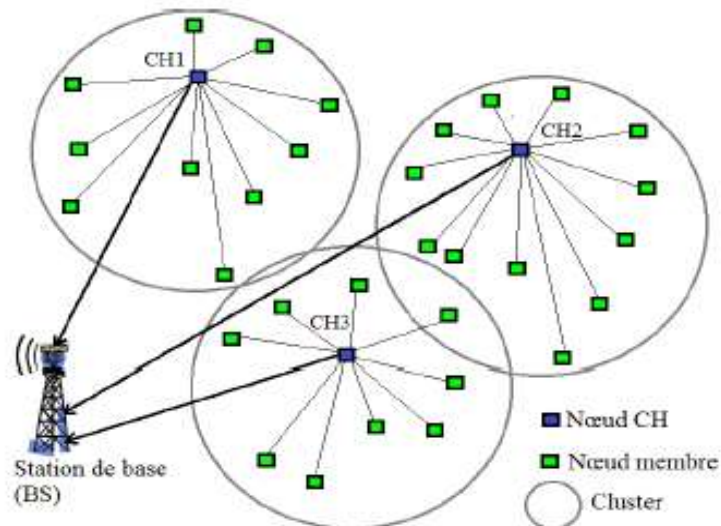


**Figure 1.3** : Exemple d'un RCSF fonctionnant selon un mode d'architecture à plat [7].

### 1.3.2.2 Architecture hiérarchique

Dans les architectures hiérarchiques, le réseau est partitionné en clusters. Dans chaque cluster, un nœud appelé "CH : Cluster Head" est élu et ce dernier représente tous les nœuds membres de son cluster. Ainsi, tout nœud capteur doit être soit CH, soit membre d'un cluster. Un nœud qui n'est pas CH ne pourrait envoyer directement ses données capturées à la station de base. Il les envoie à son CH qui, à son tour, peut envoyer ces données à la station de base. Les CH peuvent également agréger des données reçues de plusieurs sources différentes avant de les envoyer à la station de base, allégeant ainsi cette dernière de certaines tâches de traitement, diminuant en même temps le trafic dans le réseau.

Les avantages de ce mode hiérarchique sont la simplicité, l'économie en consommation d'énergie, surtout pour les nœuds capteurs non CH et une latence plus faible entre les CH et la station de base, puisqu'ils envoient directement les données à cette dernière. L'inconvénient majeur est que chaque CH constitue un point de défaillance unique. En plus, ce mode nécessite beaucoup de messages de signalisations pour l'élection et la maintenance des CH. Nous pouvons également noter comme inconvénient, des problèmes de passage à l'échelle. La figure 1.4 illustre un RCSF fonctionnant selon l'architecture hiérarchique en clusters.



**Figure 1.4:** Exemple d'un RCSF fonctionne selon une architecture hiérarchique en cluster [7].

## 1.4 Applications des RCSF

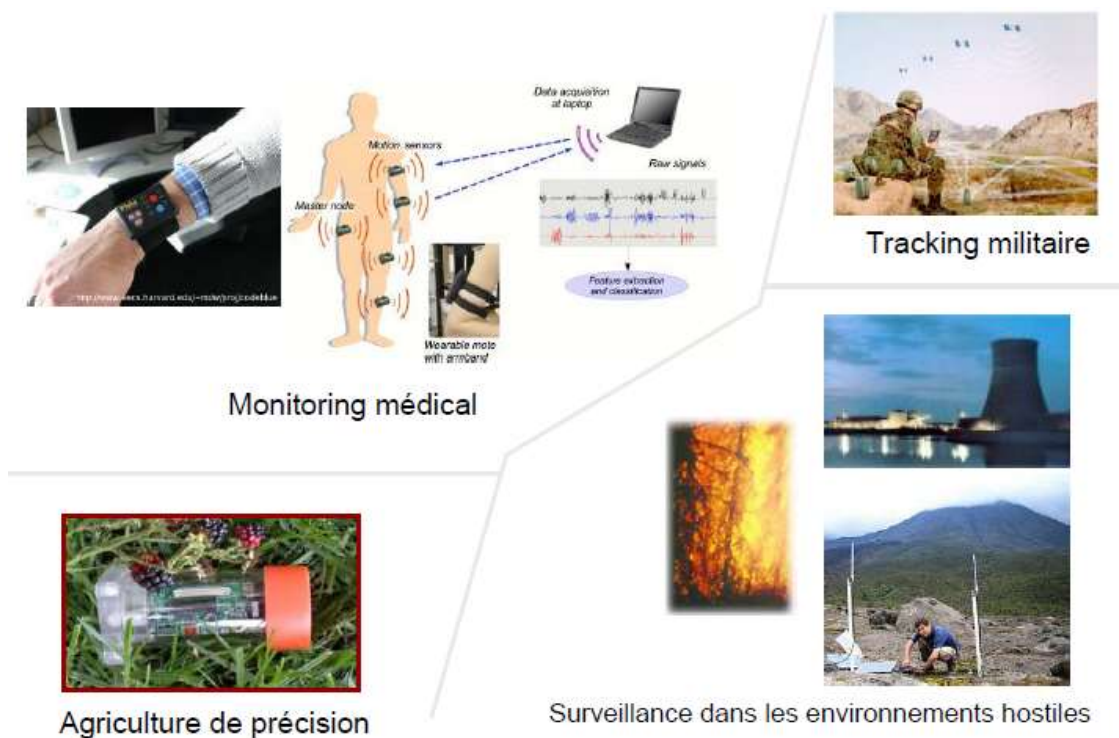
Les RCSF peuvent avoir beaucoup d'applications (voir la figure 1.5). Parmi elles, nous citons [8]:

- **Découvertes de catastrophes naturelles** : Nous pouvons créer un réseau autonome en dispersant les nœuds dans la nature. Des capteurs peuvent ainsi signaler des événements tels que feux de forêts, tempêtes ou inondations. Ceci permet une intervention beaucoup plus rapide et efficace des secours.
- **Détection d'intrusions** : En plaçant, à différents points stratégiques, des capteurs, nous pouvons ainsi prévenir des cambriolages ou des passages de gibier sur une voie de chemin de fer (par exemple) sans avoir à recourir à de coûteux dispositifs de surveillance vidéo.
- **Applications métier** : Nous pourrions imaginer devoir stocker des denrées nécessitant un certain taux d'humidité et une certaine température (min ou max). Dans ces applications, le réseau doit pouvoir collecter ces différentes informations et alerter en temps réel si les seuils critiques sont dépassés.
- **Contrôle de la pollution** : Nous pourrions disperser des capteurs au-dessus d'un emplacement industriel pour détecter et contrôler des fuites de gaz ou de produits chimiques. Ces applications permettraient de donner l'alerte en un temps record et de pouvoir suivre l'évolution de la catastrophe.
- **Agriculture** : Des nœuds peuvent être incorporés dans la terre. Nous pouvons ensuite questionner le réseau de capteurs sur l'état du champ (déterminer par exemple les secteurs les plus secs afin de les arroser en priorité). Nous pouvons aussi imaginer équiper des troupeaux

de bétail de capteurs pour connaître en tout temps, leur position ce qui éviterait aux éleveurs d'avoir recours à des chiens de berger.

- **Surveillance médicale** : En implantant sous la peau de mini capteurs vidéo, nous pouvons recevoir des images en temps réel d'une partie du corps sans aucune chirurgie pendant environ 24h. Nous pouvons ainsi surveiller la progression d'une maladie ou la reconstruction d'un muscle.
- **Contrôle d'édifices** : Nous pouvons inclure sur les parois des barrages des capteurs qui permettent de calculer en temps réel la pression exercée. Il est donc possible de réguler le niveau d'eau si les limites sont atteintes. Nous pouvons aussi imaginer inclure des capteurs entre les sacs de sables formant une digue de fortune. La détection rapide d'infiltration d'eau peut servir à renforcer le barrage en conséquence. Cette technique peut aussi être utilisée pour d'autres constructions tels que ponts, voies de chemins de fer, routes de montagnes, bâtiments et autres ouvrages d'art.

Les images de la figure 1.5 montrent quelques exemples d'illustrations d'applications de RCSF.



**Figure 1.5** : Quelques exemples d'illustrations d'applications des RCSF

## 1.6 Contraintes de conception des RCSF

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

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

· **Le facteur d'échelle** (Scalability) : Le nombre de nœuds de capteurs augmente sur un réseau sans fil et ce nombre peut atteindre le million. Un nombre aussi important de nœuds engendre beaucoup de transmissions entre les nœuds et peut imposer des difficultés pour le transfert de données.

· **Les coûts de production** : Souvent les réseaux de capteurs sont composés d'un très grand nombre de nœuds. Le prix d'un nœud est critique afin de pouvoir concurrencer un réseau de surveillance traditionnel.

· **L'environnement** : Les capteurs sont souvent déployés en masse dans des endroits tels que des champs de bataille, à l'intérieur de grandes machines, au fond d'un océan, dans des champs biologiquement ou chimiquement souillés, par conséquent, ils doivent pouvoir fonctionner sans surveillance dans des régions géographiques éloignées.

· **La topologie de réseau** : Le déploiement d'un grand nombre de nœuds nécessite une maintenance de la topologie. Cette maintenance est réalisée en trois phases : Déploiement, post-déploiement (les capteurs peuvent bouger, ne plus fonctionner,...) et redéploiement de nœuds additionnels.

· **Les contraintes matérielles** : La principale contrainte matérielle est la taille du capteur, qu'il s'adapte aux différents environnements (fortes chaleurs, eau,...), qu'il soit autonome et très résistant vu qu'il est souvent déployé dans des environnements hostiles.

· **Les médias de transmission** : Dans un réseau de capteurs, les nœuds sont reliés par une architecture sans fil. Pour permettre des opérations sur ces réseaux dans le monde entier, le média de transmission doit être standardisé. On utilise le plus souvent l'infrarouge, le Bluetooth et les communications radio Zig Bee.

· **La consommation d'énergie** : Un capteur, de par sa taille, est limité en énergie ( $<1.2J$ ). Dans la plupart des cas le remplacement de la batterie est impossible. Ce qui veut dire que la durée

de vie d'un capteur dépend grandement de la durée de vie de la batterie. Dans un réseau de capteurs (multi-sauts) chaque nœuds collecte des données et envoie/transmet des valeurs. Le dysfonctionnement de quelques nœuds nécessite un changement de la topologie du réseau et un routage des paquets. Toutes ces opérations sont gourmandes en énergie, c'est pour cette raison que les recherches actuelles se concentrent principalement sur les moyens de réduire cette consommation.

## 1.7 La synchronisation dans les réseaux de capteurs sans fil

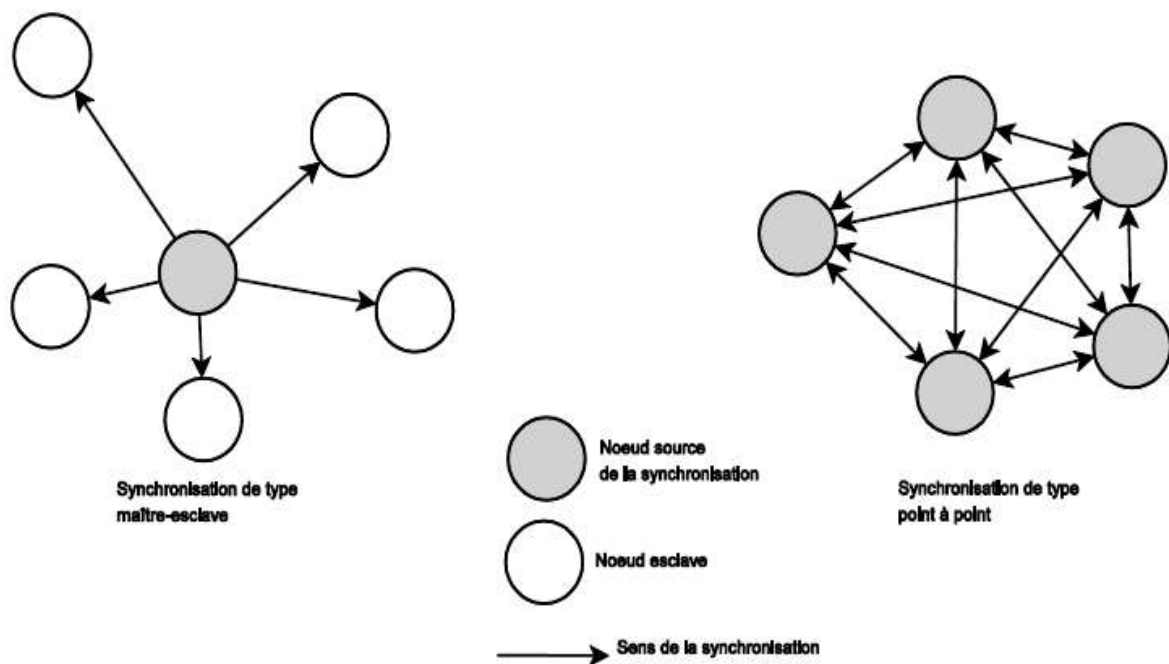
Le processus de synchronisation des nœuds dans un réseau de capteurs sans fil est un challenge important et coûteux (en débit). Les méthodes de synchronisation classiques des réseaux filaires ne sont pas applicables aux domaines des RCSF à cause de leurs caractéristiques et contraintes très différentes. Parmi ces caractéristiques, [9] [10] nous pouvons citer :

- ❖ **L'énergie** : La durée de vie limitée de la source d'énergie des nœuds capteurs rend les méthodes de synchronisation difficiles à mettre en place. En effet, ces méthodes nécessitent des échanges de messages trop longs et nombreux ce qui induit des périodes d'écoute excessivement longues.
- ❖ **La bande passante** : Les méthodes de synchronisation sont contraintes à ne pas saturer le réseau avec un nombre de transmissions important.
- ❖ **La capacité de calcul** : Certaines approches de synchronisation nécessitent une capacité de calcul importante que les nœuds capteurs ne peuvent pas satisfaire à cause de leur architecture généralement simplifiée.
- ❖ **La mobilité** : Les méthodes de synchronisation doivent prendre en considération les changements de topologie.
- ❖ **La latence** : L'un des problèmes des méthodes de synchronisation est le non-déterminisme de ses solutions causé par la latence et par l'impression des instants d'émission et de réception (inconvenient plus marqué pour les réseaux de capteurs qui sont basse énergie et bas débit)
- ❖ **La précision** : Pour des solutions efficaces, les méthodes de synchronisation doivent avoir une précision de l'ordre de la microseconde ( $\mu s$ ).
- ❖ **La durée de validité de la synchronisation** : Pour économiser leur énergie, les capteurs doivent impérativement avoir de longues périodes de sommeil. Les méthodes de synchronisation doivent s'adapter à ce mode de fonctionnement et proposer des solutions valides hors des périodes de synchronisation.

- ❖ **Le coût de déploiement** : L'un des avantages des réseaux de capteurs est leur coût assez faible, les solutions de synchronisation doivent elles aussi être suffisamment économiques.

### 1.7.1 Les méthodes de synchronisations

Deux catégories de méthodes de synchronisation se distinguent dans la littérature pour synchroniser un réseau de capteurs sans fil : les méthodes de type maître-esclave et les méthodes point-à-point. Ces deux catégories sont représentées dans la figure 1.6 :



**Figure 1.6** : Synchronisation de type maître-esclave et point-à-point

Dans les méthodes de type maître-esclave, tous les nœuds esclaves se synchronisent aux dépens du nœud maître. Tandis que dans les approches point-à-point, aucune distinction entre les nœuds n'est nécessaire et tous les nœuds se synchronisent entre eux. Les avantages et inconvénients de ces deux méthodes peuvent se résumer comme suit :

- Synchronisation maître-esclave :
  - Facilité de mise en place.
  - Nécessité de désigner un maître.
  - Indisponibilité lors de la perte du maître.

- Synchronisation point-à-point :
  - Robustesse aux aléas de la connectivité dans le réseau.
  - Complexité de mise en place.

Nous citons dans ce qui suit quelques exemples de méthodes de synchronisation basées sur les deux approches définies auparavant :

a) Synchronisation maître-esclave :

- **NTS** (*Network-wide Time Synchronization*) [11] : le réseau est construit en topologie arborescente, la racine de l'arbre représente le maître qui initie et propage la synchronisation à tous les niveaux inférieurs.
- **LTS** (*Lightweight Tree-based Synchronization*) [12] : dans cette méthode, deux approches peuvent être utilisées pour synchroniser le réseau. La première se base sur le principe de l'arborescence et la propagation de la synchronisation comme défini dans NTS. La seconde utilise des nœuds références, les nœuds qui désirent se synchroniser s'adressent directement aux nœuds références les plus proches.
- **IEEE 1588** [13] : Le protocole de précision temporelle (PTP) IEEE 1588 est méthode standard de synchronisation d'horloges sur des réseaux tels qu'Ethernet avec une précision inférieure à la microseconde. Lorsqu'un système est initialisé, le protocole IEEE 1588 utilise l'algorithme Best Master Clock (meilleure horloge maître) afin de déterminer automatiquement l'horloge la plus précise dans le réseau. Cette horloge devient alors l'horloge maître et les autres horloges deviennent esclaves qui doivent se synchroniser auprès du maître.

b) Synchronisation point-à-point :

- **RBS** (*Reference Broadcast Synchronization*) [14] : cette approche nécessite que chaque nœud diffuse son horloge locale dans le réseau afin de réaliser une synchronisation globale.
- **TS** (*Time Synchronization*) [15] : le but escompté dans cette méthode est de réaliser une synchronisation faible en effectuant un ordonnancement entre les événements plutôt qu'un ordonnancement des horloges des nœuds.
- **Tiny Synchronization** [16] : la synchronisation s'effectue par le calcul des dérivées d'horloges au fur et à mesure des transmissions des nœuds dans le réseau.

## 1.8 Conclusion

Dans ce chapitre, des généralités sur les RCSFs sont présentées à savoir l'architecture Internet d'un nœud capteur ainsi que celle d'un réseau de capteur sans fil. Nous avons présenté par la suite les différents domaines d'applications et leurs synchronisations. Dans le prochain chapitre nous allons introduire les différents algorithmes de localisations existant dans la nature.

## Chapitre 2

# La localisation dans les réseaux de capteurs sans fil

## 2.1 Introduction

La problématique de la localisation dans les RCSFs est particulièrement importante, lorsque les nœuds capteurs ne sont pas en mesure d'identifier leurs positions. Les applications de localisation en espace confiné disponibles actuellement reposent sur des solutions coûteuses en énergie et/ou en matériel. Les solutions basées sur le système GPS sont une alternative pour obtenir avec précision la position d'un nœud. Cependant, ces dernières ne sont pas applicables lorsqu'il s'agit de localiser des entités mobiles dans des milieux confinés ou d'intérieur.

## 2.2 Définition d'un système de localisation

La localisation est un procédé permettant de positionner un objet sur un plan ou une carte géographique, cette opération est réalisée à l'aide d'un terminal capable d'être localisé en temps réel ou de façon différée [17].

## 2.3 Objectifs de localisation

La localisation dans les réseaux de capteurs sans fil déployés de manière aléatoire consiste à déterminer les coordonnées géographiques des différents capteurs. La localisation des nœuds est nécessaire, non seulement pour localiser les différents événements survenus dans la zone surveillée, mais aussi pour le développement de protocoles de routage de l'information récoltée, pour la couverture de la zone d'intérêt, pour l'agrégation des données, etc. Elle est la première tâche exécutée par les nœuds après leur déploiement.

## 2.4 Propriétés d'un système de localisation

Un ensemble de propriétés permettent de comparer les systèmes de localisation, les plus importantes sont les suivantes [18] [19] [20] :

- **Une technique de l'estimation de position** : Trilatération ou triangulation.
- **Un repère** : Qui permet d'obtenir des positions et qui les organise de façon cohérente.

Trois types de position sont observés :

- **Les positions absolues** : Renseignent sur la position réelle de l'objet sur le globe terrestre- longitude et latitude - ou dans l'espace - longitude, latitude et altitude.
  - **Les positions relatives** : Indiquent juste une direction par rapport à un voisinage donné à droite au bout de la rue par exemple.
  - **Les positions symboliques** : Désignent par exemple une salle, un espace particulier
- **La précision de la position** : La position peut aller d'un point dans le cas d'une grande précision à une surface (ou volume) si la précision de position est moins importante.
  - **Une architecture particulière** : Un système de positionnement en intérieur dans un bâtiment par exemple ne possède pas les mêmes contraintes qu'un système de localisation d'extérieur.
  - **Coût** : Les coûts d'un système de localisation peuvent être exprimés en termes de :
    - Temps : Installation de l'infrastructure, administration et maintenance.
    - Espace : La taille de chaque capteur ainsi que la taille de l'infrastructure.
    - Énergie : Consommation électrique pour chaque capteur et pour l'infrastructure.
    - Budget : Coûte de chaque capteur, frais de l'installation et maintenance.

## 2.5 Critères d'évaluations d'un système de la localisation

Un algorithme de localisation est évalué selon une liste de critères dont nous citons [21] :

### 2.5.1 Précision de la localisation

L'erreur de la localisation est souvent définie comme étant, la distance euclidienne entre les vraies positions des nœuds et celles estimées par l'algorithme. L'objectif d'un algorithme de localisation est de minimiser cette erreur pour augmenter la précision de localisation. Généralement, cette imprécision vient de l'imprécision des méthodes d'estimation de la distance. Les

obstacles environnementaux et les terrains irréguliers peuvent influencer la précision des algorithmes de localisation.

### **2.5.2 Contraintes de ressources**

Les nœuds capteurs possèdent généralement des ressources très limitées. Ils possèdent de faibles processeurs et de petites mémoires, ce qui rend les grands calculs irréalisables. Par conséquent, un algorithme de localisation doit être simple et non complexe et son développement n'exige pas de grands calculs ni de grande capacité de stockage.

### **2.5.3 Contraintes énergétiques**

La seule source d'énergie d'un nœud capteur est sa batterie. Pour cela, dans les RCSFs, une bonne gestion de l'énergie est nécessaire. Comme le facteur dominant de la consommation d'énergie est la communication radio, il faut trouver un algorithme de localisation qui communique le moins possible via la radio.

### **2.5.4 Techniques de mesure**

Certain système de la localisation dans les RCSFs repose sur des mesures. Il y a plusieurs facteurs qui influent sur le choix de l'algorithme à utiliser tel que la précision de l'emplacement estimé. Nous pouvons citer quelques architectures du réseau, le degré moyen de nœuds (à savoir, le nombre moyen de voisins par capteur), la forme géométrique de la zone du réseau et la distribution de capteurs dans ce domaine, la synchronisation du temps du capteur et la bande passante de signalisation entre les capteurs.

## **2.6 Composition d'un système de localisation**

Un système de localisation peut se décomposer en trois parties. Chaque partie à son objectif et sa méthode de résolution. Nous les présenterons comme suit [22] (La figure 2.1 illustre la composition d'un système de localisation) :

### **2.6.1 Estimation de la distance**

Cette partie consiste à estimer la distance entre deux nœuds différents, en utilisant différentes méthodes (RSSI, TOA...) que nous allons détailler dans le paragraphe 2.7.1, le résultat obtenu sera utilisé dans la partie suivante.

## 2.6.2 Calcul de position

Cette partie consiste à estimer la position d'un nœud en utilisant les informations de distance obtenues précédemment et les positions des nœuds de référence en utilisant par exemple la trilatération et la triangulation que nous verrons en détails.

## 2.6.3 Algorithme de localisation

Les algorithmes de localisation définissent la manière avec laquelle les informations déjà calculées seront utilisées afin que tous les nœuds du réseau puissent estimer leur position. Nous pouvons distinguer deux façons d'implémenter les algorithmes de localisation : les algorithmes centralisés et les algorithmes distribués.

La figure ci-après, illustre la décomposition d'un système de localisation.

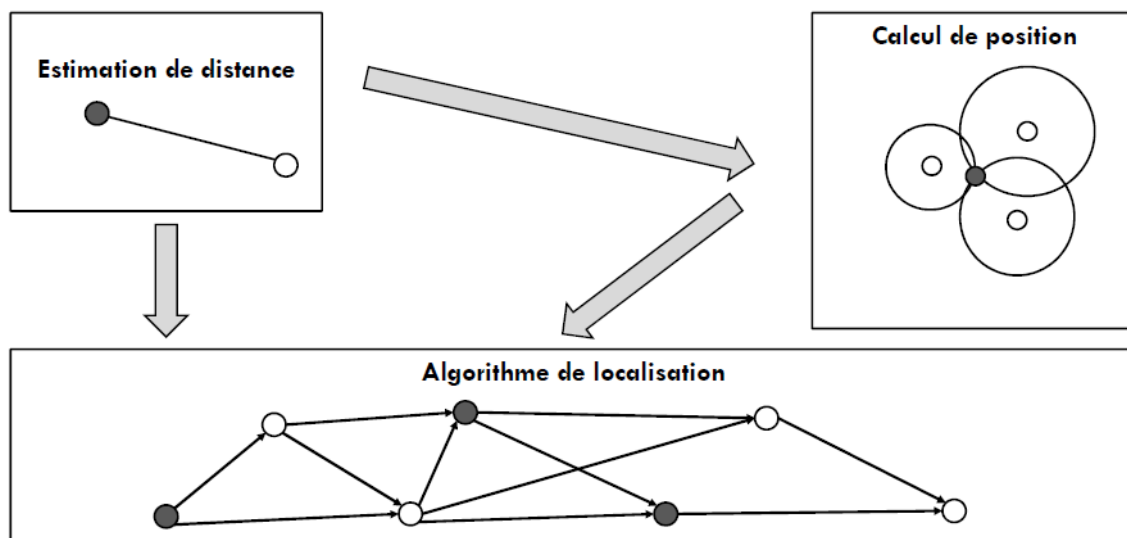


Figure 2.1 : Composition d'un système de localisation [22].

## 2.7 Méthodes de localisation

### 2.7.1 Méthodes de localisation physique

La localisation physique dans les réseaux de capteurs sans fil se base essentiellement sur les propriétés des ondes radio reçues par les nœuds du réseau, voici les principales méthodes de localisation basées sur le traitement du signal.

### 2.7.1.1 Trilatération

Cette méthode nécessite la connaissance de la distance entre le capteur inconnue et chaque nœud d'ancrage [23].

Soient trois nœuds d'ancrages  $B_1$ ,  $B_2$  et  $B_3$ ,  $M$  est le nœud que nous voulons localiser. Les positions  $(x_i, y_i)$  des balises ainsi que les distances  $D_i$  sont connus. La relation entre  $M$ ,  $B_1$  et  $B_2$  peut être écrite comme :

$$(x - x_1)^2 + (y - y_1)^2 = d_1^2$$

$$(x - x_2)^2 + (y - y_2)^2 = d_2^2$$

$M$  est donc situé à l'intersection des cercles  $C_1 (B_1, d_1)$  et  $C_2 (B_2, d_2)$ . Un troisième cercle généré en utilisant les informations de  $B_3$  nous permet de choisir la position correcte entre les deux solutions du cas général de deux cercles entrecroisés,

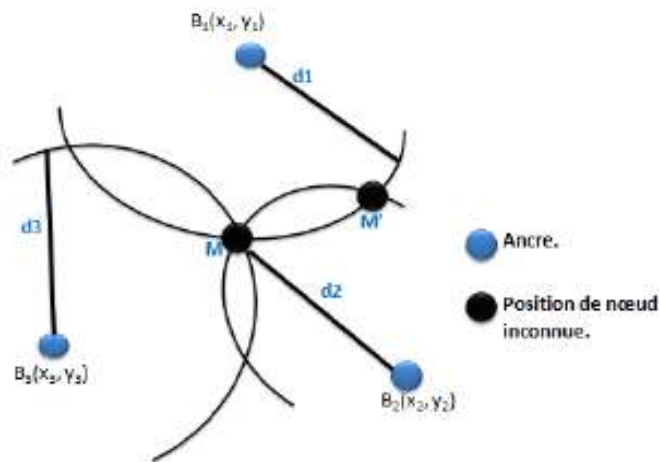


Figure 2.2 : Trilatération [23].

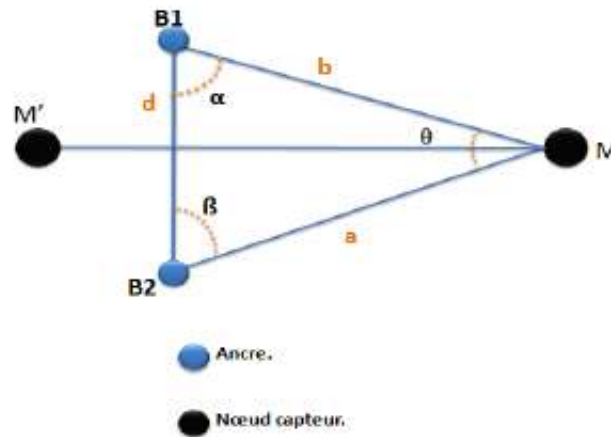
### 2.7.1.2 Triangulation

Afin d'appliquer cette méthode, le nœud capteur inconnue doit être conscient de la distance entre les deux ancres et les angles  $\alpha$  et  $\beta$  comme indiqué sur la Figure (2.3). La loi des rendements sinus :

$$a = \frac{d \times \sin \alpha}{\sin(\pi - \alpha - \beta)}$$

$$b = \frac{d \times \sin \beta}{\sin(\pi - \alpha - \beta)}$$

Une fois que les caractéristiques du triangle sont connues, les coordonnées du troisième sommet peuvent être calculées. Comme indiqué en décrivant trilatération, troisième ancre ou une hypothèse supplémentaire permettra l'identification de la solution correcte [23].

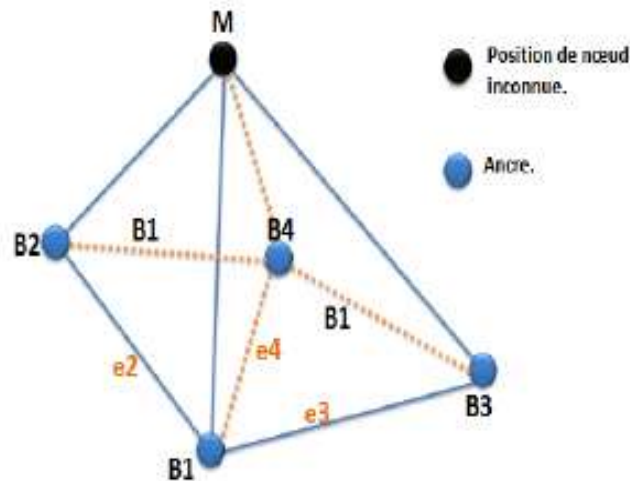


**Figure 2.3** : Triangulation [23].

Le point M se retrouve à l'intersection des droites passant par les couples (B1, M) et (B2, M) figure (2.3). En définissant un nouveau repère dont l'origine est B1 où (B1, B2) se confond avec l'axe des ordonnées,

### 2.7.1.3 Multilatération

Multilatération nécessite des informations en temps plutôt que la distance. La différence de temps d'arrivée d'un signal (TDOA) est utilisée pour déterminer la position. La figure (2.4) présente le cas avec le capteur étant en mesure d'atteindre quatre ancres qui sont synchronisés. Le nœud capteur diffuse une frame. Chaque ancre reçoit les trame de l'heure locale de réception, et calcule ensuite la différence entre ce marqueur de temps et celui fourni par une ancre de référence réglée [23].



**Figure 2.4** : Multilatération [23].

$e_i$  : La distance de la référence choisie pour le nœud,  $B_i$  : Pour le cas représenté à la figure (2.4)  $B_1$  est la balise de référence.

#### 2.7.1.4. Calcul des distances par calcul de temps de transit du signal

La localisation proposée par les systèmes GPS est basée essentiellement sur le principe de trilatération, cette méthode utilise la méthode du calcul des temps de transit du signal reçu comme suit : à partir d'ondes radio reçues en visibilité directe par trois nœuds du réseau dont la localisation est connue, la position du nœud récepteur peut être calculée à partir de la formule suivante :

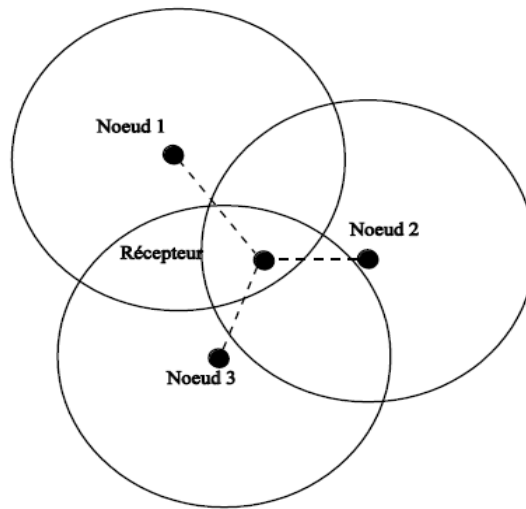
$$D = V \times T$$

Où  $D$  : représente la distance entre les nœuds émetteur et récepteur.

$V$  : la vitesse de propagation du signal.

$T$  : le temps de propagation du signal sur le médium.

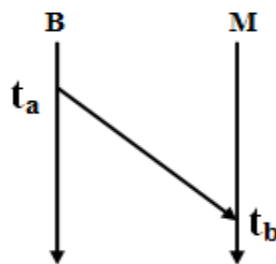
La figure 2.5 illustre une configuration avec trois nœuds références et un nœud récepteur qui a besoin de connaître sa position. Ce dernier calcule la distance qui le sépare des trois nœuds références ce qui lui permet d'estimer sa position (en appliquant par exemple des méthodes géométriques pour estimer sa position à la jonction des trois cercles de la figure 2.5).



**Figure 2.5** : Localisation par trilatération [24].

Pour le calcul des distances, plusieurs méthodes sont proposées :

**a) ToA (Time of Arrival)** : [24] Si la vitesse de propagation du signal dans l'air ainsi que le temps du début de l'émission sont connus, il est possible dans ce cas de déterminer avec précision la distance qui sépare un nœud émetteur d'un nœud récepteur. En utilisant la formule (2.3) avec les nœuds références, le nœud récepteur peut déterminer sa position. Néanmoins, cette méthode présente malgré cela une faille car elle nécessite une forte synchronisation des nœuds ainsi qu'une horloge interne stable et précise. Bien que des solutions existent pour pallier les problèmes de synchronisation, ces solutions nécessitent quand même un nombre important de nœuds références de calculs, ce qui conduit à des coûts en équipements et en énergie.



**Figure 3.6** : Principe de ToA [24].

$$\text{Distance} = \text{Temps} \times \text{Vitesse} \quad (2.3)$$

Où : vitesse =  $3 \times 10^8$  m/s et Temps =  $t_b - t_a$

$t_b$  : La date de réception.

$t_a$  : La date d'envoi.

Pour calculer la distance nous distinguons deux cas :

- **Lorsque les nœuds sont synchrones (les horloges des nœuds battent aux même rythme) :**  
La distance qui sépare deux capteurs se déduit de la vitesse de propagation du signal et de la différence entre les dates d'émission et de réception du message. Cette technologie est celle utilisée par le système GPS.
- **Lorsque les nœuds ne sont pas synchrones (l'horloge du nœud émetteur indépendante de celle du récepteur) :** L'envoi d'un message aller-retour est nécessaire.

En fonction de son horloge, de la vitesse de propagation du signal et du temps de traitement du signal reçu, un capteur récepteur obtient la distance qui le sépare du capteur émetteur en calculant la différence entre les dates d'émission et de réception, en y soustrayant le temps de traitement du signal, puis en divisant le résultat par deux.

**Avantages :**

- Algorithmes de positionnement simple ;
- Précision plus élevée en milieu confiné.

**Inconvénients :**

- Nécessite la synchronisation parfaite entre l'émetteur et le récepteur ;
- Nécessité d'avoir le trajet direct ;
- Nécessité d'une résolution temporelle élevée au récepteur.

**b) TDoA (*Time Difference of Arrival*) :** [25] Pour pallier la contrainte de forte synchronisation de la méthode ToA, TDoA a une approche différentielle et utilise des signaux de natures différentes, l'ultrason et la radio par exemple. La différence des temps d'arrivée de ces signaux permettra de calculer la distance entre deux nœuds sans qu'il y ait la nécessité d'une forte synchronisation. Il faut cependant connaître les temps de propagation des deux types de signaux. La seule contrainte de cette méthode, est que chaque nœud dans le réseau soit pourvu d'antennes pour les différents signaux et d'une horloge précise, ce qui n'est pas forcément le cas dans les produits disponibles actuellement. De plus, la communication à travers les deux signaux ainsi que la gestion des deux modules sans fil entraînent un surcoût énergétique.

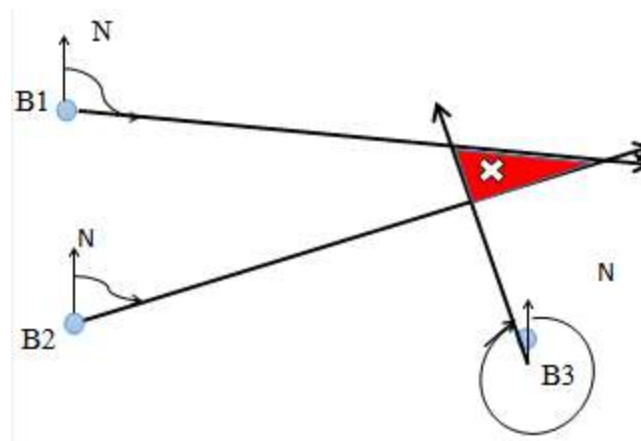
**Avantages :**

- Algorithme de positionnement simple ;
- Précision plus élevée en milieu confiné ;
- Pas besoin de synchronisation d'horloges entre le mobile et les stations de bases.

**Inconvénients :**

- Nécessité d’avoir le trajet direct ;
- Synchronisation d’horloges nécessaire entre les paires de stations de base ;
- Nécessité d’une résolution temporelle élevée au récepteur.

c) **AoA (*Angle of Arrival*)** : [26] Dans cette méthode, chaque nœud capteur est doté de plusieurs antennes (ou d’une antenne sectorielle). Il ainsi est capable d’estimer l’angle d’arrivée du signal reçu. Cette information reçue par plusieurs nœuds voisins permettra d’appliquer une triangulation et d’obtenir une distance relative. L’inconvénient de cette méthode est le même que la précédente en terme de consommation d’énergie due à la présence de plusieurs antennes.



**Figure 2.7** : Principe d’AoA [26].

d) **Combinaison TDoA et AoA** : [27] La solution proposée est une combinaison des deux approches TDoA et AoA, afin d’accroître la précision de la localisation et de l’étendre à des environnements 3D. Cette solution consiste en une amélioration des techniques utilisées en 2D en y combinant l’exploitation du temps et des angles d’arrivée.

### 2.7.1.5 Calculs des distances par mesure de puissance du signal

Cette approche utilise le principe de l’atténuation progressive du signal au fur et à mesure de la distance parcourue. La mesure de la puissance ou RSSI (*Receive Signal Strength Indicator*) du signal reçu par nœud permet d’estimer la distance entre les nœuds. La trilatération est ensuite utilisée par le nœud récepteur avec au moins trois nœuds de référence afin de déduire sa position. L’avantage de cette solution est son indépendance à la forte synchronisation des nœuds.

Néanmoins, dans un environnement confiné et/ou fortement bruité, la mesure du RSSI est fortement perturbée. En effet, les obstacles de types murs, personnes ou objets métalliques altèrent considérablement le signal durant son parcours. De plus, les nœuds mobiles dans le réseau impliquent des imprécisions lors de l'estimation de la distance [28].

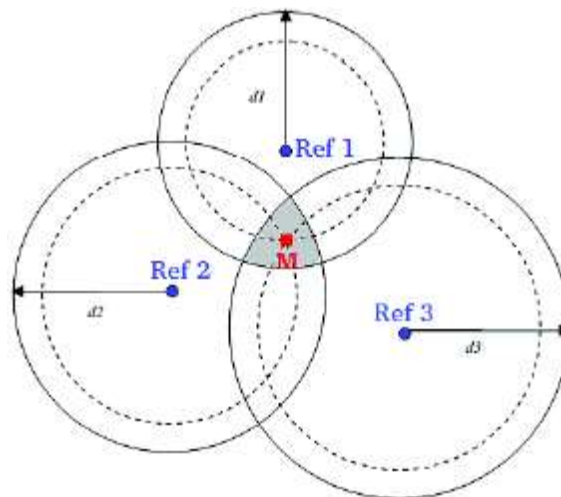
La technique RSSI se base sur la puissance du signal reçu pour estimer la distance entre les deux points d'émission et de réception, la formule générale pour calculer la puissance du signal reçu dans un espace libre est :

$$Pr = Pr(d_0) - 20 \log_{10}(d/d_0)$$

Où

$d$  : Est la distance entre les stations de bases.

$p_0$  : La puissance du signal reçu à distance  $d_0$  de l'antenne mobile.



**Figure 2.8** : Principe de l'utilisation des RSSI [28].

**Tableau 2.1** : Liste des caractéristiques essentielles des méthodes décrites dans les paragraphes précédents.

Méthodes	Précision	Distance maximale	Matériel supplémentaire	Défis
RSSI	2-4 m	La portée	Rien	Variations du RSSI, interférences
ToA	2-3 cm	La portée	Rien	Synchronisation des nœuds
TDoA	2-3 cm	Quelques mètres (2-10 m)	Capteur à ultrason	La distance maximale
AoA	Quelques degrés (5°)	La portée	Ensemble de récepteurs	Travailler avec des nœuds de petite taille

### 2.7.2 Méthodes de localisation logique

La localisation logique repose sur l'échange d'informations de voisinage entre les nœuds du réseau telles que les tables de routage ou la topologie du réseau. Plusieurs solutions sont proposées dans la littérature, nous citons :

- **Méthode des badges actifs** : [29] Dans cette méthode, chaque nœud du réseau transmet périodiquement (chaque 10 secondes par exemple) des messages de type *hello* à destination de nœuds statiques appelées ancres (ou balises) placés à des endroits précis. Par exemple, dans un bâtiment, le rôle principal d'une ancre est de relayer les informations jusqu'au puits de données. La position du nœud sera associée à celle du nœud ancre qui a reçu le message *hello* en dernier.
- **Méthode des intersections des signaux** : [30] Le même principe des nœuds ancres est repris dans cette approche. Chaque nœud capteur tient compte du nombre d'ancres à sa portée, de leur portée radio et de leur position afin d'estimer sa position dans une zone. Cette zone correspond à l'intersection des zones couvertes par les ancres qu'il reçoit. Par conséquent, il est nécessaire d'avoir un nombre important d'ancres à portée des nœuds pour augmenter la précision de la localisation.
- **Méthode de Monte-Carlo (MCL)** : [31] Dans cette approche, la localisation est réalisée sous forme de filtrage des positions impossibles des nœuds dans le réseau en utilisant aussi le principe des nœuds ancres. La méthode Monte-Carlo calcule d'abord les distributions des positions

possibles des nœuds et en fonction des positions précédentes. En fonction des positions précédentes des stations mobiles et des observations obtenues à l'aide des nœuds ancres déployés dans le réseau, l'algorithme filtre les positions impossibles.

## 2.8 Algorithmes de localisation

Dans cette section nous allons présenter les différentes catégories des algorithmes de localisation.

### 2.8.1 Algorithmes à ancres statiques et nœuds statiques

Dans ce cas tous les nœuds ont des positions fixes. Ces algorithmes peuvent être classifiés en deux sous classes :

- **Algorithme Range-based** : Exigent de mesurer la distance entre les nœuds à localiser et les nœuds ancres et utilisent différentes techniques de mesure de distance (RSSI, AOA etc.) pour estimer la position d'un nœud.
- **Algorithme Range-Free** : Les algorithmes de cette catégorie n'exigent pas la mesure de la distance ou de l'angle entre les nœuds, mais estiment la distance entre deux nœuds suivant l'information de connectivité ou celle de l'énergie consommée lors de la transmission d'un signal.

### 2.8.2 Algorithmes à ancres statiques et nœuds mobiles

L'utilisation du nœud mobile est étroitement liée à notre vie quotidienne. Ceux-ci peuvent être divisés en deux catégories :

- **Algorithmes prédictifs** : Le principe de ces algorithmes consiste en la prédiction des coordonnées des nœuds mobiles en se basant sur l'historique de leurs positions précédentes.
- **Algorithmes à base de clusters** : Ces algorithmes sont plus appropriés pour les RSCF. Ils divisent le réseau en plusieurs clusters, et chaque nœud ancre localise respectivement les nœuds simple se trouvant dans son propre cluster.

### 2.8.3 Algorithmes à ancres mobiles et nœuds statiques

À présent, quelques algorithmes de localisation qui utilisent des nœuds ancres mobiles afin de localiser des nœuds statiques. Nous distinguons deux catégories :

- **Algorithmes de localisations géométriques** : Ces algorithmes changent le problème de localisation en un problème géométrique et calculent ainsi les coordonnées des nœuds inconnus en se basant sur les relations géométriques entre les nœuds mobiles et les nœuds fixes.
- **Algorithmes pour planification de la trajectoire** : Une ancre mobile diffuse des paquets contenant des informations sur sa position à un moment donné tout en se déplaçant selon une trajectoire spécifique et les nœuds reçoivent ces paquets afin de pouvoir estimer leurs positions. Cette approche réduit le coût d'un RCSF, et offre plus de précision.

### 2.8.4 Les algorithmes à ancres mobiles et nœuds mobiles

Le processus de localisation de tels algorithmes peut être compliqué si nous tenons compte de l'environnement ou les nœuds peuvent être déployés, ce qui peut représenter une contrainte pour la mobilité des nœuds, par exemple : algorithmes à distribution probabiliste.

Nous remarquons que tous les algorithmes de localisation ont une caractéristique commune est qu'ils requièrent tous des nœuds ancres (ou nœud de référence) afin de localiser les nœuds mobiles (dans la position est inconnue).

## 2.9 Forme d'implémentation

Nous distinguons deux façons d'implémenter un algorithme de localisation selon leur organisation de calcul [31] :

### 2.9.1 Les méthodes centralisées

Les algorithmes centralisés sont conçus pour fonctionner sur une machine centrale très puissante au niveau ressources. Les nœuds capteurs recueillent des informations (signal, voisins, distances, etc.) de leur environnement et les transmettent à une station de base qui à son tour les analyse, calcule les positions et les transmet aux nœuds. Les algorithmes centralisés contournent le problème des ressources limitées des nœuds en acceptant des coûts de communications très élevés pour envoyer les informations à la machine centrale.

Ces algorithmes deviennent de plus en plus coûteux quand la taille du réseau augmente, car ça épuise les nœuds qui sont trop proches de la station de base qui subissent un très grand nombre de communications.

En outre, les algorithmes centralisés exigent qu'une station de base puissante soit déployée parmi les nœuds, ce qui n'est pas toujours possible. Dans le cas où c'est possible, le problème de la mise à l'échelle peut être résolu en déployant plusieurs stations de bases.

Cependant, la centralisation permet à un algorithme d'être plus complexe, car les calculs se font sur la machine centrale et non pas par les nœuds eux-mêmes.

### **2.9.2 Les méthodes distribuées**

Dans le cas d'un algorithme distribué, tous les nœuds communiquent avec leurs voisins pour estimer les distances et échanger les informations de voisinage, afin de dériver leur position. Par conséquent, à la fin du processus de localisation, chaque nœud doit connaître sa position ainsi que celles de ses voisins sans l'aide d'aucune unité centrale. Les algorithmes distribués, extrapolent généralement les positions des nœuds à partir de celles des ancres. Ainsi, ils localisent les nœuds directement dans le système de coordonnées global de ces ancres.

Comme le calcul des positions se fait par les nœuds eux-mêmes, les algorithmes distribués ne sont pas complexes. Pour les réseaux à grande échelle, on considère qu'une méthode distribuée est nécessaire car les méthodes centralisées demanderaient trop de communication pour l'acheminement des informations vers l'unité centrale et consommeraient donc trop d'énergie.

## **2.10 Conclusion**

Dans ce chapitre, nous avons présenté, une vue globale sur les différents points qui régissent les systèmes de localisation dans les RCSFs, les propriétés d'un système de localisation, ainsi que sa composition, et les méthodes de localisation physique et logique, et enfin nous avons présenté et décrit de façon détaillé les algorithmes de localisation.

## Chapitre3

# Ancre mobile avec une trajectoire en Scan pour la localisation dans les RCSFs

### 3.1 Introduction

Au cours de notre travail de fin cycle nous nous sommes intéressés au problème de localisation dans le RCSF, dont la zone d'intérêt est sous forme rectangulaire. Dans ce chapitre nous proposons une amélioration d'un algorithme de localisation proposé dans [32], il est basé sur l'utilisation d'une ancre mobile équipée par un GPS. Une fois déployé, l'ancre mobile scanne toute la zone en diffusant des informations autour de lui pour aider les autres nœuds à trouver leurs positions.

### 3.2 Quelques définitions et terminologies

Nous avons jugé utile de commencer ce troisième chapitre par la présentation de quelques notions et terminologies. Ces notions sont nécessaires pour la compréhension des solutions proposées pour la problématique de la localisation.

#### 3.2.1 Ancre/Beacon

Les ancres (souvent appelées aussi beacons) sont au préalable nécessaires pour localiser les nœuds d'un réseau dans un système de coordonnées global. Les ancres sont simplement des nœuds ordinaires qui connaissent leurs coordonnées à priori. Cette connaissance pourrait être difficilement codée, ou bien facilement acquise par un certain matériel supplémentaire comme un récepteur GPS.

Les ancres peuvent être utilisées de plusieurs façons. Certains algorithmes de localisation trouvent une carte arbitraire relative pour les coordonnées des nœuds, puis ils utilisent les ancres pour déterminer une transformation rigide des coordonnées relatives vers les coordonnées globales. D'autres algorithmes, partants des positions des ancres, calculent les positions des nœuds non-ancres dans un système global.

Les récepteurs GPS sont chers, ne sont pas utilisables pour des applications à l'intérieur et consomment aussi une quantité non négligeable de l'énergie de la batterie, qui peut être un problème pour le pouvoir énergétique limité des nœuds. L'emplacement manuel des ancres peut être non pratique (par exemple lors du déploiement de 20000 nœuds avec 1000 ancres), ou même impossible (par exemple lors du déploiement à partir d'un avion).

Pour pallier à certains de ses inconvénients, des travaux dans la littérature ont proposé l'utilisation d'une seule ancre mais cette fois-ci il s'agit d'une ancre mobile à la place de plusieurs fixes.

### **3.2.2 Ancre mobile**

Le principe est l'utilisation d'une seule ancre mobile. Au lieu d'avoir plusieurs ancres statiques, une seule mobile est déployée avec les nœuds, puis elle traverse la zone de surveillance en communiquant avec les autres nœuds afin de les aider à s'auto-localiser. L'ancre mobile diffuse des informations tout au long de sa trajectoire. Elle peut être un opérateur humain, un robot déployé avec le réseau de capteurs, ou dans le cas d'un déploiement d'un avion, l'avion lui-même. Ce type d'architecture offre des avantages pratiques importants :

- l'ancre mobile n'est pas limitée en énergie comme c'est le cas d'un simple nœud capteur.
- La taille d'une ancre est beaucoup plus grande que la taille d'un capteur et, par conséquent, il est beaucoup plus facile d'installer une unité GPS.
- Facilite la phase de déploiement.
- Réduire le coût (pas d'utilisation d'un grand nombre d'unités GPS qui sont chères).
- La précision de localisation peut également être améliorée (spécification de la bonne trajectoire).

Pour ces raisons nous avons choisi d'implémenter un algorithme de localisation avec une ancre mobile.

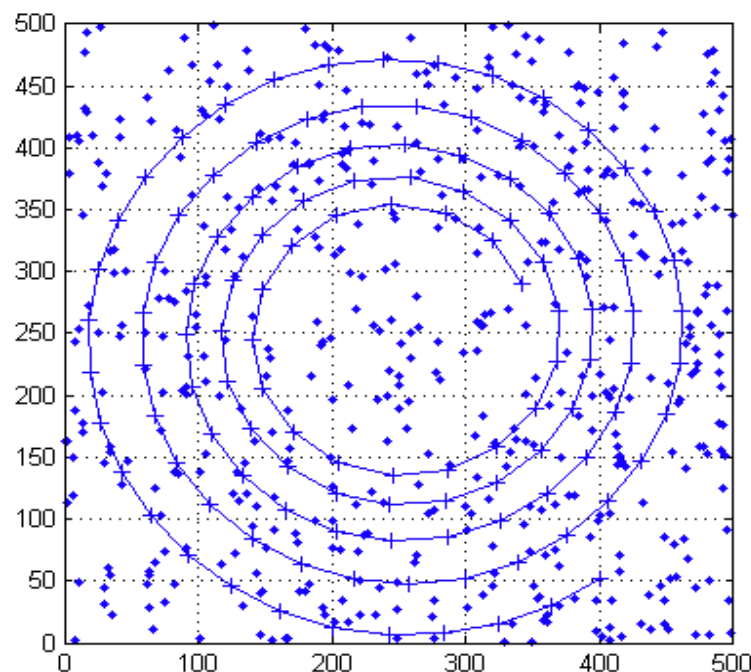
### 3.2.3 Trajectoire

Représente l'itinéraire emprunté par l'ancre mobile afin d'aider un maximum de nœuds capteurs appartenant au réseau, à définir leurs positions. La détermination de la trajectoire est la problématique principale lors du développement des algorithmes de localisation se basant sur les ancrs mobile. Elle peut être statique ou dynamique.

### 3.3 Problématique

Les auteurs de l'approche de localisation rapportée dans [32] se sont intéressés aux RCSF dont le déploiement des nœuds est ciblé tout autour d'une zone d'intérêt donnée. En effet, dans cette solution la trajectoire de l'ancre mobile en spirale peut s'avérer plus adapté, mais dans la majorité des autres cas, tous les nœuds du RCSF sont déployés sur toute la zone d'intérêt représentée comme un rectangle. Et dans ce cas, la trajectoire en spirale d'une ancre mobile ne permet pas d'atteindre les capteurs des quatre coins de la zone d'intérêt.

La figure 3.1 illustre la distribution des nœuds dans la région d'intérêt, la trajectoire adoptée par l'ancre mobile ainsi que les positions de diffusion des messages beacon.



**Figure 3.1** : Environnement de la solution de localisation proposée dans [32] avec la trajectoire en Spiral.

### 3.4 Élément de solution

Pour répondre à la problématique posée précédemment concernant le travail mené dans [32], nous avons pensé à l'utilisation d'une autre forme de trajectoire de l'ancre mobile, cela nous permettra notamment de couvrir toute la zone d'intérêt. Pour cela dans la section suivante nous allons exposer les différentes trajectoires existantes dans la littérature afin d'en choisir celle qui convient le mieux à notre étude.

### 3.5 Quelques trajectoires existantes dans la littérature

Cette approche a comme objectif de réduire le coût de RCSF, et d'obtenir une plus grande précision de localisation. Cependant, comment trouver le chemin optimal est le problème fondamental. Parmi les solutions nous situons par exemple :

**Dimitrios et al** : A proposé trois trajectoires différentes pour une ancre mobile, le chemin HILBERT, SCAN et SCAN DOUBLE [33], respectivement comme le montre la figure 3.2.

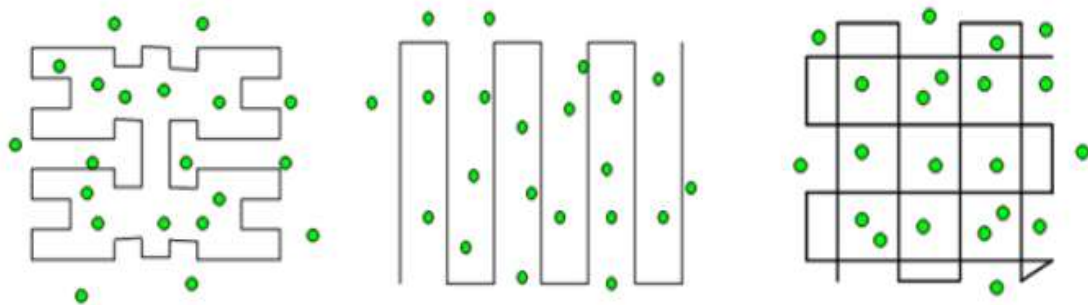


Figure 3.2 : Trois différentes trajectoires [33].

### 3.6 La Trajectoire

Plusieurs recherches scientifiques ont été effectuées afin de trouver la meilleure trajectoire.

La définition d'une trajectoire se base sur les critères :

- Il faut que cette trajectoire offre une couverture de la totalité de la zone d'intérêt.
- fournit des paquets de bonne qualité en évitant la colinéarité des paquets avec un fort signal de sorte qu'ils seront capables d'estimer la distance avec une bonne précision.
- doit être courte que possible pour réduire la consommation d'énergie, les points de diffusion du paquet, et le temps pour la localisation.

Pour trouver la meilleure trajectoire (trajectoire optimale) du mobile pour la localisation des réseaux de capteurs est un problème très difficile.

Par suite, nous décrivons les trois trajectoires différentes pour une ancre mobiles.

**Scan (balayage) [34]** : Est une trajectoire simple et facilement a mis en œuvre. L'ancre mobile traverse la zone de réseau suivant une dimension, l'ancre mobile se déplace le long de l'axe y, et la distance entre deux segments successifs de la trajectoire parallèle à l'axe y, définit la résolution de la trajectoire (L). Si le rayon de communication des capteurs est R, la résolution L devrait être au plus 2R, pour vous assurer que tous les capteurs seront en mesure de recevoir des paquets. La trajectoire Scan a comme avantage d'offrir une couverture uniforme de la zone d'intérêt, et l'inconvénient majeur est la colinéarité de paquets.

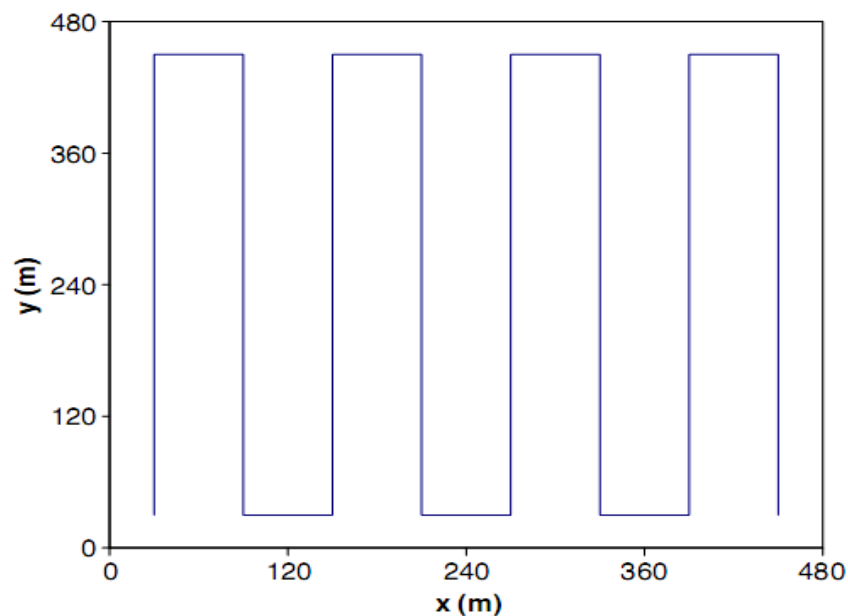
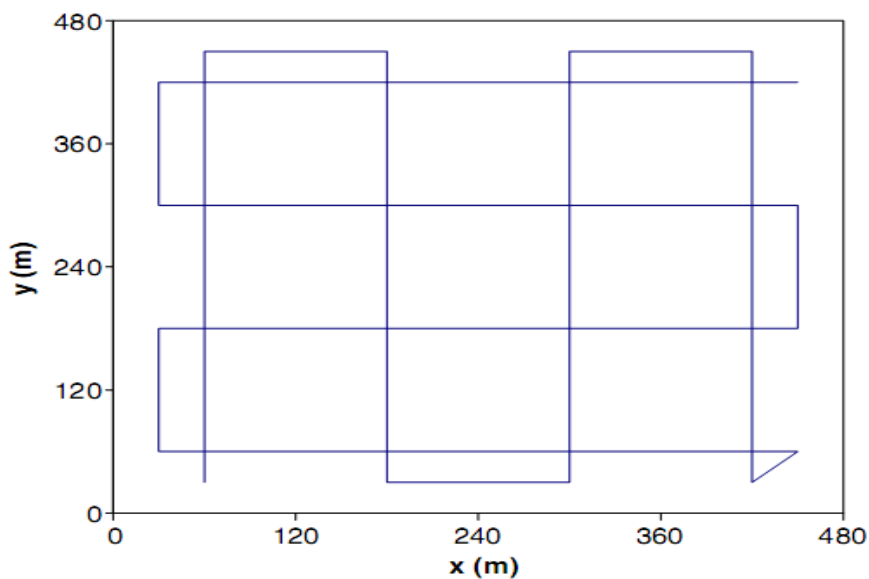


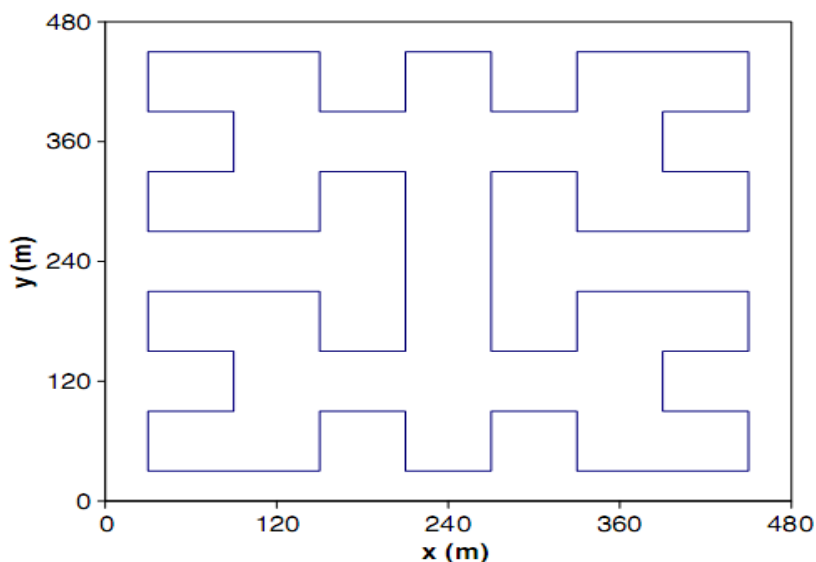
Figure 3.3 : Trajectoire en scan [34].

**Double-Scan [34]** : Une autre façon simple de résoudre le problème de colinéarité du SCAN consiste à balayer le réseau le long des deux directions. Le problème avec cette méthode est qu'elle nécessite à l'ancre mobile de parcourir une distance doublée par rapport à SCAN. Pour garder la distance parcourue similaire pour toutes les trajectoires, il faut réaliser Double-Scan avec une résolution L double par rapport à SCAN.



**Figure 3.4** : Trajectoire en double scan [34].

**Hilbert [34]** : La Courbe de Hilbert est une courbe fractale unidimensionnelle continue remplissant l'espace, qui visite tous les points dans un espace de 2 ou 3 dimensions, exactement une et une seule fois sans jamais se recouper. La courbe de base dite courbe d'ordre 1. Pour obtenir une courbe d'ordre  $i$ , chaque sommet de la courbe de base est remplacé par la courbe d'ordre  $i - 1$ , avec une rotation et/ou réflexion appropriée pour s'adapter à la nouvelle courbe. Une courbe de Hilbert est généralement décrite comme limite d'une suite de courbes.



**Figure 3.5** : Trajectoire en Hilbert [34].

### 3.7 Les paquets et les points de diffusion

Après le choix de la trajectoire, il faut définir les positions (ou du temps) à partir desquelles les paquets seront envoyés ainsi que leur contenu. Le nœud mobile traverse la zone de déploiement suivant une courbe statique au choix et à chaque fois il arrive à la position bien défini selon la trajectoire choisi, il diffuse une information contenant la valeur de leur position actuel. Après à leur tour, les nœuds recevant ces informations exécutent l'algorithme de localisation qui sera détaillé dans la section suivante.

### 3.8 Architecture et fonctionnement

Il existe deux types de nœuds de capteurs dans le réseau, le nœud dont les positions sont inconnues et l'ancre mobile (beacon). Tous les nœuds capteurs ont le même rayon de communication radio. Les nœuds sont déployés aléatoirement dans la zone de captage. L'ancre mobile se déplace suivant la trajectoire prédéfinie en diffusant périodiquement son emplacement actuel pour aider les nœuds inconnus à s'auto-localiser. L'architecture de réseau est illustrée dans la figure 3.6.

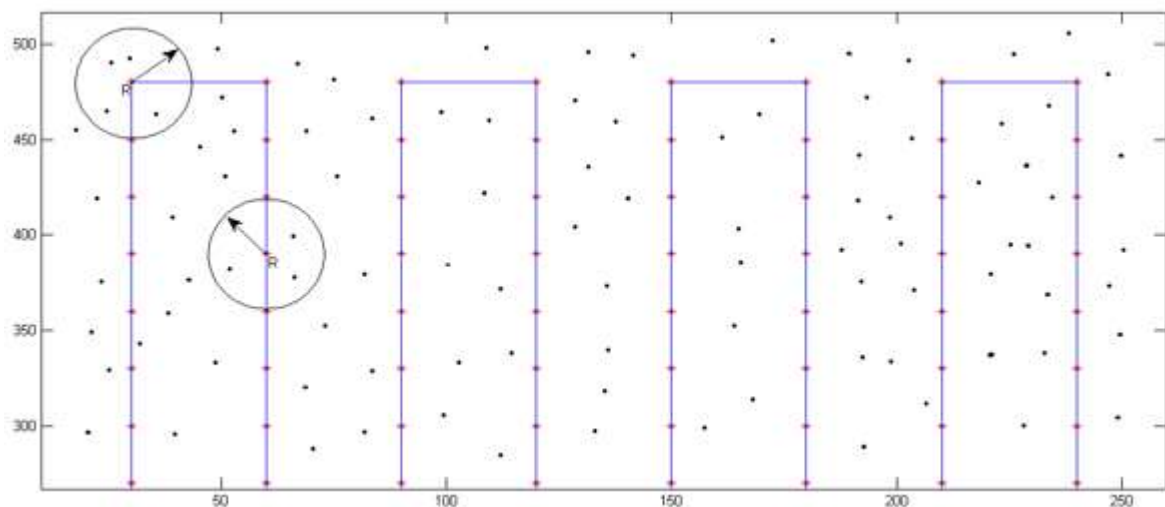


Figure 3.6 : Architecture du réseau.

#### 3.8.1 Le processus de localisation

Le processus de la localisation avec un nœud mobile est illustré dans la figure 3.7. Il est constitué de trois parties essentielles. La première est consacrée à la trajectoire, la deuxième

à l'estimation des distances entre l'ancre et les nœuds et la troisième est dédiée à l'algorithme de localisation.

La deuxième et la troisième partie du processus de localisation avec une ancre mobile sont très proches de celles d'autres processus à ancres fixes. L'estimation de la distance entre l'ancre et les nœuds est faite en utilisant les mêmes méthodes discutées précédemment comme la « RSSI », la « TDoA », etc. La dérivation des positions se fait soit par le mobile soit par les nœuds eux même. Dans la plupart des cas ce sont les nœuds qui calculent leurs positions pour réduire les communications.

La partie « trajectoire » est le cœur du processus de localisation avec une ancre mobile, et la plus difficile. Cette partie est divisée en deux sous-parties : (a) la définition de la trajectoire (quel chemin l'ancre mobile va suivre afin d'assurer la localisation de tous les nœuds), et (b) les paquets (qu'est-ce qu'ils contiennent ? À quels instants ou positions l'ancre doit-il le envoyer ?).

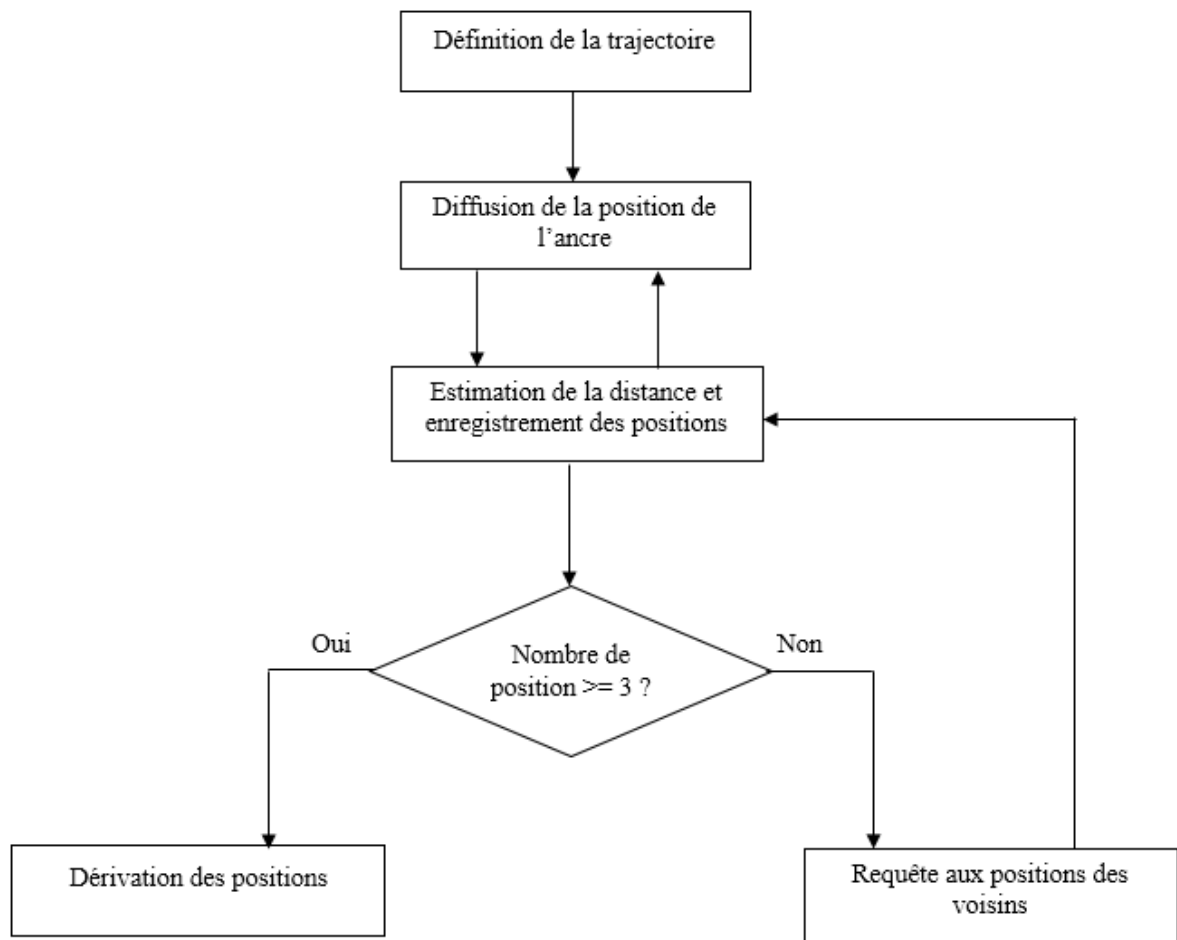


Figure 3.7 : Processus de la localisation en utilisant une ancre mobile.

### 3.8.2 Algorithme de localisation

Nous avons opté à utiliser un algorithme de localisation distribue dont chaque nœud de réseau calcule sa propre position. De plus, nous avons choisi la technologie RSSI vue en chapitre 2 pour estimer la distance entre les nœuds à cause de sa large utilisation dans le domaine de localisation dans le RCSF ainsi que sa simplicité et son coût faible puisqu'elle ne nécessite pas l'utilisation d'un matériel supplémentaire. D'ailleurs, la puissance du signal émis subit une dégradation en fonction de la distance. C'est-à-dire, plus la distance entre l'ancre et les capteurs est grande, plus la perte du signal est augmentée, plus la puissance de signal est faible. En bref, cet algorithme est simple et compose de trois phases :

- **Phase 1** : Le nœud ancre se déplace dans le réseau selon une trajectoire statique définie à l'avance (Scan), dans des points bien définis dits points de diffusion, ce nœud diffuse des paquets qui contiennent sa position.
- **Phase 2** : Le nœud inconnu qui reçoit un paquet enregistre la position du nœud ancre et la puissance du signal. En utilisant la valeur de la puissance du signal, le nœud calcule la distance estimée entre sa position et la position du nœud ancre mobile. Chaque nœud maintient une liste appelée ListePositionAncre, dans laquelle on trouve les positions du nœud ancre et la distance estimée par RSSI du message reçu ; il faut avoir au moins trois paquets non-colinéaires (de bonne qualité) pour calculer la position du nœud.
- **Phase 3** : Une fois le nœud inconnu a reçu les trois paquets, il calcule sa position en utilisant la méthode de trilatération.

#### Les performances de l'algorithme de localisation

Les performances de l'algorithme de localisation sont influencées par les facteurs suivants :

- (a) Portée de communication** : Un plus grand rayon de communication de l'ancre mobile couvre plusieurs nœuds inconnus. Ainsi, les nœuds inconnus ont plus de choix des paquets pour calculer leurs coordonnées.
- (b) Intervalle de diffusion** : Un intervalle de diffusion plus court signifie que l'ancre mobile diffusera son emplacement plus fréquemment, ce qui peut amener une meilleure performance de localisation.
- (c) La longueur du chemin** : Une longueur de trajet plus long signifie que l'ancre mobile a plus de possibilités de diffuser son emplacement et passer par plus des nœuds inconnus, Toutefois, il consomme plus d'énergie.

### **3.9 Conclusion**

Dans ce chapitre nous avons proposé d'implémenter une amélioration d'une méthode de localisation pour le RCSF, à l'aide d'une ancre mobile en trajectoire Scan. Cette approche est très prometteuse vu que l'ancre mobile est équipée d'un GPS et n'a pas de contrainte d'énergie. Cette ancre mobile peut être un robot mobile, un opérateur humain etc. En outre, le processus de localisation dans l'approche de l'ancre mobile dépend de deux principaux enjeux. Le premier enjeu est le choix de la trajectoire de l'ancre mobile. Le deuxième est le choix de l'algorithme de localisation.

## Chapitre 4

# Simulation et évaluations des performances

## 4.1 Introduction

La simulation est un processus qui consiste à concevoir un modèle du système réel étudié, mener des expérimentations sur ce modèle, interpréter les observations fournies par le déroulement du modèle et formuler des décisions relatives au système.

Il existe de nombreux simulateurs capables de gérer les réseaux de capteurs sans fil (NS-2, OMNet++, SENSE ...). En effet, ils sont parfois dédiés à des architectures spécifiques.

Nous allons en premier lieu, présenter le logiciel que nous avons utilisé pour les simulations qui est MATLAB, ensuite, nous allons présenter le contexte de simulation et les résultats pour le protocole de localisation étudié avec une discussion de résultat.

## 4.2 Environnement de la simulation

### 4.2.1 Choix du langage de programmation

Pour effectuer nos simulations, nous avons utilisé le logiciel MATLAB. Un choix motivé par le fait que la localisation peut être vue comme un problème purement géométrique. De plus MATLAB (MATriX LABoratory) est un logiciel de calcul numérique produit par MathWorks. Il est disponible sur plusieurs plateformes. MATLAB est un langage simple est très efficace, optimisé pour le traitement des matrices, d'où son nom, il optimise le code des programmes en utilisant des fonctions prédéfinies. Il permet l'affichage des courbes et des données, la mise en œuvre des algorithmes et la création des interfaces.

## 4.2.2 Variables descriptives du système

### 4.2.2.1 Coordonnées d'un nœud capteur

C'est l'affectation de deux valeurs aléatoires  $(x, y)$ , comprise entre 0 et la taille de la zone de déploiement  $(500 \times 500)$ , à chaque capteur sur un plan à deux dimensions.

### 4.2.2.2 Trajectoire d'une ancre mobile

Se déplace avec une trajectoire appelé scan et qui diffuse régulièrement sa position.

### 4.2.2.3 Distance entre deux capteurs

C'est la distance euclidienne entre un capteur de coordonnées  $(x, y)$  et un autre de coordonnées  $(x', y')$  :

$$d = \sqrt{(x - x')^2 + (y - y')^2}$$

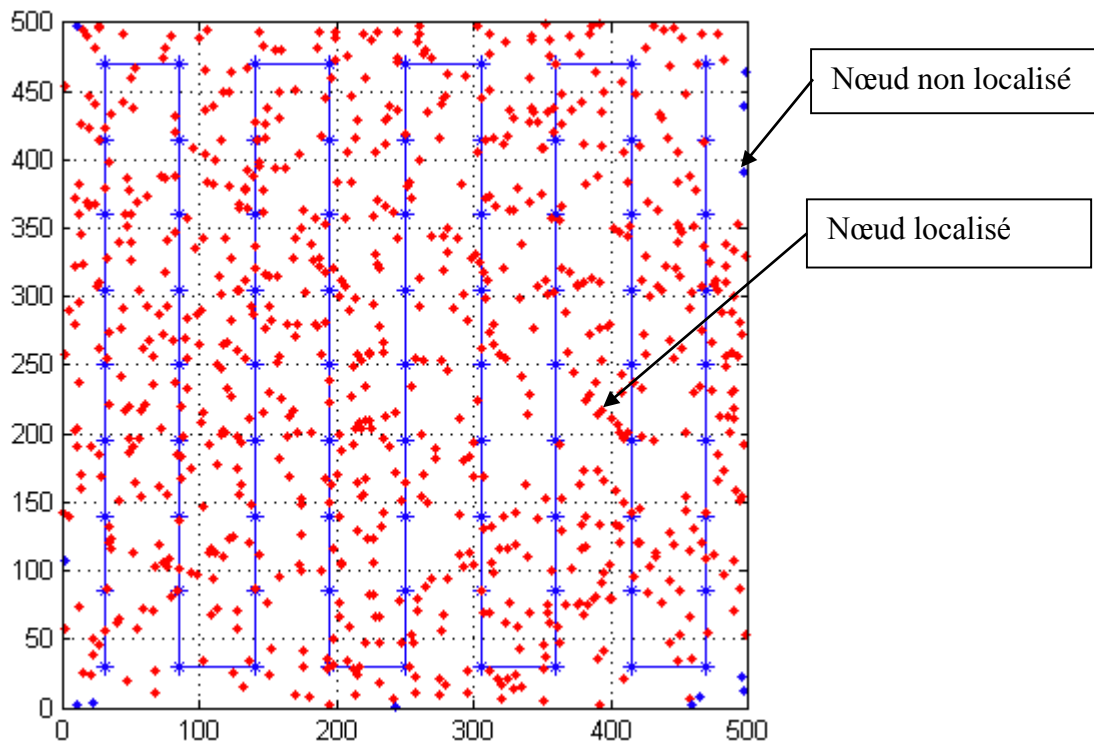
## 4.3 Paramètres de simulation

**Tableau 4.1** : Les paramètres de simulation.

Paramètre	Valeur
Dimension de la région	$500 \times 500 \text{ (m}^2\text{)}$
Le nombre de nœuds	800
Le temps entre diffusion	7 s
La vitesse de l'ancre	6 m/s
Nombre de diffusion	196
La portée de communication de l'ancre mobile	70 (m)
La portée de communication de nœud capteur	30 (m)
Rayon minimum de la trajectoire en Spiral	100 (m)
Rayon maximum de la trajectoire en Spiral	250 (m)
Coordonnées X du centre de la région	250 (m)
Coordonnées Y du centre de la région	250 (m)

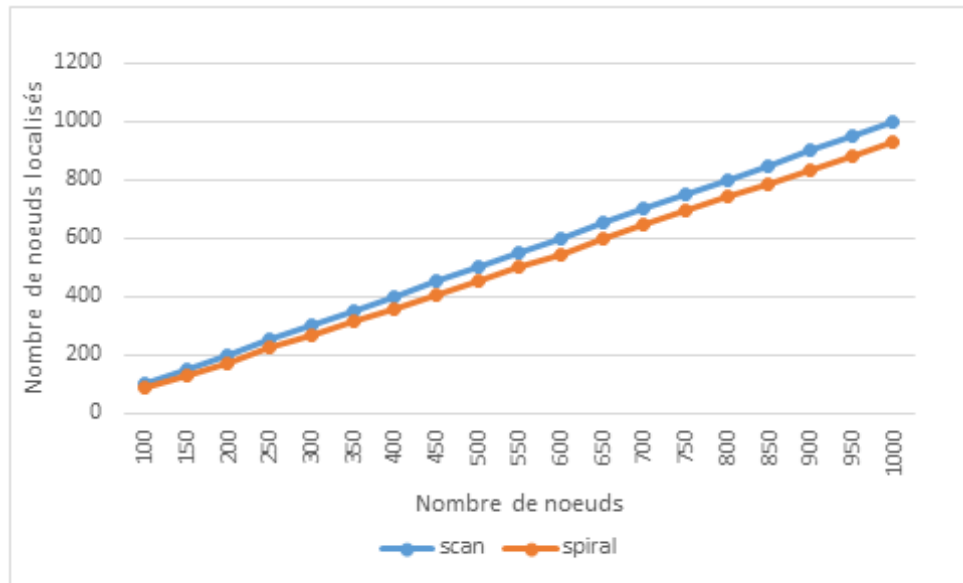
## 4.4 Résultats de simulation

Les données dans la figure (4.1) sont tirées à partir d'un réseau de 800 à 1000 nœuds déployés aléatoirement sur une surface carré de 500×500 mètre, fournissant un réseau entièrement connecté. Nous supposons que tous les nœuds ont une position fixe durant toute la période de simulation.



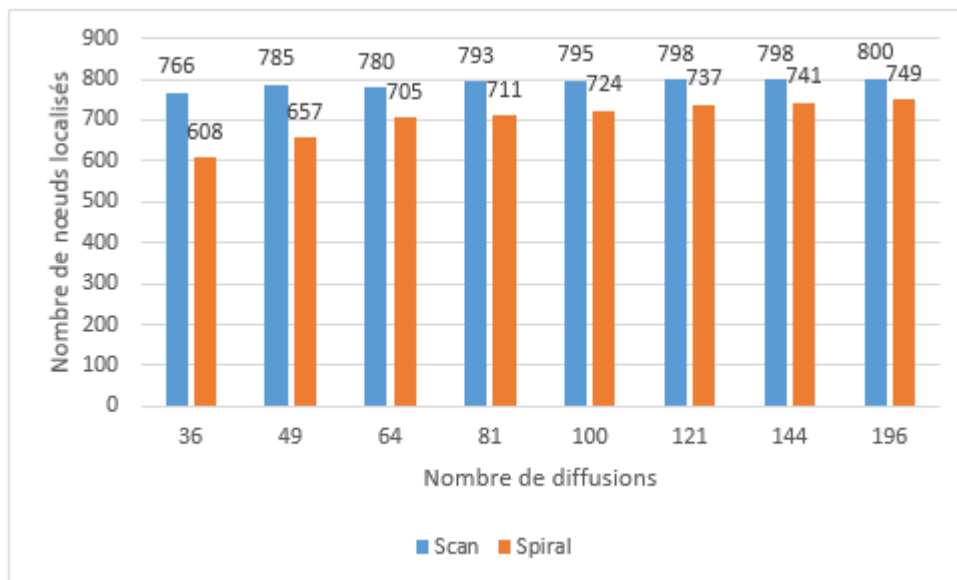
**Figure 4.1** : Environnement de la solution de localisation proposée.

La figure 4.2 représente les résultats de simulation du nombre de nœuds localisés en fonction de nombre de nœuds déployés, le nombre de nœuds varie entre 100 et 1000 sur une surface de 500\*500 (m<sup>2</sup>).



**Figure 4.2 :** Nombre de nœuds localisés en fonction de nombre de nœuds sur la surface.

Les résultats illustrés dans la figure 4.2, montrent que le nombre de nœuds localisés dans notre approche (trajectoire scan) est meilleur que celle de la trajectoire en spirale, Nous avons remarqués que lorsque en augmente le nombre de nœuds sur la zone d'intérêt le nombre de nœuds localisés devient très important, et l'écart se creuse davantage entre les deux approches.



**Figure 4.3 :** Nombre de nœuds localisés en fonction de nombre de diffusion.

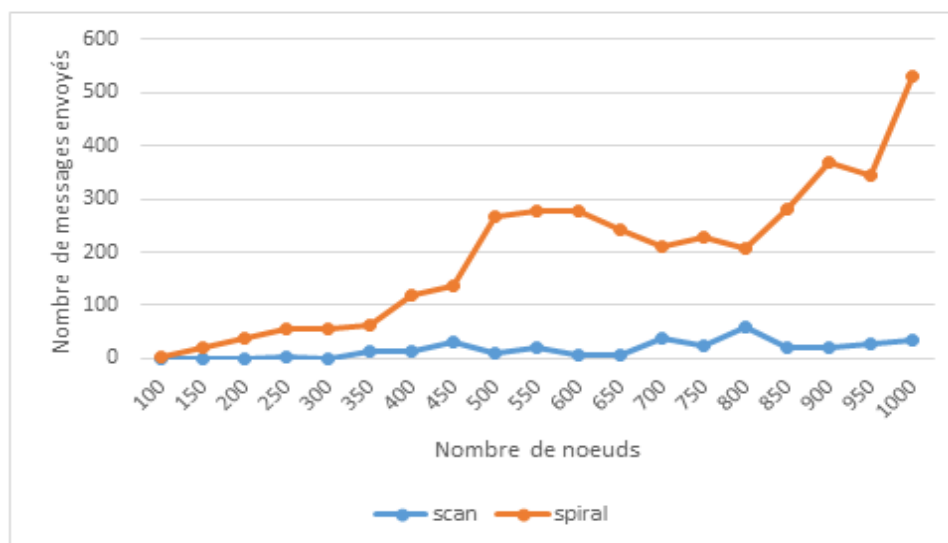
Nous avons effectué plusieurs simulations en variant le nombre de diffusion, les résultats de ces simulations ont montré que le nombre de nœuds localisés diffère d'une trajectoire à une

autre. Il est à noter que dans la figure 4.3, l'écart entre les deux approches se resserre de plus en plus en augmentant le nombre de diffusion. Ceci peut être expliqué par le fait que lorsque nous augmentons le nombre de diffusion alors la zone d'intérêt est de plus en plus couverte.

Les résultats de simulation ont montré la pertinence du choix de la trajectoire en scanne. Cette dernière est un facteur essentiel pour la couverture d'un RCSF.

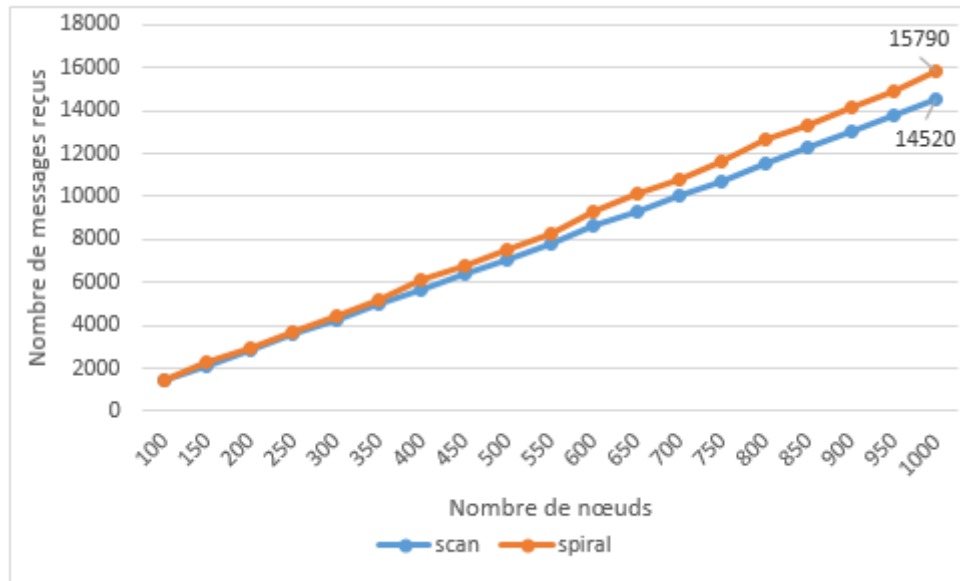
La figure 4.4 compare les deux approches, par rapport au nombre de messages envoyés, en fonction du nombre de nœuds déployés.

Nous remarquons qu'avec la trajectoire en spirales, le nombre de messages envoyés est très élevé par rapport à notre approche (en scan). Ceci est expliqué par le fait que lorsque le nombre de messages broadcast reçus par un nœud n'est pas suffisant pour se localiser, les nœuds non localisés sollicitent leurs voisins déjà localisés afin de récupérer leurs positions, en utilisant la technique de trilatération, qui nécessite trois voisins au minimum, pour calculer leurs positions.



**Figure 4.4** : Nombre de messages envoyés en fonction de nombre de nœuds.

La Figure 4.5 compare le nombre de messages reçus. Dans laquelle nous constatons que l'approche basée sur une trajectoire en spirale, présente un nombre de messages reçus supérieur à celui de l'approche basée sur une trajectoire en scanne. Partant d'une différence infime au départ mais par la suite l'écart devient de plus en plus considérable, surtout à partir de la valeur 400, pour culminer à la fin un écart de 1270 messages.



**Figure 4.5** : Nombre de messages reçus en fonction de nombre de nœuds.

## 4.5 Conclusion

Dans ce chapitre, nous avons fait une série de simulation sur quelques paramètres. Les résultats de simulation ont permis de déduire que le nombre de nœuds déployé sur la zone d'intérêt et le nombre de diffusion peuvent influencer sur la qualité de la localisation en termes de nombre de nœuds localisés. D'après les résultats obtenus, notre trajectoire est mieux adaptée aux réseaux de capteur sans fil dont la région d'intérêt est sous forme carrée.

---

## Conclusion générale

---

Les réseaux de capteurs sans fil ont un large potentiel et constituent un sujet de recherche innovant ainsi qu'un outil d'une grande utilité dans plusieurs domaines. C'est sans aucun doute une technologie qui va nous accompagner pour les prochaines années, et faire ainsi partie de notre vie quotidienne. Cependant, il y a encore beaucoup de problèmes qui doivent être abordés pour un fonctionnement efficace de ces réseaux dans des applications réelles. Parmi les problèmes fondamentaux dans ces réseaux de capteurs, nous en citons le problème de la localisation qui est une nécessité absolue à laquelle des solutions adéquates doivent être proposées.

Dans ce mémoire, nous avons mis en place une méthode de localisation qui se base sur une ancre mobile qui traverse la région d'intérêt suivant un chemin défini (SCAN) en diffusant des informations pour aider les nœuds à se localiser par l'application de la technique de trilatération.

Nous avons évalué, par une série de simulations, les performances de deux types de trajectoires, par rapport au nombre de nœuds localisés et le nombre de messages échangés en utilisant le logiciel MATLAB. Au cours de ce travail, qui a commencé par une étude bibliographique, nous avons découvert un champ de recherche très intéressant qui touche presque tous les domaines d'application. Cela nous a permis de nous initier à la recherche, et nous espérons avoir apporté une contribution, bien que modeste, à ce domaine en pleine évolution.

Comme perspective, nous allons continuer à améliorer ce travail, surtout en prenant d'autres métriques comme la consommation d'énergie, la scalabilité, et la précision de localisation. Nous envisageons tout cela dans nos futurs travaux de recherche, qu'ils soient d'ordre individuel, professionnel ou académique.

---

# Bibliographie

---

- [1] B.Nawel, Auto-localisation dans les réseaux de capteurs sans fil. Master en informatique, Université : MOHAMED BOUDIAF - M'SILA, 2015 /2016.
- [2] S.MOAD. Optimisation de la consommation d'énergie dans les réseaux de capteurs sans fil. Master en informatique, Université : IFSIC-Rennes1, 2007/2008.
- [3] Y.Challal, 'systèmes intelligents de transport Réseaux de capteurs sans fil', version SIT60, 2008.
- [4] P.Levis, "TinyOS Programming manual", 27 October 2006.
- [5] A.Dunkels, B.Gronval, and T.Voigt, "Contiki-A: Lightweight and Flexible Operating System for Tiny Networked Sensors", in Proceedings of the 29th annual IEEE International Conference on Local Computer Network, November 2004.
- [6] D.NGOM "Optimisation de la durée de vie dans les réseaux de capteurs sans fil sous contraintes de couverture et de connectivité" Thèse de Doctorat en informatique, École Doctorale Mathématique-Informatique de l'université Haute Alsace (France), 2016.
- [7] B.Sahraoui, "La géolocalisation dans les réseaux de capteurs sans fil", Mémoire de fin d'études cycle ingénieur d'état en Informatique, 2011.
- [8] Y.Challal, "Réseaux de capteurs sans fils", version 1, 2016.
- [9] C.BAOUICHE, "Une solution tolérante aux délais pour des applications de localisation et de traçabilité a posteriori en milieux confinés", Thèse de Doctorat en informatique, université blaise pascal, 2012.
- [10] B.Sundararaman, U.Buy, and A.Kshemkalyani, Clock Synchronization in Wireless Sensor Networks : A Survey. Ad Hoc Networks, 2005.
- [11]. S.Ganeriwala, R.Kumar, and S. Adlakha, Network-wide Time Synchronization in Sensor Networks. Networked and Embedded Systems Lab (NESL), 2003.
- [12]. V.Greunen and J.Rabaey. Lightweight Time Synchronization for Sensor Networks, in Proceedings of the 2nd ACM international conference on Wireless sensor networks and applications, 2003.

- 
- [13]. K.Lee and J.Eidson, IEEE-1588 Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems, in 34 th Annual Precise Time and Time Interval (PTTI) Meeting, 2002.
- [14]. J.Elson, L.Girod, and D.Estrin, Fine-Grained Network Time Synchronization using Reference Broadcasts. SIGOPS Oper. Syst. Rev, 2002.
- [15]. K.Romer. Time synchronization in ad hoc networks, in International symposium on Mobile ad hoc networking and computing MobiHoc, 2001.
- [16]. M.Sichitiu and C.Veerarittiphan. Simple Accurate Time Synchronization for Wireless Sensor Networks. in Wireless Communications and Networking, 2003.
- [17] A.R. Kulaib, R.M. Shubair, M. A. Al-Qutayri, and J. W. P. Ng, 'An overview of localization techniques for wireless sensor networks', 2011.
- [18] M.Terwilliger, 'Localization in wireless sensor networks,' Ph.D. dissertation, Western Michigan University, Kalamazoo, MI, USA, 2006.
- [19] H.Karl, A.Willig, 'Protocols and Architectures for Wireless Sensor Networks', John Wiley & Sons, Ltd. ISBN : 0-470-09510-5.
- [20] J.Hightower and G. Borriello, 'Location Systems for Ubiquitous Computing', IEEE Computer Journal, 2001.
- [21] A.Makhoul, 'Réseaux de capteurs : localisation, couverture et fusion de données', grade de DOCTEUR, l'Université de Franche-Comté, 14 novembre 2008.
- [22] M.Kara, 'Réseau de capteurs sans fil : étude en vue de la réalisation d'un récepteur GPS différentiel à faible coût', Réseaux et télécommunications, Université Blaise Pascal - Clermont-Ferrand II, 2009.
- [23] R.Dalce, T.Val, A. V. D.Bossche. Comparison of indoor localization systems based on wireless communications, Wireless Engineering and Technology, october 2011.
- [24]. S.Capkun, M.Hamdi, and J.Hubaux, GPS Free Positioning in Mobile Ad-Hoc Networks, international Conference on System Sciences, 2001.
- [25] L.Peralta and M.Rodriguez. Collaborative Localization in Wireless Sensor Networks, in International Conference on Sensor Technologies and Applications SENSORCOMM, 2007.
- [26] P.Rong and M.Sichitiu., Angle of Arrival Localization for Wireless Sensor Networks, Sensor and Ad Hoc Communications and Networks, 2006.
- [27] H.Du and P.Lee, Simulation of Multi-Platform Geolocation Using a Hybrid TDOA/AOA Method, Technical Memorandum DRDC Ottawa TM 2004-256, Décembre 2004.

- [28] A.Roxin, J.Gaber, and M.Wack, Survey of Wireless Geolocation Techniques. IEEE Globecom Workshops, 2007.
- [29] A.Hopper, A.Harter, and T.Blackie, The active badge system, in Conference on Human factors in computing systems, 1993.
- [30] N.Bulusu, J.Heidemann, and D.Estrin, GPS-less Low Cost Outdoor Localization For Very Small Devices, IEEE Personal Communications Magazine, 2000.
- [31] D.Fox, W.Burgard, and F.Dellaert, Monte Carlo Localization: Efficient Position Estimation for Mobile Robots. National Conference on Artificial Intelligence, 1999.
- [32] W.Larbi-Mezeghrane, L.Bouallouche-Medjkoune et A.larbi « localization approche based on multilateration and geometric center of cloud of points in WSN ». In : First International Conference on Electronic Engineering and Renewable Energy Springer Conference Saidia 2018 – Morocco.
- [33] Han, Guangjie, Huihui Xu, Trung Q. Duong, Jinfang Jiang, and Takahiro Hara. "Localization algorithms of wireless sensor networks: a survey." *Telecommunication Systems* 52, no. 4 (2013): 2419-2436.
- [34] Koutsonikolas, Dimitrios, Saumitra M. Das, and Y. Charlie Hu. "Path planning of mobile landmarks for localization in wireless sensor networks." *Computer Communications* 30, no. 13 (2007): 2577-2592.

## Résumé

La localisation est l'un des problèmes importants dans les réseaux de capteurs sans fil, qui sert à trouver les positions des nœuds suite à un déploiement aléatoire. Elle est nécessaire non seulement pour pouvoir localiser les événements mais aussi elle fournit un soutien fondamental pour de nombreux protocoles et applications de localisation. L'utilisation du GPS est une solution matérielle qui offre une bonne précision, mais les contraintes de coût et de consommation d'énergie rendent impossible d'équiper chaque nœud capteur dans le réseau avec un système de positionnement global (GPS), en particulier pour les réseaux de capteurs à grande échelle. Une méthode prometteuse pour localiser les nœuds est d'utiliser une ancre mobile équipé d'unités GPS se déplaçant dans la zone d'intérêt et diffusant périodiquement son emplacement actuel afin d'aider les nœuds inconnus à s'auto localiser. Dans ce mémoire, nous étudions une méthode de localisation basée sur une ancre mobile avec la Trilatération. Nous examinons deux différentes trajectoires de l'ancre mobile tels que : Scan, Spiral. Afin d'analyser cet algorithme selon le nombre de nœuds localisés, nous avons utilisés un logiciel MATLAB pour la simulation.

**Mots-clés :** réseaux de capteurs sans fil, localisation, ancre mobile, trajectoire.

## Abstract

Localization is one of the major problems in wireless sensor networks, which is used to find the positions of the nodes following a random deployment. It is necessary not only to locate events but also provides fundamental support for many localization protocols and applications. The use of GPS is a hardware solution that offers good accuracy, but the constraints of cost and energy consumption make it impossible to equip each sensor node in the network with a global positioning system (GPS), in particularly for large scale sensor networks. A promising method for locating unknown nodes is to use a mobile anchor with GPS units moving in the area of interest and periodically diffusing its current location to help unknown nodes self-locate. In this thesis, we study a method of localization based on a mobile anchor with Trilateration. We examine two different trajectories of the mobile anchor such as: Scan, Spiral. In order to analyze this algorithm according to the number of localized nodes, we used MATLAB software for the simulation.

**Keywords:** wireless sensor networks, localization, mobile anchor, trajectory.