



Mémoire de fin de cycle

En vue de l'obtention du diplôme Master professionnel en Informatique

Option : Administration et Sécurité des Réseaux (ASR)

THÈME

Modèles de mobilité du sink pour l'amélioration des performances dans les réseaux de capteurs sans fils

Réalisé par :

Ms. MAHLOUL Yasmina

Ms. MESSAOUDI Lynda

Encadré par :

Dr. BELKHIRI Louiza

Soutenu le 30 juin 2025, Devant le jury composé de :

Mr. Djoudi TOUZI : - Président

Mr. Lachemi KHENOUS : - Examineur

Mme. Ferroudja ZIDANI : - Examinatrice

Mme. Kenza HOCINI : - Examinatrice

Promotion : 2024/2025

Remerciements

Avant tout, nous remercions **Allah**, le Tout-Puissant, de nous avoir donné la force, le courage et la patience nécessaires pour mener à bien ce travail.

Nous tenons à exprimer notre profonde gratitude à **Madame Belkhiri Louiza**, pour sa disponibilité, son accompagnement constant, ses précieux conseils et sa confiance tout au long de ce projet. Son soutien nous a permis d'avancer avec confiance dans la réalisation de ce mémoire.

Nous adressons également nos sincères remerciements aux **membres du jury** qui nous feront l'honneur d'évaluer ce travail.

Enfin, nous remercions chaleureusement nos familles, nos amis et toutes les personnes qui nous ont soutenus, encouragés et motivés tout au long de ce parcours.

Merci à toutes et à tous.

Dédicace

Je dédie ce mémoire à mes parents,
pour leur soutien indéfectible, leur patience,
et les valeurs qu'ils m'ont transmises.

À mon frère **Lounes** et à mes sœurs,
pour leur présence constante, leur bienveillance
et leurs encouragements tout au long de mon parcours.

À toutes les personnes qui, de près ou de loin,
ont contribué à la réalisation de ce travail,
je témoigne ici toute ma reconnaissance.

M. Lynda

Dédicace

Je dédie ce mémoire à Allah (SWT),
source de toute sagesse, de toute patience et de toute force.
Sans Sa guidance et Sa miséricorde, rien de tout cela n'aurait été possible.

À mes parents bien-aimés, pour leur amour inconditionnel, leurs prières silencieuses
et leur soutien constant.

À mes frères et sœurs, mes premiers complices et mes piliers dans les moments
d'épreuve comme dans ceux de réussite.

À ma famille toute entière, pour m'avoir entourée d'amour, de patience et
d'encouragement tout au long de ce parcours.

M. Yasmina

Table des matières

| | |
|---|----------|
| Liste des figures | iv |
| Liste des tableaux | vi |
| Liste des algorithmes | vii |
| Listes des notation et symboles | viii |
| Introduction Générale | 1 |
| 1 Généralités sur les réseaux de capteurs sans fils | 3 |
| 1.1 Introduction | 4 |
| 1.2 Généralités sur les capteurs | 5 |
| 1.2.1 Définition d'un capteur | 5 |
| 1.2.2 Architecture d'un nœud capteur | 5 |
| 1.2.3 Caractéristiques des capteurs | 6 |
| 1.3 Réseaux de capteurs sans fil (RCSFs) | 6 |
| 1.3.1 définition | 6 |
| 1.3.2 Les types de réseaux de capteurs sans fil | 7 |
| 1.3.3 Caractéristiques et contraintes des réseaux de capteurs sans fil | 9 |
| 1.3.4 Architecture des réseaux de capteurs sans fil | 10 |
| 1.3.5 Classification selon le mode d'acquisition des données | 11 |
| 1.3.6 Protocoles de communication dans les réseaux de capteurs sans fil | 12 |
| 1.3.7 Domaines d'application des réseaux de capteurs sans fil | 14 |
| 1.4 Intégration des réseaux de capteurs sans fils dans l'Internet des objets | 14 |
| 1.4.1 Définition de l'Internet des Objets | 14 |
| 1.4.2 Rôle des réseaux de capteurs sans fils dans l'Internet des objets | 15 |
| 1.4.3 Avantage de l'intégration des réseaux de capteurs sans fils dans les systèmes internet des objets | 15 |
| 1.4.4 Exemples d'applications internet des objets basées sur les réseaux de capteurs sans fils | 16 |
| 1.4.5 Architecture de l'internet des objets | 17 |
| 1.5 Défis liés à l'intégration de l'intelligence artificielle dans les réseaux de capteurs sans fils | 19 |
| 1.6 Conclusion | 19 |

| | | |
|----------|---|-----------|
| 2 | Mobilité dans les réseaux de capteurs sans fils | 21 |
| 2.1 | Introduction | 22 |
| 2.2 | Raisons de la mobilité dans les réseaux de capteurs sans fils | 22 |
| 2.3 | Comment atteindre la mobilité dans les réseaux de capteurs sans fils . . . | 23 |
| 2.3.1 | Protocoles de routage adaptés à la mobilité | 23 |
| 2.3.2 | Localisation et suivi des nœuds mobiles | 24 |
| 2.3.3 | Prédiction de la mobilité | 24 |
| 2.3.4 | Les collecteurs des données mobiles | 25 |
| 2.3.5 | Techniques de regroupement adaptatif (clustering) | 25 |
| 2.4 | Types d'entités mobiles | 25 |
| 2.4.1 | Nœuds mobiles | 25 |
| 2.4.2 | Sink mobile | 26 |
| 2.4.3 | Relais Mobile | 26 |
| 2.5 | Défis majeurs de la mobilité | 26 |
| 2.5.1 | Localisation | 26 |
| 2.5.2 | Mobilité du Sink | 26 |
| 2.5.3 | Latence de Transmission | 26 |
| 2.6 | Architectures des réseaux de capteurs sans fils mobiles | 27 |
| 2.6.1 | Architectures a puits fixe et nœuds capteurs fixes | 27 |
| 2.6.2 | Architectures a puits statiques et nœuds capteurs mobiles | 28 |
| 2.6.3 | Architectures a puits mobile et nœuds capteurs fixes | 28 |
| 2.6.4 | Architectures a puits mobile et nœuds capteurs mobiles | 29 |
| 2.7 | Conclusion | 30 |
| 3 | Mobilité du Sink dans les RCSF – État de l'Art | 31 |
| 3.1 | Introduction | 32 |
| 3.2 | Éléments d'une solution de mobilité du collecteur selon notre point de vue | 32 |
| 3.2.1 | Le modèle de mobilité | 32 |
| 3.2.2 | Le protocole de routage | 33 |
| 3.2.3 | La stratégie de collecte de données | 33 |
| 3.3 | Classification des solutions de mobilité du Sink dans les réseaux de capteurs sans fils | 33 |
| 3.3.1 | Classification selon le Modèle de Mobilité | 33 |
| 3.3.2 | Classification selon les protocoles de routages | 35 |
| 3.3.3 | Classification selon l'architecture des solutions proposées | 36 |
| 3.4 | Étude comparative de quelques solutions utilisant un sink mobile dans les réseaux de capteurs sans fils | 37 |
| 3.4.1 | Présentation de quelques solutions utilisant un sink mobile dans les réseaux de capteurs sans fils | 37 |
| 3.4.2 | Comparaison et classification des solutions étudiées | 43 |
| 3.4.3 | Synthèse et discussion | 43 |
| 3.5 | Conclusion | 46 |

| | |
|--|-----------|
| 4 Déplacement intelligent d'un sink mobile dans RCSFs basé sur l'apprentissage par renforcement | 47 |
| 4.1 Introduction | 48 |
| 4.2 Concepts fondamentaux du Q-learning | 48 |
| 4.2.1 Fonction Q et équation de Bellman | 48 |
| 4.2.2 Algorithm du Q-Learning | 49 |
| 4.2.3 Exploration vs Exploitation : Stratégie ϵ -greedy | 49 |
| 4.3 Description des solutions proposées | 50 |
| 4.3.1 Modélisation du réseau | 50 |
| 4.3.2 Description de la première solution : Q-Learning Based sink MOBilty (QLB-MOB) | 52 |
| 4.3.3 Caractéristiques de l'algorithme de routage utilisé | 54 |
| 4.3.4 Stratégie globale de collecte de données | 56 |
| 4.3.5 Description de la deuxième solution : Deep Q-Learning based sink MOBilty (DQL-MOB) | 59 |
| 4.4 Simulation et Résultats | 68 |
| 4.4.1 Environnements et outils de simulation | 69 |
| 4.4.2 Configuration et paramètres de simulation | 69 |
| 4.4.3 Configuration du réseau | 70 |
| 4.4.4 Métriques de performances | 71 |
| 4.4.5 Présentation et discussion des résultats | 73 |
| 4.5 Conclusion | 77 |
| Conclusion Générale | 78 |

Liste des figures

| | | |
|------|---|----|
| 1.1 | Exemple de capteurs[1]. | 5 |
| 1.2 | Architecture d'un nœud capteur. | 5 |
| 1.3 | Schéma de l'architecture RCSF[2]. | 7 |
| 1.4 | RCSF Plat utilisant un mode de transmission multi-saut[3]. | 10 |
| 1.5 | RCSF hiérarchique[4]. | 11 |
| 1.6 | Classification des application des RCSFs. | 11 |
| 1.7 | Exemple d'une application orientée événements dans RCSFs. | 12 |
| 1.8 | La pile protocolaire des RCSF[5]. | 13 |
| 1.9 | Exemples d'applications des RCSF[6]. | 14 |
| 1.10 | Les maisons intelligentes [7] | 16 |
| 1.11 | Les villes intelligentes [8] | 17 |
| 1.12 | L'énergie [9] | 17 |
| 1.13 | Architecture de l'internet des objets[10]. | 18 |
| 1.14 | Défis liés à l'utilisation de l'IA dans les RCSF[11]. | 19 |
| 2.1 | Architecture du système pour le habitat monitoring. | 27 |
| 2.2 | exemple du fonctionnement de CENWITS. | 28 |
| 2.3 | L'architecture à trois niveaux des MULEs[12]. | 29 |
| 2.4 | Architecture de collecte de données dans ZebraNet. | 29 |
| 3.1 | Les éléments d'une solution de mobilité du collecteur dans un réseau de capteurs sans fil. | 32 |
| 3.2 | Les classes de modèles de mobilité. | 33 |
| 3.3 | Modèle de mobilité Aléatoire [13]. | 34 |
| 3.4 | Modèle de mobilité prévisible [13]. | 34 |
| 3.5 | Classification des Protocoles de Routage [14]. | 35 |
| 3.6 | Catégories de protocoles de routage dans les RCSF [15]. | 35 |
| 3.7 | Network structure constructed by CTEER[16]. | 37 |
| 3.8 | WSN, RPs, MS et trajet de déplacement[17]. | 38 |
| 3.9 | Stratégie de collecte de données basée sur les points de rendez-vous[18]. | 39 |
| 3.10 | Structure de la roue virtuelle dans EDVWDD[19]. | 39 |
| 3.11 | Illustration de la transmission de requêtes dans VGRQ, du sink mobile vers la zone d'intérêt [20]. | 40 |
| 3.12 | Modèle de réseau et stratégies de collecte des données[21]. | 41 |
| 3.13 | Architecture du modèle HGFF basé sur un graphe hétérogène et l'apprentissage par renforcement profond [22]. | 42 |

| | | |
|------|--|----|
| 3.14 | Fonctionnement hybride du protocole EHRP combinant transmission directe et multi-saut.[23] | 42 |
| 3.15 | Formation des clusters et sélection des chefs de cluster dans EE-hHSS.[24] | 43 |
| 4.1 | Réseau de capteurs organisé en zones de collecte (État initial). | 52 |
| 4.2 | Organigramme de la stratégie de collecte basée sur QLB-MOB | 58 |
| 4.3 | Modèle de réseau et stratégies de collecte des données. | 60 |
| 4.4 | Organigramme de la stratégie de collecte basée sur DQL-MOB | 67 |
| 4.5 | Énergie Totale Consommée (ETC). | 73 |
| 4.6 | Équilibrage énergétique (Fairness) | 74 |
| 4.7 | Durée de vie du réseau en fonction de la taille du tampon. | 75 |
| 4.8 | Taux de livraison des paquets en fonction de la taille du tampon. | 75 |
| 4.9 | Taux de perte des paquets en fonction de la taille du tampon. | 76 |

Liste des tableaux

| | | |
|-----|---|----|
| 3.1 | Tableau comparatif des caractéristiques de mobilité des solutions à sink mobile | 44 |
| 3.2 | Tableau comparatif des caractéristiques des protocoles de routage pour les solutions à puits mobile | 45 |
| 4.1 | Table des notations | 51 |
| 4.2 | Caractéristiques des modèles de mobilité des approches impémentees. . | 68 |
| 4.3 | Paramètres de simulation | 71 |

Liste des algorithmes

| | | |
|---|--|----|
| 1 | Q-Learning | 49 |
| 2 | Algorithme de Q-learning pour la sélection du prochain point de collecte | 54 |
| 3 | Routage par Dijkstra-Chemin | 55 |
| 4 | Sélection de nœud via DQN avec fonction hybride π_i | 64 |
| 5 | Procédure d’Agrégation et de Transmission des Paquets (DQN) | 65 |
| 6 | Algorithme de routage hybride basé sur la distance | 66 |

Listes des notation et symboles

| | | |
|----------|----------|---|
| A | ADCs | Analog to Digital Converters |
| | AODV | Ad hoc On-Demand Distance Vector |
| | AO | Aquila Optimizer |
| | AACO | Adaptive Ant Colony Optimization |
| | AoIs | Areas of Interest |
| B | BS | Base Station |
| | BLE | Bluetooth Low Energy |
| C | CCR | Clustering by Cluster-based Routing |
| | CUR | Cluster Updating by Rotation |
| | CTEER | Cluster Tree-based Energy-efficient Routing |
| | CH | Cluster Head |
| | CH | Cell Header |
| D | DTN | Delay Tolerant Network |
| | DSQPS | Dynamic Sink-based Query Processing Strategy |
| E | EE-hHHSS | Energy-Efficient hybrid Harris Hawk–Salp Swarm |
| | EHRP | Effective Hybrid Routing Protocol |
| | ECMMS | Enhanced Clustering and ACO-based Multiple Mobile Sinks |
| G | GPS | Global Positioning System |
| | GEAR | Geographic and Energy Aware Routing |
| | GA-MAR | Genetic Algorithm–based Mobile Agent Routing |
| | GNN | Graph Neural Network |
| | GMRE | Greedy Maximum Residual Energy |
| H | HHO | Harris Hawk Optimization |
| | HGFF | Heterogeneous Graph-based Feature Fusion |
| I | IdO | internet des objets |
| | IoGHR | Integration of Geographic and Hierarchical Routing |
| L | LEACH | Low-Energy Adaptive Clustering Hierarchy |
| | LF | Localization Factor |
| M | MANET | Mobile Ad hoc Network |
| | MPR | Multi-Point Relay |
| | MAC | Medium Access Control |
| | MWSN | mobile Wireless Sensor Network |
| | MSDRF | Mobile Sink Data Routing Framework |
| | MS | Mobile Sink |
| | RM | Random Mobility |
| O | OLSR | Optimized Link State Routing |
| | OCNTMS | Optimal Clustering and Network Topology for Mobile Sink |
| | OPT | Optimal Mobility Protocol |
| Q | QoS | Quality of Service |
| | QDWSN | Query-Driven Wireless Sensor Network |
| | QWRP | Query-aware Wireless Routing Protocol |

| | | |
|----------|--------|--|
| | QDVGDD | Query-Driven Virtual Grid-Based Data Dissemination |
| | QRRP | Query-driven Ring Routing Protocol |
| R | RCSF | Réseaux de Capteurs Sans Fil |
| | RFID | Radio Frequency Identification |
| | RREQ | Route Reques |
| | RREP | Route Reply |
| | RSSI | Received Signal Strength Indicator |
| | RP | Rendez-vous Points |
| | RoI | Region of Interest |
| | RRP | Rendezvous-Based Routing Protocol |
| S | SDN | Software Defined Networking |
| | SPIN | Sensor Protocols for Information via Negotiation |
| | SSA | Salp Swarm Algorithm |
| V | VGRQ | Virtual Grid-based Routing for Query-driven |
| W | WSN | Wireless Sensor Network |
| Z | ZRP | Zone Routing Protocol |

Introduction Générale

LES Réseaux de Capteurs Sans Fil (RCSF) représentent une technologie clé dans le domaine des systèmes embarqués et des communications distribuées. Ils sont composés de nœuds capteurs capables de collecter, de traiter localement des données, puis de les transmettre par liaison sans fil. Ces réseaux sont largement utilisés dans de nombreuses applications, notamment la surveillance de l'environnement, la sécurité, la santé, et l'agriculture de précision, grâce à leur déploiement facile, leur coût modéré et leur capacité d'autonomie énergétique.

Dans un RCSF, chaque capteur est généralement alimenté par une source d'énergie restreinte (souvent une batterie non rechargeable), rendant la gestion de l'énergie un véritable défi. La transmission des données représente l'une des tâches les plus énergivores, notamment pour les capteurs proches du sink (la station de base ou collecteur de données), car ils ont la responsabilité de relayer les informations de tous les autres nœuds.

Pour aborder ce défi, une approche efficace consiste à rendre le collecteur de données mobile. En déplaçant intelligemment le sink à travers le réseau, il devient possible de diminuer la pression exercée sur certains capteurs, d'équilibrer la consommation énergétique totale et par conséquent d'allonger la durée de vie du réseau. Cependant, cette méthode pose de nouveaux problèmes complexes, notamment la nécessité d'identifier la trajectoire optimale du sink pour améliorer les performances du réseau. Il est aussi essentiel de définir les critères clés qui orienteront ce mouvement, comme l'énergie restante des nœuds, la charge de trafic ou la distance à parcourir. Enfin, choisir un modèle de mobilité approprié à la dynamique du réseau s'avère crucial pour garantir une efficacité énergétique tout en préservant une qualité de service satisfaisante.

De nombreuses approches ont été proposées dans la littérature, allant des modèles de mobilité à trajectoire fixe aux déplacements aléatoires, en passant par les stratégies adaptatives basées sur des paramètres du réseau. Toutefois, ces approches restent souvent limitées par leur manque de flexibilité ou leur forte dépendance à des hypothèses fixes.

Afin de dépasser ces contraintes, nous avons adopté deux approches fondées sur l'apprentissage par renforcement, en explorant deux techniques complémentaires : le Q-Learning classique et le Deep Q-Learning. Ces deux méthodes permettent au sink mobile d'apprendre, de manière autonome, les meilleures décisions de déplacement à prendre

en fonction de l'état du réseau (charge des nœuds, stabilité des flux, énergie résiduelle, etc.). Dans cette optique, le sink est modélisé comme un agent intelligent qui interagit continuellement avec l'environnement, perçoit des récompenses en retour selon la qualité des décisions prises, et ajuste sa stratégie au fil du temps pour maximiser la durée de vie du réseau et la qualité de service.

Le Q-Learning offre une solution simple et efficace pour des espaces d'état discrets, tandis que le Deep Q-Learning permet de gérer des environnements plus complexes grâce à l'utilisation de réseaux de neurones pour approximer la fonction de valeur. Ces deux approches confèrent au sink une capacité d'adaptation intelligente face aux dynamiques imprévisibles du réseau.

Dans ce mémoire, nous proposons deux modèles de mobilité contrôlée du sink, fondés sur ces deux techniques, visant à améliorer les performances globales dans les réseaux de capteurs sans fil. L'objectif est de concevoir une stratégie de déplacement optimale, capable de réduire la consommation d'énergie, d'éviter les congestions et de maintenir un haut taux de livraison des données, même dans des environnements fortement contraints.

Afin de mener à bien notre travail, nous avons structuré ce mémoire en quatre chapitres. Le premier chapitre présente les notions de base relatives aux réseaux de capteurs sans fil. Le deuxième aborde la question de la mobilité dans ces réseaux, tandis que le troisième est consacré à un état de l'art des modèles de mobilité du sink existants, en mettant en évidence leurs points forts et leurs limites. Le quatrième chapitre, qui constitue la principale contribution de ce travail, est dédié à la proposition et à l'évaluation de deux modèles intelligents de déplacement du sink basés sur l'apprentissage par renforcement. Il commence par une présentation des fondements du Q-Learning, notamment la fonction Q, l'équation de Bellman, l'algorithme d'apprentissage et la stratégie ϵ -greedy. Ensuite, il décrit nos deux solutions : la première, QLB-MOB (Q-Learning Based Mobility), utilise le Q-Learning classique pour guider le déplacement du sink en fonction de l'état du réseau ; la seconde, DQL-MOB (Deep Q-Learning Based Mobility), repose sur le Deep Q-Learning et intègre un réseau neuronal pour améliorer la prise de décision. Ce chapitre inclut également la modélisation du réseau, l'algorithme de routage utilisé, la stratégie de collecte des données, ainsi que les détails de la simulation : environnement, paramètres, métriques de performance et configuration du réseau. Enfin, les résultats obtenus permettent une comparaison des deux approches, montrant que DQL-MOB offre de meilleures performances en termes de fiabilité, de consommation énergétique et de durée de vie du réseau.

Chapitre 1

Généralités sur les réseaux de capteurs sans fils

Sommaire

| | | |
|-------|---|----|
| 1.1 | Introduction | 4 |
| 1.2 | Généralités sur les capteurs | 5 |
| 1.2.1 | Définition d'un capteur | 5 |
| 1.2.2 | Architecture d'un nœud capteur | 5 |
| 1.2.3 | Caractéristiques des capteurs | 6 |
| 1.3 | Réseaux de capteurs sans fil (RCSFs) | 6 |
| 1.3.1 | définition | 6 |
| 1.3.2 | Les types de réseaux de capteurs sans fil | 7 |
| 1.3.3 | Caractéristiques et contraintes des réseaux de capteurs sans fil | 9 |
| 1.3.4 | Architecture des réseaux de capteurs sans fil | 10 |
| 1.3.5 | Classification selon le mode d'acquisition des données | 11 |
| 1.3.6 | Protocoles de communication dans les réseaux de capteurs sans fil | 12 |
| 1.3.7 | Domaines d'application des réseaux de capteurs sans fil | 14 |
| 1.4 | Intégration des réseaux de capteurs sans fils dans l'Internet des objets | 14 |
| 1.4.1 | Définition de l'Internet des Objets | 14 |
| 1.4.2 | Rôle des réseaux de capteurs sans fils dans l'Internet des objets | 15 |
| 1.4.3 | Avantage de l'intégration des réseaux de capteurs sans fils dans les systèmes internet des objets | 15 |
| 1.4.4 | Exemples d'applications internet des objets basées sur les réseaux de capteurs sans fils | 16 |
| 1.4.5 | Architecture de l'internet des objets | 17 |
| 1.5 | Défis liés à l'intégration de l'intelligence artificielle dans les réseaux de capteurs sans fils | 19 |
| 1.6 | Conclusion | 19 |

1.1 Introduction

Grâce aux progrès récents en matière de communication sans fil ,d'électronique numérique et de développement de nœuds captures multifonctionnels ,la réalisation de réseaux de captures sans fil(RCSFs) est devenue possible .Ces réseaux sont constitués de petits nœuds captures auto-organisés, capables de détection ,de traitement de données et de communication.

Les RCSFs sont des systèmes autonomes responsables de la collecte, de la coopération et de la surveillance de quantités physiques et environnementales telles que la température, le son, les vibrations, la pression et les polluants. Chaque nœud capteur dispose d'un niveau d'énergie limité, ce qui rend la conservation de l'énergie et l'équilibrage de la consommation énergétique essentiels pour améliorer la durée de vie du réseau. Des nœuds capteurs statiques ou mobiles sont déployés de manière aléatoire dans la zone de détection pour collecter des données provenant de leur environnement et les transmettre à un nœud puits ou à une station de base (BS).

Initialement, les RCSF étaient destinés à des applications militaires et de défense. Progressivement, ils sont devenus des outils essentiels pour des applications sociétales et civiles dans divers domaines tels que la santé, la gestion des transports et du trafic, la surveillance environnementale et écologique, et l'automatisation industrielle.

Cependant, la mise en œuvre de ces réseaux pose plusieurs défis, notamment en ce qui concerne la gestion de l'énergie, la scalabilité et la fiabilité des données. Parmi ces défis, la mobilité du nœud puits (sink) joue un rôle crucial dans l'amélioration des performances du réseau. En effet, un nœud puits mobile permet de réduire la consommation d'énergie des nœuds capteurs, d'optimiser la collecte des données et d'augmenter la durée de vie du réseau.L'objectif de ce chapitre est de donner d'abord une vue générale sur les réseaux de capteurs sans fils, en commençant d'abord définir les concepts fondamentaux liés aux capteurs, en mettant en évidence leur architecture et leurs principales caractéristiques. Ensuite, nous nous intéresserons aux RCSF en présentant leurs types, leur caractéristique, leur classification, leur architecture, et les protocoles de communication qui assurent leur bon fonctionnement. Enfin, nous verrons comment ces réseaux s'intègrent dans l'Internet des Objets (IoT), en abordant leur rôle, leurs avantages et leur architecture.

1.2 Généralités sur les capteurs

1.2.1 Définition d'un capteur

Un capteur sans fil est un dispositif électronique de petite taille capable de mesurer divers paramètres environnementaux (tels que la température, l'éclairement, la pression, l'humidité, les vibrations, etc.) et d'envoyer ces données à un centre de contrôle via une station de base[25]. Chaque capteur remplit les trois fonctions essentielles suivantes : l'acquisition de données, leur traitement et leur transmission aux stations principales. La Fig. 1.1 montre une liste des capteurs actuellement utilisés.

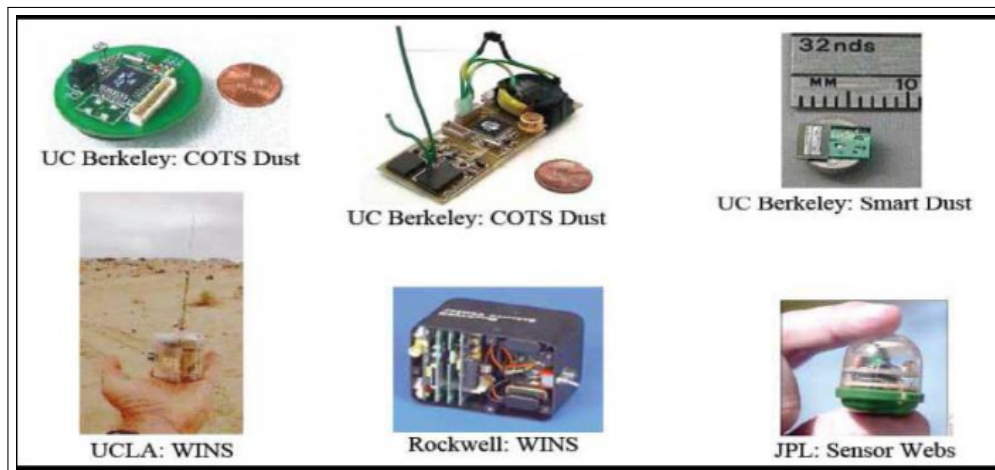


FIG. 1.1 – Exemple de capteurs[1].

1.2.2 Architecture d'un nœud capteur

1.2.2.1 Architecture matérielle

Un nœud capteur est composé de quatre unités principales, qui sont présentées dans la figure 1.2 :

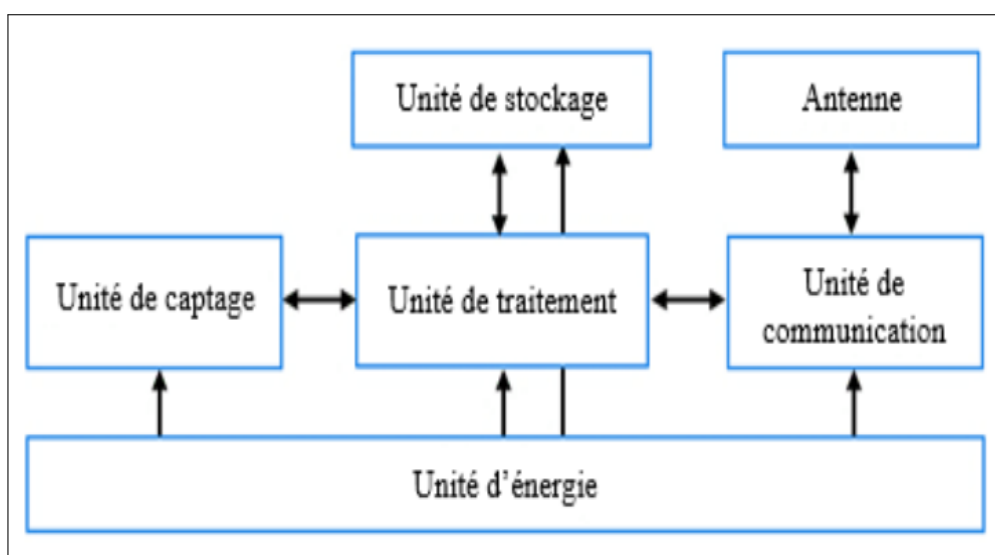


FIG. 1.2 – Architecture d'un nœud capteur.

- **Unité de captage (Sensing unit)** : Généralement composée de deux sous-unités (les capteurs et les convertisseurs analogique-numérique appelé ADC)[26]. Les capteurs recueillent des données numériques concernant les paramètres environnemen-

taux et les transforment en signaux analogiques. Par la suite, les ADCs (Analog to Digital Converters) transforment ces signaux analogiques en données numériques.

- **Unité de traitement (Processing unit)** :Elle est composée d'une mémoire de stockage et d'un processeur. Elle est dotée de deux interfaces : l'une connectée à l'Unité de captage pour recevoir les données collectées, et l'autre connectée à l'unité de communication pour transmettre les données traitées.

Cette unité contrôle le bon fonctionnement des autres parties du capteur. Dans certains cas, elle contient un petit système d'exploitation pour gérer les fonctions du capteur[27].

- **Unité de communication (Communication Unit)** :(Radio, Antenne) elle est chargée de toutes les transmissions via un moyen de communication radio qui connecte le nœud au réseau. Elle a pour rôle de communiquer et d'accueillir les informations. Elle possède un ensemble émetteur/récepteur pour établir une communication au sein du réseau. Toutefois, d'autres méthodes de transmission sont possibles (optique, infrarouge, etc.)[28].
- **Unité d'énergie (Power unit)** :Généralement, c'est la batterie qui n'est ni rechargeable ni interchangeable. Le principal défi lors de l'élaboration de protocoles pour les réseaux de capteurs réside dans la contrainte liée à la capacité énergétique restreinte des capteurs. Les unités mesurant l'énergie Peuvent être alimentées par des cellules photovoltaïques capables de transformer l'énergie lumineuse en électricité[29].

1.2.2.2 Architecture logicielle

Les systèmes d'exploitation destinés aux capteurs doivent démontrer une amélioration en termes d'efficacité, de rapidité, d'utilisation de la mémoire et surtout en ce qui concerne la consommation énergétique. Le système d'exploitation le plus utilisé pour les réseaux de capteurs est TinyOs. TinyOs est un système d'exploitation open source conçu pour les réseaux de capteur sans fil. Ce système d'exploitation est dédié aux capteurs dont la sortie date de 2000. Les applications destinées à TinyOs sont rédigées en NesC, une variante du langage C. Il utilise une programmation orientée événements, ce qui signifie que le système reste en attente jusqu'à ce qu'un événement survienne. Dans TinyOs, chaque élément propose des instructions et répond à des événements.

1.2.3 Caractéristiques des capteurs

Les capteurs doivent satisfaire les caractéristiques suivantes :

- **La Sensibilité** :elle représente la variation du signal de sortie par rapport au signal d'entrée.
- **La Rapidité** :Il s'agit du délai de réponse d'un capteur entre le moment où il détecte un changement dans la grandeur physique qu'il mesure et le moment où cette information est prise en compte par l'unité de commande.
- **La répétition** :Il s'agit de la capacité d'un capteur à présenter la même valeur dans des conditions identiques.

1.3 Réseaux de capteurs sans fil (RCSFs)

1.3.1 définition

Les RCSF se composent d'un grand nombre de nœuds capteurs économiquement efficaces, déployés pour collecter des informations sur l'environnement à surveiller, telles

que la température, le son, les vibrations, la pression, le mouvement ou les polluants [30]. Ces nœuds communiquent de manière sans fil et collaborent pour transmettre leurs informations à une station de base ou sink, qui sert d'interface entre le réseau et les utilisateurs[31]. L'architecture générale d'un réseau de capteurs sans fil est illustrée dans la Fig. 1.3 suivante.

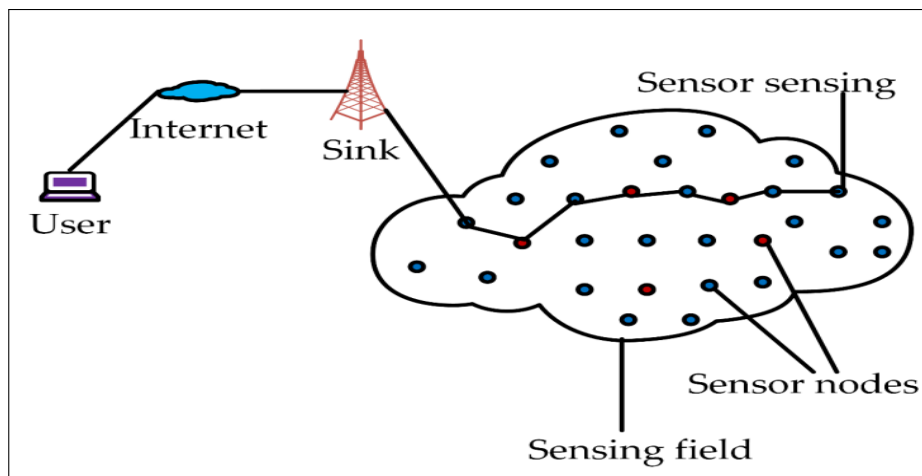


FIG. 1.3 – Schéma de l'architecture RCSF[2].

1.3.2 Les types de réseaux de capteurs sans fil

Selon des critères bien spécifiques, comme la mobilité, l'homogénéité des nœuds du réseau, la nature de l'application et le type des données captées, les RCSFs peuvent être classés en plusieurs classes[32].

1.3.2.1 selon le critère de mobilité

Les nœuds capteurs et la station de base d'un réseau de capteurs sans fil peuvent être fixes ou mobiles. On distingue ainsi les réseaux de capteurs statiques et les réseaux de capteurs mobiles.

- **Les RCSFs statiques** : Dans les réseaux de capteurs statiques, et les nœuds capteurs et la station de base sont stationnaires ; ils gardent leurs positions initiales tout au long de leur durée de vie. Ce type de réseaux de capteurs est bénéfique dans certains types d'applications qui exigent que les capteurs soient placés dans des endroits stratégiques pour les contrôler. Effectivement, ce genre de RCSFs se distingue par une topologie statique, une identification facile des nœuds au sein du réseau et des méthodes de routage relativement simples.
- **Les RCSFs mobiles** : Un RCSF mobile est un réseau de capteurs sans fil constitué de nœuds capables de se déplacer et de s'auto-organiser dans l'environnement. Ces nœuds assurent la collecte, le traitement et la transmission des données tout en s'adaptant dynamiquement aux changements du réseau[33]. Contrairement aux réseaux statiques, ils nécessitent un routage dynamique et doivent relever plusieurs défis comme la gestion de la mobilité, la localisation, le maintien de la couverture et la connectivité, ainsi que l'optimisation de la consommation d'énergie. Les RCSF mobiles sont utilisés dans des domaines tels que la surveillance environnementale, militaire, le suivi de cibles et les missions de secours.

1.3.2.2 Selon le critère d'homogène

Selon ce critère, on distingue deux types de RCSF : les réseaux homogènes et les réseaux hétérogènes.

- **Les RCSFS homogènes** :Le réseau de capteurs est dit homogène si tous les nœuds de capteurs ont équivalents sur le plan capacités et contraintes(faibles ressources et de durée de vie courte). C'est le type qui est commun dans la plupart des applications dans les réseaux de capteurs car il répond au besoin d'autonomie.
- **Les RCSFs hétérogènes** :Dans un réseau de capteurs homogènes, les réseaux de capteurs hétérogènes incluent deux types de nœuds de capteur. Dans ce type de RCSF les nœuds capteurs contraints et les nœuds capteurs puissants non limités en ressources .Dans ce type de RCSFs, les nœuds contraints doivent préserver autant que possible leur réserve énergétique en minimisant les tâches énergétiques les plus chères et les communications radio. Par conséquent, les calculs et les traitements complexes des nœuds puissants sont équilibrés pour compenser les charges et maximiser la durée de vie du réseau. Bien que les RCSFs hétérogènes soient plus avantageux que les RCSFs ordinaires (homogènes), leur adoption est limitée à un nombre réduit d'applications. Cela est dû à la difficulté du déploiement des RCSFs hétérogènes dans des milieux hostiles, isolés ou inaccessibles.

1.3.2.3 Selon le type de l'application

Le processus de captage de données dans un réseau de capteurs sans fil dépend des besoins de l'application et de l'importance des données recueillies. On distingue ainsi deux types de RCSFS : les réseaux temporels (time-driven) et les réseaux événementiels (event-driven).

- **Les réseaux de capteurs temporels** : Un réseau de capteurs temporel est adapté pour des applications nécessitant un prélèvement périodique des données, comme dans les application de monitoring (feu ou météo).Un écoulement en rafale, périodique, du trafic est très susceptible dans ce type d'applications. Par conséquent, des mécanismes de gestion raisonnable des ressources sont primordiaux.
- **Les réseaux de capteurs évènementiels** : Dans certaines applications, les capteurs doivent réagir rapidement à des changements brusques des valeurs captées et donner des réponses immédiates à l'occurrence des évènements. Un prélèvement périodique des données est inadapté pour ce type de scénario.

1.3.2.4 Selon les données captées

Les données collectées par les nœuds d'un réseau de capteurs peuvent être simples ou multimédia. Un nœud capteur peut enregistrer un seul type de données, comme la température, ou plusieurs types simultanément, comme une image, la température et l'humidité.

- **Les réseaux de capteurs multimédia** : Un WMSN est un réseau de nœuds capteurs interconnectés sans fil, équipés de dispositifs multimédias tels que des caméras et des micros, capables de capturer des flux vidéo et audio, des images fixes, ainsi que des données scalaires. Les WMSNs promettent un large éventail d'applications dans les domaines civils et militaires nécessitant des informations visuelles et sonores, telles que les réseaux de surveillance, les rapports des forces de l'ordre, la gestion du trafic, les soins médicaux avancés, l'assistance aux personnes âgées, la télémédecine et le contrôle industriel[34].Les WMSN se distinguent des réseaux de capteurs classiques par plusieurs spécificités :
 - **le déploiement** : les nœuds dans les réseaux de capteurs standards sont souvent déployés aléatoirement. En revanche, dans les réseaux de capteurs multimédia, il est précis et planifié, notamment pour la capture d'images.

- **Qualité de service** : les réseaux de capteurs multimédia nécessitent une bande passante suffisante et une faible latence pour fonctionner efficacement, contrairement aux réseaux de capteurs standards où la qualité de service est relâchée pour un besoin en un moindre coût et une faible dissipation des ressources.
- **La puissance de traitement** : les traitements à effectuer sur les données scalaires sont faibles. Néanmoins, pour le cas des données multimédia, les nœuds capteurs effectuent des traitements intensifs ce qui demande plus de performance matérielle.
- **Consommation d'énergie** : Étant donné que la qualité de service et les traitements intensifs consomment beaucoup d'énergie, les mécanismes de gestion de la consommation énergétique dans les réseaux de capteurs multimédia doivent être particulièrement efficaces. Le remplacement des batteries des nœuds reste souvent difficile, voire impossible, selon la zone de déploiement.
- **Les réseaux de capteurs standards** : Il s'agit des RCSFs ordinaires où les données collectées sont de type scalaire, comme par exemple l'humidité, la température, la pression, etc. les RCSFs de tel type partagent les caractéristiques déjà mentionnées.
- **Les réseaux de capteurs multimodaux** : Un nœud capteur dans un RCSF multimodal peut collecter plusieurs types d'informations, qu'elles soient scalaires ou multimédia. il peut mesurer la température tout en capturant une image. Ainsi, un seul nœud capteur multimodal peut remplacer tout un groupe de capteurs ordinaires. Ceci est particulièrement utile lorsque l'on souhaite obtenir plusieurs informations environnementales sur un même lieu d'intérêt.

1.3.3 Caractéristiques et contraintes des réseaux de capteurs sans fil

Il existe plusieurs caractéristiques qui distinguent les RCSFs des autres réseaux sans fil, les plus importantes sont :

- **Sécurité limitée** : La sécurité est un problème majeur des RCSF, car ils sont souvent déployés dans des lieux hostiles, ce qui risque de détruire les nœuds[35]. Les capteurs ont des ressources limitées, donc il faut proposer des protocoles de sécurité économes en énergie et en traitement.
- **Absence d'infrastructure** : Généralement, les RCSF se différencient des autres réseaux mobiles par le fait qu'ils ne reposent sur aucune infrastructure existante et ne sont soumis à aucune forme de gestion centralisée. Il s'applique aux hôtes mobiles de garantir et de maintenir toujours la connexion au réseau[35].
- **Topologie dynamique** : La topologie des réseaux de capteurs évolue de manière fréquente et rapide en raison de : la possibilité de déploiement des nœuds capteurs dans des environnements difficiles (comme un champ de bataille), rendant une défaillance d'un nœud capteur très probable[1]. Par ailleurs, les nœuds capteurs ainsi que les nœuds destinataires peuvent être mobiles.
- **Scalabilité** : Contrairement aux réseaux sans fil traditionnels, un RCSF peut contenir un très grand nombre de nœuds capteurs. Un réseau de capteurs est Scalable car il a la capacité d'accueillir de nombreux nœuds qui travaillent ensemble vers un but commun[36].
- **Tolérance aux pannes** : Si un ou plusieurs capteurs tombent en panne à cause d'une pénurie d'énergie, d'un problème physique ou d'une perturbation, les fonctions du réseau doivent être conservées sans interruptions[36].

- **Coût de production** : Le coût des RCSF, qui sont constitués d'un grand nombre de nœuds, dépend de celui d'un seul nœud qui ne doit pas, par conséquent, être cher.

1.3.4 Architecture des réseaux de capteurs sans fil

1.3.4.1 Les réseaux de capteurs sans fil plats

Un réseau de capteurs sans fil plat est un réseau homogène, où tous les nœuds sont identiques en ce qui concerne la batterie et la complexité du matériel, à l'exception du Sink qui agit en tant que passerelle et qui est chargé de transmettre les données recueillies à l'utilisateur final. En fonction du service et des types de capteurs, une densité importante de capteurs (un grand nombre de nœuds par mètre carré) ainsi qu'une communication à plusieurs sauts peuvent s'avérer nécessaires pour une architecture plate. Avec un nombre excessif de capteurs, la capacité à évoluer devient cruciale. La gestion et l'organisation des nœuds doivent être effectuées de manière extrêmement efficace en termes d'énergie par le biais du routage et du contrôle d'accès au médium (MAC)[37].

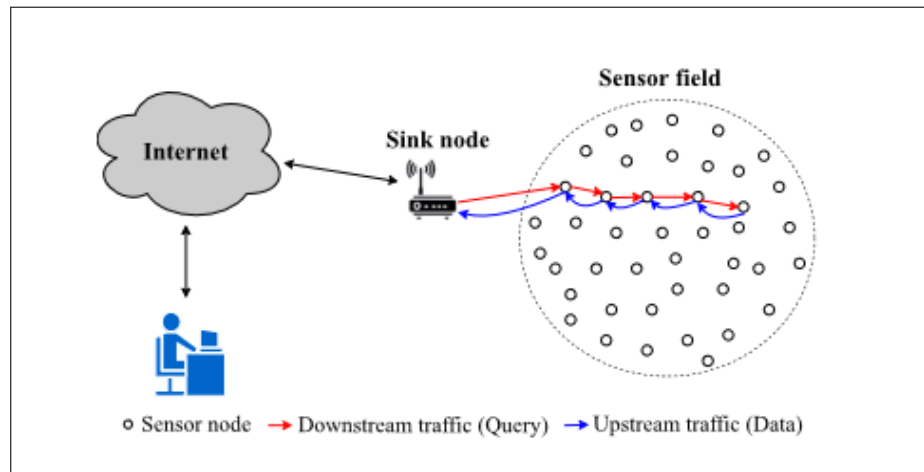


FIG. 1.4 – RCSF Plat utilisant un mode de transmission multi-saut[3].

1.3.4.2 Les réseaux de capteurs sans fil hiérarchiques

On proposait une organisation hiérarchique pour réduire le coût et la complexité de la plupart des nœuds capteurs en intégrant un groupe de nœuds capteurs plus onéreux et plus performant, ceci en mettant en place une structure qui soulage la majorité des nœuds basique à faible coût de nombreuses fonctions du réseau. L'architecture en plusieurs niveaux comprend : une couche de détection, une couche de transmission des données et une couche de point d'accès[37].

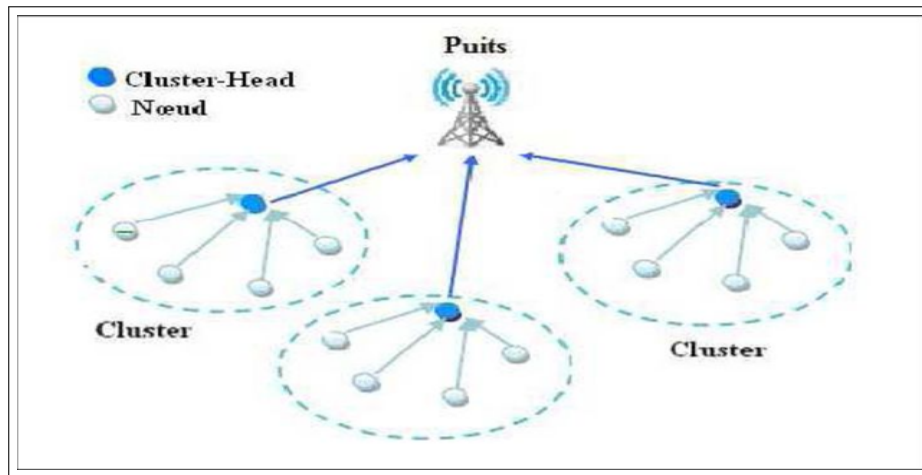


FIG. 1.5 – RCSF hiérarchique[4].

1.3.5 Classification selon le mode d'acquisition des données

Les RCSF peuvent être classés selon la manière dont ils collectent les données. Cette classification aide à mieux comprendre comment les capteurs fonctionnent dans différents types d'applications. On distingue généralement quatre types : orientées événements, orientées temps, orientées requêtes et hybrides.

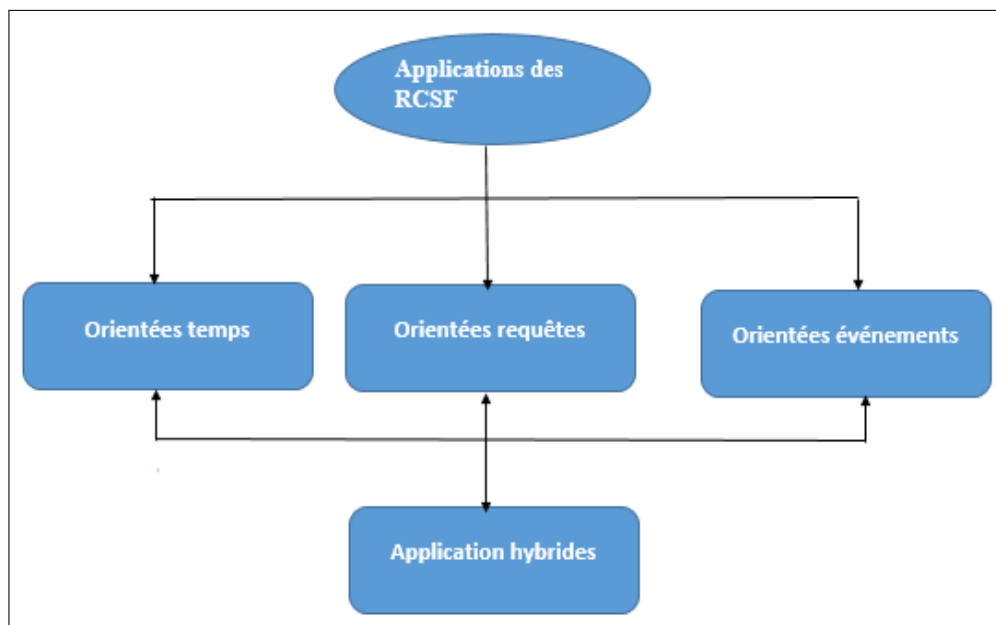


FIG. 1.6 – Classification des application des RCSFs.

1.3.5.1 Applications orientées événements

Dans ce type d'applications, les capteurs envoient les données à la station de base seulement si un événement spécial se produit. Nous pouvons citer l'exemple de surveillance des feux dans les forêts où un capteur envoie des alarmes à la station de base dès que la température dépasse un certain seuil[38].

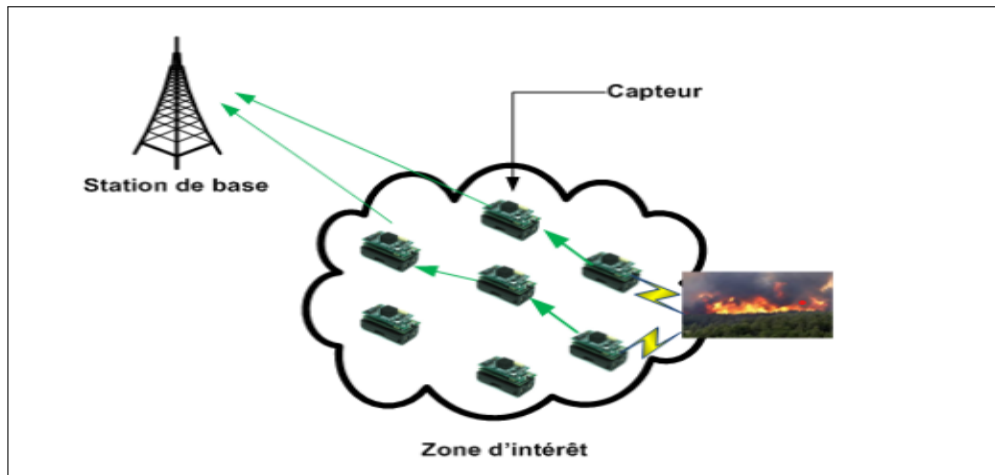


FIG. 1.7 – Exemple d’une application orientée événements dans RCSFs.

1.3.5.2 Applications orientées requêtes

Dans ce contexte, un capteur transmet des informations uniquement en réponse à une requête explicite de la part de la station de base. Cette classe d’application est conçue pour les applications adaptées pour l’utilisateur. Ce dernier peut demander des données provenant de certaines zones du réseau ou consulter les capteurs pour obtenir des mesures pertinentes. Dans ce contexte, il est essentiel de comprendre la topologie du réseau et la position des capteurs[38].

1.3.5.3 Applications orientées temps

Il s’agit d’applications où l’obtention et la diffusion des données capturées sont associées au temps : moment exact, durée de collecte. La durée d’acquisition peut varier en fonction de l’application, allant de quelques secondes à plusieurs heures, voire même des jours. Par conséquent, le volume de données qui circule sur le réseau est fonction de la fréquence des observations à effectuer dans l’environnement[38].

1.3.5.4 Applications hybrides

Le type d’application hybride est une combinaison entre les trois types d’applications précédemment décrites. Par exemple, dans un réseau conçu pour le suivi d’objets, le réseau peut combiner entre un réseau de surveillance et un réseau de collecte de données par événements.

1.3.6 Protocoles de communication dans les réseaux de capteurs sans fil

1.3.6.1 Pile protocolaire

Une architecture en couches est adoptée, dans le but d’améliorer la robustesse du réseau de capteurs sans fil et assurer certaines fonctionnalités. Chaque couche garantit de façon autonome une portion des fonctions requises pour la communication entre les entités du réseau. Elle exploite le service des niveaux inférieurs et en distribue à celui de niveau supérieur.

Ce modèle se compose de cinq strates (couche application, couche transport, couche réseau, couche liaison de données et couche physique) qui correspondent aux mêmes fonctions que celles du modèle OSI, en plus de trois niveaux : un niveau de gestion d’énergie, un niveau de gestion de mobilité et un niveau de gestion des tâches. La Fig. 1.8

illustre la pile de protocoles employée par la station de base et tous les autres capteurs du réseau[39].

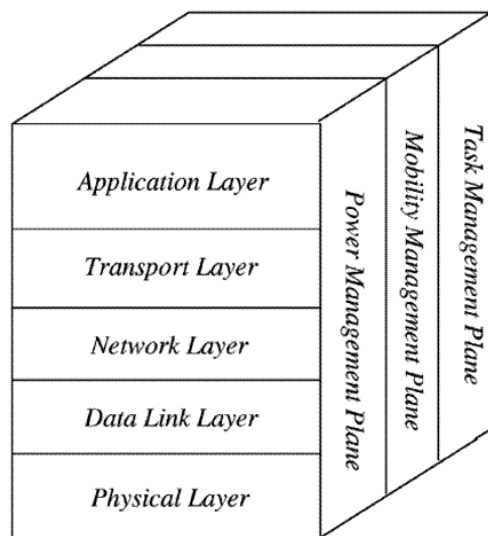


FIG. 1.8 – La pile protocolaire des RCSF[5].

Différents types d'applications peuvent être construites selon la tâche assignée aux capteurs. La couche application constitue le niveau le plus proche de l'architecture, elle assure l'interface avec les applications et représente le point de contact direct avec l'utilisateur. La couche transport aide au maintien du flux de données si l'application du RCSF le nécessite. La couche réseau quant à elle, est responsable du routage des données fournies par la couche transport dans le RCSF. La couche liaison gère l'accès au support physique grâce au protocole MAC (Contrôle d'Accès au Support). Cette couche illustre comment les données sont transférées d'un nœud à un autre. Elle a pour responsable le multiplexage des flux de données, l'identification du cadre de données, la vérification d'erreurs et l'accès au support. La couche physique s'occupe de la sélection de fréquence, de la génération de la fréquence porteuse, de la détection du signal, de la modulation/démodulation et du cryptage/décryptage des données. L'énergie consommée à ce niveau peut varier selon le schéma de modulation/démodulation ou la bande de fréquence utilisée.

Les fonctions intégrées au plan de gestion de l'énergie permettent de contrôler l'utilisation de la batterie. Par exemple suite à la réception d'un message, le capteur désactive son récepteur pour prévenir la redondance des messages déjà captés. Par ailleurs, lorsque le niveau d'énergie s'affaiblit, le nœud envoie une notification à ses voisins pour les informer de son incapacité à contribuer au routage. Le reste de l'énergie est destiné à la capture[39].

La gestion de la mobilité suit et enregistre les mouvements des capteurs, permettant de garder une route vers l'utilisateur final et d'avoir une image récente des nœuds voisins. Cette image est essentielle pour équilibrer l'exécution des tâches et la consommation d'énergie[39].

Le niveau de gestion des tâches assure l'équilibrage et la distribution des tâches sur les différents nœuds du réseau, dans le but de favoriser une collaboration efficace en termes de consommation d'énergie, ce qui contribue à prolonger la durée de vie du réseau[39].

1.3.7 Domaines d'application des réseaux de capteurs sans fil

Le domaine d'applications des RCSF est très varié. Ces réseaux sont présents (fig.1.9) dans le domaine environnemental, médicale, militaire, commerciales, sécurité,...etc.

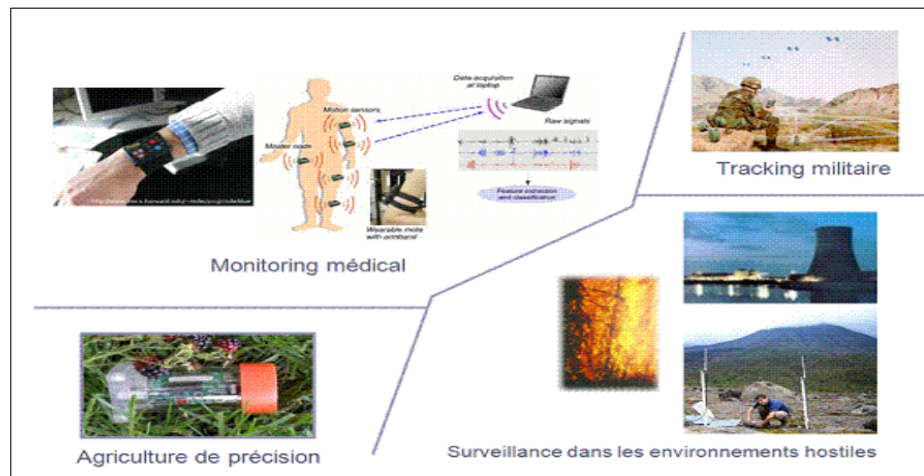


FIG. 1.9 – Exemples d'applications des RCSF[6].

- **Application environnementale** : Dans ce secteur, on utilise fréquemment les réseaux de capteurs pour détecter des incendies, surveiller des phénomènes naturels, identifier des pollutions et observer des écosystèmes[40].
- **Applications médicales** : Les réseaux de capteurs ont également diverses applications dans le secteur du diagnostic médical. Par exemple, des micro-caméras peuvent, sans intervention chirurgicale, transmettre des images de l'intérieur du corps humain avec une autonomie de 24 heures[6].
- **Applications militaires** : Les RCSF, dans Domaines militaires, permettent de surveiller les mouvements ennemis, de détecter les attaques chimiques ou biologiques, et d'aider à la prise de décision stratégique. Ils assurent une collecte d'informations en temps réel tout en limitant les risques pour les soldats[41].
- **Applications commerciales** : Dans ce domaine, on peut citer diverses applications telles que : le suivi de l'état des équipements, le pilotage et l'automatisation des procédés d'usinage, entre autres.
- **Applications sécurité** : comme la détection et le suivi d'incendies de forêts, d'avalanches, d'ouragans ou de risque de panne d'équipement à grande échelle.

1.4 Intégration des réseaux de capteurs sans fils dans l'Internet des objets

1.4.1 Définition de l'Internet des Objets

L'IdO est une infrastructure réseau mondiale dynamique dotée de capacités d'auto-configuration, basée sur des protocoles de communication standardisés et interopérables, où les objets physiques et virtuels possèdent des identités, des attributs physiques et des personnalités virtuelles, utilisent des interfaces intelligentes, et sont intégrés de manière transparente dans le réseau d'information. Ces objets échangent fréquemment des données liées aux utilisateurs et à leurs environnements[42].

1.4.2 Rôle des réseaux de capteurs sans fils dans l'Internet des objets

Dans IdO, les RCSF occupent une place cruciale en tant que base pour la collecte de données, la communication et le monitoring en direct. Les RCSF se composent de capteurs disposés spatialement qui observent des paramètres physiques ou environnementaux, comme la température, l'humidité, la pression ou le mouvement, et transmettent ces informations à un système central via une connexion sans fil pour traitement et analyse.

- **Acquisition de données environnementales :** Les capteurs embarqués dans les nœuds RCSF capturent en continu des informations issues de leur environnement. Ces données, une fois numérisées, servent de matière première pour les applications IdO, facilitant ainsi des décisions automatisées et contextuelles.[42].
- **Communication et Connectivité :** Les RCSF permettent une communication fluide entre les nœuds de capteurs et les systèmes IdO. Ils utilisent des protocoles de communication sans fil tels que Zigbee¹, LoRa² ou Bluetooth Low Energy (BLE)³ pour transmettre les données à des passerelles ou à des plateformes cloud. Cette connectivité assure que les informations recueillies par les capteurs peuvent être traitées et analysées ultérieurement[42].
- **Surveillance et Contrôle en Temps Réel :** Les RCSF permettent une surveillance en direct des environnements, ce qui est essentiel pour des applications comme les résidences intelligentes, l'automatisation de l'industrie et le suivi environnemental. En proposant des flux de données ininterrompus, les RCSF permettent aux systèmes IdO de détecter des anomalies, de déclencher des alertes et d'automatiser des réponses[42].
- **Interopérabilité :** Les RCSF utilisent souvent des protocoles de communication standardisés, ce qui leur permet de s'intégrer de manière transparente avec d'autres appareils et systèmes IdO. Cette interopérabilité est essentielle pour créer des écosystèmes IdO cohérents où diverses plateformes et dispositifs peuvent collaborer de manière efficace[42].

1.4.3 Avantage de l'intégration des réseaux de capteurs sans fils dans les systèmes internet des objets

- **Connectivité et couverture améliorées :** Les RCSF peuvent être déployés dans des environnements variés, y compris des zones difficiles d'accès ou à risque. Cette polyvalence les rend particulièrement utiles pour des applications telles que l'agriculture de précision, la surveillance écologique ou la gestion des infrastructures urbaines[43].
- **Évolutivité et modularité :** Grâce à leur nature évolutive, les RCSF s'adaptent parfaitement aux besoins croissants des projets IdO. Ils permettent d'ajouter des capteurs supplémentaires pour améliorer la précision des données ou étendre la couverture géographique[43].
- **Optimisation des coûts :** Avec l'avancée des technologies de communication sans fil et des capteurs, les RCSF sont devenus économiques tout en offrant des

¹Zigbee est un protocole de communication sans fil à faible consommation d'énergie, conçu pour l'automatisation domestique, le contrôle industriel et les applications IdO.

²LoRa (Long Range) est un protocole de communication sans fil à longue portée et faible consommation d'énergie, utilisé principalement pour les réseaux IdO, permettant la transmission de petites quantités de données sur de grandes distances.

³Le Bluetooth Low Energy (BLE) est une technologie de communication sans fil à faible consommation d'énergie, utilisée principalement pour les applications IdO, les dispositifs portables et la communication à courte portée.

fonctionnalités avancées. cela en fait une solution abordable pour de nombreuses applications IdO[43].

- **Analyse en temps réel des données :** Les RCSF permettent une acquisition et analyse rapide des données, essentielles pour des applications comme la gestion du trafic, la surveillance médicale ou l'automatisation industrielle[43].

1.4.4 Exemples d'applications internet des objets basées sur les réseaux de capteurs sans fils

- **Les maisons intelligentes (Smart Homes) :** La maison du futur sera entièrement connectée à Internet, offrant aux propriétaires un accès et un contrôle à distance via leurs smartphones, tablettes ou ordinateurs. ce qui améliore le confort, la sécurité et l'efficacité énergétique.

Tous les équipements du foyer (de la porte d'entrée à la télévision, en passant par le thermostat, le réfrigérateur et même les objets du quotidien comme les montres ou les parapluies et les systèmes de détection d'intrusion qui alertent les utilisateurs en cas de mouvement suspect) seront intégrés dans cet écosystème intelligent. Les détecteurs de fumée et de gaz intelligents envoient des alertes lors d'un mouvement suspect, alors que les systèmes de contrôle centralisés permettent de gérer tous ces dispositifs via une interface unique[42].

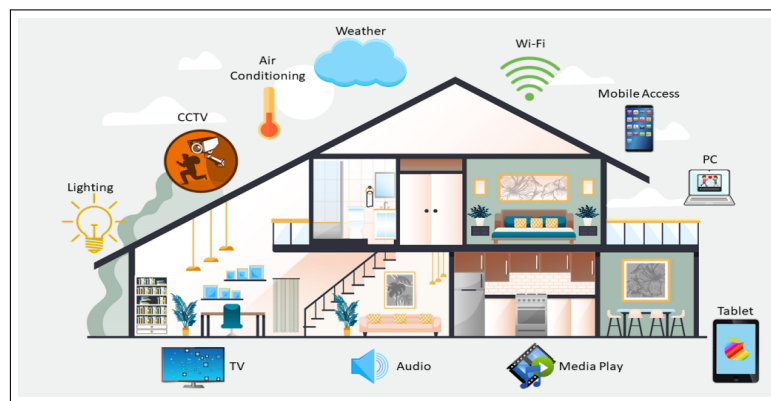


FIG. 1.10 – Les maisons intelligentes [7]

- **Les villes intelligentes (Smart Cities) :** L'IdO ne se limite pas aux maisons intelligentes ; il s'étend aux routes, bâtiments, véhicules, magasins et parkings, qui communiqueront entre eux pour améliorer la gestion urbaine et le quotidien des citoyens. Le trafic routier sera optimisé, réduisant les embouteillages et fournissant aux automobilistes des informations en temps réel sur les parkings disponibles, les hôtels, restaurants, hôpitaux et autres points d'intérêt. Les autorités disposeront d'outils avancés pour surveiller la pollution, gérer l'éclairage public adaptatif et collecter des données environnementales précises, telles que la température, le taux d'humidité et les niveaux de radiation. Au-delà des villes, ces technologies bénéficieront également à des secteurs comme l'agriculture de précision, où l'accès ubiquitaire aux informations des objets connectés permettra d'automatiser le contrôle et d'optimiser les rendements. La mise en place d'une ville intelligente reposera sur une coexistence massive de multiples technologies, notamment les réseaux de capteurs, transformant ainsi notre manière de vivre et d'interagir avec notre environnement[42].

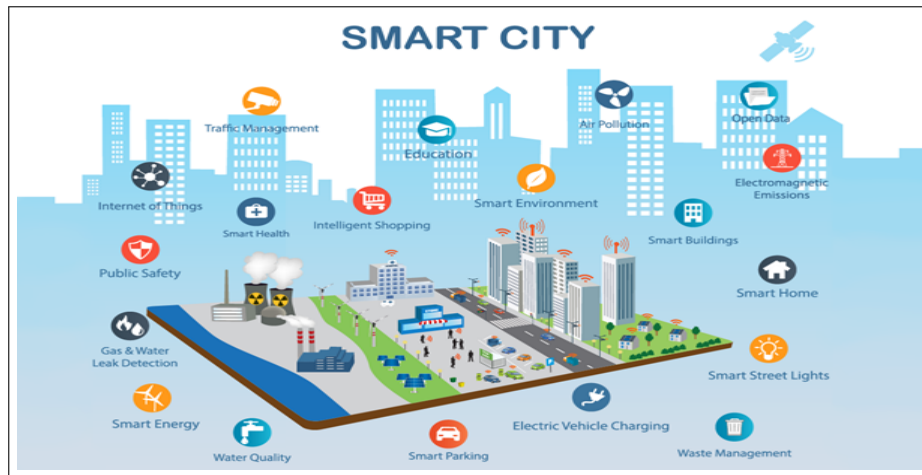


FIG. 1.11 – Les villes intelligentes [8]

- L'énergie :** L'IdO transforme le secteur de l'énergie en permettant une gestion optimisée des réseaux électriques intelligents (Smart Grids). Ces réseaux collectent et analysent des données en temps réel sur la production, la distribution et la consommation d'électricité, améliorant par conséquent l'action énergétique, réduisant les pannes et prévenant les vols d'électricité grâce à des compteurs intelligents. L'intégration des énergies renouvelables (Renewable Energy Systems), à cause l'éolien et le solaire, est facilitée par des capteurs IdO qui mesurent les variables électriques (telles que la tension et le courant) et régulent la production pour maintenir la stabilité du réseau, notamment en gérant les flux d'énergie bidirectionnels. Enfin, les systèmes de "Prognostics" surveillent en temps réel l'état des équipements critiques (comme les turbines éoliennes ou les transformateurs) grâce à des capteurs IdO, permettant de augurer les défaillances, de établir la incessant proactive et d'essayer la secret et la clause des systèmes énergétiques. Ces technologies IdO contribuent ainsi à une gestion plus durable et résiliente des ressources énergétiques[42].

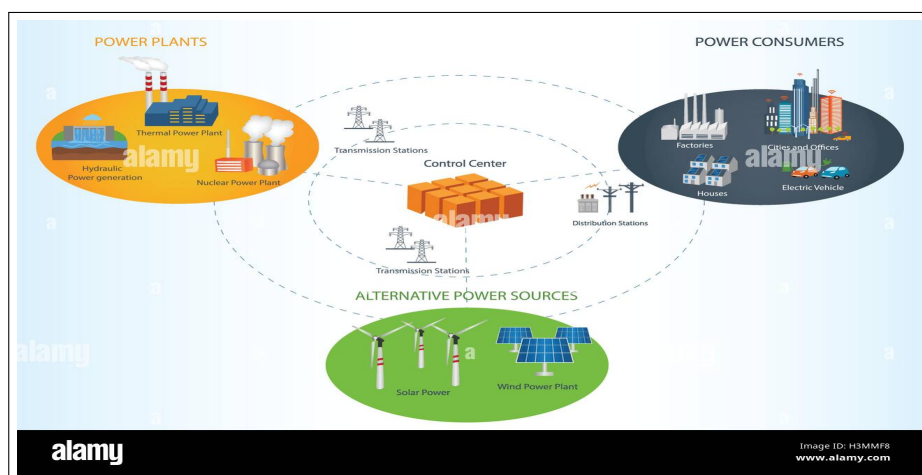


FIG. 1.12 – L'énergie [9]

1.4.5 Architecture de l'internet des objets

Du point de vue de l'architecture, on peut considérer que l' IdO est structuré en trois niveaux majeurs : la couche de collecte de données, la couche réseau et enfin la couche des applications. L'organisation est illustrée dans la figure 1.13[32].

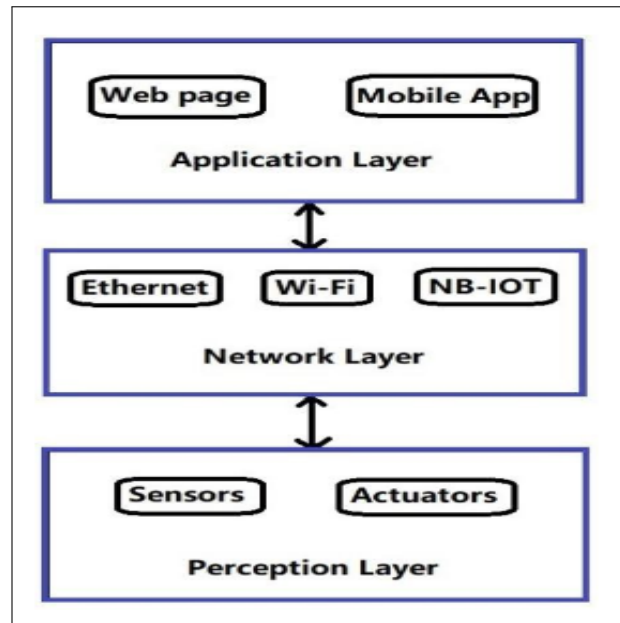


FIG. 1.13 – Architecture de l'internet des objets[10].

1.4.5.1 La couche perception

La couche de perception, qui occupe une position inférieure dans la hiérarchie, est chargée de recueillir des données et de les reconnaître dans leur contexte. Cette couche intègre donc l'équipement nécessaire pour recueillir des données contextuelles à partir des objets connectés, tels que les capteurs, les étiquettes de radio-identification RFID⁴, les caméras, le GPS (Système de Positionnement Global)[44], etc.

1.4.5.2 La couche réseau

Cette couche est responsable de la transmission fiable des données de la couche perception et de la connectivité entre les objets connectés et les autres hôtes de l'Internet. Avec l'augmentation du nombre d'objets connectés, les données générées deviennent énormes, rendant essentiel le stockage et le traitement à faible coût sur Internet. Les services cloud permettent une gestion flexible des ressources de stockage et de traitement, en utilisant le concept de SDN⁵ pour améliorer la gestion des équipements réseau et réduire la latence. Cela permet également l'automatisation de la gestion des serveurs sur le cloud.

1.4.5.3 La couche application

Elle utilise les données traitées par la couche précédente et constitue la partie frontale de l'architecture, permettant d'exploiter tout le potentiel de ces données. Elle a également pour rôle de mettre à disposition des développeurs les outils nécessaires pour concrétiser la vision et déployer toutes les applications possibles de l'Internet des Objets (IdO)[44].

⁴RFID (Radio Frequency Identification) est une technologie utilisée pour l'identification et le suivi sans fil d'objets ou de personnes à l'aide d'ondes radio. Elle se compose d'une étiquette (contenant une puce et une antenne) et d'un lecteur.

⁵Le SDN (Réseau Définie par Logiciel) est une approche de gestion des réseaux qui sépare le plan de contrôle du plan de données, permettant une gestion centralisée et plus flexible des ressources réseau.

1.5 Défis liés à l'intégration de l'intelligence artificielle dans les réseaux de capteurs sans fils

L'intelligence artificielle (IA) occupe une place importante dans les systèmes modernes, notamment dans les RCSF. Elle permet une prise de décision intelligente, une adaptation dynamique et une exploitation optimisée des données collectées. Cependant, l'intégration de ces technologies dans les RCSF soulève plusieurs défis majeurs, tant sur le plan technique qu'opérationnel, comme illustré à la Fig.1.14.

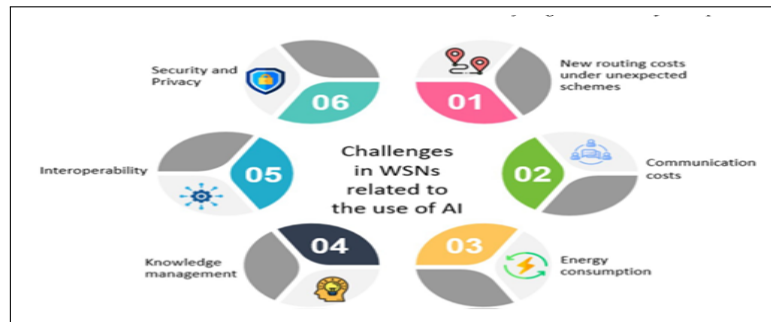


FIG. 1.14 – Défis liés à l'utilisation de l'IA dans les RCSF[11].

- **Nouveaux coûts de routage sous des schémas imprévus** : Lorsque des événements inattendus surviennent, les réseaux doivent adapter leurs schémas de routage. L'IA autorise une auto-configuration des noeuds, mais cela peut provoquer une surcharge en messages et conduire à des problèmes de connectivité.
- **Surcharge de communication** : Les techniques d'apprentissage automatique nécessitent fréquemment plusieurs cycles de communication pour observer l'environnement et apprendre de celui-ci. Cela peut provoquer une surcharge du réseau, surtout dans des environnements complexes.
- **Consommation d'énergie** : L'exécution de modèles d'IA consomme des ressources énergétiques importantes. Il faut donc trouver un équilibre entre la précision du modèle et l'économie d'énergie, en privilégiant des solutions centralisées lorsque cela est possible.
- **Gestion des connaissances** : Les algorithmes doivent pouvoir être capables d'utiliser les connaissances passées pour faire face à des situations futures. Cependant, en l'absence de données historiques suffisantes, les modèles d'IA peuvent être inefficaces dans des environnements hautement dynamiques.
- **Interopérabilité** : Les capteurs intelligents proviennent souvent de différents fabricants avec des protocoles variés. Cela pose des défis d'interopérabilité pour assurer une communication fluide.
- **Sécurité et confidentialité** : Les systèmes intelligents sont plus exposés aux attaques. La protection des données sensibles et personnelles devient une priorité, surtout lorsque des capteurs collectent des informations critiques.

1.6 Conclusion

Dans ce chapitre, nous avons introduit les bases des RCSF en commençant par une présentation des capteurs, leur architecture, ainsi que leurs principales caractéristiques. Ensuite, nous avons donné une vue d'ensemble des RCSF, en abordant leurs différents types, leurs architectures, leurs contraintes, ainsi que les protocoles de communication qui

leur sont associés. Nous avons également présenté les domaines d'application des RCSF dans divers secteurs.

Nous avons poursuivi avec une section dédiée à l'intégration des RCSF dans l'IdO, en mettant en évidence leur rôle, les avantages qu'ils apportent à ces systèmes, ainsi que des exemples d'applications concrètes. Enfin, nous avons conclu ce chapitre par une discussion sur les défis liés à l'intégration de l'intelligence artificielle dans les RCSF, soulignant les opportunités et les enjeux à relever pour rendre ces réseaux plus intelligents, autonomes et performants.

Chapitre 2

Mobilité dans les réseaux de capteurs sans fils

Sommaire

| | | |
|-------|--|----|
| 2.1 | Introduction | 22 |
| 2.2 | Raisons de la mobilité dans les réseaux de capteurs sans fils | 22 |
| 2.3 | Comment atteindre la mobilité dans les réseaux de capteurs sans fils | 23 |
| 2.3.1 | Protocoles de routage adaptés à la mobilité | 23 |
| 2.3.2 | Localisation et suivi des nœuds mobiles | 24 |
| 2.3.3 | Prédiction de la mobilité | 24 |
| 2.3.4 | Les collecteurs des données mobiles | 25 |
| 2.3.5 | Techniques de regroupement adaptatif (clustering) | 25 |
| 2.4 | Types d'entités mobiles | 25 |
| 2.4.1 | Nœuds mobiles | 25 |
| 2.4.2 | Sink mobile | 26 |
| 2.4.3 | Relais Mobile | 26 |
| 2.5 | Défis majeurs de la mobilité | 26 |
| 2.5.1 | Localisation | 26 |
| 2.5.2 | Mobilité du Sink | 26 |
| 2.5.3 | Latence de Transmission | 26 |
| 2.6 | Architectures des réseaux de capteurs sans fils mobiles | 27 |
| 2.6.1 | Architectures a puits fixe et nœuds capteurs fixes | 27 |
| 2.6.2 | Architectures a puits statiques et nœuds capteurs mobiles | 28 |
| 2.6.3 | Architectures a puits mobile et nœuds capteurs fixes | 28 |
| 2.6.4 | Architectures a puits mobile et nœuds capteurs mobiles | 29 |
| 2.7 | Conclusion | 30 |

2.1 Introduction

Au cours des dernières décennies, l'intérêt pour les technologies de communication des réseaux de capteurs sans fil (RCSFs) a connu une croissance rapide. Un tel réseau est composé d'un ou plusieurs puits (sinks) distants et de nombreux nœuds capteurs de petite taille et à faible consommation d'énergie, chacun étant équipé d'actionneurs, de dispositifs de détection, et d'un émetteur-récepteur sans fil. Ces nœuds sont déployés massivement dans une zone d'intérêt afin de collecter des informations sur leur environnement et de les transmettre en continu aux puits distants. Les nœuds capteurs sont généralement alimentés par batterie, ce qui les rend susceptibles aux pannes. À mesure que certains nœuds cessent de fonctionner en raison de l'épuisement de leur énergie, des zones non couvertes peuvent apparaître et la connectivité du réseau peut être perturbée. Dans de nombreux cas, il est difficile de recharger ou de remplacer ces nœuds défectueux. L'introduction de la mobilité dans les RCSFs peut améliorer leur flexibilité et leur capacité à gérer plusieurs missions tout en résolvant ces problèmes. La mobilité dans les RCSFs désigne la capacité des nœuds à changer de position après leur déploiement initial. Grâce à l'introduction de nœuds ou de puits mobiles, les RCSF mobiles (MWSN) offrent une solution prometteuse pour gérer la congestion et optimiser la topologie. Des recherches ont démontré que ces réseaux surpassent les RCSFs statiques en termes de durée de vie, de consommation énergétique, de capacité de canal et de connectivité. Cependant, la mobilité engendre de nouveaux défis, notamment des changements fréquents de topologie, surtout en cas de mobilité du puits. Les messages de contrôle nécessaires pour la découverte et la mise à jour des routes génèrent un trafic important, pouvant provoquer des retards et des pertes de données. De plus, les nœuds mobiles doivent transmettre des données en mouvement, ce qui complique le routage et nécessite des protocoles spécifiques adaptés à cette situation. Ce chapitre suivant s'attache à explorer les principales raisons d'intégrer la mobilité dans ces réseaux, tout en étudiant les mécanismes tels que les protocoles de routage, la localisation, la prédiction des déplacements et les techniques de collecte mobile de données. Une attention particulière est portée aux différentes entités mobiles (qu'il s'agisse de capteurs, de relais ou de puits de données) ainsi qu'aux défis complexes que pose leur intégration, notamment en termes de latence, de gestion de l'énergie et de localisation. Enfin, une analyse des diverses architectures possibles des réseaux mobiles permettra de mettre en évidence les implications structurelles de chaque choix de mobilité.

2.2 Raisons de la mobilité dans les réseaux de capteurs sans fils

Les modèles traditionnels de RCSFs reposent sur l'hypothèse que le réseau est dense, de telle sorte que deux nœuds peuvent communiquer entre eux à travers une communication multi-sauts. Par conséquent, dans la plupart des cas les capteurs sont considérés comme statiques et la mobilité n'est pas considérée comme une option. De façon similaire et plus récente, la tendance de la recherche dans les réseaux mobiles Ad-hoc (MANET)[45] et les réseaux tolérants au retard (DTNs)[46], la mobilité a également été introduite dans les RCSFs. En fait, la mobilité dans les RCSFs est utile pour plusieurs raisons :

- **Amélioration de la connectivité** : Etant donné que les nœuds sont mobiles, une structure de RCSFs très dense pourrait ne pas être nécessaire. Effectivement, les composants mobiles ont la capacité de relier des zones éloignées, ce qui permet d'assouplir les exigences de connectivité du réseau, notamment en ce qui concerne le repositionnement des nœuds. Ainsi, l'architecture RCSF décentralisée devient une possibilité réaliste [47].

- **Réduction des coûts** : En diminuant le nombre de nœuds dans un RCSF mobile, on peut réduire les coûts du réseau bien que l'ajout de fonctionnalités de mobilité aux nœuds peut être coûteux. Dans plusieurs situations, il est possible d'exploiter des éléments mobiles qui sont déjà disponibles dans la région (par exemple : trains, bus, navettes, ou les voitures), et leurs attacher des capteurs [47].
- **Amélioration de la fiabilité** : Depuis que les RCSFs traditionnels sont caractérisés par leur densité et que le modèle de communication est fréquemment basé sur des sauts multiples, la fiabilité se trouve menacée par les interférences et les collisions. De plus, l'augmentation du nombre de sauts entraîne une hausse dans la perte de messages. Les éléments mobiles visitent les nœuds dans le réseau et récupèrent les données directement via des transmissions à un seul saut. Ceci réduit non seulement les conflits et les collisions, mais aussi la perte de messages [47].
- **Amélioration du rendement énergétique** : le modèle de trafic inhérent aux RCSFs est le convergent. En d'autres termes, les messages sont créés par les nœuds capteurs et sont ensuite collectés par le sink. Par conséquent, les nœuds les plus proches du sink sont surchargés par rapport aux autres, ce qui provoque une décharge prématurée de leur énergie. Ce problème est connu sous le nom de trou énergétique (energy hole) qui résulte directement de l'effet Funneling (effet d'entonnoir), où les voisins du sink deviennent un goulot d'étranglement. Les composants mobiles (ex. sinks mobiles) atténuent cet effet en redistribuant dynamiquement la charge énergétique, même dans des architectures RCSF densifiées [48].

2.3 Comment atteindre la mobilité dans les réseaux de capteurs sans fils

Pour atteindre une mobilité efficace, plusieurs mécanismes et stratégies sont utilisés dans la conception des protocoles et de l'architecture des RCSFs.

2.3.1 Protocoles de routage adaptés à la mobilité

Le routage est l'un des aspects les plus sensibles à la mobilité. Les protocoles conçus pour des réseaux statiques deviennent rapidement inefficaces lorsque les nœuds sont en mouvement. On distingue principalement trois catégories :

2.3.1.1 Les protocoles proactifs (table-driven)

Protocoles où chaque nœud maintient en permanence une table de routage complète, mise à jour périodiquement, contenant des chemins vers tous les autres nœuds du réseau, même en l'absence de communication. Idéaux pour les réseaux stables et petits (ex : surveillance industrielle) [49].

Comme exemple de protocoles de cette catégorie nous citons :

- **OLSR (Optimized Link State Routing)** : est un protocole proactif qui est basé sur le principe de l'état de liens, Il réduit la surcharge en utilisant des relais multipoints ou **MPR**(Multi-Point Relay) où chaque nœud choisit un sous-ensemble de ses voisins qu'il désignera comme MPR pour retransmettre ses paquets en cas de diffusion [50].

2.3.1.2 Les protocoles réactifs (on-demand)

Protocoles qui découvrent les routes uniquement à la demande via des requêtes(RREQ) et des réponses (RREP). Adaptés aux réseaux mobiles ou énergétique-

ment contraints (ex : les capteurs sur les robots , véhicules, êtres humains ou animaux) [49].

Parmi les protocoles de cette classe nous citons :

- **AODV(Ad hoc On-Demand Distance Vector)** : présente des caractéristiques telles qu’une topologie dynamique, sujette à des changements constants. Ce processus repose sur l’envoi d’un grand nombre de paquets pour découvrir les routes possibles, ce qui illustre l’un des objectifs clés du protocole : optimiser la bande passante disponible tout en réduisant le flux de messages liés à la mise à jour des routes. Cette méthode de découverte permet à chaque nœud mobile du réseau d’agir comme un routeur spécialisé. [51].

2.3.1.3 Les protocoles hybrides :

combinent les deux approches, souvent en utilisant une stratégie proactive à l’intérieur des clusters et une stratégie réactive entre eux, ce qui les rend adaptés à des environnements mobiles.

- **ZRP (Zone Routing Protocol)** : est un protocole de routage hybride conçu pour les réseaux mobiles ad hoc. Il combine les avantages des protocoles **proactifs** (dans une zone locale appelée zone de routage) et **réactifs** (pour la communication entre zones). Chaque nœud maintient de manière proactive les routes dans sa propre zone, tandis que le routage vers les destinations situées hors de cette zone est effectué à la demande [52].

2.3.2 Localisation et suivi des nœuds mobiles

La localisation est essentielle pour que les nœuds mobiles connaissent leur position, ce qui est crucial pour le routage et la gestion du réseau. Le **GPS**¹ offre une précision élevée, mais il est coûteux, énergivore et inefficace en environnement intérieur [53] [54]. Une alternative consiste à utiliser des méthodes de localisation fondées sur la puissance du signal reçu, notamment le **RSSI**² (Received Signal Strength Indicator). Le RSSI permet d’estimer la distance entre deux nœuds en mesurant la puissance du signal reçu. Bien qu’il soit peu gourmand en énergie, il reste sensible aux interférences et aux variations de l’environnement [54]. Ces méthodes peuvent être combinées à des algorithmes de suivi pour limiter les mises à jour fréquentes et améliorer l’efficacité globale du réseau.

2.3.3 Prédiction de la mobilité

La prédiction de mobilité vise à anticiper la position future d’un nœud mobile sur la base de son historique, sa vitesse ou sa direction. Des outils comme le **filtre de Kalman**³, les **chaînes de Markov**⁴ ou des modèles basés sur l’**apprentissage automatique**⁵ sont utilisés pour modéliser les mouvements, cela permet de préparer les changements de route à l’avance, réduisant ainsi les pertes de paquets et les coûts de communication [55].

¹GPS (Global Positioning System) est un système de localisation par satellite qui permet de déterminer la position géographique d’un récepteur avec une précision élevée.

²RSSI (Received Signal Strength Indicator) est un indicateur qui mesure la puissance du signal reçu par un récepteur, permettant ainsi d’estimer la distance entre deux nœuds.

³Le filtre de Kalman est un algorithme récursif utilisé pour estimer l’état d’un système dynamique à partir de mesures bruitées. Il est couramment utilisé en traitement du signal et en estimation de trajectoire.

⁴Les chaînes de Markov sont des modèles probabilistes utilisés pour représenter des systèmes qui évoluent d’un état à un autre selon des probabilités fixes. Elles sont particulièrement utiles pour modéliser des processus stochastiques.

⁵L’apprentissage automatique est un sous-ensemble de l’intelligence artificielle qui permet aux ordinateurs d’apprendre à partir de données sans être explicitement programmés pour chaque tâche.

2.3.4 Les collecteurs des données mobiles

Dans RCSF mobiles, la collecte des données doit s'adapter aux connexions intermittentes et à la mobilité des nœuds. Plusieurs méthodes sont utilisées pour surmonter ces défis. **Les collecteurs mobiles (data mules)** sont des dispositifs mobiles tels que des robots, des drones ou des véhicules qui parcourent le réseau pour recueillir les données [56] [57]. **Le routage opportuniste** consiste à transmettre les données lorsqu'un nœud passe à proximité [58], tandis que les techniques de **ferrying** optimisent les trajets des collecteurs pour réduire la latence et la consommation énergétique. Ces approches sont particulièrement adaptées aux environnements où la connectivité continue n'est pas garantie [59].

2.3.5 Techniques de regroupement adaptatif (clustering)

Le clustering permet d'améliorer la scalabilité et l'efficacité énergétique des RCSF. Dans un environnement mobile, ces regroupements doivent être dynamiques pour s'adapter aux variations de la mobilité. **Le clustering adaptatif à la mobilité (MAC)** sélectionne les chefs de cluster en fonction de critères tels que la stabilité du nœud, sa vitesse relative ou sa position future. Des **algorithmes de regroupement dynamique** permettent d'ajuster les groupes en temps réel en fonction de la mobilité des nœuds ; et **les approches hybrides** combinent coordination centralisée et décisions locales et offrent un compromis efficace entre performance et coût [60] [61].

2.4 Types d'entités mobiles

2.4.1 Nœuds mobiles

Les nœuds mobiles dans RCSF peuvent être classés en deux grandes catégories : ceux qui sont spécifiquement conçus comme faisant partie de l'infrastructure du réseau, et ceux qui appartiennent à l'environnement dans lequel le réseau est déployé. Dans le premier cas, leur mobilité est généralement totalement contrôlable, car ces nœuds sont souvent robotisés. En revanche, les nœuds issus de l'environnement peuvent être difficiles à contrôler, dont la mobilité peut être difficile à prédire. Certains suivent un emploi du temps fixe (prédictible), d'autres ont un comportement aléatoire, tandis que certains adoptent un schéma de mobilité partiellement prévisible, comme un bus en ville. Ces schémas peuvent être appris et estimés avec le temps [62].

On distingue principalement deux types de nœuds mobiles :

- **Pairs Mobiles (Mobile Peers)** : Les pairs mobiles sont des nœuds capteurs ordinaires qui possèdent la capacité de se déplacer librement dans le réseau. Ils peuvent jouer à la fois le rôle de source (émetteur de données) et de relais (transmetteur de données d'autres nœuds). Contrairement aux relais ou puits mobiles qui ont un rôle spécifique, les pairs mobiles sont symétriques dans leurs interactions et peuvent transmettre des données lorsqu'ils passent à proximité d'une station de base ou d'un autre nœud. Ils sont typiquement utilisés dans des réseaux sparse (à faible densité) où la mobilité contribue activement à la collecte des données [47].
- **Nœuds Relocalisables (Relocatable Nodes)** : Les nœuds relocalisables sont des capteurs mobiles capables de changer de position dans un RCSF afin d'améliorer la couverture ou la connectivité du réseau. Contrairement aux nœuds collecteurs de données, ils ne transportent pas d'informations pendant leurs déplacements. Une fois leur nouvelle position atteinte, ils restent généralement stationnaires et servent

à relayer les données via des communications multi-sauts. Ils sont souvent utilisés pour gérer la topologie du réseau ou combler des trous de couverture[47].

2.4.2 Sink mobile

Ce sont des nœuds mobiles utilisés comme points de collecte des données dans RCSF mobiles (WSN-MEs). Contrairement aux sinks statiques des RCSF traditionnels, qui sont limités par les contraintes de ressources des capteurs (mémoire, traitement, communication, énergie, etc.), les puits mobiles permettent une collecte plus efficace des données grâce à leur capacité de se déplacer dans la zone d'intérêt. Ce déplacement peut être assuré par un humain, un animal, un véhicule ou un robot. En plus de consommer les données localement, ces puits peuvent aussi les transmettre à distance via une connexion Internet sans fil à longue portée[47].

2.4.3 Relais Mobile

Ce sont des nœuds mobiles utilisés dans RCSF pour faciliter le transfert des données entre les capteurs (nœuds sources) et les stations de base (ou sinks). Contrairement aux nœuds terminaux, ils n'ont pas pour rôle de consommer les données, mais servent uniquement d'intermédiaires mobiles en les collectant, les stockant temporairement, puis en les transportant jusqu'à un point de collecte lorsqu'ils se trouvent à proximité[63][47].

2.5 Défis majeurs de la mobilité

2.5.1 Localisation

Dans les réseaux déployés de manière statique, la position des nœuds peut être déterminée une fois lors de l'initialisation. Cependant, les nœuds mobiles doivent continuellement obtenir leur position lorsqu'ils traversent la région de détection. Cela nécessite du temps et de l'énergie supplémentaires, ainsi que la disponibilité d'un service de localisation rapide[64].

2.5.2 Mobilité du Sink

Dans les applications RCSF centralisées, les données des capteurs sont transmises à une station de base, où elles peuvent être traitées à l'aide de méthodes nécessitant des ressources importantes. L'acheminement des données et leur agrégation peuvent entraîner des frais généraux significatifs. Certains MWSNs utilisent des stations de base mobiles, qui traversent la région de détection pour collecter les données, ou se positionnent de manière à minimiser le nombre de sauts de transmission nécessaires pour les nœuds capteurs[64].

2.5.3 Latence de Transmission

Les déplacements des nœuds peuvent allonger le chemin que les paquets doivent parcourir pour atteindre le puits (Sink), ou même créer des zones temporairement inaccessibles. Cette augmentation de la latence compromet certaines applications critiques comme la surveillance en temps réel ou le suivi médical. De plus, les interruptions temporaires de certains liens peuvent entraîner la perte de données ou des retards dans la collecte des informations. Bien que des stratégies comme la prédiction des déplacements des nœuds ou l'utilisation de collecteurs mobiles puissent contribuer à réduire ce problème, elles nécessitent des ressources de calcul supplémentaires[65].

2.6 Architectures des réseaux de capteurs sans fils mobiles

2.6.1 Architectures a puits fixe et nœuds capteurs fixes

Dans cette architecture traditionnelle, le puits et les nœuds capteurs sont tous deux statiques. Les capteurs collectent des données environnementales et les transmettent soit directement, soit à travers des nœuds relais, vers le puits fixe. Cependant, cette architecture peut entraîner une consommation d'énergie déséquilibrée, notamment pour les nœuds proches du puits, ce qui peut réduire la durée de vie du réseau[66][67]. comme type d'application environnementale utilisant ce type d'architecture, nous citons le projet suivant :

Projet Habitat Monitoring : Ce projet se concentre sur l'utilisation des RCSF pour la surveillance des habitats naturels, en particulier dans des environnements sensibles tels que les îles et les colonies d'oiseaux marins. L'objectif est de surveiller en temps réel des paramètres environnementaux comme la température, l'humidité et le mouvement, tout en minimisant l'impact humain sur ces habitats. Les capteurs sont déployés dans des zones difficiles d'accès pour collecter des données sans perturber les espèces étudiées. Pour assurer cette surveillance efficace et discrète, le système repose sur une architecture en couches intégrant des capteurs autonomes, des passerelles de communication, une station de base distante, et une interface utilisateur destinée aux chercheurs. Cette architecture est représentée dans la Figure 2.1[68].

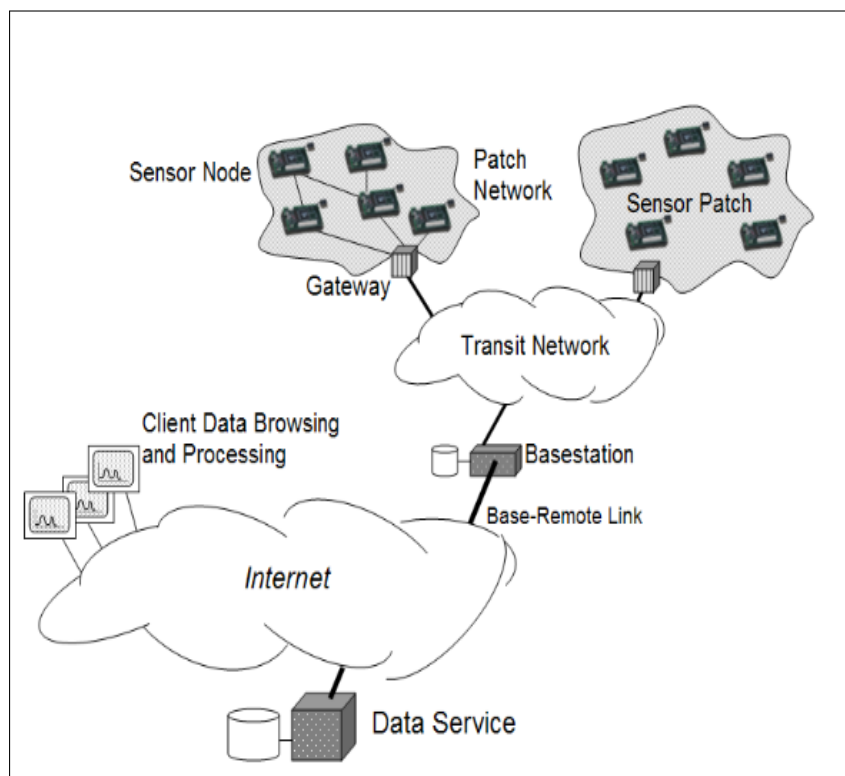


FIG. 2.1 – Architecture du système pour le habitat monitoring.

2.6.2 Architectures a puits statiques et nœuds capteurs mobiles

Cette mobilité des nœuds capteurs permet une couverture de captage de la zone surveillée, améliorant ainsi la collecte de données dans des environnements changeants. Comme exemple de projets utilisant ce type d'architecture où les nœuds capteurs bougent tandis que le puits est fixe, nous citons le projet CENWITS [69].

Projet CENWITS : CENWITS est un système de suivi et de recherche conçu pour fonctionner dans des environnements difficiles. Il repose sur une architecture composée de plusieurs éléments clés : des capteurs mobiles et in-situ portés par des sujets (personnes, animaux sauvages ou objets inanimés), et de points d'accès (AP) qui collectent les informations provenant de ces capteurs, et de récepteurs GPS ainsi que de points de localisation (LP) qui fournissent des données de position aux capteurs. Un exemple du fonctionnement de CENWITS est illustré dans les Figures 2.2.

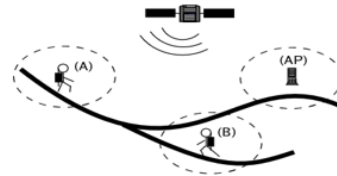


Figure 1: Hiker A and Hiker B are not in the range of each other

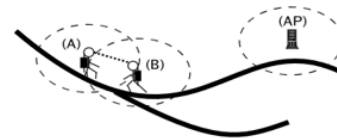


Figure 2: Hiker A and Hiker B are in the range of each other. A records the presence of B and B records the presence of A. A and B become each other's witnesses.

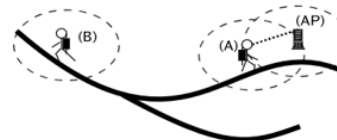


Figure 3: Hiker A is in the range of an access point. It uploads its recorded witness information and clears its memory.

FIG. 2.2 – exemple du fonctionnement de CENWITS.

2.6.3 Architectures a puits mobile et nœuds capteurs fixes

Ce type d'architecture repose sur l'utilisation d'un ou plusieurs puits mobiles, tandis que les capteurs restent fixes dans l'environnement surveillé. Cette solution représente une approche efficace pour équilibrer la consommation énergétique dans le réseau, en évitant la surcharge des nœuds proches d'un puits statique. Le déplacement du puits, contrôlé ou aléatoire, permet une collecte plus équitable des données, prolongeant ainsi la durée de vie globale du réseau[70]. les agents mobiles sont souvent désignés sous le terme de MULEs (Mobile Ubiquitous LAN Extensions) et peuvent prendre la forme de robots, de véhicules, d'animaux ou même d'agents humains. Lorsqu'un MULE se trouve à proximité d'un capteur, il récupère les données, les stocke temporairement, puis les transfère ultérieurement vers des points d'accès câblés. Ce mécanisme permet de réaliser d'importantes économies d'énergie au niveau des capteurs[12].

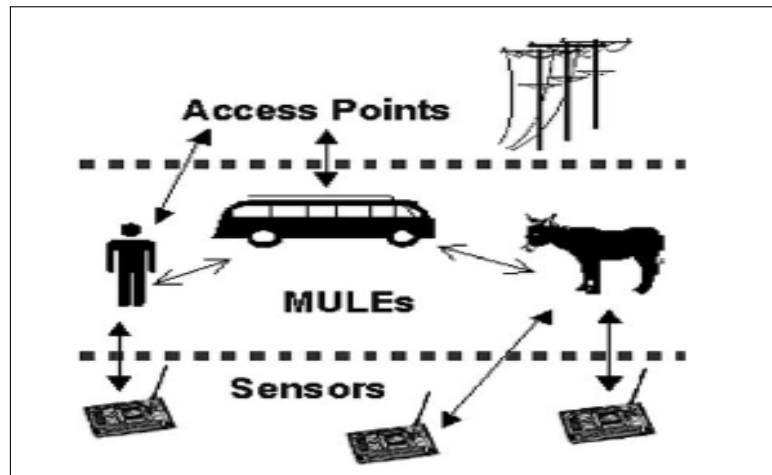


FIG. 2.3 – L’architecture à trois niveaux des MULEs[12].

La Figure 2.3 illustre cette abstraction à trois niveaux qui peut être ajustée à différents types de situations et besoins de distribution :

- Un niveau supérieur composé de dispositifs connectés au WAN.
- Un niveau intermédiaire d’agents de transport mobiles.
- Un niveau inférieur constitué de nœuds fixes de capteurs sans fil.

2.6.4 Architectures a puits mobile et nœuds capteurs mobiles

Cette architecture repose sur la mobilité conjointe du puits collecteur de données et des nœuds capteurs déployés dans la zone d’intérêt. Elle offre une flexibilité maximale pour la collecte d’informations dans des environnements dynamiques et vastes, où la topologie du réseau peut évoluer fréquemment. comme type d’application environnementale utilisant ce type d’architecture, nous trouvons le projet de suivi des zèbres(ZebraNet)

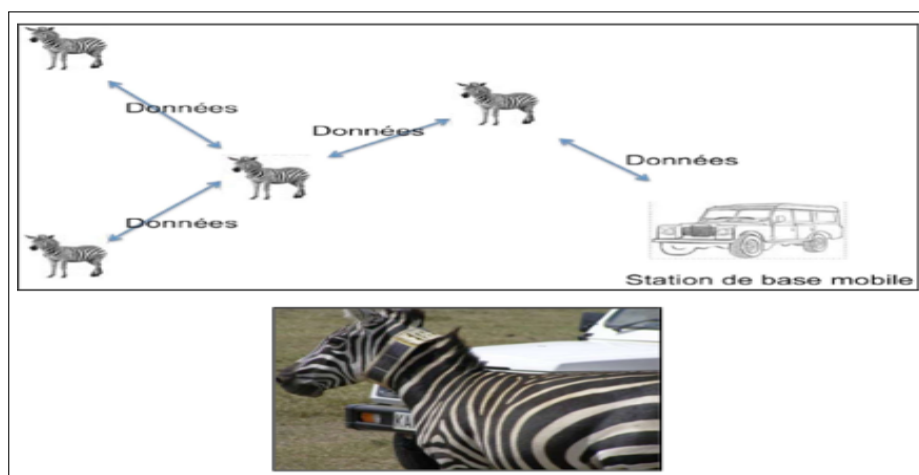


FIG. 2.4 – Architecture de collecte de données dans ZebraNet.

Projet ZebraNet : Il s’agit d’une collaboration directe et continue entre des chercheurs en systèmes informatiques expérimentaux et en biologie de la faune. Ce projet figure parmi les premières études dans le domaine de la surveillance environnementale à intégrer simultanément la mobilité des nœuds du réseau et de la station de base. Dans ce cadre, les zèbres, équipés de récepteurs GPS, servent de nœuds mobiles, tout comme la station de base. Les chercheurs peuvent ainsi collecter des données lorsque leur véhicule ou avion passe à proximité des zèbres, comme le montre la figure 2.4[71].

Les biologistes définissent les objectifs de conception globaux du dispositif de suivi comme suit :

- Échantillons de position GPS prélevés toutes les trois minutes.
- Journaux d'activité détaillés enregistrés pendant 3 minutes chaque heure.
- une année de fonctionnement du réseau sans aucune intervention humaine direct.
- le système doit être fonctionner sur une grande surface de plusieurs milliers de km².
- Aucune station de base fixe, antenne ou service cellulaire.
- Bien que la latence ne soit pas critique, un taux de réussite élevé pour la livraison éventuelle de toutes les données enregistrées est important.
- une taille du collier GPS 1,5 et 2,3 kg.

2.7 Conclusion

La mobilité dans RCSF représente une avancée essentielle pour surmonter les contraintes inhérentes aux architectures statiques. Elle permet au réseau de mieux s'adapter aux environnements dynamiques, tout en renforçant la couverture, la longévité du système, l'efficacité énergétique et la qualité globale du service. Néanmoins, cette souplesse introduit également de nouveaux enjeux techniques, tels que la gestion de la localisation, le routage adaptatif, les délais de transmission et la fiabilité des échanges.

Ce chapitre a étudié la mobilité dans RCSFs, nous avons d'abord mis en évidence les principales raisons qui justifient l'introduction de la mobilité dans les RCSFs. Nous avons ensuite exploré les différentes approches permettant de mettre en œuvre cette mobilité, notamment à travers des protocoles de routage adaptés, des techniques de localisation et de suivi, la prédiction de la mobilité, l'utilisation de collecteurs mobiles et les techniques de regroupement adaptatif. Nous avons également distingué les divers types d'entités mobiles (nœuds, relais et puits) et analysé les défis majeurs liés à leur intégration dans le réseau. Enfin, une étude des différentes architectures possibles a permis de mieux comprendre les implications structurelles de la mobilité selon le type de déploiement choisi. Ce cadre théorique offre les fondations nécessaires à l'étude approfondie de la mobilité optimisée du sink, cœur de notre problématique de recherche.

Chapitre 3

Mobilité du Sink dans les RCSF – État de l’Art

Sommaire

| | | |
|-------|---|----|
| 3.1 | Introduction | 32 |
| 3.2 | Éléments d’une solution de mobilité du collecteur selon notre point de vue | 32 |
| 3.2.1 | Le modèle de mobilité | 32 |
| 3.2.2 | Le protocole de routage | 33 |
| 3.2.3 | La stratégie de collecte de données | 33 |
| 3.3 | Classification des solutions de mobilité du Sink dans les réseaux de capteurs sans fils | 33 |
| 3.3.1 | Classification selon le Modèle de Mobilité | 33 |
| 3.3.2 | Classification selon les protocoles de routages | 35 |
| 3.3.3 | Classification selon l’architecture des solutions proposées | 36 |
| 3.4 | Étude comparative de quelques solutions utilisant un sink mobile dans les réseaux de capteurs sans fils | 37 |
| 3.4.1 | Présentation de quelques solutions utilisant un sink mobile dans les réseaux de capteurs sans fils | 37 |
| 3.4.2 | Comparaison et classification des solutions étudiées | 43 |
| 3.4.3 | Synthèse et discussion | 43 |
| 3.5 | Conclusion | 46 |

3.1 Introduction

Après avoir présenté les concepts de base des RCSF et leur extension mobile dans les chapitres précédents, ce chapitre se concentre sur un aspect central de ces réseaux : la mobilité du collecteur de données (Sink).

Le Sink mobile facilite le rapprochement temporaire du collecteur aux capteurs, minimisant ainsi les distances de communication, réduisant la consommation d'énergie des nœuds et diminuant les pertes de paquets causées par les sauts multiples. Elle contribue également à diminuer la surcharge sur les nœuds relais proches du Sink.

Cet état de l'art propose une synthèse des principales approches de mobilité du Sink dans les RCSF. Initialement, nous présentons les éléments constitutifs d'une solution de mobilité du collecteur, à savoir le modèle de mobilité, les protocoles de routage et les stratégies de collecte de données. Par la suite, une classification des solutions en place est réalisée en fonction de divers critères techniques. Enfin, une étude comparative de certaines solutions intégrant un puits mobile permet de mettre en évidence les avantages et les limites des différentes approches.

3.2 Éléments d'une solution de mobilité du collecteur selon notre point de vue

Une solution de mobilité du collecteur dans un RCSF repose sur la coordination de plusieurs éléments clés. Ces éléments interagissent pour permettre une collecte de données efficace, fiable en énergie, tout en s'adaptant à la dynamique du réseau. À notre avis, une solution de ce type, illustrée à la figure 3.1, repose sur trois parties principales qui doivent être bien conçues pour garantir les performances du réseau. Dans cette section, nous présentons ces trois parties essentielles : le modèle de mobilité, le protocole de routage et la stratégie de collecte des données.

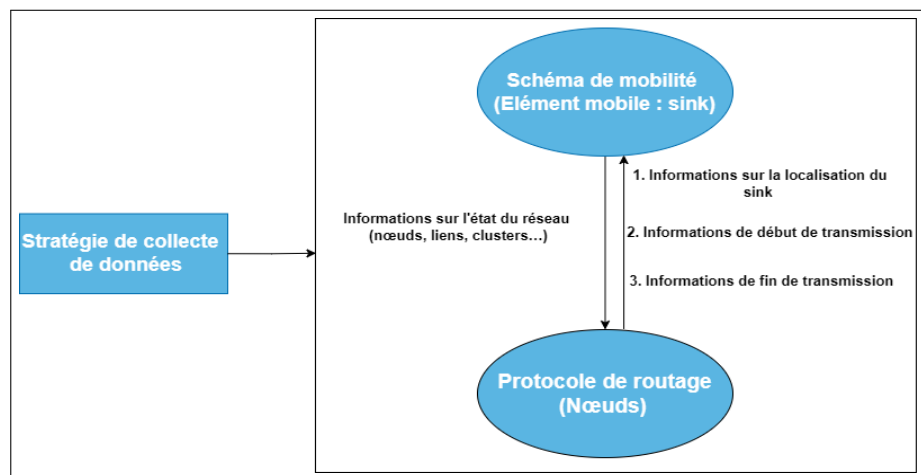


FIG. 3.1 – Les éléments d'une solution de mobilité du collecteur dans un réseau de capteurs sans fil.

3.2.1 Le modèle de mobilité

Un modèle de mobilité, également appelé "mobility models"[72] ou "mobility scheme", illustre le mouvement d'un élément mobile dans un réseau, dans ce cas, le puits dans un RCSF. Il permet de représenter la manière dont le puits se déplace à travers la zone de déploiement, en prenant en compte des facteurs tels que le terrain, les routes, la taille de la zone et la possibilité de navigation. Ces modèles sont essentiels pour simuler, analyser et

concevoir des protocoles de communication adaptés, car le mouvement du puits impacte directement la façon dont les données sont collectées et acheminées dans le réseau. Le modèle de mobilité est utilisé pour formaliser le comportement dynamique du puits dans le réseau, afin d’améliorer la compréhension et l’optimisation de son interaction avec les nœuds capteurs.

3.2.2 Le protocole de routage

Un protocole de routage est un ensemble de règles et d’algorithmes utilisés pour déterminer comment les données sont transmises dans un réseau. Il permet de choisir le chemin le plus efficace entre les nœuds pour acheminer les informations de manière optimale.

Selon Jamal N[73], dans les RCSF, un protocole de routage est essentiel pour garantir que les données collectées par les capteurs soient transmises de manière fiable vers un nœud central, souvent appelé sink (puits de données), tout en optimisant la consommation d’énergie et en gérant la topologie dynamique du réseau.

3.2.3 La stratégie de collecte de données

La stratégie de collecte des données, aussi appelée schéma de collecte des données[74], mécanisme d’acquisition des données[75] ou encore protocole de collecte des données[76], désigne les méthodes utilisées pour transmettre les informations captées par les nœuds vers le collecteur (ou "sink"). Dans un RCSF, cette stratégie joue un rôle essentiel car elle doit prendre en compte plusieurs contraintes comme l’économie d’énergie et la fiabilité de la transmission des données. Il existe plusieurs façons d’organiser la collecte. Dans la collecte directe, chaque capteur transmet ses données directement au collecteur, tandis que dans la collecte indirecte (ou multi-saut), les données transitent par d’autres nœuds avant d’atteindre le collecteur.

3.3 Classification des solutions de mobilité du Sink dans les réseaux de capteurs sans fils

3.3.1 Classification selon le Modèle de Mobilité

La mobilité du Sink dans les RCSF peut être catégorisée en fonction de la manière dont ses déplacements sont définis ou planifiés. Dans ce contexte, nous adoptons une classification standardisée en trois catégories principales : la mobilité aléatoire, la mobilité contrôlée, et la mobilité prédictive ou adaptative, comme illustré dans la Fig. 3.2.

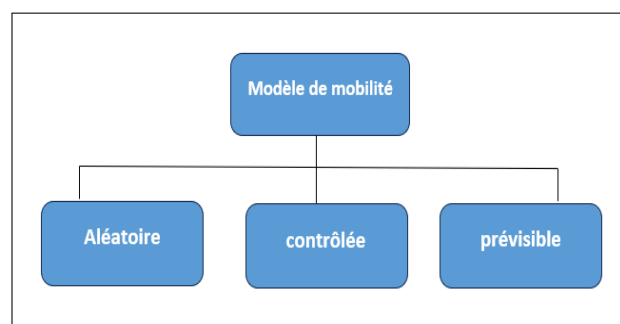


FIG. 3.2 – Les classes de modèles de mobilité.

- **Mobilité Aléatoire** : La mobilité aléatoire se caractérise par des déplacements imprévisibles du Sink dans le réseau, sans trajectoire définie à l'avance. Ce type de mobilité est généralement modélisé par des modèles tels que le Random Waypoint [77] (inclut des pauses entre les changements de destination et de vitesse) ou le Random Walk [77] (basé sur des directions et des vitesses aléatoires), où le Sink détermine de manière aléatoire sa prochaine position et sa vitesse.

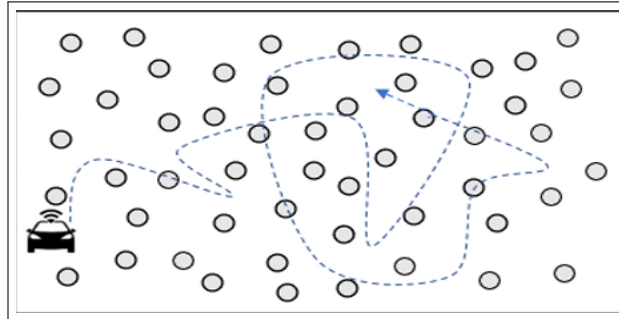


FIG. 3.3 – Modèle de mobilité Aléatoire [13].

- **Mobilité Prévisible** : La mobilité prévisible fait référence à une approche où le Sink suit un chemin préétabli et connu à l'avance. Ce modèle permet aux nœuds du réseau de connaître la trajectoire du Sink et ainsi de préparer la collecte des données à l'avance. Lorsque le Sink se déplace selon ce chemin prévu, les nœuds savent quand il sera proche, ce qui minimise les temps d'attente pour le transfert des données. Cette méthode est souvent utilisée dans des applications spécifiques, telles que les transports urbains [78], où le sink suit une route bien définie. Cela permet d'optimiser la collecte des données et de réduire l'énergie consommée par le réseau [79].

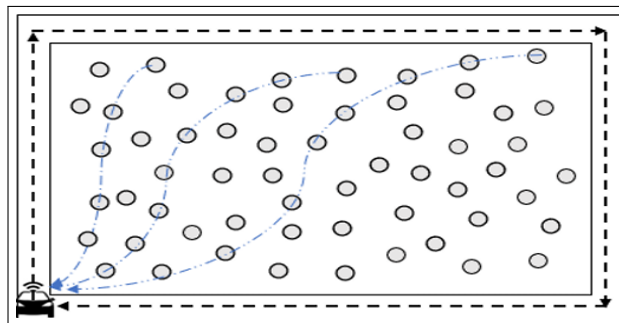


FIG. 3.4 – Modèle de mobilité prévisible [13].

- **Mobilité Contrôlée** : La mobilité contrôlée se caractérise par la capacité des nœuds mobiles, notamment le Sink, à ajuster activement leur trajectoire et leur vitesse dans le but d'optimiser les performances du réseau. Contrairement à la mobilité aléatoire, cette approche permet de planifier les déplacements en fonction des besoins en communication. Elle contribue directement à l'efficacité de la collecte de données dans les RCSF mobiles (RCSF) [80]. Grâce à cette mobilité supervisée, les éléments mobiles peuvent être chargés de visiter les nœuds à des instants précis, réduisant ainsi les délais de découverte et augmentant le temps de contact disponible pour transférer les données.

Un exemple concret de cette approche est présenté dans l'étude de Somasundara [81], où un robot mobile est utilisé comme station de base mobile. Sa vitesse et son parcours sont soigneusement contrôlés pour maximiser le débit de données tout en réduisant la consommation d'énergie globale du réseau. Ce modèle explore aussi les

compromis énergétiques liés au déplacement du Sink, la gestion des tampons, et propose des protocoles de communication améliorés adaptés à ce type de mobilité .

3.3.2 Classification selon les protocoles de routages

Dans les RCSF, la gestion de la mobilité du collecteur (Sink) est essentielle pour garantir une communication efficace et une collecte de données optimale. Le choix des protocoles de routage dépend de plusieurs facteurs, tels que la nature de la mobilité du Sink, la topologie du réseau, et les exigences en termes de consommation d'énergie.

Comme détaillé dans le Chapitre 2 et illustré dans la Fig.3.5. ci-dessous , les protocoles de routage peuvent être classés en trois grandes catégories : réactifs, proactifs et hybrides.

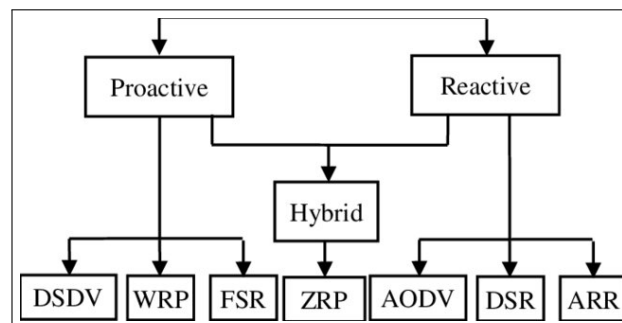


FIG. 3.5 – Classification des Protocoles de Routage [14].

En complément à la classification fonctionnelle (réactif, proactif, hybride), une autre approche largement adoptée dans la littérature consiste à classer les protocoles de routage selon leur structure de conception et les informations utilisées pour le routage. Comme indiqué dans la Fig.3.6, les différents protocoles de routage dans les RCSF sont ainsi classés selon ces critères :

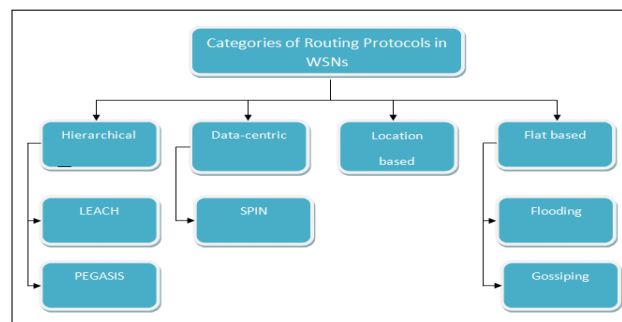


FIG. 3.6 – Catégories de protocoles de routage dans les RCSF [15].

- **Protocoles basés sur la localisation** : utilisent les informations de position des nœuds pour guider le routage des données. Un exemple représentatif est le protocole GEAR (Geographic and Energy Aware Routing) [82]. Ce protocole oriente les requêtes vers des régions spécifiques du réseau en tenant compte à la fois de la position géographique et du niveau d'énergie des nœuds. En restreignant la diffusion des messages à une zone ciblée, GEAR permet une économie d'énergie significative par rapport à d'autres protocoles comme Directed Diffusion.
- **Protocoles hiérarchiques ou en cluster** : ils organisent les nœuds en clusters avec un chef de cluster chargé de collecter et de transmettre les données au Sink. Le protocole LEACH (Low-Energy Adaptive Clustering Hierarchy)[82], est un exemple classique. Il sélectionne de manière aléatoire des chefs de cluster, répartissant ainsi

la charge énergétique entre les nœuds. Ce mécanisme permet de prolonger la durée de vie du réseau tout en minimisant les transmissions redondantes.

- **Protocoles basés sur le centrage des données (Data-Centric Protocols) :** Ce sont des protocoles basés sur l'intérêt des nœuds pour des données spécifiques. Les intérêts sont diffusés à travers le réseau, et les nœuds qui trouvent que l'information les intéresse peuvent demander ces données. Ce fonctionnement suit un modèle de publication et d'abonnement. Des exemples incluent le protocole SPIN [15], où les nœuds publient des informations intéressantes et s'abonnent à celles qui les concernent.
- **Protocoles plats (Flat-Based Protocols) :** Aussi appelés protocoles de routage pair-à-pair, ces protocoles traitent tous les nœuds de manière égale, sans hiérarchie particulière. Aucun type spécifique de configuration de réseau n'est appliqué, et chaque nœud aide au transfert des informations. Les protocoles sous cette catégorie incluent Flooding et Gossiping [15], qui permettent une diffusion d'informations sans hiérarchie.

3.3.3 Classification selon l'architecture des solutions proposées

- **Solutions à architecture plate :** Les solutions plates reposent sur une structure sans hiérarchie, tous les nœuds sont considérés comme équivalents en termes de responsabilité et de rôle. En général, la transmission de données se fait de façon pair-à-pair, sans avoir besoin de nœuds dédiés comme des chefs de cluster. Ces approches sont généralement plus faciles à implémenter et s'adaptent bien aux réseaux de petite taille. Cependant, elles peuvent entraîner une surcharge de communication dans les grands réseaux, une consommation énergétique élevée, et une réduction de la durée de vie du réseau, surtout lorsque le Sink est mobile.
- **Solutions hiérarchiques :** Les solutions hiérarchiques exploitent une organisation en niveaux ou en clusters, où certains nœuds, généralement appelés chefs de cluster (cluster-heads) jouent un rôle central dans la collecte, l'agrégation et la transmission des données. Ce modèle offre une gestion efficace de l'énergie, car les communications entre les nœuds standards et le Sink passent par les chefs de cluster, réduisant ainsi le nombre de transmissions longues distances. De plus, il est particulièrement efficace dans les scénarios avec un Sink mobile, puisqu'il modifie de façon dynamique la hiérarchie en fonction des déplacements du Sink.
- **Solutions hybrides :** Les approches hybrides visent à combiner les avantages des solutions plates et hiérarchiques. Par exemple, certains protocoles peuvent adopter une architecture plate pour les communications locales, tout en faisant appel à une structure hiérarchique pour la transmission longue distance vers le Sink. Cela permet une meilleure adaptabilité, notamment dans les réseaux à topologies dynamiques ou variées.

3.4 Étude comparative de quelques solutions utilisant un sink mobile dans les réseaux de capteurs sans fils

De nombreuses approches ont été proposées pour intégrer la mobilité de sink dans les RCSF, dans le but d’optimiser les performances globales, en particulier la consommation énergétique et la durée de vie du réseau.

3.4.1 Présentation de quelques solutions utilisant un sink mobile dans les réseaux de capteurs sans fils

Dans cette partie, nous présentons quelques travaux de recherche ayant fait recours à la mobilité du sink pour améliorer la collecte de données dans les RCSF. Les études sélectionnées sont regroupées en trois catégories, en fonction de la structure réseau sur laquelle elles s’appuient : les solutions à structure plate, les solutions à structure hiérarchique, et les solutions hybrides, qui combinent les deux premières.

3.4.1.1 Solutions à structure hiérarchique

L’article, écrit par J. Lu[16], présente un protocole de routage efficace en énergie appelé CTEER (Cluster Tree-based Energy-efficient Routing) pour les RCSF avec un collecteur mobile. Le protocole vise à améliorer l’efficacité énergétique et réduire la latence dans ces réseaux en construisant un arbre de routage dynamique qui s’ajuste en fonction de la position de l’émetteur.

CTEER introduit le concept de région de communication croisée, une zone centrale autour du collecteur où un arbre de routage en forme de croix est établi afin d’optimiser la transmission des données. Les nœuds situés en dehors de cette zone sont organisés en clusters à l’aide d’un algorithme de clustering (G-Means), chaque cluster étant dirigé par un chef de cluster chargé de l’agrégation et de l’envoi des données vers l’arbre principal. Le protocole fonctionne en deux étapes : une phase de configuration, destinée à la formation des clusters et à l’établissement des routes, suivie d’une phase de collecte stable durant laquelle les données sont transmises vers le collecteur mobile de manière efficace.

Les résultats expérimentaux montrent que CTEER permet une réduction notable de la consommation d’énergie, du nombre de sauts et des délais de transmission, par rapport à des protocoles tels que RRP et FRM. Il contribue ainsi à prolonger la durée de vie du réseau, tout en offrant de bonnes performances pour les applications sensibles au temps, comme la télésurveillance médicale ou le suivi d’objets en temps réel.

Li et al. dans [17] abordent la problématique de la collecte de données dans les RCSF (WSN) en présence d’une génération de données non uniforme, causant un déséquilibre

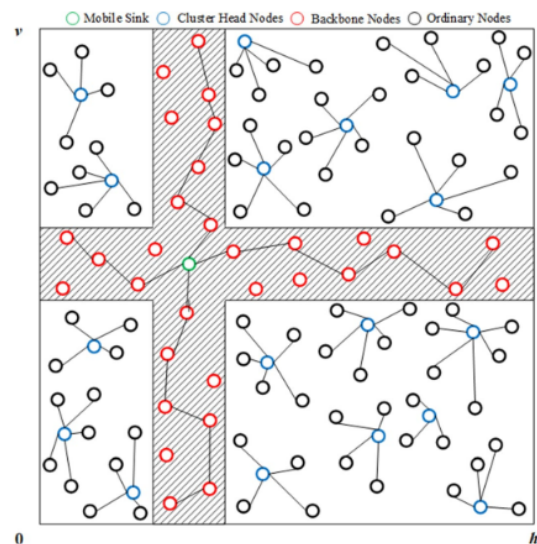


FIG. 3.7 – Network structure constructed by CTEER[16].

énergétique important et une réduction de la durée de vie du réseau. Pour résoudre ce problème, ils proposent une stratégie conjointe de mobilité du puits mobile (mobile sink) et d’organisation hiérarchique du réseau par clustering.

L’approche proposée, nommée OCNTMS (Optimal Clustering and Network Topology for Mobile Sink), repose sur la division du réseau en clusters, avec une sélection optimale des chefs de clusters (CH) qui agissent comme points de rendez-vous (RPs) pour la collecte des données. Les nœuds membres du cluster transmettent leurs données à leur CH (Cluster Head) en utilisant un routage multi-sauts intra-cluster optimisé. Pour déterminer ces clusters, les auteurs utilisent un algorithme k-medoids pondéré prenant en compte à la fois l’énergie résiduelle des nœuds et leur taux de génération de données. Ensuite, une fonction de coût multi-critères est définie afin d’identifier la configuration optimale du réseau, en équilibrant la longueur du trajet du puits mobile entre les points de rendez-vous, la consommation énergétique totale des nœuds, ainsi que la distribution équitable de la charge entre les différents clusters. La planification du trajet du puits mobile est ensuite optimisée à l’aide d’un algorithme hybride GA-MARL, combinant un algorithme génétique avec un apprentissage par renforcement multi-agent. Cela permet d’adapter dynamiquement le parcours du puits mobile pour minimiser les délais de collecte et éviter les zones de surcharge énergétique. Les résultats expérimentaux ont montré que cette stratégie conjointe améliore significativement les performances du réseau. En particulier, la consommation énergétique est réduite de 18%, et la durée de vie du réseau est prolongée de 40% par rapport aux approches existantes ne prenant pas en compte la génération non uniforme des données.

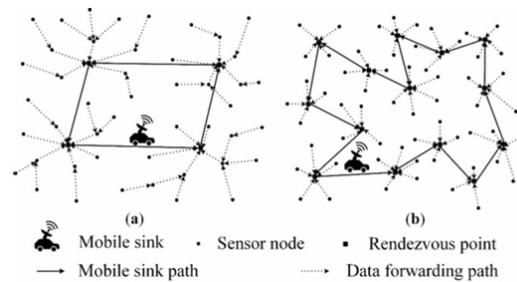


FIG. 3.8 – WSN, RPs, MS et trajet de déplacement[17].

L’étude de **Xu**[18] introduit une approche novatrice pour améliorer le traitement des requêtes dans les RCSF (WSN) intégrés à l’Internet des objets (IoT), en réponse aux limites des méthodes classiques fondées sur des puits statiques, notamment en termes de latence élevée, de consommation énergétique déséquilibrée et de faible durée de vie du réseau. Pour y remédier, l’auteur introduit DSQPS (Dynamic Sink-based Query Processing Strategy), une stratégie dynamique exploitant la mobilité intelligente des puits pour optimiser à la fois l’efficacité énergétique, la réactivité du système et la pérennité du réseau.

L’originalité de DSQPS réside dans la combinaison de plusieurs mécanismes d’optimisation. Le réseau est tout d’abord structuré autour d’un ensemble optimal de points de rendez-vous (RPs), déterminés à partir d’un problème de couverture minimale. Ensuite, le nombre de puits mobiles nécessaires est calculé de manière optimale grâce à l’algorithme Aquila Optimizer (AO), qui s’inspire du comportement de chasse de l’aigle pour équilibrer exploration globale et exploitation locale de l’espace de solutions. Enfin, chaque puits mobile se voit attribuer une sous-tournée optimisée via une heuristique de type Christofides, garantissant une collecte de données rapide et efficace en énergie.

Lorsque la station de base reçoit une requête relative à une région d’intérêt (RoI), elle la redirige vers le puits correspondant. Celui-ci se déplace alors vers le RP concerné, orchestre la récupération des données auprès des nœuds capteurs environnants via un mécanisme d’ordonnancement temporel, puis transmet les résultats au serveur cloud. Ce processus permet une exécution fluide et réactive des requêtes, tout en limitant la surcharge des nœuds et les pertes énergétiques (voir Fig.3.9).

Les simulations réalisées démontrent une nette supériorité de DSQPS par rapport à des approches de référence comme QDWSN, QWRP et QDVGDD. DSQPS permet notamment de réduire la latence de traitement des requêtes jusqu’à 38% , de diminuer la consommation énergétique moyenne de 30% , tout en augmentant la durée de vie du réseau de 150% et le débit global de 60% . Ces gains sont rendus possibles grâce à une conception fine du parcours des puits et une organisation topologique adaptative du réseau.

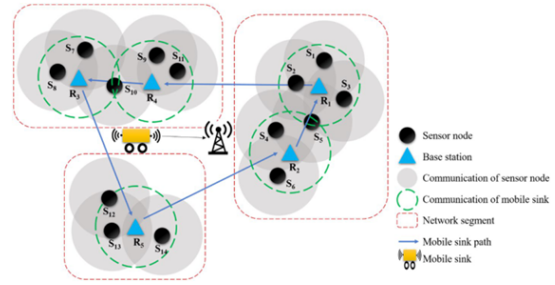


FIG. 3.9 – Stratégie de collecte de données basée sur les points de rendez-vous[18].

3.4.1.2 Solutions à structure plate

L’article[19] aborde la question de la distribution de données dans des RCSF équipés d’un puits mobile. Bien que ce type de réseau présente de nombreux bénéfices, y compris une couverture améliorée de la zone de détection et un meilleur équilibre énergétique, il pose également des défis dus à la mobilité du puits, particulièrement en ce qui concerne la transmission efficace des données. La mise à jour régulière de la localisation du puits auprès de tous les nœuds du réseau pourrait causer une surcharge de communications et une trop grande consommation d’énergie. Pour traiter ce problème, les auteurs introduisent un protocole hiérarchique nommé EDVWDD (Diffusion de Données Virtuelles Basée sur une Roue, Déclenchée par Événements), qui s’appuie sur une structure virtuelle ronde composée de nœuds stratégiques. Ces nœuds, désignés comme nœuds de roue, ont pour tâche de maintenir à jour la localisation du puits, tandis que les capteurs standards peuvent y accéder par un saut direct. La transmission des données est activée uniquement lors de la détection d’un événement, ce qui aide à réduire les communications superflues. Le protocole inclut également un mécanisme de reconstruction afin de prévenir l’épuisement énergétique des nœuds de roue. En comparaison avec d’autres méthodes telles que RRP, GCRP ou NRP, EDVWDD présente des résultats supérieurs en termes de consommation d’énergie, de temps de transmission et de taux de livraison des données. L’article conclut en proposant des voies pour intégrer plusieurs puits mobiles ou prendre en considération la présence d’obstacles dans la zone de détection.

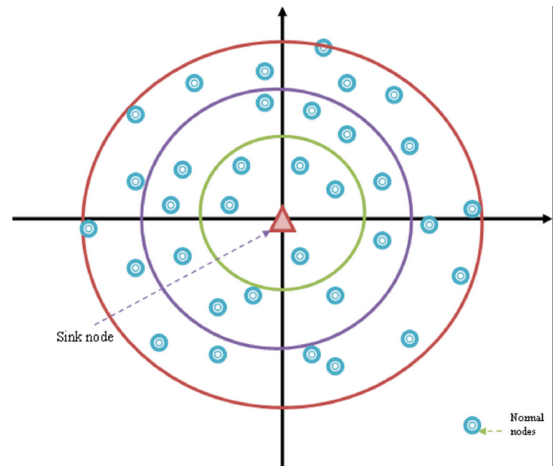


FIG. 3.10 – Structure de la roue virtuelle dans EDVWDD[19].

Jain et al[20]. proposent une nouvelle approche de routage à base de grille virtuelle, nommée VGRQ (Virtual Grid-based Routing for Query-driven WSNs), destinée aux réseaux de capteurs sans fil (WSN) avec puits mobile dans un contexte de requêtes orientées. Contrairement aux méthodes hiérarchiques fondées sur le clustering, cette solution repose sur une topologie plane, où l’ensemble des nœuds est organisé selon une structure de grille virtuelle sans hiérarchie explicite.

L’algorithme VGRQ divise le champ de capteurs en cellules carrées virtuelles, dont le nombre correspond à celui des nœuds du réseau. Chaque cellule est associée à un cell header (CH) élu selon sa position géographique et son niveau d’énergie. Ces CHs sont ensuite liés entre eux en une chaîne de communication, facilitant à la fois la transmission des requêtes émises par le puits mobile et la collecte des données issues des zones d’intérêt (AoI). Le protocole fonctionne en cinq phases principales : (1) construction de la grille, (2) élection des CHs, (3) mise à jour de la position du puits mobile auprès des CHs, (4) propagation des requêtes vers les AoI, et (5) retour des données via la chaîne de CHs. Grâce à cette architecture, le protocole évite le routage par inondation, réduit la redondance des messages et équilibre la charge énergétique entre les nœuds.

Les résultats des simulations ont mis en évidence la supériorité de VGRQ par rapport aux approches QRRP et QDVGDD. En particulier, VGRQ permet une réduction de la consommation énergétique d’environ 10 % par rapport à QRRP et 27 % par rapport à QDVGDD, tout en diminuant le délai de livraison des données et en améliorant le taux de livraison. Ces performances sont attribuées à l’utilisation judicieuse des chaînes de CHs pour localiser dynamiquement le puits et transmettre efficacement les informations, sans introduire de surcharge majeure dans le réseau.

En tant qu’approche à topologie plane optimisée pour les scénarios à puits mobile et requêtes dynamiques, VGRQ se distingue par sa simplicité de mise en œuvre et son efficacité énergétique, en particulier dans les environnements où les mécanismes hiérarchiques sont difficiles à maintenir.

Sharma et al[21]. ont présenté dans leur article ”Rendezvous-Based Routing Protocol for Wireless Sensor Networks with Mobile Sink” un nouveau protocole de routage destiné aux RCSF avec sink mobile. Leur méthode, désignée sous le nom de Rendezvous-Based Routing Protocol (RRP), consiste à établir une zone de rendez-vous virtuelle au cœur du réseau, où un arbre de communication, dénommé backbone-tree, est constitué à partir de nœuds spécifiquement sélectionnés. Le protocole prévoit deux modes de transmission des données. Dans la Méthode 1, l’arbre est orienté vers le sink, et les nœuds sources relaient leurs données à travers cet arbre. Dans la Méthode 2, le sink transmet sa position à l’arbre, permettant aux nœuds sources d’obtenir cette position depuis l’arbre afin de transmettre directement les données au sink.

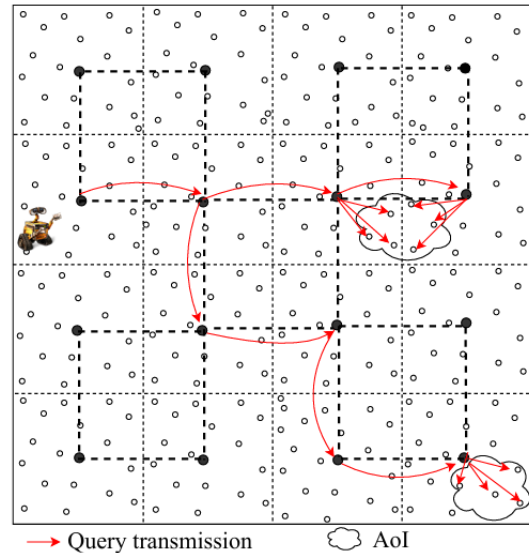
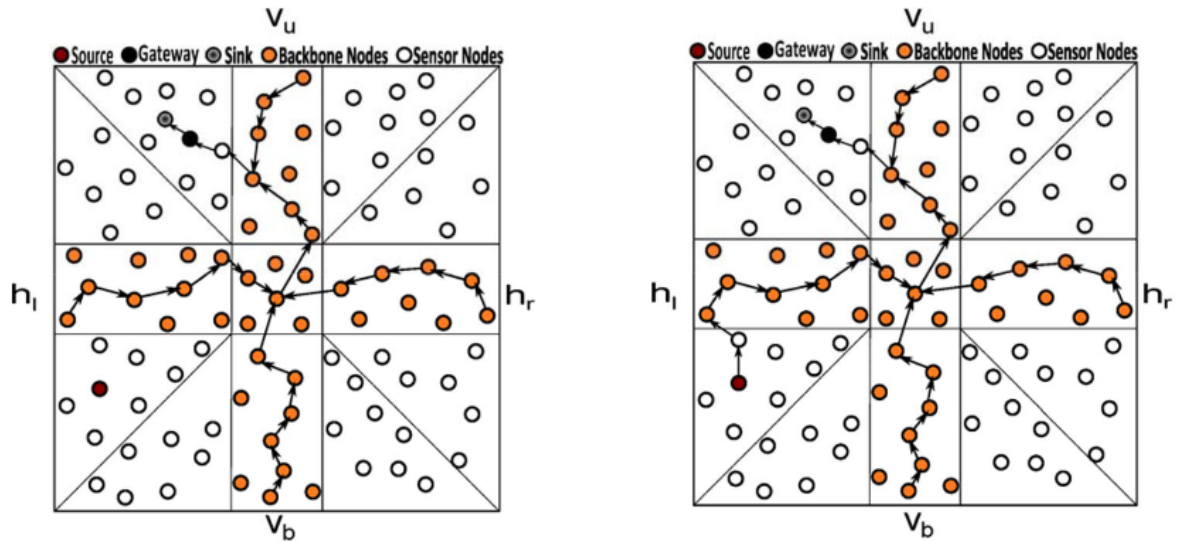


FIG. 3.11 – Illustration de la transmission de requêtes dans VGRQ, du sink mobile vers la zone d’intérêt [20].



(a) Lien des nœuds de l'arbre de dorsale dirigé vers le sink.

(b) Transmission des données à travers les nœuds de l'arbre dorsale.

FIG. 3.12 – Modèle de réseau et stratégies de collecte des données[21].

Les nœuds capteurs peuvent faire parvenir leurs données au sink en passant par les nœuds de l'arbre de dorsale (backbone-tree). Chaque nœud capteur détient des informations sur ses voisins, notamment leur position et leur énergie résiduelle. Cela leur permet de transmettre aisément les données générées vers un nœud de l'arbre de dorsale en utilisant des nœuds voisins, en se basant sur le facteur de localisation (LF) établi par les auteurs. Le nœud source peut choisir un nœud ayant une énergie résiduelle adéquate et une distance minimale par rapport à la destination pour l'envoi des données (voir FIG. 3.12).

Han et al. dans l'article [22], proposent HGFF (Heterogeneous Graph-based Feature Fusion), un cadre d'apprentissage par renforcement profond destinée à prolonger la durée de vie des RCSF en optimisant de manière dynamique le parcours d'un sink mobile. Cette approche répond aux limites des techniques classiques fondées sur la programmation mathématique ou les heuristiques. La particularité de HGFF réside dans la modélisation du RCSF sous forme de graphe hétérogène, qui inclut deux types de nœuds : capteurs et sites potentiels pour le sink. Un réseau de neurones graphiques (GNN) est utilisé pour obtenir des représentations riches des nœuds, en intégrant des embeddings¹ de type de nœud et un mécanisme d'attention multi-têtes global, permettant aux sites d'exploiter les informations énergétiques du réseau. Le modèle fonctionne selon un processus d'apprentissage agent-environnement, où chaque déplacement du sink influence la configuration énergétique globale. À l'aide de l'algorithme de Double Q-learning, HGFF apprend une politique de mouvement efficace, sélectionnant les sites optimaux pour équilibrer la consommation d'énergie et maximiser la durée de vie. La collecte de données repose sur des transmissions multi-sauts, comme illustré dans la Fig. 3.13.

Les simulations, réalisées sur dix types de cartes statiques et dynamiques, montrent que HGFF dépasse les méthodes de référence telles que GMRE, ACO, SL-GEP et DDQN, en prolongeant significativement la durée de vie du réseau, en réduisant la surcharge des nœuds proches du sink, et en maintenant un taux de collecte de données élevé. Une étude d'ablation confirme l'importance des mécanismes d'attention et d'embedding, et les visualisations révèlent une gestion intelligente de la dissipation énergétique dans divers scénarios.

¹embeddings : une représentation vectorielle apprise

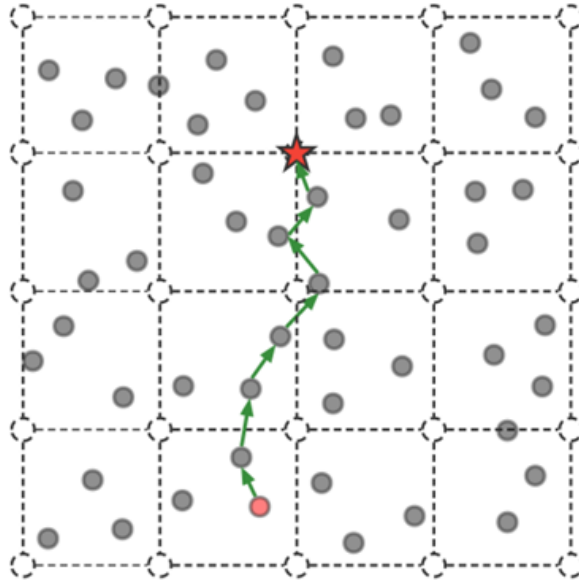


FIG. 3.13 – Architecture du modèle HGFF basé sur un graphe hétérogène et l’apprentissage par renforcement profond [22].

3.4.1.3 Solutions à structure hybride

Une solution hybride a été introduite par **Moussa et al.** [23], sous le nom de EHRP (Effective Hybrid Routing Protocol), destinée à améliorer les performances des RCSF à sink mobile tout en maintenant un bon compromis entre consommation énergétique, efficacité de transmission et latence. Cette solution repose sur une architecture en grilles virtuelles, dans laquelle chaque cellule est représentée par un chef de cluster (CH) responsable de la collecte des données des nœuds voisins.

La stratégie de routage adoptée combine deux types de transmission : directe (un seul saut) lorsque le sink est proche, et multi-saut lorsqu’il est distant (voir Fig.3.14). Cette flexibilité permet d’optimiser la communication sans compromettre la stabilité du réseau. Un algorithme de colonie de fourmis amélioré, basé sur une approche bio-inspirée, est utilisé pour choisir les meilleures routes, en se basant sur des critères comme la puissance du signal reçu (RSSI), l’énergie résiduelle minimale et moyenne des nœuds.

Les résultats montrent que EHRP surpasse les performances de plusieurs protocoles existants (tels que ECMMS et IoGHR), en termes de réduction de la latence et de la consommation, et d’augmentation de la durée de vie du réseau. Son architecture adaptative et son mécanisme de routage hybride, associée à une prise de décision locale optimisée, en fait une alternative efficace pour les réseaux de capteurs à grande échelle, où la stabilité et la flexibilité du routage sont essentielles.

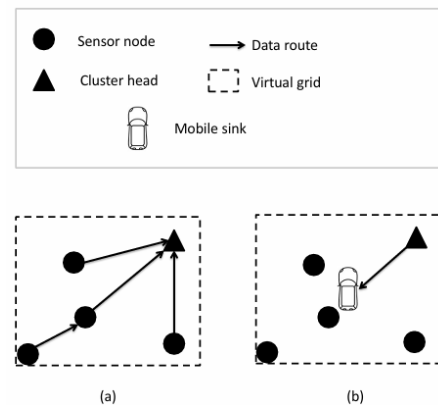


FIG. 3.14 – Fonctionnement hybride du protocole EHRP combinant transmission directe et multi-saut.[23]

Une autre approche hybride, nommée EE-hHSS (Energy-Efficient hybrid Harris Hawk–Salp Swarm) et proposée par **Srinivas et Amgoth** [24], combine une organisation hiérarchique du réseau avec plusieurs algorithmes d’intelligence artificielle bio-inspirés afin d’améliorer la collecte de données dans les RCSF à sink mobile.

Le réseau est d’abord divisé en clusters via l’algorithme K-Medoids, qui prend en compte la distance entre les nœuds et leur énergie résiduelle. Ensuite, la sélection des chefs de cluster (CH) est réalisée à l’aide d’un mécanisme hybride basé sur Harris Hawk Optimization (HHO) et Salp Swarm Algorithm (SSA), permettant un équilibrage optimal de la charge et une répartition énergétique intelligente. Une fois les clusters formés, la trajectoire du puits mobile est planifiée à l’aide de l’algorithme de colonie de fourmis adaptatif (AACO), visant à minimiser les déplacements tout en garantissant la couverture des zones de collecte.

Les résultats expérimentaux, obtenus à travers des simulations sur Java, confirment la supériorité du protocole EE-hHHSS sur les plans de la durée de vie du réseau, de la réduction de la latence, et du taux de livraison. Grâce à sa combinaison de clustering intelligent, de bio-algorithmes et d’optimisation de la mobilité, cette approche constitue une solution robuste, particulièrement adaptée aux environnements dynamiques exigeant une gestion fine de l’énergie.

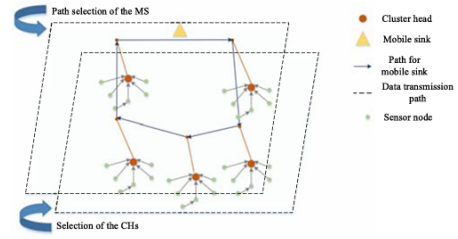


FIG. 3.15 – Formation des clusters et sélection des chefs de cluster dans EE-hHHSS.[24]

3.4.2 Comparaison et classification des solutions étudiées

Dans cette partie, nous comparons les différentes solutions précédemment présentées à travers plusieurs critères : le type de structure du réseau (plate ou hiérarchique), le modèle de mobilité adopté par le puits, le protocole de routage utilisé et les performances obtenues (durée de vie, latence, consommation énergétique, etc.). Le tableau suivant permette de synthétiser ces caractéristiques pour faciliter l’analyse comparative.

3.4.3 Synthèse et discussion

À partir de l’analyse des différentes solutions étudiées, plusieurs constats peuvent être établis :

- La mobilité discrète du sink est largement utilisée car elle permet de minimiser la surcharge générée par les déplacements réguliers du sink.
- L’optimisation contrôlée de la trajectoire du sink, fréquemment réalisée par des algorithmes comme GA-MARL ou des méthodes heuristiques comme Christofides, contribue significativement à minimiser les temps de transmission et à améliorer l’efficacité énergétique.
- Les solutions à mobilité contrôlée montrent de meilleures performances pour les réseaux de petite à moyenne taille.
- L’utilisation de la communication à un saut entre les nœuds capteurs et le sink permet d’éviter la surcharge liée à la localisation du sink et à la mise à jour des routes, cependant cela peut provoquer un retard supplémentaire et des pertes de données en raison du débordement des buffers dans les réseaux plus denses.
- Le routage multi-saut est souvent utilisé pour minimiser la latence et la perte de données, mais nécessite des stratégies d’optimisation efficaces pour limiter le flux supplémentaire généré.

TAB. 3.1 – Tableau comparatif des caractéristiques de mobilité des solutions à sink mobile

| Solution | CM | Trajectoire du sink | N | Temps de pause | Décision de mouvement |
|---------------|----|--|---|---------------------------------------|--|
| CTEER [16] | P | Déplacement du sink dans une région en croix, clusters à l’extérieur | D | Adaptatif (selon la position du sink) | Se déplace selon la position du sink et la formation croisée |
| OCNTMS[17] | C | Ensemble des points de RDV optimaux | D | Optimisé via GA-MARL | Minimise la longueur du trajet et équilibre la charge |
| DSQPS [18] | P | Ensemble des points de RDV (RPs) | D | Trajets optimisés | Basé sur un ensemble de couverture minimale et l’optimiseur Aquila |
| EDVWDD [19] | P | Nœuds virtuels en roue met-tant à jour la position du puits | C | Sans pause | Déplacements déclenchés par événement |
| VGRQ [20] | P | ROUTAGE en grille virtuelle avec mise à jour du sink mobile | D | Sans pause | Se déplace vers les cell headers dynamiques |
| RRP [21] | P | Zone de rendez-vous centrale | C | Fixe | Se déplace via un arbre dorsal |
| HGFF [22] | C | Déplacement appris via graph embedding | D | Dynamique (selon l’énergie) | Politique de déplacement optimisée par Q-learning |
| EHRP [23] | C | Clusters heads de la structure virtuelle | D | Basé sur la distance au sink | Selon la proximité des CHs |
| EE-hHHSS [24] | C | Trajet du sink planifié par optimisation de colonie de four-mis | D | Jusqu’à collecte complète des données | Minimise le coût global de déplacement |

CM : Classe de mobilité (R : Aléatoire, P : Prédicible, C : Contrôlée)

N : Nature du mouvement (D : Discret, C : Continu)

”Temps de pause” décrit le temps d’attente du puits mobile.

”Décision de mouvement” indique comment et quand le puits choisit sa prochaine position.

TAB. 3.2 – Tableau comparatif des caractéristiques des protocoles de routage pour les solutions à puits mobile

| Solution | LA | SLI | | Protocole de routage | | | | Mise à jour | Charge |
|--------------|-----|-----------|--------|---|--|--|--|-------------|--------|
| | | comment ? | Propag | Fonction de routage | RT | | | | |
| CTEER [16] | Oui | BS | T | Clustering + arbre croisé dynamique | Routage arborescent adaptatif | | Partielle (à la fin de t_k) | F | |
| OCNTMS [17] | Oui | BS | P | Routage intra-cluster + points de rendez-vous (RPs) | Optimisation par GA-MARL | | Partielle (optimisation dynamique) | M | |
| DSQPS [18] | Oui | BS | P | Routage événementiel via points de rendez-vous dynamiques | Heuristique basée sur AO + Christofides | | Partielle (chemins optimisés) | F | |
| EDVWDD [19] | Non | BS | T | Suivi du puits par nœuds de roue | Diffusion déclenchée par événements | | Partielle (mise à jour par événements) | F | |
| VGRQ [20] | Non | BS | P | Transmission par chaînes de CH sur grille virtuelle | Routage via CHs dynamiques | | Partielle (reconstruction partielle) | F | |
| RRP [21] | Oui | BS | T | Communication via arbre dorsal centré | Routage sur backbone-tree | | Partielle (lors des déplacements) | F | |
| HGFF [22] | Non | KP | A | Routage par apprentissage profond (GNN + Q-learning) | Politique apprise par renforcement profond | | Dynamique (récompense énergie) | M | |
| EHRP [23] | Oui | BS | P | Hybride : direct ou multi-saut selon distance | Routage basé sur RSSI et distance au puits | | Partielle (choix local optimisé) | F | |
| EE-hHSS [24] | Oui | BS | P | Clustering intelligent + optimisations bio-inspirées | Routage via colonie de fourmis adaptative | | Partielle (mise à jour locale) | M | |

LA : Localisation Awareness (Oui/Non)

comment ? : Mode d'obtention (BS = Beacon envoyé par le Sink, KP = Known a priori : connu à priori)

Propagation : Propagation de l'information (P = Partielle, T = Transmission, A = Adaptatif)

RT : Type de structure de routage (Arbre, Multi-saut)

Mise à jour : Partielle ou Dynamique

Charge : Overhead réseau (Faible, Moyenne, Élevée)

- Dans les architectures à structure plate, des approches basées sur des divisions spatiales (grilles virtuelles, roues virtuelles) sont utilisées pour organiser les communications, nécessitant une mise à jour régulière de la position du sink pour maintenir l’efficacité.
- Dans les approches hiérarchiques, la formation de clusters par des algorithmes comme k-medoids ou G-Means permet de réduire le nombre de transmissions, en concentrant les communications au niveau des chefs de clusters (CH) ou des points de rendez-vous (RP).
- Les méthodes hiérarchiques garantissent une meilleure performance énergétique en réservant les interactions directes au sink uniquement aux CHs, même si cela implique une plus grande complexité de configuration.
- Certaines approches adoptent une stratégie hybride, comme EHRP, qui combine transmission directe et multi-saut en fonction de la distance au sink, optimisant ainsi la consommation énergétique selon la situation.
- L’intégration de l’intelligence artificielle, notamment par apprentissage par renforcement profond (comme dans HGFF), permet d’apprendre dynamiquement des trajectoires optimales sans se baser sur des règles préexistantes, ce qui améliore l’adaptation aux conditions réseau dynamiques.
- Les solutions bio-inspirées, telles que EE-hHHSS, utilisent l’optimisation par essaim pour sélectionner intelligemment les clusters et planifier efficacement la trajectoire du sink dans des environnements évolutifs.
- De manière générale, le choix entre une architecture plate, hiérarchique ou hybride dépend fortement du type d’application, de la dynamique du réseau, et de l’importance accordée à la facilité d’installation par rapport à l’efficacité énergétique et la longévité du réseau.

3.5 Conclusion

Ce chapitre a présenté les principales approches liées à la mobilité du collecteur dans les RCSF. Après avoir défini les éléments essentiels d’une solution de mobilité, une classification des différentes méthodes a été proposée selon le modèle de mobilité, le protocole de routage et l’architecture. Une étude comparative a ensuite permis d’analyser certaines solutions existantes. Ce travail met en évidence les défis actuels et ouvre des perspectives pour améliorer la collecte de données dans ces réseaux.

Dans ce contexte, le chapitre suivant introduit deux approches intelligentes basées sur l’apprentissage par renforcement, notamment le Q-Learning et le Deep Q-Learning, visant à optimiser le déplacement du sink de manière adaptative et efficace.

Chapitre 4

Déplacement intelligent d'un sink mobile dans RCSFs basé sur l'apprentissage par renforcement

Sommaire

| | | |
|-------|--|----|
| 4.1 | Introduction | 48 |
| 4.2 | Concepts fondamentaux du Q-learning | 48 |
| 4.2.1 | Fonction Q et équation de Bellman | 48 |
| 4.2.2 | Algorithm du Q-Learning | 49 |
| 4.2.3 | Exploration vs Exploitation : Stratégie ϵ -greedy | 49 |
| 4.3 | Description des solutions proposées | 50 |
| 4.3.1 | Modélisation du réseau | 50 |
| 4.3.2 | Description de la première solution : Q-Learning Based sink MOBilty (QLB-MOB) | 52 |
| 4.3.3 | Caractéristiques de l'algorithme de routage utilisé | 54 |
| 4.3.4 | Stratégie globale de collecte de données | 56 |
| 4.3.5 | Description de la deuxième solution : Deep Q-Learning based sink MOBilty (DQL-MOB) | 59 |
| 4.4 | Simulation et Résultats | 68 |
| 4.4.1 | Environnements et outils se simulation | 69 |
| 4.4.2 | Configuration et paramètres de simulation | 69 |
| 4.4.3 | Configuration du réseau | 70 |
| 4.4.4 | Métriques de performances | 71 |
| 4.4.5 | Présentation et discussion des résultats | 73 |
| 4.5 | Conclusion | 77 |

4.1 Introduction

Après avoir examiné les principales méthodes existantes fondées sur la mobilité du puits dans les réseaux de capteurs sans fil, ce chapitre se concentre sur la présentation de notre propre solution. Les RCSF sont déployés dans divers domaines, incluant l'agriculture, la surveillance environnementale et les applications militaires, afin de rassembler des données à partir de capteurs aux ressources limitées, en particulier sur le plan énergétique. Garantir une collecte de données efficace est crucial pour allonger la durée de vie du réseau et assurer des performances optimales. L'utilisation d'un sink mobile permet d'améliorer cette efficacité, mais elle soulève un défi majeur comment déterminer de manière judicieuse les trajectoires et les points de collecte pour minimiser les pertes, réduire les temps d'attente et équilibrer la consommation d'énergie.

L'objectif principal de notre proposition est de démontrer comment l'apprentissage par renforcement peut être exploité pour permettre au sink mobile de réaliser des choix optimaux et adaptatifs. Par cette méthode, le sink s'adapte en temps réel aux conditions du réseau (énergie restante des nœuds, quantités de données à transmettre, qualité des connexions) afin d'optimiser les performances de collecte et prolonger la durée de vie du réseau. Ce chapitre détaillera la modélisation du réseau, les caractéristiques du modèle de mobilité proposé, l'algorithme de routage utilisé, ainsi que la stratégie globale adoptée pour la collecte des données.

4.2 Concepts fondamentaux du Q-learning

Le Q-learning est un algorithme d'apprentissage par renforcement sans modèle qui vise à maximiser une fonction de valeur Q via l'équation de Bellman. Au centre du Q-learning, la Q-table est essentielle. Cette matrice se compose de lignes représentant différentes actions et de colonnes correspondant à divers états ou caractéristiques de l'environnement. L'agent se sert des valeurs Q contenues dans ce tableau comme des prévisions des récompenses cumulées attendues liées à l'exécution d'une action précise dans un état donné.

4.2.1 Fonction Q et équation de Bellman

- **Fonction de valeur Q** : désignée par $Q(s, a)$, représente la valeur d'une paire état-action. Elle est formulée comme la valeur moyenne cumulée des récompenses à venir lorsque l'agent se trouve dans un état s et opte pour une action a .
- **Équation de Bellman** : L'équation de Bellman pour la fonction de valeur Q décrit la connexion entre la valeur d'une paire état-action et celle des états actions suivants accessibles en suivant la politique optimale. Elle se définit de la manière suivante :

$$Q(s, a) \leftarrow Q(s, a) + \alpha \left[r + \gamma \cdot \max_{a'} Q(s', a') - Q(s, a) \right]$$

où :

- s est l'état actuel.
- a est l'action prise dans l'état s .
- s' est l'état suivant.
- a' est une action possible dans l'état s' .
- r récompense immédiate obtenue après une action.

- α est le taux d'apprentissage qui détermine à quel point la valeur Q est mise à jour avec la nouvelle information.
- γ est le facteur d'actualisation (discount factor).
- $\max_{a'} Q(s', a')$ qui représente la récompense cumulative future maximale attendue à partir de l'état s' en considérant toutes les actions possibles a' .

4.2.2 Algorithm du Q-Learning

L'algorithme standard du Q-learning, qui permet d'apprendre une politique optimale par le biais de l'interaction avec l'environnement, sans nécessiter un modèle de la dynamique de l'environnement, est présenté par l'algorithme 1 ci-dessous :

Algorithm 1 Q-Learning

Entrées : $s, a, \alpha, \gamma, \varepsilon, R(s, a)$

Initialisation :

$$Q(s, a) \leftarrow 0$$

Début

for chaque épisode **do**

$s \leftarrow$ état initial

while s n'est pas terminal **do**

$$a \leftarrow \begin{cases} \text{action aléatoire dans } A, & \text{avec probabilité } \varepsilon \\ \arg \max_{a'} Q(s, a'), & \text{sinon} \end{cases}$$

Calculer la récompense $r \leftarrow R(s, a)$

$$Q(s, a) \leftarrow Q(s, a) + \alpha \left(r + \gamma \max_{a'} Q(s', a') - Q(s, a) \right)$$

$s \leftarrow s'$

end while

end for

$$a^*(s) = \arg \max_a Q(s, a)$$

Fin

4.2.3 Exploration vs Exploitation : Stratégie ϵ -greedy

Le défi principal du dilemme entre exploitation et exploration dans l'apprentissage par renforcement consiste à déterminer si l'agent doit se concentrer sur l'utilisation des connaissances et actions établies afin d'optimiser les récompenses à court terme, ou s'il devrait explorer de nouvelles actions pour identifier des stratégies plus performantes sur le long terme. Le Q-learning doit aborder le dilemme entre exploration et exploitation. Pour ce faire, diverses méthodes sont mises en œuvre :

- **Epsilon-greedy** : Introduit un facteur d'exploration contrôlé par epsilon.
- **Softmax** : Utilise les valeurs Q pour calculer la probabilité de choisir chaque action.

- **Decay epsilon-greedy** : dérivé de ϵ -greedy, cet algorithme réduit progressivement la valeur de ϵ à chaque étape de façon exponentielle.

4.3 Description des solutions proposées

4.3.1 Modélisation du réseau

Dans nos propositions, le RCSF est constitué d'un ensemble de N nœuds capteurs statiques et homogènes et d'une station de base (sink) mobile. Chaque capteur possède des capacités restreintes, notamment en énergie, en mémoire tampon pour conserver les paquets, et en puissance d'émission pour communiquer avec ses voisins dans un rayon de communication R . Ces capteurs produisent régulièrement des données qu'ils doivent envoyer au sink mobile, ou au nœud de collecte, qui se déplace dans la zone de déploiement pour récupérer les informations directement auprès des capteurs. Le sink joue un rôle clé en collectant et en agrégeant les données générées par le réseau, tout en optimisant ses déplacements pour maximiser l'efficacité énergétique, minimiser les pertes de paquets et améliorer les performances globales. Le mouvement du sink est guidé par un algorithme fondé sur l'apprentissage par renforcement afin de prendre des décisions intelligentes et adaptées aux conditions du réseau.

Nous posons les hypothèses suivantes sur les caractéristiques du réseau :

- Le réseau est composé d'un ensemble de N nœuds capteurs statiques et homogènes et d'une station de base (sink) mobile.
- Le champs de capture prend une forme rectangulaire ou carrée de dimensions $L \times L$.
- Chaque nœud capteur se voit attribuer un identifiant unique $i = 1, 2, \dots, N$, ce qui permet de l'identifier de manière unique au sein du réseau.
- Une même quantité d'énergie initiale est attribuée à chaque capteur ; celle-ci décroît avec les activités de capture, traitement et surtout de communication.
- La capacité mémoire tampon de chaque nœud capteur est restreinte, ce qui peut provoquer des pertes de paquets en cas de surcharge.
- Le sink est illimité en terme de capacité mémoire, de calcul et d'énergie.

La Table 4.1 présente les notations utilisées dans les approches proposées, qui sont expliquées en détail dans le reste de ce cette section.

| Symbole | Description |
|-------------------------|---|
| $d(i, j)$ | Distance euclidienne entre les nœuds i et j |
| E_0 | Énergie initiale de chaque nœud capteur |
| E_{elec} | Énergie consommée par les circuits électroniques |
| E_{Recv} | Énergie consommée pour recevoir 1 bit |
| E_{Trans} | Énergie consommée pour transmettre 1 bit |
| E_{fs} | Facteur d'atténuation dans le modèle en espace libre |
| E_{amp} | Facteur d'atténuation dans le modèle multi-trajet |
| d_0 | Distance seuil entre les deux modèles (espace libre / multi-trajet) |
| i | Identifiant d'un nœud capteur |
| j | Indice représentant la position actuelle du sink |
| PC_j | Point de collecte j (position actuelle du sink) |
| m | Nombre total de régions |
| N | Nombre total de nœuds dans le réseau |
| u | Nœud source dans un algorithme de routage |
| v, w | Nœuds voisins considérés dans le routage |
| $dist[i]$ | Distance estimée du nœud source u au nœud i |
| $prev[i]$ | Nœud précédent de i dans le chemin optimal |
| $E[i]$ | Énergie résiduelle du nœud i |
| E_{max} | Énergie maximale d'un nœud |
| $B[i]$ | Quantité de paquets dans le tampon du nœud i |
| B_{max} | Capacité maximale du tampon d'un nœud |
| R | Portée de transmission d'un nœud |
| L | Taille du champ de déploiement |
| PDR | Taux de livraison des paquets |
| PLR | Taux de perte des paquets |
| NLT | Durée de vie du réseau |
| s, s_t | État courant du système (énergie, buffers, position du sink, etc.) |
| a, a_t | Action choisie à l'instant t (déplacement vers un nœud) |
| $Q(s, a)$ | Valeur Q pour l'état s et l'action a |
| Q_θ | Réseau principal DQN avec poids θ |
| Q_{θ^-} | Réseau cible stabilisé |
| α | Taux d'apprentissage |
| γ | Facteur d'actualisation |
| ϵ | Paramètre d'exploration (ϵ -greedy) |
| r, r_t | Récompense obtenue après action a_t |
| s_{t+1} | Nouvel état observé après exécution de a_t |
| y | Cible d'apprentissage dans la mise à jour DQN |
| L | Fonction de perte du réseau de neurones Q |
| π_i | Score hybride attribué au nœud i |
| π_i^* | Politique optimale apprise pour le nœud i |
| \mathcal{B} | Mémoire d'expérience contenant les transitions (s, a, r, s') |
| $P = [i_1, \dots, i_k]$ | Chemin de transmission multi-saut vers le sink |
| $MAX_AGGREGATION$ | Seuil maximal de paquets que le sink peut agréger |
| $aggregated$ | Nombre de paquets envoyés du dernier nœud i_k vers le sink |
| $relayed$ | Nombre de paquets transmis entre deux nœuds consécutifs |
| $espace$ | Espace libre dans le tampon du nœud $i + 1$ |
| $chemin$ | Liste des nœuds dans le routage vers une cible |
| PQ | File de priorité (Dijkstra) |

TAB. 4.1 – Table des notations

4.3.2 Description de la première solution : Q-Learning Based sink MOBilty (QLB-MOB)

4.3.2.1 Caractéristiques du modèle de mobilité utilisé par la solution QLB-MOB

Trajectoire du sink et nature du mouvement

Dans cette première solution, le réseau de capteurs est modélisé sous la forme d'une grille composée de cellules de surface égale, couvrant l'ensemble de la zone de déploiement. Cette organisation spatiale a pour effet de simplifier la gestion du mouvement du sink et l'organisation des points de collecte. À l'initialisation, un point de collecte est assigné au centre de chaque cellule, servant de position de départ à partir de laquelle le sink peut planifier ses déplacements. Cette approche facilite la couverture du réseau et permet une gestion plus prévisible et équilibrée de la collecte des données, notamment dans le cadre d'un algorithme de décision ou d'optimisation basé sur la position du sink mobile.

La trajectoire du sink consiste en l'ensemble des points de collecte visités, au niveau desquels le sink s'arrête pendant une période dite **temps de collecte** (TC_j) pour rassembler les données des nœuds se trouvant dans la **zone de collecte** ou la **cellule** correspondante.

Une zone de collecte est ici définie comme l'ensemble des nœuds du réseau situés dans la cellule contenant le point de collecte actuel, c'est-à-dire la position courante du sink. Ainsi, la zone de collecte reste fixe à l'échelle de la cellule, tandis que le point de collecte peut être ajusté de manière adaptative à l'intérieur de la même cellule en fonction des conditions actuelles des nœuds et du réseau.

Le mouvement du sink est ainsi discret et la collecte des données se fait de manière discontinue au niveau de chaque cellule.

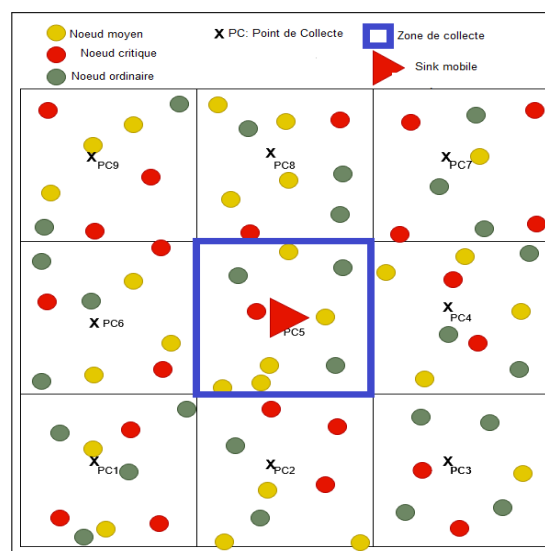


FIG. 4.1 – Réseau de capteurs organisé en zones de collecte (État initial).

4.3.2.2 Formulation du problème comme un Processus de Décision Markovien (MDP)

Dans cette solution, nous modélisons le processus de prise de décision du sink mobile comme un processus de décision markovien (Markov Decision Process – MDP). L'objectif est de guider le sink vers le nœud capteur le plus approprié pour la collecte des données, en tenant compte du volume de données accumulées à chaque nœud, de la distance entre le sink et ces nœuds, ainsi que du niveau d'énergie résiduelle de chacun.

Le MDP est défini par les éléments suivants :

- **États** (S) : Chaque état encode la position actuelle du sink mobile (Current-pos), la charge potentielle du prochain nœud (mesurée par le nombre de paquets dans sa file de transmission), ainsi que le niveau d'énergie résiduelle du nœud. Afin de simplifier le modèle, la charge et l'énergie sont représentées sous forme discrétisée. La charge est classée en quatre niveaux : Non chargé, Chargé, Fortement chargé, et Surchargé. Le niveau d'énergie est quant à lui catégorisé comme suit : Élevé, Moyen, Faible et Critique.
- **Actions** (A) : Chaque action correspond au déplacement du sink vers un nouveau point de collecte à proximité d'un nœud capteur. À chaque étape temporelle, l'agent représentant le sink mobile doit choisir le prochain nœud à visiter en se basant sur plusieurs critères, notamment la charge du nœud, son énergie résiduelle et sa distance par rapport à la position actuelle du sink.
- **Fonction de transition** (P) : Le système évolue d'un état à un autre en fonction de l'action choisie. Bien que les transitions liées aux déplacements du sink soient déterministes, les charges de données et les niveaux d'énergie peuvent varier dans le temps selon le nombre de paquets transmis lors du processus de relaying.
- **Fonction de récompense** (R) : La récompense traduit l'intérêt de visiter un nœud donné. Elle est d'autant plus élevée que le sink sélectionne un nœud fortement chargé, proche et non contraint en énergie. La fonction de récompense est définie comme suit :

$$R(s, a) = \alpha \cdot \text{charge}(a) - \beta \cdot \text{distance}(PCj, a) - \delta \cdot \text{pénalité_énergie}(a)$$

où α , β et δ sont des facteurs de pondération permettant de trouver un équilibre entre l'importance de la charge du nœud, la distance à parcourir, et la disponibilité énergétique.

- **Facteur d'actualisation** (γ) : Nous utilisons un facteur d'actualisation $\gamma = 0.9$, indiquant que les récompenses futures sont prises en compte, mais avec une importance légèrement inférieure à celle des récompenses immédiates.

Stratégie de déplacement du sink : désignation du prochain point de collecte

La désignation du prochain point de collecte pour le sink est réalisée à l'aide d'une stratégie intelligente reposant sur le Q-learning. À chaque moment, le sink analyse les diverses actions possibles (c'est-à-dire les emplacements vers lesquels il pourrait évoluer) en

se basant sur une fonction de récompense tenant compte de plusieurs éléments : l'énergie restante des capteurs adjacents, le nombre de paquets en attente, la distance à parcourir par le sink mobile et l'efficacité énergétique globale. Cette approche allie à la fois l'exploration de nouveaux espaces et l'exploitation des trajectoires déjà identifiées comme les plus efficaces, grâce à une politique ϵ -greedy. Le choix final du prochain point de collecte est donc le résultat d'un apprentissage progressif qui permet au sink de s'ajuster aux changements du réseau.

L'algorithme d'apprentissage utilisé par la première approche est donné par l'algorithme 2 suivant :

Algorithm 2 Algorithme de Q-learning pour la sélection du prochain point de collecte

Entrées : Nombre d'épisodes N , Taux d'apprentissage α , Facteur d'actualisation γ , Taux d'exploration ε

Sorties : Table Q optimale $Q(s, a)$ pour la stratégie de déplacement du sink

Initialiser arbitrairement $Q(s, a)$ pour tous les états s (position du sink + état des nœuds) et toutes les actions a (nœuds possibles)

for chaque épisode de 1 à N **do**

Initialiser la position du sink et l'état du réseau s

while $Epuisement=false$ **do**

Choisir une action a selon la politique ε -greedy :

Avec une probabilité ε , sélectionner un nœud voisin aléatoire (exploration)

Sinon, sélectionner $a = \arg \max_a Q(s, a)$ (exploitation)

Déplacer le sink prêt du nœud sélectionné a Observer la récompense r basée sur :

- Un grand nombre de paquets (buffer chargé) au niveau du nœud visité (récompense positive)
- Une faible énergie restante au niveau du nœud visité (pénalité)
- Une faible distance parcourue (récompense positive)

Observer le nouvel état s' (PCj , niveaux de charge et niveau d'énergie mis à jour)

Mettre à jour la valeur Q : $Q(s, a) \leftarrow Q(s, a) + \alpha[r + \gamma \max_{a'} Q(s', a') - Q(s, a)]$

$s \leftarrow s'$

4.3.3 Caractéristiques de l'algorithme de routage utilisé

Dans les RCSFs, le trajet des paquets de données représente une partie cruciale pour garantir une communication efficace entre les nœuds de capteurs et le sink. En raison des capacités limitées des capteurs, notamment en matière d'énergie, il est essentiel d'élaborer des mécanismes de routage qui prennent en considération la consommation d'énergie, la fiabilité des itinéraires de transmission et l'état du réseau. Un algorithme de routage efficace doit non seulement assurer la livraison des données mais aussi équilibrer l'utilisation des nœuds pour prévenir une dégradation prématurée du réseau.

Dans notre solution, l'algorithme de routage que nous avons implémenté s'appuie sur une version modifiée de l'algorithme de Dijkstra. Ce dernier permet d'identifier le chemin le plus efficace entre un nœud source et le sink, ou vers un nœud relais proche du sink. Contrairement à la méthode traditionnelle de Dijkstra qui se base uniquement sur la distance, notre version altérée prend en compte un paramètre lié à l'énergie restante

des capteurs. Le coût entre deux nœuds augmente si le capteur voisin dispose de peu d'énergie, permettant ainsi de sauvegarder les nœuds vulnérables et de privilégier ceux qui sont encore bien alimentés. En outre, le protocole de routage utilisé évite les capteurs énergétiquement faibles, surchargés ou situés en dehors de la zone de collecte considérée, garantissant ainsi une meilleure fiabilité. Une fois les coûts déterminés, le chemin avec le coût total le plus bas est reconstruit jusqu'au point de collecte optimal, positionné le plus près possible du sink. Cette méthode permet de maintenir un bon équilibre énergétique au sein du réseau tout en favorisant un routage dynamique et robuste face aux déplacements du sink.

La détermination du chemin vers le sink par les nœuds est réalisée avec la fonction de routage "Dijkstra-Chemin(cible)" décrite par l'algorithme 3 présenté ci-dessous :

Algorithm 3 Routage par Dijkstra-Chemin

Entrées : Nœud source u , Énergie E , Tampons B , Portée R , Position du sink (j)

Sortie : Chemin optimal du nœud u vers un nœud proche du sink

Initialiser $dist[i] \leftarrow \infty$ et $prev[i] \leftarrow -1$ pour tous les nœuds $dist[u] \leftarrow 0$ Initialiser la file de priorité $PQ \leftarrow \{(0, u)\}$

while $PQ \neq \emptyset$ **do**

 Extraire le nœud v de plus petite distance

if $d(v, j) \leq R/2$ **then**

 └ Reconstruire et retourner le chemin en remontant via $prev$

for chaque voisin i de v **do**

if $E[i] < 0.1 \cdot E_{\max}$ **ou** $B[i] \geq B_{\max}$ **then**

 └ **continuer**

 Calculer le coût :

$$\text{coût} = dist[v] + 0.6 \cdot \frac{d(v, i)}{E[i]} + 0.3 \cdot \frac{B[i]}{B_{\max}} - 0.1 \cdot \frac{d(i, j)}{R}$$

if $\text{coût} < dist[i]$ **then**

 └ $dist[i] \leftarrow \text{coût}$ $prev[i] \leftarrow v$ Insérer $(\text{coût}, i)$ dans PQ

return \emptyset (aucun chemin trouvé)

L'algorithme de Dijkstra que nous avons utilisé a été modifié pour mieux répondre aux contraintes des RCSF avec un sink mobile. Contrairement à la version classique qui cherche à atteindre une destination fixe en minimisant la distance, notre version introduit un critère d'arrêt dynamique : la recherche s'interrompt dès qu'un nœud est situé à une distance inférieure ou égale à la moitié de la portée de transmission du sink, permettant ainsi une transmission directe et réduisant le nombre de sauts. Par ailleurs, la fonction de coût a été enrichie en intégrant plusieurs paramètres, à savoir la distance entre les nœuds, l'énergie résiduelle du voisin, le taux d'occupation de son buffer, ainsi que sa proximité par rapport au sink. Cette pondération permet d'éviter les nœuds faibles ou surchargés, en leur attribuant un coût plus élevé dans le calcul du chemin. De plus, les nœuds dont l'énergie est inférieure à 10% de leur réserve énergétique maximale ou dont le buffer est saturé sont automatiquement exclus du routage. Ces ajustements rendent le mécanisme de sélection de chemin plus intelligent, plus fiable, et mieux adapté aux

variations de topologie et d'énergie dans le réseau, ce qui contribue à équilibrer la charge et à prolonger la durée de vie globale du système.

Dans le pire des cas, la construction d'un arbre de plus courts chemin selon l'algorithme de routage présenté a une complexité temporelle de $O(N^2)$.

4.3.4 Stratégie globale de collecte de données

La stratégie globale de collecte des données mise en œuvre dans notre solution vise à améliorer la collecte et l'extraction des données produites par les capteurs dispersés sur le champ de capture. Au lieu de s'appuyer sur un système centralisé ou un chemin prédéfini, la collecte d'informations est réalisée de manière flexible en alliant le routage réactif multi-sauts à distance limitée au mouvement adaptatif du sink mobile.

Les capteurs envoient leurs paquets directement au sink si celui-ci est à proximité, ou en passant par d'autres nœuds comme intermédiaires pour délivrer les données. De son côté, le sink planifie ses déplacements en tenant compte du niveau énergétique du réseau, de la quantité de données à relayer par les nœuds et des coûts de communication, afin d'optimiser l'efficacité de la collecte tout en minimisant les délais et les pertes de paquets. Cette méthode permet de garantir un équilibrage des charges, d'éviter la surcharge de certains nœuds et de prolonger la longévité générale du réseau.

La stratégie globale de collecte définit aux noeuds capteurs quand et vers où envoyer les paquets de données pour atteindre le sink mobile et quand faudra t-il arrêter cette transmission. Les détails sont présentés ci-dessous :

Communication de l'information de début de transmission : Lorsque le sink atteint un nouveau point de collecte PC_j , il envoie un message à tous les nœuds qui se situent à une distance d de sa position actuelle afin de les inciter à envoyer leurs données. Ces nœuds utilisent alors l'algorithme 3 décrit dans la section 4.3.3 pour découvrir le meilleur chemin vers la nouvelle position du sink PC_j . Une fois la route vers le sink est définie, les nœuds se mettent à transmettre leurs paquets vers ce dernier.

Information d'arrêt de transmission : Le sink s'arrête au niveau du point de collecte (PC_j) sélectionné et se met à récolter les données des nœuds de la nouvelle zone de collecte. La récolte s'arrête après avoir allégé tous les nœuds surchargés ou après une période de séjour fixe et prédéfinie.

Notre stratégie combine ainsi le routage réactif avec un algorithme intelligent qui tente d'optimiser la trajectoire du sink et qui se base sur l'apprentissage par renforcement. La stratégie globale de collecte utilisée par QLB-MOB est illustrée par l'organigramme de la figure (voir Fig. 4.2). De plus, notre stratégie comprend les procédures suivantes :

4.3.4.1 Procédure Lancer-collecte (PC_j)

Cette procédure débute lorsque le sink mobile atteint un nouveau point de collecte PC_j , puis envoie un beacon à l'ensemble des nœuds de la nouvelle zone de collecte pour les inviter à transmettre leurs paquets stockés vers cette position. Par conséquent, les nœuds utilisent leur fonction de routage pour envoyer leurs données vers le sink, comme expliqué dans la section 4.3.3.

A noter que toutes les métriques de performance de l'approche seront calculées durant cette procédure.

4.3.4.2 Procédure Arrêter-collecte (PC_j)

La collecte de données est interrompue pour permettre au sink d'atteindre le prochain point en exécutant l'algorithme Q-learning 2. Lors du déplacement d'un point de collecte à un autre, les nœuds ne peuvent recevoir aucun autre paquet de données. Cette procédure sera initiée par le sink mobile, qui transmettra un message à l'ensemble des nœuds capteurs de sa zone de collecte. Ce message aura pour objectif d'inciter les nœuds à arrêter la transmission des données vers le point courant actuel PC_j . La procédure de collecte est interrompue après la collecte d'un pourcentage spécifié de paquets de données et lorsque les nœuds sont déchargés. Par la suite, les nœuds capteurs devront stocker temporairement leurs paquets et attendre la prochaine invitation du sink pour reprendre le processus de transmission des données.

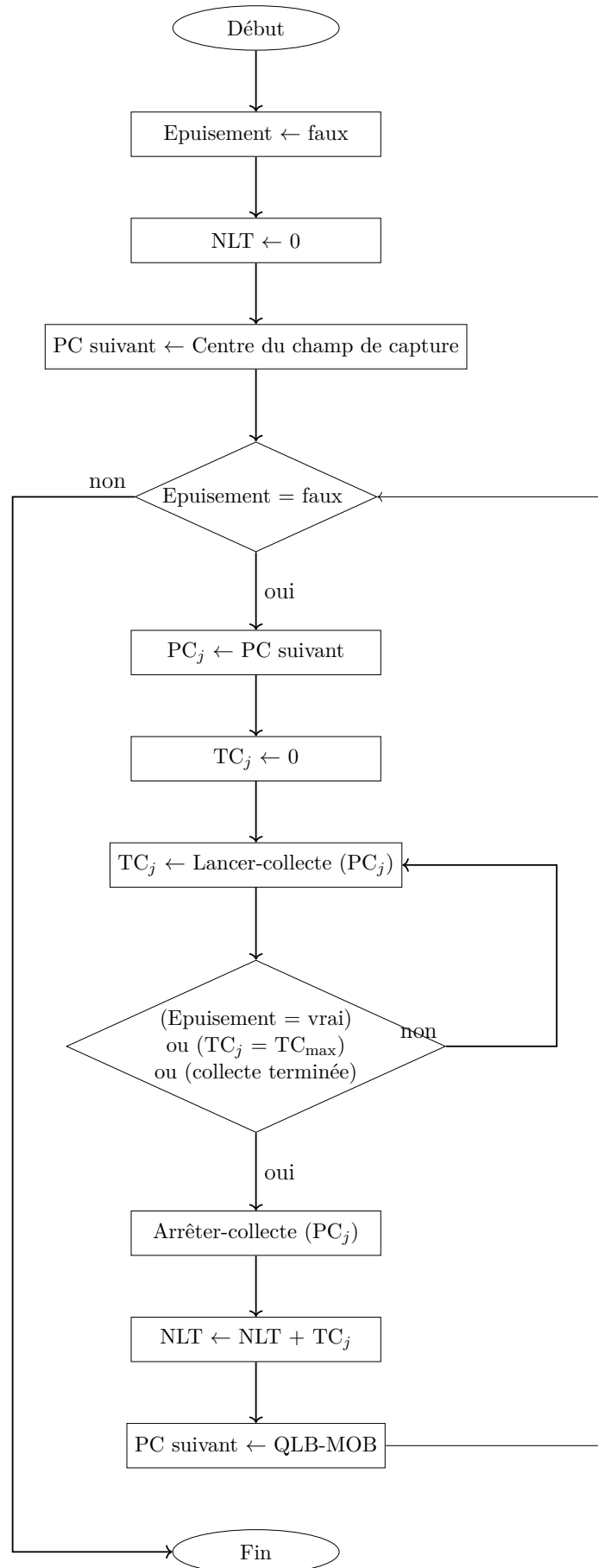


FIG. 4.2 – Organigramme de la stratégie de collecte basée sur QLB-MOB

4.3.5 Description de la deuxième solution : Deep Q-Learning based sink MOBilty (DQL-MOB)

4.3.5.1 Caractéristiques du modèle de mobilité utilisé par la solution DQL-MOB

Trajectoire du sink et nature du mouvement

Dans cette deuxième solution, le réseau est considéré comme plat et constitué de N nœuds déployés de manière aléatoire, sans aucune structuration logique préétablie. La topologie est donc non hiérarchisée et ne repose sur aucune organisation en cellules ou clusters. La trajectoire du sink consiste en l'ensemble des points de collecte visités (PCj), au niveau desquels le sink s'arrête pendant une période dite **Temps de collecte** (TCj) pour rassembler les données des nœuds se trouvant dans la **zone de collecte** correspondante. Une zone de collecte est définie comme l'ensemble des nœuds du réseau situés à une distance d du point de collecte actuel (PCj). En effet, cette zone est dynamique et change à chaque déplacement du sink mobile.

Le mouvement du sink est ainsi discret et la collecte des données se fait de manière discontinue au niveau de chaque point de collecte PCj .

DQL-MOB (Deep Q-Learning Mobility) repose sur un mécanisme de déplacement intelligent du sink mobile dans un RCSF. Elle vise à maximiser l'efficacité de la collecte de données tout en assurant une répartition équitable de la consommation énergétique et en prolongeant la durée de vie globale du réseau. Contrairement aux méthodes statiques ou tabulaires comme le Q-Learning classique, DQL-MOB s'appuie sur une stratégie d'apprentissage par renforcement profond, permettant de prendre des décisions optimales dans des environnements à grande dimension d'état et à dynamique complexe. Le sink agit comme un agent autonome. À chaque instant t , il observe l'état global du réseau (énergie résiduelle, niveau de remplissage des buffers, position des nœuds, etc.), puis désigne un nœud capteur comme cible de déplacement. Une fois la décision prise, le sink se déplace vers ce nœud, considéré comme point de collecte, et entre dans une période de collecte, pendant laquelle les nœuds voisins lui transmettent leurs paquets. Ce mécanisme décisionnel s'inscrit dans une approche proactive : le sink anticipe les besoins du réseau en planifiant ses déplacements à l'avance, sur la base d'une politique optimisée, plutôt que de réagir uniquement à des événements critiques. Ce comportement permet au sink d'ajuster sa trajectoire en temps réel en fonction des conditions du réseau.

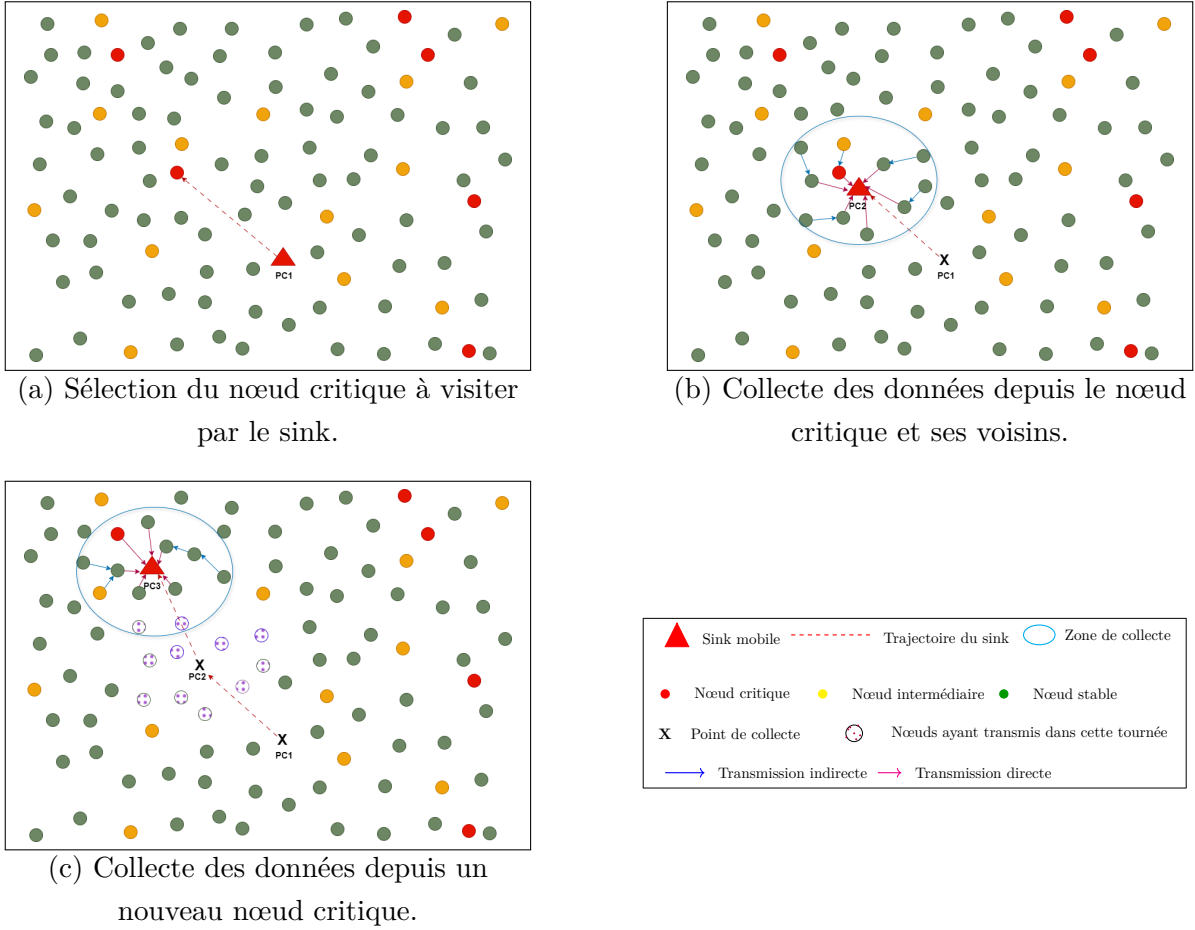


FIG. 4.3 – Modèle de réseau et stratégies de collecte des données.

La zone de collecte associée à un point donné est définie comme l'ensemble des nœuds situés dans un rayon R autour de la position actuelle du sink. Cette zone est dynamique : elle se redéfinit à chaque étape, ce qui confère à la stratégie de déplacement un rôle central dans l'optimisation des performances du réseau. La trajectoire du sink est donc construite pas à pas, en tenant compte de l'historique d'apprentissage et des changements contextuels.

Le choix du nœud cible repose sur un équilibre entre exploration et exploitation. L'agent utilise une politique ϵ -greedy : avec une probabilité ϵ , il sélectionne aléatoirement un nœud admissible (exploration), tandis qu'avec une probabilité $1 - \epsilon$, il choisit celui ayant la plus haute valeur estimée (exploitation). Ce mécanisme permet au sink de découvrir de nouvelles opportunités tout en consolidant ses connaissances sur les zones les plus stratégiques. Le paramètre ϵ décroît au fil des épisodes, ce qui favorise l'apprentissage progressif d'une politique stable.

Pour estimer la valeur des actions possibles, nous avons intégré une architecture Dueling Deep Q-Network (Dueling DQN) [83]. Celle-ci améliore significativement la précision des estimations en séparant la valeur globale de l'état $V(s)$ de l'avantage spécifique d'une action $A(s, a)$. La fonction de valeur Q est alors reconstruite par la relation :

$$Q(s, a) = V(s) + \left(A(s, a) - \frac{1}{|\mathcal{A}|} \sum_{a'} A(s, a') \right)$$

Cette séparation permet au réseau d'évaluer la qualité d'un état même lorsque les différences entre les actions sont faibles, ce qui est fréquent dans les environnements bruités ou à actions redondantes comme les RCSF.

Dans notre solution DQL-MOB, cette architecture permet une meilleure stabilité d'apprentissage, une convergence plus rapide et une décision plus robuste. Elle est intégrée dans notre pipeline d'entraînement, où les transitions (s_t, a_t, r_t, s_{t+1}) sont stockées dans une mémoire d'expérience, puis utilisées pour mettre à jour les paramètres du réseau par descente de gradient sur mini-lots, en parallèle d'un réseau cible.

Stratégie de déplacement du sink : désignation du prochain point de collecte

Le processus de déplacement du sink mobile dans notre solution DQL-MOB est modélisé par un processus de décision de Markov (MDP). Ce cadre mathématique permet de décrire rigoureusement l'interaction entre l'agent (le sink) et l'environnement (le réseau de capteurs), en vue d'apprendre une politique optimale de déplacement maximisant la durée de vie du réseau.

Un MDP est formellement défini par le quintuplet $\langle \mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R}, \gamma \rangle$, où chaque composante est décrite comme suit pour notre cas :

Espace des États (\mathcal{S}) : L'état de l'environnement à un instant t est représenté par un vecteur s_t regroupant les caractéristiques globales et locales du réseau :

- s_1 : Ratio des nœuds vivants ($\in [0, 1]$),
- s_2 : Niveau moyen de remplissage des files (normalisé),
- s_3 : Moyenne de l'énergie résiduelle des nœuds (normalisée),
- s_4 : Écart-type de l'énergie résiduelle (normalisé),
- s_5 : Distance moyenne entre les nœuds et le sink,
- s_6, s_7 : Coordonnées (x, y) normalisées du sink,
- s_8 : Indicateur binaire d'activité (repos) du sink,
- s_9 : Indice d'équité énergétique (indice de Jain),
- s_{10} : Rapport entre buffer utilisé et taille de buffer maximale,
- s_{11} : Log-ratio du buffer pour renforcer les variations faibles.

Ainsi, l'état s_t capture à la fois les conditions énergétiques du réseau, l'état des files et la position du sink.

Espace des Actions (\mathcal{A}) : Une action correspond au choix d'un nœud capteur vers lequel le sink mobile va se déplacer à l'instant t . L'espace des actions est discret, dynamique et dépendant de l'état du réseau : seuls les nœuds vivants, ayant une file partiellement remplie et possédant un niveau énergétique suffisant, sont considérés. Le choix de l'action repose sur la Q-valeur estimée pour chaque nœud, avec une exploration ϵ -greedy durant l'apprentissage.

La fonction de récompense (\mathcal{R}) : La fonction de récompense $\mathcal{R}(s, a)$ joue un rôle central dans l'optimisation de la mobilité du sink, en guidant ses décisions vers les nœuds les plus pertinents à visiter. Elle évalue l'utilité de chaque action en prenant en compte

plusieurs critères critiques du réseau, à savoir la quantité de données récupérées, l'énergie consommée, la mortalité des nœuds et l'équité globale de la consommation énergétique. Elle est définie comme suit :

$$R(s, a) = \alpha \cdot P'_{\text{collecte}} + \rho \cdot R_{\text{protection}} - \beta \cdot E'_{\text{utilisée}} \cdot f(B) - \gamma \cdot N_{\text{morts}} - \lambda \cdot (1 - J)^2$$

où :

- α , β , γ , λ et ρ sont des coefficients de pondération qui équilibrent l'influence de chaque terme,
- $P_{\text{collectés}}$ est le nombre de paquets collectés durant l'action, incitant la collecte efficace,
- $E_{\text{utilisée}}$ représente l'énergie totale consommée pour réaliser l'action,
- $f(B)$ est une fonction non-linéaire du taux de remplissage du buffer, accentuant la pénalité lorsque le tampon est saturé,
- N_{morts} est le nombre de nœuds épuisés pendant l'action, représentant une forte perte pour le réseau,
- J est l'indice d'équité de Jain, mesurant la distribution équitable de la consommation énergétique,
- $R_{\text{protection}}$ est un bonus attribué lorsque l'agent contribue à préserver un nœud critique ou faible en énergie.

Fonction de Transition (T) : La fonction de transition décrit l'évolution de l'environnement suite à une action effectuée par le sink. Elle définit le passage de l'état actuel s_t à l'état suivant s_{t+1} en fonction de l'action a_t choisie par l'agent.

$$\mathcal{T}(s_t, a_t) \rightarrow s_{t+1}$$

l'état du réseau est mis à jour, incluant : la consommation d'énergie, la variation du buffer, l'éventuelle mort de nœuds et la position actuelle du sink.

Facteur d'actualisation (γ) : Un facteur d'actualisation $\gamma \in [0, 1]$ est utilisé pour pondérer les récompenses futures. Typiquement, $\gamma = 0.98$, ce qui accorde une grande importance à la durée de vie du réseau à long terme.

Politique Apprise (π) : La politique π définit la stratégie optimale de l'agent, consistant à choisir, pour chaque état s , l'action a qui maximise la récompense cumulée attendue :

$$\pi^*(s) = \arg \max_{\pi} \mathbb{E} \left[\sum_{t=0}^T \gamma^t R(s_t, a_t) \right]$$

Dans notre approche, cette politique est apprise à l'aide d'un réseau Dueling Deep Q-Network, permettant une prise de décision adaptative basée sur l'estimation des valeurs d'action.

4.3.5.2 Algorithme d'apprentissage du DQN :

Dans notre solution DQL-MOB, la sélection du prochain nœud à visiter par le sink est réalisée à l'aide d'un réseau Dueling Deep Q-Network (DQN), combiné à une fonction hybride de décision intégrant plusieurs critères du réseau. Cette stratégie permet de tenir compte à la fois de la valeur d'action prédite par le DQN, de l'énergie restante, du remplissage des files et de la proximité des nœuds, pour prendre une décision plus robuste et contextuelle.

Les principales variables utilisées dans cette procédure sont :

- $s_t = (E, B, P_j)$: état courant du réseau, incluant les niveaux d'énergie, les tampons, et la position du sink.
- ϵ : paramètre d'exploration utilisé dans la stratégie ϵ -greedy.
- Q_θ : réseau DQN principal évaluant les Q-valeurs pour chaque nœud admissible.
- Q_{θ^-} : réseau cible utilisé pour la mise à jour stabilisée.
- π_i : score hybride associé au nœud i .
- E_i, B_i : énergie résiduel et niveau de tampon du nœud i .
- $d(i, j)$: distance entre le nœud i et le sink.
- $\sigma(\cdot)$: fonction d'écrasement de la variance (sigmoïde ou autre).
- \mathcal{B} : mémoire d'expérience contenant les transitions (s, a, r, s') .
- γ : facteur d'actualisation.

Algorithm 4 Sélection de nœud via DQN avec fonction hybride π_i

Entrées : État du réseau $s_t = (E, B, P_j)$, paramètre d'exploration ϵ

Sortie : Nœud cible a_t sélectionné

Initialiser le réseau principal Q_θ et la mémoire d'expérience $\mathcal{B} \leftarrow \emptyset$

for *chaque épisode* **do**

 Initialiser l'état s_t

while *l'épisode n'est pas terminé* **do**

for *chaque nœud admissible* i **do**

 Calculer le score hybride π_i :

$$\pi_i \leftarrow 0.5 \cdot Q_\theta(s_t, i) + 0.3 \cdot \left(1 - \frac{E[i]}{E_{\max}}\right)^2 + 0.15 \cdot \sigma \left(3 \cdot \left(\frac{B[i]}{B_{\max}} - 0.7\right)\right) + 0.05 \cdot \left(1 - \frac{d(i, j)}{R}\right)$$

if $\text{rand}() < \epsilon$ **then**

 Sélectionner un nœud a_t de manière aléatoire parmi les nœuds admissibles

else

$a_t \leftarrow \arg \max_i \pi_i$

 Exécuter l'action a_t , observer la récompense r_t et le nouvel état s_{t+1} . Ajouter (s_t, a_t, r_t, s_{t+1}) à la mémoire \mathcal{B}

if *la mémoire \mathcal{B} est suffisamment remplie* **then**

 Échantillonner un mini-lot de transitions (s, a, r, s') . Calculer la cible :

$$y \leftarrow r + \gamma \cdot \max_{a'} Q_\theta(s', a')$$

 Mettre à jour Q_θ en minimisant :

$$\mathcal{L} = (y - Q_\theta(s, a))^2$$

 Mettre à jour $s_t \leftarrow s_{t+1}$. Réduire $\epsilon \leftarrow \epsilon \cdot 0.995$

return $\pi^*(s) = \arg \max_a Q_\theta(s, a)$

Cet algorithme repose sur une version hybride du Deep Q-Learning dans laquelle la sélection d'action n'est pas effectuée uniquement à partir des Q-valeurs. Une fonction de score π_i est calculée pour chaque nœud en combinant la valeur apprise $Q_\theta(s, i)$ avec des heuristiques spécifiques à l'environnement : l'inverse de l'énergie résiduelle (priorité aux nœuds faibles), le niveau de saturation du buffer (favorise les nœuds surchargés), et la proximité du sink. Cette approche permet de guider l'apprentissage vers des décisions plus efficaces et réalistes, tout en conservant les propriétés adaptatives du Deep Q-Learning.

4.3.5.3 Caractéristiques de l'algorithme de Routage utilisé par les nœuds

La fonction de routage utilisée dans cette approche est identique à celle utilisée dans la première solution (voir section 4.3.3). Toutefois, afin d'optimiser davantage les performances du réseau et de réduire la quantité de données à transmettre, un mécanisme complémentaire d'agrégation des paquets a été intégré.

Ce mécanisme permet de regrouper les données à chaque nœud relais, tout en respectant la capacité maximale des files, avant de les transmettre vers le sink.

Les principales variables utilisées dans cette procédure sont :

- $P = [i_1, i_2, \dots, i_k]$: Chemin multi-saut menant du nœud source au sink, avec i_k le dernier nœud connecté au sink j .
- B_{\max} : Capacité maximale du tampon de chaque nœud.
- $MAX_AGGREGATION$: Seuil maximal de paquets que le sink peut recevoir en une seule transmission.
- $B[i]$: Quantité actuelle de paquets stockés dans le tampon du nœud i .
- $espace$: Espace libre dans le tampon du nœud suivant j .
- $relayed$: Nombre de paquets transmis du nœud i vers le nœud suivant j .
- $aggregated$: Nombre de paquets transmis du dernier nœud i_k vers le sink j , limité par $MAX_AGGREGATION$.

Algorithm 5 Procédure d'Agrégation et de Transmission des Paquets (DQN)

Entrées :

$P = [P[1], P[2], \dots, P[k]]$: Chemin multi-saut vers le sink ($P[k] = j$)

B_{\max} : Capacité maximale du tampon

$MAX_AGGREGATION$: Seuil maximal de paquets pour le sink

Sortie : Transmission des paquets vers le sink tout en respectant les contraintes de tampon

for $j = 1$ **to** k **do**

```

if  $j = k$  // Dernier nœud connecté au sink then
     $aggregated \leftarrow \min(B[P[j]], MAX\_AGGREGATION)$ 
    Transmettre  $aggregated$  paquets au sink
     $B[P[j]] \leftarrow B[P[j]] - aggregated$ 
else
     $espace \leftarrow B_{\max} - B[P[j + 1]]$ 
     $relayed \leftarrow \min(B[P[j]], espace)$ 
     $B[P[j + 1]] \leftarrow B[P[j + 1]] + relayed$ 
     $B[P[j]] \leftarrow B[P[j]] - relayed$ 

```

L'algorithme parcourt successivement chaque nœud du chemin P . Pour les nœuds intermédiaires (i_1 à i_{k-1}), une quantité maximale de paquets est transférée vers le nœud suivant i_{i+1} , sans dépasser la capacité disponible dans son tampon. Cela permet de limiter la surcharge et de réduire les pertes liées à l'encombrement.

Arrivé au dernier nœud i_k , connecté directement au sink, une agrégation contrôlée est appliquée : seule une quantité maximale de paquets (définie par $MAX_AGGREGATION$) est transmise au sink. Cette approche permet de réduire la congestion, optimiser la consommation énergétique et améliorer la stabilité du routage multi-saut dans le réseau.

4.3.5.4 Algorithme de Routage Hybride

Dans notre solution, la sélection du prochain point de collecte (PC_{suivant}) repose sur une stratégie hybride combinant apprentissage par renforcement et routage classique.

L'agent intelligent (le sink, représenté par le nœud j) sélectionne dynamiquement une cible à visiter via une politique de Q-Learning ou DQN, puis choisit le mode de transmission le plus adapté en fonction de la distance entre la cible et le sink.

Si cette cible est située à une distance inférieure ou égale à la portée R , une **transmission directe** est effectuée. Sinon, un **chemin multi-saut** est calculé à l'aide de l'algorithme de Dijkstra, en tenant compte des contraintes d'énergie et de buffer des nœuds intermédiaires.

Les principales variables impliquées dans cet algorithme sont :

- s_t : état courant du réseau observé par le sink (niveau d'énergie, tampons, position, etc.)
- **cible** : nœud sélectionné comme prochaine destination via apprentissage
- j : position actuelle du sink mobile
- R : portée de communication maximale directe
- **chemin** : séquence de nœuds définissant un itinéraire multi-saut vers la cible
- $d(i, j)$: fonction de distance entre deux nœuds i et j

Algorithm 6 Algorithme de routage hybride basé sur la distance

Entrées : État courant s_t , position actuelle du sink j , portée R

Sortie : Mode de transmission (directe ou multi-saut), et chemin s'il existe

Initialiser **cible** $\leftarrow \emptyset$, **chemin** $\leftarrow \emptyset$

cible \leftarrow DQN-Sélection(s_t) *(nœud choisi par apprentissage)*

if $d(\mathbf{cible}, j) \leq R$ **then**

⌊ **Retourner** Transmission directe à la cible

else

| **chemin** \leftarrow Dijkstra-Chemin(**cible**, j)

| **if** $\mathbf{chemin} \neq \emptyset$ **then**

| ⌊ **Retourner** Transmission multi-saut via **chemin**

| **else**

| ⌊ **Retourner** Aucun chemin valide

Cet algorithme combine l'intelligence de la sélection de cible par apprentissage avec la robustesse du routage multi-saut classique. Il permet au sink de s'adapter aux conditions du réseau tout en assurant une transmission efficace.

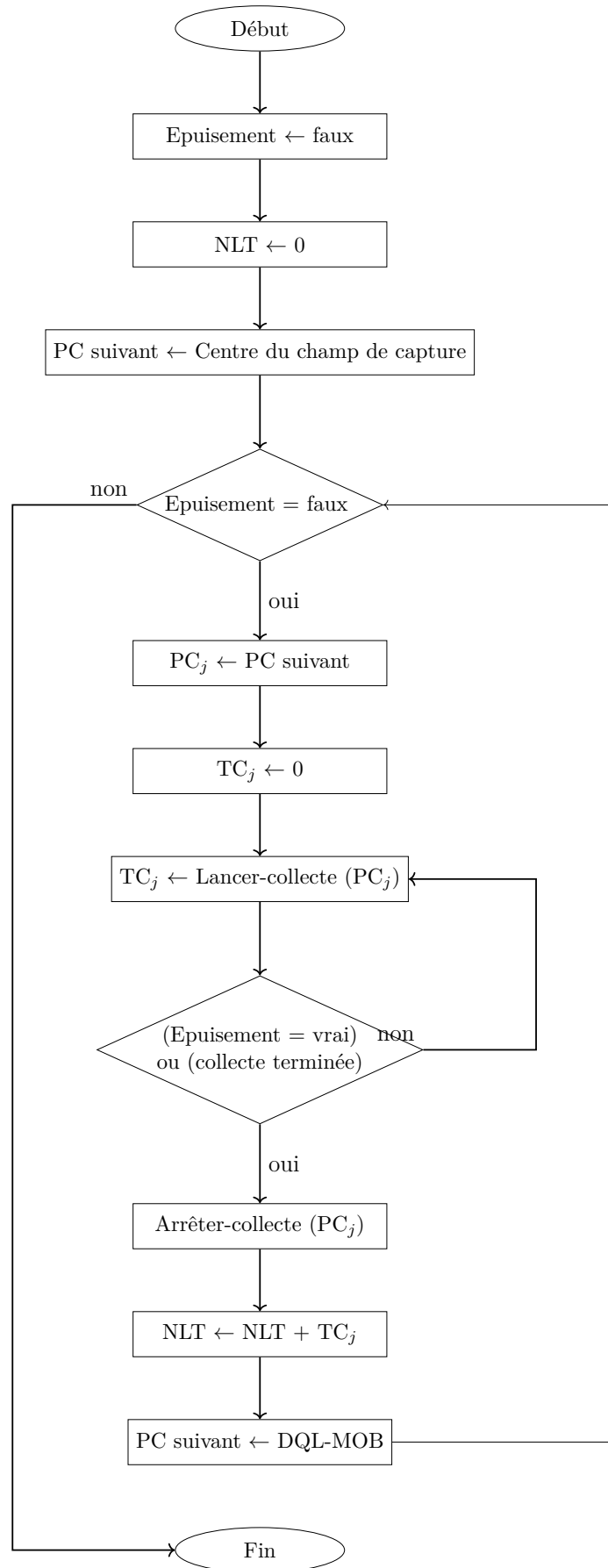


FIG. 4.4 – Organigramme de la stratégie de collecte basée sur DQL-MOB

4.4 Simulation et Résultats

Nous avons évalué les performances de nos approches QLB-MOB et DQL-MOB en les comparant avec des stratégies de collecte existantes, telles que la mobilité aléatoire (Random) et la mobilité circulaire (Circular). Les critères de comparaison incluent la consommation d'énergie, la durée de vie du réseau (NLT), le taux de délivrance de données (PDR), le taux de perte de données (PLR) et l'équilibrage énergétique (fairness) représenté par l'indice de Jain (J). Les différentes expérimentations ont été réalisées en Python.

Nous présentons d'abord une brève description des configurations retenues pour les expérimentations, puis nous discutons et analysons les résultats obtenus après des simulations intensives.

Pour évaluer l'efficacité de nos approches, nous les comparons à deux scénarios de mobilité du sink : un modèle de mobilité aléatoire (Random) et un mouvement circulaire répétitif (Circular) :

- Le modèle de mobilité aléatoire (Random) : dans ce cas, après l'expiration d'une période de collecte choisie aléatoirement, le sink calcule sa nouvelle position de manière aléatoire dans l'ensemble des points de collecte définis au centre des grilles.
- La mobilité circulaire (Circular) : Selon ce scénario, le puits suit un mouvement circulaire et répétitif en passant par les points de collectes définis au centre des grilles. Il se déplace du point j au point $j+1$ à la fin de la période de collecte prédéfinie et invariable.

Les principales caractéristiques des modèles de mobilité des solutions mises en œuvre sont résumées dans le tableau 4.2.

| Stratégie de collecte | Classe de mobilité | Période de collecte | Quand se déplacer ? | Prochaine position |
|--------------------------------|--------------------|-----------------------|------------------------------------|---|
| Mobilité aléatoire (Random) | Aléatoire | Variable Aléatoire | A la fin de la période de collecte | $Position = Random(1..m)$ |
| Mobilité circulaire (Circular) | Predictible | Fixe | A la fin de la période de collecte | $Position = i + 1$ If $Position > m$ then Position=1 |
| QLB-MOB | Controlée | Fixe | Après récolte des données | Depend de l'état des noeuds - Calculé par l'algorithme 2 |
| DQL-MOB | Controlée | Variable | Après récolte des données | Depend de l'état des noeuds - Calculé par l'algorithme 4 |

TAB. 4.2 – Caractéristiques des modèles de mobilité des approches implémentées.

4.4.1 Environnements et outils de simulation

Dans le cadre de cette étude, nous avons opté pour le langage de programmation Python pour implémenter et simuler nos solutions de mobilité adaptative du sink. Le développement a été réalisé dans l'environnement Visual Studio Code (VS Code), qui offre une interface ergonomique, une bonne gestion des extensions Python et un support efficace pour le débogage et la visualisation.

Le choix de Python se justifie par sa simplicité, sa flexibilité, ainsi que sa capacité à intégrer des algorithmes d'apprentissage par renforcement tels que le Q-Learning. Bien que Python ne soit pas le simulateur le plus couramment utilisé dans le domaine des réseaux de capteurs (comparé à NS-3, OMNET++ ou MATLAB), il s'est révélé particulièrement adapté à notre besoin : simuler un comportement intelligent, dynamique et visuel du sink mobile dans un réseau de capteurs sans fils.

4.4.2 Configuration et paramètres de simulation

- **Modèle énergétique :** Dans cette simulation, nous utilisons le modèle radio de premier ordre présenté par Heinzelman [84], qui est fréquemment appliqué pour simuler la consommation d'énergie dans les RCSF. Ce modèle suppose que la consommation d'énergie des circuits électroniques de transmission et de réception est constante par bit traité, tandis que l'énergie nécessaire pour transmettre un bit dépend de la distance entre l'émetteur et le récepteur.

Lorsque la distance de transmission d est inférieure ou égale à un seuil d_0 , la propagation du signal suit un modèle en espace libre, où l'énergie est proportionnelle à d^2 . Si la distance dépasse ce seuil, un modèle de propagation multi-trajets est utilisé, dans lequel l'énergie est proportionnelle à d^4 .

L'énergie nécessaire pour transmettre k bits sur une distance d est donnée par :

$$E_{TX}(k, d) = \begin{cases} E_{elec} \cdot k + E_{fs} \cdot d^2 \cdot k, & \text{si } d \leq d_0 \\ E_{elec} \cdot k + E_{amp} \cdot d^4 \cdot k, & \text{si } d > d_0 \end{cases} \quad (4.2)$$

$$E_{RX}(k) = E_{elec} \cdot k$$

Les différents paramètres utilisés sont :

- $E_{TX}(k, d)$: énergie consommée pour transmettre k bits à une distance d ,
- $E_{RX}(k)$: énergie consommée pour recevoir k bits,
- E_{elec} : énergie utilisée pour le traitement électronique (codage, modulation, filtrage...),
- E_{fs} : facteur d'atténuation dans le modèle en espace libre,
- E_{amp} : facteur d'atténuation dans le modèle multi-trajets,
- d_0 : distance seuil, au-delà de laquelle le modèle multi-trajets est utilisé. Elle se calcule comme suit :

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

4.4.3 Configuration du réseau

Nous faisons les hypothèses suivantes concernant la configuration du réseau :

- Chaque nœud capteur possède un identifiant unique prédéfini **id** et une quantité d'énergie initiale égale à **E₀**.
- Un paquet de données est généré par chaque capteur à intervalle régulier, appelé **intervalle de génération de données (T)**.
- L'émetteur-récepteur suit le modèle radio du premier ordre de Heinzelman [84], où la dissipation d'énergie pour l'émission ou la réception est constante par bit transmis. De plus, l'énergie dépensée pour transmettre un bit sur une distance **d** est proportionnelle à **d²**.
- Nous utilisons un protocole MAC sans collision, garantissant un accès au canal pour tous les nœuds et évitant la congestion au niveau des liens.
- Tous les nœuds capteurs ont la même portée de transmission **R**.
- Pour les expérimentations, la zone de capture est carrée, de diamètre **L**.
- Les nœuds capteurs sont répartis dans la zone de capture de façon aléatoire.
- Le puits (*sink*) peut ajuster sa portée de communication pour atteindre directement n'importe quel nœud du réseau.
- Le puits se déplace librement et peut s'arrêter à n'importe quel endroit du champs de capture.

Les principaux paramètres utilisés pour les simulations sont présentés dans la table

| Paramètre | Valeur |
|--|--------------------------|
| Longueur d'un paquet de données | 2000 octets (16000 bits) |
| Protocole de routage | Dijkstra adapté |
| E_{elec} (Énergie consommée par circuit) | 50 nJ/bit |
| E_{Recv} (Énergie pour recevoir 1 bit) | 50 nJ |
| E_{Trans} (Énergie pour transmettre 1 bit) | 50 nJ |
| E_{fs} (Facteur d'atténuation du signal dans le modèle d'espace libre) | 10 pJ |
| E_{amp} (Facteur d'atténuation du signal dans le modèle multi-trajet) | 0.0013 pJ |
| E_0 (Énergie initiale) | 1,0 J |
| N (Nombre de nœuds) | 80 |
| R (Portée de transmission) | 20 m |
| L (Taille du champ de déploiement) | 100 × 100 m |
| Taille du tampon (buffer) | 50 paquets |
| Débit de transmission | 1 Mbps |

TAB. 4.3 – Paramètres de simulation

4.4.4 Métriques de performances

Les performances des trois modèles de mobilité ont été évaluées à l'aide des métriques suivantes :

- **Durée de vie du réseau (NLT – *Network Lifetime*)** : Plusieurs définitions de la durée de vie du réseau existent. Nous la définissons comme le temps écoulé jusqu'à la défaillance (épuisement de son énergie) d'un certain pourcentage des nœuds du réseau. Cette approche tient compte de la capacité du sink à collecter des données auprès de n'importe quel nœud actif du réseau. Dans notre cas, la durée de vie du réseau correspond au nombre de tournée du sink avant l'épuisement du pourcentage défini de nœuds. Nous définissons une tournée comme le temps nécessaire pour collecter les données de tous les nœuds du réseau.
- **Le Taux de Livraison des Paquets (Packet Delivery Ratio ou PDR)** : représente une mesure de performance cruciale dans les RCSFs, car il évalue la fiabilité de la transmission des données du point de vue du récepteur, c'est-à-dire du sink.

Le PDR traduit la proportion de paquets produits par les nœuds capteurs qui ont été correctement reçus par le sink. Il représente ainsi l'aptitude du réseau à transmettre les informations de manière fiable et efficace, sans pertes, en dépit des limitations du réseau telles que la mobilité du sink, les interférences ou la consommation d'énergie.

$$\text{PDR} = \left(\frac{\text{Nombre de paquets reçus}}{\text{Nombre de paquets envoyés}} \right) \times 100\% \quad (4.4)$$

- **Le Taux de Perte de Paquets (PLR)** : Le taux de perte de paquets (Packet Loss Ratio) constitue une mesure fondamentale pour analyser les performances des

réseaux de capteurs sans fil (WSN). Il permet d'évaluer la fraction de paquets de données émis par les capteurs qui échouent à atteindre le sink, que ce soit à cause de collisions, d'interférences, de pertes de connexion ou de l'épuisement de l'énergie des nœuds. De ce fait, le PLR se présente comme un indicateur immédiat de la fiabilité du réseau et, par conséquent, de l'efficacité du schéma de mobilité du sink à maintenir une connectivité de qualité avec les nœuds présents dans le réseau. Un PLR élevé indique une détérioration des performances, tandis qu'un PLR faible suggère une communication efficace et fiable entre les capteurs et le sink.

$$\text{PLR} = \left(\frac{\text{Nombre de paquets perdus}}{\text{Nombre de paquets envoyés}} \right) \times 100\% \quad (4.5)$$

- **Énergie Totale Consommée (ETC)** : est une métrique fondamentale qui mesure la quantité totale d'énergie dépensée par l'ensemble des nœuds du réseau. Elle reflète l'**efficacité énergétique globale** et permet de comparer différentes stratégies de communication. Elle est généralement calculée par la formule suivante :

$$ETC = \sum_{i=1}^N (E_{\max}^i - E_i) \quad (4.6)$$

où :

N = nombre total de nœuds

E_{\max}^i = énergie initiale du nœud i

E_i = énergie résiduelle du nœud i

- **Équilibrage énergétique (Fairness)** : Elle évalue dans quelle mesure la consommation d'énergie est équitablement distribuée entre les nœuds d'un réseau. Cet indicateur aide à repérer les déséquilibres susceptibles de provoquer des trous énergétiques (nœuds surchargés).

L'indice de Jain est la métrique de fairness la plus utilisée :

$$J = \frac{\left(\sum_{i=1}^N E_i \right)^2}{N \cdot \sum_{i=1}^N E_i^2} \in [0, 1] \quad (4.7)$$

où :

— N : nombre total de nœuds

— E_i : énergie résiduelle du nœud i

Interprétation :

— $J = 1$: Équité parfaite (tous les nœuds ont la même énergie)

— $J \rightarrow 0$: Fort déséquilibre (certains nœuds sont surutilisés)

— Seuil acceptable : $J > 0.75$

4.4.5 Présentation et discussion des résultats

La figure 4.5 illustre une comparaison de la consommation d'énergie des protocoles QLB-MOB, DQL-MOB, Random et Circular en fonction de la taille du buffer des nœuds capteurs. La taille du tampon (buffer) correspond au nombre maximal de paquets qu'un nœud capteur peut stocker dans sa file d'attente. Chaque nœud doit y conserver à la fois les paquets qu'il génère lui-même et ceux qu'il reçoit de ses nœuds voisins. L'ensemble de ces paquets doit ensuite être transmis au sink. Lorsqu'un paquet est transmis avec succès, il est supprimé de la file d'attente du nœud émetteur.

Durant ces de simulation, la taille du tampon a été progressivement variée entre 30 et 120 paquets, afin d'évaluer l'impact de ce paramètre sur les différentes métriques de performance. La figure 4.5 montre qu'en termes de consommation énergétique, QLB-MOB et DQL-MOB présentent de meilleures performances que les deux approches aléatoire et circulaire.

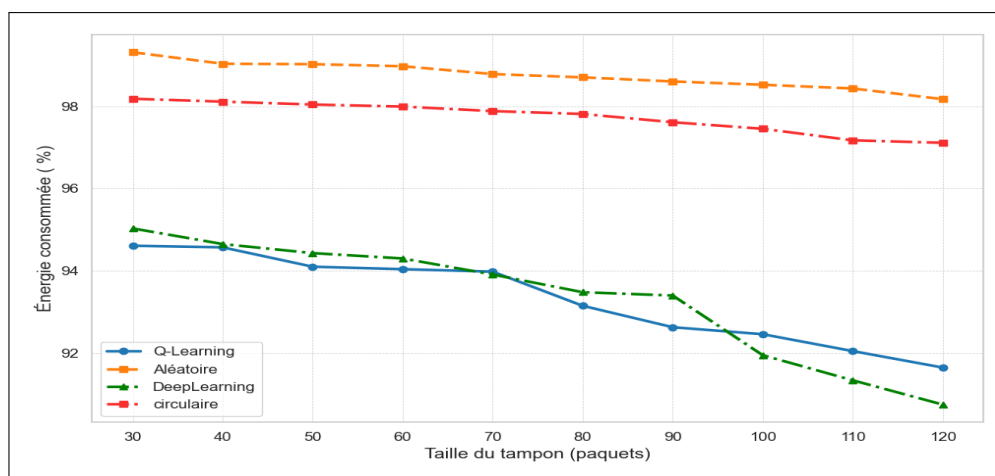


FIG. 4.5 – Énergie Totale Consommée (ETC).

Nos approches QLB-MOB et DL-MOB permettent d'équilibrer la charge entre les différents nœuds du réseau en ajustant intelligemment la position du sink grâce à l'utilisation de l'apprentissage par renforcement. En se rapprochant des nœuds les plus chargés, le sink peut collecter leurs données avec un nombre réduit de sauts, limitant ainsi le transit des paquets à travers des nœuds intermédiaires. De plus, la stratégie de collecte restreint le routage aux seuls nœuds situés dans une zone de collecte définie par un rayon d autour de la position actuelle du sink. Cette double optimisation permet non seulement de réduire la distance parcourue par les paquets, mais également de diminuer significativement la consommation d'énergie au niveau des nœuds, contribuant ainsi à une meilleure efficacité énergétique du réseau.

Cet équilibrage de charge est confirmé par la figure 4.6 qui montre l'impact de la variation de la taille du buffer sur l'évolution de l'indice J (fairness) pour les quatre approches comparées. D'après les résultats obtenus dans la figure 4.6, l'indice de fairness J est supérieur à 0.75 et augmente avec la taille du buffer, atteignant une

valeur supérieure à 0.9. Ceci signifie que nos approches répartissent équitablement la charge des nœuds en combinant une stratégie intelligente pour le positionnement du sink avec une politique de routage qui évite l'utilisation des nœuds à faible énergie (ce qui implique qu'ils ont réalisé plusieurs transmissions) comme intermédiaires vers le récepteur.

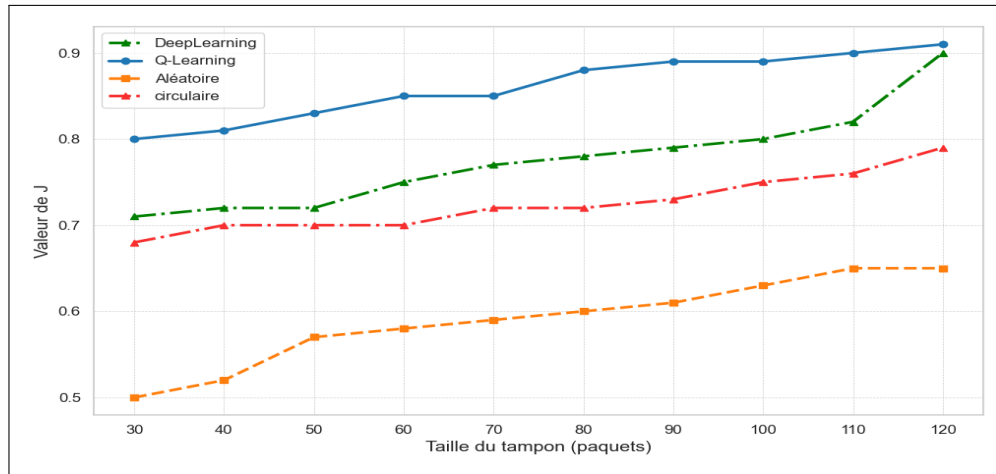


FIG. 4.6 – Équilibrage énergétique (Fairness) .

La figure 4.7 montre l'effet de la variation de la taille du buffer sur la longévité du réseau pour les quatre approches comparées. Nous remarquons d'abord l'augmentation de la durée de vie du réseau avec l'augmentation de la taille du buffer pour toutes les approches étudiées. Quand un paquet ne parvient pas à sa destination, il est nécessaire de le retransmettre par le nœud source, ce qui entraîne une consommation d'énergie supplémentaire. Cette surconsommation accélère l'épuisement des nœuds, diminuant ainsi leur durée de vie. Par contre, augmenter la taille du buffer augmente la capacité à gérer le trafic, réduit les pertes et minimise les retransmissions, conduisant à une prolongation de la durée de vie du réseau. La réduction de l'énergie consommée par notre approche est aussi justifiée l'équilibrage de charge qu'elles réalisent.

La figure 4.7 montre aussi que nos solutions améliorent considérablement la durée de vie globale du réseau par rapport à l'approche aléatoire et l'approche circulaire. Ceci est justifié par le positionnement intelligent du sink qui évite d'utiliser les nœuds à faible énergie comme relais. En effet, si l'arbre de propagation des données vers le sink comprend des nœuds à faible énergie résiduelle, en particulier lorsqu'ils occupent des positions proches de la racine de l'arbre (c'est-à-dire en tant que relais principaux), la récompense est pénalisée. Cette stratégie incite le sink à adapter sa position afin de contourner ces nœuds vulnérables, évitant ainsi de les surcharger. Ce mécanisme contribue à préserver leur énergie, prolongeant ainsi non seulement leur durée de vie, mais également celle du réseau dans son ensemble.

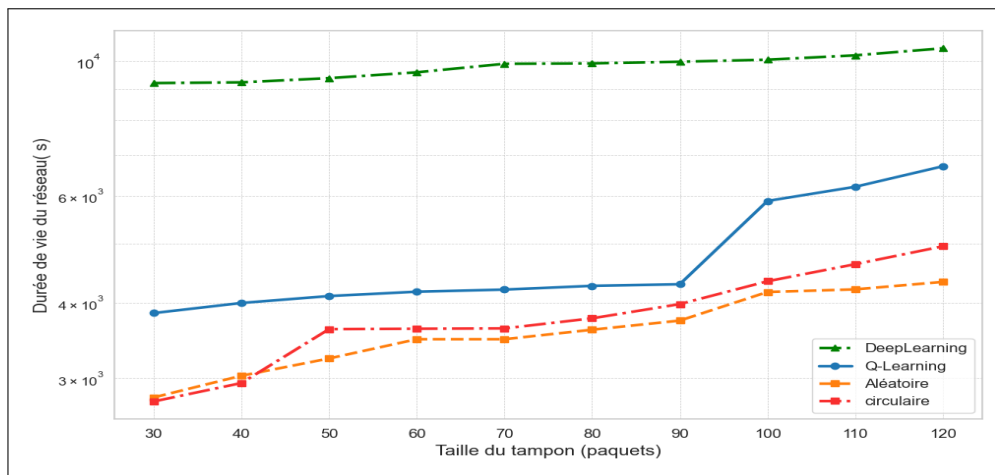


FIG. 4.7 – Durée de vie du réseau en fonction de la taille du tampon.

Figure 4.8 illustre la variation du taux de livraison des paquets (PDR) des quatre stratégies en fonction de la taille du buffer. Nous constatons que le taux de succès de transmission des paquets est directement lié à la taille du tampon. En réalité, plus la taille du tampon du nœud destinataire est grande, plus il y a de chances qu'un message envoyé à ce nœud puisse être intégré dans sa file d'attente, ce qui augmente le PDR. Ces résultats sont validés par les courbes de la figure 4.8 qui analysent l'impact de la variation de la taille du buffer sur le taux de perte de paquets (PLR).

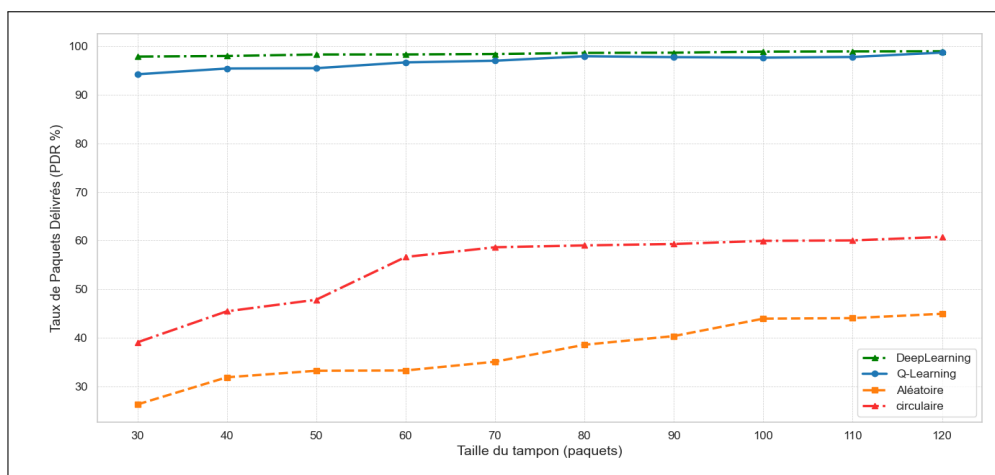


FIG. 4.8 – Taux de livraison des paquets en fonction de la taille du tampon.

Les figures 4.7 et 4.8 montrent aussi que QLB-MOB et DQL-MOB donnent de meilleures performances en terme de taux de succès de délivrance des paquets et de perte de paquets par rapport aux deux autres approches comparatives. En effet, nos stratégies de collecte rapprochent le sink des nœuds les plus chargés en données pour collecter leurs paquets par communication directe ou via un nombre réduit de sauts. Cette proximité améliore la fiabilité des transmissions, ce qui se traduit par une augmentation du PDR. Par ailleurs, cette stratégie de positionnement intelligent permet d'alléger la charge des nœuds et de maintenir un espace libre dans leur file pour l'insertion des nouveaux messages qu'ils soient générés ou à relayer. Ce qui

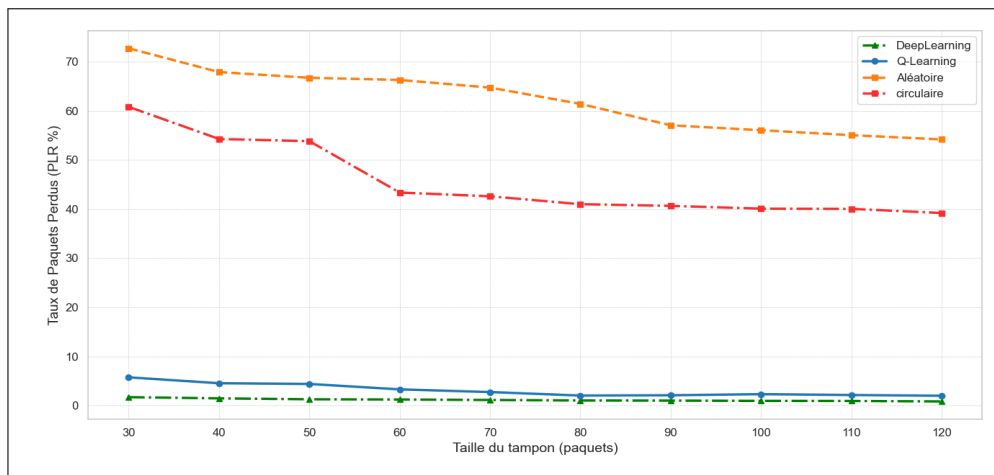


FIG. 4.9 – Taux de perte des paquets en fonction de la taille du tampon.

implique une réduction considérable du PLR.

Les résultats de simulations montrent aussi que l'approche DQL-MOB dépasse systématiquement QLB-MOB en termes de taux de livraison (PDR), de taux de perte (PLR) et de durée de vie du réseau (NLT). Cela s'explique par la capacité du Deep Q-Learning à prendre des décisions plus globales et optimisées grâce à la généralisation offerte par le réseau de neurones. Le modèle DQL prend en compte plusieurs caractéristiques (énergie, charge, distance) pour sélectionner les nœuds critiques, ce qui permet une collecte de données plus fiable et efficace, avec moins de sauts et moins de pertes. De plus, la solution DQL-MOB utilise une fonction d'agrégation qui a pour effet de réduire le volume de données à transmettre et ainsi l'énergie dépensée au niveau des nœuds, ce qui contribue à prolonger la durée de vie globale du réseau.

En revanche, une analyse plus fine des figures 4.5 et 4.6 montre que QLB-MOB conserve certains avantages spécifiques. Sur la figure 4.5, on constate que QLB-MOB consomme légèrement moins d'énergie moyenne que DQL-MOB, en particulier lorsque la taille du buffer augmente. Cette différence peut être attribuée à la simplicité algorithmique du Q-learning, qui repose sur une table Q statique et entraîne une charge computationnelle minimale. En revanche, DQL-MOB utilise un réseau de neurones profond pour approximer les valeurs Q, ce qui permet de mieux généraliser les décisions à partir de caractéristiques complexes de l'environnement (énergie, buffer, distance), mais implique parfois une exploration plus coûteuse. De plus, la figure 4.6 montre que l'indice d'équité (J) est systématiquement plus élevé pour QLB-MOB. Cela indique que cette approche répartit la charge plus uniformément entre les nœuds, tandis que DQL-MOB, en recherchant les solutions optimales, tend à exploiter les nœuds les plus performants, ce qui peut entraîner une utilisation légèrement moins équitable des ressources réseau par rapport à QLB-MOB. Ainsi, bien que DQL-MOB démontre des capacités supérieures d'apprentissage, d'adaptation et d'optimisation, QLB-MOB reste compétitif sur le plan de l'efficacité énergétique et d'équité de répartition de charge. Le choix entre les

deux approches dépend donc des priorités du réseau : performance globale maximale ou durabilité équitable des ressources.

4.5 Conclusion

Ce chapitre a présenté deux approches intelligentes de déplacement d'un sink mobile dans un réseau de capteurs sans fil, basées sur l'apprentissage par renforcement : QLB-MOB (Q-Learning) et DQL-MOB (Deep Q-Learning). Après avoir formalisé le problème en tant que processus de décision markovien, nous avons proposé des stratégies de déplacement adaptatif et de routage efficaces, tenant compte de l'état du réseau (énergie, distance, congestion).

Les résultats de simulation ont montré que ces approches améliorent significativement les performances du réseau, notamment en termes de taux de livraison, consommation énergétique et durée de vie. Ces travaux démontrent le potentiel de l'intelligence artificielle pour optimiser la collecte de données dans les réseaux de capteurs.

Conclusion Générale

Les RCSF constituent actuellement une technologie clé dans divers secteurs, comme la surveillance de l'environnement, l'agriculture de précision, la sécurité et les applications militaires. Composés de nombreux capteurs répartis dans une zone géographique, ces réseaux permettent de recueillir et transmettre des données vers un ou plusieurs nœuds collecteurs, désignés sous le terme de sinks. Cependant, lorsque le sink demeure fixe, les capteurs voisins sont surchargés, ce qui entraîne une utilisation inégale de l'énergie, une congestion accrue et une diminution de la longévité du réseau. Pour remédier à cette situation, l'intégration de la mobilité du sink est apparue comme une solution viable pour optimiser la performance et l'équilibre énergétique du réseau.

Dans ce cadre, ce mémoire visait principalement à élaborer et à évaluer une stratégie de mobilité adaptative du sink, basée sur des mécanismes d'intelligence artificielle. L'objectif était de permettre au sink de sélectionner intelligemment ses points de collecte en tenant compte de divers critères tels que la stabilité des flux de données, l'énergie restante des capteurs, la distance de communication et le degré de saturation du réseau. Le principal défi était d'étendre la durée de vie du réseau, de réduire les pertes de données et de garantir une transmission des informations plus fiable.

Pour réaliser cet objectif, deux stratégies ont été mises en place. La première, appelée QLB-MOB (Mobilité du Sink Basée sur Q-Learning), repose sur un algorithme d'apprentissage par renforcement permettant au sink d'apprendre progressivement à choisir les meilleurs emplacements de collecte selon l'état du réseau. La seconde, qui est plus sophistiquée, se nomme DQL-MOB (Mobilité du Sink Basée sur Deep Q-Learning) et utilise le Deep Q-Learning, intégrant un réseau de neurones pour traiter des scénarios plus complexes et dynamiques. Ces deux approches emploient des techniques d'apprentissage pour ajuster les déplacements du sink en fonction des besoins réels du réseau, tout en prenant en compte les contraintes énergétiques et structurelles des RCSF.

Le projet a été organisé en plusieurs phases : la conception du réseau et des paramètres décisionnels, la création d'une plateforme de simulation en Python, et enfin, l'évaluation expérimentale des performances obtenues. La plateforme comprend une interface graphique, une animation du mouvement du sink et un système de routage dynamique, fournissant ainsi un environnement complet pour analyser les différentes stratégies. Une série de tests a été réalisée pour évaluer les performances de QLB-MOB et DQL-MOB par rapport à deux méthodes traditionnelles : la mobilité aléatoire et la mobilité circulaire.

L'évaluation a examiné divers indicateurs clés, tels que le taux de livraison des paquets (PDR), le taux de perte (PLR), la durée de vie du réseau (NLT), la consommation d'énergie et l'indice d'équité énergétique (J-fairness). Les résultats obtenus indiquent clairement que les deux approches proposées surpassent les modèles traditionnels, avec une amélioration significative dans l'ensemble des indicateurs analysés.

Plus spécifiquement, l'approche DQL-MOB s'est avérée la plus efficace, réussissant à équilibrer à la fois la charge entre les capteurs, à prolonger considérablement la durée de vie du réseau et à maintenir un taux de livraison élevé, même dans des environnements complexes.

Les résultats prometteurs obtenus à travers cette étude ouvrent la voie à plusieurs perspectives de recherche. Tout d'abord, une extension naturelle consisterait à adapter l'approche proposée à un environnement multi-sink, permettant ainsi d'explorer la coordination et l'interaction entre plusieurs collecteurs intelligents au sein de réseaux de grande échelle. Par ailleurs, l'intégration de techniques d'apprentissage profond plus avancées, telles que les réseaux neuronaux récurrents (RNN) ou les modèles d'attention, pourrait renforcer la capacité du système à prédire les zones critiques et à s'adapter de manière plus fine aux évolutions dynamiques du réseau. En outre, une validation expérimentale sur des plateformes physiques (telles que des objets connectés, Raspberry Pi ou cartes IoT) constituerait une étape importante pour évaluer la robustesse, la scalabilité et la faisabilité pratique des solutions proposées dans un contexte réel. Enfin, la sécurité des communications dans un environnement où le sink est mobile et intelligent représente un défi majeur, nécessitant l'intégration de mécanismes de protection contre les attaques et de gestion des identités.

Bibliographie

- [1] A. B. et K. HAMIDATOU, “Une nouvelle technique d’optimisation pour l’économie d’énergie dans le réseau de capteur sans fils,” mémoire de master en informatique, Université Ahmed Draia - Adrar, 2018.
- [2] S. R. N. A. K. K. G. Sahar, K. Abu Bakar and T. Bibi, “Recent advancement of data-driven models in wireless sensor networks : A survey,” *Technologies*, vol. 9, no. 4, p. 76, 2021.
- [3] S. Jain, R. K. Verma, K. K. Pattanaik, and A. Shukla, “A survey on event-driven and query-driven hierarchical routing protocols for mobile sink-based wireless sensor networks,” *The Journal of Supercomputing*, vol. 78, no. 8, pp. 11492–11538, 2022.
- [4] “Les réseaux de capteurs - ppt video online télécharger.” <https://slideplayer.fr/slide/1319604/>.
- [5] G. YAMINA, “rotocolos de communication et optimisation de l’énergie dans les réseaux de capteurs sans fil,” mémoire de master en génie Électrique, Universitaire Belhadj Bouchaib d’Ain-Temouchent.
- [6] Université de technologie de Compiègne, “Réseaux de capteurs sans fils – applications des rcsf.” https://moodle.utc.fr/file.php/498/SupportWeb/co/Module_RCSF_34.html. Module_RCSF_34, accès restreint.
- [7] Parametric Architecture, “Smart home technology interface,” 2023. Accessed : 2025-04-11. <https://parametric-architecture.com/wp-content/uploads/2023/05/Smart-Home-Technology1.png>,.
- [8] Cyberjustice, “Smart cities concepts,” 2019. Accessed : 2025-04-11. <https://cyberjustice.blog/wp-content/uploads/2019/11/image-3.png>,.
- [9] Alamy, “Concept of smart grid and renewable energy technology,” 2023. Accessed : 2025-04-11. <https://c8.alamy.com/compfr/h3mmf8/concept-smart-grid-et-smart-grid-industriels-peripheriques-d-un-reseau-connecte-jpg>,.
- [10] F. Zijie, M. A. Al-Shareeda, M. A. Saare, S. Manickam, and S. Karuppayah, “Wireless sensor networks in the internet of things : review, techniques, challenges, and future directions,” *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 31, pp. 1190–1200, August 2023.
- [11] S. El Khediri, A. Benfradj, A. Thaljaoui, T. Moulahi, R. U. Khan, A. Alabdulatif, and P. Lorenz, “Integration of artificial intelligence (ai) with sensor networks : Trends,

- challenges, and future directions,” *Journal of King Saud University-Computer and Information Sciences*, 2023. Available online : <https://www.sciencedirect.com/science/article/pii/S1319157823002007>.
- [12] R. C. Shah, S. Roy, S. Jain, and W. Brunette, “Data mules : Modeling a three-tier architecture for sparse sensor networks,” *Ad Hoc Networks*, vol. 1, no. 2–3, pp. 215–233, 2003.
- [13] S. Messai, *Gestion de la Mobilité dans les Réseaux de Capteurs Sans Fil*. PhD thesis, Université Ferhat Abbas Sétif 1, Algérie & Université Claude Bernard Lyon 1, France, 2019. Soutenue publiquement le 24/11/2019.
- [14] L. B. T, “A scope for manet routing and security threats,” *Journal of Computer Networks*, 2013. Department of Computer Science and Engineering, Manonmaniam Sundaranar University, India.
- [15] D. K. N. Kaur, Dr. S. Verma, “A survey of routing protocols in wireless sensor networks,” *International Journal of Engineering & Technology*, vol. 7, no. 4.12, pp. 20–25, 2018. Website : www.sciencepubco.com/index.php/IJET.
- [16] J. Lu, K. Hu, X. Yang, C. Hu, and T. Wang, “A cluster-tree-based energy-efficient routing protocol for wireless sensor networks with a mobile sink,” *The Journal of Supercomputing*, vol. 77, no. 12, pp. 6078–6104, 2021.
- [17] H. Li, Y. Dai, Q. Chen, D. Liao, and H. Jin, “Energy efficient mobile sink driven data collection in wireless sensor network with nonuniform data,” *Scientific Reports*, vol. 14, p. 28190, 2024.
- [18] D. Xu, “Dynamic sink movement strategy for expedited query processing in internet of things-based sensor networks,” *Journal of Engineering and Applied Science*, vol. 72, no. 43, p. 31, 2025.
- [19] S. Jain, K. K. Pattanaik, R. K. Verma, and A. Shukla, “Edvwdd : Event-driven virtual wheel-based data dissemination for mobile sink-enabled wireless sensor networks,” *The Journal of Supercomputing*, vol. 77, p. 11432–11457, 2021.
- [20] S. K. Jain, R. Bhatia, N. Shrivastava, S. Salunke, M. F. Hashmi, and N. D. Bokde, “Virtual grid-based routing for query-driven wireless sensor networks,” *Future Internet*, vol. 15, no. 8, p. 259, 2023.
- [21] S. K. J. A. Y. Z. S. Sharma, D. Puthal and R. Ranjan, “Rendezvous based routing protocol for wireless sensor networks with mobile sink,” *Wireless Networks*, July 2016. Published online : 7 July 2016.
- [22] X. Han, X. Mu, and J. Zhong, “HGFF : A Deep Reinforcement Learning Framework for Lifetime Maximization in Wireless Sensor Networks,” *IEEE Transactions on Artificial Intelligence*, vol. 6, pp. 859–873, April 2024.
- [23] N. Moussa, Z. Hamidi-Alaoui, and A. E. B. E. Alaoui, “EHRP : An effective hybrid routing protocol to compromise between energy consumption and delay in WSNs,” *arXiv preprint arXiv :2201.03910*, 2022. Preprint available on arXiv.

- [24] M. Srinivas and T. Amgoth, “Ee-hhss : Energy-efficient wireless sensor network with mobile sink strategy using hybrid harris hawk-salp swarm optimization algorithm,” *International Journal of Communication Systems*, vol. 33, no. 4, p. e4569, 2020. Received : 1 April 2020, Revised : 11 July 2020, Accepted : 16 July 2020.
- [25] M. BENAZZOUZ, “Surveillance de tout point d’une zone d’intérêt à l’aide d’un réseau de capteur multimédia sans fil,” 2013.
- [26] B. Yazid, “Optimisation de déploiement dans les réseaux de capteurs sans fil,” 2021.
- [27] S. M. Amal, “Etude du rssi pour l’estimation de la distance dans les reseaux de capteurs sans fil,” 2017.
- [28] B. Abdallah and B. Ilyes, “Clustering et routage dans les réseaux de capteurs corporels sans fil,” 2021. Spécialité : Réseaux de Télécommunications.
- [29] S. MOAD, “La consommation d’énergie dans les réseaux de capteurs sans fil,” 2008. Encadrant : Nizar Bouabdallah.
- [30] M. I. M.A. Matin, “Overview of wireless sensor network, journal of computer applications,” vol. 10, 2013.
- [31] M. Manshahia, “Wireless sensor networks : A survey,” *International Journal of Scientific Engineering Research*, vol. 10, 2016.
- [32] S. SAHRAOUI, “Mécanismes de sécurité pour l’intégration des rcsfs à l’iot (internet of things),” mémoire de master, Université de Batna 2, novembre 2016.
- [33] B. Souad, “L’utilisation des algorithmes génétiques pour l’optimisation du déploiement des réseaux de capteur vidéo sans fils,” Master’s thesis, UNIVERSITE BADJI MOKHTAR - ANNABA, 2021.
- [34] I. T. Almalkawi, M. G. Zapata, J. N. Al-Karaki, and J. Morillo-Pozo, “Wireless multimedia sensor networks : Current trends and future directions,” *Sensors*, vol. 10, no. 7, pp. 6662–6717, 2010.
- [35] “Réseaux de capteurs.” http://www.tafats.fr/Techniques/Reseaux_de_capteurs/Reseaux_capteur_text.html. Consulté le 10 mars 2025.
- [36] O. Boudaa, “Conception et réalisation d’un protocole de routage pour les réseaux de capteurs sans fil,” 2015.
- [37] K. Salaheddine, “État de l’art sur les réseaux de capteurs sans fil,” Juin 2010.
- [38] Université Badji Mokhtar - Annaba, “Chapitre sur les réseaux de capteurs sans fil.” <https://elearning-facsci.univ-annaba.dz/mod/resource/view.php?id=743>, 2023. Cours en ligne, Faculté des sciences.
- [39] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, “A survey on sensor networks,” *IEEE Communications Magazine*, vol. 40, no. 8, 2002.
- [40] S. Kaissari, “Conception d’un réseau de capteurs sans fil,” mémoire de master en génie Électrique, Université Mohammed V – ENS de Rabat, juillet 2015.
- [41] I. Stojmenovic, ed., *Handbook of Sensor Networks : Algorithms and Architectures*. University of Ottawa : Wiley-Interscience, 2005.

- [42] A. Bahga and V. Madisetti, *Internet of Things : A Hands-On Approach*. VPT, first ed., 2014.
- [43] H. Srivastava, “Wireless sensor network technology in iot-components,functions benefits,” 2024. Consulté le 12 mars 2025, disponible sur : <https://richestsoft.com/fr/blog/wireless-sensor-network-technology-in-iot/>.
- [44] A. MARTIN, “Iotfla : Une architecture de domotique sécurisée respectueuse de la vie privée,” mémoire de master, UNIVERSITÉ DU QUÉBEC À MONTRÉAL, DÉCEMBRE 2019.
- [45] W. Zhao, M. J. Lee, and A. Srinivasan, “Message ferrying : Proactive routing in highly-partitioned wireless ad hoc networks,” in *Proceedings of The Ninth IEEE Workshop on Future Trends of Distributed Computing Systems (FTDCS 2003)*, (San Juan, PR, USA), pp. 308–314, IEEE, 2003. Date of Conference : 30 May 2003. Added to IEEE Xplore : 11 June 2003.
- [46] K. Fall, “A delay tolerant networking architecture for challenged internets,” in *Proceedings of the 2003 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM’03)*, (Karlsruhe, Germany), pp. 27–34, ACM, 2003.
- [47] M. D. Francesco, S. K. Das, and G. Anastasi, “Data collection in wireless sensor networks with mobile elements : A survey,” *ACM Transactions on Sensor Networks (TOSN)*, vol. 8, no. 1, pp. 7:1–7:31, 2011.
- [48] J. Li and P. Mohapatra, “Analytical modeling and mitigation techniques for the energy hole problem in sensor networks,” *Pervasive and Mobile Computing*, vol. 3, no. 8, pp. 233–254, 2007.
- [49] R. C. Shah and J. M. Rabaey, “Energy aware routing for low energy ad hoc sensor networks,” in *Proceedings of the 2002 IEEE Wireless Communications and Networking Conference (WCNC)*, (Orlando, FL, USA), pp. 350–355, IEEE, March 2002.
- [50] S. Mammeri, “Le routage dans les réseaux mobiles ad hoc.” Cours de Master 2 en Administration et Sécurité des Réseaux, Université de Béjaïa, Module Réseaux Mobiles, Chapitre 4, 2023-2024. Disponible sur la plateforme eLearning de l’Université de Béjaïa = <https://elearning.univ-bejaia.dz>.
- [51] L. A. Shihab and M. Basher, “Aodv routing protocol performance assessment for wireless sensor network scenarios,” *International Journal Of Engineering And Computer Science*, vol. 10, no. 03, pp. 25292–25301, 2021.
- [52] Z. J. Haas, M. R. Pearlman, and P. Samar, “The zone routing protocol (zrp) for ad hoc networks.” Internet-Draft, IETF MANET Working Group, 2002. Draft expired, originally published August 12, 2002.
- [53] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, “Wireless sensor networks : A survey,” *Computer Networks*, vol. 38, no. 4, pp. 393–422, 2002.
- [54] N. Patwari, J. N. Ash, S. Kyperountas, A. O. Hero, R. L. Moses, and N. S. Correal, “Locating the nodes : Cooperative localization in wireless sensor networks,” *IEEE Signal Processing Magazine*, vol. 22, pp. 54–69, July 2005.

- [55] R. C. Doss, A. Jennings, and N. Shenoy, "A review on current work in mobility prediction for wireless networks," in *Proceedings of the Australian Telecommunications, Networks and Applications Conference (ATNAC)*, (Sydney, Australia), 2004.
- [56] D. Bhadauria, O. Tekdas, and V. Isler, "Robotic data mules for collecting data over sparse sensor fields," *Journal of Field Robotics*, vol. 28, no. 3, pp. 388–404, 2011.
- [57] O. Tsilomitrou and A. Tzes, "Mobile data-mule optimal path planning for wireless sensor networks," *Sensors*, vol. 22, no. 3, p. 1087, 2022.
- [58] A. A. Lata and M. Kang, "A survey on the evolution of opportunistic routing with asynchronous duty-cycled mac in wireless sensor networks," *Sensors*, vol. 20, no. 15, p. 4112, 2020.
- [59] J. G. Alviz, *Opportunistic Data Collection and Routing in Segmented Wireless Sensor Networks*. PhD thesis, Université Paul Sabatier - Toulouse III, 2020. Thèse de Doctorat, Université Toulouse 3 Paul Sabatier, soutenue le 07 décembre 2020.
- [60] S. Zafar, A. Bashir, and S. A. Chaudhry, "Mobility-aware hierarchical clustering in mobile wireless sensor networks," *IEEE Access*, vol. XX, pp. 1–1, 2019. Early Access ; article accepted and not fully edited.
- [61] A. Abuarqoub, M. Hammoudeh, B. Adebisi, S. Jabbar, A. Bounceur, and H. Al-Bashar, "Dynamic clustering and management of mobile wireless sensor networks," *Computer Networks*, vol. 117, pp. 62–75, 2017.
- [62] M. D. F. G. Anastasi, M. Conti and A. Passarella, "Energy conservation in wireless sensor networks : A survey," *Ad Hoc Networks*, vol. 7, no. 3, p. 537–568, 2009.
- [63] G. X. S. Misra, S. D. Hong and J. Tang, "Constrained relay node placement in wireless sensor networks to meet connectivity and survivability requirements," *Proceedings of IEEE INFOCOM 2008*, pp. 281–285, 2008.
- [64] J. Rezazadeh, M. Moradi, and A. S. Ismail, "Mobile wireless sensor networks overview," *IJCCN International Journal of Computer Communications and Networks*, vol. 2, 2012.
- [65] S. MESSAI, "Gestion de la mobilité dans les réseaux de capteurs sans fil," Master's thesis, l'Université Ferhat Abbas Sétif 1 et l'Université Claude Bernard Lyon 1, 2019.
- [66] Y. Wang, "Sink node placement and partial connectivity in wireless sensor networks," *Extended Version of the 2019 IEEE International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA)*, 2023.
- [67] J. Li and P. Mohapatra, "Analytical modeling and mitigation techniques for the energy hole problem in sensor networks," *Pervasive and Mobile Computing*, vol. 3, no. 3, pp. 233–254, 2007.
- [68] A. Mainwaring, J. Polastre, R. Szewczyk, D. Culler, and J. Anderson, "Wireless sensor networks for habitat monitoring," *ACM SIGCOMM Computer Communication Review*, vol. 32, no. 1, pp. 88–97, 2002.

- [69] J.-H. Huang, S. Amjad, and S. Mishra, “Cenwits : a sensor-based loosely coupled search and rescue system using witnesses,” *ACM*, 2005.
- [70] H. Al-Mahdi, M. Elshrkawey, S. Saad, and S. Abdelaziz, “An intelligent energy-efficient data routing scheme for wireless sensor networks utilizing mobile sink,” *Wireless Communications and Mobile Computing*, vol. 2024, pp. Article ID 7384537, 20 pages, 2024.
- [71] P. Juang, H. Oki, Y. Wang, M. Martonosi, L.-S. Peh, and D. Rubenstein, “Energy-efficient computing for wildlife tracking : Design tradeoffs and early experiences with zebranet,” *SIGOPS Operating Systems Review*, vol. 36, no. 5, pp. 96–107, 2002.
- [72] S. M. Mohamed, H. S. Hamza, and I. A. Saroit, “Coverage in mobile wireless sensor networks (m-wsn) : A survey,” *Computer Communications*, 2017. Accepted Manuscript.
- [73] J. N. Al-Karaki and A. E. Kamal, “Routing techniques in wireless sensor networks : A survey,” *IEEE Wireless Communications*, vol. 11, no. 6, pp. 6–28, 2004.
- [74] W. L. A. K. S. J. Wang, Y. Gao and H. J. Kim, “An intelligent data gathering schema with data fusion supported for mobile sink in wireless sensor networks,” *International Journal of Distributed Sensor Networks*, vol. 15, no. 3, pp. 1–9, 2019.
- [75] Y. Y. M. Ma and M. Zhao, “Tour planning for mobile data-gathering mechanisms in wireless sensor networks,” *IEEE Transactions on Vehicular Technology*, vol. 62, pp. 1472–1483, May 2013.
- [76] D. Z. J. Yang and Y. Zhang, “An energy-efficient data gathering protocol for wireless sensor networks,” pp. 1–6, 2009.
- [77] M. K. Marina and S. R. Das, “A survey of mobility models for ad hoc network research,” *Wireless Communications and Mobile Computing*, vol. 2, no. 5, pp. 483–502, 2002.
- [78] A. S. A. Chakrabarti and B. Aazhang, “Using predictable observer mobility for power efficient design of sensor networks,” in *Proceedings of the 2nd International Workshop on Information Processing in Sensor Networks (IPSN)*, (Palo Alto, CA, USA), pp. 129–145, Springer, 2003.
- [79] B. L. H. J. K. J. Wang, Y. Cao and S. Lee., “Particle swarm optimization based clustering algorithm with mobile sink for wsns,” *Journal Name*, vol. Volume Number, no. Issue Number, p. Page Range, 2023. Accessed : Date accessed.
- [80] W. N. G. M. I. Khan and G. Haring, “Static vs. mobile sink : The influence of basic parameters on energy efficiency in wireless sensor networks,” *Journal Name*, vol. Volume Number, no. Issue Number, p. Page Range, 2023. Accessed : Date accessed.
- [81] D. D. J. D. E. A. A. Somasundara, A. Kansal and M. B. Srivastava, “Controllably mobile infrastructure for low energy embedded networks,” *IEEE Transactions on Mobile Computing*, vol. 5, pp. 958–973, Aug. 2006.

- [82] M. M. J. M. C. L. K. Ketshabetswe, A. M. Zungeru and B. Sigweni, “Communication protocols for wireless sensor networks : A survey and comparison,” *Journal of Network and Computer Applications*, vol. 175, p. 102927, 2021.
- [83] M. H. H. V. H. M. L. Z. Wang, T. Schaul and N. D. Freitas, “Dueling network architectures for deep reinforcement learning,” *arXiv preprint arXiv :1511.06581*, 2015.
- [84] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, “Energy-efficient communication protocol for wireless microsensor networks,” *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences*, 2000. Volume 1.
- [85] S. A. H. Sedjelmaci, “Mise en œuvre de mécanismes de sécurité basée sur les ids pour les réseaux de capteurs sans fil,” 2012.
- [86] J. Yick, B. Mukherjee, and D. Ghosal, “Wireless sensor network survey,” *Computer Networks*, vol. 52, no. 12, pp. 2292–2330, 2008.
- [87] K. Dantu, M. Rahimi, H. Shah, S. Babel, A. Dhariwal, and G. S. Sukhatme, “Robomote : Enabling mobility in sensor networks,” pp. 404–409, 2005.
- [88] M. Matin and M. Islam, “Overview of wireless sensor network,” *Journal of Computer Applications*, vol. 10, pp. 25–29, 2013.
- [89] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, “An application-specific protocol architecture for wireless microsensor networks,” *IEEE Transactions on Wireless Communications*, vol. 1, no. 4, pp. 660–670, 2002.
- [90] F. Li, Y. Yang, and J. Wu, “Mobility management in manets : Exploit the positive impacts of mobility,” pp. 211–235, 2009.
- [91] S. Najjar-Ghabel, L. Farzinvash, and S. N. Razavi, “Data harvesting in wireless sensor networks using mobile sinks under real-world circumstances,” *The Journal of Supercomputing*, vol. 79, p. 5486–5515, 2023.

Résumé

Maximiser la durée de vie d'un réseau de capteurs sans fil (RCSF) tout en optimisant l'efficacité de la collecte de données constitue un défi majeur, en particulier dans des environnements dynamiques et contraints en énergie. Si les stratégies classiques du puits mobile offrent une mise en œuvre simple, elles entraînent souvent des performances limitées. Les approches basées sur l'apprentissage par renforcement ouvrent de nouvelles perspectives, en permettant une prise de décision autonome et adaptative. Ce travail s'inscrit dans ce contexte en proposant deux approches de mobilité du sink qui visent à améliorer les performances des RCSFs en se basant sur l'apprentissage par renforcement. Ces approches sont nommées : QLB-MOB (Q-Learning Based-MOBility), qui utilise l'algorithme Q-learning pour assurer un déplacement intelligent du sink ; et DQL-MOB (Deep Q-Learning Based-MOBility), fondée sur un Dueling Deep Q-Network (Dueling DQN). Ces approches adaptent dynamiquement le déplacement du sink selon l'énergie, la distance et la charge des nœuds. Les stratégies d'apprentissage sont modélisées comme des processus de décision de Markov (MDP), avec une politique ϵ -greedy et sont combinées à un routage hybride basé sur une version adaptée de l'algorithme de Dijkstra. DQL-MOB intègre en outre un mécanisme léger d'agrégation de paquets pour réduire la surcharge des nœuds. Une comparaison entre quatre stratégies de mobilité du sink : mobilité aléatoire, mouvement circulaire, QLB-MOB, et DQL-MOB a été effectuée afin d'évaluer l'efficacité de nos solutions. Les simulations montrent que nos approches améliorent significativement le taux de livraison des paquets et la longévité du réseau en offrant une meilleure équité énergétique. Les résultats obtenus confirment la pertinence de l'apprentissage profond pour la mobilité intelligente dans les RCSF.

Mots-clés : Réseaux de capteurs sans fil mobiles, apprentissage par renforcement, Q-Learning, Deep Q-Learning, MDP, routage hybride, agrégation de données, équilibrage de charge, efficacité énergétique.

Abstract

Maximizing the lifetime of a Wireless Sensor Network (WSN) while optimizing the efficiency of data collection remains a major challenge, particularly in dynamic and energy-constrained environments. Although traditional mobile sink strategies offer simple implementation, they often result in limited performance. Reinforcement learning-based approaches open up new possibilities by enabling autonomous and adaptive decision-making. This work fits within this context by proposing two sink mobility approaches aimed at improving the performance of WSNs through reinforcement learning. These approaches are named : QLB-MOB (Q-Learning Based-Mobility), which uses the Q-learning algorithm to ensure intelligent sink movement ; and DQL-MOB (Deep Q-Learning Based-Mobility), based on a Dueling Deep Q-Network (Dueling DQN). These approaches dynamically adapt the sink's movement based on the energy, distance, and load of the nodes. The learning strategies are modeled as Markov Decision Processes (MDP), with an ϵ -greedy policy, and are combined with a hybrid routing mechanism based on an adapted version of Dijkstra's algorithm. In addition, DQL-MOB integrates a lightweight packet aggregation mechanism to reduce node overhead. A comparison among four sink mobility strategies—random mobility, circular movement, QLB-MOB, and DQL-MOB—was conducted to evaluate the effectiveness of our solutions. Simulations show that our approaches significantly improve packet delivery rates and network longevity by providing better energy fairness. The results confirm the relevance of deep learning for intelligent mobility in WSNs.

Keywords : Mobile wireless sensor networks, reinforcement learning, Q-Learning, Deep Q-Learning, MDP, hybrid routing, data aggregation, load balancing, energy efficiency.