

UNIVERSITÉ KASDI MERBAH OUARGLA ALGERIE  
FACULTÉ DES NOUVELLES TECHNOLOGIES DE L'INFORMATION ET DE LA  
COMMUNICATION  
DÉPARTEMENT D'INFORMATIQUE ET DE TECHNOLOGIE DE L'INFORMATION



**Domaine : Mathématiques et Informatique**  
**Piste: Informatique**  
**Spécialité: Administration et sécurité des réseaux**  
**Mémoire Master ACADÉMIQUE**  
**Présenté par:**  
**Benraghda ahmed yasser**  
**Berguia sara**  
**Titre:**

---

**Optimisation de la sélection des cluster heads  
pour améliorer l'efficacité des réseaux de  
capteurs sans fil**

---

**Débat public sur :2022.2023**

**Les membres du comité:**

Dr.Abdessalam Messiaid    encadré    UKM    Ouargla  
Dr. Abdelatif Cheradid    Examiner    UKM    Ouargla  
Dr.Mohammed Benbeziane    président    UKM    Ouargla

**Année académique: 2022 / 2023**



# Reconnaissance

Tout d'abord, nous remercions Dieu Tout-Puissant de nous avoir donné du courage et Volonté dans les moments difficiles qui nous a donné force et patience pour accomplir ce travail. Nous exprimons notre profonde gratitude et notre respect à Notre cher Dr : **Abdessalam Messiaid**

Pour sa volonté d'accepter notre supervision tout le temps Il nous a donné tous les conseils qu'il nous a donnés. Nous remercions également tous nos professeurs , qui nous a donné les bases de la science. Sans oublier nos chers professeurs, les membres du jury et tous les enseignants Département de mathématiques et des médias à. Merci à nos camarades de classe de 2022/23 Pour leur partenariat et leur aide à nous tous,

Cette page n'aurait peut-être pas été possible d'écrire sans le soutien moral des membres de notre famille. Nos plus chaleureux remerciements à tous ceux de près et de loin Contribuez à la réalisation de cet humble travail

# Dédicace

Je dédie cet humble travail à nos parents, **Hossine** et **Ayoub**, qui m'ont aidé avec cela et plus encore Son soutien et son sacrifice tout au long de ma vie sont mon espoir. fréquentation pour réussir mes études.

A ma chère mère **Warda** et **Hasina** pour leur soutien moral et affectif pour Mes sœurs sont **chaima Malak**, mon frère **Belkacem Malak**.

pour Toute ma famille. pour tous mes amis. A mon amie et soeur **Shifa** pour son soutien A un ami de l'islam pour aide et soutien A tous mes professeurs qui ont fait de leur mieux pour moi Donnez-moi le plus d'informations possible sur mes études. A tous mes camarades de classe

A toutes les personnes qui nous ont aidé de près ou de loin pour compléter cette lettre.

## Abstract

The research field of data routing in wireless sensor networks (WSN) is very active due to the peculiarities of these networks, where the energy consumption is considered to be one of the main challenges of communications protocols. Hierarchical data routing in WSNs is a class specific routing protocol that involves the integration of the physical network into a logical hierarchy system, in order to optimize energy consumption. The concept of biomimicry consists in artificially reproducing essential properties of one or more biological systems. Within the framework of this thesis, the objective is to design a hierarchical routing protocol based on the biomimicry. This protocol aims to minimize the energy consumption and, by therefore, to prolong the lifetime of the network. In summary, the thesis proposes to design a hierarchical routing protocol based on biomimicry in RCHS. The main objective is to optimize energy consumption and increase the duration life of the network.

**Key words:** biomimicry, HHO, LEACH, clustering, optimization, hierarchical, routing, energy consumption.

## ملخص

يعتبر مجال البحث الخاص بتوجيه البيانات في شبكات الاستشعار اللاسلكية (وا) نشطًا جدًا نظرًا لخصائص هذه الشبكات ، حيث يعتبر استهلاك الطاقة أحد التحديات الرئيسية لبروتوكولات الاتصال المقترحة. يعد توجيه البيانات الهرمي في شبكات وا فئة محددة من بروتوكولات التوجيه التي تتضمن إعادة هيكلة الشبكة المادية إلى نظام هرمي منطقي ، بهدف تحسين استهلاك الطاقة. يتكون مفهوم محاكاة الطبيعة من إعادة إنتاج الخصائص الأساسية لنظام بيولوجي واحد أو أكثر بشكل مصطنع. كجزء من هذه الأطروحة ، يتمثل الهدف الرئيسي في تصميم بروتوكول توجيه هرمي يعتمد على محاكاة الطبيعة. يهدف هذا البروتوكول إلى تقليل استهلاك الطاقة ، وبالتالي إطالة عمر الشبكة. باختصار ، تقترح الأطروحة تصميم بروتوكول توجيه هرمي يعتمد على المحاكاة الحيوية في غنصذ. الهدف الرئيسي هو تحسين استهلاك الطاقة وزيادة عمر الشبكة.

**الكلمات المفتاحية:** التحسين ، التسلسل الهرمي ، التوجيه ، استهلاك الطاقة ، هاريس هوكس ، التسلسل الهرمي التجميعي التكيفي منخفض الطاقة ، التجميع ، مستوحاة من الطبيعة

## Resume

Le domaine de recherche du routage des données dans les réseaux de capteurs sans fil (RCSF) est très actif en raison des particularités de ces réseaux, où la consommation d'énergie est considérée comme l'un des principaux défis des protocoles de communication proposés. Le routage hiérarchique des données dans les RCSF est une classe spécifique de protocoles de routage qui implique la restructuration du réseau physique en un système de hiérarchie logique, dans le but d'optimiser la consommation d'énergie. Le concept de bio mimétisme consiste à reproduire artificiellement les propriétés essentielles d'un ou plusieurs systèmes biologiques. Dans le cadre de ce mémoire, l'objectif principal est de concevoir un protocole de routage hiérarchique basé sur le bio mimétisme. Ce protocole vise à minimiser la consommation d'énergie et, par conséquent, à prolonger la durée de vie du réseau. En résumé, le mémoire propose de concevoir un protocole de routage hiérarchique basé sur le bio mimétisme dans les RCSF. L'objectif principal est d'optimiser la consommation d'énergie et d'accroître la durée de vie du réseau.

**Mots clés:** HHO, LEACH, clustering, optimisation , hiérarchique, routage , consommation d'énergie .

# Contents

<b>I</b>	<b>Introduction générale</b>	<b>2</b>
1	Context general . . . . .	2
2	Motivations et problématique . . . . .	3
3	Contributions et structure du mémoire . . . . .	4
<b>II</b>	<b>Généralités sur les réseaux de capteurs sans fil</b>	<b>2</b>
1	Introduction . . . . .	2
2	Capteurs . . . . .	3
2.1	Définition . . . . .	3
2.2	Types des capteurs . . . . .	3
2.2.1	Les capteurs actifs . . . . .	3
2.2.2	Les capteurs passif . . . . .	4
2.3	Composants de capteur . . . . .	4
2.3.1	Architecture matérielle . . . . .	5
2.3.2	Architecture logicielle . . . . .	6
3	Réseaux de Capteur sans fil . . . . .	7
3.1	Définition . . . . .	7
3.2	Architecture d'un réseau de capteurs sans fil . . . . .	7
3.2.1	RCSF Plat . . . . .	7
3.2.2	RCSF hiérarchiques . . . . .	9
3.3	Caractéristiques d'un RCSF . . . . .	9
3.4	Architecture protocolaire d'un RCSF . . . . .	10
3.5	Classification des RCSF . . . . .	11
3.6	Domaines d'application de RCSF . . . . .	12
3.6.1	Applications militaires . . . . .	13
3.6.2	Applications médicales . . . . .	13

3.6.3	Applications domestiques . . . . .	13
3.6.4	Applications environnementales . . . . .	14
3.6.5	Applications commerciales . . . . .	14
<b>4</b>	<b>Conclusion . . . . .</b>	<b>14</b>
<b>III</b>	<b>Routage dans RCSF . . . . .</b>	<b>16</b>
<b>1</b>	<b>Introduction . . . . .</b>	<b>16</b>
<b>2</b>	<b>Le routage dans les RCSFs . . . . .</b>	<b>16</b>
2.1	Les type de routage . . . . .	17
2.1.1	Protocole proactif . . . . .	17
2.1.2	Protocole réactif . . . . .	17
<b>3</b>	<b>Protocoles de routage . . . . .</b>	<b>17</b>
3.1	les protocole de routage géographique . . . . .	17
3.2	Les protocoles de routage plat . . . . .	18
3.3	Protocoles de routage hiérarchiques . . . . .	19
<b>4</b>	<b>Approche basée sur le clustering . . . . .</b>	<b>19</b>
<b>5</b>	<b>Notions de base . . . . .</b>	<b>20</b>
5.1	Définition . . . . .	20
5.1.1	Nœud membre . . . . .	21
5.1.2	cluster . . . . .	22
5.1.3	Cluster head . . . . .	22
5.1.4	Station de base . . . . .	22
5.1.5	Utilisateur final . . . . .	22
5.2	Processus de clustering . . . . .	23
5.2.1	Phase d'élection des cluster-heads: . . . . .	23
5.2.2	Communication intra-cluster et inter-cluster : . . . . .	23
5.2.3	Maintenance des clusters . . . . .	24
5.3	Avantages de l'approche de clustering . . . . .	24

<b>6</b>	<b>Objectifs du clustering</b>	<b>25</b>
6.1	Equilibrage des charges	26
6.2	Tolérance aux fautes	26
6.3	Amélioration de la connectivité et réduction des délais de transmission	27
6.4	Augmentation de la durée de vie du réseau	27
6.5	Caractéristiques d'un protocole hiérarchique	27
6.5.1	Algorithme de clustering utilisé	28
6.5.2	sélection des cluster-heads	29
6.5.3	Nature des clusters générés	29
6.5.4	Communication intra-cluster	29
6.5.5	Niveau d'agrégation de données	30
<b>7</b>	<b>Exemples de protocoles de routage hiérarchiques pour les réseaux de capteurs</b>	<b>31</b>
<b>8</b>	<b>Conclusion</b>	<b>31</b>
<b>IV</b>	<b>Algorithme harris hawks optimisation</b>	<b>34</b>
<b>1</b>	<b>Introduction</b>	<b>34</b>
<b>2</b>	<b>Harris Hawks Optimization</b>	<b>34</b>
2.1	définition	34
2.2	Composants du Harris	35
2.2.1	Exploration Phase	35
2.2.2	d'exploitation Phase	37
2.3	algorithme harris hawks optimisation code[32]	41
<b>3</b>	<b>conclusion</b>	<b>43</b>
<b>V</b>	<b>Simulation</b>	<b>45</b>
<b>1</b>	<b>Introduction</b>	<b>45</b>

<b>2</b>	<b>Algorithme proposé: HHO-LEACH Algorithme . . . .</b>	<b>45</b>
2.1	Description générale et les objectifs de l'algorithme proposé	45
2.1.1	Algorithm1:HHO-LEACH algorithm de selection efficace De CH . . . . .	47
<b>3</b>	<b>Implémentation . . . . .</b>	<b>52</b>
3.1	Choix du langage . . . . .	52
3.2	Simulation . . . . .	52
3.3	Résultats de simulation . . . . .	55
3.4	Résultats et Discussion . . . . .	57
<b>4</b>	<b>Conclusion . . . . .</b>	<b>58</b>
<b>VI</b>	<b>Conclusion</b>	<b>61</b>

## List of Figures

1	Example of capteur . . . . .	3
2	Example of capteur snas fil . . . . .	5
3	Architecture matérielle d'un capteur sans fil[5] . . . . .	6
4	réseau de capteur sans fil[7] . . . . .	8
5	Protocoles de routage à plat[7] . . . . .	9
6	Protocoles de routage hiérarchique[7] . . . . .	10
7	Pile protocolaire[10] . . . . .	11
8	Classification des RCSF selon le type de communication entre nœuds et SB . . . . .	12
9	Les classes de protocoles de routage [16] . . . . .	18
10	Structure hiérarchique d'un réseau de capteurs[18] . . . . .	24
11	Différentes phases de HHO [32] . . . . .	35
12	Exemple de vecteurs globaux en cas de siège dur[32] . . . . .	38
13	Exemple de vecteurs globaux dans le cas d'un siège dur avec des plongées rapides progressives dans l'espace 2D et 3D[32] . . . . .	41
14	ALGORITHEM DE HHO-LEACH . . . . .	49
15	ALGORITHEM DE HHO-LEACH . . . . .	49
16	ALGORITHEM DE HHO-LEACH . . . . .	50
17	ALGORITHEM DE HHO-LEACH . . . . .	50
18	ALGORITHEM DE HHO-LEACH . . . . .	51
19	Instructions . . . . .	52
20	id sensore de chaque noudes . . . . .	53
21	Sélection cluster heads (CH) . . . . .	54
22	Batteries de chaque noudes . . . . .	54
23	Sélection de cluster(rayen) . . . . .	54
24	L'énergie Consommée du ALGORITHEM LEACH et HHO . . . . .	55
25	La durée de vie du ALGORITHEM LEACH et HHO . . . . .	56
26	La durée de vie du ALGORITHEM LEACH et HHO . . . . .	57

## List of Tables

1	d'études antérieures . . . . .	31
2	Les paramètres de simulation. . . . .	53

## List of Abbreviations

**HHO:** Harzis Hawks optimisation

**RCSF:** Réseaux de Capteurs Sans Fil

**LEACH:** Low Energy Adaptive Clustering Hierarchy

**CH:** Cluster Head

**WSN:** Wireless Sensor Network

# List of Equations

1	harris hawks Exploration Phase equation . . . . .	36
2	la moyenne des positions initiales . . . . .	36
3	Transition de l'exploration à l'exploitation . . . . .	37
4	Assiège doux . . . . .	38
5	La différence de position entre la proie et la première position	38
6	saut aléatoire de la proie . . . . .	38
7	la mise à jour des positions actuelles . . . . .	38
8	la décision des faucons de se déplacer pour mettre en œuvre l'encercllement doux . . . . .	39
9	les combats de Levy (LF) . . . . .	39
10	LF . . . . .	39
11	Sigma . . . . .	39
12	le modèle mathématique pour mettre à jour les positions des faucons . . . . .	40
13	la mise à jour des positions des faucons . . . . .	40
14	la décision des faucons de se déplacer pour mettre en œuvre l'encercllement doux avec $X_m$ . . . . .	40

# Introduction générale

### Chapter I

# Introduction générale

## 1 Context general

Les progrès de la microélectronique et des technologies sans fil ont donné naissance aux réseaux de capteurs sans fil (WSN), qui consistent en de petits appareils électroniques interconnectés appelés nœuds de capteurs sans fil. Développés à l'origine pour des applications militaires, les WSN sont désormais largement utilisés dans les secteurs civils tels que la surveillance de l'environnement, l'industrie, la domotique et la santé.

Les WSN fonctionnent en groupes et collaborent pour transmettre des données à une station de base centrale. Chaque nœud de capteur est équipé d'un microprocesseur de faible puissance, d'une petite batterie, d'une antenne radio et de capteurs pour la collecte de données. Compte tenu de leurs ressources limitées, les WSN doivent fonctionner dans le cadre de ces contraintes. Ils sont souvent déployés de manière aléatoire dans des environnements difficiles et s'appuient sur la communication sans fil pour s'auto-organiser. La station de base agit comme l'interface entre le réseau et le monde extérieur, bénéficiant de plus de ressources par rapport aux nœuds capteurs. Le réseau de capteurs agit comme un pont, fournissant des mesures et des propriétés physiques réelles au système informatique.

Cependant, la petite taille des capteurs, ainsi que leur mémoire limitée, leur puissance de calcul et leur dépendance à l'égard de batteries de faible puissance, introduisent de nouveaux défis de mise en œuvre. Les chercheurs ont identifié divers défis, notamment la découverte du réseau, le routage, le contrôle du réseau, le traitement collaboratif de l'information, ainsi que la collecte et la diffusion des données.

L'efficacité énergétique est une préoccupation majeure pour les WSN, car les nœuds de capteurs reposent généralement sur de petites batteries

avec une puissance limitée. Le remplacement de ces batteries est souvent difficile, voire impossible, ce qui signifie que l'épuisement des réserves d'énergie peut entraîner une panne du réseau. La sécurité est un autre défi crucial, car les WSN gèrent des données sensibles qui pourraient être vulnérables aux attaques physiques ou aux attaques via des canaux sans fil.

Essentiellement, les WSN sont confrontés à des défis liés aux limitations des ressources, telles que l'efficacité énergétique, la découverte de réseaux, le routage, le traitement collaboratif de l'information et la collecte et la diffusion de données. Surmonter ces défis est essentiel pour la mise en œuvre et le fonctionnement réussis des réseaux de capteurs sans fil dans divers domaines d'application.

## 2 Motivations et problématique

Le routage hiérarchique de données est un sujet de recherche très important dans les réseaux de capteurs sans fil (RCSF). Il constitue un fondement essentiel pour les autres couches supérieures de ces réseaux. Comparé aux autres types de routage, tels que le routage à topologie plate et le routage géographique, le routage hiérarchique est considéré comme un outil offrant de meilleures performances en termes de consommation d'énergie. Dans le routage hiérarchique, les nœuds sont regroupés en clusters (groupes), et chaque cluster est dirigé par un Cluster Head (CH) qui gère les communications des paquets transmis par les membres du groupe ou à destination de ceux-ci. Plusieurs solutions de routage hiérarchique sont disponibles, utilisant différentes techniques pour minimiser la consommation d'énergie dans l'ensemble du réseau. Cependant, des améliorations restent nécessaires pour atteindre de meilleures performances. En résumé, le routage hiérarchique de données dans les réseaux de capteurs sans fil joue un rôle crucial dans la gestion de la consommation d'énergie. Il permet de regrouper les nœuds en clusters et de

confier la gestion des communications aux Cluster Heads, offrant ainsi de meilleures performances globales. Des recherches supplémentaires sont nécessaires pour améliorer les solutions existantes et atteindre des performances encore meilleures. notre travail vise à adapter un protocole de routage basé sur un algorithme bio-inspiré, en utilisant le comportement de broutage des faucons, pour le regroupement des nœuds dans les réseaux de capteurs sans fil. Nous proposons également une stratégie de sélection du Cluster Head efficace dès le premier tour. Ces contributions peuvent améliorer les performances énergétiques et la stabilité du réseau dans les RCSF.

### 3 Contributions et structure du mémoire

Ce mémoire est organisé comme suit:

le premier chapitre de notre travail sera consacré à la présentation des RCSF, de leurs architectures de communication et de leurs applications. Nous discuterons des facteurs et contraintes qui influencent leur conception, en mettant l'accent sur la problématique de la consommation d'énergie. Dans le deuxième chapitre, nous présenterons un état de l'art sur l'ensemble des protocoles de routage de la littérature, hiérarchique et non hiérarchique. Dans le troisième chapitre, nous présenterons l'algorithme de faucon (algorithme HHO) et son principe de fonctionnement. Le dernier chapitre est consacré à l'introduction de notre protocole de routage hiérarchique pour appelle quoi HHO-LEACH . Ce dernier est basé sur l'algorithme (HHO) de selection des CH. En réduisant la fonction objectif également dans ce chapitre nous détaillerons le simulateur que nous avons conçu pour valider notre protocole et nous discuterons des résultats obtenus en contribuant au protocole de Leach.

*Chapitre 02*

# Généralités sur les réseaux de capteurs sans fil

## Chapter II

# Généralités sur les réseaux de capteurs sans fil

## 1 Introduction

Les réseaux de capteurs sans fil fournissent des captures pour mesurer le monde physique. Ces systèmes électroniques et informatiques communiquent principalement à travers des réseaux radio, ce qui les rend autonomes en tant qu'objets communicants. Les réseaux de capteurs offrent l'opportunité de suivre les évolutions temporelles et spatiales de l'environnement physique. Les réseaux de capteurs sont utilisés dans de nombreuses applications, notamment dans les domaines de la santé, de la domotique, des transports et de la sécurité. Le succès de ces réseaux est dû à plusieurs facteurs, notamment la petite taille des nœuds qui permet un déploiement facile dans divers environnements, tels que les bâtiments, les champs et même sous l'eau. Les capteurs sont dotés d'une intelligence intégrée qui leur permet de s'auto-organiser et de démarrer le réseau sans rencontrer de problèmes significatifs. Les réseaux sans fil peuvent varier en taille, allant de quelques nœuds à des milliers de nœuds. En outre, il est possible de déployer des capteurs dans des zones où la présence humaine est dangereuse, par exemple dans des zones militaires ou des régions vulnérables aux éruptions volcaniques. Pour étudier les réseaux de capteurs, nous commençons par fournir un bref aperçu de ces réseaux, puis nous énumérons les applications potentielles et les principaux facteurs qui influencent leur conception. Nous constaterons que la communication au sein d'un réseau de capteurs suit des architectures spécifiques et des protocoles développés pour chaque couche. Dans ce chapitre, nous présentons un aperçu général des capteurs et des réseaux de capteurs sans fil.

## 2 Capteurs

Dans cette section, nous en apprenons davantage sur le capteur et ses types

### 2.1 Définition

Un capteur est un dispositif compact et autonome qui peut mesurer diverses grandeurs physiques de l'environnement qui l'entoure, telles que la température, la pression, les vibrations, etc. En général, un capteur remplit trois fonctions principales : il collecte des données, les traite, puis les transmet à une ou plusieurs stations de base (SB) pour analyse ultérieure. [1]

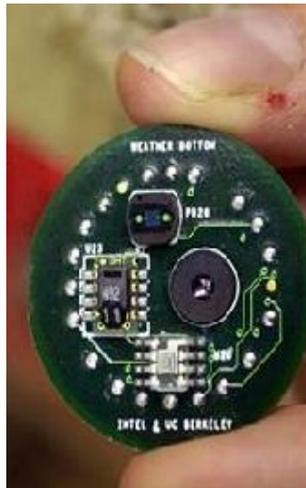


Figure 1: Example of capteur

### 2.2 Types des capteurs

La classification des capteurs se fait généralement en deux parties distinctes selon la source consultée [2]:

#### 2.2.1 Les capteurs actifs

Les capteurs actifs sans fil requièrent une source d'énergie intégrée et comprennent un circuit électronique qui traite le signal (filtrage, amplification et conversion) et un module d'émission qui leur permet de

communiquer avec le lecteur. Les données acquises par ces capteurs sont transmises instantanément dans un format approprié pour une utilisation facile et efficace. [3]. Plusieurs caractéristiques limitent l'autonomie de ce type de capteur. La distance d'interrogation et le débit d'information transmis par rapport à la quantité d'énergie consommée sont des paramètres clés dans la conception de ces systèmes.

## 2.2.2 Les capteurs passif

Un capteur sans fil est généralement considéré comme passif s'il n'est pas équipé d'un module d'émission. La plupart des capteurs passifs sont des dispositifs qui ne possèdent pas de source d'énergie embarquée, ce qui les rend facilement intégrables. L'information est transmise en interrogeant le capteur à l'aide du lecteur. Cela est réalisé en modulant la charge connectée à l'antenne du capteur. En effet, ce type de capteur est généralement constitué d'une impédance dont l'un des paramètres clés est sensible au mesurande. Pour mesurer l'impédance d'un capteur passif, il est intégré dans un circuit électrique qui le conditionne et le alimente. Ce type de capteur offre une solution de mesure passive, précise, miniature et économique pour la mesure de différentes grande.

Il existe aujourd'hui un large éventail de capteurs ayant des fonctionnalités différentes et variées. Leur conception dépend souvent de l'application à laquelle ils sont destinés, comme les capteurs aquatiques ou souterrains, par exemple. Au cours de la dernière décennie, la technologie des capteurs sans fil a considérablement progressé, avec des modules de plus en plus petits et des durées de vie plus longues. Le marché des nœuds a ainsi été ouvert à l'industrie, avec des fournisseurs tels que Cross Bow Inc., qui propose des capteurs Mica2 et MicaZ [4].

## 2.3 Composants de capteur

Nous allons en apprendre davantage sur les composants du capteur sans fil:



Figure 2: Example of capteur sans fil

### 2.3.1 Architecture matérielle

Un capteur remplit trois tâches principales : il collecte des données sur son environnement, il les traite si nécessaire, et il les transmet à d'autres nœuds. Ainsi, pour remplir ces fonctions, un nœud est composé de quatre unités de base [5].

- **Unité captage (Sensing unit):** La composition de la capteur inclut un dispositif physique de capture qui collecte les données de l'environnement local, ainsi qu'un convertisseur analogique/numérique (ADC) qui transforme les signaux analogiques en données numériques pour être traitées par l'unité de traitement.
- **Unité de traitement (Processing unit):** L'unité en question est équipée de deux interfaces, l'une pour l'unité d'acquisition et l'autre pour l'unité de transmission. Elle comprend également un processeur et une mémoire. Son rôle est de collecter les données provenant de l'unité d'acquisition, de les stocker en mémoire ou de les transmettre à l'unité de transmission.
- **Unité de communication (Transceiver unit):** Cette unité est équipée d'un module radio, qui lui permet d'émettre et de recevoir des données.
- **Unité d'énergie (Power unit) :** La gestion de l'énergie et de

# Généralités sur les réseaux de capteurs sans fil

l'alimentation de tous les composants du capteur relève de sa responsabilité. Cette tâche est généralement effectuée par une batterie, qui est souvent limitée et non remplaçable. En conséquence, l'énergie est devenue la principale contrainte pour les capteurs.

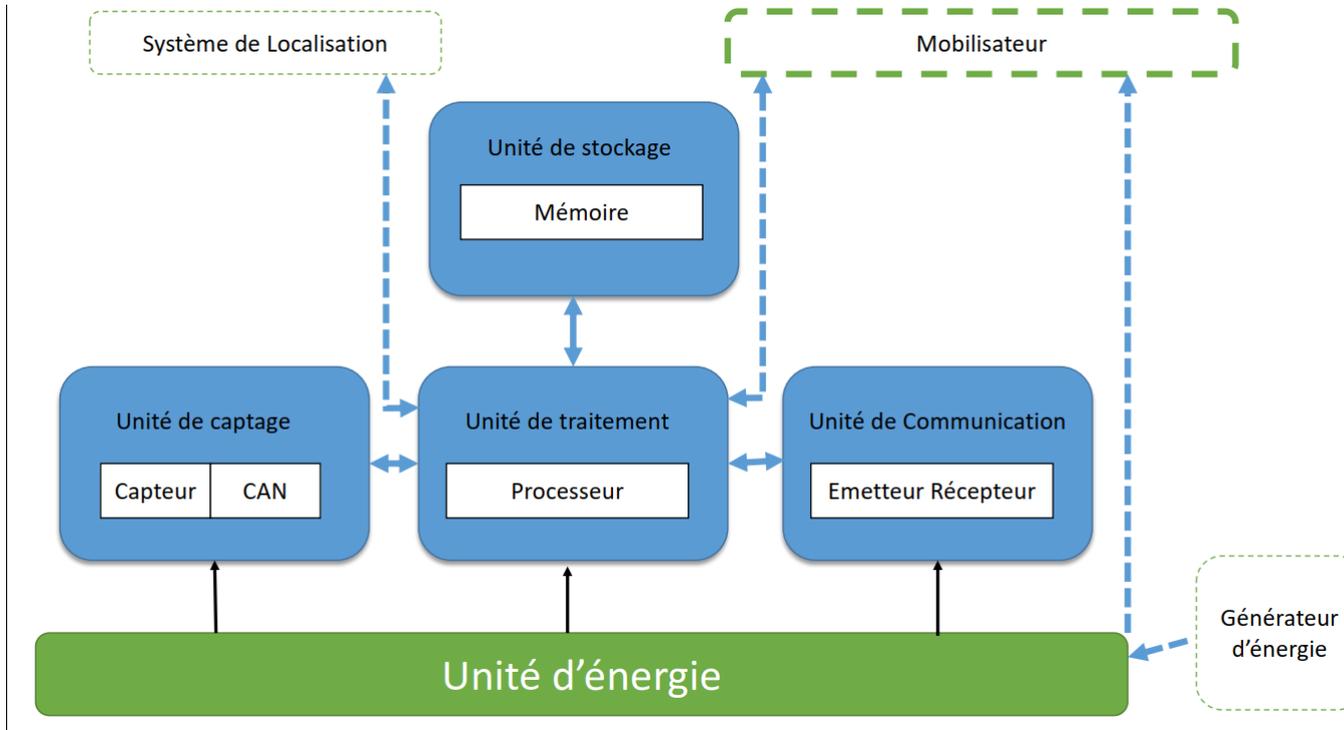


Figure 3: Architecture matérielle d'un capteur sans fil[5]

## 2.3.2 Architecture logicielle

Les capteurs sont équipés d'un mini-système d'exploitation qui gère leurs différentes opérations. Ces systèmes d'exploitation sont basés sur l'architecture des composants du capteur, ce qui facilite une implémentation et une innovation rapides, ainsi qu'un code source compact. Il existe plusieurs systèmes d'exploitation pour les réseaux de capteurs sans fil, notamment Mote, Mote2, ZigBee, SOS et TinyOS. TinyOS est l'un des plus populaires. Il est ouvert, modulaire et spécialement conçu pour les réseaux de capteurs miniatures. Ce système d'exploitation, développé par l'Université de Berkeley, est enrichi par de nombreux utilisateurs et peut rapidement intégrer les innovations liées aux applications et aux

réseaux, tout en minimisant la taille du code source pour répondre aux contraintes de mémoire des réseaux de capteurs. La bibliothèque TinyOS inclut des protocoles de réseau, des services de distribution, des pilotes de périphériques pour capteurs et des outils d'acquisition de données. Bien que la plupart des parties de TinyOS soient écrites en C, il est facile de créer des applications personnalisées en utilisant des langages tels que C, NesC et Java.[6] Réseaux de capteur sans fil

## 3 Réseaux de Capteur sans fil

Dans le premier sur le capteur et dans un deuxième temps le capteur sans fil Dans la section nous allons toucher le capteur sans fil dans le cadre des réseaux.

### 3.1 Définition

Un réseau de capteurs sans fil (RCSF) ou Wireless Sensor Network (WSN) est un système informatique constitué de petits dispositifs autonomes équipés de capteurs, qui sont répartis de manière axée ou aléatoire dans une zone d'intérêt. Ces dispositifs travaillent en coopération pour surveiller les conditions environnementales ou physiques, telles que la température, le son, les vibrations, la pression, le mouvement, etc. [7].

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

Deux architectures de réseaux de capteurs sans fil sont couramment utilisées : les réseaux de capteurs sans fil plats et les réseaux de capteurs sans fil hiérarchiques, selon .[8]

#### 3.2.1 RCSF Plat

Un RCSF plat se réfère à un réseau homogène dans lequel tous les nœuds ont des ressources similaires en termes d'énergie, de calcul et de

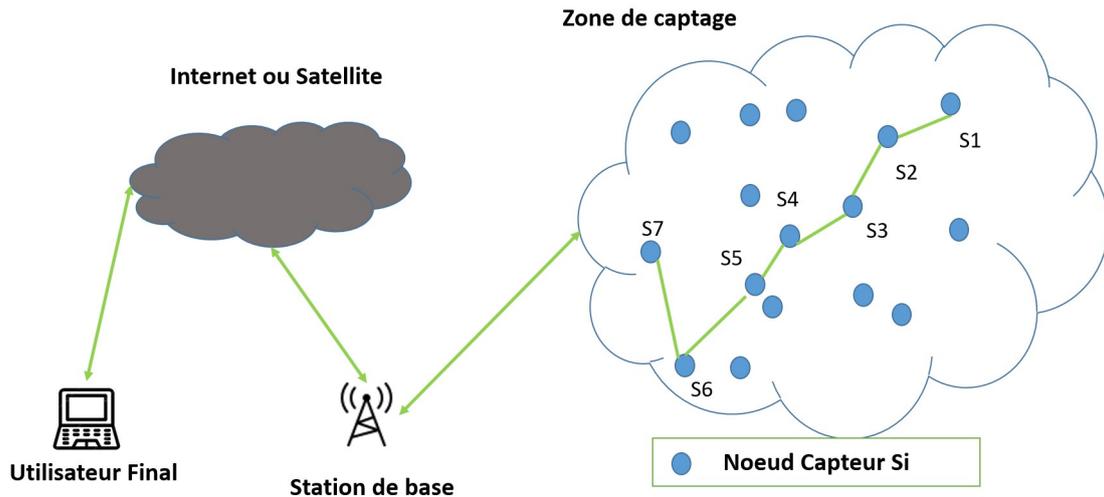


Figure 4: réseau de capteur sans fil[7]

mémoire. Cette architecture est généralement utilisée pour les réseaux avec un grand nombre de nœuds capteurs déployés par mètre carré et une communication multi-sauts sans fil où les informations doivent passer par plusieurs nœuds intermédiaires avant d'atteindre leur destination finale, sans traitement supplémentaire de la donnée transportée. Ces deux conditions rendent la mise à l'échelle très critique et la consommation d'énergie pour le routage des informations énorme.

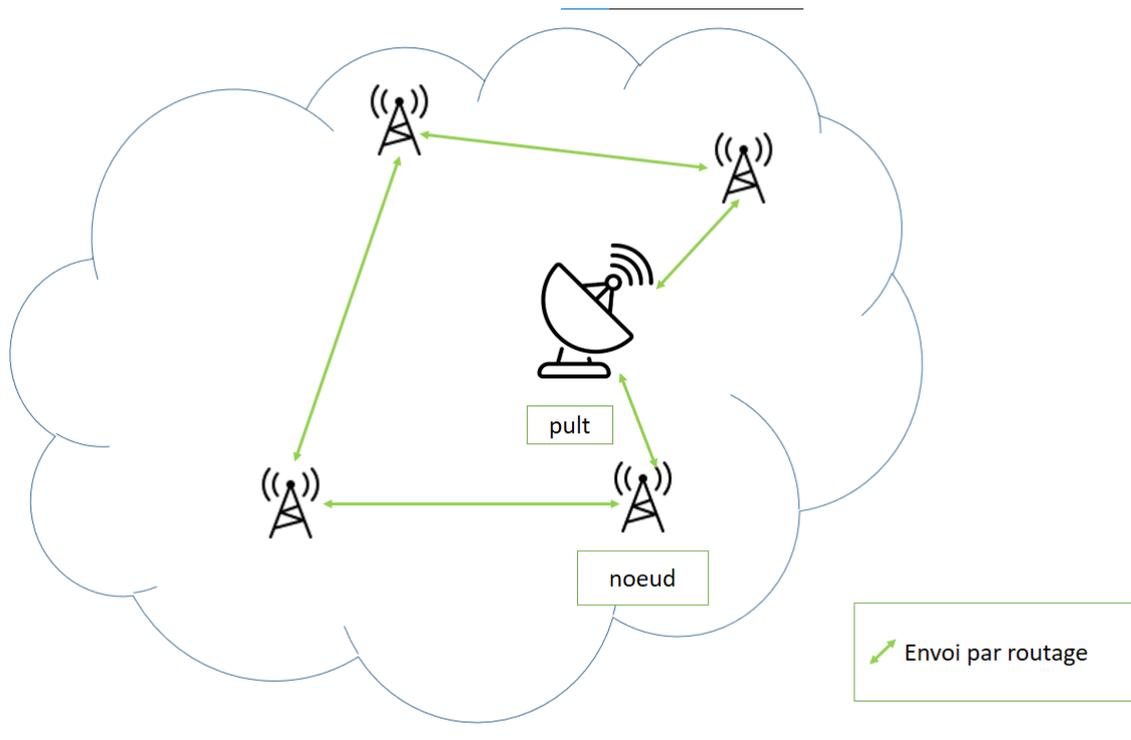


Figure 5: Protocoles de routage à plat[7]

### 3.2.2 RCSF hiérarchiques

Contrairement à l'architecture plat présentée précédemment, l'architecture hiérarchique d'un RCSF implique que tous les nœuds n'ont pas les mêmes fonctions ni les mêmes ressources. En effet, l'introduction de nœuds puissants permet de décharger la plupart des nœuds ordinaires de certaines tâches du réseau. Ainsi, les tâches complexes et nécessitantes des ressources importantes sont effectuées par les nœuds intégrés, tandis que les nœuds ordinaires s'occupent des tâches de base, comme la capture de données.

### 3.3 Caractéristiques d'un RCSF

Un réseau de capteurs sans fil possède plusieurs caractéristiques [8] dont :

- Les capteurs sont caractérisés par des ressources limitées en termes de puissance de calcul, de capacité de stockage de données et d'autonomie énergétique

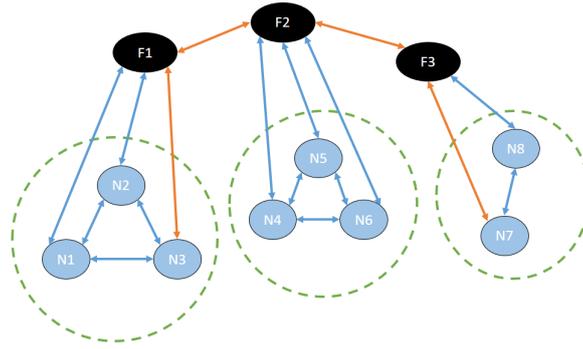


Figure 6: Protocoles de routage hiérarchique[7]

- La durée de vie des capteurs est limitée.
- Le réseau peut utiliser soit un mode de communication directe, soit un mode en multi-sauts.
- Densité importante des capteurs qui peuvent atteindre des dizaines de millions pour certaines applications.
- Il est possible de découper le réseau de capteurs en clusters et d'utiliser les capteurs comme des calculateurs ou des agrégateurs de données.
- La coopération entre les nœuds capteurs pour les tâches complexes
- deux modes de fonctionnement pour les réseaux de capteurs sans fil. Le premier est appelé "Un à plusieurs", où la station de base envoie des informations à plusieurs capteurs, et le deuxième est appelé "Plusieurs à un", où les capteurs envoient des informations à la station de base.

### 3.4 Architecture protocolaire d'un RCSF

La figure 1.7 présente la pile de protocoles utilisée par le Sink et les capteurs. Cette pile combine les protocoles de routage, de gestion de l'énergie et de données réseau pour permettre une communication efficace en termes d'énergie et encourager la coopération entre les capteurs.

# Généralités sur les réseaux de capteurs sans fil

Elle comprend une couche application, une couche transport, une couche réseau, une couche liaison de données et une couche physique, ainsi que des plans de gestion de l'énergie, de la mobilité et des tâches. Les logiciels d'application peuvent être construits pour répondre aux différents types de tâches de détection. La couche transport garantit le flux de données tandis que la couche réseau achemine les données de la couche transport. Étant donné que l'environnement est sujet aux interférences, le protocole MAC doit prendre en compte la consommation d'énergie et éviter les collisions entre les nœuds voisins. La couche physique utilise une modulation simple et robuste et des techniques de transmission et de réception. Les plans de gestion de l'énergie, de la mobilité et des tâches surveillent et gèrent la consommation d'énergie, les mouvements et la réparation des tâches entre les capteurs, ce qui aide à coordonner les tâches de détection et à réduire la consommation d'énergie globale [10].

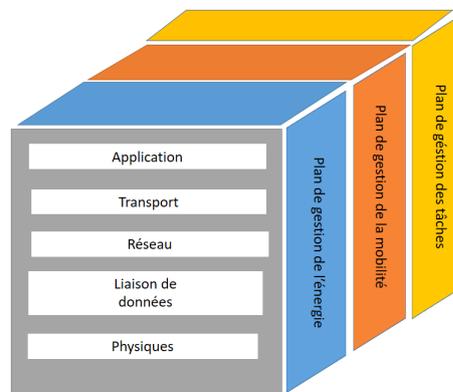


Figure 7: Pile protocolaire[10]

## 3.5 Classification des RCSF

Les réseaux de capteurs peuvent être classifiés selon plusieurs critères tels que le mode d'acquisition et de livraison des données, la distance entre les nœuds capteurs et le puits, le modèle de mobilité dans le réseau et les capacités des nœuds. Selon le type de communication entre les

nœuds et le puits, on peut distinguer deux sous-types : les réseaux multi-sauts et les réseaux à un seul saut.

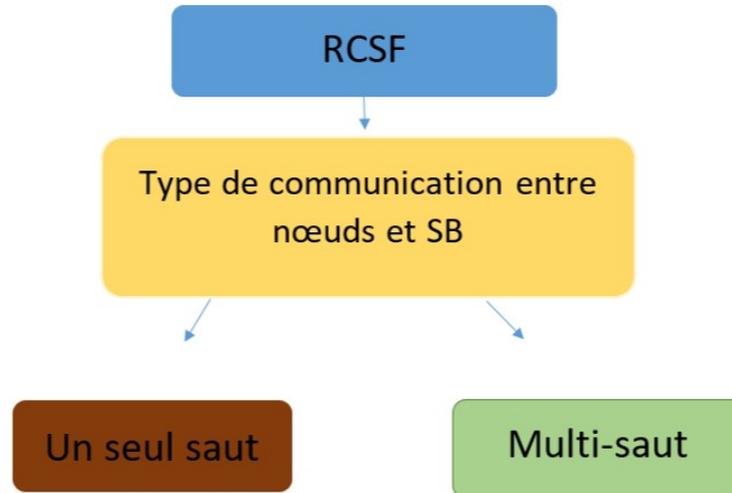


Figure 8: Classification des RCSF selon le type de communication entre nœuds et SB

Dans un réseau de capteurs à un seul saut, les nœuds capteurs sont situés à proximité immédiate du puits et envoient directement leurs données captées au puits. En revanche, dans un réseau de capteurs multi-sauts, la distance entre les nœuds capteurs et le puits dépasse leur portée maximale, ce qui nécessite le transfert des données par l'intermédiaire d'autres nœuds. Bien que les réseaux multi-sauts offrent une large gamme d'applications, ils sont plus difficiles à mettre en place que les réseaux à un seul saut.[11]

### 3.6 Domaines d'application de RCSF

Les réseaux de capteurs présentent un intérêt réel en raison de leur large gamme d'applications possibles. Les applications de ces réseaux peuvent être regroupées en cinq catégories distinctes selon une classification établie [12]

## 3.6.1 Applications militaires

Les réseaux de capteurs peuvent être considérés comme faisant partie intégrante des systèmes C4ISRT (Commande, Contrôle, Communications, Informatique, Renseignement, Surveillance, Reconnaissance et Ciblage). Dans le domaine militaire, les applications incluent :

- La surveillance du champ de bataille.
- L'examen des forces ennemies et des terrains.
- L'estimation des dégâts de la bataille.
- La détection et la reconnaissance des attaques nucléaires, biologiques et chimiques [13].

## 3.6.2 Applications médicales

les fonctions vitales de l'être humain pourraient être surveillées grâce à des micro-capteurs qui peuvent être ingérés ou implantés sous la peau. Des micro-caméras ingérables existent déjà et peuvent transmettre des images de l'intérieur du corps humain sans intervention chirurgicale. L'utilisation des réseaux de capteurs dans le domaine de la médecine pourrait permettre une surveillance continue des patients et la collecte d'informations physiologiques de meilleure qualité pour faciliter le diagnostic de certaines maladies. [14]

## 3.6.3 Applications domestiques

Avec les avancées technologiques, les nœuds capteurs intelligents et les actionneurs peuvent être intégrés dans des appareils tels que des aspirateurs, des micro-ondes ou des réfrigérateurs. Les capteurs dans ces appareils domestiques peuvent interagir les uns avec les autres et avec un réseau externe via Internet ou satellite. Cela permet à l'utilisateur final de gérer plus facilement les appareils domestiques localement ou à distance.

## 3.6.4 Applications environnementales

Les réseaux de capteurs sont adaptés pour diverses applications, telles que l'agriculture, les sites industriels, les centrales nucléaires, les plates-formes pétrolières, les forêts, et la surveillance de l'environnement marin.

## 3.6.5 Applications commerciales

Les entreprises peuvent utiliser les réseaux de capteurs pour surveiller l'ensemble du processus de production, depuis l'approvisionnement en matières premières jusqu'à la livraison du produit final[13].

## 4 Conclusion

Ce chapitre a abordé les notions générales sur les capteurs, tels que leur définition, leur structure et leurs différents types. Nous avons également présenté une vue d'ensemble des réseaux de capteurs sans fil (RCSF). Ainsi, il est clair que les réseaux de capteurs sont devenus un sujet de recherche de plus en plus important. Les nombreux problèmes rencontrés dans ce domaine suscitent l'intérêt de la recherche pour trouver des solutions adéquates.

*Chapitre 03*

# Routage dans RCSF

### Chapter III

# Routage dans RCSF

## 1 Introduction

Le principal critère influençant les protocoles de routage dans les RCSF est la minimisation de la consommation d'énergie sans compromettre l'efficacité. Dans ces réseaux, chaque nœud remplit à la fois les fonctions de source et de relais de données, ce qui signifie que la défaillance énergétique d'un nœud peut entraîner des perturbations significatives dans la topologie du réseau, nécessitant ainsi une reconfiguration coûteuse. Contrairement aux réseaux sans fil traditionnels, où les protocoles de routage sont conçus pour établir des routes entre les nœuds pour acheminer les paquets, les protocoles de routage dans les RCSF établissent des routes entre chaque nœud du réseau et la station de base pour garantir la fiabilité du routage. Dans le prochain chapitre, nous examinerons certains protocoles de routage et leur classification.

## 2 Le routage dans les RCSFs

La fonctionnalité la plus cruciale d'un RCSF est la propagation et la distribution de données dans le réseau, qui doit prendre en compte toutes les caractéristiques des capteurs pour garantir les meilleures performances du système, notamment la durée de vie, la fiabilité et le temps de réponse. Étant donné la spécificité des RCSFs, de nombreuses recherches se concentrent sur la violation de la découpe en couches protocolaires indépendantes, en introduisant la notion d'optimisation multicouches. Par exemple, en utilisant des mécanismes d'agrégation, les routeurs intermédiaires doivent accéder aux données pour établir des résumés des lectures de la région. [15].

## 2.1 Les type de routage

Dans cette section, nous allons découvrir les types routages.[15]

### 2.1.1 Protocole proactif

Un protocole proactif est un protocole qui construit les tables de routage avant qu'une demande ne soit effectuée, en identifiant en permanence la topologie du réseau.

### 2.1.2 Protocole réactif

Un protocole réactif est un protocole qui établit une table de routage seulement lorsqu'un nœud en fait la demande. Contrairement au protocole proactif, il ne connaît pas la topologie du réseau à chaque instant. Le protocole réactif détermine le chemin à prendre pour accéder à un nœud du réseau lorsque cela lui est demandé.

## 3 Protocoles de routage

Il existe trois catégories de protocoles de routage dans les réseaux de capteurs sans fil : les protocoles de routage non hiérarchiques (ou à plat), les protocoles hiérarchiques et les protocoles géographiques.

### 3.1 les protocole de routage géographique

Dans ce type de routage, les nœuds capteurs sont identifiés en fonction de leur emplacement géographique. La distance entre les nœuds voisins peut être estimée en utilisant les forces des signaux entrants. Les coordonnées relatives des nœuds voisins peuvent être obtenues en échangeant ces informations entre les voisins. En alternative, la localisation des nœuds peut être directement obtenue en communiquant avec un satellite via le système de positionnement global (GPS) [17].

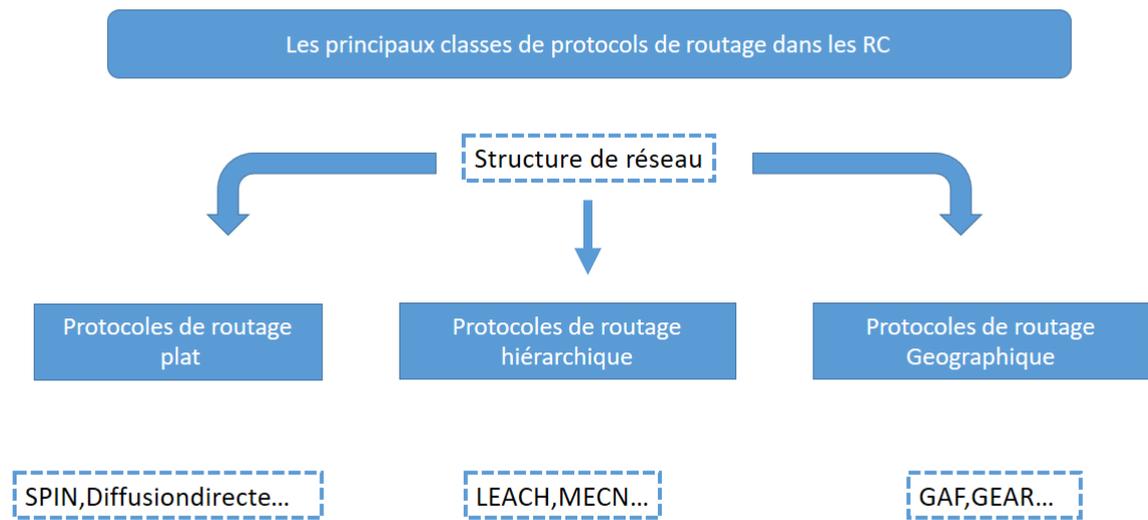


Figure 9: Les classes de protocoles de routage [16]

## 3.2 Les protocoles de routage plat

L'objectif principal des protocoles de routage dans les réseaux de capteurs sans fil est de fournir des itinéraires fiables et efficaces entre deux nœuds, permettant ainsi l'acheminement des messages. Ces protocoles permettent aux nœuds de se connecter directement entre eux et de relayer les messages via des sauts multiples. Le protocole de routage plat est un type de protocole de communication utilisé dans les réseaux de capteurs sans fil. Contrairement aux protocoles hiérarchiques qui impliquent de diviser le réseau en niveaux hiérarchiques avec des nœuds spéciaux tels que des nœuds de cluster ou des nœuds maîtres, le protocole plat n'a pas de structure hiérarchique.

Dans un protocole statique, tous les nœuds du réseau sont considérés comme égaux et ont la même responsabilité dans le processus de routage des données. Chaque nœud est capable de prendre des décisions de routage et de transmettre des paquets de données à d'autres nœuds du réseau. [16]

### 3.3 Protocoles de routage hiérarchiques

Lorsque le nombre de nœuds dans un réseau devient important, sa gestion devient plus complexe. Les protocoles de routage plats fonctionnent bien lorsque le nombre de nœuds est limité. Pour économiser l'énergie dans chaque nœud et ainsi prolonger la durée de vie du système, la structuration du réseau est un outil clé. La hiérarchie est l'une des structures les plus connues et permet de partitionner le réseau en sous-ensembles pour faciliter sa gestion, notamment le routage, qui peut se faire à plusieurs niveaux. Les protocoles hiérarchiques permettent une vue locale du réseau et des nœuds spéciaux peuvent avoir des rôles supplémentaires. La littérature présente plusieurs contributions en matière de techniques de hiérarchisation de réseau, que nous avons classées dans l'état de l'art ci-dessous pour faciliter leur comparaison.[16]

## 4 Approche basée sur le clustering

Les réseaux de capteurs sont soumis à de fortes limitations en termes de ressources énergétiques. Actuellement, une grande partie des travaux de recherche porte sur l'allongement de la durée de vie de ces réseaux en optimisant la consommation d'énergie des capteurs. Pour ce faire, l'agrégation a été proposée comme une approche de routage qui non seulement réduit la consommation d'énergie, mais réduit également le nombre de messages échangés dans le réseau et optimise l'utilisation de la bande passante.

Cette section est organisée comme suit : Dans un premier temps, nous allons définir tous les concepts liés au mécanisme d'assemblage. Ensuite, nous présenterons le processus d'agrégation et passerons en revue les principaux avantages d'un protocole de routage hiérarchique basé sur ce mécanisme. Ensuite, nous aborderons les différentes caractéristiques de ce protocole de routage hiérarchique. Nous présenterons également un ensemble de solutions hiérarchisées proposées dans la littérature, et

expliquerons leurs caractéristiques et étapes de construction.

## 5 Notions de base

La façon dont les nœuds capteurs sont déployés dans un réseau a un impact significatif sur l'application en question. Le déploiement du réseau peut être effectué de manière aléatoire et non uniforme ou être réalisé manuellement sur le terrain de capture [31]. Lorsqu'il s'agit d'une topologie non uniforme, l'utilisation du clustering se révèle être une méthode efficace permettant de gérer plus efficacement le trafic dans le réseau et de réduire le nombre de messages échangés entre les nœuds.

Il est essentiel de choisir un bon schéma de déploiement des nœuds capteurs afin de garantir une couverture adéquate et une collecte d'informations efficace. L'approche du clustering offre des avantages significatifs en termes de gestion du trafic, de conservation de l'énergie et de prolongation de la durée de vie du réseau. En regroupant les nœuds capteurs en clusters, il est possible de réduire les redondances dans les transmissions de données et d'optimiser l'utilisation des ressources disponibles.

Il convient de noter que la sélection d'un schéma de déploiement et d'un protocole de clustering appropriés dépend des spécificités de l'application et des contraintes du réseau de capteurs. Une analyse approfondie des besoins et des objectifs de l'application est nécessaire pour choisir la meilleure approche de déploiement et de clustering.

### 5.1 Définition

En général, les réseaux de capteurs sans fil (RCSF) sont déployés de manière très dense dans des environnements difficiles où un déploiement manuel du réseau n'est pas envisageable. Cela soulève la nécessité de trouver un schéma d'organisation logique du réseau. Parmi les approches proposées à cet effet, le clustering a été présenté comme une méthode performante pour améliorer l'évolutivité du réseau et prolonger sa durée

de vie [18].

Les algorithmes de clustering permettent de diviser le réseau en groupes de nœuds appelés clusters, qui ont pour rôle de relayer efficacement les données captées vers la station de base [19, 18, 20, 21]. Chaque cluster est représenté par un CH, également appelé cluster-head. Chaque nœud au sein d'un cluster envoie ses données vers le CH. Ce dernier a pour responsabilité de relayer les données reçues des membres de son cluster vers la station de base, soit directement, soit en passant par des CHs intermédiaires.

Le rôle d'un CH ne se limite pas au relais de données, il effectue également des traitements sur ces données, tels que l'agrégation des paquets reçus des nœuds du cluster. Cette opération permet de réduire significativement le nombre de paquets transmis vers la station de base et de diminuer les interactions entre les CH. Ce mécanisme permet d'économiser l'énergie des nœuds et de prolonger la durée de vie du réseau. En effet, les nœuds effectuent des transmissions sur de courtes distances vers leur CH, et le nombre de paquets redondants est réduit, ce qui optimise l'utilisation du support de transmission.

Il convient de noter que le choix d'un algorithme de clustering approprié dépend des caractéristiques spécifiques du réseau de capteurs sans fil et des exigences de l'application. Une sélection soignée de l'algorithme de clustering et des CHs est essentielle pour garantir un fonctionnement efficace du réseau et une utilisation optimale des ressources disponibles.

### 5.1.1 Nœud membre

Dans un réseau de capteurs sans fil (RCSF), le nœud capteur est l'élément central et joue un rôle crucial. Les nœuds capteurs peuvent assumer diverses fonctions telles que la collecte des données environnementales, le stockage des données, le routage des informations ou encore le traitement des données captées. Le rôle précis d'un capteur dépend de la façon dont il est affecté par le protocole de clustering.

### 5.1.2 cluster

Les réseaux de capteurs sans fil (RCSFs) sont organisés en clusters, qui sont des unités d'organisation essentielles. Étant donné le déploiement dense de ces réseaux, il est nécessaire de diviser les nœuds en groupes afin de faciliter les communications et de répondre aux diverses contraintes.

### 5.1.3 Cluster head

Les cluster-heads (CH) sont les représentants des clusters. Ils jouent un rôle essentiel dans la coordination des nœuds au sein de leur cluster, l'agrégation des données et les communications avec d'autres cluster-heads [22]. Chaque cluster est dirigé par un seul CH, avec éventuellement un CH de secours prévu pour prendre le relais en cas de défaillance du CH principal.

### 5.1.4 Station de base

La station de base (SB) est équipée de ressources énergétiques, de capacités de traitement et de stockage considérables. Elle occupe le premier niveau dans la hiérarchie d'un RCSF. Son rôle principal est d'établir une liaison entre le réseau de capteurs et l'utilisateur final. Les données sont traitées et stockées au niveau de la SB en fonction du type d'application et des exigences de l'utilisateur final.

### 5.1.5 Utilisateur final

Les données générées par un réseau de capteurs sont utilisées par une large gamme d'applications [23]. Ces données sont accessibles via internet en utilisant un ordinateur bureau. L'utilisateur envoie des requêtes qui seront diffusées dans le réseau afin de recueillir les données d'intérêt.

## 5.2 Processus de clustering

L'efficacité d'un algorithme de clustering est évaluée en termes du nombre de clusters formés et de leur stabilité face aux changements de la topologie du réseau. Les méthodes de formation de clusters sont généralement exécutées en trois phases:

### 5.2.1 Phase d'élection des cluster-heads:

La première étape d'un algorithme de clustering consiste à élire les cluster-heads (CH). Le CH joue le rôle de coordinateur local ou de nœud central au sein de son cluster. Il assume diverses tâches de coordination, telles que la collecte des données des nœuds membres de son cluster, l'agrégation des données et leur transmission vers le sink. La division des nœuds du réseau en clusters dépend du nombre et de l'emplacement des CH souhaités. Par conséquent, la sélection du CH revêt une importance capitale pour le reste du processus de clustering. Cette phase détermine les performances, la durée de vie et l'efficacité énergétique du réseau[24]. Différentes métriques peuvent être utilisées pour élire les CH, telles que le plus grand/petit ID dans le voisinage, le degré de connectivité, la puissance de transmission, l'énergie résiduelle, etc., ou une combinaison de ces critères. De nombreux travaux de recherche se sont concentrés sur l'optimisation de la sélection des CH.

### 5.2.2 Communication intra-cluster et inter-cluster :

Le cluster-head (CH) assume la responsabilité des communications au sein de son cluster et de la gestion des informations de routage nécessaires pour établir une connexion avec la station de base. Pour atteindre la station de base, les cluster-heads ont deux options : ils peuvent communiquer directement avec celle-ci, ou bien utiliser une communication multi-saut en passant par d'autres cluster-heads qui agissent en tant que nœuds relais.

## 5.2.3 Maintenance des clusters

Dans le but de s'adapter aux changements de la topologie du réseau, une mise à jour des clusters est dynamiquement réalisée pour reconnaître les nouveaux nœuds ajoutés ou ceux qui ont disparus (les nœuds qui ont épuisé leurs énergies). D'autre part, un CH consomme beaucoup plus d'énergie comparé aux nœuds ordinaires, du moment qu'il assure des tâches plus complexes. Si le CH garde son statut durant toute la vie du réseau, même s'il ne possède pas par exemple le poids maximum dans son propre cluster alors, il perdra son rôle une fois sa batterie sera épuisée et ceci conduira à la déconnexion de son cluster du reste du réseau [25]. Dans le but d'éviter de telles situations, le processus de clustering est généralement exécuté enround pour alterner le rôle de CH entre les nœuds et balancer la consommation de l'énergie dans le réseau.

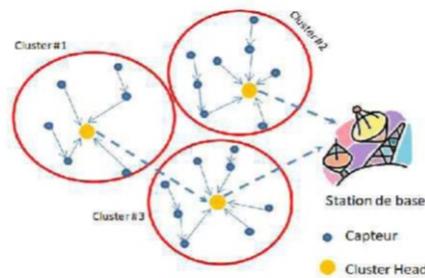


Figure 10: Structure hiérarchique d'un réseau de capteurs[18]

## 5.3 Avantages de l'approche de clustering

Récemment, de nombreux algorithmes de clustering ont été proposés dans le but principal d'optimiser l'utilisation des ressources des réseaux de capteurs. L'approche du clustering permet de[18][24]:

- La localisation des routes à l'intérieur des clusters réduit la taille des tables de routage des nœuds, simplifiant ainsi la gestion des routes et économisant les ressources.

- Permet d'augmenter considérablement la scalabilité du réseau sans dégrader ses performances.
- En limitant la portée des communications inter-clusters et en réduisant le nombre de messages redondants échangés entre les nœuds, le clustering permet une utilisation optimisée de la bande passante.
- CH peut ordonnancer les tâches des nœuds à l'intérieur de son cluster, ainsi, les nœuds peuvent passer en mode sleep durant les périodes d'inactivité ce qui permet de réduire la consommation de l'énergie. Les transmissions des nœuds vers le CH peuvent aussi être ordonnancées afin d'éviter les collisions sur le canal de transmission.
- En outre, les CH agrègent les données captées par les nœuds membres de son cluster ce qui permet de diminuer le nombre de paquets relayés vers la station de base.
- Stabiliser la topologie du réseau au niveau des nœuds de manière à ce que chaque nœud se soucie seulement du cluster à joindre et ne sera pas affecté par les changements au niveau supérieur. Ceci réduit la charge de la maintenance de la topologie.

## 6 Objectifs du clustering

Les objectifs des algorithmes de clustering varient selon les besoins des applications. Par exemple, si une application est sensible à la teneur des données, à la durée de vie des nœuds ou autre, la connectivité intra cluster et inter cluster ainsi que la longueur des chemins sont des facteurs à considérer lors de l'élection des CHs et la formation des clusters []. Nous allons exposer, dans cette section, les objectifs souvent visés par le processus de clustering.

## 6.1 Equilibrage des charges

L'équilibre de la répartition des capteurs entre les clusters est souvent un objectif recherché dans les configurations où les cluster-heads (CHs) sont choisis par les capteurs eux-mêmes pour assumer des fonctions de traitement de données et de gestion intra-cluster. Cependant, l'équilibre de la taille des clusters revêt une importance cruciale pour prolonger la durée de vie du réseau. Il est primordial d'éviter l'épuisement prématuré de l'énergie dans certains CHs qui gèrent un grand nombre de nœuds, ce qui pourrait entraîner leur dysfonctionnement. Ainsi, il est essentiel d'obtenir un nombre équivalent de nœuds dans chaque cluster afin que les données agrégées soient prêtes simultanément pour un traitement ultérieur, que ce soit au niveau de la station de base ou à un niveau supérieur dans la hiérarchie du réseau.

## 6.2 Tolérance aux fautes

Dans de nombreuses applications, les réseaux de capteurs sont déployés dans des environnements hostiles où les nœuds sont exposés à des risques constants de dysfonctionnement dus à des dommages physiques. Dans de tels scénarios, il est essentiel de tolérer les défaillances des cluster-heads (CHs) afin d'éviter la perte de données critiques. La méthode traditionnelle pour pallier la défaillance des CHs consiste à re-clustering du réseau. Cependant, cette approche peut perturber les opérations en cours. Par conséquent, de nouvelles techniques de tolérance aux pannes sont préférables, telles que l'assignation de CHs de secours (backup CH) et la rotation du rôle de CH entre les nœuds. Dans la première méthode, les backup CHs peuvent être réaffectés pendant le fonctionnement normal du réseau. La deuxième technique, en plus de ses avantages en termes d'équilibrage de charge, offre également une solution de tolérance aux pannes.

### 6.3 Amélioration de la connectivité et réduction des délais de transmission

La connectivité inter-cluster est essentielle pour de nombreuses applications. Lorsqu'un cluster-head (CH) est un nœud ordinaire, l'objectif de la connectivité consiste à assurer la disponibilité des chemins entre chaque CH et la station de base, ou à réduire la longueur des chemins. Lorsque la latence des données est critique pour l'application, la connectivité intra-cluster devient une recommandation à prendre en compte lors de la conception des clusters. Afin d'optimiser la latence des données, il est nécessaire de réduire le nombre de sauts nécessaires pour atteindre la destination.

### 6.4 Augmentation de la durée de vie du réseau

Étant donné que les capteurs disposent de ressources énergétiques limitées, la durée de vie du réseau représente un défi majeur, en particulier pour les applications déployées dans des environnements difficiles. Lorsque les cluster-heads (CHs) disposent de ressources plus importantes, il est crucial de minimiser la consommation d'énergie lors des communications intra-cluster en les positionnant au centre du cluster, à proximité des nœuds membres. Si les CHs sont des nœuds ordinaires, leur durée de vie peut être prolongée en équilibrant les charges. De plus, le clustering adaptatif et la configuration des routes peuvent également être pris en compte pour étendre la durée de vie du réseau[27][26].

### 6.5 Caractéristiques d'un protocole hiérarchique

Un protocole de routage hiérarchique doit spécifier plusieurs tâches que nous pouvons classer selon :

### 6.5.1 Algorithme de clustering utilisé

Du fait de la variété des objectifs qui sont visés, il existe plusieurs types d'algorithmes de clustering. Selon [28] Ces derniers peuvent être classés en 3 types : centralisé, distribué et géographique.

1. **Centralisé:** Dans les solutions d'ordonnement centralisées, il existe une entité centrale qui détient toutes les informations sur l'état du réseau. Cette entité est chargée de calculer en temps réel le chemin optimal entre deux nœuds. Ainsi, tout nœud source souhaitant établir une connexion doit contacter cette entité, ce qui entraîne un temps supplémentaire pour le calcul d'une route. De plus, il existe un problème de fiabilité important. En effet, si cette entité de routage venait à être hors service, ou si l'un des liens la reliant au reste du réseau était coupé, cela aurait un impact sur le bon fonctionnement du réseau.
2. **Distribué:** La technique du clustering réparti est une méthode locale dans laquelle les stations voisines échangent des messages pour partager des informations sur l'état du trafic et du réseau, afin de synchroniser les tâches de contrôle. Cette approche permet de minimiser la communication nécessaire pour maintenir une vue globale du réseau, avec un certain délai. Les travaux de recherche se tournent vers cette méthode, car avec l'avènement des réseaux haut débit, il est crucial de réagir rapidement aux changements de la topologie du réseau. Cette méthode reste fiable, et un dysfonctionnement d'un nœud n'affecte pas le bon fonctionnement du système dans son ensemble [28][29].
3. **Géographique:** L'algorithme de la tâche de contrôle à exécuter par un nœud dépend de sa position géographique

### 6.5.2 sélection des cluster-heads

La rotation des CHs est un facteur important dans l'organisation des réseaux de capteurs. Étant donné que la station de base (BS) est généralement éloignée des capteurs, les CHs consomment plus d'énergie pour transmettre les données à la BS. Par conséquent, si un même nœud fonctionne continuellement en tant que CH, il épuisera rapidement sa batterie. Afin d'éviter cela et de préserver la puissance de la batterie des capteurs individuels, les algorithmes de clustering étudiés jusqu'à présent intègrent la rotation des CHs parmi les capteurs. Cela permet de distribuer équitablement la charge de travail et d'optimiser la durée de vie du réseau.

### 6.5.3 Nature des clusters générés

Les algorithmes de clustering peuvent générer deux types de clusters : des clusters disjoints et des clusters interconnectés (overlapped). Dans le premier type, un nœud ne peut appartenir qu'à un seul cluster à la fois, ce qui est le cas le plus courant. Cependant, dans certaines applications spécifiques telles que le routage inter-cluster, des clusters interconnectés sont utilisés. Ce type de clustering permet aux nœuds d'appartenir à un ou plusieurs clusters simultanément. Cela offre une flexibilité supplémentaire dans la gestion des communications et des interactions entre les nœuds, en permettant des relations multiples et une connectivité accrue entre les clusters.

### 6.5.4 Communication intra-cluster

La communication entre le Cluster Head (CH) et les autres membres du cluster peut se faire soit en une seule étape, soit en plusieurs étapes. Dans le cas d'une communication directe en une seule étape, les paquets de données sont envoyés directement au CH. Cependant, cela nécessite que les membres du cluster soient capables d'atteindre le CH en utilisant une transmission suffisamment puissante pour une réception de données

fiable, ce qui entraîne une consommation d'énergie élevée si la distance entre le CH et les nœuds est importante.

Afin de réduire la consommation d'énergie, une communication en plusieurs étapes sur de courtes distances est utilisée. Chaque membre du cluster envoie alors ses données au membre le plus proche de son cluster, qui les relaie ensuite au membre suivant, et ainsi de suite, jusqu'à ce que les données parviennent finalement au CH. Les communications de type Unicast sont souvent utilisées pour réduire les collisions et assurer un transfert efficace des données.

Au niveau du protocole MAC (Medium Access Control), différentes techniques sont utilisées pour garantir un accès équitable et sans erreurs. Par exemple, le CDMA (Code Division Multiple Access) utilise des codes spécifiques pour chaque cluster, le TDMA (Time Division Multiple Access) attribue à chaque nœud du cluster un intervalle de temps spécifique pour l'envoi de ses données, et le FDMA (Frequency Division Multiple Access) attribue une fréquence spécifique à chaque nœud pour la transmission des données. Toutes ces techniques contribuent à optimiser les performances de communication et à réduire la consommation d'énergie.

### 6.5.5 Niveau d'agrégation de données

L'agrégation de données est une technique utilisée pour optimiser la consommation d'énergie lors du transfert de données dans les réseaux de capteurs. Étant donné que les nœuds capteurs peuvent produire à la fois des données significatives et des données redondantes, l'agrégation de données permet de regrouper les paquets provenant de différentes sources en utilisant une fonction d'agrégation spécifique, telle que la suppression, le minimum, le maximum ou la moyenne .

Cette approche permet de réduire le nombre de transmissions nécessaires, car les données agrégées sont envoyées au lieu des données individuelles. Cela réduit la charge de communication et la consommation d'énergie globale du réseau. L'agrégation de données est largement utilisée dans de

nombreux protocoles de routage afin d'optimiser l'efficacité du transfert de données et d'améliorer les performances du réseau de capteurs.[30]

## 7 Exemples de protocoles de routage hiérarchiques pour les réseaux de capteurs

Dans le tableau suivant, nous allons découvrir un groupe de protocoles de routage hiérarchique et leurs caractéristiques, selon des études antérieures.

/	Famille (Plat ou Hiérarchique)	Mobilité	Agrégation des donne	Scalability	consommation énergie	Qualité de service	Chois cluster head	Multi-chemin	Complexité
FIZZY LEACH HEED	Hiérarchique	-Possible -Station de base fixe	oui	Bonne	Max	Non		Non	CHs
PEGASIS SENSOR MAC	Hiérarchique	-Possible -Station de base fixe	Non	Bonne	Max	Non		Non	Faible
VHG- LEACH VSG-LEACH	Hiérarchique	-Possible -Station de base fixe	oui	Bonne	Max	Non		Non	CHs
RIBeep	Hiérarchique	-Possible -Station de base fixe	oui	Bonne	Max	Non		Non	CHs
(GWO) (E-LEACH)	Hiérarchique	-Possible -Station de base fixe	oui	Bonne	Max	Non		Non	CHs

Table 1: d'études antérieures

## 8 Conclusion

Au fil des années, l'économie d'énergie dans les réseaux de capteurs sans fil est devenue l'un des domaines de recherche les plus importants. L'objectif principal est de maintenir les capteurs en fonctionnement pendant une longue période et de maximiser la durée de vie du réseau. Les travaux de recherche se concentrent principalement sur les protocoles de routage économes en énergie basés sur des techniques

de clustering efficaces. Dans ce chapitre, nous avons exposé les principales caractéristiques des approches les plus importantes proposées dans la littérature. Le clustering a été considéré comme l'approche la plus efficace pour soutenir l'évolutivité des réseaux de capteurs sans fil. Cette technique facilite l'agrégation des données et réduit généralement les communications, ce qui se traduit par une consommation d'énergie réduite. Nous avons commencé par une étude approfondie du clustering dans la littérature, en présentant les concepts clés de ces techniques. Ensuite, nous avons réalisé une analyse détaillée du fonctionnement de plusieurs algorithmes de clustering spécifiquement conçus pour les réseaux de capteurs sans fil. Nous avons comparé ces différents algorithmes selon plusieurs critères tels que la mobilité, l'évolutivité, l'efficacité énergétique, l'agrégation de données, les besoins en informations, la qualité de service, le modèle de livraison des données, le multithreading et la complexité. Cette étude nous a permis de comprendre le fonctionnement des algorithmes proposés et de prendre en compte leurs limitations.

*Chapter 03*

**Algorithme harris  
hawks  
optimisation**

## Chapter IV

# Algorithme harris hawks optimisation

## 1 Introduction

Les réseaux de capteurs sans fil ont un nombre énorme d'algorithmes d'clustering . Le but de ces algorithmes est de trouver des solutions aux problèmes d'exploitation optimaux dans différents domaines Dans ce chapitre, nous aborderons Algorithme des harris hawks optimisation et son principe, ses caractéristiques et ensuite divers domaines d'application.

## 2 Harris Hawks Optimization

Nous découvrirons l'algorithme de Harris Hawks et ses étapes

### 2.1 définition

Harris Hawks Optimization (HHO) est un algorithme d'optimisation inspiré de la nature basé sur le comportement de chasse des faucons de Harris, une espèce de rapaces. Il imite la hiérarchie sociale et les stratégies de chasse coopérative de ces oiseaux pour résoudre des problèmes d'optimisation.

L'algorithme HHO commence par une population initiale de solutions candidates, représentée comme un ensemble d'individus. Ces individus sont considérés comme des "faucons" dans l'algorithme. Chaque faucon représente une solution potentielle au problème d'optimisation.

L'algorithme met à jour itérativement les positions des faucons à la recherche de meilleures solutions. Il emploie différents opérateurs inspirés du comportement des faucons, tels que l'exploration, l'exploitation et la fuite. Ces opérateurs permettent aux faucons d'explorer l'espace de

# Algorithme harris hawks optimisation

recherche, d'exploiter des régions prometteuses et d'échapper aux optima locaux.

Pendant le processus d'optimisation, les faucons communiquent et coopèrent entre eux pour améliorer les performances globales. Ils échangent des informations, partagent des connaissances et apprennent de leurs expériences pour orienter la recherche vers des solutions optimales.

L'objectif principal de Harris Hawks Optimization est de trouver la meilleure solution qui optimise une fonction de fitness donnée. Cela peut être appliqué à divers problèmes d'optimisation, y compris la conception technique, la planification, l'exploration de données et bien d'autres. [32]

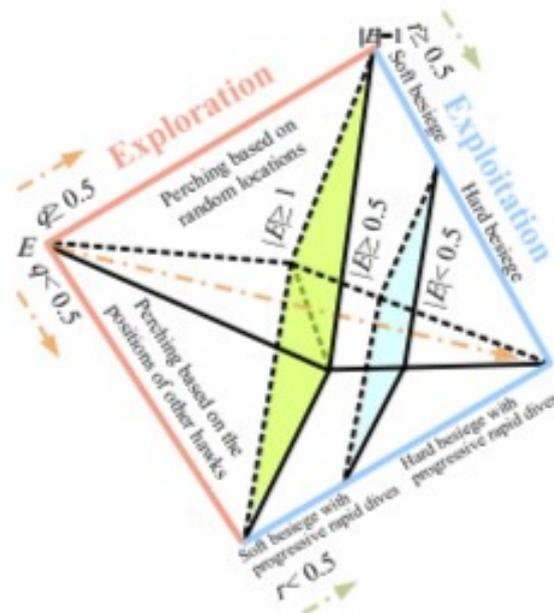


Figure 11: Différentes phases de HHO [32]

## 2.2 Composants du Harris

Nous allons en apprendre davantage sur les composants et les étapes de l'algorithme:[32]

### 2.2.1 Exploration Phase

Il s'agit de la phase initiale du processus de chasse, qui englobe l'observation, le suivi et la localisation de la proie. Dans le cadre de

l'algorithme HHO, cette étape est qualifiée de mécanisme exploratoire. Dans la nature, les faucons de Harris peuvent consacrer plusieurs heures à la recherche de proies. Ainsi, la probabilité de repérer la proie (c'est-à-dire la cible) dépend des faucons de Harris (c'est-à-dire des solutions évidentes). Par conséquent, la meilleure solution candidate est celle qui se rapproche le plus de la proie. Les faucons de Harris adoptent deux stratégies lorsqu'ils attendent leur proie : soit ils se positionnent à proximité des membres de leur famille pour attaquer en groupe, soit ils choisissent des positions aléatoires comme des arbres élevés. Les deux cas sont formulés dans l'équation suivante.

$$X(t+1) = \begin{cases} X_{\text{mand}}(t) - r_2 |X_{\text{rand}}(t) - 2r_3 X(t)| & q \geq 0.5 \\ (X_{p\dots y}(t) - X_m(t)) - r_4 (LB + r_5 (UB - LB)) & q < 0.5 \end{cases} \quad (1)$$

où  $X(t+1)$  représente les nouvelles positions des faucons à l'itération  $t$ ,  $X_{\text{proie}}$  correspond à l'emplacement de la proie,  $X_m$  est la moyenne des positions initiales de la population,  $X_{\text{rand}}$  désigne un faucon sélectionné aléatoirement dans l'espace de recherche,  $X(t)$  indique les emplacements initiaux des faucons, qui sont calculés selon l'équation 2. Les variables  $r_2$ ,  $r_3$ ,  $r_4$ ,  $r_5$  et  $q$  sont des valeurs aléatoires comprises entre 0 et 1. Il est important de noter que ces nombres aléatoires sont mis à jour à chaque itération  $t$ .

$$X_m(t) = \frac{1}{N} \sum_{i=1}^N X_i(t) \quad (2)$$

où  $X_i$  représente la position du faucon à l'itération  $t$  et  $N$  est le nombre total de faucons.

1. **Transition de l'exploration à l'exploitation:** L'algorithme HHO est capable de passer de la phase d'exploration à celle d'exploitation, en adaptant ses comportements d'exploitation en fonction de l'énergie d'évasion de la proie. Pendant le comportement d'évasion, l'énergie de la proie diminue de manière significative. Pour prendre en compte

cette diminution, l'énergie d'une proie est modélisée de la manière suivante :

$$E = 2E_0\left(1 - \frac{t}{T}\right) \quad (3)$$

## 2.2.2 d'exploitation Phase

Après avoir découvert la proie lors de l'exploration (c'est-à-dire la recherche étendue), cette phase marque le début de l'exploitation. Les faucons de Harris tentent alors de fondre sur leur proie de manière soudaine. En revanche, la proie tente de s'échapper, ce qui est communément appelé les sept évasions [30]. L'optimisation des faucons de Harris a proposé quatre approches potentielles pour modéliser les stratégies de chasse et les comportements d'évasion. Un nombre aléatoire, noté  $r$ , est utilisé pour représenter la probabilité de succès de la proie dans le combat ( $r < 0,5$ ) ou l'échec ( $r \geq 0,5$ ). De plus, les faucons de Harris utilisent soit un blocage souple, soit un blocage dur pour capturer la proie, en fonction de la puissance de la proie  $E$ . Par exemple, si le blocage est souple, la condition sera  $|E| \geq 0,5$ , sinon  $|E| < 0,5$ .

1. **Assiège doux:** Comme mentionné précédemment, lorsque les valeurs de  $|E|$  sont supérieures ou égales à 0,5 avec  $r$  supérieur ou égal à 0,5, cela signifie que la proie a suffisamment d'énergie pour se défendre contre les faucons de Harris en suivant des chemins aléatoires et en effectuant des sauts trompeurs. Malheureusement, la proie échouera car les faucons de Harris épuisent son énergie en l'encerclant, puis en lançant une attaque surprise. L'équation 4 montre la modélisation de ce comportement.

$$X(t+1) = \Delta X - E |JX_{prey}(t) - X(t)| \quad (4)$$

$$\Delta X(t) = X_{\text{prey}}(t) - X(t) \quad (5)$$

$$J = 2 \times (1 - r_6) \quad (6)$$

Dans cette phase,  $\Delta X$  représente la différence de position entre les proies et leur position initiale dans la répétition  $t$ . La valeur  $r_6$  est choisie aléatoirement dans l'intervalle  $(0,1)$ . Quant à  $J$ , il fait référence au saut aléatoire de la proie, et sa valeur change aléatoirement pour imiter la nature des mouvements de la proie.

2. **Assiéger dur:** Dans ce cas,  $t < 0,5$  et  $r \geq 0,5$ . Ainsi, la proie n'a pas assez d'énergie pour s'échapper. De plus, les faucons de Harris sont prêts à encercler la proie et à effectuer une attaque surprise, difficilement. L'équation 7 illustre la mise à jour des positions actuelles dans cette situation.[33]

$$X(t+1) = X_{\text{prey}}(t) - E|\Delta X(t)| \quad (7)$$

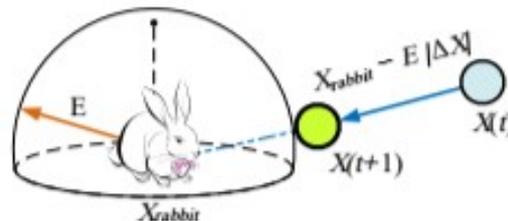


Figure 12: Exemple de vecteurs globaux en cas de siège dur[32]

3. **Siège souple avec plongeurs rapides progressifs:** Ce cas est plus compliqué que les cas précédents, car il est utilisé lorsque

## Algorithme harris hawks optimisation

---

$|E| \geq 0.5$  et  $r < 0,5$ . Ainsi, la proie a suffisamment de puissance pour s'échapper avec succès. D'autre part, les faucons de Harris continuent d'exécuter de nombreuses plongées rapides pour forcer la proie à changer de trajectoire et la distraire. Le processus se poursuit jusqu'à ce qu'il soit choisi le meilleur moment pour attraper la proie. L'équation suivante décrit la décision des faucons de se déplacer pour mettre en œuvre l'encerclement doux.[33]

$$Y = X_{\text{prey}}(t) - E|JX_{\text{prey}}(t) - X(t)| \quad (8)$$

Si les faucons de Harris remarquent que la proie effectue des mouvements trompeurs et qu'elle est sur le point de s'échapper, ils intensifieront leurs plongées brusques, irrégulières et rapides. La nouvelle technique des faucons repose sur les combats de Lévy (LF) comme le montre l'équation suivante.

$$Z = Y + S * LF(D) \quad (9)$$

où D indique la dimension du problème, S fait référence à un vecteur aléatoire de taille  $1 \times D$ , et LF est calculé selon l'équation suivante.

$$LF(x) = 0.01 * \frac{u * \sigma}{|v|^{\frac{1}{\beta}}} \quad (10)$$

$$\sigma = \left( \frac{\Gamma(1 + \beta) * \sin \frac{\pi\beta}{2}}{\Gamma(\frac{1+\beta}{2}) * \beta * 2^{\frac{\beta-1}{2}}} \right) \quad (11)$$

où u et v font référence à une valeur aléatoire dans l'intervalle (0,1),  $\beta$  indique la variable fixe définie à 1.5. Par conséquent, le modèle

## Algorithme harris hawks optimisation

---

mathématique pour mettre à jour les positions des faucons dans l'étape d'encerclement souple est donné par l'équation suivante.

$$X(t+1) = \begin{cases} Y & \text{if } F(Y) < F(X(t)). \\ Z & \text{if } F(Z) < F(X(t)). \end{cases} \quad (12)$$

dans le cas de l'équation 8,  $Y$  représente la valeur calculée selon la formule spécifiée, tandis que dans l'équation 9,  $Z$  représente la valeur calculée selon la formule spécifiée.

Le résumé des procédures de l'algorithme HHO est présenté sous forme de pseudo-code, où la valeur de  $X_m(t)$  est calculée en utilisant l'équation 4.

4. **Siège rigide avec plongeurs rapides progressifs:** Dans le dernier cas, lorsque les valeurs de  $r < 0,5$  et  $|E| < 0,5$ , cela signifie que la proie n'a pas suffisamment de force pour s'échapper. Dans le même temps, les faucons cherchent à réduire l'espace entre eux et la proie avant de la surprendre et de l'attaquer. L'équation suivante décrit la mise à jour des positions des faucons.[33]

$$X(t+1) = \begin{cases} Y & \text{if } F(Y) < F(X(t)). \\ Z & \text{if } F(Z) < F(X(t)). \end{cases} \quad (13)$$

$$Y = X_{\text{prey}}(t) - E |JX_{\text{prey}}(t) - X_m(t)| \quad (14)$$

Le résumé des procédures de l'algorithme HHO est présenté sous forme de pseudo-code, où la valeur de  $X_m(t)$  est calculée en utilisant l'équation 1.[33]

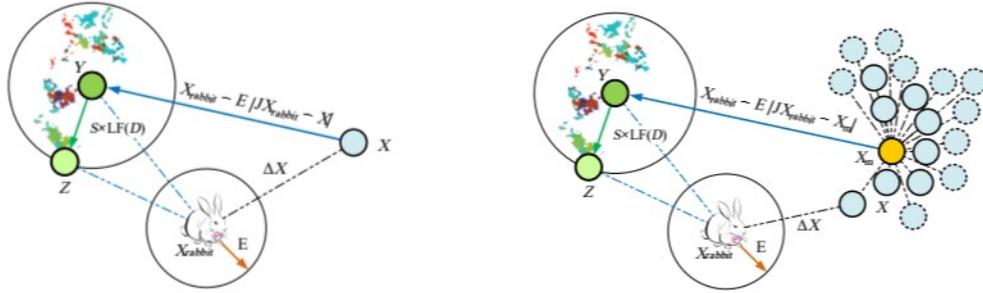


Figure 13: Exemple de vecteurs globaux dans le cas d'un siège dur avec des plongées rapides progressives dans l'espace 2D et 3D[32]

## 2.3 algorithme harris hawks optimisation code[32]

---

**Algorithm 1** Harris Hawks Optimization (HHO)

---

```
1: Input: the population size  $N$  and maximum number of iteration  $T$ 
2: Outputs: The location of rabbit and its fitness value initialize the random population  $X_i = (i = 1, 2, \dots, N)$ 
3: while Stopping condition is not met do
4:   Calculate fitness values of hawks
5:   set  $X_{\text{rabbit}}$  as the location of rabbit (best location)
6:   for each hawk  $x_i$  do
7:     Update the initial energy  $E_0$  and jump strength  $J$ .  $E_0 = 2rand() - 1 = 2(1 - rand())$ 
8:     Update the  $E$  using Eq. 1
9:     if  $|E| \geq 1$  then
10:      Update the location vector using Eq. 3
11:    end if
12:    if  $|E| < 1$  and  $r \geq 0.5$  and  $|E| \geq 0.5$  then
13:      Update the location vector using Eq. 5
14:    else if  $r \geq 0.5$  and  $|E| < 0.5$  then
15:      Update the location vector using Eq. 8
16:    else if  $r < 0.5$  and  $|E| \geq 0.5$  then
17:      Update the location vector using Eq. 13
18:    else if  $r < 0.5$  and  $|E| < 0.5$  then
19:      Update the location vector using Eq. 14
20:    end if
21:  end for
22: end while
23: Return  $X_{\text{rabbit}}$ 
```

---

## 3 conclusion

ALgorithem d'optimisation Harris Hawks (HHO) présente de réels avantages pour les algorithmes d'optimisation inspirés du comportement de chasse des faucons Harris. En utilisant des stratégies basées sur l'observation, le suivi et la détection des proies, HHO permet une exploration efficace de l'espace de recherche et améliore les performances globales de l'algorithme.

L'utilisation de différentes phases, telles que l'encerclement doux et les plonges rapides, permet à HHO de capturer les proies de manière plus efficace et de maximiser les chances de convergence vers des solutions optimales. De plus, l'intégration de mécanismes aléatoires tels que les sauts aléatoires et les vols fiscaux ajoute une composante d'exploration plus diversifiée à l'algorithme.

Il est important de souligner que HHO peut également être combiné avec d'autres protocoles et techniques d'optimisation, tels que LEACH, pour résoudre des problèmes spécifiques dans des domaines tels que les réseaux sans fil. Cette combinaison permet d'optimiser davantage l'allocation des ressources et d'améliorer les performances globales du système.

En résumé, Algorithme Harris Hawks Optimisation offre une approche innovante et prometteuse pour les problèmes d'optimisation. Son utilisation de stratégies inspirées de la chasse des faucons Harris permet d'améliorer l'exploration, l'exploitation et la convergence, et il peut être adapté et combiné avec d'autres techniques pour répondre à des besoins spécifiques. Cependant, des recherches supplémentaires sont nécessaires pour explorer pleinement son potentiel et son application dans divers domaines.

## *Chapter 04*

# Simulation

## Chapter V

# Simulation

## 1 Introduction

Pour atteindre cet objectif, une technique ou un protocole de routage efficace est nécessaire.

Dans ce chapitre, nous présenterons notre algorithme hybride détaillé, qui combine les avantages de protocole LEACH et algorithm HHO. Nous décrirons l'architecture opérationnelle de notre algorithme et expliquerons son concept de base. L'objectif principal de notre approche hybride est de maximiser la durée de vie du réseau en optimisant la consommation d'énergie lors de la communication.

## 2 Algorithme proposé: HHO-LEACH Algorithme

Nous allons en apprendre plus sur l'algorithme en particulier.

### 2.1 Description générale et les objectifs de l'algorithme proposé

L'algorithme proposé vise à minimiser la consommation d'énergie du RCSF, donc une sélection efficace de CH est une étape importante pour atteindre cet objectif.

En d'autres termes, l'énergie dissipée lors du processus de communication est le facteur principal que nous devons réduire. Une sélection efficace des Cluster Heads (CH) peut prendre en compte cette préoccupation. L'efficacité de la sélection des CH a un impact direct sur l'efficacité énergétique du réseau. Si la sélection des CH est efficace, cela conduit à une meilleure efficacité énergétique. En revanche, une sélection inefficace des CH peut entraîner une consommation d'énergie accrue, car les CH ont tendance à consommer plus d'énergie que les nœuds non-CH.

Pour atteindre une sélection efficace des CH et de leurs emplacements, nous utilisons l'algorithme HHO. L'objectif est de minimiser l'énergie dissipée dans le processus de communication, en réduisant une fonction cible spécifique  $OF_i$

$$OF_i = XOF_{energy} + \varphi(1 - OF_{Dis}) \quad (15)$$

Où  $OF_{Energy}$  c'est pour tous les l'énergie de cluster heads vivants dans le champ du capteur. Il est décrit par:

$$OF_{energy} = \frac{1}{K} \left\{ \sum_{p=1}^K X(C_N^p) \right\} + \frac{1}{T_{CH}} \left\{ \sum_{i=1}^{CH} X(X_{CH}^i) \right\}. \quad (16)$$

où  $X_{p+1}(C_i^{CH})$  déterminent l'énergie retenue après transmission des paquets de données vers le CH.  $X(C_p^N)$  est l'énergie dissipation du pième nœud.

$$X_{p+1}(C_P^N) = X_P(C_N^P) - X(C_N^p) \quad (17)$$

où  $X_{p+1}(C_i^{CH})$  déterminer l'énergie retenue dans CH après recevoir les paquet données du nœud.  $X(C_i^{CH})$  est l'énergie dissipation de ith CH

$$X_{p+1}(C_i^{CH}) = X_P(C_i^{CH}) - X(C_N^p) \quad (18)$$

Où  $OF_{Distance}$  Le CH optimal est identifié sur la base de la distance minimale entre le nœud et la BS. La distance entre CH et le nœud et CH à BS est représenté à l'aide de la fonction objectif. Il est décrit par

$$OF_{Distance} = \sum_{p=1}^P \sum_{i=1}^{T_{CH}} \frac{\|D_N^p - D_{CH}^i\| + \|D_{CH}^i - D_{BS}\|}{Q * Q} \quad (19)$$

$\|D_N^p - D_{CH}^i\|$ :détermine la distance entre le pième nœud et avec CH.

$\|D_{CH}^i - D_{BS}\|$  : détermine la distance entre le ième CH et le BS.

Q:désigne la zone de détection en mètres.

[35]

### 2.1.1 Algorithm1:HHO-LEACH algorithm de selection efficace De CH

L'algorithme d'optimisation de Harris Hawks (HHO) peut être appliqué au problème de la sélection des CH dans les réseaux de capteurs sans fil (WSN). La sélection du CH est une tâche cruciale dans les WSN, où un sous-ensemble de nœuds de capteurs est choisi pour agir en tant que CHs responsables de l'agrégation des données et de la communication avec la station de base.

Voici comment l'algorithme HHO peut être utilisé pour la sélection des têtes de cluster dans les WSN :

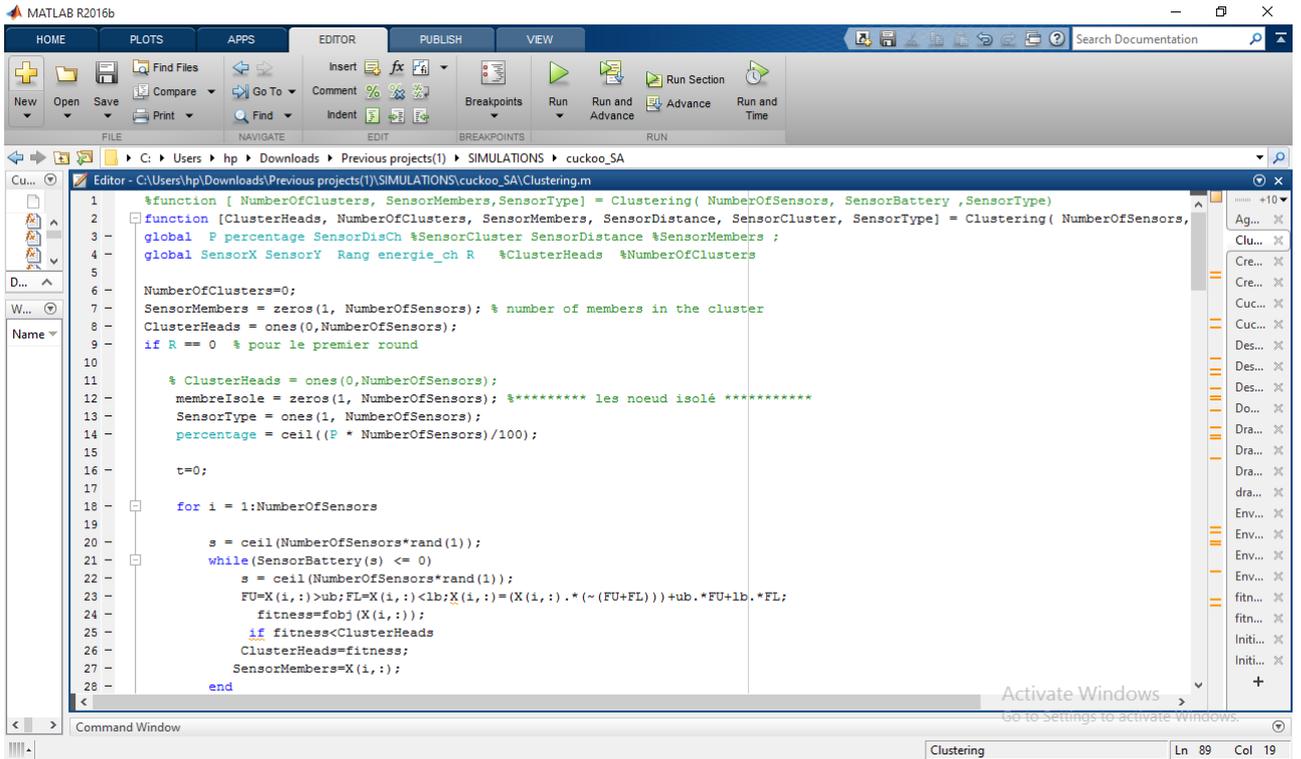
1. **Initialisation :** Initialiser une population de CH potentiels. Chaque CH potentiel représente un nœud de capteur candidat pour la sélection.
2. **Évaluation de la condition physique :** évaluez la condition physique de chaque CH potentiel en fonction de certains critères, tels que le niveau d'énergie, la distance à la station de base ou une combinaison de plusieurs facteurs. La fonction de fitness doit refléter les objectifs du processus de sélection du CH, tels que la réduction de la consommation d'énergie ou la maximisation de la durée de vie du réseau.
3. **Processus de chasse :** L'algorithme HHO imite le comportement de chasse des faucons de Harris, où les faucons collaborent pour capturer efficacement les proies. Dans le contexte des WSN, le processus de chasse implique la mise à jour itérative des positions des CHs potentiels en fonction de leurs valeurs de fitness. Ce processus peut inclure des phases d'exploration et d'exploitation pour équilibrer l'exploration de l'espace de recherche et l'exploitation de solutions prometteuses.
4. **Collaboration et échange d'informations :** dans l'algorithme HHO, les faucons communiquent et partagent des informations pour améliorer les performances globales de la population. De même,

dans les WSN, lesCHs potentiels peuvent échanger des informations sur leurs valeurs de fitness, leurs niveaux d'énergie ou d'autres paramètres pertinents pour prendre des décisions éclairées pendant le processus de sélection. Cette collaboration peut aider à identifier lesCHs les plus appropriés pour le réseau

5. **Itération et terminaison** : répétez le processus de recherche pendant un certain nombre d'itérations ou jusqu'à ce qu'une condition de terminaison soit remplie. La condition de fin peut être un nombre maximum d'itérations, atteignant un seuil de fitness souhaité, ou d'autres critères prédéfinis.
6. **Sélection finale** : après les itérations, sélectionnez lesCHs en fonction de leurs valeurs de fitness ou de tout autre critère prédéfini. LesCHs sélectionnés assumeront la responsabilité de l'agrégation et de la communication des données au sein de leurs clusters respectifs.

En appliquant l'algorithme HHO à la sélection des CH dans les WSN, il est possible d'optimiser le processus de sélection en fonction des objectifs et des contraintes définis du réseau. La nature collaborative et itérative de l'algorithme HHO peut aider à identifier les CH optimales ou quasi optimales qui peuvent améliorer l'efficacité énergétique, la durée de vie du réseau et les performances globales du WSN.

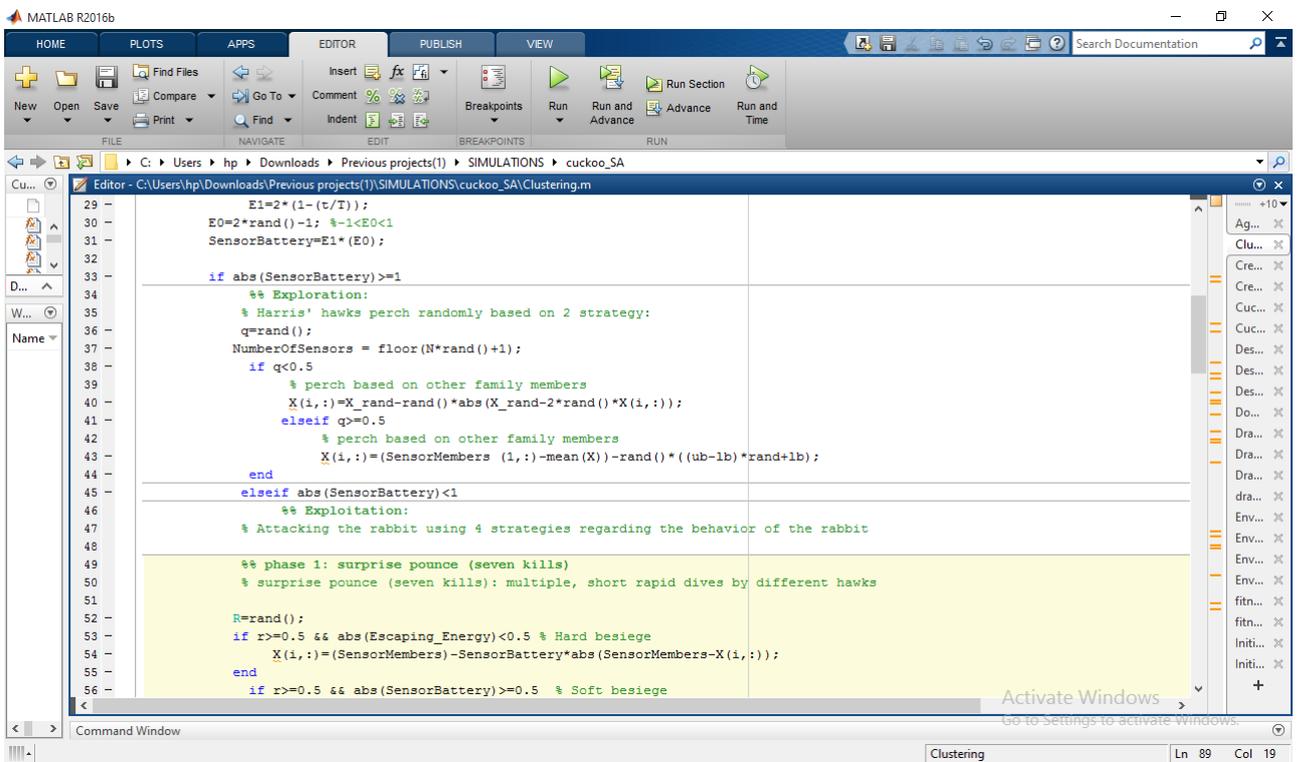
# Simulation



The image shows the MATLAB R2016b editor interface. The main window displays a MATLAB script named 'Clustering.m'. The code defines a function for clustering based on sensor data. It includes initialization of variables, a loop for each sensor, and a while loop for finding the best cluster head. The code is as follows:

```
1 %function [ NumberOfClusters, SensorMembers, SensorType ] = Clustering( NumberOfSensors, SensorBattery, SensorType)
2 function [ ClusterHeads, NumberOfClusters, SensorMembers, SensorDistance, SensorCluster, SensorType ] = Clustering( NumberOfSensors,
3 global P percentage SensorDisCh %SensorCluster SensorDistance %SensorMembers ;
4 global SensorX SensorY Rang energie_ch R %ClusterHeads %NumberOfClusters
5
6 NumberOfClusters=0;
7 SensorMembers = zeros(1, NumberOfSensors); % number of members in the cluster
8 ClusterHeads = ones(0,NumberOfSensors);
9 if R == 0 % pour le premier round
10
11 % ClusterHeads = ones(0,NumberOfSensors);
12 membreIsole = zeros(1, NumberOfSensors); %***** les noeud isolé *****
13 SensorType = ones(1, NumberOfSensors);
14 percentage = ceil((P * NumberOfSensors)/100);
15
16 t=0;
17
18 for i = 1:NumberOfSensors
19
20 s = ceil(NumberOfSensors*rand(1));
21 while (SensorBattery(s) <= 0)
22 s = ceil(NumberOfSensors*rand(1));
23 FU=X(i,:)>ub;FL=X(i,:)<lb;X(i,:)=(X(i,:).*(~(FU+FL)))+ub.*FU+lb.*FL;
24 fitness=fobj(X(i,:));
25 if fitness<ClusterHeads
26 ClusterHeads=fitness;
27 SensorMembers=X(i,:);
28 end
```

Figure 14: ALGORITHME DE HHO-LEACH

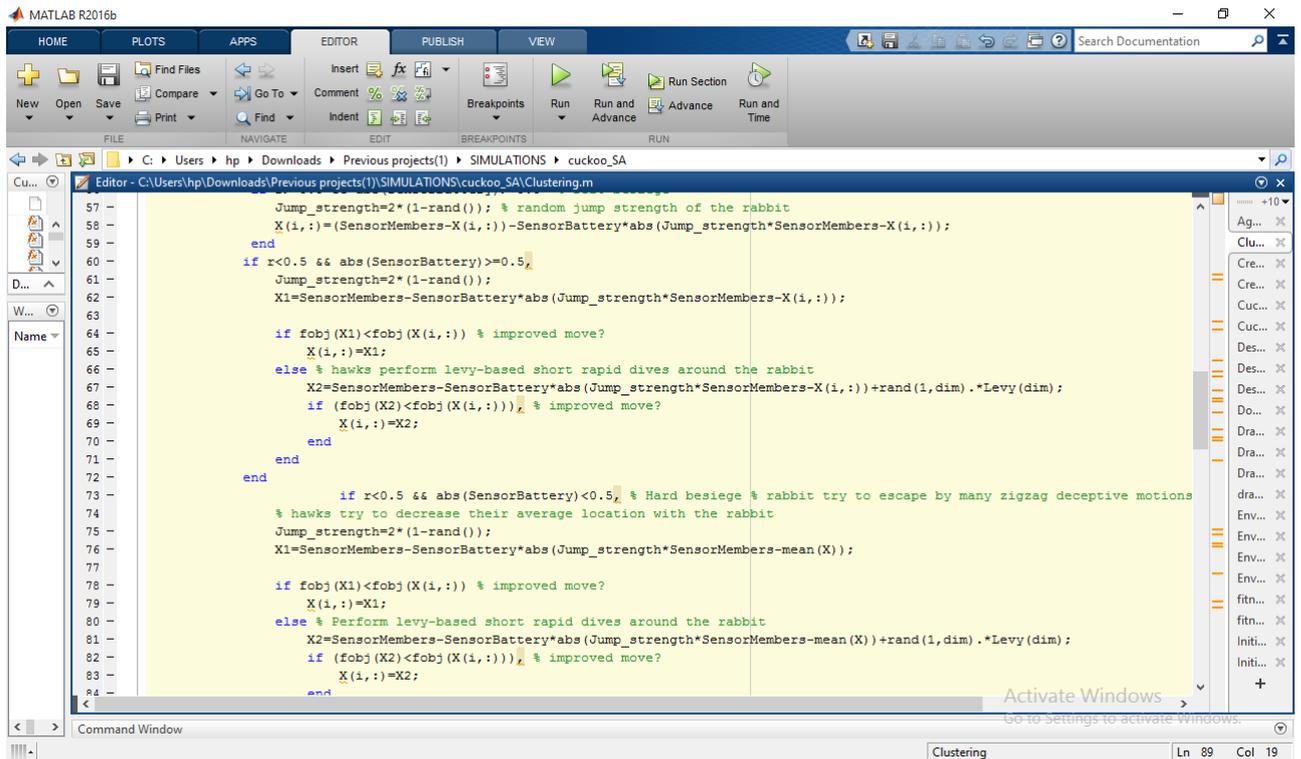


The image shows the MATLAB R2016b editor interface. The main window displays a MATLAB script named 'Clustering.m'. The code continues from the previous figure, showing the exploration and exploitation phases of the HHO-LEACH algorithm. The code is as follows:

```
29 E1=2*(1-(t/T));
30 E0=2*rand()-1; %-1<E0<1
31 SensorBattery=E1*(E0);
32
33 if abs(SensorBattery)>=1
34 %% Exploration:
35 % Harris' hawks perch randomly based on 2 strategy:
36 q=rand();
37 NumberOfSensors = floor(N*rand()+1);
38 if q<0.5
39 % perch based on other family members
40 X(i,:)=X_rand-rand()*abs(X_rand-2*rand()*X(i,:));
41 elseif q>=0.5
42 % perch based on other family members
43 X(i,:)=(SensorMembers(1,:)-mean(X))-rand()*((ub-lb)*rand+lb);
44 end
45 elseif abs(SensorBattery)<1
46 %% Exploitation:
47 % Attacking the rabbit using 4 strategies regarding the behavior of the rabbit
48
49 %% phase 1: surprise pounce (seven kills)
50 % surprise pounce (seven kills): multiple, short rapid dives by different hawks
51
52 R=rand();
53 if r>=0.5 && abs(Escaping_Energy)<0.5 % Hard besiege
54 X(i,:)=(SensorMembers)-SensorBattery*abs(SensorMembers-X(i,:));
55 end
56 if r>=0.5 && abs(SensorBattery)>=0.5 % Soft besiege
```

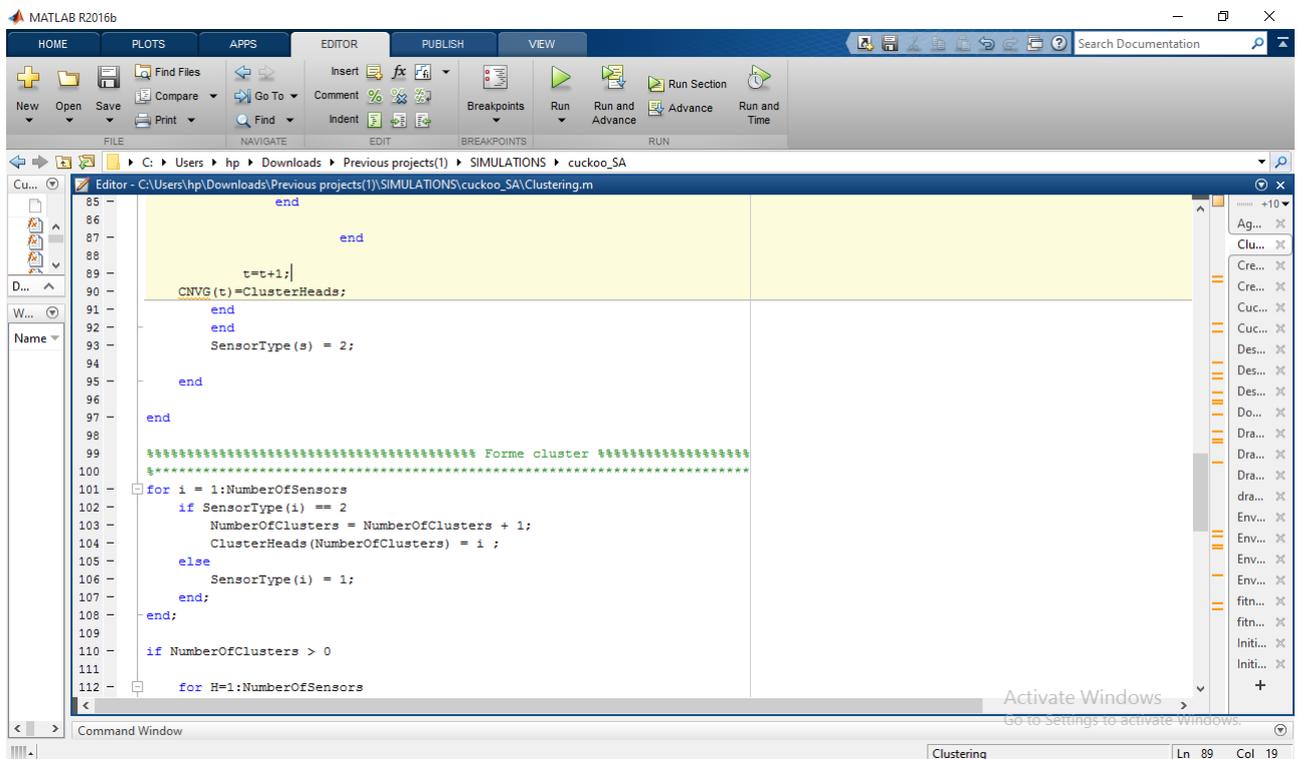
Figure 15: ALGORITHME DE HHO-LEACH

# Simulation



```
57 Jump_strength=2*(1-rand()); % random jump strength of the rabbit
58 X(i,:)=(SensorMembers-X(i,:))-SensorBattery*abs(Jump_strength*SensorMembers-X(i,:));
59 end
60 if r<0.5 && abs(SensorBattery)>=0.5
61 Jump_strength=2*(1-rand());
62 X1=SensorMembers-SensorBattery*(Jump_strength*SensorMembers-X(i,:));
63
64 if fobj(X1)<fobj(X(i,:)) % improved move?
65 X(i,:)=X1;
66 else % hawks perform levy-based short rapid dives around the rabbit
67 X2=SensorMembers-SensorBattery*abs(Jump_strength*SensorMembers-X(i,:))+rand(1,dim).*Levy(dim);
68 if (fobj(X2)<fobj(X(i,:))) % improved move?
69 X(i,:)=X2;
70 end
71 end
72 end
73 if r<0.5 && abs(SensorBattery)<0.5, % Hard besiege % rabbit try to escape by many zigzag deceptive motions
74 % hawks try to decrease their average location with the rabbit
75 Jump_strength=2*(1-rand());
76 X1=SensorMembers-SensorBattery*(Jump_strength*SensorMembers-mean(X));
77
78 if fobj(X1)<fobj(X(i,:)) % improved move?
79 X(i,:)=X1;
80 else % Perform levy-based short rapid dives around the rabbit
81 X2=SensorMembers-SensorBattery*abs(Jump_strength*SensorMembers-mean(X))+rand(1,dim).*Levy(dim);
82 if (fobj(X2)<fobj(X(i,:))) % improved move?
83 X(i,:)=X2;
84 end
```

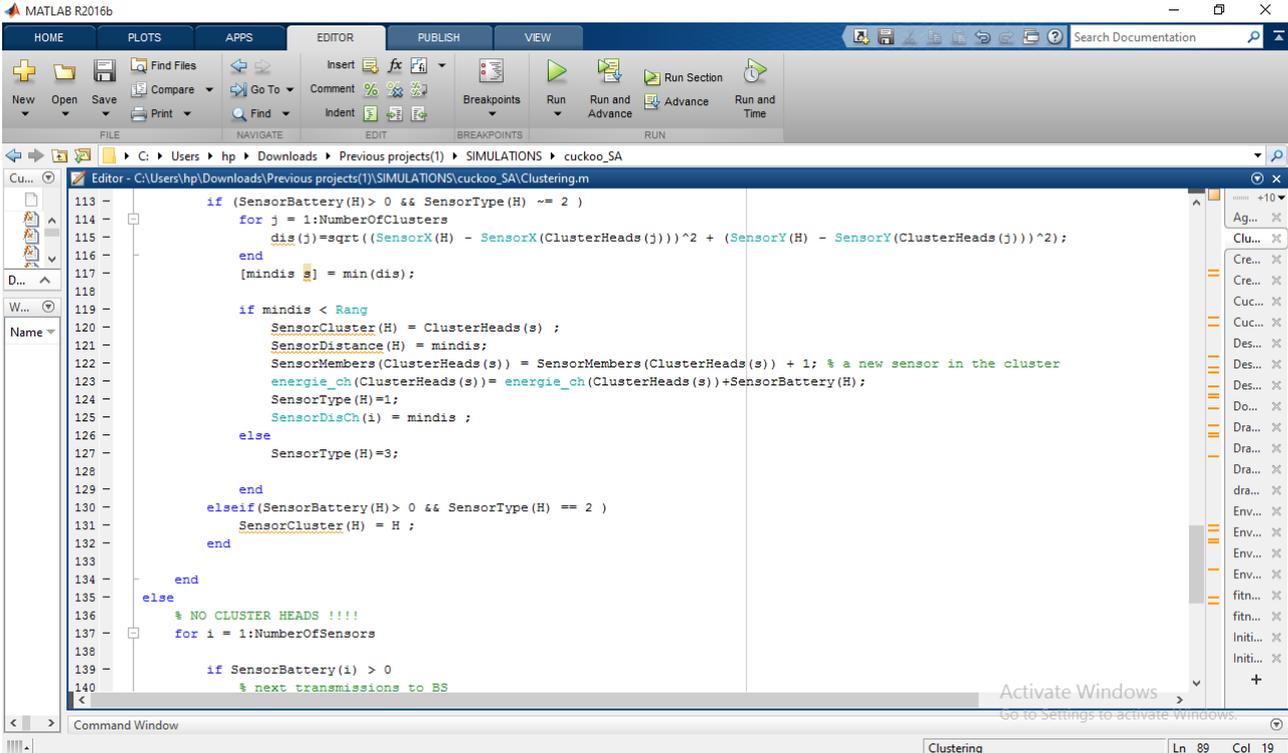
Figure 16: ALGORITHM DE HHO-LEACH



```
85 end
86 end
87 end
88 t=t+1;
89 CNVG(t)=ClusterHeads;
90 end
91 end
92 end
93 SensorType(s) = 2;
94
95 end
96
97 end
98
99 %***** Forme cluster %*****
100
101 for i = 1:NumberOfSensors
102 if SensorType(i) == 2
103 NumberOfClusters = NumberOfClusters + 1;
104 ClusterHeads(NumberOfClusters) = i ;
105 else
106 SensorType(i) = 1;
107 end;
108 end;
109
110 if NumberOfClusters > 0
111 for H=1:NumberOfSensors
```

Figure 17: ALGORITHM DE HHO-LEACH

# Simulation



The image shows the MATLAB R2016b editor interface. The main window displays a MATLAB script for a clustering simulation. The script is located at `C:\Users\hp\Downloads\Previous projects(1)\SIMULATIONS\cuckoo_SA\Clustering.m`. The code is as follows:

```
113 - if (SensorBattery(H) > 0 && SensorType(H) ~= 2 )
114 -     for j = 1:NumberOfClusters
115 -         dis(j) = sqrt((SensorX(H) - SensorX(ClusterHeads(j)))^2 + (SensorY(H) - SensorY(ClusterHeads(j)))^2);
116 -     end
117 -     [mindis s] = min(dis);
118 -
119 -     if mindis < Rang
120 -         SensorCluster(H) = ClusterHeads(s) ;
121 -         SensorDistance(H) = mindis;
122 -         SensorMembers(ClusterHeads(s)) = SensorMembers(ClusterHeads(s)) + 1; % a new sensor in the cluster
123 -         energie_ch(ClusterHeads(s)) = energie_ch(ClusterHeads(s)) + SensorBattery(H);
124 -         SensorType(H) = 1;
125 -         SensorDisCh(i) = mindis ;
126 -     else
127 -         SensorType(H) = 3;
128 -     end
129 -
130 - elseif (SensorBattery(H) > 0 && SensorType(H) == 2 )
131 -     SensorCluster(H) = H ;
132 - end
133 -
134 - end
135 -
136 - else
137 -     % NO CLUSTER HEADS !!!!
138 -     for i = 1:NumberOfSensors
139 -         if SensorBattery(i) > 0
140 -             % next transmissions to BS
```

Figure 18: ALGORITHM DE HHO-LEACH

## 3 Implémentation

### 3.1 Choix du langage

Nous avons choisi le langage MATLAB 2016b à la programmation de notre Algorithme HHO-LEACH

### 3.2 Simulation

Dans cette simulation notre modèle d'expérimentation est établi sur 100 nœuds répartis aléatoirement sur une surface carrée de 100 x 100 m<sup>2</sup> présentée par la figure suivante :



Figure 19: Instructions

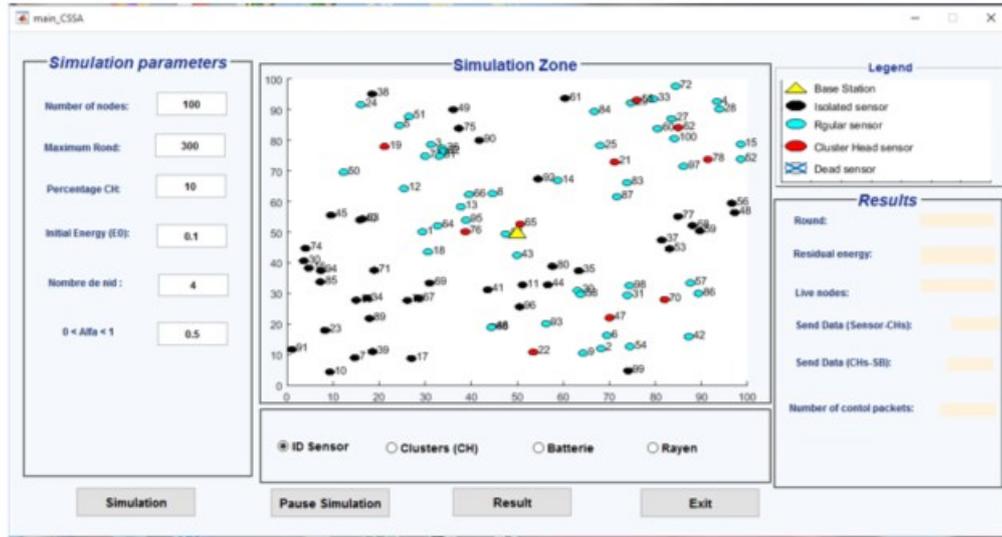


Figure 20: id senseur de chaque nœuds

*Nous supposons que tous les nœuds ont une position fixe durant toute la période de simulation. Notre modèle de simulation utilise les paramètres résumés dans le tableau suivant :*

Paramètre	Valeur
La taille du réseau.	100 m × 100 m
La localisation de la station de la base.	(2.2, 13)
Le nombre des nœuds	100 N
L'énergie initiale des nœuds	0.50 J

Table 2: Les paramètres de simulation.

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 données sans oublier aussi le cas où il exécute des opérations de traitement de données (collecte et agrégation de données). On néglige l'énergie consommée lors des calculs simples et les mises à jour. Tous les nœuds du réseau commencent la simulation par énergie initiale égale à 0.5 J et une quantité de données illimitées à transmettre à la station de base. De plus, l'énergie de la station de base est illimitée. Chaque nœud consomme sa réserve d'énergie limitée tout au long de la durée de simulation, ce qui implique l'épuisement de celle-ci. Ainsi, tout nœud qui a épuisé sa réserve d'énergie est considéré comme mort. Par



conséquent, il ne peut ni transmettre ni recevoir des données.

### 3.3 Résultats de simulation

Après la simulation de notre protocoles HHO et le protocole LEACH, nous montrons les performances du Algorithme HHO-LEACH par une comparaison sur certaines métriques :

1. **L'énergie Consommée :** Figure 25 représente l'énergie consommée dans le Algorithme HHO proposé et dans LEACH en fonction du temps. Nous remarquons que dans notre algorithme proposé, il présente une augmentation très lente de la consommation d'énergie HHO, par contre dans LEACH nous observons une augmentation très rapide de la consommation d'énergie, où le protocole LEACH atteint déjà la valeur de consommation 14 en un instant 50, sur le d'autre part notre protocole présente l'énergie totale

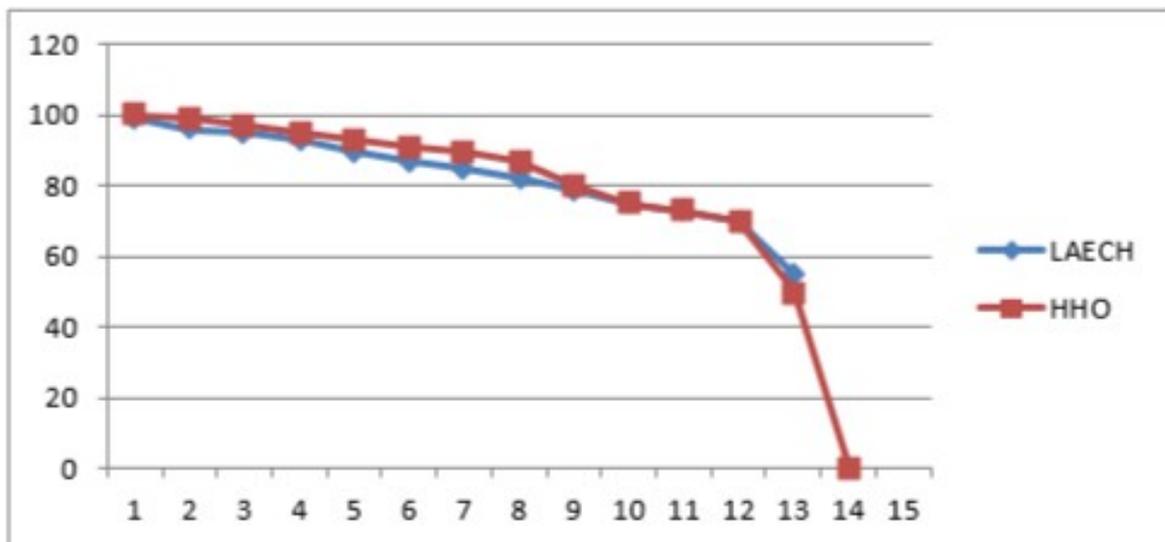


Figure 24: L'énergie Consommée du ALGORITHEM LEACH et HHO

2. **Comparaison de la durée de vie :** Figure 26 représente le nombre de nœuds actifs dans les deux protocoles. Dans LEACH, nous voyons une diminution très rapide du nombre de nœuds actifs en fonction du temps, tous les nœuds sont morts dans le temps 13 C'est

ce qui provoque une durée de vie très courte de LEACH. D'autre part, l'algorithme proposé présente une durée de vie plus longue, car la durée de vie dure jusqu'au temps 14 avec une diminution très lente et progressive du nombre de nœuds actifs en fonction du temps.

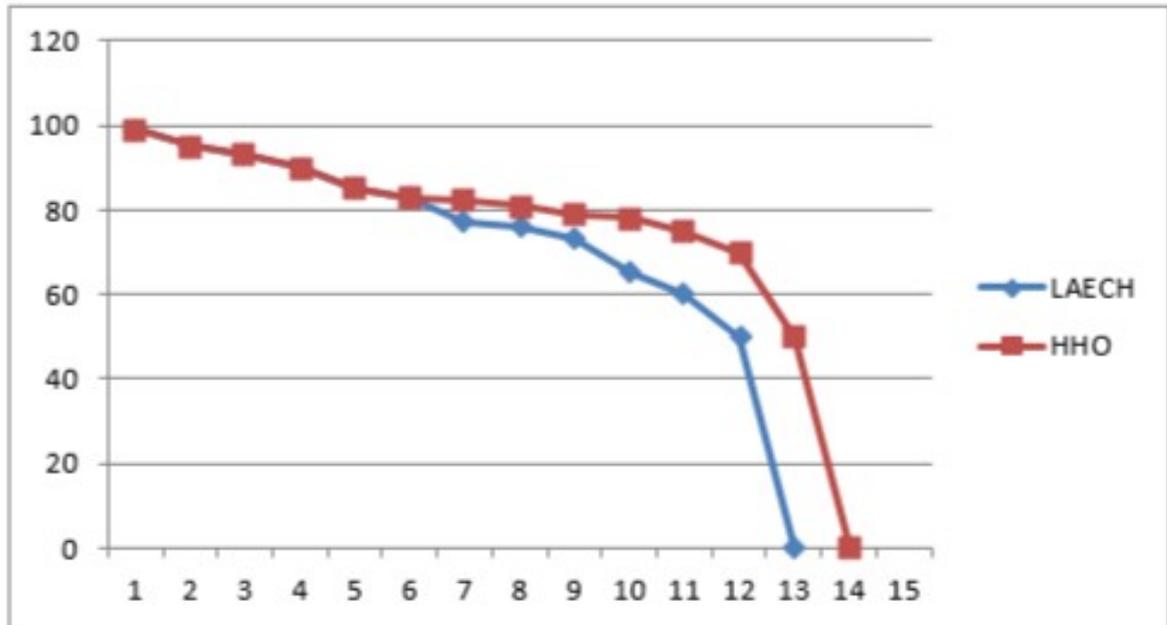


Figure 25: La durée de vie du ALGORITHHEM LEACH et HHO

### 3. Comparaison des données reçues :

Figure 27 ci-dessous est une comparaison des deux protocoles à la fin des données reçues. Le schéma ci-dessous montre que notre protocole, HHO, permet la transmission de données jusqu'à 14 fois avec un grand nombre de messages, 60 000 messages. En revanche, la transmission de données dans LEACH s'arrête à 14 avec 49950 messages. Cela indique que notre algorithme offre une amélioration significative de la quantité de données reçues dans le réseau.

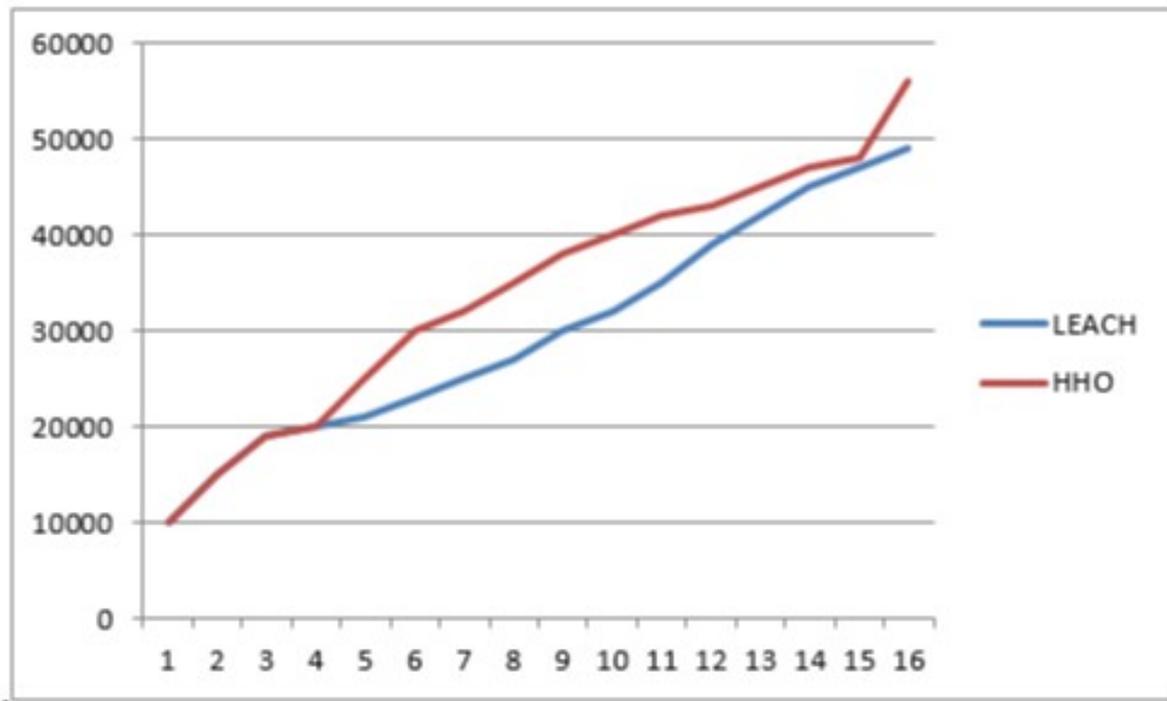


Figure 26: La durée de vie du ALGORITHHEM LEACH et HHO

### 3.4 Résultats et Discussion

Après avoir comparé les performances des protocoles LEACH et HHO, nous avons observé plusieurs avantages de notre algorithme proposé, notamment :

1. Faible consommation d'énergie : Notre algorithme présente une consommation d'énergie inférieure par rapport à LEACH, ce qui permet de prolonger la durée de vie du réseau. Cela est obtenu grâce à une sélection efficace des têtes de cluster à l'aide de l'algorithme génétique (HHO).
2. Augmentation de la réception des données à la station de base (BS) : Avec notre algorithme, la quantité de données reçues à la BS est plus élevée par rapport à LEACH. Cela est dû à une sélection optimisée des têtes de cluster, ce qui garantit une meilleure agrégation et transmission des données vers la BS.
3. Durée de vie du réseau plus longue : En sélectionnant efficacement

les têtes de cluster à l'aide de l'algorithme HHO, notre algorithme proposé atteint une durée de vie du réseau plus longue lors des simulations. Cela est dû à une meilleure efficacité énergétique et à une distribution équilibrée des têtes de cluster dans l'ensemble du réseau.

En revanche, LEACH présente des variations significatives du nombre de têtes de cluster par cycle, ce qui entraîne une mauvaise couverture du réseau et une distribution inégale des têtes de cluster. Cela se traduit par des nœuds isolés qui transmettent directement leurs données à la station de base, entraînant une consommation d'énergie plus élevée et ayant un impact négatif sur la durée de vie globale du réseau.

En résumé, notre algorithme proposé basé sur HHO offre des avantages tels qu'une faible consommation d'énergie, une réception accrue des données à la station de base et une durée de vie du réseau plus longue par rapport à LEACH. L'utilisation de HHO pour la sélection des têtes de cluster garantit une couverture réseau efficace et une distribution optimale des têtes de cluster, ce qui améliore les performances et l'utilisation des ressources dans les WSN (Wireless Sensor Networks).

## 4 Conclusion

Dans ce chapitre, nous décrivons l'algorithme HHO-LEACH proposé qui améliore les performances du protocole LEACH classique en introduisant une méthode nouvelle et efficace pour la sélection des têtes de cluster (CH) basée sur un algorithme (HHO). Le processus de sélection prend en compte des paramètres importants dans la fonction objectif, à savoir l'énergie résiduelle des nœuds et l'intensité des nœuds. Les résultats de simulation à l'aide de MATLAB ont démontré que notre algorithme proposé surpasse LEACH en termes de conservation d'énergie, de quantité de données envoyées à la station de base et de durée de vie du réseau.

En résumé, l'algorithme HHO-LEACH améliore les performances de LEACH en optimisant la sélection des CHs en fonction de paramètres importants. Les résultats de la simulation valident la supériorité de notre proposition sur LEACH en termes d'efficacité énergétique, de transmission de données et de durée de vie du réseau.

# Conclusion

## Chapter VI

# Conclusion

### Conclusion

Les WSN sont composés de nombreux capteurs miniaturisés déployés à une densité élevée dans des zones nécessitant une surveillance et un suivi. Comparé aux systèmes traditionnels câblés, le déploiement de ces capteurs se fait à moindre coût. Chaque capteur est équipé d'un ou plusieurs éléments de détection (comme le mouvement, la température, la pression, etc.), d'une batterie, de récepteurs radio à faible puissance, d'un microprocesseur et d'une mémoire limitée. Un aspect important de ces réseaux est que les nœuds fonctionnent sans surveillance, disposent d'une énergie limitée et la topologie du réseau est inconnue. De nombreux défis de conception découlent des ressources limitées dont disposent ces nœuds et de leur déploiement dans des environnements hostiles. Il s'agit de l'algorithme HHO-LEACH. Ce protocole est inspiré par le comportement des faucons. Afin de démontrer les performances en termes de conservation d'énergie de notre algorithme HHO-LEACH, nous l'avons simulé et comparé avec le protocole LEACH. Les résultats de la simulation ont montré de bons résultats dans la plupart des cas, avec une consommation d'énergie très réduite, ce qui prolonge la durée de vie du réseau. Dans une perspective d'amélioration des résultats obtenus, nous envisageons les adaptations suivantes :

1. Une hybridation entre notre algorithme avec d'autres méthodes de la littérature (ACO, PSO...).
2. L'évitement de toutes sortes d'envois à des longues distances (d'un CH vers la station de base) en intégrant une technique de routage multi-sauts (d'un CH vers ses voisins jusqu'à la station de base).

## References

- [1] : Amélioration de l'efficacité énergétique de RCSF à l'aide d'une approche multicouche
- [2] :D. Bein and A.K. Datta. "a self-stabilizing directed diffusion protocol for sensor networks". In International Conference on Parallel Processing Workshops, 2004.
- [3] : Vijay K. Varadan, K.J. Vinoy, K.A. Jose, 'RF MEMS and Their Applications', John Wiley & Sons, 2003.
- [4] : Aurélie Bunel, « Les Réseaux de Capteurs Sans Fil », mars 2007.
- [5] : J. Joy Winston and B. Paramasivan, A Survey on Connectivity Maintenance and Preserving Coverage for Wireless Sensor Networks, International Journal of Research and Reviews in Wireless Sensor Networks (IJRRWSN), Vol. 1, No. 2, ISSN : 2047-0037, pp.1118, June 2011.
- [6] : Aurélie Bunel, « Les Réseaux de Capteurs Sans Fil », mars 2007.
- [7] : J. Joy Winston and B. Paramasivan, A Survey on Connectivity Maintenance and Preserving Coverage for Wireless Sensor Networks, International Journal of Research and Reviews in Wireless Sensor Networks (IJRRWSN), Vol. 1, No. 2, ISSN : 2047-0037, pp.1118, June 2011.
- [8] : I. Chlamtac, I. Carreras, H Woesner, 'From internets to bionets :biological kinetic service oriented networks". The case study of Bionetic Sensor networks. CREATE-NET Research Consortium, Trento, Italy. 2005.
- [9] : I.F. Akyildiz, W.S. Sankarasubramaniam, E. Cayirci, "Wireless Sensor Network: A Survey". Computer networks, Vol. 38, pp. 393-422. 2002.
- [10] : Rahim Kacimi, "techniques de conservation d'énergie pour les réseaux de capteurs sans fil ", thèse de doctorat en réseaux et télécommunication, université de Toulouse, Septembre 2009.
- [11] : Shio Kumar Singh et al, "Applications, Classifications, and Selections of Energy-Efficient Routing Protocols for Wireless Sensor Networks", INTERNATIONAL JOURNAL OF ADVANCED ENGINEERING SCIENCES AND TECHNOLOGIES (IJAEST), Vol No. 1, Issue No. 2, p85 - 95
- [12] : J.Deng, Y.S.Han, W.B.Heinzelman and P.K.Varshney, "Scheduling Sleeping Nodes in High Density Cluster-based Sensor Networks", mobile Networks and Applications, vol.10, pp.825-835, 2005.
- [13] : T.Nieberg, S.Dulman, P.Havinga, L.V.Hoesel, J.Wu, "Collaborative Algorithms for Communication in Wireless Sensor Networks", Ambient Intelligence : Impact on Embedded Systems Design, Kluwer Academic publishers, 2003.
- [14] : M.E.KHANOUCHE, "Le traitement du problème de la couverture dans les réseaux de capteurs sans fil", thèse, faculté des sciences exactes universités A.MIRA-BEJAIA, 2008.

- [15] :Anis Laouiti, Cédric Adjih. Mesures de performances du protocole OLSR. Projet Hipercom. 2003. Rapport technique.
- [16] :W. Heinzelman, J. Kulik, and H. Balakrishnan. Adaptive protocols for information dissemination in wireless sensor networks. 5th annual ACM/IEEE international conference on Mobile computing and networking . August 1999, pp. 174 - 185.
- [17] :Yang X.S, "Nature-Inspired Metaheuristic Algorithms". Luniver Press, UK. 2008
- [18] :Ameer Ahmed Abbasi, Mohamed Younis, "A survey on clustering algorithms for wireless sensor networks", Computer Communications. vol 30,pp 2828-2541,2007.
- [19] :Ossama Younis and Sonia Fahmy: "HEED: A Hybrid, Energy-Efficient, Distributed Clustering Approach for Ad-hoc Sensor Networks", Department of Computer Sciences, Purdue University.
- [20] :Chunjuan Wei, Junjie Yang, Yanjie Gao, Zhimei Zhang, « Cluster-based Routing Protocols in Wireless sensor networks », International Conference on Computer Science and Network Technology, 2011.
- [21] :Jamil Ibriq, Imad Mahgoub, « Cluster-based Routing in Wireless Sensor Networks: issues and challenges », Departement of computer science and engineering, Florida Atlantic University, 2004.
- [22] : Jun Zheng, Abbas Jamalipour "Wireless sensor networks: a networking perspective", Institute of Electrical and Electronics Engineers, 2009
- [23] :I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci "A survey on sensor networks", IEEE Communications Magazine, Vol. 40, pp. 102-116, August 2002.
- [24] :D. J. Dechene, A. El Jardali, M. uccini, A. Sauer, "A Survey of Clustering Algorithms for Wireless Sensor Networks", Project Report, Department of Electrical and Computer Engineering, The University of Western Ontario, Canada, December 2006-2007.
- [25] :Chunjuan Wei, Junjie Yang, Yanjie Gao, Zhimei Zhang "Cluster-based Routing Protocols in Wireless Sensor Networks: A Survey",IEEE International Conference on Computer Science and Network Technology, 2011.
- [26] :M. Younis, M. Youssef, K. Arisha, "Energy-aware management in cluster-based sensor networks", Computer Networks 43 (5) (2003) 649–668.
- [27] : Lyes KHELLADI, Nadjib BADACHE, « les réseaux de capteurs : état de l'art », rapport de recherche, laboratoire des logiciels de base, CERIST, Algérie, février 2004.
- [28] :Luis Javier García Villalba, Ana Lucila Sandoval Orozco, Alicia Triviño Cabrera, Cláudia Jacy, Barenco Abbas, "Routing Protocols in Wireless Sensor Networks",
- [29] :khaled BOUCHAKOUR, « Routage hiérarchique sur les réseaux de capteurs sans fil: Protocole KhLCH (K-hop Layered Clustering Hierarchy) », mémoire de magister, ESI, 2012.

- [30] :E. Yoneki and J. Bacon, "A survey of wireless sensor network technologies: Research trends and middleware's role", Technical Report, Computer Laboratory, Septembre 2005.
- [31] :Jamal N. Al-Karaki Ahmed E. Kamal, "Routing Techniques in Wireless Sensor Networks: A Survey", Departement of Electrical and Computer Engineering, Iowa State University.
- [32] Heidari, A. A., Mirjalili, S., Faris, H., Aljarah, I., Mafarja, M., & Chen, H. (2019). Harris hawks optimization: Algorithm and applications. *Future generation computer systems*, 97, 849-872.
- [33] Shehab, M., Mashal, I., Momani, Z., Shambour, M. K. Y., AL-Badareen, A., Al-Dabet, S., ... & Abualigah, L. (2022). Harris hawks optimization algorithm: variants and applications. *Archives of Computational Methods in Engineering*, 29(7), 5579-5603.
- [34] Lichti Abdelkader & Messiaid Abdessalam, Approche biomimétique pour un routage hiérarchique efficace en énergie dans un réseau de capteurs sans fil, chapitre 4, 2016/2017
- [35] A. A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja and H. Chen, Harris hawks optimization: Algorithm and applications, *Future Gener. Comput. Syst.*, vol. 97, pp. 849-872, Aug. 2019.