

République Algérienne Démocratique et Populaire  
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université A/Mira de Béjaïa  
Faculté des Sciences Exactes  
Département d'Informatique  
Option : Réseaux et Système Distribués



MÉMOIRE DE MASTER RECHERCHE

## Thème

---

Intelligence des Objets Volants : Algorithme de  
Simulation pour le Tracking des Objets Volants

---

Présenté par :

Mlle. *HEMECHE* Selma

Devant le jury composé de :

Président	M. MIR Foudil	M.A.A	U. A/Mira Béjaïa.
Examineur	Dr. AMROUN Kamel	M.C.A	U. A/Mira Béjaïa.
Examineur	Dr. SLIMANI Hachem	M.C.A	U. A/Mira Béjaïa.
Encadreur	Dr. BOUNCEUR Ahcene	H.D.R	U. Bretagne Occidentale.
Co-Encadreur	M. BEZOUÏ Madani	M.A.A	U. M/Bougara Boumerdes.
Invité	M. ABBACHE Bournane	M.A.A	U. A/Mira Béjaïa.

Béjaia, Juin 2016.

# Remerciements

*En préambule à ce travail, je remercie ALLAH le tout puissant et miséricordieux, qui m'a donné la force et la patience d'accomplir ce modeste travail.*

*Je voudrais tout d'abord adresser toute ma gratitude à mon directeur de recherche **M. BOUNCEUR Ahcene**, pour sa patience, sa disponibilité et surtout ses judicieux conseils, qui ont contribué à alimenter ma réflexion.*

*Je remercie également **M. BEZOUI Madani**, mon co-directeur de recherche, pour tout le soutien, l'aide, l'orientation, la guidance qu'il m'a apportés durant ainsi que pour ses précieux conseils et ses encouragements lors de la réalisation de mon mémoire.*

*Je tiens à exprimer ma profonde reconnaissance et toutes mes pensées de gratitude à **M.ABBACHE Bournane**, qui m'a accompagné de près durant tout ce travail, pour sa disponibilité, pour les conseils précieux qu'il m'a prodigués tout au long de la réalisation de ce modeste travail.*

*Je remercie tous les membres du jury pour avoir bien voulu donner de leur temps pour lire ce travail et faire partie des examinateurs.*

*Mes remerciements les plus vifs vont à mes chers parents pour leurs soutiens inconditionnels tout au long de mes études.*

*Enfin, je remercie également mes amis, et toutes les personnes qui ont contribué de près ou de loin à la réalisation de ce projet de fin d'études.*

# Dédicaces

*Je dédie ce modeste travail À :*

*Mes très chers parents*

*Vous représentez pour moi le symbole de la bonté par excellence, la source de tendresse et l'exemple du dévouement qui n'a pas cessé de m'encourager et de prier pour moi. Aucune dédicace ne saurait être assez éloquente pour exprimer ce que vous méritez pour tous les sacrifices que vous n'avez cessé de me donner depuis ma naissance, Je vous dédie ce travail en témoignage de mon profond amour. Puisse Dieu, le tout puissant, vous préserver et vous accorder santé, longue vie et bonheur.*

*Mes très chers frères et sœurs*

*Abdelhalim, Assia, Aderrahmen, Oussama, Asma, Moad, Sarah, Ishak, SELSABIL En témoignage de l'attachement, de l'amour et de l'affection que je porte pour vous.*

*Ma chère Sarah, son époux Adel et mon chère boutchou abdelwadoud*

*Pour Toi ma Sarah qui occupe une place unique dans ma vie. Je n'oublierai jamais ton amour et attention envers moi. Que dieu te procure tout le bonheur que tu mérites.*

*Mon très cher frère Ishak*

*Mon cher petit frère, Je te souhaite un avenir plein de joie, de bonheur, de réussite et de sérénité. Je t'exprime à travers ce travail mes sentiments de fraternité et d'amour.*

*Mes chers petits neveux et nièces*

*Yacine, djida, Amina, Mohamed, Iméne, Islam, Ikram, lina, Abdelhafid, Hadjer, Bassema, Yacoub, Abdelwadoud et Abdelhak.*

*Mes très chers amis de toujours*

*Dalia, Soraya, Ferial, Amine, Amirouche, Zobida, Sarah, Chahra. En souvenir de notre sincère et profonde amitié et des moments agréables que nous avons passés ensemble.*

# Table des matières

<b>Table des matières</b>	<b>iii</b>
<b>Table des figures</b>	<b>iv</b>
<b>Liste des algorithmes</b>	<b>v</b>
<b>Liste des abréviations</b>	<b>v</b>
<b>Liste des tableaux</b>	<b>vii</b>
Introduction générale . . . . .	1
<b>1 Généralités sur les réseaux de capteurs</b>	<b>3</b>
1.1 Introduction . . . . .	3
1.2 Réseaux de capteurs sans fil . . . . .	4
1.2.1 Architecture physique d'un capteur sans fil . . . . .	4
1.2.1.1 Unité de capture . . . . .	5
1.2.1.2 Unité de traitement . . . . .	5
1.2.1.3 Unité de communication . . . . .	5
1.2.1.4 Batterie . . . . .	5
1.2.2 Architecture d'un réseau de capteurs sans fil . . . . .	5
1.2.3 Domaines d'application . . . . .	6
1.2.3.1 Applications militaires . . . . .	7
1.2.3.2 Applications environnementales . . . . .	7
1.2.3.3 Applications médicales . . . . .	7
1.2.3.4 Applications domotiques . . . . .	8
1.2.3.5 Applications industrielles . . . . .	8
1.2.4 Caractéristiques et Contraintes . . . . .	8
1.2.4.1 Puissance de calcul . . . . .	8

1.2.4.2	Communication . . . . .	9
1.2.4.3	Énergie . . . . .	9
1.3	Vers les réseaux de capteurs multimédia sans fil . . . . .	9
1.3.1	Caractéristiques et contraintes . . . . .	10
1.3.1.1	Capture et traitement d'image . . . . .	10
1.3.1.2	Besoin en mémoire . . . . .	10
1.3.1.3	Transmission de données multimédia . . . . .	11
1.3.2	Domaines d'application . . . . .	11
1.3.2.1	Surveillance multimédia . . . . .	11
1.3.2.2	Gestion et contrôle de trafic . . . . .	11
1.4	Conclusion . . . . .	12
<b>2</b>	<b>État de l'art sur les algorithmes de Suivi de cible dans les réseaux de capteurs</b>	<b>13</b>
2.1	Introduction . . . . .	13
2.2	Tracking des objets . . . . .	14
2.3	Classification des algorithmes de suivi dans la littérature . . . . .	15
2.3.1	Solutions basées sur une architecture hiérarchique . . . . .	16
2.3.2	Solutions basées sur une architecture en cluster . . . . .	24
2.3.2.1	Clustering statiques . . . . .	24
2.3.2.2	Clustering dynamiques . . . . .	24
2.3.3	Solutions prédictives . . . . .	26
2.3.4	Solutions hybrides . . . . .	29
2.4	Métriques pour l'analyse des protocoles de poursuite de cible . . . . .	31
2.5	Conclusion . . . . .	34
<b>3</b>	<b>Proposition</b>	<b>35</b>
3.1	Introduction . . . . .	35
3.2	Motivations . . . . .	35
3.3	Description du système . . . . .	36
3.3.1	Définitions et hypothèses . . . . .	36
3.3.2	Aperçu du protocole . . . . .	38
3.3.3	Processus de suivi . . . . .	39
3.3.3.1	Phase de découverte de face . . . . .	39
3.3.3.2	Phase de découverte de cible . . . . .	39
3.3.3.3	Phase de détection . . . . .	40

3.3.3.4	Phase de suivi . . . . .	41
3.4	Raffinement . . . . .	41
3.4.1	Pseudo algorithme de suivi d'objet après raffinement . . . . .	45
<b>4</b>	<b>Simulation</b>	<b>47</b>
4.1	Introduction . . . . .	47
4.2	Environnement de simulation . . . . .	47
4.2.1	Contexte et paramètres de simulation . . . . .	48
4.3	Simulation et évaluation de performances . . . . .	50
4.3.1	Métrique à évaluer . . . . .	50
4.3.2	Résultats de la simulation . . . . .	50
4.4	Conclusion . . . . .	54
	Conclusion générale . . . . .	55
	<b>Bibliographie</b>	<b>57</b>

# Table des figures

1.1	<i>Composants physiques d'un capteur sans fil [8]. . . . .</i>	4
1.2	<i>Architecture d'un Réseau de Capteur Sans Fil [5] . . . . .</i>	6
1.3	<i>Domaines d'application des réseaux de capteurs sans fil [8]. . . . .</i>	7
2.1	<i>Classification des algorithmes de suivi. . . . .</i>	16
2.2	<i>Exemple de construction d'arbre DAB pour construire une hiérarchie de message d'élagage [18]. . . . .</i>	19
2.3	<i>Arête de Gabriel [30]. . . . .</i>	20
2.4	<i>Exemple d'un graphe planaire [30]. . . . .</i>	21
2.5	<i>Domaines d'application des réseaux de capteurs sans fil. . . . .</i>	26
2.6	<i>Protocole de suivi de cible à base de cluster hybride [32]. . . . .</i>	30
3.1	<i>Arête de Gabriel [30]. . . . .</i>	37
3.2	<i>Exemple d'un graphe planaire. . . . .</i>	38
3.3	<i>Un aperçu pour le suivi d'objet [30]. . . . .</i>	39
3.4	<i>Exemple de detection d'objet. . . . .</i>	43
3.5	<i>Détection d'objet Après l'amélioration du protocole. . . . .</i>	44
4.1	<i>Modèle d'expérimentation. . . . .</i>	48
4.2	<i>Variation de la consommation d'énergie en fonction du nombres de nœuds dans le réseau. . . . .</i>	51
4.3	<i>Variation de la consommation d'énergie en fonction du rayon de communication. . . . .</i>	52
4.4	<i>Nombres de messages échangés en fonction du temps simulation. . . . .</i>	53
4.5	<i>Le nombre de nœuds vivants dans le réseau en fonction du temps de simulation. . . . .</i>	54



# Liste des abréviations

<b>RCSF</b>	<b>R</b> éseau de <b>C</b> apteur <b>S</b> ans <b>F</b> il.
<b>CAN</b>	<b>C</b> onvertisseur <b>A</b> nalogique <b>N</b> umérique.
<b>STUN</b>	<b>S</b> calable <b>T</b> racking <b>U</b> sing <b>N</b> etworked sensors.
<b>DCTC</b>	<b>D</b> ynamic <b>C</b> onvoy <b>T</b> ree- <b>B</b> ased <b>C</b> ollaboration.
<b>DAT</b>	<b>D</b> eviation <b>A</b> voidance <b>T</b> ree.
<b>DAB</b>	<b>D</b> rain <b>A</b> nd <b>B</b> alance.
<b>ZDAT</b>	<b>Z</b> one-based <b>D</b> eviation <b>A</b> voidance <b>T</b> ree.
<b>DAT</b>	<b>D</b> ynamic <b>O</b> bject <b>T</b> racking.
<b>OCO</b>	<b>O</b> ptimized <b>C</b> ommunication <b>O</b> rganization.
<b>IDSQ</b>	<b>I</b> nformation <b>D</b> riven <b>S</b> ensor <b>Q</b> uerying.
<b>RAR</b>	<b>R</b> educed <b>A</b> rea <b>R</b> eporting.
<b>CH</b>	<b>C</b> luster <b>H</b> ead.
<b>GPS</b>	<b>G</b> lobal <b>P</b> ositioning <b>S</b> ystem.
<b>DPR</b>	<b>D</b> ual <b>P</b> rediction <b>R</b> eporting.

<b>HCTT</b>	<b>H</b> ybride <b>C</b> lustering <b>T</b> arget <b>T</b> racking.
<b>HCAT</b>	<b>H</b> ybride <b>C</b> lustering for <b>A</b> coustic <b>T</b> racking.
<b>HPS</b>	<b>H</b> ierarchical <b>P</b> rediction <b>S</b> trategy.
<b>PET</b>	<b>P</b> rediction based <b>E</b> nergy <b>T</b> arget tracking.
<b>PES</b>	<b>P</b> rediction based <b>E</b> nergy <b>S</b> aving sckeme.
<b>LEACH</b>	<b>L</b> ow <b>E</b> nergy <b>A</b> adaptive <b>C</b> lustering <b>H</b> ierarchy .
<b>BS</b>	<b>B</b> ase <b>S</b> tation.

# Liste des tableaux

2.1	Comparaison des solutions de suivi. . . . .	34
4.1	Paramètres de simulation. . . . .	49

# Introduction générale

Les dernières avancées technologiques dans le domaine des communications sans fil ont permis l'émergence des réseaux de capteurs sans fil. Ces derniers sont composés d'un ensemble de nœuds, capables de capturer et de mesurer certains événements et phénomènes physiques voir d'interagir avec l'environnement où ils sont déployés. Ces réseaux se prêtent fort bien à de nombreuses applications dans des domaines tels que : les applications militaires, la surveillance environnementale, industrielle et médicale, etc.

Nous nous intéressons aux applications de surveillance et en particulier au suivi de cible mobile. Cette application s'avère très utile, par exemple, pour détecter la présence d'un intrus et enregistrer ses déplacements dans une zone sensible ou encore pour l'assistance aux personnes munies d'un appareil avec interface radio.

Le suivi ou le tracking d'objets mobiles est une application qui consiste à détecter et à localiser une cible mobile tout au long de son évolution dans une zone d'intérêt. Pour la précision du suivi, deux ou plusieurs capteurs sont nécessaires pour détecter l'objet simultanément. De plus, le traitement des données étant distribué, le relai et le partage des informations récoltées doivent être pris en compte.

Contrairement aux systèmes de surveillance classiques qui nécessitent une infrastructure fixe, les réseaux de capteurs sans fil sont aussi faciles à installer qu'à désinstaller. Cependant, étant autonomes et donc contraints énergétiquement, les capteurs ne peuvent pas rester actifs tout le temps. Afin de résoudre ce problème, l'idée simple est d'activer uniquement les capteurs qui se trouvent sur la trajectoire de la cible au moment où cette dernière est à leur portée.

Ce mémoire s'articule autour de quatre chapitres. Dans le premier chapitre, nous présentons les réseaux de capteurs sans fil, leurs domaines d'application et leurs caractéristiques physiques et architecturales.

Le deuxième chapitre est un état de l'art sur le suivi de cibles mobiles dans les réseaux de capteurs. Nous décrivons ce qu'est le suivi de cible mobile, nous classifions les solutions existantes dans la littérature et discutons les métriques d'évaluation de ces dernières.

Dans le troisième chapitre, nous y exposons, d'une part, les contraintes liées au suivi d'objets et, d'autre part, nous décrivons en détail notre contribution, et cela, en élaborant un algorithme distribué de suivi de cibles mobiles implémenté sur un réseau hiérarchique.

Le quatrième chapitre est, quant à lui, consacré à l'évaluation des performances de l'algorithme proposé. Nous présentons l'environnement de simulation. Par la suite nous donnons les résultats sous forme de graphes de plusieurs simulations, effectuées pour obtenir des mesures, nous permettons de voir l'efficacité de l'algorithme.

Nous terminerons par une conclusion générale qui ouvre sur quelques perspectives.

---

# Généralités sur les réseaux de capteurs

---

## 1.1 Introduction

L'apparition des nouvelles technologies ainsi que les progrès effectués dans le domaine des réseaux et du traitement de l'information ont entraîné l'apparition de nouveaux outils et objets tels que les réseaux de capteurs et leurs applications. Depuis quelques années, les réseaux de capteurs sans fil (RCSF) s'ouvrent à plusieurs domaines d'applications : militaire, sécurité civile (surveillance des risques d'incendie, des catastrophes naturelles, des centrales nucléaires, ...), transport (automobile, ferroviaire, aéronautique, spatial), industriel (contrôle de la qualité de production, surveillance des lieux, ...), environnement, etc. L'environnement intègre de plus en plus des capteurs sans fil à faible consommation énergétique [4].

Ces petits dispositifs sont interconnectés les uns aux autres pour former des réseaux de capteurs sans fil et sont ensuite déployés dans une zone d'intérêt afin de gérer des tâches spécifiques tels que la détection de flux de radiation ou ; le suivi d'objets en déplacement et leur positionnement. L'attrait croissant suscité par les réseaux de capteurs sans fil aussi bien dans monde de la recherche que celui de l'industrie peut aisément être compris si on considère le large éventail d'applications qu'offre ce type d'infrastructures [3, 25].

Dans la section suivante, nous nous attachons à décrire les bases nécessaires à la compréhension des réseaux de capteurs sans fil. L'architecture, les applications et les spécificités de ce type de réseaux sont les points abordés. La section 1.3 décrit les réseaux de capteurs multimédia sans fil et les contraintes spécifiques liées aux données qui y transitent.

## 1.2 Réseaux de capteurs sans fil

**Définition 1.2.1.** [21] Un RCSF est un type spécial de réseau ad-hoc défini par un ensemble coopérant de capteurs déployés dans une zone géographique appelée zone de captage ou zone d'intérêt, afin de surveiller un phénomène quelconque et de récolter des données d'une manière autonome. Les capteurs utilisent une communication sans fil (Zig Bee ou Wifi) pour acheminer les données captées avec un routage multi-sauts vers un nœud considéré comme "point de collecte", appelé station de base ou nœud puits. Cette dernière peut être connectée à une machine puissante via internet, réseaux GPRS ou par satellite. L'utilisateur peut adresser ses requêtes aux capteurs en précisant l'information d'intérêt.

**Définition 1.2.2.** [19] Un capteur sans fil est un petit dispositif électronique capable de mesurer une valeur physique environnementale (température, lumière, pression, humidité, vibration, etc.), et de la communiquer à un centre de contrôle via une station de base.

Chaque capteur assure les trois principales fonctions de base qui sont : l'acquisition de données, les traitements sur ces données et leurs communications aux stations de bases. La figure 1.1 montre quelques capteurs.

### 1.2.1 Architecture physique d'un capteur sans fil

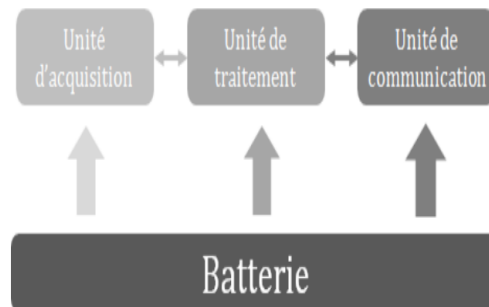


FIGURE 1.1 – Composants physiques d'un capteur sans fil [8].

Les capteurs sans fil sont des petits dispositifs généralement dotés de quatre composants élémentaires : une unité de capture, une unité de traitement, une unité de communication et une batterie [3]. La figure 1.1 illustre les composants physiques élémentaires d'un capteur sans fil.

Les capteurs peuvent aussi être dotés d'une unité de localisation pour déterminer leur

position ou encore d'une unité de mobilité leur permettant de se déplacer. Ces composants sont optionnels et dépendent de l'application visée [8].

#### 1.2.1.1 Unité de capture

L'unité de capture ou d'acquisition est comme son nom l'indique en charge de capturer des mesures sur le phénomène observé et de les transmettre à l'unité de traitement. Un CAN (Convertisseur Analogique Numérique) gère la conversion des signaux analogiques provenant de l'unité de capture en signaux numériques interprétables par l'unité de traitement [8].

#### 1.2.1.2 Unité de traitement

L'unité de traitement ou processeur est souvent associée à une petite unité de stockage. Elle récupère les données acquises par l'unité de capture, les analyse, les traite et les transmet à l'unité de communication [8].

#### 1.2.1.3 Unité de communication

Habituellement dotée d'un transcepteur radio, cette unité est en charge d'émettre et de recevoir les données produites par l'unité de traitement, fournissant ainsi aux capteurs la capacité de communiquer avec les autres composants du réseau [8].

#### 1.2.1.4 Batterie

Chaque capteur dispose d'une batterie pour alimenter le reste de ses composants. Néanmoins, en raison de sa taille, elle est assez limitée. L'énergie devient donc une ressource précieuse car la durée de vie de la batterie influe directement sur celle des capteurs [8].

### 1.2.2 Architecture d'un réseau de capteurs sans fil

Un réseau de capteurs sans fil est composé d'un grand nombre de nœuds. Chaque capteur est doté d'un module d'acquisition qui lui permet de mesurer des informations environnementales : température, humidité, pression, accélération, sons, image, vidéo etc. Les données collectées par ces nœuds capteurs sont routées vers une ou plusieurs stations de base ou nœud puits (sink en anglais). Ce dernier est un point de collecte de données capturées. Il peut communiquer les données collectées à l'utilisateur final à travers un réseau de communication, éventuellement l'internet ou un satellite. L'utilisateur peut à



son tour utiliser la station de base comme passerelle, afin de transmettre ses requêtes au réseau (Voir la figure 1.2). En général, un RCSF est composé de quatre éléments montrés par la figure 1.2 : les nœuds de capteurs, une station de base, phénomène à mesurer et l'utilisateur [5, 28].

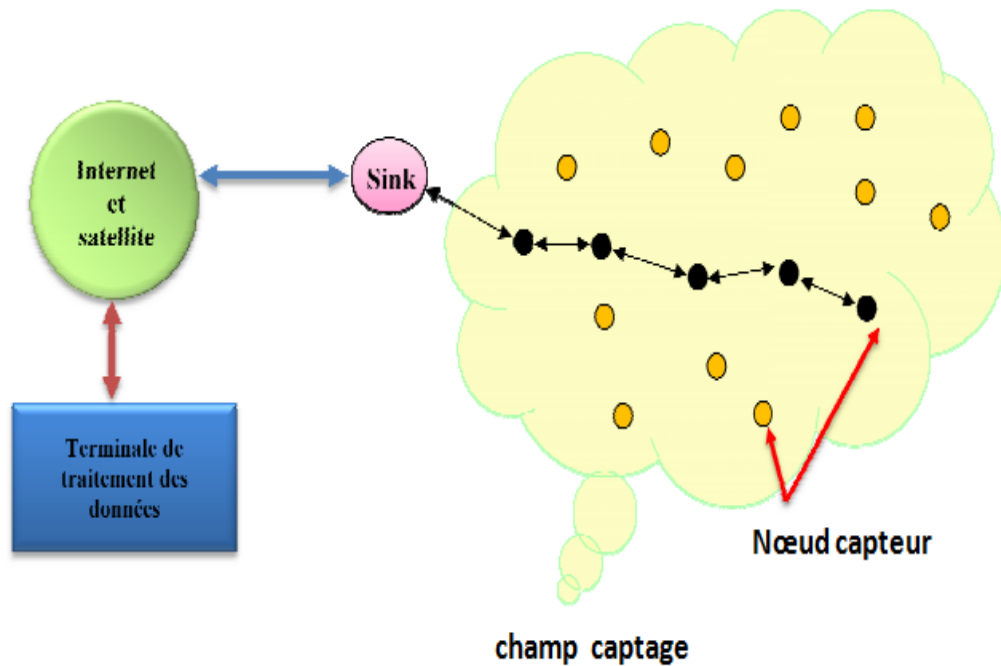


FIGURE 1.2 – Architecture d'un Réseau de Capteur Sans Fil [5]

### 1.2.3 Domaines d'application

Les réseaux de capteurs sont généralement déployés dans une zone donnée pour observer un phénomène spécifique. En fonction du type d'informations récoltées, ils peuvent être destinés à plusieurs domaines d'applications parmi lesquels nous comptons les applications : militaires, environnementales, médicales, domotiques ou encore industrielles [3]. La figure 1.3 illustre quelques applications possibles.



FIGURE 1.3 – Domaines d'application des réseaux de capteurs sans fil [8].

### 1.2.3.1 Applications militaires

Les réseaux de capteurs sans fil sont particulièrement adaptés aux applications militaires étant donné leur faible coût et leur rapidité de déploiement. Dans ce domaine, ils peuvent être utilisés pour : la surveillance de zones d'intérêt, la surveillance et la gestion des troupes et des munitions, l'évaluation des pertes et dommages post-bataille ou encore la détection d'attaques biologique ou chimique [7, 13].

### 1.2.3.2 Applications environnementales

Parmi les applications environnementales des réseaux de capteurs sans fil, nous pouvons citer : la surveillance des flux migratoires d'oiseaux ou de certains animaux [20, 22], la surveillance des domaines forestiers et la détection d'incendies facilitant ainsi la lutte contre les feux de forêt [11]. Le suivi agricole, la détection de pollution dans les sites urbains ou les fuites de produits toxiques dans les sites industriels et bien d'autres applications encore [27].

### 1.2.3.3 Applications médicales

Les réseaux de capteurs sans fil trouvent facilement leur utilité dans le domaine médical. À l'aide de micro-capteurs déployés sur le corps humain ou éventuellement implantés, une surveillance permanente des constantes vitales d'un patient peut être assurée. De plus, certains diagnostics peuvent être facilités en mesurant en temps réel ou en étudiant

certaines données physiologiques (tension artérielle, rythme cardiaque, etc.) collectées et stockées dans la mémoire des capteurs. Enfin, certains comportements inhabituels peuvent être détectés (chute, évanouissement, etc.) [15].

#### 1.2.3.4 Applications domotiques

Avec les progrès technologiques, les capteurs sans fil peuvent être embarqués dans des appareils du quotidien (thermostat, lampe, interrupteurs, etc) et les appareils électroménagers [23]. Ces capteurs peuvent interagir aussi bien les uns avec les autres ou bien avec un réseau externe via internet permettant ainsi à l'utilisateur de contrôler ses appareils à distance [8].

#### 1.2.3.5 Applications industrielles

Il est possible d'utiliser les réseaux de capteurs sans fil dans des contextes industriels. À titre d'exemple nous citons : le suivi de chaîne de production, la gestion d'inventaire, le contrôle de qualité, etc. De façon générale, les réseaux de capteurs sans fil permettent aux entreprises d'améliorer la qualité de service tout en réduisant les coûts [12].

### 1.2.4 Caractéristiques et Contraintes

En raison de leur petite taille et des contraintes de coût imposées, les capteurs sont généralement caractérisés par de fortes limitations en termes de puissance de calcul, de portée de communication et tout particulièrement d'énergie. D'un autre côté, ces mêmes capteurs doivent disposer de facultés d'auto-gestion et de mesures à la hauteur des applications pour lesquelles ils ont été conçus. Ces caractéristiques contradictoires représentent le principal défi auquel doivent faire face les concepteurs.

#### 1.2.4.1 Puissance de calcul

Dans les réseaux de capteurs sans fil, les nœuds disposent de la faculté d'extraire, de traiter et de combiner des données, tout en offrant accès à celles-ci, interagissant ainsi avec leur environnement. Les contraintes de taille, d'énergie et de coût imposées régissent la production de capteurs, limitant ainsi la puissance de calcul de l'unité de traitement [3]. À titre d'exemple, l'Imote2 de MEMSIC dispose d'un processeur 32 bits cadencé en mode économie à 13 Mhz et d'un espace mémoire de 256 Ko. Le TelosB a un processeur de 16 bits pour une fréquence de 8 Mhz et un espace mémoire total de 58 Ko. En dépit de leur

disparité, les capteurs doivent pouvoir collaborer et s'échanger des informations pour des applications communes [8].

#### 1.2.4.2 Communication

Les technologies de communication sans fil sont de nos jours bien répandues. La plus part des réseaux de capteurs sans fil utilisent des communications radio même si des alternatives (laser, infrarouge) existent [25]. L'énergie dépensée pour la communication et sans conteste la plus élevée, la nécessité de protocoles faibles consommation s'accroît. Les réseaux de capteurs sans fil bénéficient donc de mécanismes de communication adaptés, leur assurant un faible coût énergétique. Le plus répandu est le standard IEEE 802.15.4 qui offre un débit maximal de 250 kb/s, une taille de trame réduite, une portée de transmission pouvant atteindre les 100m et un mode veille intégré. De plus, les réseaux de capteurs sans fil disposent de protocoles de routages non-hiérarchiques tel que le protocole réactif AODV ou hiérarchiques, plus adaptés aux grands réseaux. Le choix du protocole de routage diffère donc selon l'application, l'architecture et la densité du réseau. Parmi les atouts des réseaux de capteurs sans fil nous pouvons citer : l'absence d'infrastructures fixes et de câblages, la rapidité et facilité de déploiement, le passage à l'échelle, la gestion de nœuds mobiles, etc. Néanmoins, l'absence de gestion centralisée implique la conception de mécanisme d'auto-gestion permettant aux capteurs de découvrir leur environnement (capteurs voisins, position, structures, etc) [8].

#### 1.2.4.3 Énergie

L'énergie est la plus importante contrainte à laquelle sont soumis les capteurs, la durée de vie du réseau étant directement liée à l'autonomie des capteurs déployés. L'accès physique à ces nœuds est souvent impossible compte tenu des applications visées : à titre d'exemple la surveillance de feux de forêt ou de zones militaires dans lesquelles les capteurs sont disséminés par voie aérienne, ou bien la surveillance d'infrastructures urbaines comme les ponts, les capteurs étant incorporés à la structure même. Beaucoup de recherches s'orientent donc naturellement vers la gestion économe de cette ressource.[8].

### 1.3 Vers les réseaux de capteurs multimédia sans fil

Avec la miniaturisation de certains dispositifs de captures multimédia tels que les caméras et les microphones, les réseaux de capteurs multimédia sans fil ont vu le jour. Dans ces réseaux, les capteurs capturent et traitent des flux multimédia (son, image, vidéo)

élargissant un peu plus leur champ applicatif. En plus de la capacité de capturer des flux multimédia, ces réseaux peuvent traiter et fusionner en temps réel des données multimédia provenant de sources hétérogènes et interagir ainsi avec l'environnement physique. La notion de réseaux de capteurs multimédia sans fil provient de la fusion de deux concepts : les réseaux de capteurs sans fil et les systèmes de surveillance classiques apportant ainsi la flexibilité de l'un et l'efficacité de l'autre [2].

### 1.3.1 Caractéristiques et contraintes

En plus des contraintes communes partagées avec les réseaux de capteurs sans fil classiques telles que les contraintes énergétiques, les réseaux de capteurs multimédia sans fil souffrent de contraintes bien spécifiques liées principalement aux données multimédia qui circulent. Parmi ces contraintes nous citons : la limitation en bande passante, les contraintes temps réel, la capacité changeante du canal de transmission, le codage du flux multimédia, etc [2]. Ci-dessous sont décrits les facteurs principaux engendrant les contraintes spécifiques liées aux réseaux de capteurs multimédia sans fil.

#### 1.3.1.1 Capture et traitement d'image

Dans les systèmes de surveillance classiques, les traitements des images, l'extraction d'informations et la compression sont effectués au niveau local au niveau de la source. Alors que ces actions sont aisées à mettre en œuvre dans ce type de systèmes qui ne connaissent pas de limitations de ressources, la tâche se complexifie dans le contexte des réseaux de capteurs multimédia sans fil, compte tenu de leurs limitations aussi bien matérielles qu'énergétiques [8].

#### 1.3.1.2 Besoin en mémoire

Alors que le codage de données simples (scalaires) ne nécessite que quelques octets : de 1 à 8 en fonction du capteur, le codage d'une image nécessite l'utilisation d'un nombre beaucoup plus important d'octet. La taille mémoire nécessaire dépend à la fois de la résolution et du format de l'image. Une image de 128 \* 128 pixels utilisera 4 fois plus d'espace mémoire qu'une image de 64 \* 64 [17]. Ces critères sont à prendre en considération lors du développement de protocoles de communication pour les réseaux de capteurs sans fil.

### 1.3.1.3 Transmission de données multimédia

La transmission de données est une des actions les plus consommatrice d'énergie dans les réseaux de capteurs sans fil. L'utilisation de protocoles de communication adaptés est donc nécessaire. Dans les réseaux de capteurs classiques où des données simples sont capturées, celles-ci peuvent être fusionnées et envoyées en un seul paquet. Pour les données multimédia, la tâche s'avère plus compliquée si on considère la taille des images. En effet, une seule image doit généralement être fragmentée en plusieurs paquets [8].

## 1.3.2 Domaines d'application

Les réseaux de capteurs multimédia sans fil n'étendent pas seulement le champ applicatif des réseaux de capteurs déjà existants mais améliorent aussi certaines applications déjà existantes [1].

### 1.3.2.1 Surveillance multimédia

La surveillance multimédia peut être utilisée en complément aux systèmes de surveillance classiques. Les capteurs étant faciles à déployer, ils permettent de couvrir plus de surface et d'obtenir un plus grand nombre de perspectives. Les images et les vidéos capturées fournissent des informations complémentaires et appréciables sur l'environnement ou l'événement surveillé. L'information multimédia et particulièrement l'information visuelle est très riche et fait partie des informations les plus exploitées par l'être humain pour percevoir son environnement. Dans certaines applications, l'identification des objets traversant le réseau est nécessaire. C'est le cas pour certaines applications de surveillances environnementales comme l'étude du comportement de certains animaux. Cette identification n'est possible qu'à travers la capture d'images ou de vidéos. D'autres applications n'ont pas directement besoin d'enregistrement multimédia mais cela est fortement appréciable. À titre d'exemple, dans la surveillance des feux de forêt, les capteurs sans fil scalaires sont en mesure de calculer la température afin de détecter les départs de feux, mais la capture d'images aide à avoir une idée plus précise de la situation. De façon plus générale, les données scalaires aident à se faire une idée de la situation mais les données multimédia permettent d'identifier et de classifier plus précisément l'événement étudié [2].

### 1.3.2.2 Gestion et contrôle de trafic

Grâce au déploiement des réseaux de capteurs multimédia sans fil, des applications de gestion de trafic routier deviennent possibles. Les accidents et les ralentissements peuvent

être détectés en traitant les images de la circulation capturées par les capteurs, l'information est ensuite relayée afin d'éviter aux automobilistes les voies encombrées et de ce fait congestionner encore plus le trafic. En outre, des systèmes de stationnement intelligent permettant de guider les voitures directement aux places libres dans les parkings deviennent facilement réalisables [9].

## 1.4 Conclusion

Dans ce chapitre nous avons parcouru les réseaux de capteurs sans fil nous avons donné une vue générale en décrivant leurs architecture, leurs applications, puis nous avons décrit les caractéristiques de ces réseaux , ensuite nous avons abordé les réseaux de capteurs multimédias en présentant leurs caractéristiques, contraintes et leurs domaines d'application, RCSF est un domaine de recherche très répondu, et devient de plus en plus vaste et de nouvelles tendances de recherches sont apparues surtout de ce qui concerne l'optimisation d'énergie et l'application de ces réseaux dans domaines spécialisés.

---

# État de l'art sur les algorithmes de Suivi de cible dans les réseaux de capteurs

---

## 2.1 Introduction

Au cours des dernières années, les réseaux de capteurs sans fil ont trouvé de plus en plus rapidement des applications dans des domaines tels que la surveillance, la surveillance environnementale, la collecte de données automatisée. Une des utilisations importantes des réseaux de capteurs est le suivi d'une cible mobile par le réseau.

Le réseau utilise les techniques de tracking (suivi) d'objets pour signaler en permanence la position de l'objet en termes de coordonnées cartésiennes à un nœud ou à une station de base centrale. Avec les réseaux de capteurs sans fil, les objets peuvent être suivis par un simple marquage avec un petit nœud de capteur [16]. Les nœuds peuvent être utilisés sous forme de balises actives qui annoncent la présence d'un dispositif. Une base de données peut être utilisée pour enregistrer l'emplacement des objets suivis par rapport à l'ensemble des nœuds à des emplacements connus. Avec ce système, il est possible d'interroger ou est l'objet en ce moment, pas simplement là où il a été analysé la dernière fois [24].

Contrairement aux réseaux de détection ou de sécurité, les applications de suivi de nœuds seront continuellement avoir des changements de topologie en tant que les nœuds se déplacent à travers le réseau. Alors que la connectivité entre les nœuds à des endroits fixes restera relativement stable, la connectivité aux nœuds mobiles va changer en permanence. De plus, l'ensemble de nœuds étant suivi va changer continuellement comme des objets qui entrent et sortent du système. Il est essentiel que le réseau soit capable de détecter efficacement la présence de nouveaux nœuds qui entrent dans le réseau [6].



Selon l'environnement, un réseau de capteurs est confronté à différents défis et contraintes. Comme les nœuds de capteurs fonctionnent sur une puissance de batterie limitée, la consommation d'énergie est une préoccupation très importante dans un RCSF et il y a eu des recherches importantes qui se concentre sur la récolte et la minimisation de l'énergie [14].

Ce chapitre est organisé comme suit : dans la section 2.2 nous décrivons ce qu'est le tracking de cible dans les réseaux de capteurs. La section 2.3 quant à elle détaille les fondamentaux du suivi de cible dans les réseaux de capteurs sans fil, classification des solutions existantes dans la littérature. Enfin, dans la section 2.4 nous discutons les métriques d'évaluation des différentes solutions proposées.

## 2.2 Tracking des objets

Le Tracking d'objet appelé aussi le suivi de cible est un domaine majeur de la recherche en RCSFs et a de nombreuses applications de la vie réelle comme la surveillance de la vie sauvage, les applications de sécurité pour les bâtiments et les composés pour empêcher toute intrusion, la surveillance des frontières de internationales pour les traversées illégales et la poursuite des objets volants non identifiés [34].

Le suivi de cible est une application qui consiste à détecter et à localiser une cible mobile tout au long de son évolution dans une zone d'intérêt [8]. En outre, le suivi de l'objet est considéré comme l'une des applications les plus exigeantes en RCSFs en raison de ses exigences d'application qui font peser un lourd fardeau sur les ressources du réseau, en particulier la consommation d'énergie. La tâche principale du Tracking d'objet dans les réseaux dans sans fil est de suivre un objet en mouvement et de faire rapport de sa dernière position dans la zone surveillée par l'application d'une manière opportune acceptable ; et, ce processus dynamique de détection et de déclaration maintient les ressources du réseau sous une forte pression [33].

Le Tracking d'objet dans les réseaux de capteurs est considéré comme l'une des applications les plus consommatrices d'énergie du RCSFs. De ce fait, il existe une nécessité de développer des techniques économes en énergie qui adhèrent aux exigences de l'application d'un système de suivi d'objets qui réduisent la consommation d'énergie totale du Tracking d'objets dans les réseaux de capteur [33].

### 2.2.1 Type de cible

Dans la vie réelle, deux types de cibles existent : les cibles communicantes et les cibles non-communicantes. Les cibles communicantes sont des cibles dites coopératives qui émettent des signaux établissant ainsi un contact avec le réseau. Les cibles non communicantes quant à elles sont non-collaboratives et peuvent dans certains cas être hostiles. Cette différenciation entraîne des conséquences sur la façon de gérer le processus de suivi. En dépit du type de cible à suivre, les applications de suivi partagent des caractéristiques communes : d'une part elles doivent faire face aux contraintes liées au réseau utilisé (limitation énergétique, bande passante, portée de communication, etc), d'autre part, la position de la cible doit être restituée avec précision et en considérant d'éventuelles contraintes temps réel. Le suivi de cible mobile dans les réseaux de capteurs sans fil est une application qui a gagné en popularité pour les raisons suivantes [8] :

- Le prix décroissant des capteurs sans fil permet de les déployer en grand nombre, cette densité croissante permettant de couvrir de plus grandes zones et d'obtenir plusieurs perspectives de celles-ci offrant ainsi une meilleure qualité de suivi.
- L'hétérogénéité des capteurs permet de capturer des données complémentaires sur la cible (taille, forme, type, etc), cette diversité de données aide à fournir des informations plus précises ;
- Les systèmes de surveillance classiques utilisent une approche centralisée, alors que le suivi dans les réseaux de capteurs privilégie une approche distribuée plus évolutive.

## 2.3 Classification des algorithmes de suivi dans la littérature

En RCSFs le suivi d'objets en mouvement a reçu une attention considérable ces dernières années. Plusieurs classifications pour les algorithmes et protocoles de suivi ont été proposées dans la littérature.

Si l'architecture du réseaux est hiérarchique, les solutions de suivi peuvent être classées principalement en quatre régimes :

- Solutions basées sur une architecture hiérarchique.
- Solutions basées sur une architecture en cluster.
- Solutions prédictives.
- Solutions hybrides.

En revanche, un consensus à base de filtre embarqué est la solution de suivi pour les réseaux peer to peer.

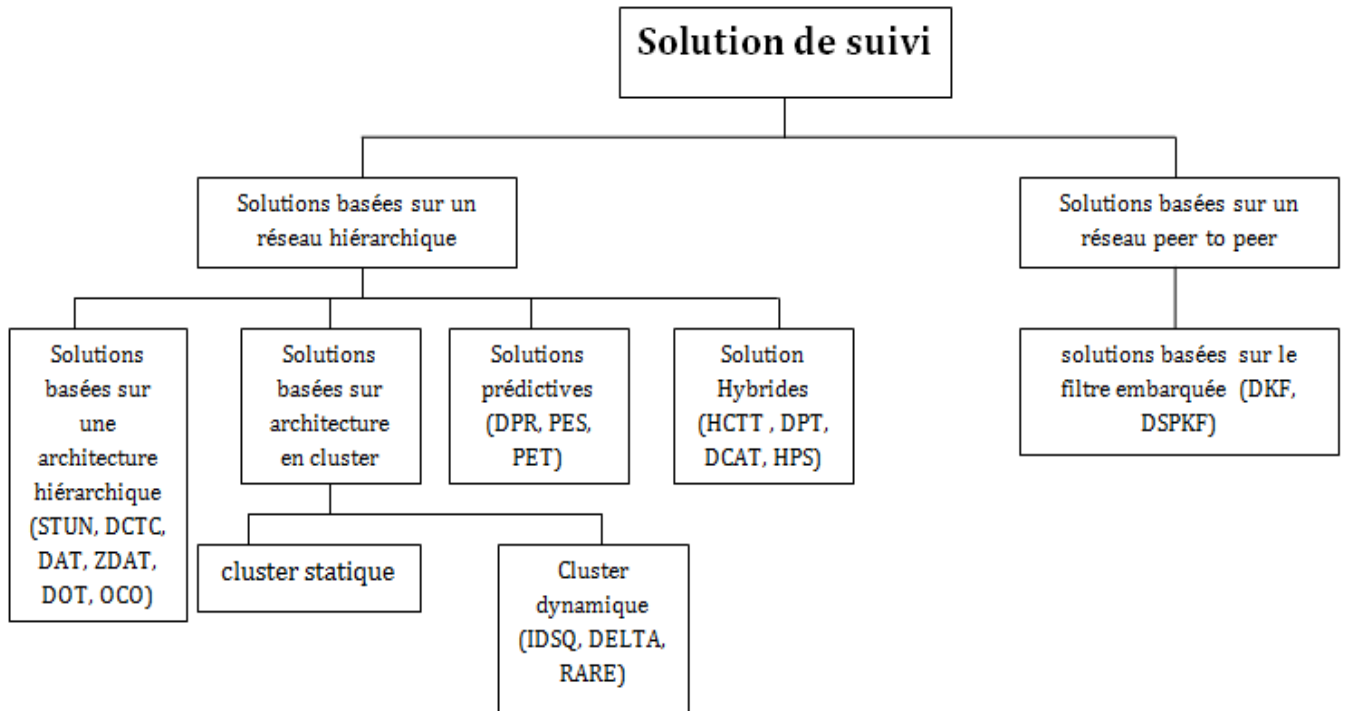


FIGURE 2.1 – Classification des algorithmes de suivi.

### 2.3.1 Solutions basées sur une architecture hiérarchique

Les solutions basées sur une architecture hiérarchique organisent le réseau en structure ou en graphe hiérarchique. Les sommets représentent les nœuds et les arêtes définissent les liens entre les nœuds pouvant directement communiquer. Dans cette architecture [26], Lors du suivi d'une cible les nœuds qui détectent la cible communiquent les uns avec les autres et sélectionne un nœud racine. Le nœud racine collecte les informations de tous les nœuds via un arbre couvrant distribué. Si le nœud racine est loin de la cible, l'arbre sera reconfiguré. Bien que les approches basées en arbre couvrant le suivi des objets mobiles est avec plus de précision, les approches centralisées de suivi de cible sont à la fois consommatrices du temps et de l'énergie ; pour éviter cette limitation des méthodes de suivi à base d'arbres sont proposées. Scalable Tracking Using Networked sensors (STUN) [18], Dynamic Convoy

Tree-Based Collaboration (DCTC) [36], Deviation Avoidance Tree (DAT)[10], Dynamic Object Tracking (DOT) [30], Optimized Communication and Organization (OCO) [29].

### 2.3.1.1 Scalable Tracking Using Networked sensors (STUN)

Le suivi évolutif Utilisant les réseaux de capteur [18] développé par King et Vlah est l'un des approches basé sur la hiérarchie d'arbre crée par la procédure DAB et suit l'objet cible sur la base de la prédiction de sa vitesse et de la distance qu'il traverse dans une durée de temps donnée.

Une fois la configuration du réseau est faite, les feuilles d'un arbre agit comme des nœuds de capteurs qui prend réellement l'activité de détection des nœuds, et les nœuds intermédiaires travaillent comme des nœuds de communication qui passaient généralement l'information à partir de la feuille à la racine (nœud collecteur).

les nœuds intermédiaires/ communications dans la hiérarchie stock les informations d'objets suivis ou être suivis par ses nœuds descendants, Les nœuds intermédiaires tiennent un registre de l'objet détecté et à chaque changement qui se passe dans l'ensemble d'objets suivis de l'un des nœud de la hiérarchie, il met à jour aux nœuds parents et propage les modifications apportées au nœud racine.

#### a. Contenu de l'approche

Soit  $G(V_g, E_g, L_g, w)$  un graphe pondéré, appelé le graphe de capteur, pour représenter les capteurs et certaines caractéristiques attendues des objets en mouvement étant détectés par les capteurs.

Les sommets  $V_g$  représentent les capteurs. Les emplacements des capteurs  $L_g$  sont représentées par une fonction.. Les arêtes  $E_g$  indiquent les liens entre les paires de capteurs ; on dit que deux capteurs sont adjacents s'il est possible pour les objets de se déplacer à partir de la zone de détection de l'un des deux capteurs à celle de l'autre sans passer par celle de tout autre capteur. Ils supposent que  $G$  est reliée (connexe), dans le sens où chaque deux nœuds peuvent être reliés par un chemin. Les poids  $w$  représentent le motif de déplacement des objets dans les différentes régions en unités de taux de détection. A chaque fois qu'un objet arrive ou sort d'une plage de détection du capteur, le capteur génère un événement de détection.

Ils supposent que les objets se déplacent d'un capteur à de telle sorte que pour chaque paire de capteurs adjacents qu'un objet passe à travers, les deux capteurs feront l'expérience aux mêmes taux de détection par rapport à cet objet. Cette hypothèse dit essentiellement que l'objet ne sera pas mettre fin à son mouvement dans le milieu entre deux capteurs. Avec cette hypothèse, on peut attribuer un poids à chaque paire de capteurs adjacents, ce qui représente le taux de détection que chacun de ces deux capteurs feront l'expérience par rapport à des objets se déplaçant entre eux. STUN repose sur une structure arborescente appelée "Drain and Balance Tree". C'est une méthode qui construit la structure hiérarchique de STUN basée sur la fréquence attendue des mouvements d'objets sur une région.

## b. Méthode de construction de l'arbre d'élagage

DAB est une méthode qui construit l'arbre d'une manière ascendante, à savoir à partir des feuilles à la racine, à travers une série d'étapes de DAB. Au sein de chaque étape de DAB, un sous-ensemble des capteurs est fusionné dans le sous arbre équilibré, sans tenir compte les taux d'événements. L'efficacité du procédé est de choisir correctement les nœuds de fusion dans chacun de ces étapes, cette fois-ci en utilisant les informations de taux d'événement. En particulier, les capteurs sont séparées en utilisant un ou plusieurs seuils de taux d'événement, que nous appellerons "drainage de seuils, et les sous-ensembles à haut débit sont fusionnés en premier. Le processus est décrit comme suit :

L'entrée de la méthode est un graphe de capteur  $G = (Vg, Eg, Lg, w)$ , et sa sortie est un arbre hiérarchique  $T = (Vt, Et, Lt)$ . Le procédé est paramétrée par une séquence décroissante de "drainage" de seuils  $H = h_1, \dots, h_k$ , ou  $h_k = 0$ . La  $k$ -étape DAB pour la construction de l'arbre pour un capteur de 1D se déroule comme suit :

- 1 Initialisation de  $T$  d'être un graphe vide.
- 2 Pour chaque seuil de drainage  $h_i$ , dans l'ordre croissant des  $i$ , effectuer une étape de DAB, comprenant les deux phases suivantes :
  - 2.1 Drainage : Ajouter ces nœuds dans  $Vg$  à  $Vt$  dans lequel ont au moins une arête de l'incident dont le poids est supérieur ou égal à  $h_i$ . Cela permet d'insérer un certain nombre d'arbres uniques dans  $T$ . On dit que deux arbres  $T$  sont adjacents si certains de leurs feuilles sont adjacents dans  $G$ .
  - 2.2 Équilibrage : fusionner plusieurs reprises paires d'arbres adjacents pour former des groupes de capteurs, d'une manière non décroissante. Autrement dit, à chaque étape de fusion relier les racines de deux arbres voisins par un nouveau

nœud intermédiaire, de sorte que l'arbre fusionnée aura le plus petit nombre de capteurs parmi tous fusion possible de paires d'arbres adjacents. Le processus de fusion se termine quand il ne reste plus que les arbres non adjacentes à T.

**Exemple**

La figure 2.2 illustre un DAB construction de l'arbre en 3 étapes pour un graphe de capteur 1D. La figure 2.2(a) décrit le graphe de capteur 1D, avec les poids représentés en utilisant des barres verticales, et les seuils de drainage indiqués sur le côté. La figure 2.2(b-d) montre l'arbre à la fin de la première, la deuxième et la troisième étape de la DAB, respectivement. L'arbre final à la figure d est le message d'élagage d'un arbre hiérarchique résultant. Comme nous pouvons le voir, le traitement des capteurs avec des arêtes de coûts élevés plus tôt nous permet de placer leurs nœuds de raccordement à proximité des feuilles de la hiérarchie, comme souhaité. De même, les capteurs moins chers sont traités en dernier, et donc leurs nœuds de connexion finissent éparpillés entre les grandes régions dérivées dans les étapes précédentes.

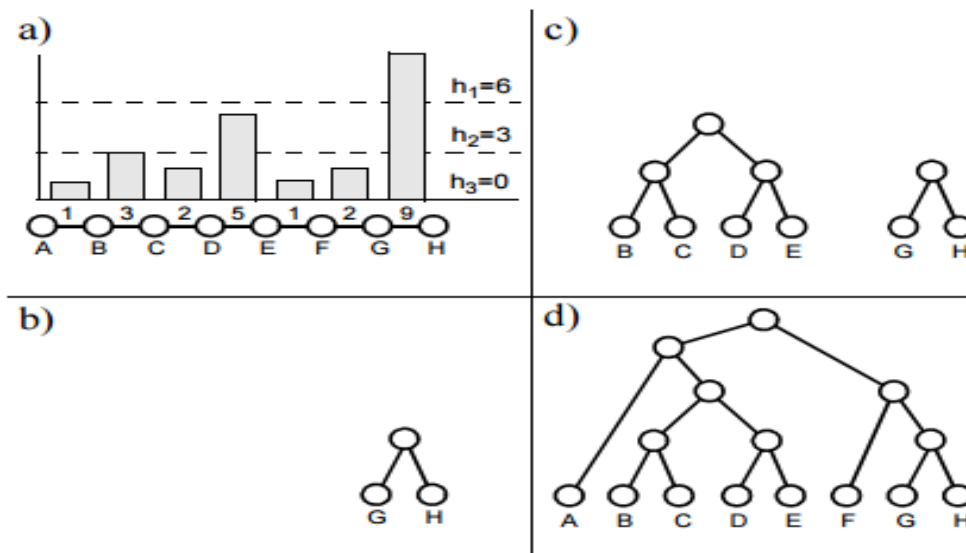


FIGURE 2.2 – Exemple de construction d'arbre DAB pour construire une hiérarchie de message d'élagage [18].

### 2.3.1.2 Dynamic Object Tracking DOT

Un utilisateur mobile, appelé source  $S$  veut suivre un objet mobile, appelé cible  $o$ , les auteurs de cet article supposent que la source connaît les informations de la cible.

Ce protocole [30] est appliqué pour aider une source  $S$  dans la poursuite d'un objet mobile  $o$ , en utilisant les réseaux de capteurs sans fil pour obtenir l'emplacement de l'objet. Ce travail propose un protocole qui permet non seulement la protection de l'énergie des capteurs mais poursuit également la cible rapidement et avec précision.

$S$  utilise une demande d'inondation (une requête) pour obtenir l'emplacement de la cible en envoyant une requête au réseaux de capteur. Les capteurs doivent enregistrer les pistes de la cible, lorsque  $S$  se déplace au nœud proche de la cible, elle collecte les informations de l'emplacement de la cible par celui-ci.

Ce protocole utilise les composantes de face pour construire un voisinage spatiale à fin d'empêcher de perdre la trace de la cible. Il utilise la connaissance du voisinage spatial défini sur un graphe planaire, il est bien connu que le graphe Gabriel est un graphe planaire .

Une arête  $e = (u, v)$  est appelée arête de Gabriel s'il n'y a aucun autre nœud à l'intérieur du disque qui utilise  $e$  comme diamètre.

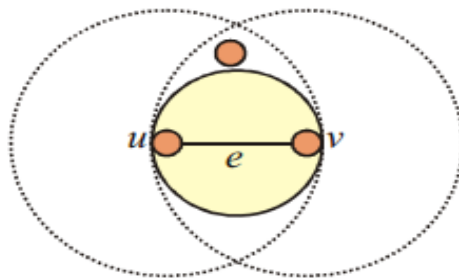


FIGURE 2.3 – Arête de Gabriel [30].

Un graphe est dit graphe Gabriel s'il contient que les arêtes de Gabriel.

Un nœud de capteur a trois états :

- Actif : le capteur peut détecter, transmettre et recevoir des données à tout moment.
- Sommeil : arrêter de détecter, recevoir et transmettre.

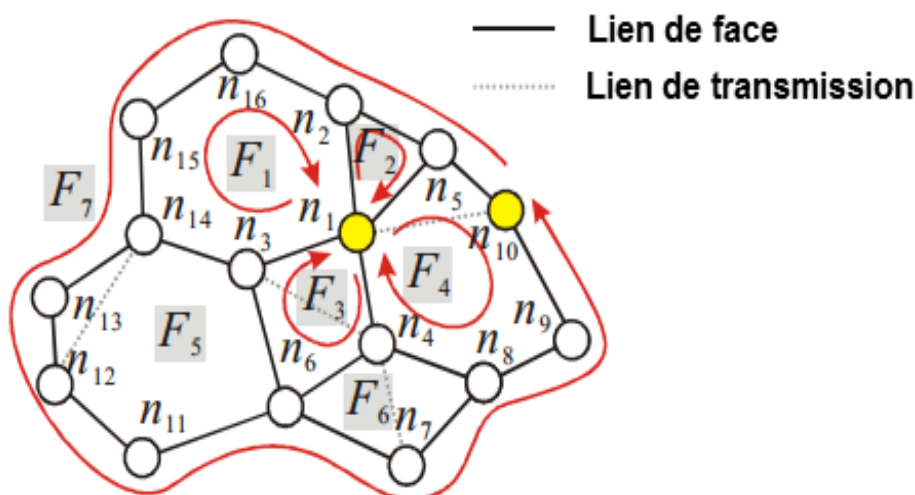


FIGURE 2.4 – Exemple d'un graphe planaire [30].

- Réveil : le capteur se réveille périodiquement à une période prédéfinie.

### 1. Phase de découverte de la cible

Une source  $S$  veut suivre une cible mais ne connaît pas son emplacement, donc elle utilise le réseau pour découvrir la cible. Chaque nœud collecte l'emplacement de tout ses voisins spatiaux en phase de découverte de face. Donc il peut juger s'il est le plus proche de la cible.  $S$  émet un paquet de demande en mode réveil. Le nœud le plus proche de la cible détecte cette dernière et envoie un message de réponse à la source, lorsque le capteur émet la réponse à  $S$ , il met son état à actif en un temps infini et il devient un nœud beacon (nœud d'entrée). Si  $S$  ne reçoit pas de réponse à une heure prédéfinie, il diffuse un paquet de demande à la prochaine période de réveil.

### 2. Phase de détection

Si un capteur  $n$  est un nœud proche de la cible ou qui se situe dans la face  $F_a$ , ce capteur  $n$  devient un nœud d'entrée de  $F_a$  et suit la cible. Les capteurs coopèrent entre eux pour détecter la cible pour ne pas perdre les pistes de la cible. Le capteur  $n$  envoie un paquet de réveil à tout ses voisins des faces pour coopérer à la poursuite de la cible, lorsque les voisins reçoivent le paquet de réveil, ils restent en état actif et détectent la cible en collaboration.



Lorsque la cible se déplace de la zone de détection du nœud d'entrée, un autre nœud  $m$  peut détecter la cible et devient un nœud d'entrée (comme chaque nœud possède les informations de localisation des voisins, il peut vérifier s'il est le nœud le plus proche de la cible). le temps actif du nœud d'entrée est infini c'est à dire il doit rester actif jusqu'à ce que la source  $S$  arrive.

### 3. Phase du suivi de cible

Tandis que la cible  $o$  est suivi, le réseau de capteur doit enregistrer les piste (chemins) de la cible. Lorsque  $S$  reçoit la réponse du nœud d'entrée, elle atteint sa position et l'interroge pour la position suivante de la cible. Comme La cible se déplace, le nœud  $n$  arrête son état et se transforme en mode de sommeil. Dans ce cas le nœud  $m$  suivant devient le nœud d'entrée. Chaque nœud d'entrée enregistre toute les informations collecté et l'envoie à la source. Ce processus est répété jusqu'à ce que la cible est prise par la source.

#### 2.3.1.3 Collaboration basée sur l'arbre de convoi dynamique (DCTC)

DCTC [2] détecte d'abord la cible et la surveille par le suivi de la région environnante de la cible. Elle repose sur une structure arborescente appelé arbre de convoi, qui incluent les nœuds de capteurs autour de l'objet en mouvement, et l'arbre est configuré pour ajouter des nœuds et élaguer certains nœuds lorsque la cible se déplace.

Lors de la première entrée de la cible à la zone de détection, les nœuds de capteurs qui peuvent détecter la cible collaborent entre eux pour sélectionner une racine et pour construire un arbre convoi initial. Le nœud racine récolte de l'information à partir des nœuds de capteurs et affine cette information pour obtenir des informations plus complètes et précises sur la cible en utilisant des algorithmes de classification.

Le grand défi de mettre en œuvre le cadre DCTC est la reconfiguration de l'arbre du convoi d'une manière efficace de l'énergie, lorsque la cible se déplace. Certains des régimes optimisent la consommation d'énergie, mais ne sont pas pratiques [3]. parmi les solutions pratiques à mettre en œuvre le Framework DCTC sont la construction de l'arbre initial, Expansion et élagage d'arbres, reconfiguration de l'arbre.

### 2.3.1.4 Deviation-Avoidance Tree (DAT)

le suivi de l'objet implique généralement deux opérations de base : mise à jour et interrogation. L'emplacement des mises à jour et les requêtes peuvent être effectuées de diverses manières. En général, les mises à jour sont lancées lorsque l'objet se déplace d'un emplacement du capteur à un autre emplacement du capteur. une façon d'interroger est de fournir une requête pour inonder l'ensemble du réseau.

Ce processus est inefficace en raison d'une quantité considérable d'énergie sera consommée lorsque l'échelle du réseau est grande ou lorsque le taux de requête est élevé. Une autre méthode est de localiser toutes les informations dans un capteur spécifique (Sink) .Ensuite aucune inondation est obligatoire. Un inconvénient est que lorsque les objets se déplacent fréquemment, les messages de mise à jour abondantes seront générées. Pour remédier à ça "DAB" est proposé. Dans cette requête les messages ne sont pas tenus d'être inondées et les messages de mise à jour ne sont pas toujours transmis à l'évier.

Les méthodes proposées pour surmonter ces problèmes sont DAT et Z-DAT.

- L'arbre d'évitement de déviation DAT [10] est construit pour pallier l'inconvénient de la structure de DAB. Dans cette méthode de DAT traite chaque nœud comme un sous-arbre singleton. Plus de liens peuvent être utilisés pour connecter tous les sous arbres ensemble. A la fin tous les sous-arbres seront reliés en un seul arbre. DAT a deux approches de la scène. La phase primaire a pour but de réduire la valeur mise à jour, tandis que la seconde étape vise à réduire le coût de la requête.

- L'arbre d'évitement de déviation à base de zone Z-DAT est également similaire à la DAT, mais Z-DAT examine les liens dans un ordre différent. Le DAT et Z-DAT réduisent presque le coût de mise à jour, mais échoue parfois à réduire le coût de la requête. Pour surmonter ces problèmes de réduction des coûts de requête QCR (Query Cost Reduction) est conçu. Méthode qui réduit les coûts de mise à jour et de requête totale en ajustant l'arbre de suivi de l'objet obtenu par DAT / Z-DAT.

### Limites des solutions basées sur une architecture hiérarchique

les inconvénients principales des solutions basées sur une architecture hiérarchique sont les suivantes :

- Une panne au niveau d'un nœud peut engendrer de fortes conséquences.
- le traitement des données soit centralisé au niveau du nœud racine, cela engendre un coût d'acheminement des informations non-négligeable.
- l'aspect centralisé des traitement fige l'évolution du réseau.

### 2.3.2 Solutions basées sur une architecture en cluster

La méthode à base de Cluster divise le réseau en groupes (clusters) pour appuyer le traitement de données en collaboration. Un cluster est constitué de cluster Head et de capteurs membres. Lorsqu'un capteur détecte un objet, il se porte volontaire pour agir en tant que CH. Il n'y a pas besoin d'élection explicite de leader. Donc, les échanges de messages ne sont pas engagés. Si plus d'un capteur puissant peut détecter le signal, les nœuds bénévoles multiples peuvent exister. Donc, une approche décentralisée doit être appliquée pour garantir que un seul Cluster chef est actif dans le voisinage d'une cible pour être suivis avec une forte probabilité. la méthode basée sur Cluster est divisé en deux types. Clustering Statique, Clustering Dynamic. Le clustering dynamique se compose de quelques méthodes comme IDSQ [11], DELTA [12], et RARE [13].

#### 2.3.2.1 Clustering statiques

Les clusters sont formées statiquement au moment du déploiement du réseau. Les attributs de chaque groupe, tels que la taille d'un cluster, la zone qu'il couvre, et les éléments qu'il possède, sont statiques. En dépit de sa simplicité, l'architecture de cluster statique souffre de plusieurs inconvénients. Tout d'abord, l'adhésion fixe n'est pas robuste du point de vue de la tolérance aux pannes. Si un chef de cluster meurt de pouvoir d'épuisement, tous les capteurs du cluster rendent inutile. Deuxièmement, l'adhésion fixe empêche les nœuds de capteurs dans différents groupes de partager des informations et collaborer sur le traitement des données. Enfin, l'adhésion fixe ne peut pas adapter à des scénarios très dynamiques dans lesquels les capteurs dans la région de la concentration d'événement de haut peuvent être instrumentés pour rester éveillé.

#### 2.3.2.2 Clustering dynamiques

l'architecture dynamique du cluster a plusieurs caractéristiques souhaitables. Lorsqu'un capteur avec une batterie et une puissance de calcul suffisante détecte les signaux d'intérêt, il se porte volontaire pour agir comme un cluster Head. En comparaison avec les approches de clustering statiques, un cluster peut soutenir différents groupes à différents moments.

### 2.3.2.2.1 Distributed event localization and tracking with wireless sensors (DELTA)

DELTA [31] c'est un Framework de suivi d'objet distribué permettant d'éviter la charge à la station de base. Il utilise les mesures de lumière du capteur dans les décisions et les calculs comme l'élection du chef et la localisation. DELTA maintient les clusters qui sont dynamiquement formés. Un algorithme d'élection fondé sur la mesure détermine le chef du cluster qui est responsable de :

- La maintenance du cluster.
- La collecte de données et le traitement.
- Envoyer des rapports à la station de base.

La station de base est connecté à internet et peut déclencher l'alarme en cas de comportement inhabituel est détecté. Une exigence de DELTA est la connaissance des informations de localisations par les nœuds de capteur. ceci peut être réalisé par des GPS ou toute autre service de localisation.

## 1. Architecture de DELTA

Un problème clé dans le suivi des événements le surcout(frais de générations) d'identifications et d'organisations des nœuds de capteur appropriés d'une manière réparties toute en offrant un degré satisfaisant de précision.

Comme DELTA est une approche distribuée, les nœuds de capteur collaborent pour calculer l'emplacement de l'objet en mouvement avant d'envoyer les données à la station de base, pour y prévenir différents rôles sont attribués aux nœuds. Les états et les changements d'états des nœuds ainsi que leurs rôles sont représentés dans la figure 2.5.

- les lignes en pointillées montrent les changements d'états causés par les observations de l'événement.
- les lignes pleines montrent les actions provoquées par des messages provenant d'autres nœuds.

Dans un premier temps, tout les nœuds sont à l'état inactif, dès que la cible à été observé par un nœud, ce dernier passe à l'état d'élection en cour d'exécution, et planifie un minuterie selon l'intensité de sa mesure de la lumière. Lorsque le délai expire, un message de pulsation est diffusé périodiquement pour informer ses membres voisins sur la présence

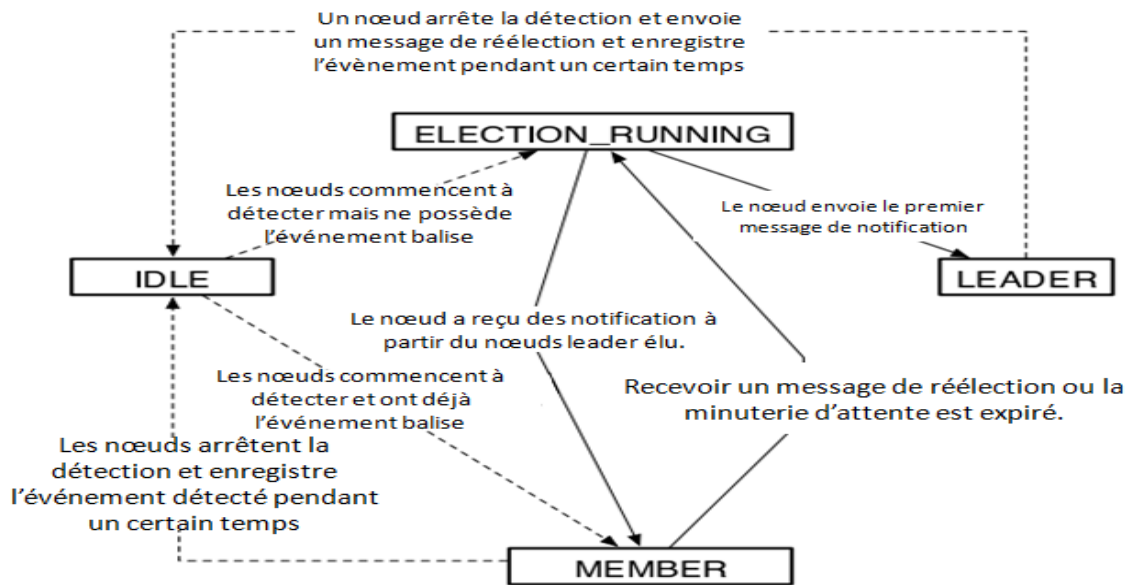


FIGURE 2.5 – Domaines d'application des réseaux de capteurs sans fil.

du leader (le calcul de la minuterie est important car il détermine le nœud leader). Une fois l'observation de l'objet en mouvement échoue, le leader passe à l'état repos. Les membres du cluster attendent les messages de pulsations périodiquement pour être informer de la présence du leader. Ce processus d'élection du chef vise à déterminer rapidement un nœud leader unique capable de couvrir une cible d'une manière fiable.

Le principal inconvénient de cette méthode est qu'elle ne peut faire face à vitesse constante, alors que, la vitesse variable n'est pas considérée.

### Limites des solutions basées sur une architecture en cluster

Ce type d'architecture offre un certain nombre d'avantages parmi lesquels : la facilité de passage à l'échelle, la diminution du trafic et l'efficacité énergétique. Cependant, une panne au niveau des nœuds, particulièrement au niveau des nœuds-parents peut avoir de fortes conséquences sur le processus de suivi et le relai d'informations.

### 2.3.3 Solutions prédictives

Dans cette catégorie, la trajectoire de la cible est prédite pour n'activer que les capteurs se trouvant sur celle-ci et ainsi économiser de l'énergie. Cette prédiction peut être réalisée

à l'aide de modèles prédictifs parmi lesquels : les filtres de Kalman, ou encore en utilisant des mécanismes probabilistes comme les chaînes de Markov.

**Définition 2.3.1.** *Le Filtre de Kalman est un filtre à réponse impulsionnelle infinie qui estime les états d'un système dynamique à partir d'une série de mesures incomplètes ou bruitées.*

**Définition 2.3.2.** *un processus de Markov est un processus stochastique possédant la propriété de Markov, d'une manière simplifiée : prédiction du futur sachant le présent.*

### 2.3.3.1 Dual Prediction-based Reporting for Object Tracking Sensor Networks (DPR)

#### 1. Contexte

Dans cette approche [35] Les nœuds de capteurs sont déployés dans une zone de surveillance avec des limites géographiques prédéfinies. La station de base (BS) agit comme l'interface entre le réseau de capteur et les applications émettant des commandes et collecte des données de l'intérêt.

Un nœud de capteur a la responsabilité de suivre l'intrus de sa zone de détection et les rapports des états des objets mobiles, ces nœuds de capteurs sont supposés statiques et la SB a une bonne connaissance de l'emplacement de chaque nœud. Les auteurs de cet article déclarent que Les objets mobiles sont identifiables pour que les nœuds de capteurs seront capables de stocker l'historique du mouvement des objets mobiles qui est utilisé pour la prédiction du futur mouvement d'objets.

#### 2. Mécanisme de suivi

Pour l'optimisation de l'énergie, les auteurs de cet article ont proposé un système de surveillance appelé système de contrôle continu. Ce système active le nœud de capteur seulement quand un objet entre dans sa zone de détection.

Le mécanisme de PDR comporte deux composantes principales à savoir le modèle de localisation et le modèle de prédiction.

- Modèles de localisation : régissent sur la granularité de l'information de localisation que le système désire.

- Modèles de prédiction : analysent l'historique de déplacement des objets suivi et estime leur futur états de déplacement.

Le modèle de prédiction est déployé au niveau d'un nœud de capteur ainsi que dans la station de base, en utilisant les mêmes données historiques. Pour faire la prédiction, l'historique du mouvement de tout les objets est nécessaire. La station de base peut obtenir l'historique soit par ses propres prédictions correctes ou mettre à jour les paquets à partir des nœuds de capteurs. Par contre les nœuds de capteurs n'ont pas une vue globale sur le réseau par conséquent, les paquets historiques doivent être transmis entre les nœuds de capteurs qui sont impliqués dans le suivis d'objets.

### **3.L'idée de DPR est :**

Un nœud de capteur qui a un objet mobile sur son territoire prévoit d'abord le mouvement de l'objet pour la prochaine période, pendant ce temps, la SB effectue la même prédiction basée sur l'historique des mouvements de ces mêmes objets. Si le déplacement de l'objet observé correspond à la même prédiction du nœud de capteur aucune transmission n'est nécessaire, car la SB a les mêmes prédictions que le nœud de capteur, dans le cas contraire, le nœud de capteur corrige la SB en lui envoyant des objets réels.

#### **a. Modèle de prédiction**

Le modèle de prédiction se réfère aux fonctions qui intègre des heuristiques et des stratégies pour prédire le mouvement de l'objet. 3 modèles de prédictions sont proposés :

- Instantané : suppose que l'objet continue à se déplacer dans la même direction et vitesse que le dernier état observé.
- Moyenne : prédit le futur déplacement d'un objet en faisant la moyenne des données historiques.
- Moyenne exponentielle : donne plus de poids à l'historique récent qui a plus de chance à refléter.

#### **b. Modèle de localisation**

La localisation permet de décider la précision des données historiques, donc indirectement affecter la précision des modèles de prédiction. Pendant ce temps, l'application peut choisir

le modèle d'emplacement approprié pour l'emplacement des objets mobiles déclarés par les nœuds de capteurs.

Deux facteurs contrôlent la précision des données historiques. L'un est l'unité de temps, le deuxième est le modèle de l'emplacement. En général, les modèles de localisation peuvent être classés en deux modèles différents : géométrique et symbolique. Un modèle géométrique fournit les coordonnées à  $n$  dimensions, ce qui est à grains fins, mais engage des coûts et la complexité considérable. Un modèle symbolique désigne une zone comme l'emplacement de l'objet.

### limites des solutions prédictives

Les solutions prédictives ont l'avantage certain d'exploiter au mieux les informations disponibles aussi bien sur la cible que sur l'environnement. Elles restent cependant inadaptées à certains contextes critiques (applications militaires, détection d'intrusion, etc) ou la perte, même momentanée de la cible est in-envisageable.

### 2.3.4 Solutions hybrides

Comme son nom l'indique, cette catégorie inclue les solutions où plusieurs approches sont combinées. À titre d'exemple, des solutions prédictives basées sur des architectures hiérarchiques sont proposées dans [4] alors que dans [16] un mécanisme prédictif implémenté sur un réseau organisé en clusters est présenté. Ce type de solutions permet de combiner et d'exploiter au mieux les avantages des différents schémas.

#### 2.3.4.1 A Hybrid Cluster-Based Target Tracking Protocol (HCTT)

##### 1. Hypothèses

Dans cet approche [32], le RCSF à grande échelle contient  $N$  nœuds de capteurs statique déployés dans une zone à deux dimensions , pour détecter une cible mobile unique.

Le nœud puits est déployé au centre du réseau et les nœuds de capteurs sont déployé de façon aléatoire, chaque nœud peut fonctionner dans trois états, il peut transmettre des paquets, recevoir des paquets, détecter la cible à l'état actif. Dans l'état de détection, il ne peut effectuer qu'une opération de détection. Dans l'état de sommeil, il dort pour la plupart du temps et se réveille périodiquement pour détecter la cible et d'écouter les messages.



Le réseau est organisé en cluster en utilisant toute algorithmes de classification approprié. Dans cette article ils ont utilisé l'algorithme LEACH. Chaque nœud connaît son emplacement et les informations locales des ses voisins mais n'a pas d'information globale sur la topologie du réseaux.

## 2. problème de frontières

Lors de suivi d'une cible dans une zone de surveillance, de multiples nœuds entourant la cible collaborent pour rendre les informations collectées plus complète, fiable et précise. Il n'y a pas de problème lorsque la cible est à l'intérieur d'un cluster, comme tous les capteurs activés appartiennent au même groupe, et ils peuvent communiquer efficacement. Cependant, lorsque la cible se déplace à travers les limites des clusters multiples, le problème limite se produit.

## 3. Solution du problème de frontières

Lorsque la cible se déplace à l'intérieur d'un cluster, ce dernier est responsable de la collaboration de nœud local et le suivi de la cible ; lorsque la cible se rapproche des limites de clusters, un processus de clustering dynamique sera déclenchée pour former un cluster dynamique à fin de résoudre le problème de frontière. Le cluster dynamique à la demande disparaîtra peu après la cible se déplace loin des limites. Comme la cible se déplace dans le réseau, les clusters statiques et les clusters dynamiques à la demande gèrent alternativement la tâche de suivi efficace.

## 4. Protocole de suivi de cible à base de cluster hybride (HCTT)

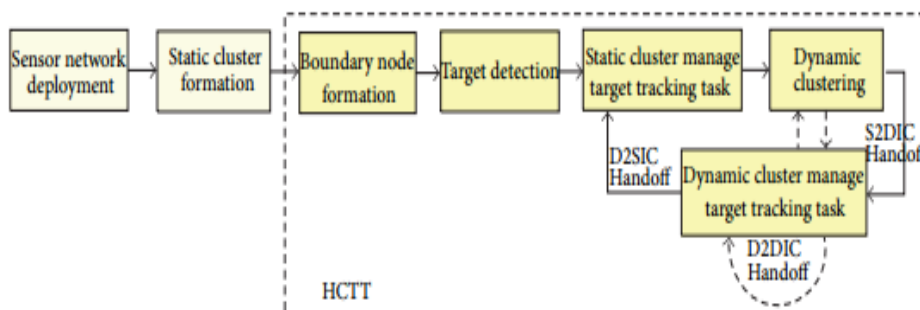


FIGURE 2.6 – Protocole de suivi de cible à base de cluster hybride [32].

La figure 2.6 illustre la description du système et illustre l'organigramme de HCTT. Les nœuds de capteurs sont organisés en cluster statiques selon l'algorithme de classification approprié. Les nœuds de frontières de chaque groupe sont également formés. Lorsqu'une cible est dans le réseau, un cluster statique de détection de la cible se réveille pour suivre la cible. Lorsque la cible se rapproche de la limite, les nœuds frontières peuvent détecter la cible, et un cluster dynamique à la demande sera construit à l'avance pour le suivi de la cible avant que la cible atteigne la limite. Comme la cible se déplace à travers les frontières, les clusters statiques et les clusters dynamiques sur demande gèrent alternativement la tâche de suivi. Différents types de transfert intercluster (à savoir le transfert S2DIC d'un cluster statique à un cluster dynamique, le transfert de D2SIC d'un cluster dynamique à un cluster statique, et le transfert D2DIC d'un ancien cluster dynamique à un nouveau cluster dynamique). En outre, le cluster dynamique rejettera après que la cible se déplace loin de la frontière et entre dans un autre cluster. Avec l'aide de nœuds limites, HCTT intègre le clustering dynamique à la demande dans un WSN à base de cluster évolutive pour le suivi de la cible, ce qui facilite la collaboration des capteurs entre les clusters et résout le problème de frontière.

### Limites du protocole HCTT

L'exploitation interactive des clusters dynamiques et statiques rend efficacement HCTT immunitaire pour la perte de la cible (ie, il existe toujours un cluster qui garde la trace de la cible), cependant, il se fait au détriment des échanges de messages lourds et un niveau élevé de la consommation d'énergie. Par exemple, un seul transfert statique au dynamique inter-cluster, seul, déclenche l'échange de six types de messages de contrôle, à savoir, la demande, la réponse, le travail, le sommeil, la démission, et le rejet. La construction de cluster dynamique dans la région d'alerte nécessite également une installation coûteuse.

## 2.4 Métriques pour l'analyse des protocoles de poursuite de cible

Dans cette section, nous résumons notre État de l'art et signalons dans quelle mesure ces systèmes sont bien adaptés pour un scénario de suivi d'objets.

Cette comparaison des systèmes de suivi est basée sur la structure du réseau en terme des métriques suivantes :

- stratégie pour désactiver les nœuds non nécessaires.
- Mécanisme de récupération de cible.
- Surcout du protocole comprenant : cout minimum du pré traitement et le nombre de paquets de contrôle.
- Service de localisation.
- Suivi d'objet multiple.

On peut constater dans le tableau 2.1 que, la plus part des mécanismes de suivi utilise une certaine stratégie de prédiction à fin de désactiver les nœuds non nécessaire pour réduire la consommation d'énergie.

Le mécanisme de récupération, utilisé en réaction avec la perte de la cible, peut être considéré comme l'un des facteurs distinctifs lorsque il est question de comparer les différentes méthodes de suivi. Les régimes de suivi à base de la prédiction ne bénéficient pas d'une immunité total de la perte de la cible pour cela un mécanisme de récupération est sensé prendre en charge à chaque fois qu'une cible se perd.

L'augmentation de la réactivation des nœuds réduit la probabilité de perte de cible (DCTC, DAB, DAT).

Dans DOT, le problème de perte de cible n'est pas abordé.

Dans HCTT, presque aucune perte de cible car la cible est contrôlée par un cluster statique et un cluster dynamique ; par conséquent, aucun mécanisme de recouvrement est prévu par le système.

Le maintien de la structure au cour de l'opération est un aspect important pour les méthodes basées sur une structure hiérarchique ou en cluster, et donne également un compromis entre la précision et robustesse d'une part et le surcout de contrôle d'autre part. En générale, on peut faire valoir que par rapport aux méthodes à base de cluster, les algorithmes basées sur une architectures hiérarchique entraînent plus de surcout en terme de construction et communications. De plus les systèmes structurés sont soumis à une augmentation de la consommation de l'énergie.

Type de solution	Stratégie pour désactiver les nœuds non nécessaires	Mécanisme de récupération de cible	Coût minimum du pré traitement	Nombre de paquets de contrôle	Algorithme de localisation
Architecture hiérarchique [18]	Non	Non	Non	1 + construction d'arbre	Le plus proche emplacement du capteur à la cible par la racine
Architecture hiérarchique [30]	Oui	Non	Oui	16 + construction du graphe planaire	trilatération
Architecture hiérarchique [36]	Non	Non	Oui	1 + construction d'arbre	Le plus proche emplacement du capteur à la cible par la racine
Architecture hiérarchique [10]	Oui	Non	Non	6	Le plus proche emplacement du capteur à la cible par la racine
Architecture en cluster [31]	Non	Non	Non 33	5	calcul de la localisation de la cible

Type de solution	Stratégie pour désactiver les nœuds non nécessaires	Mécanisme de récupération de cible	Cout minimum du pré traitement	Nombre de paquets de contrôle	Algorithme de localisation
Solutions prédictives [35]	Oui	Oui	Oui	6 + Construction du cluster	Le plus proche emplacement du capteur à la cible par la racine
Solutions hybrides [32]	Oui	Non	Oui	16 + Construction du cluster	calcul de la localisation de la cible

TABLE 2.1 – Comparaison des solutions de suivi.

## 2.5 Conclusion

Dans ce chapitre, nous avons détaillé les méthodes de suivi d'objets dans les réseaux de capteurs, souligné leurs avantages et inconvénients, nous avons ensuite proposé une classification des algorithmes de suivi existants, deux paramètres d'analyses clés sont considérés dans la réalisation des solutions pour le suivi de la cible dans les réseaux de capteurs sans fil : la précision du suivi et l'économie de l'énergie. Dans le chapitre suivant, nous attelons à décrire notre proposition.

---

# Proposition

---

## 3.1 Introduction

Après avoir effectué une étude comparative et une classification des solutions existantes dans la littérature, nous passons à l'étape de la description de notre contribution et cela en élaborant un algorithme distribué implémenté sur un réseau structuré en graphe planaire. Les nœuds de capteurs coopèrent ensemble afin de tracer et estimer la position et la trajectoire de la cible.

Dans ce qui suit, nous détaillerons les différents aspects constituant notre approche. Nous commencerons par introduire les motivations, nous y exposerons les contraintes liées au suivi d'objets. Par la suite, nous dériverons en détail le modèle d'un système déjà proposé sur lequel nous contribuerons à l'améliorer. Nous terminerons par une conclusion.

## 3.2 Motivations

Comme nous l'avons mentionné précédemment, le tracking (ou suivi de cible) est une application particulière des réseaux de capteurs, l'objectif dans le cadre du tracking est de s'appuyer sur les réseaux afin de détecter, identifier une cible (éventuellement plusieurs) traversant une zone d'intérêt et obtenir les données relatives, au minimum, à la trajectoire de cette dernière, en fonction des capacités de calcul des capteurs, d'obtenir des données supplémentaires telle que la vitesse ou l'estimation de la trajectoire. Étant donné que les nœuds de capteurs sont équipés d'une batterie limitée, la capacité énergétique des capteurs doit être utilisée efficacement afin de maximiser la durée de vie du réseau, tout en offrant une précision raffinée. La question qui se pose donc est : comment gérer le compromis entre la précision du suivi et l'économie des ressources ?

De nombreuses solutions existent pour réaliser le problème de suivi d'objets dans les réseaux de capteurs, dont certaines apportent une meilleure précision au dépit d'un coût de communication élevé. Dans ce travail nous proposons un algorithme distribué de suivi de cible mobile, en diminuant aussi bien le nombre de paquets transmis que le nombre de nœuds de capteurs à activer le long de la trajectoire de la cible.

### 3.3 Description du système

#### 3.3.1 Définitions et hypothèses

Ce protocole [30] est appliqué pour aider une source  $S$  dans la poursuite d'un objet mobile  $o$  en utilisant les réseaux de capteurs sans fil.

La source  $S$  utilise une demande par inondation (une requête) pour obtenir l'emplacement de la cible, en envoyant une requête au réseau de capteur, les capteurs doivent enregistrer la piste de la cible. Lorsque  $S$  se déplace au nœud proche de la cible, elle collecte les informations de l'emplacement de cette dernière. En outre, un capteur peut détecter l'emplacement précis de l'objet, car le capteur utilise la trilération pour calculer l'emplacement de l'objet. Chaque nœud connaît son emplacement tel que ces informations peuvent être acquises à partir du système de positionnement global (GPS) ou d'autres mécanismes.

La notation  $L(o)$  désigne l'emplacement de l'objet  $o$ . Ce travail suppose que la plage de transmission est au moins deux fois de la plage de détection.

Les capteurs peuvent suivre un objet en coopération. Quand un objet quitte la plage de détection du capteur  $X$  et se déplace dans la plage de détection du capteur  $Y$ , ce rapport de portée peut garantir que  $X$  et  $Y$  peuvent toujours communiquer entre eux. Le capteur  $Y$  pourra alors informer directement  $X$  sur la présence de l'objet. Cette coopération entre nœuds voisins permettra au réseau de ne pas perdre la trace de l'objet durant son déplacement.

Dans ce travail, on utilise la connaissance des voisins spatiaux définie par le graphe planaire. Pour qu'un nœud trouve localement qui sont ses voisins spatiaux, nous avons d'abord besoin d'une méthode pour aplanir le réseau.

Le réseau de capteurs est modélisé par un type particulier de graphe planaire, appelé Graphe de Gabriel.

**Définition 3.3.1.** [30] *Un graphe est dit graphe Gabriel s'il ne contient que des arêtes de Gabriel.*

**Définition 3.3.2.** [30] *Une arête  $e = (u, v)$  est appelée arête de Gabriel s'il n'y a aucun autre nœud à l'intérieur du disque ayant comme diamètre le segment de droite  $(u, v)$ .*

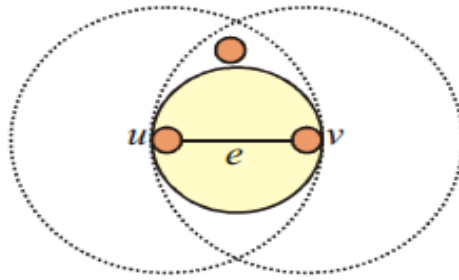


FIGURE 3.1 – Arête de Gabriel [30].

Au départ, chaque nœud échange les informations de localisation et calcule les voisins de faces.

**Définition 3.3.3.** [30] *On dit que  $u$  est un voisin de face de  $v$  si  $e$  est une arête de Gabriel. Lorsque un nœud possède  $n$  voisins de faces, cela veut dire, qu'il possède  $n$  faces adjacentes.  $F_i$  est notée comme identité de la face  $i$  dans le réseau.*

Dans la figure 1.2, le nœud  $n_1$  a 4 voisins de faces  $n_2, n_3, n_4$  et  $n_5$ .

**Définition 3.3.4.** [30] *Les voisins spatiaux d'un nœud  $n_i$  est l'ensemble des nœuds qui forment les faces adjacentes du nœud  $n_i$ .*

Dans la figure 1.2, les voisins spatiaux du nœud  $n_1$  sont :  $\{n_5, n_1, n_9, n_8, n_4\}$  dans la face  $F_4$ ,  $\{n_2, n_5\}$  dans la face  $F_2$ ,  $\{n_2, n_{16}, n_{15}, n_{14}, n_3\}$  dans la face  $F_1$  et  $\{n_3, n_6, n_4\}$  dans la face  $F_3$



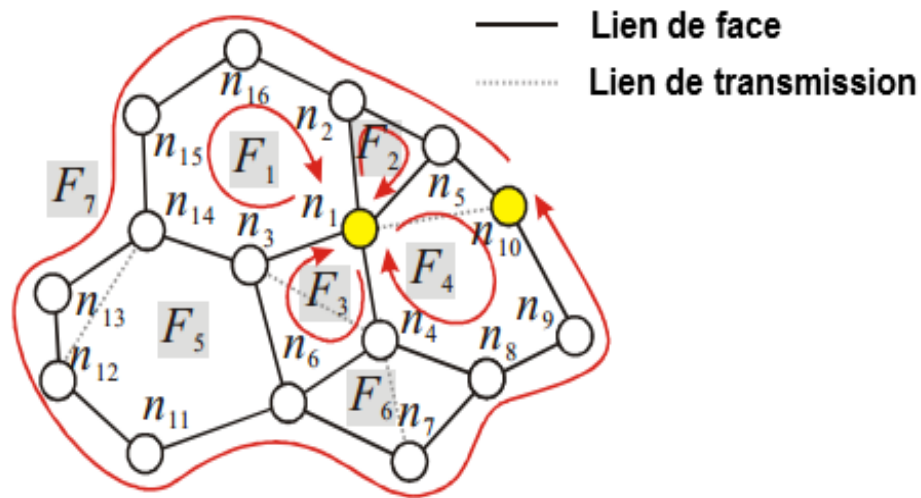


FIGURE 3.2 – Exemple d'un graphe planaire.

### 3.3.2 Aperçu du protocole

Ce paragraphe illustre un aperçu du protocole de suivi d'objets voir figure 1.3. Ce protocole [30] est appliqué pour aider une source  $S$  dans la poursuite d'un objet mobile  $o$ . Un scénario possible est que nous envoyons un petit robot (comme une abeille) pour chasser un ennemi ou un animal sauvage. Le robot doit suivre la cible. Supposons que le robot ne peut pas détecter l'emplacement de l'objet, de sorte que le robot doit utiliser les réseaux de capteurs pour obtenir l'emplacement de la cible. Lorsque  $S$  obtient l'emplacement de la cible, il chasse la cible.

Tout d'abord,  $S$  envoie une demande aux réseaux de capteurs pour obtenir l'emplacement de la cible.  $S$  obtient l'emplacement actuel de la cible grâce à un nœud d'entrée (le nœud le plus proche de la cible). Comme la cible se déplace arbitrairement, les capteurs doivent enregistrer le déplacement de la cible tel que la séquence des nœuds d'entrées  $\langle n_1, n_2, n_3, n_4, n_5, n_6 \rangle$  représente la piste (ou trajectoire) empruntée. Tant que la cible reste toujours dans la zone de détection du nœud  $n_1$ , la source  $S$  chasse la cible. En revanche, si la cible se déplace,  $S$  interroge le nœud d'entrée  $n_1$  sur la position suivante. Ce processus est répété jusqu'à ce que la cible est attrapée par la source.

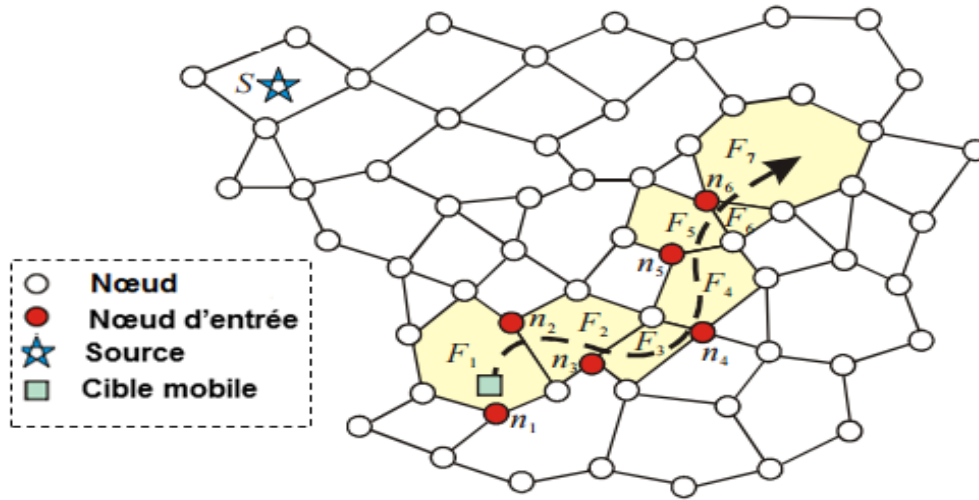


FIGURE 3.3 – *Un aperçu pour le suivi d'objet [30].*

Nous pouvons constater que le processus de suivi se déroule en quatre phases principales à savoir : la phase d'initialisation (découverte de face), phase de découverte de cible, phase de détection et enfin la phase de suivi.

### 3.3.3 Processus de suivi

#### 3.3.3.1 Phase de découverte de face

Tout d'abord, chaque nœud collecte les informations du voisin par l'envoi du message 'Hello'. Ensuite, chaque nœud calcule ses voisins de face sur le graphe de Gabriel (GG) associé. Le nœud voisin de face signifie que le lien entre ce nœud et son voisin est une arête de Gabriel. Ensuite, chaque nœud informe ses voisins sur les nœuds de face calculés. Après la découverte de face, chaque nœud obtient les informations de localisation des voisins spatiaux.

#### 3.3.3.2 Phase de découverte de cible

Si un nœud  $n_i$  détecte la cible et il est le plus proche, il transmet un paquet de réponse à la source (après la phase de découverte de face, chaque nœud peut juger s'il est le plus proche de la cible). Lorsque  $n_i$  émet le paquet de la réponse, il devient nœud d'entrée et commence à suivre la cible. Si la source ne reçoit pas un paquet de réponse à un moment prédéfini, elle rediffuse le paquet à la prochaine période de réveil.

Le format de paquet de demande est : **request** (packet type, source id, sequence number, target information)

- paquet type : Type de paquet (RREQUEST).
- source id : Identifiant de la source.
- sequence number : Numéro de séquence, il est utilisé pour éviter de transmettre le paquet en double.
- target information : Information qui identifie la cible.

Le format de paquet de réponse est : **reply** (packet type, source id, target id, the location of target, beacon node id).

- packet type : Type de paquet (REPLY).
- target id : Identifiant unique de la cible.
- the location of target : Emplacement de la cible.
- beacon node id : Identifiant du nœud d'entrée

### 3.3.3.3 Phase de détection

Si un capteur  $n$  est un nœud proche de la cible  $o$  qui se situe dans la face  $F_a$ , il devient un nœud d'entrée de  $F_a$  et suit la cible. Comme la cible est mobile, les capteurs coopèrent pour la détecter. Pour ce faire,  $n$  envoie un paquet de réveil à tout ses voisins de faces, lorsque les voisins reçoivent le paquet de réveil, ils restent en état actif et détectent la cible en collaboration. Par la suite, ils transmettent ce dernier aux capteurs qui se situent dans les faces du nœud d'entrée. Lorsque la cible se déplace de la zone de détection du nœud d'entrée, un autre nœud  $m$  peut détecter la cible et devient un nœud proche.

Lorsque le temps actif a expiré, les capteurs passent à l'état de sommeil. Par conséquent, le paquet de réveil est envoyé périodiquement par le nœud d'entrée jusqu'à ce que la cible se déplace.

Le format de paquet de réveil est comme suit : **wakeup** (packet type, ingress id, near-node id, object in face id, received node id, target id, the location of target, active time, face hop count)

- packet type : Type de paquet (WAKEUP).
- ingress id : Identifiant du nœud d'entrée.
- near-node id : Identifiant du nœud proche.
- object in face id : Identifiant de la face où se situe la cible.
- received node id : Identifiant du nœud qui va recevoir ce paquet.

- active time : Indique la période active des nœuds.
- face hop count : le nombre de faces par où la cible est passée. Initialement, il est défini à 1.

### 3.3.3.4 Phase de suivi

Tandis que la cible  $o$  est suivi, le réseau de capteur doit enregistrer les pistes (chemins) de la cible. Lorsque  $S$  reçoit la réponse du nœud d'entrée, elle atteint sa position et l'interroge pour la position suivante de la cible. Comme la cible se déplace, le nœud  $n$  arrête d'envoyer le paquet de réveil. Dans ce cas le nœud  $m$  devient le nœud d'entrée suivant. Chaque nœud d'entrée enregistre toutes les informations collectées et l'envoie à la source. Ce processus est répété jusqu'à ce que la cible est prise par la source.

Le format du paquet de demande est comme suit : **q\_next** (packet type, source id, target id, received beacon node id)

Le format du paquet de réponse est comme suit : **q\_next\_rep** (packet type, source id, target id, next beacon node id, the location of target)

- next beacon node id : Identifiant du nœud d'entrée suivant.

## 3.4 Raffinement

La structure de face permet aux nœuds de capteurs, de couvrir la cible d'une manière efficace et la détecter en coopération, donc on peut constater que ce protocole de suivi d'objets en mouvement, offre une bonne précision. Dans ce travail, nous nous intéressons à améliorer ce protocole, par l'optimisation de nombre de requêtes échangées et l'augmentation de la durée de vie du réseau.

### Amélioration du protocole

Rappelons que, dans la phase de détection, le nœud d'entrée envoie un paquet de réveil à tout ses voisins de face. Ensuite, lorsque ses voisins reçoivent ce paquet, ils le transmettent à leur tour à tous les nœuds qui appartiennent aux faces du nœud d'entrée ce qui signifie, que toutes les faces qui entourent le nœud d'entrée seront activées. Il s'avère qu'il y a un surplus de nœuds de capteurs activés, le fait, qu'il existe une possibilité d'activer moins de capteurs tout en gardant la précision du suivi.

Le raffinement du protocole s'effectue de la manière suivante : pour détecter et suivre la cible, il suffit d'envoyer le paquet de réveil aux voisins de face du nœud d'entrée, et de n'activer que la face dans laquelle se situe la cible.

### Exemple 1 (Avant l'amélioration)

Un exemple de détection d'objet mobile est illustré sur la figure 1.2. Supposons que la cible mobile est dans la face F1 à l'instant  $t$ . Trois capteurs  $\{n_1, n_2, n_3\}$  peuvent la détecter sachant que  $n_1$  est le nœud le plus proche. Lorsque  $n_1$  reçoit un paquet de demande provenant de la source S, il répond par un paquet de réponse à la source et devient un nœud d'entrée. Ensuite,  $n_1$  émet un paquet de réveil à tout ses voisins spatiaux.

Dans la figure 1.2,  $n_1$  diffuse le paquet de réveil (WAKEUP,  $n_1, n_1, -1, F1, o, L(o), t, 1$ ) à ses voisins  $\{n_2, n_6, n_9\}$ . Lorsque  $n_2$  reçoit le paquet de  $n_1$ , cela veut dire, que le paquet est dans la face F1.  $n_2$  transmet ce paquet à  $n_3$  puis  $n_3$  le transmet à  $n_4$  etc.. Tout les voisins spatiaux recevront ce paquet. Ces voisins spatiaux comprennent  $\{n_2, n_3, n_4, n_5, n_6\}$  dans F1,  $\{n_9, n_2\}$  dans F2 et  $\{n_6, n_7, n_8, n_9\}$  dans F3.

Si la cible se déplace vers la face F4 à l'instant  $t+2$ , elle doit traverser le bord( $n_5, n_6$ ). même si la cible se déplace vers  $n_5$  à l'instant  $t+1$ , elle reste encore dans la face F1. Soit  $n_5$  est le nœud le plus proche, il diffuse le paquet de réveil (WAKEUP,  $n_1, n_5, -1, F1, o, L(o), t, 1$ ) aux faces F1, F4, F5.

Si la cible rentre à la face F4 à  $t+2$ ,  $n_5$  est toujours le nœud le plus proche, comme le montre la figure 1.2(b).  $n_5$  diffuse le paquet de réveil à tout ses voisins spatiaux, à savoir,  $\{n_6, n_1, n_2, n_3\}$  dans F1,  $\{n_{10}, n_{11}, n_{12}, n_7, n_6\}$  dans F4 et  $\{n_4, n_{10}\}$  dans F5. En ce moment,  $n_5$  est le nœud d'entrée, F4 est la face dans laquelle se situe la cible, le nombre de saut de faces est 2 (F1, F2).

Le nœud  $n_1$  interroge  $n_5$  sur le prochaine nœud d'entrée, afin, de guider la source. La source suit la séquence des nœuds d'entrée pour chasser la cible.

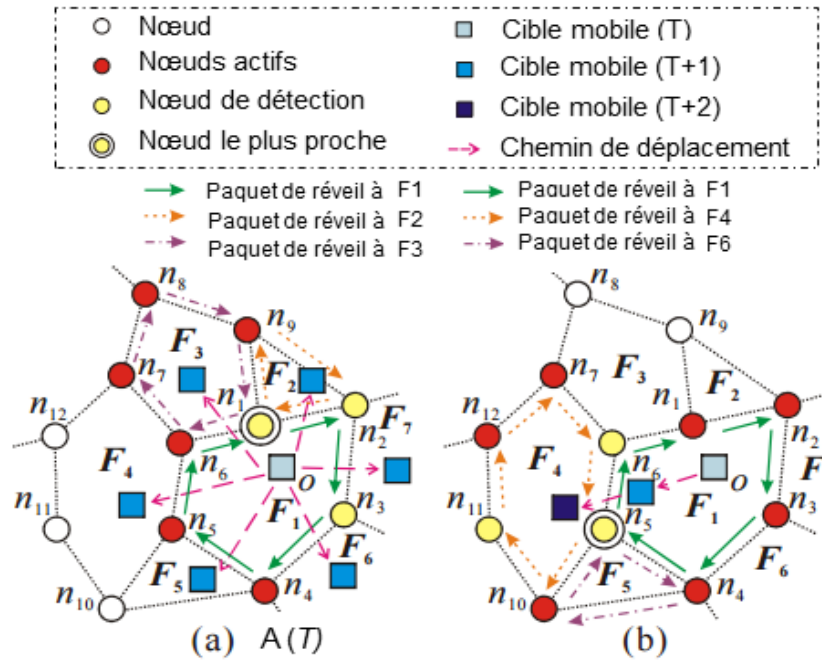


FIGURE 3.4 – Exemple de détection d'objet.

## Exemple 2 (Après l'amélioration)

Un exemple de détection d'objet mobile du protocole amélioré est illustré sur la figure 3.4. Supposons que la cible mobile est dans la face  $F_1$  à l'instant  $t$ . Trois capteurs  $\{n_1, n_2, n_3\}$  peuvent la détecter sachant que  $n_1$  est le nœud le plus proche. Lorsque  $n_1$  reçoit un paquet de demande provenant de la source  $S$ , il répond par un paquet de réponse et devient un nœud d'entrée. Ensuite,  $n_1$  émet un paquet de réveil qu'à ses voisins de faces (les nœuds qui sont reliés avec  $n_1$  par une arête de Gabriel).

Dans la figure 3.5,  $n_1$  émet le paquet de réveil (WAKEUP,  $n_1, n_1, -1, F_1, o, L(o), t, 1$ ) à ses voisins  $\{n_2, n_6, n_9\}$ . Lorsque  $n_2$  reçoit le paquet de  $n_1$ , il constate que la cible se situe dans la face  $F_1$ , et comme, il appartient à la face  $F_1$ , il diffuse le paquet vers  $F_1$  seulement. Alors que, le nœud  $n_9$  ne transmettra pas ce paquet car il n'appartient pas à la face  $F_1$ . En final il y a que  $\{n_2, n_3, n_4, n_5, n_6, n_9\}$  qui seront activés.  $\{n_7, n_8\}$  restent inactifs

La cible peut se déplacer à l'une des adjacentes à la période  $T+1$ ,  $F_2, F_3, F_4, F_5, F_6$  ou  $F_7$  dans la figure 1.3. On remarque, que tout les nœuds sensés détectés la cible sont actifs

$\{n_2, n_3, n_4, n_5, n_6\}$ .

Si la cible se déplace vers la face F4 à l'instant  $t+2$ , elle doit traverser le bord  $(n_5, n_6)$ . même si la cible se déplace vers  $n_5$  à l'instant  $t+1$ , elle reste encore dans la face F1. Soit  $n_5$  est le nœud le plus proche, il transmet le paquet de réveil (WAKEUP,  $n_1, n_5, -1, F1, o, L(o), t, 1$ ) à ses voisins de faces  $\{n_6, n_4, n_{10}\}$ . Ensuite le paquet sera retransmis qu'aux nœuds situant dans la face F1. Si la cible rentre à la face F4 à  $t+2$ ,  $n_5$  est toujours le nœud le plus proche, comme le montre la figure 2(b).  $n_5$  émet le paquet de réveil à ses voisins de faces, à savoir,  $\{n_6, n_4, n_{10}\}$ . Cette fois-ci le paquet sera transmis aux nœuds qui se situent dans la face 4. En ce moment,  $n_5$  est le nœud d'entrée, F4 est la face dans laquelle se situe la cible, le nombre de saut de faces est 2 (F1, F2).

Le nœud  $n_1$  interroge  $n_5$  sur le prochaine nœud d'entrée, afin, de guider la source. La source suit la séquence des nœuds d'entrée pour chasser la cible.

**Remarque :** A partir de cet exemple, on peut en déduire qu'on a pu diminuer le nombres de requêtes échangées, et minimiser la consommation énergétique. Tout en gardant la précision du suivi.

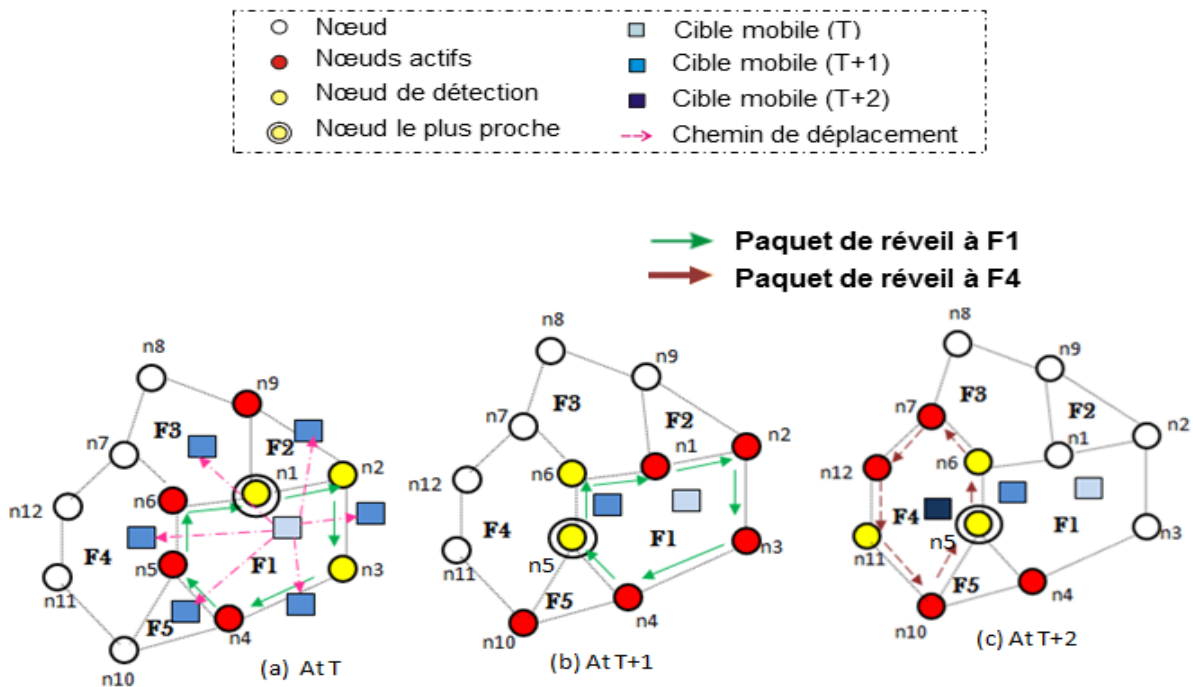


FIGURE 3.5 – Détection d'objet Après l'amélioration du protocole.

### 3.4.1 Pseudo algorithme de suivi d'objet après raffinement

#### Notations

- S : Source.
- $N_i$  : Nœud de capteur.
- $NV_i$  : Ensemble de voisins de faces du nœud  $N_i$ .
- NP : Nœud proche de la cible.
- NE : Nœud d'entrée.
- NF : Face du nœud.

Lors de la phase d'initialisation, chaque nœud de capteur exécute l'algorithme suivant :

---

**Algorithm 1** Initialisation

---

```
for all  $N_i$  do  
    Send(HELLO, NVi); /* envoyer un message hello à tout les voisins de face.  
    Calculate_Face_Node();  
end for
```

---

**Remarque :** `Calculate_Face_Node()` est une fonction qui permet de calculer les nœuds de face.

Après avoir exécuter la phase d'initialisation, chaque nœud de capteur sera capable de suivre la cible et exécuter l'algorithme de suivi suivant :



---

**Algorithm 2** Object\_Tracking

---

```

while NOT (receive(request_paquet(), S)) do
    wait();
end while
/* Lors de la réception du paquet request()
repeat
    if ((detected target) && (Ni == NP)) then
        send(reply, S);
         $n_i \leftarrow NE$ ;
        Send(wakeup_paquet(),  $NV_i$ );
        Calculate_Target_Location();
    end if
    if (receive(wakeup_paquet(), m)) then
        etat  $\leftarrow$  Actif;
        if (NF_id == ObjectFace_id) then
            suiv  $\leftarrow$  { $NV_i \in$  ObjectFace_id} - {m};
            Send(wakeup_paquet, suiv);
        end if
        if (Ni == NE) then
            send(q_next_rep(), S);
        end if
    end if
until Target_Chased

```

---

## Conclusion

Nous avons présenté dans ce chapitre le suivi d'objet mobile, en utilisant les réseaux de capteurs sans fil. Le suivi de cible est une application très utile, elle reste néanmoins difficile à implémenter étant donné les contraintes énergétiques liées au type de réseaux ainsi qu'aux données qui y transitent.

Dans ce chapitre, nous avons exposé d'une part, les contraintes liées au suivi et d'autre part, nous avons décrit en détail notre contribution. L'apport principale de notre proposition est de diminuer le nombre de messages échangés entre les nœuds de capteurs lors du suivi de cible et de n'activer que les nœuds qui se trouvent sur la trajectoire de la cible au moment où cette dernière est à leur portée.

# Simulation

---

## 4.1 Introduction

Avant la mise en place du protocole proposé, le déploiement d'un réseau de capteurs nécessite une phase de simulation, afin de s'assurer du bon fonctionnement de tous les dispositifs. En effet, pour de grands réseaux le nombre de capteurs peut atteindre plusieurs milliers, ce qui engendre un coût financier relativement important. Il faut donc réduire au maximum les erreurs de conception possibles en procédant à une phase de validation.

Dans ce chapitre, nous allons en premier lieu, présenter l'environnement de simulation ensuite, nous allons présenter les résultats pour l'étudier avec discussion de résultat.

## 4.2 Environnement de simulation

La simulation de notre algorithme constitue une étape importante de notre travail puisque on peut prouver les améliorations effectuées en terme d'économie d'énergie et de prolongement de temps de vie global du réseau à l'aide des résultats fournis. Pour effectuer nos simulations, nous avons fait appel au logiciel MATLAB. Les résultats fournis par la simulation sont comparés avec l'algorithme DOT(Dynamic Object Tracking).

Le graphe présenté dans la figure 1.1 est un graphe particulier de graphe planaire appelé graphe de Gabriel. Les sommets représentent les nœuds de capteurs et les arêtes définissent le lien entre les nœuds pouvant directement communiquer.

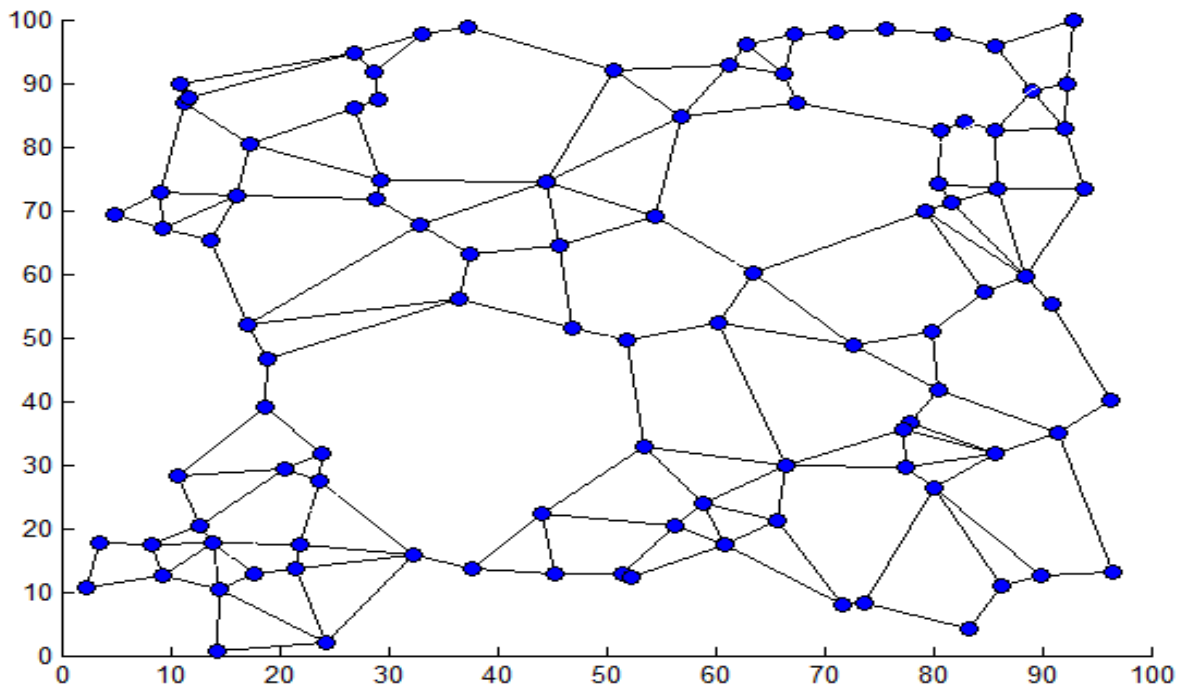


FIGURE 4.1 – *Modèle d'expérimentation.*

### 4.2.1 Contexte et paramètres de simulation

Dans cette simulation, notre modèle d'expérimentation est établie sur  $200 * 200 m^2$  présenté par la figure 3.1. Nous supposons que tout les nœuds ont une position fixe durant la période de simulation. De plus, Tous les nœuds du réseau commencent la simulation par une énergie initiale égale à 0.5 J et la portée de communication est de 50 mètres. On estime que la taille d'un paquet est égale à 500 bits. L'objet mobile se déplace avec une vitesse de 15 mètres et le temps de pause est de 3 seconde.

Nous avons utilisé un modèle simplifié pour la dissipation d'énergie radio, proposé dans [1] où chaque nœud dans le réseau va consommer de l'énergie dans le cas où il va transmettre des paquets de données ainsi que dans le cas où il reçoit des paquets de donnée sans oublier le cas où il exécute des opérations de traitement de données pour chaque transaction d'envoi ou de réception (collection, agrégation). Les paramètres de simulation utilisés sont résumé dans le tableau ci-dessous

Paramètre	valeur
La surface du réseaux	200 * 200 $m^2$
Nombre de nœuds	200
Nombre de cibles mobiles	1
L'énergie initiale des nœuds	0.5 joules
Taille de paquet	500 bytes
Portée de communication	50 mètres
Portée de détection	25 mètres
$E_{elect}$	50 x $10^{-6}$
$E_{fs}$	$10^{-9}$
$E_{mp}$	1.3 * $10^{-11}$
$E_{EDA}$	5 * $10^{-6}$

TABLE 4.1 – Paramètres de simulation.

L'énergie consommée pour transmettre q-bits de données sur une distance d à un nœud de capteur est égale à :

$$E_{Tx}(q,d) = \begin{cases} q * E_{elect} + q * E_{fs} * d^2 & \text{si } d < 0 \\ q * E_{elect} + q * E_{mp} * d^4 & \text{si } d \geq 0 \end{cases}$$

Avec

$$d_0 = \sqrt{\frac{E_{fs}}{E_{mp}}}$$

Où

$E_{elect}$  est la quantité d'énergie consommée par les circuits électroniques (énergie de calcul).

$E_{fs}$  est  $E_{mp}$  sont respectivement l'énergie dans l'espace libre et celle du modèle multi-chemins.

L'énergie consommée pour recevoir q-bit de données est égale à :

$$E_{rx}(q) = q * E_{elect}$$

Chaque nœud qui a épuisé sa ressource d'énergie est considéré comme mort. Par conséquent, il ne peut ni transmettre ni recevoir des données.

## 4.3 Simulation et évaluation de performances

Pour évaluer les performances du protocole proposé, nous avons procédé à le comparé avec le protocole DOT. Pour cela nous avons effectué des simulations avec les mêmes paramètres et métriques pour les deux protocoles.

### 4.3.1 Métrique à évaluer

L'une des métriques clés les plus utilisées pour évaluer les stratégie de suivi de cible dans les réseaux de capteurs est la consommation énergétique.

#### 4.3.1.1. Consommation énergétique

La consommation énergétique est une métrique d'évaluation importante, particulièrement dans le contexte des réseaux de capteurs sans fil où la durée de vie du réseau dépend du Coût énergétique des applications implémentées. De nombreux facteurs in-fluent directement sur l'énergie tels que le nombres de messages échangés durant le processus de suivi. Elle peut être évaluée en terme d'énergie consommée, d'énergie résiduelle ou de durée de vie moyenne d'un nœud.

### 4.3.2 Résultats de la simulation

En utilisant logiciel Mtalab, nous avons évalué la consommation moyenne d'énergie d'un nœud capteur. La figure 1.2 montre la variation de l'énergie consommée en fonction du nombre de nœuds dans le réseau et la figure 1.3 illustre l'énergie consommée par rapport à la portée de communication.

#### 4.3.2.1 Variation de la consommation d'énergie en fonction du nombres de nœuds dans le réseau

Comme le montre la figure 1.2, l'énergie consommée en moyenne par notre algorithme est inférieure à celle consommé par le protocole DOT, cela s'explique par le fait qu'il y a plus de nœuds actifs et de messages échangés. Lorsque le nombre de capteurs augmente, le nombre d'arêtes augmente, et, les faces du réseau deviennent plus petites. De ce fait, la consommation d'énergie diminue.

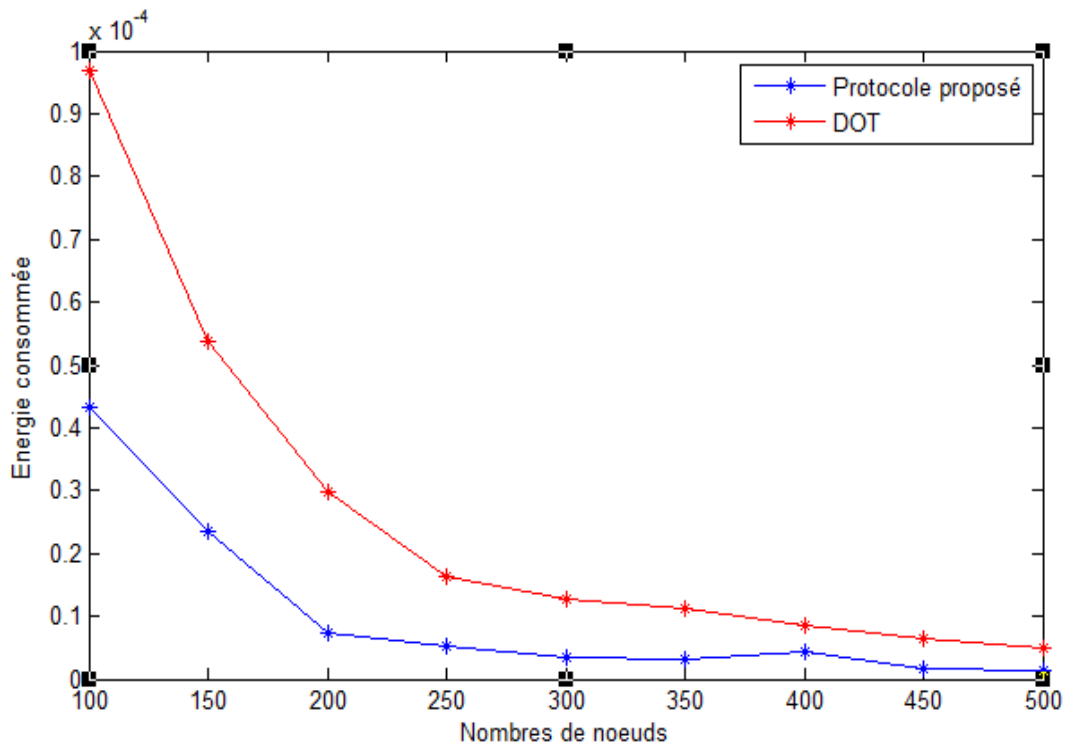


FIGURE 4.2 – Variation de la consommation d'énergie en fonction du nombres de nœuds dans le réseau.

#### 4.3.2.2 Variation de la consommation d'énergie en fonction du rayon de communication

La figure 1.3 montre également, que notre méthode améliore le protocole DOT en terme de consommation énergétique même en variant le rayon de communication. l'analyse et l'explication des courbes de la figure 1.3 peut être comme suit :

On peut diviser la courbe en deux parties, dans la première partie (10 à  $R_c=30$  mètres), on remarque que les deux courbes augmentent relativement avec le rayon de communication. Dans la deuxième partie ( $R_c \geq 30$ ), l'énergie consommée par les nœuds de capteurs devienne stable, on justifiant ce point par le fait que la structure du réseau se stabilise et reste inchangé dès que le rayon de communication égale à 30 mètres.

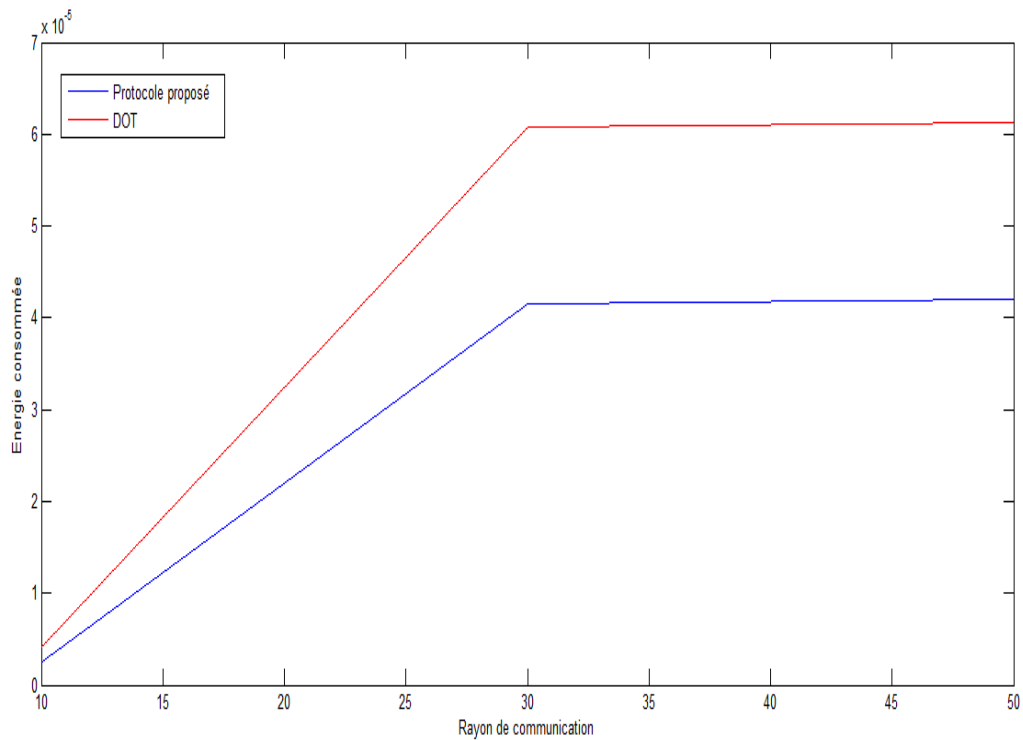


FIGURE 4.3 – Variation de la consommation d'énergie en fonction du rayon de communication.

### 4.3.2.3 Nombres de messages échangés

le nombre de messages échangés représente le coût de communication de l'algorithme de suivi. Nous l'obtenant en comptabilisant le nombre de messages échangés (reçus et transmis) sur un ensemble de trajectoires aléatoires.

La figure 1.4 représente le nombre de messages échangés en fonction du temps de simulation. Nous observons que le protocole DOT a un coût de communication plus élevé que notre algorithme tout au long de la simulation. Cela est dû au surcoût provoqué par l'activation de toutes les faces spatiales.

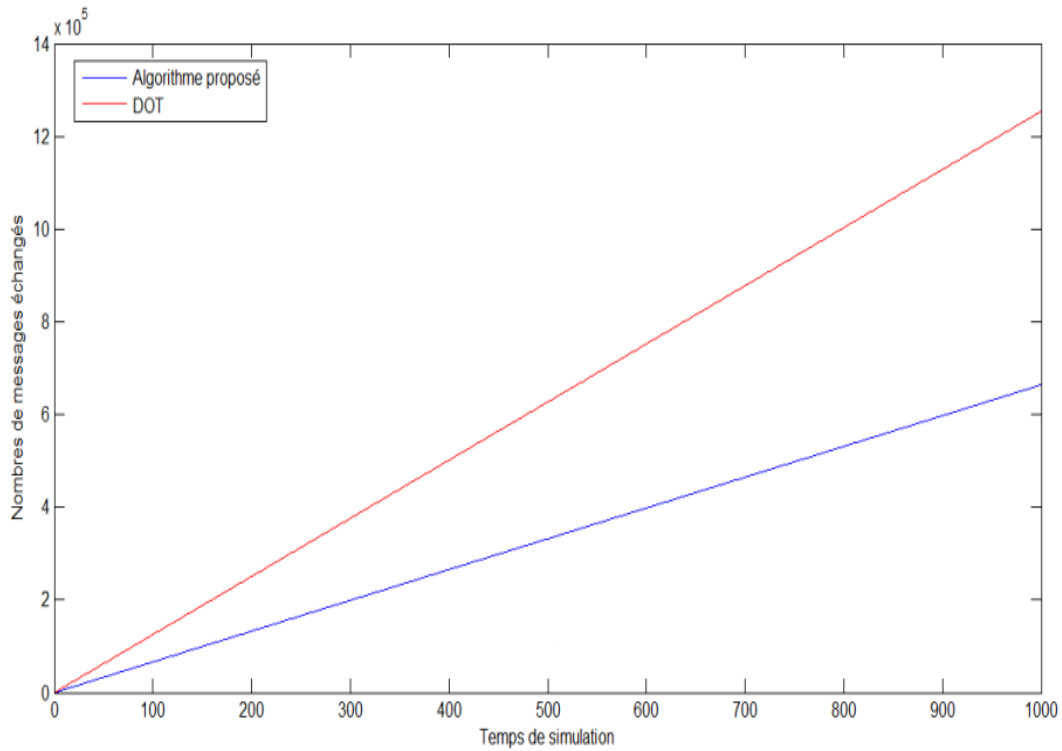


FIGURE 4.4 – *Nombres de messages échangés en fonction du temps simulation.*

#### 4.3.2.4 La durée de vie du réseau

Pour bien approfondir l'évaluation des performances de notre algorithme, nous avons comparé la durée de vie du réseau entre DOT et notre algorithme. Tout au long de la simulation, nous avons mesuré l'énergie résiduelle des nœuds de capteurs afin de calculer le nombre totale des nœuds vivants.



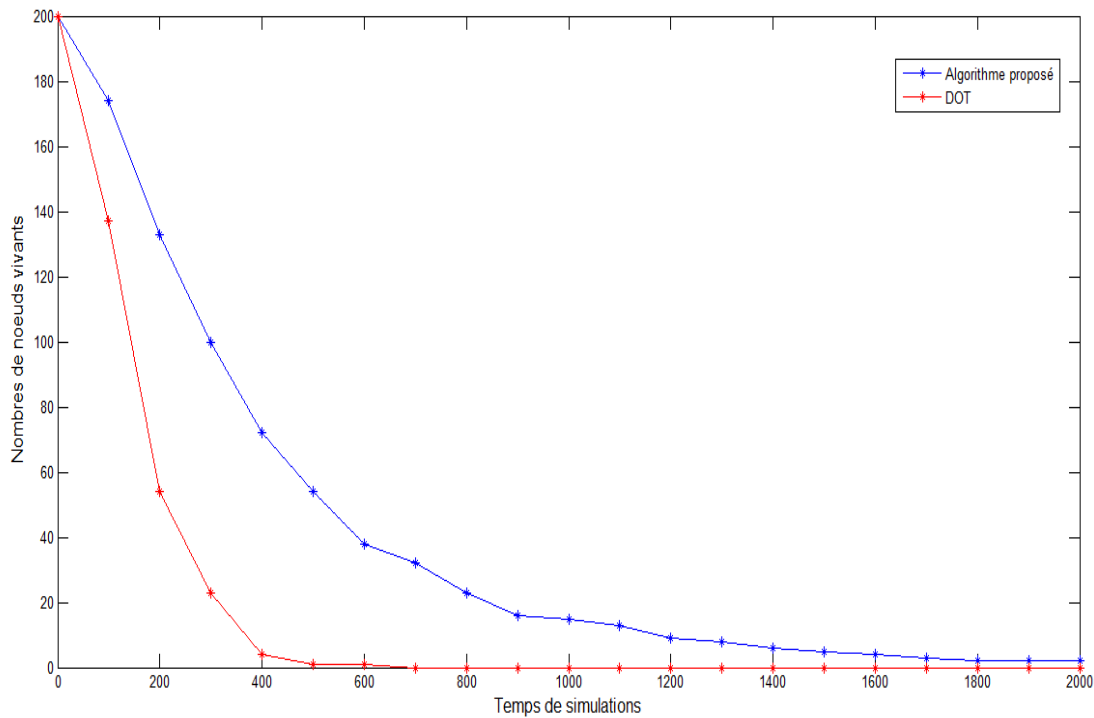


FIGURE 4.5 – *Le nombre de nœuds vivants dans le réseau en fonction du temps de simulation.*

D'après les courbes de la figure 1.4, on constate que le dernier nœud épuise son énergie dans l'algorithme DOT après 600 unités de simulation, alors que dans notre algorithme, le dernier nœud épuise son énergie après 2000 unités. Par conséquent, nous constatons que notre algorithme est plus efficace en terme de durée de vie du réseau que DOT.

## 4.4 Conclusion

Dans ce chapitre, nous avons présenté la mise en place de notre algorithme de suivi de cible mobile. Nous avons d'abord défini l'environnement de simulation avec les paramètres utilisés. Puis, nous avons mené des simulations dans le but de comparer les performances de notre algorithme. D'après les résultats obtenus, nous concluons que notre solution offre une meilleure performance par rapport à l'algorithme DOT en termes de consommation d'énergie et de durée de vie du réseau.

## Conclusion générale

L'utilisation des réseaux de capteurs sans fil(RCSF) pour le suivi(tracking) d'objets mobiles a fourni une solution pratique à une grande variété d'applications, y compris, par exemple, les opérations militaires, le suivi des intrus, la surveillance à l'intérieur des immeubles à bureau, etc. Bien que, beaucoup de recherches ont été faites dans ce domaine, les échecs ne sont pas pris en compte dans la plupart des solutions existantes. Cependant, les échecs doivent être manipulés avec précaution dans les applications de suivi de cible en raison de leur nature imprévisible et dynamique de communication, tels que l'épuisement de l'énergie du capteur, les conditions de l'environnement et les liens de communication instables

Dans ce travail, nous avons proposé un algorithme distribué de suivi de cibles mobiles dans les RCSFs. Le réseau est modélisé par un type particulier de graphe planaire, appelé graphe de Gabriel. En effet, cette structure permet aux nœuds de capteurs de couvrir la cible d'une manière efficace et la détecter en coopération.

Un scénario qui illustre le fonctionnement de l'algorithme est qu'un petit robot(comme abeille) veut chasser un ennemi ou un animal sauvage. Ce dernier, envoie une demande au réseau de capteurs pour obtenir l'emplacement de la cible. Par suite, un groupe de capteurs se coopèrent et assiègent la cible afin de la suivre.

L'idée principale de l'algorithme est que dans la phase de détection, le nœud le plus proche de la cible (nœud d'entrée) envoie un message de réveil qu'à ses voisins de face et réveille que la face dont la cible se situe. Tandis que la cible est suivie, le réseau de capteur doit enregistrer les pistes (chemins) de la cible. Lorsque le robot reçoit la réponse du nœud d'entrée, il atteint sa position et l'interroge pour la position suivante de la cible. La séquence de nœuds d'entrée forme la trajectoire de la cible. Ce processus de suivi est répété jusqu'à ce que la cible est attrapée par le robot.

Finalement, nous avons concrétisé notre proposition théorique, en faisant appel au logiciel MATLAB qui nous a permis d'effectuer nos simulations. Les résultats obtenus montrent que notre solution offre une meilleure performance par rapport à l'algorithme DOT comparé en termes de consommation d'énergie et de durée de vie du réseau.

Les réseaux de capteurs sont le monde de demain, avec un très large éventail d'utilisation et en particulier aux applications de suivi d'objets. D'un point de vue personnel ce projet

m'a permis d'acquérir de solides connaissances dans ce domaine. Toutefois, cet effort ne peut être considéré que comme un début pour un travail plus profond qui aurait comme objectif une optimisation plus poussée à l'algorithme proposé pour augmenter son efficacité.

# Bibliographie

- [1] I.F. Akyildiz, T. Melodia, and K.R. Chowdhury. A survey on wireless multimedia sensor networks. *Computer networks*, 51(4), pp. 921–960, (2007).
- [2] I.F. Akyildiz, T. Melodia, and K.R. Chowdhury. Wireless multimedia sensor networks : Applications and testbeds. *Proceedings of the IEEE*, 96(10), pp. 1588–1605, (2008).
- [3] I.F. Akyildiz, S. Weilian, Y. Sankarasubramaniam, and E. Cayirci. A survey on sensor networks. *Communications magazine, IEEE*, 40(8), pp. 102–114, (2002).
- [4] I.F. Akyildiz, S. Weilian, Y. Sankarasubramaniam, and E. Cayirci. Wireless sensor networks : A survey. *Computer Networks Journal (Elsevier)*, 38(4), pp. 393–422, (2002).
- [5] Y. Al-Obaisat and R. Braun. On wireless sensor networks : Architectures, protocols, applications, and management. *Institute of Information and Communication Technologies University of Technology, Sydney, Australia*, (2000).
- [6] Y. Al-Obaisat and Robin Braun. On wireless sensor networks : Architectures, protocols, applications, and management. *Institute of Information and Communication Technologies*.
- [7] A. Arora, P. Dutta, S. Bapat, V. Kulathumani, H. Zhang, V. Naik, V. Mittal, H. Cao, M. Demirbas, M. Gouda, and al. A line in the sand : A wireless sensor network for target detection, classification, and tracking. *Computer Networks*, 46(5), pp. 605–634, (2004).
- [8] I. Boulanouar. Algorithmes de suivi de cible mobile pour les réseaux de capteurs sans fil. *Thèse de Doctorat en Informatique, Université Paris-Est, Mars*, (2015).
- [9] J. Campbell, P.B. Gibbons, S. Nath, P. Pillai, S. Seshan, and R. Sukthankar. Irisnet : an internet-scale architecture for multimedia sensors. *In Proceedings of the 13th annual ACM international conference on Multimedia*, pp. 81–88, (2005).

- [10] C.YPeng and al. Efficient in-network moving object tracking in wireless sensor networks. *Department of Computer Science and Information Engineering-National Chiao Tung University*, (2004).
- [11] D.M. Doolin and N. Sitarl. Wireless sensor networks for habitat monitoring. In *Smart Structures and Materials. International Society for Optics and Photonics*, pp. 477–484, (2005).
- [12] M. Fitzgerald. Technnology review : Tracking a shopper’s habits. *Journal of telemedicine and telecare*, 2(3), pp. 108–113, (2008).
- [13] T. Gosnell, J. Hall, C. Jam, D. Knapp, Z. Koenig, S. Luke, B. Pohl, A. Schach, v. Wittenau, and J. Wolford. Gamma-ray identification of nuclear weapon materials. *Technical report, Lawrence Livermore National Lab., Livermore, CA (US)*, (1997).
- [14] K. Hazra and B. N. B. Ray. Target tracking in wireless sensor network : A survey. *International Journal of Computer Science and Information Technologies, (IJCSIT)*, 6(4), pp. 3720-3723, (2015).
- [15] P. Johnson and D.C. Andrews. Remote continuous physiological monitoring in the home. *Journal of telemedicine and telecare*, 2(2), pp. 107–113, (1996).
- [16] H. Kim, E. Kim, and K. Han. An energy efficient tracking method in wireless sensor networks. In *International Conference on Next Generation Wired/Wireless Networking*, pages 278–286. Springer, 2006.
- [17] S. Kita, K. Murai, and H. Ogatsu. *Color image processing*. Patent 5,636,290, US, (1997).
- [18] H. T. Kung and D. Vlah. Efficient location tracking using sensor networks. *Proceedings of IEEE Wireless Communications and Networking Conference (WCNC)*, (2003).
- [19] M. LEHSAINI. Diffusion et couverture basées sur le clustering dans les réseaux de capteurs : application à la domotique. *Thèse de Doctorat en Informatique, Université A.B Tlemcen Faculté des Sciences pour l’Ingénieur et Université de Franche-Comté U.F.R Sciences et Techniques École Doctorale SPIM*, (2009).
- [20] A. Mainwaring, D. Culler, J. Polastre, R. Szewczyk, and J. Andersonl. Wireless sensor networks for habitat monitoring. In *Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications, ACM*, pp. 88–97, (2002).
- [21] A. Makhoul. Réseaux de capteurs : localisation, couverture et fusion de données. *Thèse de Doctorat en Informatique, Université de Franche-Comté, novembre*, (2008).
- [22] T. Naumowicz, R. Freeman, A. Heil, M. Calsyn, E. Hellmich, A. Brändle, T. Guilford, and J. Schillerl. Autonomous monitoring of vulnerable habitats using a wireless sensor

- network. *In Proceedings of the workshop on Real-world wireless sensor networks, ACM*, pp. 51–55, (2008).
- [23] E.M. Petriu, N.D. Georganas, D.C. Petriu, D. Makrakis, and V.Z. Grozal. Sensor-based information appliances. *Instrumentation and Measurement Magazine, IEEE*, 3(4), pp. 31–35, (2000).
- [24] M. Phil. A survey on target tracking techniques in wireless sensor network. *International Journal of Engineering Sciences and Research Technology (IJCSIT)*, 3(4), pp. 3031-3035, (2014).
- [25] D. Puccinelli and M. Haenggi. Wireless sensor networks : applications and challenges of ubiquitous sensing. *Circuits and Systems Magazine, IEEE*, 5(3), pp. 19–31, (2005).
- [26] K. Ramya, K. P. Kumar, and V. Srinivas Rao. A survey on target tracking techniques in wireless sensor networks. *International Journal of Computer Science and Engineering Survey (IJCSES)*, 3(4).
- [27] J. Schulz, F. Reichenbach, J. Blumenthal, and D. Timmermannl. Low cost system for detecting leakages along artificial dikes with wireless sensor networks. *In Proceedings of the workshop on Real-world wireless sensor networks*, pp. 66–70. *ACM*, (2008).
- [28] S. Tilak, N. Abu-Ghazaleh, and W. Heinzelman. A taxonomy of wireless micro-sensor network models. *A taxonomy of Wireless Micro-sensor Network Models, ACM SIGMOBILE, Mobile Computing and Communications Review*, 6(2), pp. 28-36, (2002).
- [29] S. P. M. Tran and T. A. Yang. Oco : Optimized communication and organization for target tracking in wireless sensor networks. *Proceedings of the IEEE International Conference on Sensor Networks*.
- [30] H. Tsai, C. p. Chu, and T. s. Chen. Mobile object tracking in wireless sensor networks. *Computer Communications* 30, (2007).
- [31] M. Walchli, P. Skoczylas, M. Meer, and T. Braun. Distributed event localization and tracking with wireless sensors. *in Proceedings of the 5th international Conference on Wired/Wireless internet Communications*, pp. 23-25.
- [32] Z. Wang, W. Lou, Z. Wang, J. Ma, and Honglong Chen. A hybrid cluster-based target tracking protocol for wireless sensor networks. *Hindawi Publishing Corporation, International Journal of Distributed Sensor Networks, Article ID 494863, 16 pages*.
- [33] Y. Xu, J. Winter, and W.C. Lee. On localized prediction for power efficient object tracking in sensor networks. *In Proc. 1st International Workshop on Mobile Distributed computing, Providence RI*, pp. 434-439, (2003).

- [34] Y. Xu, J. Winter, and W.C. Lee. Prediction-based strategies for energy saving in object tracking sensor networks. *In Proc. International Conference on Mobile Data Management, Berkeley, CA, pp. 346-357, (2004).*
- [35] X. Yingqi, J. Winter, and W. C. Lee. Dual prediction-based reporting for object tracking sensor networks. *Proceedings of the First Annual International Conference on Mobile and Ubiquitous Systems : Networking and Services, IEEE.*
- [36] W. Zhang. Dynamic convoy tree-based collaboration for target tracking in sensor networks. *Transactions On Wirlessnetwork,30(5), (2004).*