REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université de Batna 2

Faculté des Mathématiques et de l'Informatique

Département d'Informatique

THESE

En vue de l'obtention du diplôme de

Doctorat en sciences en Informatique

Présentée par

Sabrina BOUBICHE

Support du Big Data dans le processus d'agrégation des données dans les RCSF hétérogènes

Soutenue publiquement le : 13/02/2022

Jury :

Pr. Rachid SEGHIR	*Professeur*	*Université de Batna 2*	*Président*
Pr. Azzedine BILAMI	*Professeur*	*Université de Batna 2*	*Rapporteur*
Pr. Allaoua CHAOUI	*Professeur*	*Université de Constantine 2*	*Examinateur*
Pr. Mohamed BENMOHAMMED	*Professeur*	*Université de Constantine 2*	*Examinateur*
Pr. Djalal HEDJAZI	*Professeur*	*Université de Batna 2*	*Examinateur*
Pr. Larbi GUEZOULI	*Professeur*	*Ecole Nationale Supérieure EREDD*	*Examinateur*

Remerciements

Mes remerciements s'adressent en premier lieu à mes chers parents, sans qui je ne serais jamais arrivée à réaliser mes objectifs.

Je remercie chaleureusement mon directeur de thèse Pr. Bilami Azeddine pour avoir accepté de diriger mon travail afin de le mener à bon port. Les conseils qu'il m'a prodigués et la confiance qu'il m'a témoignée ont été déterminants dans la réalisation de ce travail de recherche.

Mes vifs remerciements vont également aux membres du jury : Pr. Chaoui Allaoua et Pr. Benmohammed Mohamed de l'Université de Constantine 2, Pr. Guezouli Larbi de l'Ecole Nationale Supérieure des Energies Renouvelables, Environnement et Développement Durable-Batna, Pr. Hedjazi Djalal et Pr. Seghir Rachid de l'Université de Batna 2, pour l'intérêt qu'ils ont porté à ce travail en acceptant de l'examiner et de l'enrichir par leurs propositions.

Je tiens à offrir un remerciement spécial à tous les membres de ma famille pour leur soutien et leur accompagnement durant tout mon parcours.

A la fin, mes remerciements s'adressent à toute personne ayant contribué de près ou de loin à la réalisation de ce travail.

Résumé

Récemment, et due à la croissance impressionnante des quantités de données transmises sur les réseaux de capteurs sans fil hétérogènes, la technologie Big Data est devenue une tendance largement reconnue dans le domaine des réseaux de capteurs sans fil, et fait de plus en plus l'objet de recherches. Le terme Big Data ne concerne pas seulement le volume de données, mais également la vitesse de transmission élevée et la grande variété d'informations difficiles à collecter, stocker et à traiter en utilisant les technologies classiques disponibles. Bien que les données générées par les capteurs individuels puissent ne pas sembler significatives, toutes les données générées par les nombreux capteurs dans les réseaux de capteurs sont capables de produire des volumes importants de données. La gestion Big Data impose des contraintes supplémentaires aux réseaux de capteurs sans fil et en particulier au processus d'agrégation de données, qui représente l'un des paradigmes essentiels des réseaux de capteurs sans fil. Le processus d'agrégation de données peut représenter une solution au problème du Big Data en permettant de combiner des données provenant de sources différentes afin d'éliminer celles qui sont redondantes, et par conséquent réduire les quantités de données et la consommation des ressources disponibles dans le réseau. L'objectif principal de ce travail est de proposer une nouvelle approche pour le support du Big Data dans le processus d'agrégation de données dans les réseaux de capteurs sans fil hétérogènes. L'approche proposée vise à réduire le coût de l'agrégation de données en termes de consommation d'énergie, en équilibrant les charges de données sur les nœuds hétérogènes. L'approche proposée est optimisée en intégrant le mécanisme du feedback control afin de résoudre le problème de la planification d'agrégation des données, permettant ainsi de maintenir une précision élevée et une latence minimale.

Mots clés : Big Data, réseaux de capteurs sans fil hétérogènes, agrégation des données, Feedback control.

Abstract

Recently, and due to the impressive growth in the amounts of data transmitted over the heterogeneous sensor networks, big data has emerged as a widely recognized trend and is increasingly being talked about. The term big data does not only imply the volume of data but also the high speed of transmission and the wide variety of information that is difficult to collect, store, and process using the available classical technologies. Although the data generated by the individual sensors may not appear to be significant, all the data generated in the many sensors in the sensor networks are able to produce large volumes of data. Big data management imposes additional constraints on the wireless sensor networks and especially on the data aggregation process, which represents one of the important paradigms in the wireless sensor networks. Data aggregation process can represent a solution to the problem of big data by allowing data from different sources to be combined to eliminate the redundant ones, and consequently reduce the amounts of data and the consumption of the available resources in the network. The main objective of this work is to propose a new approach for handling big data in the aggregation process in heterogeneous wireless sensor networks. The proposed approach aims to reduce the cost of the data aggregation in terms of energy consumption by balancing the data loads on the heterogeneous nodes. The proposed approach is then improved by integrating the feedback control closed loop to solve the data aggregation planning problem, maintaining therefore a high accuracy and a minimal delay.

Index terms Big data, heterogeneous wireless sensor networks, data aggregation, Feedback control.

ملخص

في الآونة الأخيرة، ونظراً للنمو والتطور الكبيرين في كميات البيانات المنقولة عبر شبكات الاستشعار غير المتجانسة، ظهرت تكنولوجيا البيانات الضخمة كاتجاه معترف به على نطاق واسع وتزايد الحديث عنه في مجالات البحث العلمي. لا يقتصر مصطلح البيانات الضخمة على حجم البيانات فحسب، بل يتعلق أيضًا بالسرعة العالية للإرسال والتنوع الواسع للمعلومات التي يصعب جمعها وتخزينها ومعالجتها باستخدام التقنيات الكلاسيكية المتاحة. على الرغم من أن البيانات التي يتم إنشاؤها بواسطة أجهزة الاستشعار الفردية قد لا تبدو مهمة، الا أن جميع البيانات التي يتم إنشاؤها من خلال العديد من أجهزة الاستشعار في شبكات الاستشعار قادرة على إنتاج كميات كبيرة من البيانات. تفرض إدارة البيانات الضخمة قيودًا إضافية على شبكات الاستشعار اللاسلكية وخاصةً على عملية تجميع البيانات، والتي تمثل أحد النماذج الأساسية في شبكات الاستشعار اللاسلكية. يمكن أن تمثل عملية تجميع البيانات حلاً لمشكلة البيانات الضخمة من خلال السماح بجمع البيانات من مصادر مختلفة للتخلص من المصادر الزائدة، وبالتالي تقليل كميات البيانات واستهلاك الموارد المتاحة في الشبكة. الهدف الرئيسي من هذا العمل هو اقتراح تقنية جديدة لدعم البيانات الضخمة في عملية تجميع البيانات في شبكات الاستشعار اللاسلكية غير المتجانسة. يهدف العمل المقترح إلى تقليل تكلفة تجميع البيانات من حيث استهلاك الطاقة من خلال موازنة أحمال البيانات على أجهزة الاستشعار غير المتجانسة. تم أيضا تحسين التقنية المقترحة من خلال استعمال آلية التحكم بأثر ارجاعي لتعزيز توازن تجميع البيانات على أجهزة الاستشعار غير المتجانسة، وبالتالي الحفاظ على وقت استجابة ووقت تجميع أمثل.

Table des matières

Liste des figures

Liste des tableaux

Liste des algorithmes

Liste des organigrammes

Introduction générale

Les progrès réalisés dans les différents domaines technologiques, notamment les domaines des communications sans fil et l'électronique numérique, ont accru l'attention mondiale portée aux réseaux de capteurs sans fil ou RCSF au cours des dernières années [1] [2] [3][4], facilitant ainsi le développement de capteurs intelligents, multifonctionnels, économiques et puissants, qui sont miniatures et capables de communiquer librement sur de courtes distances. Les capteurs ont la possibilité de détecter, mesurer et collecter les informations de l'environnement, stocker et traiter les données et communiquer entre eux afin de les transmettre. Cela permet de distribuer leurs puissances de calcul et de stockage dans un système dit collaboratif à partir duquel ils peuvent envoyer les informations vers une station collectrice appelée puits ou station de base pour leur traitement ultérieur. Les capteurs ont l'avantage d'être flexibles, pouvant être utilisés à différents endroits pour surveiller les activités et ils consomment des quantités acceptables d'énergie.

Un réseau de capteurs sans fil est formé de collections de nœuds capteurs largement déployés dans des zones généralement inaccessibles et formant des réseaux de propagation des données [5]. Leur rôle principal est d'observer un processus, de collecter des données de l'environnement et de les transmettre vers la station de base qui est chargée de leur traitement. Les capteurs ont par conséquent introduit l'idée de réseaux de capteurs basés sur l'effort de collaboration entre de grands ensembles de capteurs.

Les réseaux de capteurs sans fil représentent une classe informatique importante qui intègre l'informatique dans le monde physique. À ce jour, la plupart des travaux de recherche se sont focalisés sur des réseaux de capteurs sans fil homogènes ou classiques [6], où tous les nœuds du réseau sont identiques. Cependant, les progrès continus, en particulier dans le processus de miniaturisation et des communications à faible puissance, ont permis le développement d'une grande variété de nœuds. Lorsque plus d'un type de nœud est intégré dans un réseau de capteurs sans fil, il est appelé hétérogène [7]. Placer des nœuds hétérogènes dans un réseau de capteurs sans fil est un moyen efficace permettant d'augmenter la durée de vie du réseau. En outre, l'intégration de l'hétérogénéité dans le réseau peut améliorer son évolutivité, réduire les besoins en énergie sans sacrifier les performances, équilibrer le coût et les fonctionnalités du réseau, encourager de nouvelles applications à large bande et permettre d'améliorer les mécanismes déployés en utilisant des systèmes plus complexes et plus énergivores, qui peuvent être tolérés par certains types de nœuds.

Un réseau de capteurs sans fil hétérogène typique se compose d'un grand nombre de nœuds homogènes et de quelques nœuds hétérogènes. Les nœuds identiques, dont les fonctions principales consistent à collecter et envoyer les données sont peu coûteux, tandis que les nœuds hétérogènes auront des tâches plus complexes et plus consommatrices, ce qui permet de donner un équilibre à l'ensemble du réseau.

Dans les réseaux de capteurs sans fil et en particulier les réseaux de capteurs sans fil hétérogènes, les données générées par les capteurs peuvent croître de façon exponentielle au fil du temps. Par conséquent, des centaines de milliers de données sont collectées et doivent être traitées efficacement. Les technologies de l'information traditionnelles dédiées pour le traitement des données peuvent prendre en charge des quantités limitées de données générées dans les réseaux de capteurs sans fil. Cependant, ces technologies deviennent rapidement limitées et coûteuses pour le traitement de très grandes quantités de données. Pour cela, une nouvelle technologie visant à traiter et stocker les larges volumes de données appelée la technologie Big Data [8] [9] a été récemment mise au point. La technologie Big Data peut représenter un aspect innovant dans les réseaux de capteurs sans fil en mettant au point des outils adaptés pour l'organisation de la collecte, l'analyse, le traitement et le stockage des données volumineuses.

La technologie du Big Data est utilisée pour caractériser des ensembles de données volumineux qui peuvent être complexes et par conséquent difficiles à traiter en utilisant les méthodes classiques de traitement des données [10]. Cette technologie a été initialement adaptée aux réseaux filaires. Cependant, et en raison des grandes quantités de données auxquelles les réseaux de capteurs sans fil sont de plus en plus confrontés, ce paradigme est de plus en plus demandé dans les RCSF [11] [12] [13] [14], augmentant par conséquent le besoin de technologies et d'architectures adaptées pour traiter les données.

L'agrégation des données [15] est l'un des principaux défis des réseaux de capteurs sans fil liés à la technologie Big Data. L'agrégation des données permet de combiner des données provenant de différentes sources afin d'éliminer la redondance et réduire par conséquent la consommation des ressources disponibles sur le réseau. Ceci implique l'utilisation de techniques qui combinent et rassemblent des données provenant de sources multiples pour réaliser des inférences plus efficaces et potentiellement plus précises.

L'objectif de ce travail consiste en premier lieu à introduire le paradigme Big Data dans les réseaux de capteurs sans fils, en particulier les RCSF hétérogènes, d'aborder ses principaux concepts et outils analytiques et d'examiner les différentes stratégies proposées pour son intégration dans les réseaux de capteurs sans fil.

Nous considérons la technique d'agrégation des données comme une des solutions les plus prometteuses dans le processus d'intégration de la technologie Big Data dans le domaine des RCSF. Pour cela, nous proposons une nouvelle approche pour le support de la technologie Big Data dans le processus d'agrégation des données dans les réseaux de capteurs sans fil hétérogènes. L'approche proposée vise à réduire le coût de l'agrégation des données en termes de consommation énergétique, en équilibrant les charges des données sur les nœuds hétérogènes. L'approche est ensuite améliorée en intégrant le mécanisme du Feedback control qui permet d'ajuster les paramètres d'agrégation des données sur les nœuds, permettant ainsi de maintenir une latence réduite et une précision élevée.

Notre travail est structuré en deux parties, la première partie permet de présenter le domaine de recherche sur lequel notre travail est basé. La deuxième partie est dédiée à notre contribution. La première partie est structurée en trois chapitres. Le premier chapitre présente les réseaux de capteurs sans fil et particulièrement les réseaux de capteurs sans fil hétérogènes. Le deuxième chapitre introduit la technologie Big Data, et dans lequel nous présentons notre proposition d'une nouvelle classification des challenges Big Data dans les réseaux de capteurs sans fil. Dans le troisième chapitre, nous définissons le mécanisme d'agrégation des données dans les réseaux de capteurs sans fil, et nous présentons un état de l'art survolant les différents protocoles d'agrégation des données dans les réseaux de capteurs sans fil dédiés à la technologie Big Data. La deuxième partie du travail est structurée en deux chapitres. Le premier chapitre présente le mécanisme d'agrégation des données proposé. Le deuxième chapitre est destiné à l'évaluation des performances du protocole proposé.

Partie I

Généralités

Chapitre 1

Les réseaux de capteurs sans fil

Chapitre1 : Les réseaux de capteurs sans fil

1. Introduction

Le réseau de capteurs sans fil est un système distribué qui représente un nouveau domaine important dans la technologie sans fil [16] [17]. Un réseau de capteurs sans fil (RCSF) est défini comme étant un réseau sans infrastructure. Il se compose de plusieurs nœuds équipés de capteurs interagissant avec l'environnement physique et dédiés à des tâches spécifiques. L'objectif principal des nœuds d'un réseau de capteurs sans fil consiste à surveiller les conditions environnementales, telles que la température, le son, ou la pollution, et de collaborer afin de transmettre les données recueillies vers un emplacement principal appelé le puits ou la station de base (BS) qui agit comme une interface entre les utilisateurs et le réseau.

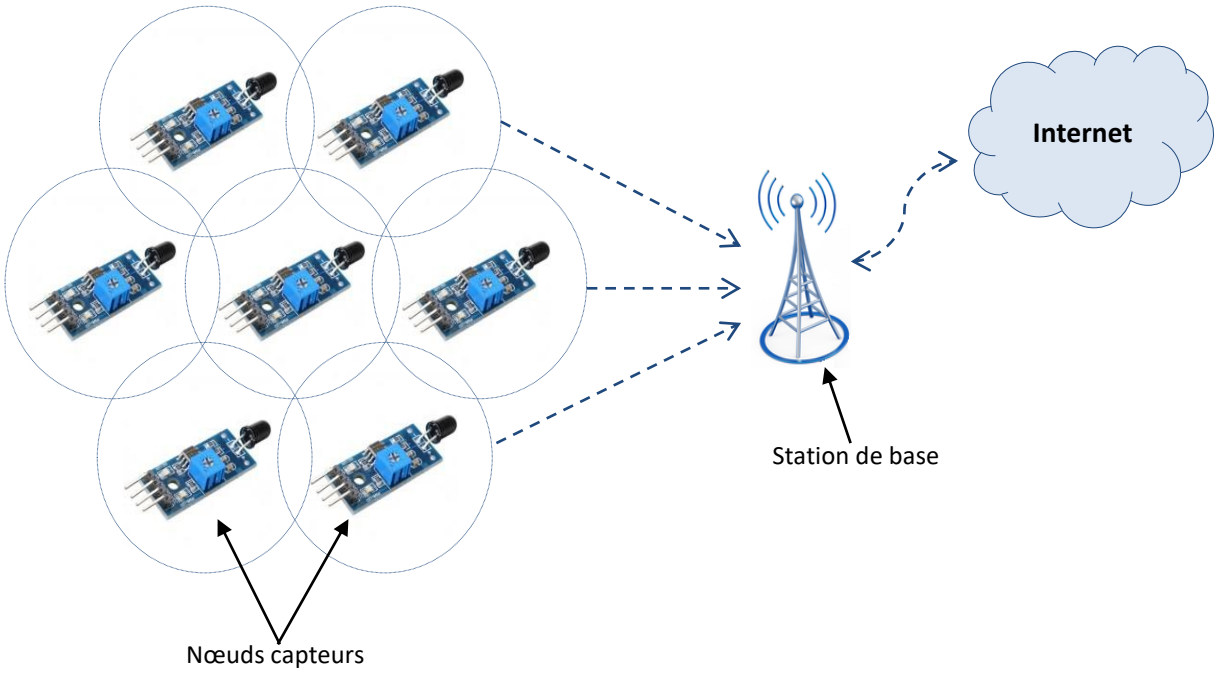


Figure 1-1. Réseau de capteurs sans fil

2. Les capteurs sans fil

Un réseau de capteurs sans fil contient généralement plusieurs nœuds capteurs [18] [19]. Les capteurs peuvent communiquer entre eux en utilisant généralement des signaux radio. Un nœud capteur sans fil est équipé généralement de dispositifs lui permettant d'effectuer la détection et le calcul. Il est aussi équipé d'émetteurs-récepteurs radio. Les capteurs possèdent

une vitesse de traitement, une capacité de stockage et une bande passante de communication limitées. Leur rôle principal consiste à collecter les informations de l'environnement et à collaborer afin de transmettre l'information vers la station de base. Le mode de fonctionnement des nœuds capteurs peut être soit continu, soit piloté par événement.

Figure 1-2. Les capteurs sans fil [20]

2.1 Structure d'un nœud capteur

Les nœuds capteurs sont généralement structurés en quatre sous-systèmes [21] [22] : sous-système de détection, sous-système de traitement, sous-système de communication et sous-système d'alimentation. Le sous-système de détection permet de recueillir les données de l'environnement et de convertir les signaux analogiques en signaux numériques. Le sous-système de traitement représente la partie cérébrale du nœud. Ce sous-système est responsable du traitement et du stockage des données recueillies. Le sous-système de communication est chargé de fournir un canal de communication d'un nœud vers un autre à l'aide d'un émetteur-récepteur. Enfin, le sous-système d'alimentation est chargé de fournir l'énergie aux nœuds à l'aide d'une batterie. Les sous-systèmes sont présentés en détail dans ce qui suit :

2.1.1 Sous-système de détection

Le sous-système de détection comprend les capteurs et les convertisseurs analogique-numérique (ADC). Il agit comme une interface entre l'environnement physique et le monde

virtuel. Autrement dit, il permet la collecte des données de l'environnement et leur conversion de signaux analogiques en signaux numériques pour leur éventuel traitement.

1. *Capteur :* Un capteur [23] est un appareil responsable de la détection des phénomènes physiques tels que la pression, le mouvement, la vitesse, etc. et leur transformation en signal analogique à l'aide d'un transducteur. Un réseau de capteurs sans fil intègre un grand nombre de nœuds, chacun contenant un ou plusieurs capteurs selon le domaine d'application. Il existe une variété de types de capteurs qui peuvent être utilisés dans les réseaux de capteurs sans fil. Un exemple de classification des capteurs est celui des capteurs actifs et passifs. Les capteurs actifs fournissent ou envoient leur propre énergie électromagnétique, puis enregistrent ce qui leur revient. Autrement dit, ils doivent émettre une sorte d'énergie (par exemple, micro-ondes, lumière, son) pour déclencher une réponse ou pour détecter un changement dans l'énergie du signal transmis. Le radar est un exemple d'un tel capteur. Alternativement, les capteurs passifs détectent l'énergie naturellement rayonnée ou réfléchie de l'environnement et puisent leur puissance dans cet apport d'énergie. Les thermomètres sont une bonne illustration d'un capteur passif.

2. *Convertisseur analogique-numérique (ADC) :* La sortie d'un capteur est un signal analogique. Cela signifie qu'il doit y avoir une interface entre le capteur et le processeur numérique (microcontrôleur). Le convertisseur analogique-numérique (ADC) [24] convertit la sortie d'un capteur en un signal numérique.

2.1.2 Sous-système de traitement

Le sous-système de traitement comprend généralement une RAM, un contrôleur, un système d'exploitation et un temporisateur, chargés respectivement du stockage, du traitement et de l'exécution des événements. Le sous-système de traitement est l'élément central du nœud et le choix d'un processeur détermine le compromis entre flexibilité et efficacité en termes d'énergie et de performances. En d'autres termes, c'est l'unité qui détermine la consommation énergétique ainsi que les capacités de calcul d'un capteur.

Il existe une variété de processeurs, chacun ayant ses propres avantages et inconvénients : microcontrôleurs, processeur de signal numérique, circuits intégrés spécifiques à l'application et matrices de portes programmables (FPGA). Le processeur le plus couramment utilisé dans les capteurs est le microcontrôleur.

1. *Microcontrôleur :* Le microcontrôleur [25] est un processeur dont le processus de conception et de mise en œuvre n'est pas aussi coûteux et complexe que les autres types de processeurs. Le microcontrôleur permet de prendre en charge les installations de code dynamique et les mises à jour de logiciels s'exécutant sur des capteurs sans fil qui nécessitent parfois des modifications à distance. De telles tâches

nécessitent une quantité considérable d'espace de calcul et de traitement au moment de l'exécution.

2. Horloge : Les microcontrôleurs sont dotés d'horloges [26] permettant aux programmes d'être exécutés à leur rythme. Ainsi, les instructions sont exécutées en synchronisation avec les tics de l'horloge.

3. *Système d'exploitation (OS*) : Le système d'exploitation permet aux applications d'interagir avec les ressources matérielles, de planifier, et de hiérarchiser les tâches. Dans les réseaux de capteurs sans fil, un système d'exploitation multitâche représente un choix idéal. Par exemple, dans un nœud capteur, le sous-système de traitement peut interagir avec le sous-système de communication tout en agrégeant les données qui arrivent du sous-système de détection. Cependant, cette fonction multitâche nécessite beaucoup de mémoire pour gérer le traitement simultané des tâches que la plupart des capteurs existants ne peuvent pas gérer en raison de leurs ressources limitées. TinyOS [27] est le système d'exploitation le plus connu dans les réseaux de capteurs sans fil.

4. *RAM :* Il s'agit d'une mémoire interne volatile utilisée pour le stockage des données. La mémoire flash (ROM) peut également être utilisée pour stocker des codes de programme de base. Le choix de la taille de mémoire appropriée est crucial car il peut affecter le coût global du nœud ainsi que la consommation d'énergie.

2.1.3 Sous-système de communication

Le rôle du sous-système de communication consiste à gérer la transmission des données entre les capteurs et les autres sous-systèmes, facilitant ainsi la communication et les interactions entre les composants des capteurs et le processeur.

Émetteur-récepteur : C'est un appareil qui peut fonctionner comme émetteur ou comme récepteur. L'émetteur-récepteur reçoit les instructions du processeur et permet de les transmettre aux autres sous-systèmes ainsi qu'aux autres capteurs. Le sous-système de communication est le sous-système dont la consommation électrique est la plus élevée. En conséquent, la plupart des émetteurs-récepteurs offrent la possibilité de réguler la consommation en se basant sur différents états de fonctionnement (actif, inactif et veille).

Les réseaux de capteurs sans fil peuvent utiliser une variété de supports de transmission sans fil pour la communication, tels que :

1. *Fréquence Radio (RF) :* Implique la transmission des données en utilisant des fréquences radio spécifiques. RF est un terme qui fait référence au courant alternatif (AC) ayant des caractéristiques telles que si le courant est transmis à une antenne, un champ électromagnétique adapté aux communications sans fil est généré. Il s'agit du

support de transmission le plus couramment utilisé par les applications des réseaux de capteurs sans fil.

2. *Infrarouge :* Pour que la communication infrarouge puisse avoir lieu, les nœuds capteurs doivent être alignés dans un plan. Ce type de communication ne nécessite pas d'antenne mais sa capacité de diffusion est limitée, avec une courte portée d'environ 1 mètre de distance. Les télécommandes de télévision constituent une analogie pratique pour décrire le fonctionnement des capteurs utilisant la technologie infrarouge pour la communication.

3. *Laser (communication optique) :* La communication laser est un système de télécommunication dans lequel l'émetteur convertit le signal en forme optique du côté émetteur puis convertit le signal optique en signal d'origine du côté récepteur. Cette communication ne nécessite pas beaucoup d'énergie, mais nécessite une visibilité directe entre émetteur et récepteur. Elle est aussi sensible aux conditions atmosphériques.

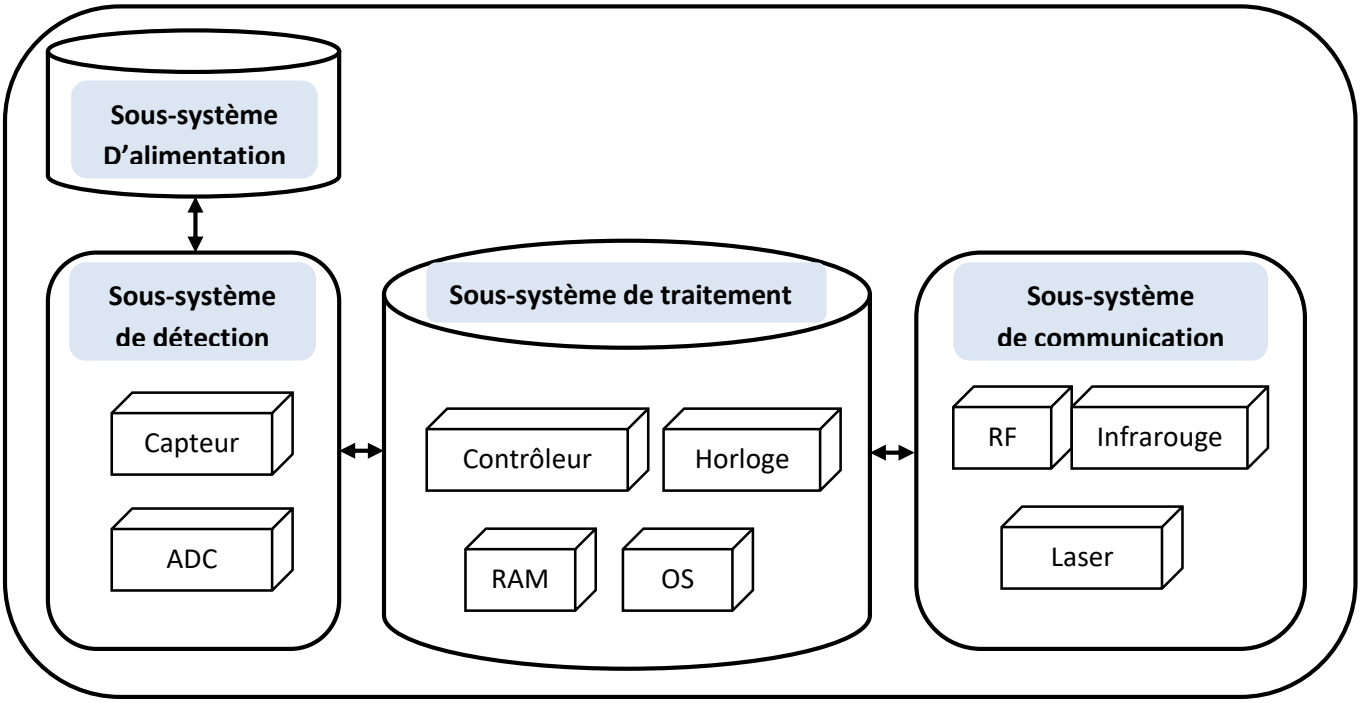


Figure 1-3. Structure d'un nœud capteur

2.1.4 Sous-système d'alimentation

Le sous-système d'alimentation fournit de l'énergie aux nœuds capteurs. La batterie ou la pile est généralement l'élément central utilisé pour alimenter les capteurs. Les batteries peuvent être remplacées ou rechargées. Pour les piles non rechargeables, elles sont éliminées une fois que leur énergie est épuisée. Pour cette raison, et afin de gérer le coût de leur remplacement, les piles sont construites pour avoir une densité énergétique élevée, ce qui signifie qu'elles

peuvent stocker plus d'énergie pour durer plus longtemps. Les piles peuvent être rechargées en utilisant par exemple l'énergie solaire.

3. Domaines d'application des réseaux de capteurs sans fil

La conception des nœuds capteurs, leur permettant d'assurer la micro-détection ainsi que leurs connexions sans fil, leur donne la possibilité d'être déployés dans de nombreux domaines d'application [5] [28] [29]. Les domaines d'utilisation des capteurs sont catégorisés en domaine militaire, environnemental, santé, habitation et autres zones commerciales. Cette classification peut aussi être étendue pour couvrir d'autres domaines d'application comme l'exploration spatiale, chimique, traitements et interventions en cas de catastrophes.

3.1 Domaine militaire

Les caractéristiques des réseaux de capteurs sans fil, comme le déploiement rapide des capteurs, leur auto-organisation et tolérance aux pannes, ont permis que leur application dans le domaine militaire soit largement exploitée. Ainsi, les réseaux de capteurs sans fil sont de plus en plus déployés pour la surveillance des champs de bataille, la reconnaissance des forces adverses, l'exploitation du terrain, le ciblage, l'évaluation des dégâts après les combats, ainsi que la détection et l'identification des différentes attaques ennemies.

3.2 Domaine environnemental

La surveillance de l'environnement représente un élément important dans l'application des réseaux de capteurs sans fil. La surveillance environnementale contrôle et surveille les différents paramètres environnementaux :

- *Surveillance des forêts :* Grâce au déploiement stratégique des capteurs dans les forêts, ils peuvent surveiller et rapporter l'origine exacte des feux. Aussi, comme les capteurs peuvent communiquer et collaborer, ils sont capables de surmonter les différents obstacles possibles tels que les arbres et les rochers.

- *Surveillance et détection des inondations :* Le déploiement des capteurs météorologiques dans l'environnement permet de détecter et de fournir des informations sur les précipitations ainsi que le niveau d'eau, permettant par conséquent d'intervenir afin de prévenir les possibles inondations.

- *Surveillance agriculturale :* Les capteurs sont déployés dans l'environnement afin de pouvoir surveiller et contrôler le niveau d'utilisation des pesticides dans les champs, ainsi que le niveau de pollution dans l'air.

3.3 Domaine de la santé

Le domaine de la santé est largement exploité par les réseaux de capteurs sans fil. L'application des RCSF dans ce domaine implique la fourniture d'interfaces adaptées pour les personnes handicapées, le diagnostic des différentes maladies et anomalies, la surveillance des constantes vitales des patients, ainsi que le suivi des médecins, des patients, et du personnel au sein des structures hospitalières. Pour réaliser le suivi des patients, chaque patient peut être équipé de capteurs miniatures dont chacun peut réaliser une tâche spécifique, comme la mesure de la fréquence cardiaque ainsi que la pression artérielle. Les médecins et le personnel peuvent également être équipés de capteurs permettant de les localiser au sein des établissements hospitaliers.

3.4 Domaine domestique

L'application des capteurs dans le domaine domestique inclus leur intégration aux différents appareils domestiques, tels que les micro-ondes, les réfrigérateurs et les téléviseurs. Ces capteurs intégrés permettent aux utilisateurs de mieux contrôler les appareils domestiques à distance, créant par conséquent un environnement intelligent centré sur l'homme et la technologie et s'adaptant aux différents besoins des utilisateurs.

4. Contraintes de conception d'un réseau de capteurs sans fil

L'objectif principal des réseaux de capteurs sans fil consiste à implémenter des dispositifs de petite taille, pas chers et efficaces. Les nœuds capteurs typiques possèdent les vitesses de traitement et les capacités de stockage des systèmes informatiques d'il y a plusieurs décennies. Ces contraintes ont un impact sur la conception globale d'un réseau de capteurs sans fil. Travailler avec ces ressources limitées tout en garantissant l'efficacité représente un défi majeur pour les concepteurs des RCSF. Certains des défis de conception les plus courants comprennent [30] :

4.1 Efficacité énergétique

Les capteurs sont des dispositifs microélectroniques alimentés généralement par des batteries, et possédant des capacités énergétiques limitées. Idéalement, la durée de vie de la batterie doit correspondre à la durée de la tâche pour laquelle le capteur est dédié. Cependant, certaines tâches consomment plus d'énergie, en particulier le sous-système de communication. Dans ce qui suit, certains défis auxquels les concepteurs sont confrontés afin de réguler la consommation d'énergie sont énumérés :

1. *États de commutation :* Afin de contrôler la consommation d'énergie pendant la communication, les émetteurs-récepteurs sont conçus pour être dans l'un des états suivants : actifs, inactifs ou en veille. L'état actif correspond à l'état dans lequel les nœuds reçoivent et transmettent. L'état inactif correspond à l'état dans lequel le

capteur est allumé mais ne transmet ou ne reçoit aucune donnée. Enfin, l'état de veille correspond à l'état dans lequel le capteur est éteint. Les concepteurs doivent décider comment et quand il est approprié d'implémenter chaque état afin de conserver l'énergie et maintenir l'efficacité du réseau.

2. *Batteries rechargeables et non rechargeables :* Le fait que la batterie puisse être rechargée ou non affecte de manière significative la stratégie appliquée pour la consommation d'énergie. Si les capteurs fonctionnent dans des conditions environnementales difficiles, ce qui rend difficile ou impossible le changement de la batterie ou le remplacement du capteur, il serait conseillé d'utiliser des batteries rechargeables telles que des panneaux solaires. Cependant, les piles rechargeables sont plus chères que les piles jetables ; ce qui signifie qu'il faut mettre un compromis entre le coût et la consommation d'énergie, et par conséquent la fiabilité globale du réseau.

4.2 Contraintes temps réel

Le réseau de capteurs sans fil interagit avec l'environnement réel, et les données des capteurs doivent être fournies le plus souvent dans des délais spécifiques pour que les informations restent pertinentes. Un exemple d'application des capteurs basé sur le temps est le système de détection d'incendie. Cependant, la réalisation en temps réel dans un RCSF est assez difficile en raison de certains problèmes de réseau courants, tels que ; la congestion et le bruit, qui pourraient entraîner la perte ou la déformation des messages, ainsi que les perturbations de la communication. Bien qu'il existe quelques résultats pour garantir le temps réel dans les RCSF, la majorité des protocoles ignorent le temps réel ou tentent simplement de traiter les données rapidement en espérant que cette vitesse soit suffisante pour respecter les délais. Ainsi, il est important pour les concepteurs de développer des protocoles temps réel qui permettent de traiter les contraintes temps réel des RCSF telles que les messages perdus, le bruit et la congestion. Cela pose un défi car très peu de résultats existent à ce jour concernant la satisfaction des exigences temps réel dans les réseaux de capteurs sans fil.

4.3 Contraintes du support sans fil

Le support des réseaux sans fil est très vulnérable aux environnements bruyants, ce qui le rend plus accessible aux différentes attaques qu'un réseau câblé. Un attaquant peut par exemple provoquer du bruit, ce qui affecte la communication. L'une des contraintes des RCSF est qu'ils soient sans infrastructure, ce qui signifie que les nœuds peuvent communiquer directement avec la station de base. La possibilité pour un attaquant d'avoir accès à un nœud spécifique lui fournit par conséquent un accès direct au centre des données. Aussi, le partage des fréquences avec d'autres réseaux contribue également à un environnement bruyant. Un autre facteur qui affecte le support sans fil est l'atténuation de la force du signal radio.

4.4 Contraintes de distance entre les nœuds

L'augmentation de la distance entre les nœuds et la station de base peut déclencher la nécessité d'utiliser plus de puissance de transmission. Ainsi, dans la conception des réseaux de capteurs sans fil, une transmission à courte portée doit être envisagée afin de réduire l'écoute, l'atténuation, et minimiser la consommation d'énergie durant la transmission. Cependant, afin de rendre le réseau plus économe en énergie en divisant les grandes distances entre les nœuds en plusieurs distances plus courtes, les concepteurs RCSF sont confrontés à un autre défi qui est la prise en charge des communications et du routage multi-sauts. Dans une communication à sauts multiples, les nœuds capteurs servent de relais pour d'autres nœuds et doivent coopérer les uns avec les autres pour trouver la route la plus efficace pour transmettre les données vers la station de base.

4.5 Autogestion

La majorité des réseaux de capteurs sans fil fonctionnent dans des zones hostiles et éloignées où l'accès humain est très difficile, dangereux voire impossible. Par conséquent, la maintenance et les réparations des capteurs, ainsi que le support des infrastructures sont très difficiles. Pour cette raison, les capteurs doivent être autonomes, doivent collaborer avec d'autres capteurs, et s'adapter aux défaillances et aux changements de l'environnement sans intervention humaine. Le défi des RCSF consiste à s'assurer que le réseau s'auto-organise, s'optimise, se protège et qu'il possède la capacité de s'auto-réparer sans encourir des frais supplémentaires de consommation énergétique.

- *Auto organisation :* Le capteur doit être capable de se configurer, d'établir des connexions avec ses voisins et de se reconfigurer en cas de défaillance de l'un d'entre eux.

- *Auto optimisation :* Le capteur doit maintenir un haut niveau d'efficacité en étant capable de gérer et surveiller l'utilisation de ses ressources.

- *Auto protection :* Le capteur doit se protéger contre les différentes attaques et menaces environnementales.

- *Auto réparation :* Le capteur doit être capable de diagnostiquer et de réparer ses défaillances.

4.6 Contraintes matérielles

La forme réduite des capteurs ainsi que leur limitation en ressources les rend difficilement capables de fonctionner avec des densités volumétriques élevées. Par conséquent, ces limitations affectent un certain nombre d'éléments de conception, tels que :

- *Tables de routage :* Les capteurs ne peuvent gérer qu'une petite allocation de mémoire. Par conséquent, les tables de routage des capteurs ne peuvent généralement contenir que la liste de leurs voisins au lieu de toutes les destinations possibles, ce qui pourrait affecter le choix des routes optimales.

- *Collecte des données :* La communication entre les nœuds capteurs, pour la transmission des paquets entre eux et vers la station de base, nécessite l'utilisation d'algorithmes et de techniques comme l'agrégation des données. Ces techniques peuvent nécessiter plus de puissance de calcul et de capacités de stockage.

- *Sécurité :* L'utilisation des réseaux de capteurs dans certains domaines critiques peut nécessiter la protection des données collectées contre les différentes attaques par l'utilisation par exemple d'un accès contrôlé aux capteurs. Pour cela, plusieurs dimensions principales sont prises en compte lors de la conception des mécanismes de sécurité comme la confidentialité, l'intégrité, la disponibilité, et l'authentification.

4.7 Tolérance aux pannes

Les nœuds capteurs peuvent se bloquer à cause de certains facteurs comme le manque de puissance, les dommages physiques ou encore les interférences environnementales. L'atteinte des nœuds capteurs ne devrait en aucun cas affecter le fonctionnement global du réseau.

4.8 Evolutivité

Le nombre de nœuds capteurs déployés dans l'étude d'un phénomène peut être de l'ordre de centaines ou de milliers. Selon l'application, le nombre peut atteindre une valeur extrême de millions. Les nouveaux systèmes doivent être capables de travailler avec ce nombre de nœuds. Ils doivent également utiliser la nature à haute densité des réseaux de capteurs. La densité peut varier de quelques nœuds à quelques centaines de nœuds capteurs dans une région et ceci selon l'application pour laquelle ils sont déployés.

4.9 Coût

Étant donné que les réseaux de capteurs sans fil se composent d'un grand nombre de nœuds, le coût d'un seul nœud justifie le coût global du réseau. En conséquent, le coût de chaque nœud capteur doit être maintenu bas.

4.10 Topologie

Le nombre important de nœuds capteurs inaccessibles et sans surveillance, qui sont sujets à des pannes fréquentes, font de la maintenance de la topologie une tâche ardue. Des centaines voire des milliers de nœuds sont déployés dans l'environnement à des dizaines de mètres les

uns des autres. Le déploiement de ce nombre élevé de nœuds nécessite une manipulation soigneuse de maintenance de la topologie.

4.11 L'environnement de déploiement

Les nœuds capteurs sont déployés soit très proches ou directement à l'intérieur du phénomène à observer. Par conséquent, les conditions dans lesquelles travaillent ces nœuds varient selon l'environnement. Ainsi, ils fonctionnent sous haute pression dans le fond d'un océan, dans des environnements difficiles tels que les champs de bataille, sous d'extrêmes chaleurs et froids comme dans la tuyère d'un moteur d'aéronef ou dans les régions arctiques.

4.12 Sécurité

Un réseau de capteurs sans fil est exposé à des menaces et des risques. Un adversaire peut compromettre un nœud capteur, altérer l'intégrité des données, intercepter les messages, ou injecter de faux messages. Par conséquent, la sécurité doit être exploitée dans les réseaux de capteurs sans fil. Il y a des contraintes dans l'intégration de la sécurité dans un RCSF telles que les limitations dans les capacités de stockage, de communication, de calcul et de traitement. Afin de concevoir des protocoles de sécurité, il faut mettre le point sur leurs limites pour atteindre des performances acceptables avec des mesures de sécurité pour répondre aux besoins d'une application.

5. Les réseaux de capteurs sans fil hétérogènes

Les réseaux de capteurs sans fil représentent une technologie innovante qui occupe une place cruciale dans le domaine de traitement des données. Cette technologie consiste principalement à combiner la communication sans fil, les fonctions de détection et les technologies embarquées. Dans un réseau de capteurs sans fil typique, tous les nœuds sont identiques. Il est par conséquent appelé réseau de capteurs sans fil homogène ou classique. Actuellement, et grâce aux progrès continus réalisés, particulièrement dans le domaine de miniaturisation des processeurs et les communications à faible puissance, une nouvelle tendance de réseaux de capteurs sans fil est apparue, dans laquelle une grande variété de nœuds sont développés. Lorsque plus d'un type de nœud est intégré dans un réseau de capteurs sans fil, il est appelé hétérogène [7] [31] [32].

Dans un réseau de capteurs sans fil hétérogène typique, on retrouve un grand nombre de nœuds identiques et quelques nœuds différents. Les nœuds identiques sont généralement peu coûteux, possèdent une puissance de calcul et une mémoire limitées. Leurs principales fonctions consistent à collecter et transmettre les données. Comme les nœuds hétérogènes sont plus puissants, ils sont utilisés pour des traitements et des tâches plus complexes nécessitant plus de puissance de calcul, plus d'énergie et de mémoire. Cette différence entre les nœuds permet de donner un équilibre au réseau [33] [34].

5.1 Avantages et limites des réseaux de capteurs sans fil hétérogènes

L'intégration de l'hétérogénéité dans un réseau de capteurs sans fil peut représenter un moyen efficace contribuant à l'augmentation de la durée de vie du réseau. Alors que la plupart des applications des réseaux de capteurs sans fil hétérogènes existantes ne diffèrent pas sensiblement de celles de leurs homologues homogènes, il existe des raisons impérieuses incitant à intégrer l'hétérogénéité dans le réseau. Les réseaux de capteurs sans fil hétérogènes peuvent présenter également certaines limites. Le tableau 1-1 répertorie les avantages et les limites des réseaux de capteurs sans fil hétérogènes.

Tableau 1-1. Avantages et limites des réseaux de capteurs sans fil hétérogènes.

Avantages	Limites
▪ Améliorer l'évolutivité des réseaux de capteurs sans fil et prolonger leur durée de vie. ▪ Réduire la consommation énergétique tout en garantissant un haut niveau de performances. ▪ Améliorer la fiabilité de transmission des données et réduire leur temps de transmission. ▪ Garantir un équilibre entre le coût et les fonctionnalités du réseau. ▪ Intégrer de nouvelles fonctionnalités à haut débit. ▪ Améliorer les mécanismes de sécurité en intégrant des protocoles plus complexes qui sont tolérés par les nœuds hétérogènes [35].	▪ Problèmes liés à la définition du nombre et de l'emplacement des nœuds hétérogènes dans le réseau. ▪ Problèmes de sécurité: Les réseaux de capteurs sans fil hétérogènes doivent tenir compte des différentes variations des capacités de sécurité.

5.1.1 Impact sur la durée de vie du réseau

Le déploiement de capteurs hétérogènes dans un réseau de capteurs sans fil a un impact important sur la durée de vie de ce dernier. En effet, les nœuds identiques peuvent envoyer

leurs données à la station de base en utilisant le nœud hétérogène le plus proche et n'ont donc pas besoin d'acheminer de grandes quantités de données. Ceci signifie que la consommation moyenne d'énergie pour transmettre un paquet à partir des nœuds homogènes vers la station de base dans les réseaux de capteurs sans fil hétérogènes sera nettement inférieure à l'énergie consommée dans les réseaux de capteurs sans fil homogènes. Avec l'augmentation de la taille du réseau, l'écart de consommation d'énergie entre ces deux types de réseaux sera plus grand.

5.1.2 Fiabilité et réduction du temps de transmission

Contrairement aux réseaux de capteurs sans fil homogènes dans lesquels les liens possèdent une faible fiabilité de transmission, due à la réduction significative du taux de livraison de bout en bout causée par chaque saut entre les nœuds, dans les réseaux de capteurs sans fil hétérogènes il y a moins de sauts entre les nœuds et la station de base, ce qui permet d'obtenir un taux de livraison de bout-en-bout beaucoup plus élevé que dans les réseaux de capteurs sans fil homogènes. Aussi, et selon le type d'hétérogénéité, le temps de transmission des données peut considérablement diminuer permettant par conséquent de réduire la latence de traitement dans les nœuds, ainsi que le temps d'attente dans la file d'attente de transmission.

5.2 Formes d'hétérogénéité

Dans un réseau de capteurs sans fil homogène ou hétérogène, les capteurs possèdent quatre éléments de base : l'unité de détection, l'unité de traitement, l'unité de communication, et l'unité d'alimentation. L'hétérogénéité peut se présenter au niveau de chacune de ces unités.

Dans un réseau de capteurs homogène, tous les capteurs sont équipés du même matériel ayant les mêmes caractéristiques et capacités. Dans un réseau hétérogène, les nœuds les plus complexes possèdent plus de besoins sur le plan matériel. Un aspect intéressant des capteurs hétérogènes est qu'ils possèdent un matériel varié. Les nœuds hétérogènes peuvent prolonger la durée de vie du réseau sans augmentation significative du coût. Ainsi, un réseau de capteurs hétérogènes prend en charge l'économie d'énergie qui se traduira par une prolongation de la durée de vie des réseaux de capteurs sans fil.

5.2.1 Hétérogénéité de calcul

Un aspect important de l'hétérogénéité concerne les capacités des nœuds. En effet, dans un réseau hétérogène, les nœuds possèdent des capacités de calcul plus élevées que celles des nœuds dans un réseau homogène. Par conséquent, les nœuds hétérogènes peuvent effectuer plus de traitement réduisant ainsi la fréquence des informations collectées et transmises à travers le réseau. En outre, basés sur des ressources de calcul puissantes, les nœuds hétérogènes peuvent fournir un traitement plus complexe des données et un stockage à plus long terme.

La présence de nœuds hétérogènes dans un réseau de capteurs sans fil peut améliorer sa fiabilité et sa longévité. En effet, comparés aux nœuds homogènes, les nœuds hétérogènes peuvent être configurés avec des microprocesseurs et des mémoires plus puissants. Ils peuvent également communiquer avec la station de base à un très haut débit.

5.2.2 Hétérogénéité d'énergie

Dans un réseau de capteurs sans fil homogène, tous les nœuds possèdent la même énergie initiale et la même complexité matérielle, tandis que dans un réseau hétérogène où il existe plusieurs types de nœuds, l'énergie et la complexité matérielle diffèrent d'un type de nœuds à un autre.

5.2.3 Hétérogénéité de transmission

Dans ce type d'hétérogénéité, les nœuds possèdent une plus grande largeur de bande et une antenne radio de réseau couvrant une plus longue distance par rapport aux nœuds homogènes, ce qui permet de fournir des transmissions de données plus fiables [36].

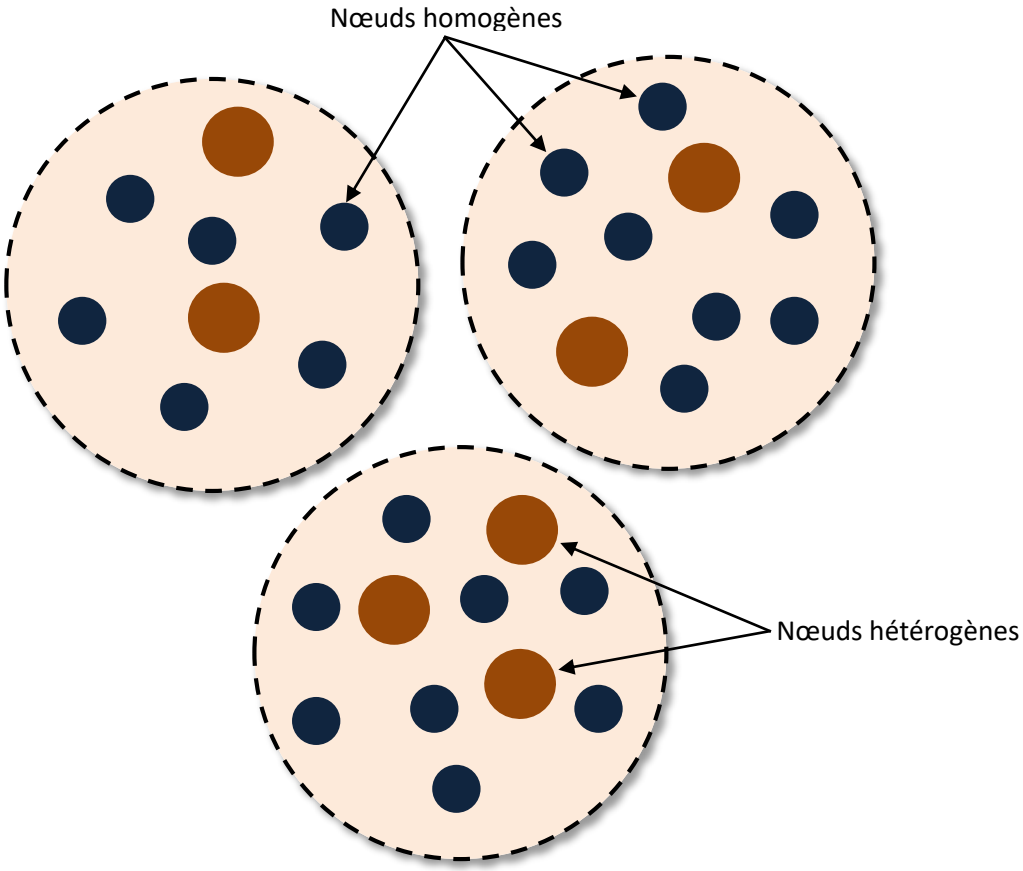


Figure 1-4. Réseau de capteurs sans fil hétérogène

5.3 Architecture des réseaux de capteurs sans fil hétérogènes

Les réseaux de capteurs sans fil hétérogènes sont composés de différents types de nœuds utilisés selon deux différentes architectures de réseaux [37] :

5.3.1 Architecture en niveaux

Dans cette architecture, les nœuds sont organisés en niveaux séquentiels. Les nœuds d'un même niveau sont tous homogènes et sont différents d'un niveau à l'autre. Le nombre de nœuds diminue du premier niveau au dernier.

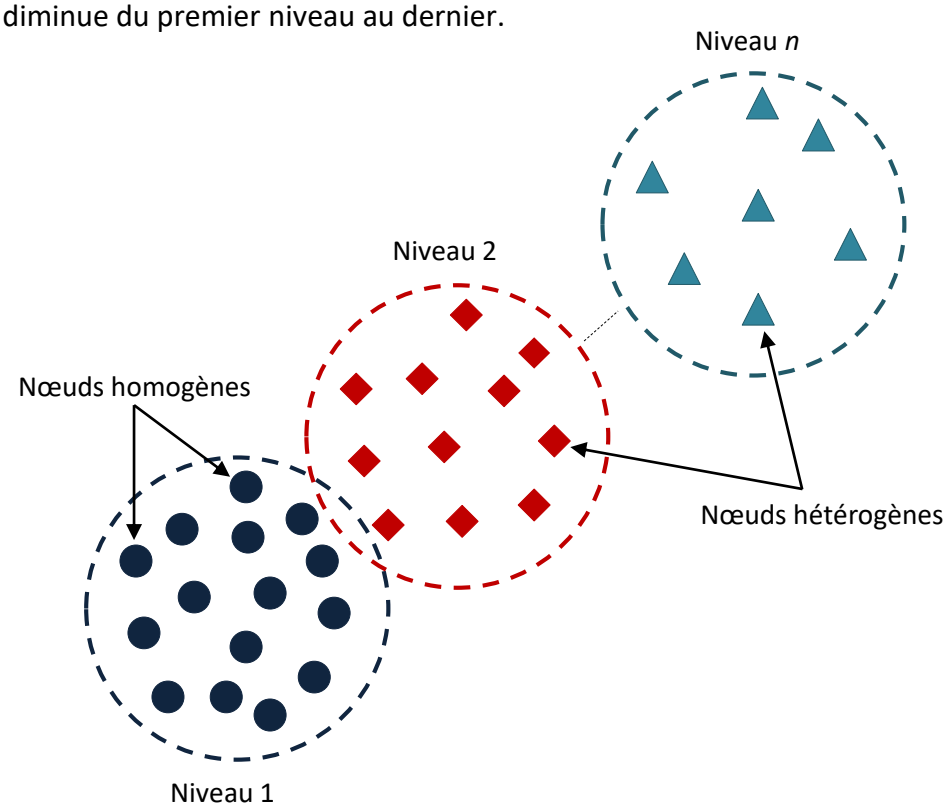


Figure 1-5. Architecture en niveaux

5.3.2 Architecture hiérarchique

Dans l'architecture hiérarchique, Les nœuds sont organisés comme une forêt composée de plusieurs arbres. Les arbres contiennent de nombreux niveaux. Chaque niveau comprend des nœuds homogènes et des nœuds hétérogènes. Deux organisations arborescentes hiérarchiques sont définies :

- *Organisation à saut unique*

  Dans cette organisation, le nombre de niveaux dans chaque arborescence est égal au nombre de types de nœuds existants. Dans chaque niveau, le nœud hétérogène est le parent auquel les nœuds homogènes sont directement liés. La figure 1-6 illustre un exemple de ce type d'organisation, dans laquelle chaque arbre est composé de deux niveaux. Les nœuds sont directement connectés à leur parent via un chemin à un seul saut.

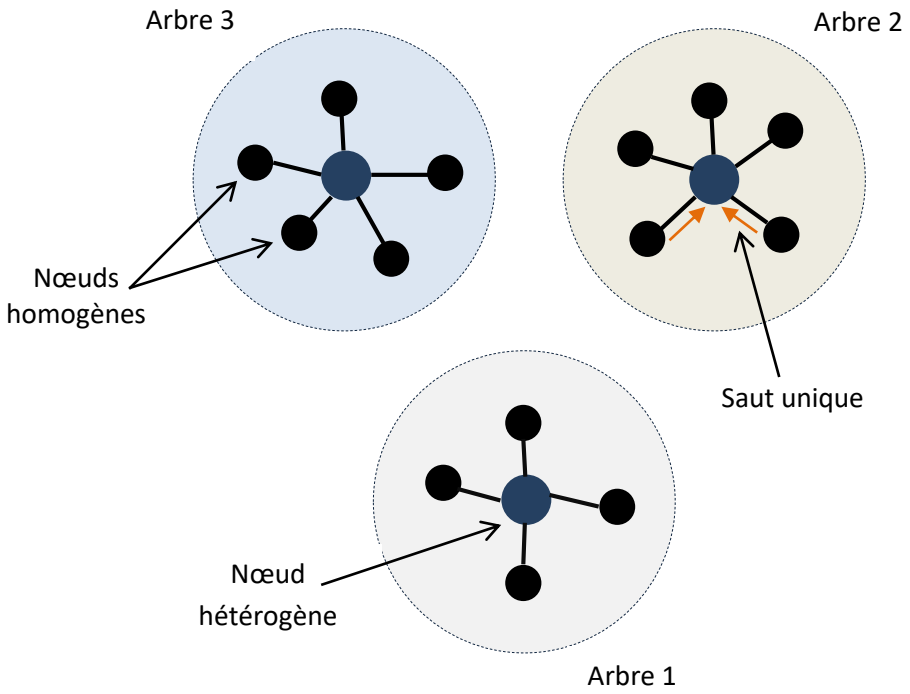


Figure 1-6. Organisation hiérarchique à saut unique

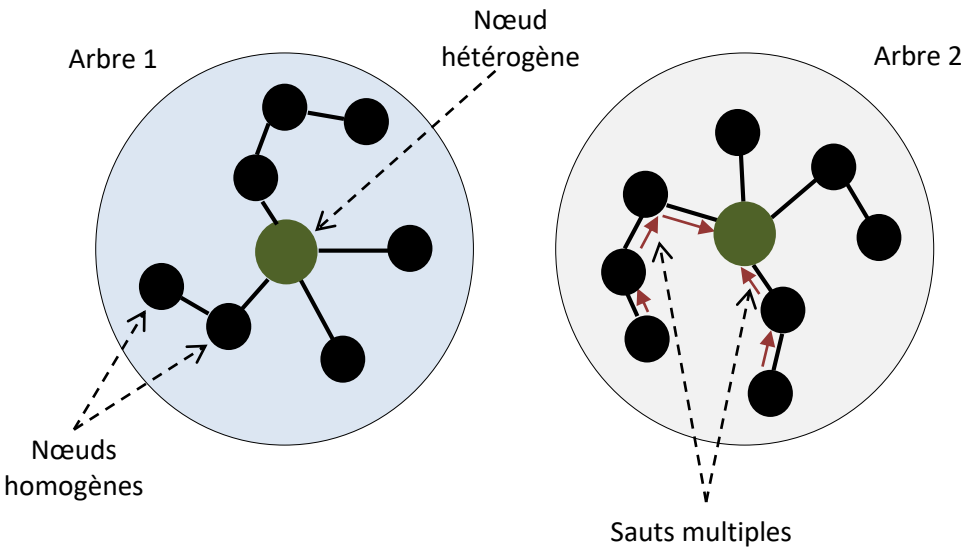


Figure 1-7. Organisation hiérarchique à sauts multiples

- *Organisation à sauts multiples*

L'organisation à sauts multiples diffère de l'organisation à saut unique de deux manières : dans une organisation à sauts multiples, le nombre de niveaux est indépendant du nombre de types de nœuds existants. De plus, les nœuds sont indirectement connectés à leur parent via des chemins multi-sauts. La figure 1-7 illustre un exemple d'organisation multi-sauts. Chaque arbre contient deux types de nœuds. Les nœuds homogènes sont indirectement connectés à leur parent via un chemin multi-sauts.

6. Conclusion

L'objectif de ce premier chapitre était de présenter le domaine de notre travail à savoir les réseaux de capteurs sans fil, qui gagnent de plus en plus de terrain et sont de plus en plus ubiquistes dans tous les aspects de la surveillance de l'environnement. Nous avons survolé d'une manière générale ce domaine, présenté l'anatomie des capteurs qui représentent le composant principal de ce type de réseaux, leurs domaines d'application, ainsi que les contraintes liées à leur conception.

Comme notre travail est basé plus particulièrement sur les réseaux de capteurs sans fil hétérogènes, nous avons mis la lumière sur l'aspect d'hétérogénéité dans les réseaux de capteurs sans fil, l'architecture des réseaux de capteurs sans fil hétérogènes, les différents types d'hétérogénéité, son impact sur les réseaux de capteurs sans fil ainsi que ses limites dans ce domaine.

Chapitre 2

Big Data dans les réseaux de capteurs sans fil

Chapitre2 : Big Data dans les réseaux de capteurs sans fil

1. Introduction

Les réseaux de capteurs sans fil et plus particulièrement les réseaux de capteurs sans fil hétérogènes se sont développés rapidement ces dernières années, et leur déploiement représente un avantage pour les nouvelles applications [38]. La grande utilisation des applications dédiées aux réseaux de capteurs sans fil ainsi que la diversité des domaines concernés, ont contribué à augmenter le volume des données collectées et traitées. En effet, lorsque les réseaux grandissent et gagnent en volume et en espace de déploiement, les données collectées et traitées croissent de façon exponentielle nécessitant ainsi un traitement plus efficace, et rendant par conséquent les méthodes de traitement des données traditionnelles difficiles à utiliser.

La croissance des données générées par les capteurs peut devenir exponentielle au fil du temps. Ainsi, des centaines de milliers de données sont collectées et doivent être traitées efficacement pour prendre les bonnes décisions [39]. Les technologies de l'information traditionnelles pour le traitement des données peuvent prendre en charge des quantités limitées de données générées dans les réseaux de capteurs sans fil. Cependant, ces technologies deviennent rapidement limitées et coûteuses pour le traitement de très grandes quantités de données. Pour cela, une nouvelle technologie visant à traiter et à stocker de gros volumes de données connue sous le nom Big Data [40] [41] a été récemment mise au point. La technologie Big Data peut représenter un aspect innovant dans les réseaux de capteurs sans fil en rationalisant la collecte, l'analyse, le traitement et le stockage des données volumineuses.

La technologie Big Data [42] [43] peut représenter une solution efficace pour collecter, analyser, stocker et transmettre les données dans de vastes réseaux de capteurs sans fil. En effet, comme les applications des RCSF augmentent massivement, les capteurs déployés sont chargés de produire les données en grands volumes faisant ainsi des réseaux de capteurs sans fil des contributeurs clés à la technologie Big Data.

Le paradigme du Big data dans les réseaux de capteurs sans fil [44] est jeune et émergent. Il a été initialement adapté aux réseaux câblés, mais cette technologie gagne du terrain dans les réseaux de capteurs sans fil, augmentant ainsi le besoin de nouvelles technologies et architectures pour gérer les données. Le terme Big Data est généralement utilisé pour caractériser de grands ensembles de données qui peuvent être complexes et donc difficiles à gérer par les méthodes de traitement des données conventionnelles. Dans un réseau de capteurs sans fil, ces énormes masses de données sont générées toutes les minutes et doivent être collectées par les nœuds capteurs avant d'être transmises à la station de base.

2. Concept du Big data

Big Data est une nouvelle technologie qui représente de grands ensembles de données qui peuvent être complexes et difficiles à manipuler en utilisant les outils de traitement des données traditionnels. En comparaison avec les ensembles de données traditionnels, la technologie Big Data définie des masses de données non structurées nécessitant plus de gestion temps réel. Le paradigme du Big Data peut également être défini comme l'association entre la collecte de données volumineuses et les algorithmes dédiés permettant des exploitations qui peuvent largement dépasser l'application classique des processus et méthodologies analytiques des données.

2.1 Définitions

Le terme Big Data représente un concept très abstrait. En effet, bien que cette technologie soit généralement liée aux masses importantes de données, différentes définitions lui sont attribuées dans la littérature [45] [46] [47]. D'une manière générale, Big data désigne des ensembles de données qui ne peuvent pas être gérés et traités dans un temps tolérable en utilisant les outils informatiques traditionnels. Les définitions suivantes représentent celles les plus utilisées afin de mieux comprendre le concept Big data :

- Big Data est définie par Apache Hadoop comme étant «des ensembles de données qui ne peuvent pas être capturés, gérés et traités par des ordinateurs généraux dans une portée acceptable».

- L'agence de conseil mondiale McKinsey & Company a présenté Big Data comme étant la prochaine frontière pour l'innovation, la concurrence et la productivité. Aussi, Big Data est définie comme des ensembles de données ne pouvant pas être acquis, stockés et gérés par un logiciel de base de données classique. Pour l'agence McKinsey & Company, Big Data ne représente pas seulement les masses importantes des données, mais aussi l'échelle croissante des données et leur gestion qui ne peuvent pas être gérées par les technologies traditionnelles.

- L'institut national des normes et de la technologie NIST a défini Big Data comme suit : «Big Data désigne les données dont le volume, la vitesse d'acquisition ou la représentation des données limitent la capacité d'utiliser des méthodes relationnelles traditionnelles pour effectuer une analyse efficace, ou les données qui peuvent être traitées efficacement avec d'importantes technologies».

- Gartner, Inc. (Gartner IT Glossary, n.d.) a défini Big Data comme suit : «Big Data sont des actifs d'information à volume élevé, à grande vitesse et à grande variété qui exigent un rapport coût/efficacité, des formes innovantes de traitement de l'information pour une compréhension et une prise de décision améliorées. »

- La Fondation TechAmerica (TechAmericaFoundation's Federal Big Data Commission, 2012) a défini Big Data comme suit: «Big Data est un terme qui décrit de grands volumes de données à grande vitesse, complexes et variables qui nécessitent des techniques et des technologies avancées pour permettre la capture, le stockage, la distribution, la gestion et l'analyse des informations. "

2.2 Histoire de développement de la technologie Big Data

Les premières apparitions de la technologie Big Data datent de la fin des années 1970. En effet, l'augmentation importante des volumes des données et l'insuffisance des capacités de stockage et de traitement ont mené à la survenue d'une technologie nommée «machine de base de données» [48] dédiée pour l'analyse et la sauvegarde des données. Par la suite et vers les années 1980, un autre système basé sur le concept de ne rien partager est apparu. Ce système est dédié pour les bases de données parallèles, et permet de traiter les volumes importants des données [49]. Dans ce système, chaque machine possède son propre processeur, stockage et disque. Parmi les systèmes de bases de données parallèles il y a le système Teradata [50].

Le développement des différents services Internet représente un grand défi pour Big data du fait de la manipulation des données volumineuses par les moteurs de recherche. Pour cela, des modèles de programmation GFS [51] et MapReduce [52] ont été développés par Google.

En janvier 2007, Jim Gray, a lancé l'idée de développer une nouvelle génération d'outils informatiques permettant de gérer les données volumineuses. Le terme Big Data a été présenté pour la première fois en juin 2011 par un rapport de recherche intitulé « Extracting Values from Chaos » [53], permettant aux entreprises mondiales telles que Oracle, IBM, Microsoft, Google, Amazon et Facebook de commencer leurs projets basés Big Data.

Depuis 2005, plusieurs travaux de recherche sont dédiés à la technologie Big Data. Par exemple, Big Data a été citée en 2008 par Nature, en 2011 par Science, en 2012 par le Consortium européen de recherche pour l'informatique et les mathématiques (ERCIM), en 2012 au Davos Forum en Suisse, et de 2012 à 2013 par l'agence de recherche internationale Gartner.

3. Dimensions du Big data

La technologie Big Data représente un paradigme technologique pour une grande variété de données générées à une grande vitesse et à un grand volume. D'une manière générale, le volume de données est la première caractéristique qui décrit la technologie Big data. Cependant, d'autres caractéristiques sont apparues pour décrire ce paradigme. Ainsi, la technologie Big Data était souvent référée aux dimensions "3V" (volume, vitesse et véracité) qui représentent les éléments essentiels caractérisant cette technologie [54].

Récemment, d'autres dimensions telles que "5V" [55] [56] sont de plus en plus utilisées pour décrire la technologie Big Data.

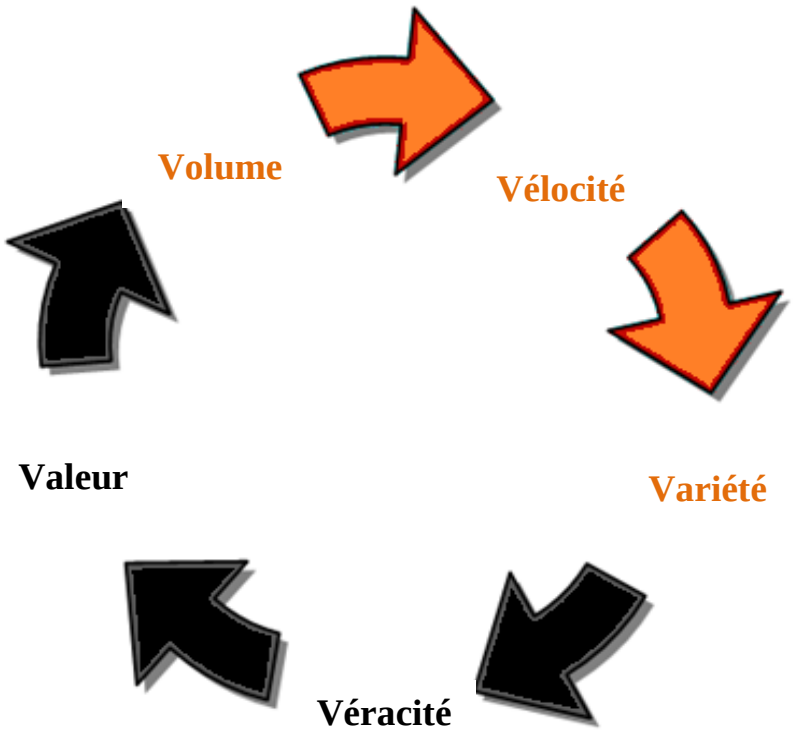


Figure 2-1. Dimensions de la technologie Big Data.

Le tableau suivant décrit les 5 dimensions Big Data :

Dimension	Définition
Volume	Décrit les grandes quantités de données nécessitant un stockage, un traitement et une organisation
Vélocité	Correspond à la vitesse de génération, de traitement et de transmission des données
Variété	Décrit les différents types de données collectées à partir de diverses sources, et qui sont traitées et stockées dans différents formats
Véracité	Concerne le problème de bruit, les différentes anomalies dans la grande quantité de données et le degré de signification des données stockées par rapport au problème analysé
Valeur	Décrit la qualité des énormes quantités de données ainsi que les relations explicites ou implicites entre les données

Tableau 2-1. Dimensions 5V du Big Data.

Les dimensions du Big data précédemment définies sont dépendantes les unes des autres. En effet, le changement d'une dimension entraine conséquemment le changement d'une autre dimension.

Malgré le fait que les dimensions 5V de la technologie Big Data sont les plus utilisées, d'autres dimensions ont récemment vu le jour [57]. De nos jour, nous pouvons compter jusqu'à dix caractéristiques et propriétés du concept Big Data :

- *Validité :* La validité définie l'exactitude des données par rapport à leurs objectifs. Elle correspond aussi à la qualité des données ainsi que leur cohérence.

- *Vulnérabilité :* Dans la technologie Big Data il est très difficile de sécuriser les volumes importants des données. La vulnérabilité des données Big Data commence généralement à leur point d'entrée. En effet, la source des données du système Big Data pourrait être compromise. Un autre point de vulnérabilité concerne le nombre massif de données à gérer. Les copies ou les données supplémentaires conservées peuvent présenter un risque important de sécurité.

- *Visualisation :* Une des caractéristiques des données Big Data est leur visualisation difficile. En effet, les limitations en mémoire et en temps de réponse ainsi que la multitude de variables résultant de la variété des données, posent de véritables défis pour les outils de visualisation des données Big Data.

- *Volatilité :* Avant l'arrivée de la technologie Big data, les données classiques avaient tendance à être stockées indéfiniment dans les bases de données du fait que leurs dépenses de stockage n'étaient pas élevées. Aujourd'hui, et avec l'arrivée de la technologie Big Data et la croissance considérable des quantités de données, la volatilité des données est étudiée et des règles de disponibilité et de récupération des données sont mises au point.

- *Variabilité :* La variabilité dans la technologie Big Data concerne l'incohérence des données. Elle est aussi due aux différentes dimensions de données résultant de la variation des types et des sources de données. Un autre point de variabilité concerne la vitesse de chargement des données dans les bases de données.

4. Outils de la technologie Big data

Le paradigme du Big Data est basé sur l'utilisation de plusieurs outils analytiques. En effet, plusieurs outils Big Data inondent le marché. Le déploiement de ces différents outils permet d'améliorer la rentabilité et de mieux gérer le temps d'analyse des données. Dans ce qui suit, nous présentons les outils Big Data les plus répondus :

4.1 Hadoop

L'écosystème Apache Hadoop (High Availability Distributed Object Oriented Platform) [58] [59] est une infrastructure open source dédiée pour le traitement des données Big Data. Hadoop est largement utilisé dans les applications exhaustives de données comme l'analyse des données Big data. Il offre un environnement de traitement parallèle et distribué flexible et tolérant aux pannes. Ainsi, Hadoop permet de créer des applications et de les exécuter sur de larges ensembles de données. Les ensembles de données sont répartis sur plusieurs grappes ou clusters d'ordinateurs permettant d'obtenir des puissances de calcul élevées à des coûts très faibles.

Hadoop utilise des modèles de programmation simples pour la gestion distribuée des données Big Data. Il est basé sur quatre modules de traitement principaux :

- Le *Hadoop Common* composé d'un ensemble d'utilitaires et de bibliothèques de sérialisation prenant en charge les modules Hadoop ;

- le *Hadoop Distributed File System* (HDFS) qui est la couche de stockage Hadoop qui permet de stocker de gros volumes de données non structurées en créant plusieurs copies des blocs de données et en les distribuant sur les différents nœuds des clusters, permettant ainsi de réaliser des calculs fiables et rapides ;

- Le *Hadoop YARN* (Yet Another Resource Negotiator) responsable de la gestion des ressources du cluster, de la planification des tâches et de la surveillance des opérations de traitement des nœuds de cluster individuels ;

- Le *Hadoop MapReduce*, qui implémente l'algorithme MapReduce.

1) Architecture de Hadoop

Hadoop est basé sur l'utilisation d'une architecture maître-esclave permettant de réaliser des traitements distribués sur les données et de les stocker, en se basant sur deux couches principales à savoir MapReduce et HDFS. Les couches de l'architecture de Hadoop sont composées des éléments suivants :

- **NameNode :**
  Dans l'architecture Hadoop, on trouve un espace de noms qui contient tous les fichiers et les répertoires. Ces derniers constituent ce qu'on appelle le NameNode.

- **DataNode :**
  Le DataNode permet aux différents blocs d'interagir ensemble et avec l'utilisateur. Il permet aussi de gérer les états des nœuds HDFS.

- **MasterNode ou nœud maître :**
  Le traitement parallèle des données Big Data est effectué au niveau de ce nœud en se basant sur l'algorithme MapReduce.

- **SlaveNode ou nœud esclave :**
  Dans l'architecture de Hadoop, les données sont stockées en vue d'être utilisées dans les traitements complexes en utilisant les nœuds esclaves.

Le traitement dans Hadoop est géré par les outils JobTracker et TaskTracker. L'outil JobTracker est exécuté sur un nœud maître du cluster Hadoop. Il possède connaissance des ressources disponibles, et son rôle consiste à planifier les demandes d'application et à les répartir sur les nœuds TaskTracker pour qu'elles soient exécutées. Pendant l'exécution des applications MapReduce, l'état des nœuds est envoyé au JobTracker afin de coordonner leur exécution.

Le processus TaskTracker est exécuté sur les nœuds esclaves du cluster Hadoop. Son rôle consiste à recevoir les demandes de traitement envoyées par le JobTracker. TaskTracker permet d'envoyer les mises à jour du statut des applications MapReduce exécutées sur son nœud au JobTracker.

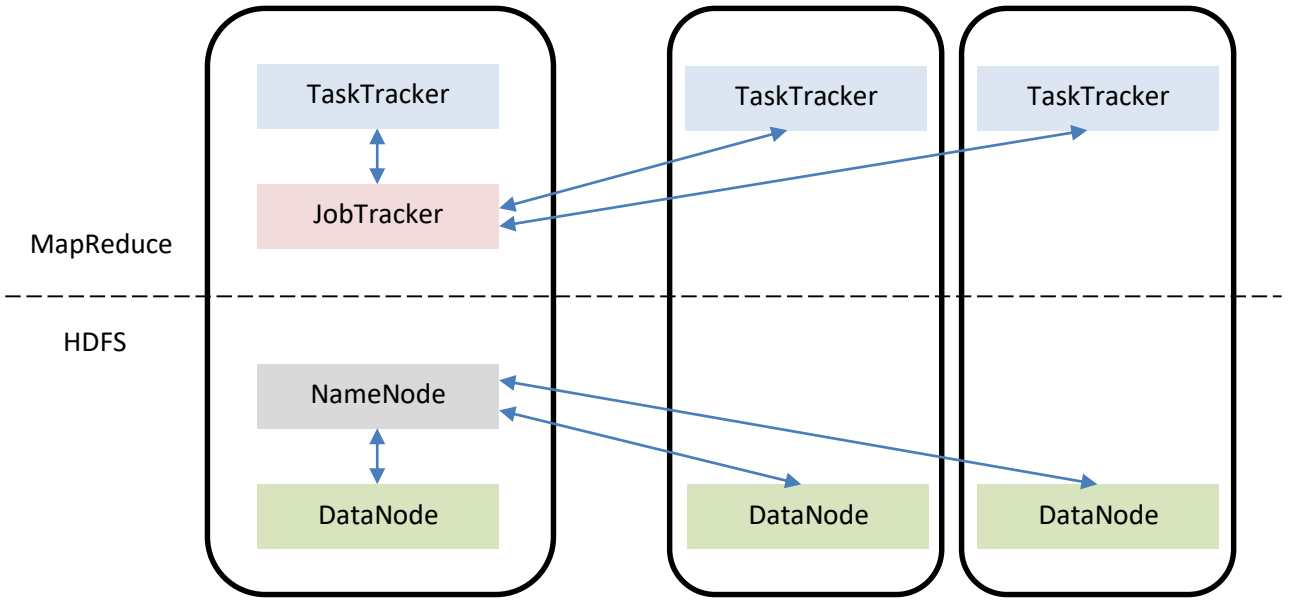


Figure 2-2. Architecture de Hadoop.

2) Avantages de Hadoop

L'utilisation de Hadoop offre plusieurs avantages, parmi lesquels nous pouvons citer les avantages suivants :

- L'écosystème open source offert par Hadoop est robuste et permet de réaliser l'analyse des données selon les besoins des utilisateurs.
- Hadoop permet de traiter les données indépendamment de leur type et leur source.
- Hadoop offre un traitement flexible des données grâce à son architecture distribuée.
- Il permet de réaliser des traitements rapides avec des coûts faibles.
- Hadoop est facile à utiliser du fait qu'il prend en charge le traitement parallèle.

4.2 Apache Spark

Apache Spark [60] [61] est un système de traitement parallèle des données dédié pour la technologie Big Data. Apache Spark est un système open source construit sur la base de Hadoop MapReduce. Il est codé en Scala [62], et permet le traitement distribué des données volumineuses en se basant sur des machines en clusters. Spark permet aussi d'augmenter les performances des applications d'analyse Big Data par la prise en charge du traitement « In-memory » qui permet de réaliser plusieurs tâches différentes en utilisant les mêmes données.

Apache Spark représente le concurrent principal du système Hadoop et une alternative possible à ce dernier. En effet, Spark permet de remédier au problème de lenteur du traitement due au stockage des données dans le système de fichier distribué avant de pouvoir démarrer chaque étape.

Avantages de Spark

L'utilisation de l'outil Apache Spark offre plusieurs avantages, parmi lesquels nous pouvons citer les avantages suivants :

- Spark offre un Framework complet et unifié permettant de répondre aux besoins de traitement de données variées.
- Rapidité : Il permet une exécution rapide des applications sur des Clusters Hadoop allant à 100 fois plus vite en mémoire et 10 fois plus vite sur le disque.
- Spark supporte les requêtes SQL.
- Propose des fonctionnalités d'apprentissage automatique, dit Machine Learning.
- Permet les traitements orientés graphe.

4.3 HPCC

HPCC (High precision congestion control) [63] est un outil Big Data open source développé par « LexisNexis Risk Solution ». Il permet un traitement de données efficace basé sur une architecture et un langage de programmation uniques.

HPCC offre une interface d'analyse et de traitement des données à large échelle. Il offre aussi un langage de programmation déclaratif et cohérent orienté données de niveau élevé appelé Enterprise Control Language (ECL), permettant d'améliorer les délais de réponse aux résultats de traitement des données.

L'architecture du système HPCC est implémentée sur des clusters permettant de réaliser des traitements parallèles et des transmissions des données Big Data à des hauts niveaux de performance. Le traitement parallèle des données est réalisé par l'outil lots (Thor). Les transmissions et livraisons des données sont réalisées par des fichiers de données indexées (ROXIE).

Avantages de HPCC

L'utilisation de HPCC offre plusieurs avantages, parmi lesquels nous pouvons citer les avantages suivants :

- HPCC permet un traitement parallèle efficace des données Big Data en utilisant moins de code.
- Il permet l'optimisation du code pour la réalisation des traitements parallèles des données Big data.
- HPCC offre une interface facile à utiliser.
- Il offre des performances rapides avec la prise en charge de requêtes temps réel.

4.4 Apache Storm

Apache Storm [64] est un outil Big data open source permettant le traitement distribué en temps réel des données Big data à des vitesses très élevées. Storm est aussi connu sous le nom de Hadoop temps réel. L'outil Storm offre une architecture de traitement des données simple et efficace avec une faible latence. Les applications développées avec Storm sont appelées Topologies. Il supporte aussi tout type de langage de programmation.

Avantages de Storm

L'utilisation de Storm offre plusieurs avantages, parmi lesquels nous pouvons citer les avantages suivants :

- Rapidité et facilité d'utilisation ;
- Tolérance aux pannes. Si un nœud meurt, le processus redémarre sur un autre nœud ;
- Fiabilité ;
- Évolutivité ;
- Permet le traitement parallèle des données.

4.5 Apache Cassandra

Apache Cassandra [65] est un système de base de données NoSQL qui permet le traitement et le stockage des données volumineuses. L'utilité d'utiliser une base de données NoSQL est qu'elle permet de prendre en charge de larges volumes de données tout en s'appuyant sur une API simple.

Apache Cassandra est un système distribué permettant le traitement des données réparties sur des serveurs de bases de données multiples. Par le déploiement de clusters Cassandra multi-nœuds. La base de données Cassandra possède la possibilité de s'adapter à toute augmentation de données. Les données traitées sont stockées dans un format tabulaire.

Avantages de Cassandra

L'utilisation du système Apache Cassandra offre plusieurs avantages, parmi lesquels nous pouvons citer les avantages suivants :

- Prise en charge des données structurées, non structurées ou semi-structurées ;
- Scalabilité ;
- Disponibilité ;
- Facilité de stockage ;
- Gestion des données à haute vélocité.
- Fiabilité : la défaillance d'un des nœuds n'affecte par les performances générales ;
- Traitement des données à des vitesses considérables.
- Tolérance aux pannes : réplication des données sur plusieurs nœuds.

4.6 Apache Hive

Apache Hive [66] est un système Big data permettant d'exécuter d'une manière facile et rapide des requêtes SQL-like [67] et d'analyser efficacement des données de Apache Hadoop. Pour cela, Apache Hive traduit les programmes rédigés en langage HiveQL en une ou plusieurs tâches Java MapReduce. Il permet ensuite d'organiser les données en tableau pour le fichier Hadoop Distributed File System (HDFS) et d'exécuter les tâches sur un cluster. Hive décrit les données stockées dans HDFS en utilisant une base de données relationnelle appelée metastore afin de garantir la persistance des données.

Avantages de Hive

L'utilisation du système Apache Hive offre plusieurs avantages, parmi lesquels nous pouvons citer les avantages suivants :

- Simplicité d'utilisation due à l'utilisation du langage SQL-like, permettant l'accélération de l'insertion des données.
- Scalabilité.
- Sécurité par la réplication des données critiques.
- Vitesse pouvant aller jusqu'à 100 000 requêtes par heure.

4.7 Apache Flink

Apache Flink [68] est un framework Big Data dédié pour le traitement distribué et les calculs des flux de données continus. Flink permet d'effectuer des calculs à des vitesses élevées à n'importe quelle échelle.

Grâce à son architecture, Flink offre plusieurs API à différents niveaux d'abstraction. Il propose aussi des bibliothèques dédiées.

Apache Flink offre plusieurs fonctionnalités comme les transformations de flux, le traitement parallèle et les affectations de ressources. Il est caractérisé par son débit élevé et sa faible latence. Flink est open source et dispose par conséquent d'une large communauté utilisant ses fonctionnalités.

Avantages de Flink

L'utilisation du système Apache Flink offre plusieurs avantages, parmi lesquels nous pouvons citer les avantages suivants :

- Haute performance
- Faible latence
- Offre un système de traitement de données distribué
- Tolérance aux fautes
- Offre un calcul itératif
- Plateforme hybride

5. Technologies liées au concept du Big data

La technologie Big Data est étroitement liée à d'autres technologies fondamentales qui sont présentées dans ce qui suit :

5.1 Cloud Computing

Le Cloud Computing [69] est une technologie basée sur l'utilisation de masses très importantes de données, dont la gestion et le stockage sont centralisés au niveau d'un nuage. Les techniques de stockage et de gestion du Cloud Computing sont déployées afin de fournir des solutions pour le stockage et le traitement des données Big Data, ce qui permet de mettre l'accent sur les capacités de stockage des données Big Data. Ainsi, le stockage distribué et le calcul parallèle de la technologie du Cloud Computing permettent de gérer efficacement et d'améliorer l'efficacité d'analyse Big Data.

Les avancées remarquables des deux technologies du Big Data et du Cloud Computing font qu'elles deviennent de plus en plus étroitement liées [70]. En effet, Le Cloud Computing permet de fournir des ressources au niveau du système tandis que la technologie du Big Data

permet de fournir des capacités de traitement des données efficaces fonctionnant au niveau supérieur qui est géré par le Cloud Computing, permettant ainsi que l'application du Big Data soit basée sur le Cloud Computing. Ainsi, le Cloud Computing fournit le calcul et le traitement des données Big Data.

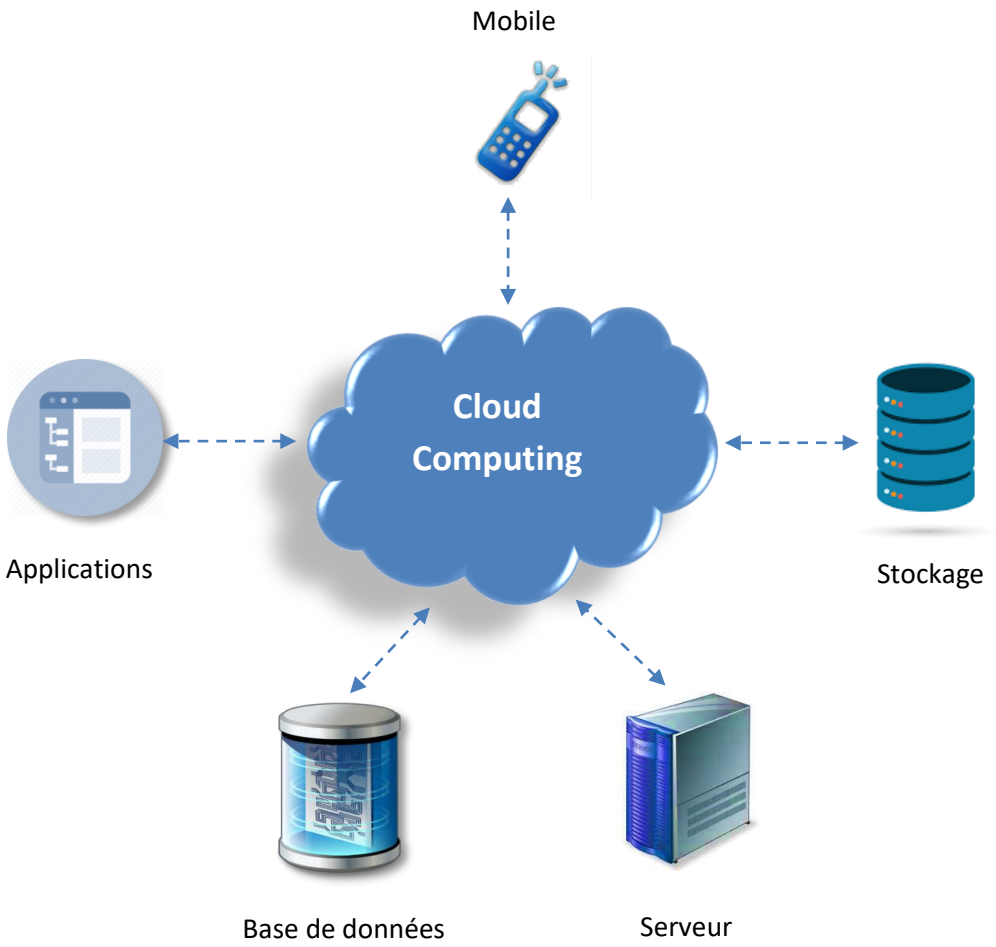


Figure 2-3. Cloud Computing

5.2 Internet des Objets IoT

La technologie de l'internet des objets IoT [71] est une technologie récente dans laquelle les appareils, tels que les équipements mobiles et les appareils électroménagers, sont équipés avec des quantités importantes de capteurs dont l'objectif principal consiste à collecter différents types de données, comme les données environnementales, militaires ou astronomiques. L'internet des objets permet de générer des données Big Data dont les caractéristiques diffèrent des données Big data générales. Ceci est dû à la différence entre les données recueillies comme l'hétérogénéité et la variété. Selon Intel, Les données Big Data ressemblent aux données IoT sur trois points essentiels :

- Les masses importantes de données générées par les terminaux ;
- Les caractéristiques semi-structurées et non structurées des données Big Data et IoT ;

- L'utilité des données après leur analyse.

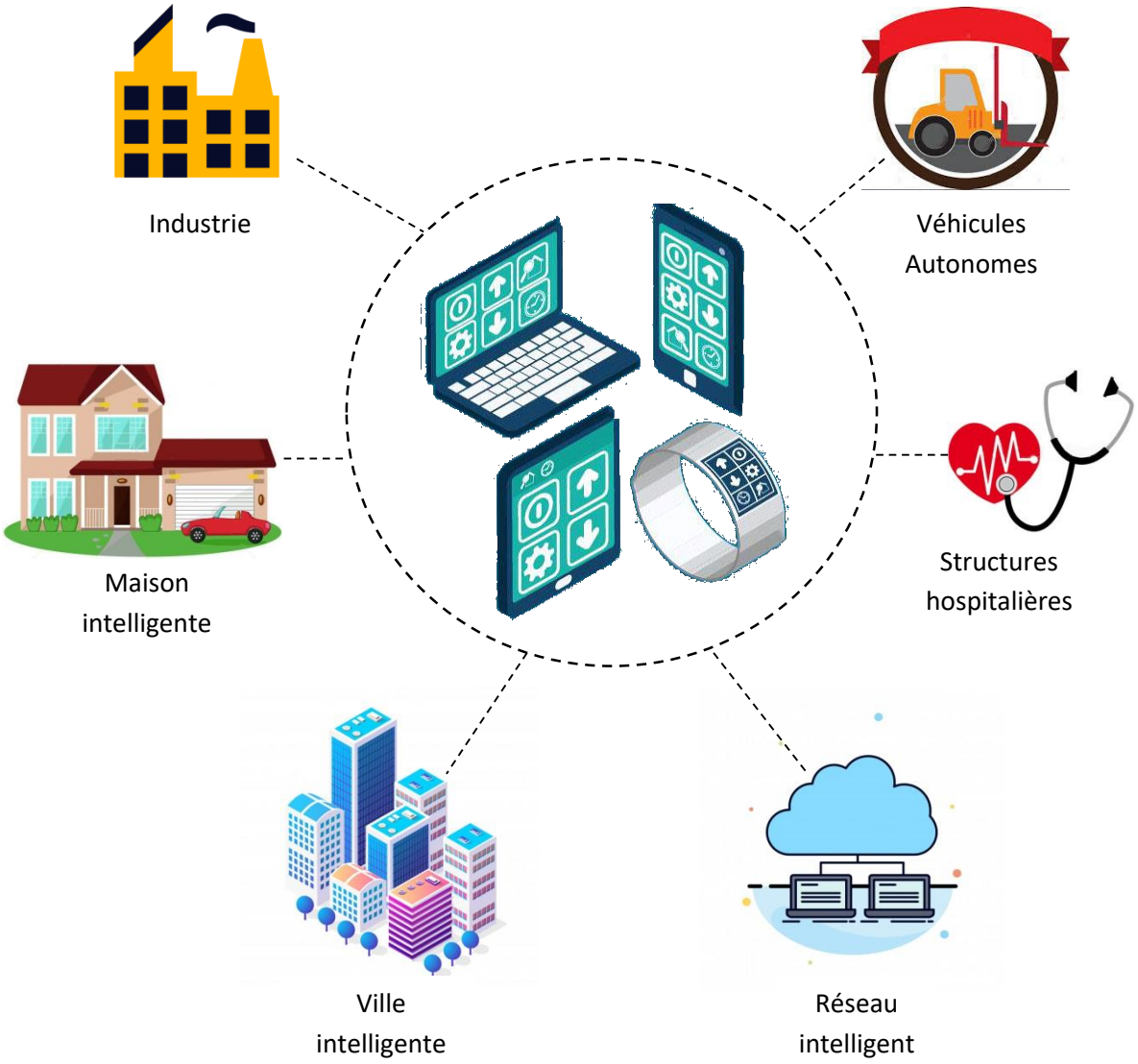


Figure 2-4. Exemple d'IoT

L'intégration de la technologie Big Data devient une nécessité afin de pouvoir promouvoir le développement de l'internet des objets [72]. En effet, l'intégration de la technologie Big Data permettra d'assurer le succès de l'IoT du fait que l'application de la technologie Big Data à l'IoT permet d'accélérer les progrès de recherche et les modèles commerciaux de l'IoT comme le développement de villes intelligentes.

5.3 Technologies de centralisation des données

La technologie Big Data représente un grand défi pour les technologies de centralisation des données [73]. En effet, ces technologies doivent fournir une plate-forme de stockage centralisé pour les grandes masses de données en plus de leur collecte, leur gestion et leur

organisation selon les principaux objectifs auxquels elles sont dédiées, et qui sont généralement très exigeants en ce qui concerne la capacité de stockage, de traitement, et de transmission. La technologie Big Data permet d'assurer une croissance explosive des infrastructures liées à la centralisation des données. Ces infrastructures doivent être fournies à un grand nombre de nœuds. La combinaison de ces technologies permet d'assurer une amélioration considérable des capacités de traitement des données centralisées, et une réduction des coûts de leur développement. En effet, cette combinaison permet de renforcer les capacités d'acquisition, de traitement, d'organisation, d'analyse et d'application des données Big Data.

5.4 Hadoop

L'application de l'écosystème Hadoop est largement reconnue dans la technologie Big Data [74]. A titre d'exemple, nous pouvons citer l'utilisation sur plusieurs serveurs Yahoo de grands clusters Hadoop comportant des milliers de nœuds afin d'assurer plusieurs services comme la recherche et le filtrage des spam. Aussi Hadoop est utilisé par plusieurs entreprises comme IBM afin d'effectuer l'analyse et le calcul distribués des grandes quantités de données.

Selon les différentes recherches réalisées, la technologie de programmation parallèle MapReduce offerte par l'infrastructure Hadoop permet de fournir aux utilisateurs des données Big Data une interface qui offre des services plus pratiques, tout en réduisant les coûts inutiles.

6. Big data dans les réseaux de capteurs sans fil

Récemment et en raison des progrès technologiques réalisés dans les différents domaines, les données des RCSF ont connu une croissance très importante. Ces données représentent une collection de valeurs et de variables analysées et utilisées pour prendre des décisions. En général, les données collectées ne sont pas significatives, en particulier lorsque le réseau contient un nombre limité de capteurs, mais elles deviennent de plus en plus volumineuses lorsqu'elles sont fournies par des millions de capteurs.

L'implication des réseaux de capteurs sans fil dans de nombreuses applications générant des données massives telles que les systèmes de sécurité et de surveillance, nécessite le traitement de grandes quantités de données, ce qui constitue un défi majeur aux méthodes de traitement des données existantes, et nécessitant par conséquent l'utilisation de techniques et d'outils pouvant gérer convenablement et efficacement ces données volumineuses.

L'explosion de la technologie du Big Data dans les réseaux de capteurs sans fil est un phénomène très récent qui vient remédier au problème des données volumineuses auxquelles sont confrontés ces derniers [11] [12]. Le terme Big Data est généralement utilisé

pour définir les grands ensembles de données ayant une plus grande complexité et qui sont difficiles à stocker, gérer et analyser avec les outils de traitement de données conventionnels.

6.1 Contraintes de la technologie Big Data

L'explosion de la technologie Big Data dans les RCSF devient difficile à gérer en fonction des RCSF et des exigences Big Data [75]. Dans ce qui suit sont présentées les contraintes principales de la technologie Big Data dans les RCSF :

6.1.1 Disponibilité des données

Les données volumineuses doivent être organisées afin d'être disponibles dans les RCSF d'une manière précise et complète pour qu'elles puissent être utilisées à temps, ce qui rend leur gestion et gouvernance très complexes.

6.1.2 Traitement des données

Le traitement des données correspond à l'ensemble des opérations réalisées sur les données comme l'analyse des données collectées par les nœuds, l'agrégation des données, et le calcul réalisé sur les données. Afin de réaliser le traitement des données Big Data dans les RCSF, il est nécessaire de définir la localisation des données collectées. Aussi, le traitement des données n'est pas seulement effectué au niveau de chaque nœud, mais il doit interpréter tous les événements distribués et les données qui leurs sont associées.

6.1.3 Gestion des données

La gestion des données représente une contrainte critique dans la technologie Big Data. En effet, il est important de gérer efficacement les volumes importants et en croissance rapide des grands ensembles de données. Pour cela, il faut définir des architectures adaptées au mécanisme d'interrogation et de stockage des données. Pour réaliser une gestion efficace des données, les données doivent être collectées à un niveau centralisé auquel sont adressées les requêtes par la suite, ce qui permet d'éviter les transmissions redondantes des données. Par conséquent, l'interrogation des données distribuées est bien prise en charge et la récupération des données à plusieurs niveaux est effectuée.

6.1.4 L'hétérogénéité des données

Les données Big Data se composent à la fois de données structurées et non structurées. Le traitement des données structurées qui sont généralement en petits volumes est réalisé par les systèmes de base de données existants, mais ces derniers ne sont pas conçus pour traiter les données non structurées. Par conséquent, l'hétérogénéité des données Big Data ainsi que leur volume important posent des difficultés quant à leur stockage et leur analyse.

6.2 Challenges de la technologie Big Data dans les RCSF

La croissance et l'émergence notables des différentes technologies réseau ainsi que l'explosion de leur utilisation, ont entraîné une augmentation impressionnante dans le processus de génération et de gestion des données Big Data. Ainsi, et en raison de cette augmentation, le développement d'applications dédiées rencontre des obstacles et des défis qui doivent être surmontés afin de pouvoir manipuler efficacement le volume impressionnant des données déployées. Des travaux proposés dans la littérature ont classifié les challenges de la technologie Big Data [9] [57] [76] [77] [78]. Dans ces classifications, les auteurs se concentrent principalement sur les dimensions Big data, la gestion et le traitement des données. Étant donné que notre point d'intérêt concerne le support de la technologie Big data dans les RCSF, nous supposons que les challenges Big Data peuvent être combinés avec ceux des RCSF afin de répondre aux exigences de ces deux technologies. Pour cela, nous proposons une nouvelle classification des challenges Big Data dans les RCSF [79] basée sur quatre axes clés qui représentent les principaux piliers sur lesquels sont fondés les réseaux de capteurs sans fil:

6.2.1 Le Clustering

Le Clustering [80] est le premier pilier de la technologie des réseaux de capteurs sans fil. Il représente l'étape principale de la hiérarchie de classification qui est étroitement liée à toutes les autres étapes. Le Clustering détermine l'organisation et le déploiement des nœuds dans le réseau, leur positionnement par rapport aux autres nœuds ainsi que la station de base (BS). Il détermine également les chemins de routage, l'ordre dans lequel les données seront transmises, la manière dont elles sont transmises et les stratégies utilisées pour leur transmission. Le Clustering définit également les stratégies de communication entre les différents nœuds du réseau [81].

6.2.2 Le traitement des données

Le traitement des données Big Data dans les réseaux de capteurs sans fil est un défi critique qui nécessite des stratégies efficaces pour collecter, analyser, stocker et agréger les volumes importants de données :

- La collecte de données volumineuses est une tâche de traitement difficile [82]. En effet, même si les données reçues par chaque nœud du réseau apparaissent insignifiantes, les données générées par l'ensemble du réseau peuvent générer des quantités très importantes de données. Ainsi, la collecte de données volumineuses devient critique, ce qui nécessite l'utilisation de techniques adaptées pour relever ce challenge.

- Les données collectées par les nœuds capteurs nécessitent une analyse et un stockage [83] [84]. Des méthodes d'analyse adaptées sont nécessaires pour gérer

simultanément les volumes croissants de données. Ces méthodes doivent également être améliorées afin de réduire le temps de réponse et d'économiser plus d'énergie afin de prolonger la durée de vie du réseau.

- L'agrégation de données dans les RCSF basés sur la technologie Big Data est un paradigme important, qui représente une solution efficace permettant de traiter les données volumineuses en combinant les données qui sont similaires, éliminant ainsi le problème des données redondantes et réduisant par conséquent la consommation des ressources.

6.2.3 La sécurité des données

Les données groupées et traitées doivent être sécurisées. La sécurité joue un rôle fondamental dans les RCSF basés sur la technologie Big Data, et représente l'un de leurs principaux challenges [85]. Différents mécanismes de sécurité adaptés à la technologie Big Data dans les RCSF peuvent être utilisés pour protéger les données à tous les niveaux du réseau. Cependant, le problème de sécurité des données volumineuses n'est pas mis en avant dans les RCSF basés Big Data en dépit de sa grande importance, et les différentes techniques et mécanismes proposés dans la littérature sont dédiés principalement aux autres challenges.

6.2.4 La consommation énergétique

Le Clustering des nœuds peut représenter un moyen efficace permettant de relever le défi de la consommation énergétique en regroupant les nœuds d'une manière efficace, permettant que les distances de communication et les quantités de messages transmis puissent être réduites, réduisant par conséquent leur consommation d'énergie et la consommation d'énergie de l'ensemble du réseau. En outre, et pour un traitement optimal des données, des mécanismes économes en énergie doivent être déployés. De plus, et en raison des données volumineuses et des contraintes de Clustering et de traitement, il est essentiel de déployer des mécanismes permettant de sécuriser le réseau et protéger par conséquent les données Big Data des différentes attaques lors du déploiement du réseau et du traitement des données, tout en économisant de l'énergie.

Le mécanisme de Clustering est étroitement lié aux autres challenges de la classification proposée. En effet, l'organisation des réseaux constitue la base sur laquelle les stratégies de traitement et de sécurité des données sont définies.

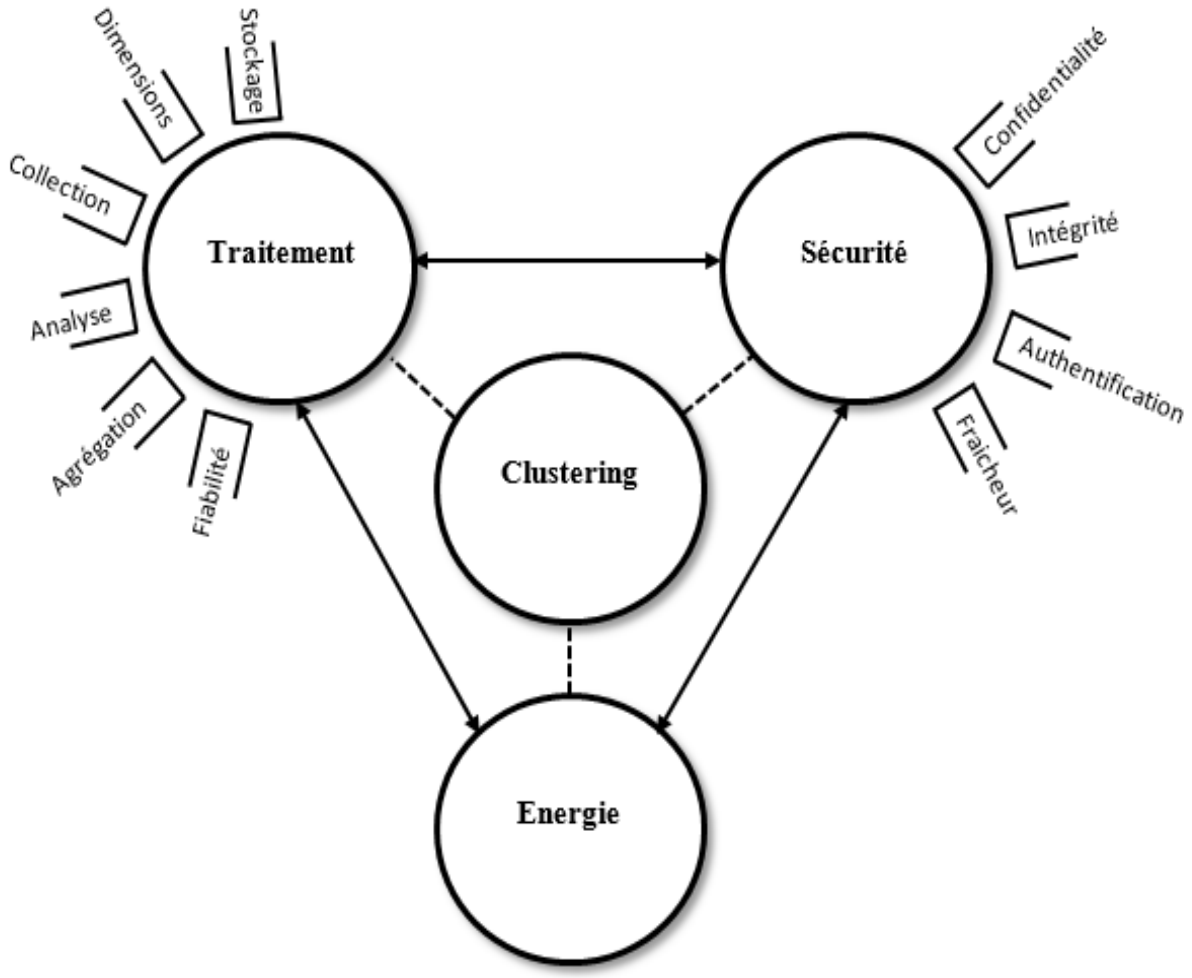


Figure 2-5. Classification proposée des challenges Big Data dans les RCSF

6.3 Stratégies basées sur les challenges Big Data dans les RCSF

Dans ce qui suit, nous présentons un état de l'art survolant les différentes stratégies proposées basées sur les challenges de la technologie Big Data dans les RCSF :

6.3.1 Clustering économe en énergie

Les algorithmes de Clustering proposés sont principalement basés sur la technique de maximisation (EM) qui représente un algorithme de Clustering classique utilisant la distribution gaussienne des nœuds du réseau. Ensuite, un nombre optimal de clusters est calculé sur la base d'une fonction objective définie comme la somme de l'énergie requise des données et les messages de transmission des requêtes de données.

Des techniques de Clustering dans les RCSF basés sur la technologie Big data ont été proposées dans la littérature :

Dans [86] et [87], les auteurs ont proposé des approches de Clustering basées sur la collecte éco énergétique des données Big data. Cette collecte est basée sur l'utilisation d'une station de base mobile, et est considérée comme une solution efficace pour les réseaux de capteurs densément distribués. Aussi, et afin de réduire la consommation d'énergie, les auteurs ont proposés des dérivations optimales des clusters.

Dans [88], un algorithme de Clustering K-means distribué adapté aux larges RCSF est proposé. Les auteurs ont basé leur travail sur le Clustering distribué, effectué sur chaque capteur qui collabore avec ses capteurs voisins afin de réduire la surcharge de communication entre tous les nœuds capteurs. L'algorithme proposé utilise une technique de régularisation «attribue-poids» pour atteindre l'affectation idéale des poids d'attributs afin de déterminer les fonctions essentielles.

6.3.2 Collecte des données

Des techniques de collecte des données dans les RCSF basées sur la technologie Big data ont été proposées dans la littérature :

Les auteurs du travail proposé dans [89], et afin d'aborder la collecte continue des données Big Data ainsi que l'efficacité énergétique des RCSF densément distribués, ont proposé une méthode efficace basée sur la mobilité des nœuds puits pour la sélection des clusters. Pour cela, un algorithme de regroupement des données nommé DGC-SOM (Dynamical Growing Cellular Self Organizing Map) basé sur SOM [90] [91] est généré afin de sélectionner la tête du cluster. L'algorithme DGC-SOM proposé permet l'introduction de nouveaux nœuds basés sur un seuil de perte d'énergie. DGC-SOM offre des performances élevées en termes de débit, de taux de paquets, de livraison de paquets, de transmission et de qualité de service (QoS).

Dans les travaux proposés dans [92] et [93], les auteurs avaient comme objectif principal de réduire le retard généré par la station de base mobile dans le réseau sans fil. Pour cela, les auteurs ont proposé des solutions pour la collecte des données qui sont basées sur des collecteurs M-mobiles. Les collecteurs utilisent des trajectoires de longueur fixe réduisant par conséquent la complexité générée par leur mobilité. Les méthodes de collecte des données proposées sont précédées par une technique améliorée de Clustering basée sur la technique de maximisation. Ensuite, et dans l'objectif de minimiser la consommation énergétique du réseau, le nombre optimal de clusters est calculé à l'aide d'une fonction objective basée sur la somme de la consommation d'énergie d'un cycle de collecte.

Dans [94], les auteurs visent à maintenir une faible distorsion structurelle des données collectées et à réduire le nombre de nœuds actifs dans le réseau. Pour cela, ils ont proposé une architecture nommées SFDC (Structure Fidelity Data Collection) ayant pour objectif principal de maintenir la fidélité des données en termes de similitude structurelle.

L'architecture proposée est basée sur une approche de distorsion structurelle qui utilise une métrique de fidélité d'images permettant d'évaluer la qualité de l'image déformée.

Dans [95], les auteurs ont proposé un protocole de collecte des données volumineuses appelé HDERP (Hybrid Dynamic Energy Routing Protocol). Le protocole proposé est caractérisé par sa fiabilité qui garantit un chemin de transmission des données saut par saut, une durée de vie plus longue et une consommation énergétique efficace, réduisant par conséquent le délai de bout en bout.

6.3.3 Analyse des données

Des techniques d'analyse des données dans les RCSF basés sur la technologie Big data ont été proposées dans la littérature :

Dans [14], les auteurs ont utilisé des approches analytiques pour calculer la consommation d'énergie des nœuds et le nombre optimal de clusters pour deux modèles de collecte des données mobiles : MULE (modèle multi sauts) et SENMA (modèle à saut unique) expérimentés avec différents scénarios de réseaux à très grande échelle dans un court laps de temps. Aussi, et afin de minimiser la consommation d'énergie dans les RCSF à grande échelle, les auteurs ont proposé des modèles multi-clusters permettant de déterminer le nombre optimal de clusters.

Dans [96], les auteurs visent à aborder l'évolutivité et les limites de la corrélation des données Big data dans les RCSF, pour la détection de capteurs défectueux. Pour cela, les auteurs ont proposé une approche de détection des valeurs aberrantes extensibles au Big Data, basée sur la corrélation et la régression dynamique SMO (Sequential Minimal Optimization). Basée sur l'architecture MapReduce de Hadoop, l'approche proposée vise à découvrir les attributs fortement corrélés et à détecter efficacement les nœuds anormaux, réduisant ainsi le nombre de fausses alarmes.

Dans [8], les auteurs ont étudié les récentes architectures proposées liées à l'analyse des données Big Data dans l'IoT (Internet des objets). Les travaux visent principalement à surmonter les challenges liés à l'analyse d'une grande quantité de données. Les auteurs ont également exploré les grandes plateformes de traitement et d'analyse des données générées par l'IoT, et étudié les exigences en matière de Big Data et d'analyse de l'IoT. Sur la base de paramètres importants, les auteurs ont taxinomisé les solutions de la technologie Big Data et d'analyse de l'IoT.

6.3.4 Efficacité énergétique

Des techniques permettant une consommation faible en énergie dans les RCSF basés sur la technologie Big data ont été proposées dans la littérature :

Les auteurs dans [97], et afin de résoudre le problème d'itinéraire multi agents, ont proposé un protocole économe en énergie pour la construction de nœuds du « spanning tree » comme base d'un schéma de planification d'itinéraire de routage. La solution proposée repose tout d'abord sur un modèle de réseaux de capteurs répartis multi-agents (DWSN) et un modèle de consommation d'énergie. Ensuite, l'algorithme d'itinéraire de routage nommé DMAIP (Dfocus-Modulation-type Active Image Processing) est construit. L'algorithme de routage est également étendu pour réduire la transmission à longue distance.

Dans [98], les auteurs ont proposé un nouveau protocole d'agrégation des données Big Data économe en énergie. Le protocole proposé est basé sur l'algorithme de collecte des données volumineuses en temps réel (RTBDG), qui permet de diviser récursivement le RCSF en différentes parties symétriquement autour d'un nœud racine. Les auteurs ont également basé leur travail sur l'algorithme de Clustering distributif HEED (Hybrid Energy-Efficient Distributed Clustering) basé sur l'énergie hybride et le coût de communication lors de l'étape de sélection des CHs [99]. Les auteurs dans [100] ont également basé leurs travaux sur l'algorithme RTBDG.

6.3.5 Stockage des données

Des techniques permettant le stockage efficace des données dans les RCSF basés sur la technologie Big data ont été proposées dans la littérature :

Dans [101], les auteurs ont développé un algorithme de stockage et de récupération des données Big Data pour les RCSF, basé sur une distribution non uniforme des nœuds, qui utilise un protocole de routage simple. L'objectif de l'algorithme est d'estimer la distribution réelle et les adresses des nœuds capteurs tout en considérant la redondance des données entre les nœuds voisins.

7. Conclusion

Nous avons présenté à travers ce chapitre la technologie Big data. Nous avons abordé les différentes définitions qui lui sont attribuées dans la littérature, l'histoire de son développement, ses principaux concepts et dimensions, ses outils analytiques, ainsi que les différentes technologies qui lui sont liées.

Nous avons introduit par la suite la technologie Big data dans les RCSF. En effet, et grâce à l'émergence de nouvelles technologies de traitement des données et d'analyse, la technologie Big Data est perçue comme étant un aspect très innovant des RCSF. La combinaison de ces deux technologies implique l'apparition de plusieurs challenges qui doivent être résolus en parallèle. Ainsi, nous avons proposé une nouvelle classification de ces challenges en fonction des besoins et des défis des RCSF et de la technologie Big Data.

Chapitre 3

Agrégation des données Big Data dans les réseaux de capteurs sans fil

Chapitre3 : Agrégation des données Big Data dans les réseaux de capteurs sans fil

1. Introduction

Les nœuds capteurs ont pour objectif principal de recueillir les mesures de l'environnement dans lequel ils sont déployés, et de collaborer entre eux afin de transmettre les données vers un centre de traitement appelé station de base. Les capteurs sont généralement limités en énergie et possèdent une capacité de stockage réduite. Comme les nœuds capteurs sont contraints en énergie, il est inefficace pour l'ensemble des capteurs de transmettre les données directement à la station de base. Les données générées par les capteurs voisins sont souvent redondantes et hautement corrélées. En outre, les quantités de données générées dans les grands RCSF dépassent la capacité de traitement de la station de base. Ainsi, il est nécessaire de faire recours à des méthodes permettant de fusionner les données sur les nœuds intermédiaires afin de réduire le nombre de paquets transmis vers la station de base, assurant par conséquent une économie d'énergie et de bande passante. Ceci peut être accompli par le mécanisme d'agrégation des données [102] [103].

L'agrégation des données est définie comme étant le processus de fusion des données provenant des capteurs au niveau de nœuds intermédiaires et la transmission des résultats vers la station de base, éliminant par conséquent les transmissions redondantes. Ce mécanisme important tente de recueillir les données les plus cruciales des capteurs et de les rendre disponibles à la station de base de manière économe en énergie et avec une latence minimale des données. Ces paramètres sont très importants dans de nombreuses applications telles que la surveillance de l'environnement où la fraîcheur des données représente un facteur important. Il est aussi essentiel de développer des algorithmes d'agrégation des données économes en énergie de sorte que la durée de vie du réseau soit améliorée.

1.1 Agrégation des données dans les réseaux de capteurs sans fil classiques

Dans un réseau de capteurs sans fil classique, tous les nœuds sont identiques, possédant ainsi les mêmes rôles et sont équipés de batteries de la même puissance. Dans de tels réseaux, l'agrégation des données est généralement effectuée en se basant sur un routage centré données.

1.2 Agrégation des données dans les réseaux de capteurs sans fil hétérogènes

Dans un réseau de capteurs sans fil hétérogène, certains nœuds sont plus puissants en termes de capacités et d'énergie, et sont par conséquent dédiés à effectuer des tâches plus complexes

et plus consommatrices en énergie. L'agrégation des données dans ces réseaux est effectuée au niveau de ces nœuds particuliers, ce qui réduit le nombre de messages transmis vers la station de base, améliorant ainsi l'efficacité énergétique du réseau.

2. Avantages et limites de l'agrégation des données

L'agrégation des données offre plusieurs avantages, parmi lesquels nous pouvons citer les avantages suivants :

- Amélioration de la robustesse et de la précision des données obtenues par l'ensemble du réseau, et ceci par l'élimination des redondances.
- Réduction de la charge du trafic et du nombre de transmissions.
- L'agrégation des données permet d'accéder rapidement aux données permettant par conséquent une meilleure prise de décisions et une amélioration des services.
- Réduction de la consommation énergétique et amélioration par conséquent de la durée de vie du réseau.

Néanmoins, le mécanisme d'agrégation des données peut aussi présenter certains inconvénients :

- Les nœuds responsables de l'agrégation sont sujets à des attaques. Si de tels nœuds sont compromis, la station de base ne peut s'assurer de la fiabilité des informations agrégées qui lui sont transmises.
- Les données agrégées sont généralement de faible qualité, ce qui peut conduire aux fausses détections positives et négatives.

3. Types d'agrégation des données

L'agrégation des données dans les RCSF peut être divisée en trois classes :

3.1 Agrégation centralisée

L'agrégation centralisée des données est réalisée dans des groupes de nœuds appelés clusters ou grappes qui sont formés en utilisant un protocole de Clustering ou de classification. Les protocoles de Clustering procèdent d'abord par la définition de zones qu'on appelle les clusters. Les données sont par la suite agrégées dans ces zones grâce à un chef de cluster ou cluster-Head. Ce dernier peut éventuellement changer au cours du temps afin de répartir au mieux la consommation d'énergie entre tous les nœuds du réseau.

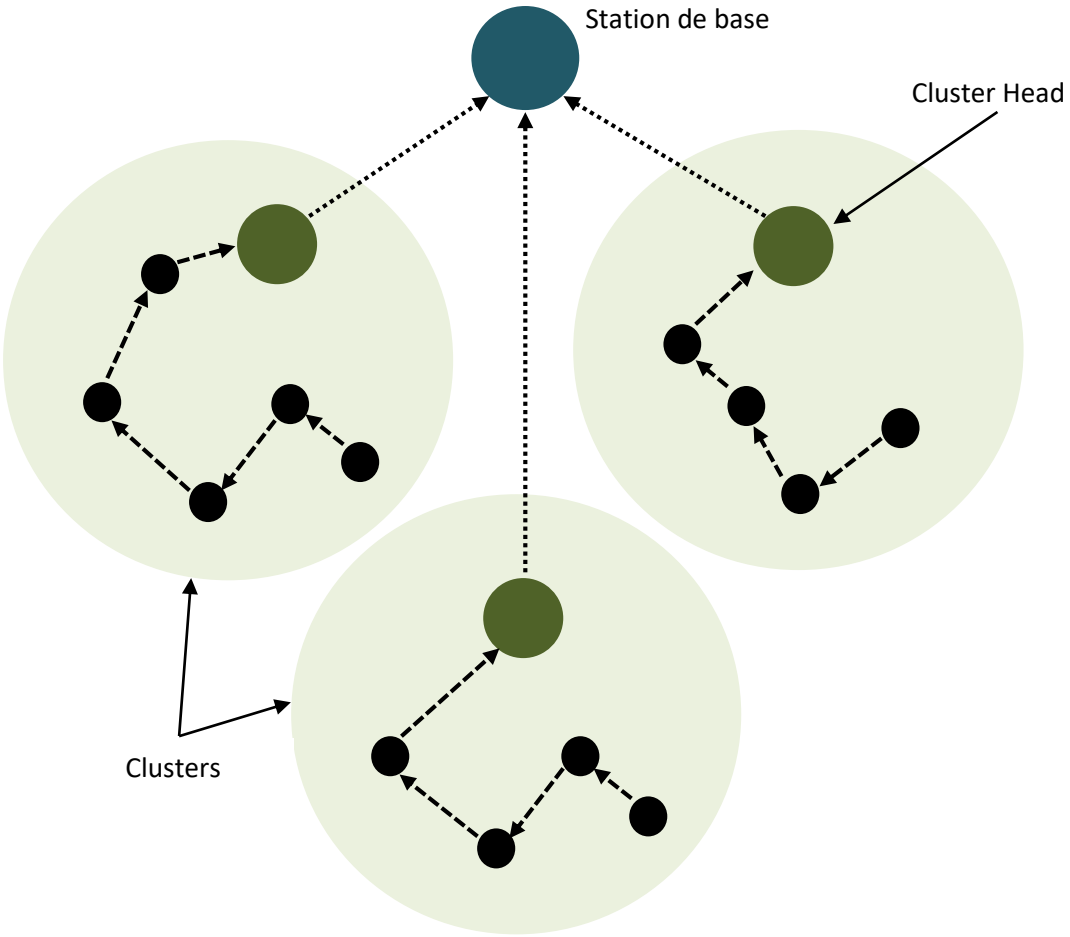


Figure 3-1. Agrégation centralisée

3.2 Agrégation distribuée

L'agrégation distribuée des données est effectuée dans un arbre à plusieurs niveaux, dans lequel la station de base représente la racine et les nœuds représentent les feuilles. Les données sont recueillies au niveau des feuilles et parcourent l'arbre jusqu'à l'arrivée à la racine. L'inconvénient de cette approche réside dans le fait que dans le cas de perte de paquets de données à n'importe quel niveau de l'arbre, les données seront perdues non seulement pour un seul niveau, mais pour toute la sous-arborescence connexe.

3.3 Agrégation hybride

Ce type d'agrégation combine les caractéristiques des deux autres types d'agrégation.

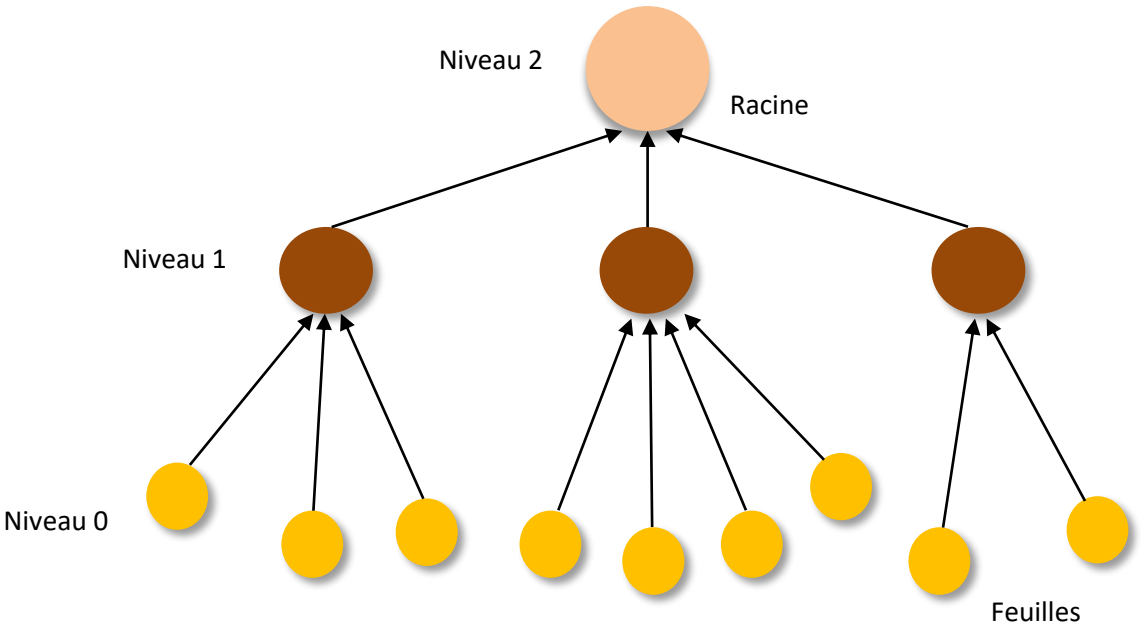


Figure 3-2. Agrégation distribuée

4. Protocoles d'agrégation des données

Le processus d'agrégation des données est généralement réalisé au niveau du routage de données. Ainsi, la majorité des protocoles de routage dédiés aux RCSF adoptent cette technique afin d'améliorer les performances du réseau.

Les protocoles de routage sont classifiés en se basant sur les différents types d'agrégation. Dans ce qui suit, nous présentons les protocoles de routage les plus répondus, qui utilisent l'agrégation des données, et ce selon les types d'agrégation définis précédemment :

4.1 Protocoles d'agrégation centralisée

4.1.1 LEACH

LEACH (Low-energy adaptive Clustering hierarchy) [104] est un protocole de routage hiérarchique des RCSF qui vise principalement à augmenter la durée de vie du réseau. L'organisation des nœuds capteurs dans LEACH se fait dans des clusters. Dans chaque cluster, un seul nœud est élu pour devenir chef de cluster. Ce dernier est le seul nœud autorisé à transmettre les données vers la station de base.

Dans LEACH, tous les nœuds ont les mêmes caractéristiques et possèdent initialement la même quantité d'énergie. La consommation énergétique des nœuds est effectuée au même degré, et chaque nœud est capable d'estimer son énergie restante. Aussi, tous les nœuds peuvent se connecter directement entre eux, ainsi qu'au chef de cluster.

L'agrégation des données dans LEACH est réalisée par le chef de cluster qui reçoit les données à partir de tous les nœuds, puis les accumule et les agrège pour les envoyer vers la station de base. LEACH est capable d'adapter, d'auto-organiser et de regrouper les nœuds.

Le protocole LEACH est basé sur le concept de cycles. Chaque cycle est divisé en deux étapes : l'étape de configuration et l'étape de stabilisation. L'étape de configuration correspond à la configuration des clusters. L'étape de stabilisation correspond au transfert des données.

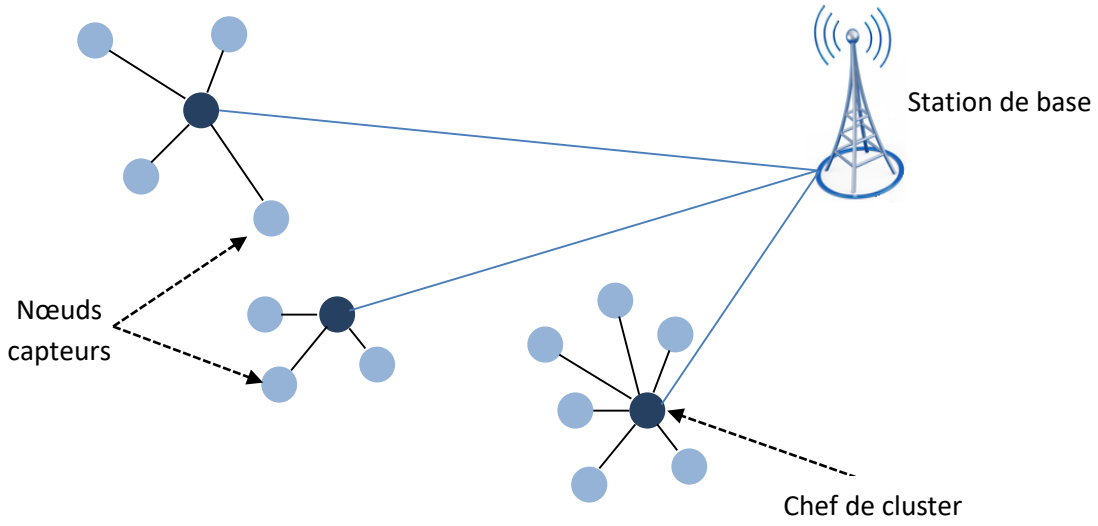


Figure 3-3. Organisation des nœuds dans LEACH

4.1.2 PEGASIS

PEGASIS (Power Efficient Gathering in Sensor Information Systems) [105] est un protocole de routage hiérarchique qui utilise une approche basée chaîne. L'approche basée chaine consiste à organiser les nœuds de sorte qu'ils forment une chaîne afin de recevoir et de transmettre les données entre les nœuds voisins. Si l'un des nœuds meurt, la chaîne est reconstruite pour contourner le nœud mort.

Dans l'approche basée chaine de PEGASIS, un seul chef de cluster est élu périodiquement d'une manière aléatoire. Son rôle principal consiste à transmettre les données vers la station de base. Chacun des autres nœuds ne communique qu'avec son voisin proche et transmet les données à tour de rôle vers la station de base.

Dans PEGASIS, les données collectées sont transmises d'un nœud vers un autre. Ces données sont agrégées et transmises vers le chef de cluster qui se charge de leur transmission vers la station de base

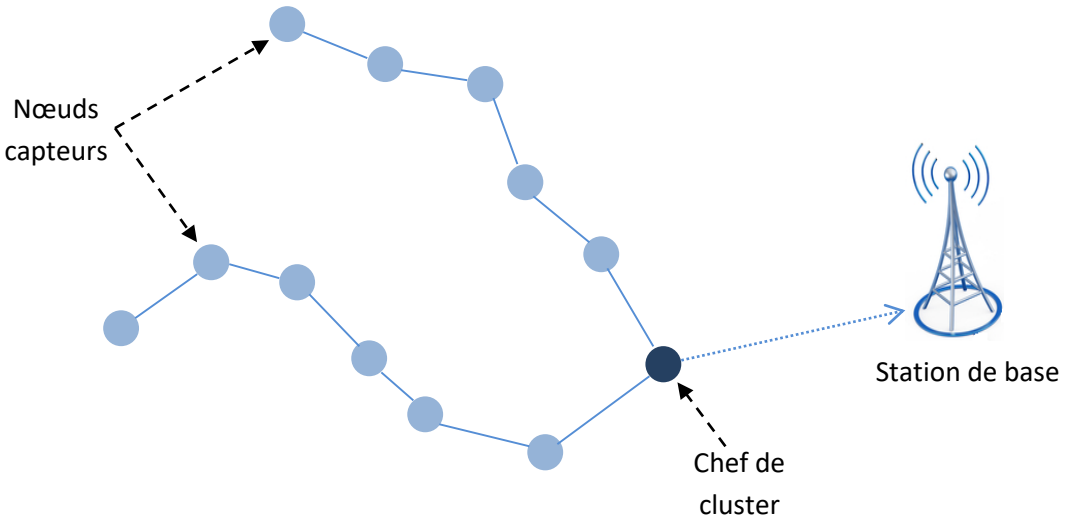


Figure 3-4. Organisation des nœuds dans PEGASIS

4.1.3 TEEN

TEEN (Threshold sensitive Energy Efficient sensor Network protocol) [106] est un protocole de routage dédié principalement pour les réseaux réactifs.

Le principe de fonctionnement de TEEN est simple : le chef de cluster diffuse deux valeurs à ses membres à chaque changement de cluster. La première valeur est Hard Threshold (HT) qui représente la valeur seuil pour l'attribut détecté. Le nœud ayant détecté cette valeur doit allumer son émetteur et faire rapport à son chef de cluster. La deuxième valeur est Seuil souple (ST) qui représente un changement dans la valeur de l'attribut détecté qui alerte le nœud pour qu'il allume son émetteur et commence la transmission.

Lorsqu'une valeur d'un attribut atteint un seuil fixe pour la première fois, le nœud correspondant allume son émetteur et envoie les données détectées qui seront stockées dans une variable interne dans le nœud, appelée la valeur détectée (SV). Les données de la période actuelle du cluster sont ensuite transmises uniquement lorsque deux conditions sont satisfaites : la valeur actuelle de l'attribut détecté est supérieure au seuil strict, et la valeur actuelle de l'attribut détecté diffère de SV d'une quantité égale ou supérieure au seuil souple. A chaque transmission des données, la valeur détectée est réglée pour qu'elle soit égale à la valeur actuelle de l'attribut détecté. Ainsi, le seuil strict tente de réduire le nombre de transmissions en permettant aux nœuds de transmettre uniquement lorsque l'attribut détecté se trouve dans la plage souhaitée. Le seuil souple réduit encore le nombre de transmissions en éliminant toutes les transmissions pouvant se produire lorsqu'il n'y a que peu ou pas de changement dans l'attribut détecté une fois le seuil strict atteint.

4.2 Protocoles d'agrégation distribuée

4.2.1 COUGAR

COUGAR [107] est un protocole centré données qui définit le réseau comme étant un système de base de données qui permet de fournir une plateforme distribuée pour la collecte des données pour les capteurs individuels, en utilisant des requêtes déclaratives définies par l'utilisateur afin d'abstraire le traitement des requêtes des fonctions de la couche réseau comme la sélection des capteurs adaptés.

COUGAR utilise le processus d'agrégation des données pour économiser l'énergie. Pour cela, un nœud leader est sélectionné par un algorithme de sélection distribué. Le nœud est chargé d'agréger et de transmettre les données vers la station de base qui génère un plan de requête permettant de commander les informations liées au flux de données, ainsi qu'au calcul de la requête entrante avant de les transmettre vers les nœuds correspondants. Le plan de requête définie la méthode de sélection du nœud leader qui obtient toutes les lectures, calcule la moyenne et l'envoie à la station de base. Pendant que le nœud leader agrège les données d'une manière optimale, les nœuds capteurs peuvent effectuer une agrégation partielle en se basant sur les données reçues des nœuds voisins

COUGAR permet de fournir une solution indépendante de la couche réseau pour l'interrogation des capteurs. Cependant, il peut présenter des inconvénients. En effet, lorsqu'une couche de requête supplémentaire est ajoutée au capteur, des frais supplémentaires liés au stockage ainsi qu'à la consommation d'énergie sont ajoutés aux capteurs. Aussi, il est important de synchroniser les nœuds lorsque les données sont calculées à partir de nombreux nœuds, ce qui signifie que le nœud qui relaie ne peut pas envoyer les données au nœud leader tant que tous les paquets de données ne sont pas collectés à partir de diverses sources.

4.2.2 TAG

TAG (Tiny AGgregation) [108] est un protocole d'agrégation dédié principalement pour les réseaux adhoc. TAG offre une interface déclarative simple permettant la collecte et l'agrégation des données, inspirée des fonctions de sélection et d'agrégation utilisées dans les langages de requête de bases de données. Les requêtes d'agrégation sont exécutées intelligemment d'une manière permettant d'économiser l'énergie et le temps. Les agrégats du réseau sont traités par le calcul continu des données lorsqu'elles traversent les capteurs, tout en supprimant les données non pertinentes. Les données pertinentes sont combinées dans des enregistrements.

Dans le protocole TAG, un arbre couvrant est créé dans le but d'avoir des transmissions de messages économes en énergie. Le nœud racine envoie les messages en premier avec le niveau 0 et son identificateur. Tous les nœuds écoutant le message incrémentent le niveau,

ajoutent leurs identifiants et rediffusent. La source du message est sélectionnée comme parent.

Les utilisateurs dans TAG émettent des requêtes d'agrégation à partir de la station de base. Les requêtes sont alors diffusées dans le réseau par des opérateurs qui les implémentent en utilisant un protocole de mise en réseau adhoc existant. Les données sont acheminées par les capteurs vers l'utilisateur à travers une arborescence de routage dont la racine correspond à la station de base. Au fur et à mesure que les données remontent dans cette arborescence, elles sont agrégées selon une fonction d'agrégation basée sur des valeurs définies dans les requêtes.

4.2.3 TiNA

TiNA [109] est un protocole d'agrégation des données qui vise à réduire la consommation énergétique et le temps de calcul en augmentant la qualité des données. Pour cela, TiNA exploite la corrélation temporelle dans les séquences de lecture des capteurs, permettant ainsi de réduire la consommation d'énergie de 60%, et prolongeant par conséquent la durée de vie du réseau de 300%. Dans TiNA, l'agrégation est effectuée au niveau des nœuds internes lorsque les informations sont transmises dans l'arbre de routage. Les lectures sont transmises dans l'arbre une fois par période, comme défini dans la requête du réseau. Les données sont envoyées uniquement en cas de changement significatif de leur valeur. Les paquets des nœuds enfants sont synchronisés et l'agrégat est envoyé selon une condition donnée. La clause WHERE est utilisée afin de filtrer les données qui ne remplissent pas la condition spécifiée. Aussi, les données dont les valeurs sont situées dans les plages de tolérance spécifiées sont filtrées, nécessitant par conséquent des besoins en mémoire plus élevés dans chaque nœud. Les résultats intermédiaires des nœuds enfants sont stockés afin d'être utilisés et comparés au prochain envoi des données. S'ils ne sont pas dans la plage spécifiée, les données sont alors transmises ce qui réduit significativement le nombre de messages.

4.2.4 DQEB

La majorité des protocoles d'agrégation supposent que l'énergie ne change pas tout au long du fonctionnement. Cependant, bien que les nœuds possèdent initialement la même réserve énergétique, l'énergie des nœuds non feuilles diminue par rapport aux autres nœuds. Ceci est justifié par le fait que les nœuds non feuilles transmettent et à reçoivent plus de données que les nœuds feuilles. Ainsi, plus d'énergie est dépensée au niveau des nœuds non feuilles par rapport aux autres nœuds. DQEB (Dynamic query-tree Energy Balancing Protocol) [110] est un protocole d'agrégation des données qui vise à remédier à ce problème en proposant une approche équilibrée en énergie. L'approche permet une modification dynamique de la structure de l'arbre qui est basée sur l'énergie résiduelle des nœuds. Pour cela, les nœuds sont organisés en grappes avec des têtes de grappe. Un poids est attribué aux nœuds. Ce dernier augmente avec la diminution de la durée de vie ou de l'énergie. Lorsque l'énergie diminue, le

nœud est transformé en nœud feuille, et est déplacé vers le bas de l'arbre. Lorsque le poids d'un nœud non feuille atteint un seuil critique, les nœuds enfants et parents sont alternés. Les nœuds parents alternants pour tous les fils sont sélectionnés par le chef de grappe. Comme les nœuds avec moins d'énergie sont devenus des nœuds feuilles, leur durée de vie augmente du fait que leur rôle consiste seulement à envoyer des données. Dans le cas où un nœud meurt en raison d'une panne, le coût supplémentaire lié à la construction de l'arbre ne sera pas nécessaire, car l'arborescence reste connectée.

4.3 Protocoles d'agrégation hybride

4.3.1 HEEP

HEEP (Hybrid Energy Efficiency Protocol) [111] est un protocole qui vise à combiner les avantages des protocoles LEACH et PEGASIS. HEEP est basé sur une agrégation de cluster et applique le principe de l'approche PEGASIS à l'intérieur des clusters.

Le fonctionnement de HEEP consiste à construire une chaîne de nœuds dans un même cluster afin d'améliorer la dissipation énergétique. Les nœuds ne communiquent pas directement avec leur chef de groupe mais uniquement avec leurs voisins les plus proches. Chaque chef de groupe envoie les données collectées à la station de base à travers ses nœuds voisins, en se basant sur une technique à sauts multiples, ce qui limite les dépenses en énergie. L'agrégation des données est utilisée dans chaque nœud dans la chaîne pour réduire la quantité de données échangées entre les nœuds et leur chef de groupe, ce qui permet de préserver les réserves énergétiques des nœuds.

HEEP est basé sur l'utilisation de deux phases essentielles : la phase d'initialisation qui permet de former les chaines de clusters et d'élire les chefs de groupes, et la phase de transmission pendant laquelle les données collectées seront transmises. À chaque phase de transmission, tous les nœuds appartenant au même cluster auront comme tâche de détecter et collecter les données. La transmission des données à la station de base est déléguée au chef de groupe qui représente le nœud le plus puissant, ce qui permet de réduire la consommation d'énergie.

5. Mesures de performance de l'agrégation des données

Il existe des mesures de performance cruciales des algorithmes d'agrégation des données, qui sont étroitement liées aux domaines d'application :

5.1 Efficacité énergétique

L'efficacité énergétique représente la métrique d'évaluation la plus importante dans le processus d'agrégation des données dans les réseaux de capteurs sans fil. Elle est définie comme étant le rapport entre le débit du réseau et la puissance totale utilisée qui représente la puissance de fonctionnement ainsi que la puissance de transmission. L'efficacité énergétique permet de mesurer l'énergie totale dépensée par le réseau lors de l'agrégation

des données. L'objectif principal des techniques d'agrégation des données consiste à maximiser l'efficacité énergétique de l'ensemble du réseau sans fil avec ou sans contrainte de puissance d'émission.

5.2 Durée de vie du réseau

Comme la consommation d'énergie, la durée de vie du réseau représente une métrique très importante dans l'évaluation des performances des RCSF. La durée de vie du réseau est étroitement liée à la consommation énergétique des nœuds ainsi que leur durée de vie. Plus la consommation d'énergie des nœuds du réseau diminue plus la durée de vie du réseau augmente. Le processus d'agrégation des données dans les RCSF permet de réduire considérablement la charge de travail globale du réseau et optimiser par conséquent la durée de vie du réseau.

5.3 Latence

Une autre métrique importante de l'agrégation des données dans les RCSF est la latence. En effet, le processus d'agrégation des données vise à recueillir les données essentielles des capteurs et de les rendre disponibles à la station de base avec une latence minimale des données. Pour cela, Il est important de développer des algorithmes d'agrégation des données permettant d'obtenir une latence minimale tout en garantissant une économie en énergie et une durée de vie du réseau optimale.

6. Protocole d'agrégation des données Big Data dans les réseaux de capteurs sans fil

L'agrégation de données est l'un des principaux défis des RCSF basés sur la technologie Big Data. En effet, la combinaison des volumes importants de données provenant de différentes sources permet d'éliminer la redondance et réduire par conséquent le nombre de ressources disponibles sur le réseau ainsi que leur consommation. L'agrégation des données est un sous-ensemble de fusion de données qui implique l'utilisation de techniques qui combinent et rassemblent des données provenant de sources multiples dans l'objectif de réaliser des associations plus efficaces et potentiellement plus précises.

Des stratégies sont proposées pour relever ce défi. Ces stratégies reposent principalement sur la corrélation entre l'agrégation des données, le Clustering et les enjeux de la consommation énergétique des données Big Data.

Dans ce qui suit, nous présentons les différentes stratégies d'agrégation des données dans les RCSF basés sur la technologie Big Data :

6.1 Agrégation de données compressive et distribuée dans les réseaux de capteurs sans fil à large échelle

Les auteurs de l'approche [112] ont proposé un algorithme distribué (Distributed Compressive Data Aggregation in large-scale Wireless Sensor Networks (DC)) basé sur la minimisation locale, dans l'objectif de construire dynamiquement un chemin de routage afin de réduire le trafic des données pour l'agrégation. L'algorithme proposé est basé sur l'échantillonnage par compression.

L'objectif principal de l'approche consiste à minimiser le trafic général dans le processus d'agrégation des données hybrides avec de faibles coûts généraux. Les auteurs supposent que l'agrégation des données s'effectue en cycles en programmant correctement le réseau. Ils supposent aussi qu'aucune erreur de transmission ne se produit lors de l'application du schéma de codage source. Par conséquent, le chemin de routage dans le processus de collecte des données formera un arbre d'agrégation enraciné à la station de base. De plus, un nœud est sélectionné comme agrégateur lorsque la taille des données qu'il relie est supérieure à un seuil M donné.

L'algorithme d'agrégation des données compressives proposé est divisé en deux étapes : la construction de l'arbre d'agrégation des données d'une manière distributive, et la réduction du trafic de données en ajustant le chemin de routage localement.

La construction de l'arbre d'agrégation des données est divisée en deux phases :

- La première phase de la construction consiste à calculer la distance la plus courte entre tous les nœuds et la station de base. Pour cela, un ensemble de nœuds non visités est créé et dans lequel tous les nœuds, à l'exception de la station de base, sont marqués comme non visités. Ensuite, pour chaque nœud donné et une fois que tous ses voisins sont pris en compte, il est marqué comme visité puis supprimé de tous les nœuds non visités et sa distance provisoire est enregistrée comme étant la distance la plus courte. Le processus est répété jusqu'à ce que l'ensemble des nœuds non visités soit vide ou que la plus petite distance provisoire entre tous les nœuds de l'ensemble non visité soit égale à l'infini.

- La deuxième phase de la construction consiste à trouver les bords de l'arbre du chemin le plus court :

- Une fois que la distance la plus courte soit trouvée pour tous les nœuds, un parent P_x est attribué à chaque sommet x différent de la station de base. Une fois que le parent de tous les nœuds différents de la station de base est déterminé, l'arbre du chemin le plus court est composé des bords entre tous les nœuds et leurs parents.

Une fois l'arbre d'agrégation construit, un algorithme de minimisation locale distribué (LM) est utilisé pour calculer si le passage à un voisin différent peut réduire le trafic des données. Chaque nœud effectue les étapes suivantes :

- Chaque nœud x collecte la taille des données reçues et l'identifiant des parents de ses voisins à deux sauts en échangeant des messages $INFO$ avec ses voisins.

- Pour chaque voisin non enfant avec une distance égale ou inférieure à la station de base, x mesure la modification du trafic des données locales si le nœud x change son parent à N_i. Le nouveau trafic local est calculé en supposant que le nœud x est redirigé vers N_i. Dans le cas où N_i et x ont le même parent, le trafic parental n'est compté qu'une seule fois. Si le nouveau trafic est inférieur au trafic d'origine, x enregistre la taille réduite du trafic et l'identificateur du voisin.

- Une fois toutes les mesures réalisées, si le nœud x n'est pas verrouillé, il sélectionne son voisin et envoie un message $LOCK$ pour l'empêcher de mettre à jour son parent. Sinon, le nœud x reporte son action jusqu'à ce qu'il soit déverrouillé par un message $UPDATE$. Une fois que le message $LOCK$ est reconnu par le nœud y (c'est-à-dire que y est correctement verrouillé), le nœud x met à jour son parent en y, puis diffuse un message $UPDATE$ pour informer ses voisins afin qu'ils puissent exécuter cet algorithme avec les informations mises à jour. Le message $UPDATE$ déverrouille également y afin qu'il puisse continuer son calcul et sa mise à jour. Si x ne reçoit pas d'accusé de réception de y après un certain temps, il attend un délai aléatoire et envoie à nouveau un message $LOCK$.

L'approche proposée est expérimentée et les résultats de la simulation démontrent que la structure arborescente a un impact significatif sur l'efficacité de l'agrégation compressive des données. De plus, les résultats montrent que la solution proposée génère des coûts généraux bien inférieurs à la solution presque optimale, ce qui la rend plus adaptée aux RCSF.

6.2 Agrégation des données de capteurs dans une infrastructure Big Data multicouches

Le travail proposé dans [113] vise à agréger les données dans les RCSF basés sur la technologie Big Data. Pour cela, les auteurs ont proposé une infrastructure d'agrégation des données volumineuses multicouches (Sensor data aggregation in a multi-layer Big Data framework), ainsi qu'un protocole d'agrégation des données dynamiques appelé PDDA (Priority-based Dynamic Data Aggregation Protocol) basé sur les priorités, qui est implémenté sur les nœuds capteurs responsables de la collecte des données.

Les auteurs ont proposé une infrastructure d'agrégation des données à trois couches, où les opérations d'agrégation des données sont effectuées sur des stations de base (BS) connectées à Internet et à de grands serveurs de données. Ensuite, les auteurs ont présenté un schéma d'agrégation des données dynamiques PDDA basé sur les priorités des réseaux de capteurs car les capteurs collectent une grande quantité de données redondantes.

Le schéma PDDA proposé est une approche hybride qui utilise des approches en cluster basées sur des arbres selon les types d'application. L'approche en cluster est utilisée pour agréger les données urgentes en temps réel, ce qui permet de réduire le délai de transmission des données de bout en bout, du fait que ces données ont la priorité la plus élevée et doivent être transmises avec un délai de transmission de données minimal. L'approche en arbre est utilisée pour les applications en temps non réel. Les topologies basées sur des clusters et des arborescences sélectionnent certains nœuds comme actifs, qui fournissent toute la couverture réseau. Ainsi, l'approche PDDA proposée atteint l'efficacité énergétique et réduit le temps de traitement des données et les frais généraux au niveau du serveur Big Data.

L'infrastructure d'agrégation des données proposée comprend trois couches :

Couche 1 - Agrégation des données au niveau des capteurs.

Couche 2 - Agrégation des données au niveau de la station de base.

Couche 3 - Agrégation des données au niveau du serveur Big Data ou du serveur NoSQL - serveur de couche 3.

Le système PDDA proposé fournit une priorité d'agrégation des données en fonction du type des données capturées. Par exemple, les applications critiques en temps réel auront plus de priorité que les applications qui ne sont pas en temps réel.

Les capteurs de la couche 1 transmettent les données aux couches supérieures à travers la station de base ou les nœuds passerelles. Cependant, pour obtenir une agrégation efficace des données, les réseaux de capteurs utilisés pour les différents types d'applications sont conçus pour avoir une topologie de réseau différente. Par exemple, pour les applications critiques ou en temps réel, l'agrégation basée sur le Clustering est utilisée lorsque les capteurs transmettent des données à la station de base par le biais de leur chef de cluster. Si ce dernier est loin de la station de base, il consommera plus d'énergie, mais il transmettra les données en utilisant un nombre minimum de sauts ce qui permet de réduire la latence des données.

Dans l'approche proposée, un certain nombre de nœuds sont sélectionnés comme nœuds actifs qui couvrent toute la zone du réseau [114]. Ensuite, les clusters sont formés et un nœud actif est sélectionné comme chef de cluster pour chaque cluster. Les nœuds actifs détectent et transmettent les données aux chefs des clusters, tandis que les chefs des clusters filtrent ou rejettent les données critiques redondantes et les transmettent au nœud passerelle afin

qu'il puisse les transmettre à la base de données centrale ou à la station de contrôle avec un délai minimum [115].

D'un autre côté, pour les applications qui ne sont pas en temps réel, l'atteinte de l'efficacité énergétique est plus importante. En conséquent, les capteurs forment une topologie arborescente et transmettent leurs données au nœud passerelle ou à la station de base à travers le chemin le plus court. Initialement, les nœuds seront identifiés comme étant situés à différents niveaux du réseau, en fonction du nombre de sauts pour le nœud passerelle. Ensuite, le chemin le plus court du nœud passerelle vers les nœuds actifs sera créé en utilisant la méthode présentée dans [116]. Les nœuds actifs au niveau le plus bas détecteront l'événement d'intérêt et transmettront aux nœuds actifs au niveau supérieur. Les nœuds parents de cette arborescence effectuent toujours l'agrégation des données à l'aide de différentes fonctions d'agrégation telles que MAX, MIN, MEAN, MEDIAN, SUM et les résultats sont envoyés aux nœuds actifs du niveau supérieur jusqu'à ce que les données atteignent la passerelle. Cette approche devrait entraîner une dissipation de puissance bien répartie sur tous les nœuds actifs et également une consommation énergétique inférieure du réseau, même si le nombre de sauts du nœud capteur vers la station de base est plus élevé. Cela est dû à la consommation d'énergie dans les nœuds capteurs qui est directement proportionnelle à la distance parcourue par un paquet de données d'un nœud à un autre [117]. Cependant, cette approche peut avoir un délai de transmission de données plus long car les données traversent plusieurs niveaux et passent du temps dans chaque nœud de cette hiérarchie pour leur traitement. Ainsi, l'approche PDDA proposée offre un compromis entre l'efficacité énergétique et le délai de transmission des données.

Généralement, les nœuds capteurs sont déployés pour une application spécifique et forment leur topologie de réseau en fonction du type de l'application. Cependant, ces capteurs peuvent être réutilisés dans d'autres applications et leur topologie peut changer si l'application change. Les capteurs vérifient le changement d'application en observant les paquets de données qu'ils détectent et transmettent car les paquets de données contiennent les types d'application, ce qui aide les couches 2 et 3 à traiter et à stocker les données aux emplacements appropriés.

La simulation de l'approche montre que le schéma PDDA proposé consomme moins d'énergie par rapport aux approches traditionnelles d'agrégation des données basées sur le Clustering. En conséquent, la durée de vie du réseau du schéma proposé devrait être plus longue que celle des approches en cluster et en arbre. Les résultats démontrent que la transmission des données de l'approche PDDA est inférieure à celle de l'approche basée sur le Clustering.

L'approche d'agrégation des données PDDA proposée ne sélectionne que quelques nœuds actifs qui couvrent l'ensemble du réseau, ce qui réduit la consommation totale du réseau. En addition, l'implication de moins de nœuds actifs dans le traitement et la transmission des données réduit considérablement le temps de transmission des données de bout en bout.

6.3 Agrégation des données basée sur la compression d'ondelettes de levage dans les réseaux de capteurs sans fil basés Big Data

Dans le travail proposé dans [118], les auteurs avaient pour objectif l'élimination efficace en énergie et l'agrégation des données redondantes dans le but de récupérer les données originales. Pour équilibrer la charge d'agrégation du RCSF à grande échelle, les auteurs ont proposé un nouvel algorithme de Clustering dynamique économe en énergie utilisant la corrélation spatiale (CDCC), qui fournit une agrégation distribuée des données dans chaque cluster. Les auteurs ont utilisé une approche d'agrégation des données rapide et distribuée basée sur l'ondelette de levage pour réduire la quantité des données brutes. De plus, l'approche offre un recouvrement élevé des données brutes.

Les auteurs ont proposé un algorithme d'agrégation des données basé sur une ondelette distribuée à grande vitesse pour agréger les données collectées pour les RCSF à large échelle (Lifting Wavelet compression based data aggregation in Big Data wireless sensor networks), permettant de réduire efficacement la quantité des données transmises et récupérer les données originales avec une grande précision. L'originalité de l'approche proposée réside dans les points suivants :

- Le réseau peut être groupé dynamiquement en exploitant la corrélation spatiale et les besoins des utilisateurs plutôt que l'agrégation complète dans le réseau.
- La redondance spatiale et temporelle peut être réduite par l'algorithme d'agrégation des données proposé.
- L'approche proposée permet d'atteindre un bon équilibre entre la précision de la récupération des données et la consommation d'énergie. La corrélation spatiale est utilisée pour déterminer les membres du cluster. Une fois que les données détectées dans un cluster possèdent une forte corrélation, un membre du cluster peut représenter les nœuds dans sa zone voisine avec ses propres données, et le chef du cluster agrège et envoie uniquement les données de ses membres à la station de base plutôt que toutes les données reçues. De plus, l'approche permet à certains membres du cluster de quitter un cluster si des données anormales sont détectées. Ainsi, l'énergie sera économisée par l'élimination de certains nœuds et la taille des données sera réduite en supprimant la redondance.

Les principales contributions de l'approche proposée sont les suivantes :

- Les auteurs proposent un algorithme de Clustering dynamique basé sur la corrélation spatiale des données pour éliminer la redondance, ce qui peut réduire la taille des données et prolonger ainsi efficacement la durée de vie du réseau.

- L'algorithme de Clustering dynamique est analysé en termes de complexité de temps et d'espace. Le nombre optimal de clusters est dérivé en fonction de la consommation énergétique associée.

- Les auteurs s'appuient sur l'utilisation d'une technique de compression d'ondelettes rapide et distribuée pour agréger les données à chaque chef de cluster et envoyer les données à la station de base. Avant la transmission, les coefficients d'ondelettes sont compressés et codés pour réduire la quantité de coefficients, ce qui peut garantir la précision de la récupération des données en n'occupant qu'une petite quantité d'espace de stockage.

1) Modèle de Clustering basé sur la corrélation spatiale (CDSC)

Le réseau est modélisé en un graphe non orienté G = (V, E), où *V* est l'ensemble des nœuds capteurs et *E* est l'ensemble des arêtes composé de toutes les liaisons du RCSF. Les nœuds avantageux en termes d'énergie et de localisation géographique sont élus en tant que chefs de clusters. Les chefs de clusters ont pour tâche non seulement la collecte des données, mais également leur transmission à la station de base, ce qui fait de ces derniers les nœuds les plus dynamiques du réseau. Il est à noter que certains nœuds d'une même région peuvent représenter toutes les métriques reçues de leurs nœuds voisins en raison des fortes corrélations spatiales entre les données. Tous les nœuds capteurs sont divisés en deux catégories : les membres candidats et les nœuds normaux, en fonction de leur niveau d'énergie résiduelle avant le regroupement. Le point critique du modèle de Clustering proposé est la sélection des chefs de clusters ainsi que les membres du cluster.

L'algorithme de Clustering proposé pour chaque sous-région est divisé en plusieurs étapes :

- Les nœuds sont divisés en deux groupes : CandSet qui représente le groupe des nœuds candidats pour devenir des chefs de clusters, et OrdSet qui représente le groupe des nœuds restants. La division se fait selon la formule suivante :

$$p = {E_{cur}}/{E_{total}} \quad (1)$$

Où : p représente l'état de l'énergie restante, E_{cur} représente l'état de l'énergie résiduelle d'un nœud donné, et E_{total} indique la capacité énergétique totale.

- Un nœud du groupe *Set* sera sélectionné comme chef de cluster, à condition qu'il soit le plus proche du centre régional. Dans chaque sous-région, le nœud capteur *i* dans CandSet trouve la distance géographique la plus courte de la sous-région centrale. Le nœud avec les diffusions distantes les plus courtes devient alors le chef de cluster. Ensuite, les nœuds CandSet restants deviennent membres du cluster.

- La dernière étape consiste à trouver les nœuds normaux proches du cluster. Pour cela, pour chaque nœud, les formules suivantes sont vérifiées pour observer si elles répondent à toutes les contraintes :

$$x_{min} \geq x_{actual} \times R_s \quad (2)$$

$$x_{max} \leq x_{actual} + x_{actual} \times (1 - R_s) \quad (3)$$

Où : x_{actual} représente les données sensorielles d'un membre existant du cluster au temps actuel. x_{min} et x_{max} représentent les valeurs sensorielles minimale et maximale des nœuds normaux adjacents d'un membre de cluster. R_s est le rapport de similitude réglable dans la station de base.

Dans ce cas, le nœud est représenté par le membre du cluster correspondant et ne rejoint pas le cluster. Sinon, le nœud sera ajouté en tant que nouveau membre. Les nœuds normaux restants seront supprimés de ce cluster.

2) Compression par ondelette de levage distribuée

Les données sont compressées sur la base de l'algorithme de Clustering de corrélation de données proposé (CDCC) qui fournit non seulement un calcul rapide, mais également une sauvegarde substantielle de l'espace mémoire.

Les données reçues par le chef de cluster sont stockées sous forme de matrice bidimensionnelle. La matrice peut être décomposée en quatre sous-bandes par des ondelettes de levage en ligne ou en colonne. Les quatre sous-bandes sont : une sous-bande basse fréquence et trois sous-bandes à haute fréquence. Les informations les plus utiles sont concentrées dans la sous-bande des fréquences basses. Par conséquent, certains coefficients de fréquence qui contiennent moins d'informations peuvent être éliminés.

Dans ce qui suit, les principales procédures de la méthode de compression des données en ondelettes proposée :

Tout d'abord, les données originales effectuent une transformation en ondelettes de premier niveau. Le processus de transformation en ondelettes est divisé en trois étapes :

Processus de fractionnement : pour chaque ligne de la matrice des données, les données à un intervalle de temps donné sont divisées en une séquence paire et une séquence impaire.

Processus de prédiction : un opérateur de prédiction est exécuté sur les signaux à des intervalles de temps pour prédire le signal de données pair.

Processus de mise à jour : consiste à exécuter un nouvel opérateur de mise à jour, qui met à jour le signal d'origine vers le nouveau signal correspondant.

Au cours du processus de transformation en ondelettes de ligne, les données originales sont remplacées par un coefficient à fréquence basse et un coefficient à haute fréquence. Plus la corrélation temporelle des données est élevée plus la valeur du coefficient élevé est faible. De même, dans le processus de transformation en ondelettes de colonnes, une plus grande corrélation spatiale implique un coefficient à haute fréquence plus bas. Lorsque la transformation en ondelettes de premier niveau est terminée pour toutes les lignes et les colonnes, les données d'origine sont converties en une partie à basse fréquence et trois parties à haute fréquence.

L'algorithme proposé est comparé à d'autres approches et les résultats de simulation montrent que l'algorithme CDCC est supérieur en termes d'économie d'énergie. Bien que la compression de l'ondelette de levage soit limitée par son taux de compression, sa précision de récupération peut être supérieure à 98% si les paramètres sont ajustés de manière appropriée. En effet, les résultats expérimentaux démontrent que la méthode de consolidation basée sur le Clustering de corrélation de données (CDSC) proposée pour l'agrégation des données surpasse les autres méthodes en termes de prolongation de la durée de vie du réseau et de réduction des quantités de données transmises. L'algorithme de Clustering dynamique proposé et la technique d'agrégation des données compressives basée sur les ondelettes peuvent atteindre de meilleures performances, par exemple des données de précision de récupération plus importantes et une économie en énergie considérable.

6.4 Agrégation des données avec analyse des composants principaux dans les réseaux de capteurs sans fil basés Big Data

Dans le travail proposé dans [119], un algorithme d'agrégation des données nommé PCA (Data aggregation with principal component analysis in Big Data wireless sensor networks) est proposé, dans l'objectif de transmettre efficacement les données volumineuses détectées avec une faible latence tout en éliminant la redondance des données dans les chefs de clusters, pour minimiser la complexité des données transmises.

Les auteurs de l'article ont proposé un algorithme de Clustering distribué basé sur la similitude pour placer les nœuds ayant une forte similitude dans le même cluster.

1) Modèle du système énergétique

Un grand nombre de capteurs dans les réseaux de capteurs sans fil produiront de grandes quantités de données. Ces données détectées sont collectées en s'appuyant sur la technologie Big Data [120]. Lorsque la station de base souhaite consommer le moins d'énergie pour accueillir toutes les données, la distribution des données Big Data doit être comprise en premier. Il existe deux manières d'obtenir la distribution des données. La première est que le nœud capteur traite et récupère les données associées, puis transmet les données traitées vers la station de base. La deuxième technique est que pendant la transmission des données

Big Data, tous les nœuds transmettent leurs données brutes à la station de base qui les traite et établie le meilleur choix pour le réseau. L'approche proposée adopte la deuxième option.

Dans le modèle proposé, l'énergie consommée pour transmettre *i* octets de données dans chaque nœud est donnée par la formule suivante :

$$E_T(l,d) = \begin{cases} l * E_{elec} + l * E_{fs} * d^2, si\ d < d_0 \\ Sinon\ l * E_{elec} + l * E_{amp} * d^4 \end{cases} \quad (4)$$

Où: d est la distance entre l'émetteur et le récepteur et d_0 est le rayon de communication du nœud, et l représente le nombre d'octets de données envoyés par l'émetteur au récepteur.

E_{elec} est la consommation d'énergie de l'émetteur et du récepteur.

$E_{fs} * d_2$ est la consommation d'énergie de l'amplificateur dans la plage de communication.

$E_{amp} * d_4$ est la consommation d'énergie de l'amplificateur au-delà de la plage de communication.

Le réseau est divisé en clusters en fonction de la similitude entre les nœuds. L'énergie consommée par un chef de cluster en agrégeant 1 octet de données de ses membres est donnée par la formule suivante :

$$E_p = k * l * E_{pr} \quad (5)$$

Où : E_p représente la consommation d'énergie du processus d'agrégation des données dans le cluster; k est le nombre de nœuds dans le cluster, et E_{pr} est l'énergie consommée par le CH en agrégeant les données d'un octet de ses membres de cluster.

L'énergie consommée dans l'ensemble du cluster est donnée par la formule suivante :

$$E_{Cluster} = E_T(l,d) + E_R(l,d) + E_p \quad (6)$$

Où $E_R\ (l,d)$ représente la consommation d'énergie pendant la réception des octets de données pour chaque nœud et qui est donnée par la formule suivante :

$$E_R(l,d) = l * E_{elec} \quad (7)$$

2) Algorithme de Clustering basé sur la similarité des données

Les auteurs ont proposé un algorithme de regroupement pour l'agrégation des données basé sur la similitude des données. La similitude des données est définie selon deux aspects différents, à savoir l'ampleur de la similitude des données et la corrélation des données.

Sur la base des valeurs de similitude, un algorithme de Clustering adapté à l'agrégation des données basé sur l'algorithme PCA est proposé. Ce dernier assure le partitionnement des nœuds capteurs dans les clusters avec un haut niveau de similitude, et permet de sélectionner un nœud approprié pour devenir un chef de cluster.

L'algorithme proposé est détaillé comme suit :

- Chaque nœud capteur calcule la similitude avec ses nœuds voisins. Si deux nœuds u et v satisfont un seuil de similitude donné ξ, ils établiront un arc uv. Ainsi, tous les nœuds formeront un graphe g. Les nœuds seront ensuite triés selon leur degré dans l'ensemble des nœuds S et le nœud avec le degré le plus large sera choisi comme chef de cluster. Pour réduire la consommation d'énergie dans chaque cluster, le nombre de nœuds est réduit à $k-1$.

  Le chef de cluster sélectionne $k-1$ nœuds présentant une forte similitude parmi ses nœuds voisins et les supprime de l'ensemble S des nœuds. L'opération est répétée jusqu'à ce que le degré le plus large de nœuds dans l'ensemble S soit inférieur à $k-1$.

  Dans le cas où le nombre de nœuds restants dans S est petit, l'opération de calcul de similitude n'est pas déclenchée, mais le nombre de nœuds dans le cluster est réduit de sorte que le nœud restant devient chef de cluster.

- Une fois que les clusters sont devenus stables, les membres effectuent une rotation du chef de cluster.

- Si la station de base constate que les données inter-cluster ont une différence supérieure à un seuil donné ou si la moitié des nœuds ne satisfont pas les valeurs de similitude, elle décide de réactiver l'algorithme de Clustering.

3) Agrégation des données basée sur PCA (Principal Component Analysis)

L'algorithme de compression de données PCA est un algorithme hautement recommandé pour les capteurs ayant des capacités limitées. Il permet de réduire le degré de dimension des ensembles de données tout en conservant leurs caractéristiques.

L'algorithme proposé est détaillé comme suit :

- Le réseau est divisé en plusieurs clusters selon la méthode de Clustering proposée.
- Le chef de cluster collecte les données de ses membres et les met dans une matrice d'observation x.
- A partir de la matrice d'observation x, la matrice de covariance C est calculée.
- les valeurs propres et les vecteurs propres correspondants de la matrice C sont calculés ;
- Les valeurs propres sont classées pour obtenir la plus grande valeur ;

- les vecteurs propres correspondant à la plus grande valeur propre sont sélectionnés pour former la matrice de transformation P;
- la matrice de projection est calculée à partir de la matrice de transformation P;
- La matrice de projection calculée est envoyée vers la station de base.

L'algorithme proposé dans cette approche est évalué et les résultats expérimentaux ont démontré son efficacité en termes de réduction de la quantité des données transmises et la consommation énergétique du réseau.

6.5 Mécanisme évolutif préservant la confidentialité pour l'agrégation des données Big Data

Les auteurs dans [121] ont proposé un schéma d'agrégation des données confidentiel et évolutif (Scalable privacy-preserving Big Data aggregation mechanism (SCA-PBDA)) pour les données Big Data afin de répondre aux exigences de confidentialité et d'efficacité de transmission.

Le schéma proposé est basé sur une structure de topologie en gradient qui permet aux nœuds capteurs d'être divisés en clusters formés d'un chef de cluster CH, d'un nombre égal de membres (CMS) et de chefs de cluster auxiliaires ($aCHs$). Le Clustering réseau permet à la configuration préservant la confidentialité et aux techniques d'agrégation des données inter-cluster d'exécuter l'agrégation des données inter-cluster.

Les travaux proposés reposent principalement sur deux points :

- La consommation énergétique du nœud est utilisée pour déterminer la topologie du réseau sur la base d'une méthode de regroupement de réseaux égaux basée sur un gradient. La méthode de Clustering permet aux nœuds identiques de prendre en charge la configuration uniforme de préservation de la confidentialité et l'agrégation des données inter-cluster.

- La configuration des données préservant la confidentialité et l'agrégation évolutive des données intra et inter cluster est assurée par une méthode d'agrégation des données préservant la confidentialité évolutive.

1) Méthode de Clustering

La méthode de Clustering proposée est basée sur un établissement de gradient. Pour cela, un message d'établissement de gradient GE est diffusé par la station de base, dont la valeur de champ de gradient est 0. Le message est envoyé aux nœuds à travers lesquels la distance par rapport à la station de base peut être déterminée. Pour cela, les nœuds et après avoir obtenu leur valeur de gradient 1 en fonction du premier GE reçu, ajoutent la valeur au champ de comptage de sauts de GE et la mettent à jour. Ensuite, les capteurs diffusent le GE mis à jour

après un délai donné. La procédure de diffusion et de mise à jour de GE est répétée jusqu'à ce que tous les nœuds capteurs aient obtenu leur valeur de gradient.

Les chefs de chaque cluster sont élus à partir des nœuds capteurs et les aCHs seront alloués en fonction de la consommation d'énergie des CHs, ce qui engendre une corrélation négative avec leurs valeurs de gradient.

La formule suivante représente la consommation totale d'énergie d'un chef de cluster i avec une valeur de gradient i:

$$EC_{CHi} = EC_{Receiving} + EC_{Aggregating} + EC_{Transmitting} \quad (8)$$

Où $EC_{Receiving}$ représente la consommation d'énergie des données reçues des CM et d'autres CH.

$EC_{Aggregating}$ est la consommation d'énergie résultant de l'agrégation des données.

$EC_{Transmitting}$ représente la consommation d'énergie résultant de la transmission des données agrégées.

Pour éviter le partitionnement du réseau et les interruptions de communication provoquées par l'épuisement énergétique des relais CH les plus proches de la station de base, leur consommation d'énergie est partagée en allouant de manière adaptative aCH aux clusters. L'agrégation des données préservant la confidentialité inter-cluster est établie via un cluster identique composé de CH et de CM.

La station de base poursuit la phase d'initialisation en diffusant le message GE pour établir le gradient du réseau. Ensuite, les nœuds capteurs, avec une valeur de gradient i, se choisissent en tant que CH avec une probabilité calculée. D'autres nœuds capteurs envoient une demande pour rejoindre le cluster en tant que CM ou aCH.

2) Agrégation des données intra-cluster préservant la confidentialité

Avant d'effectuer l'agrégation des données, les positions préservant la confidentialité sont déterminées. Premièrement, la station de base détermine la position de la valeur vraie globale $(GTPS)$. La position est utilisée pour marquer les vraies valeurs des données du capteur. Chaque nœud génère un ensemble d'index de données I pour les données des capteurs pour indiquer que les données des capteurs préservant la confidentialité sont composées des vraies valeurs des données des capteurs et des valeurs de remplissage de camouflage qui garantissent la confidentialité des vraies valeurs des données des capteurs. Plus la valeur I est grande, plus l'espace créé pour remplir plus de valeurs de camouflage des nœuds capteurs est grand. Cependant, cela peut créer une surcharge de communication supplémentaire.

Pour chaque nœud capteur, le récepteur attribue $NPPS$ $(Node\ Private\ Position\ Set)$ et $NTPS$ $(Node\ True\ Position\ Set)$. Dans le schéma proposé, la position de la valeur réelle des données des capteurs de chaque nœud est étiquetée à l'aide de $NTPS$. De plus, les positions des nœuds sont marquées par la station de base afin de placer la vraie valeur des données des capteurs et les valeurs de camouflage restreintes. Pour faciliter l'agrégation des données préservant la confidentialité entre les clusters, une méthode de remplissage de camouflage préservant la confidentialité est déployée dans chaque cluster du réseau. La vraie valeur des données des capteurs et les valeurs de camouflage restreintes et non restreintes sont placées dans les positions appropriées.

La transmission des données protégées vers la station de base est effectuée à travers des sauts multiples. Leur transmission entre les CM et les CH est effectuée à travers un saut unique.

Lorsque des données préservant la confidentialité sont reçues par le CH à partir de ses CM, il effectue l'agrégation de données MAX comme indiqué dans la formule suivante :

$$Data_{Aggregated} = \bigcup_{i=1....l\ j=1....CS} max(d_{ij}) \quad (9)$$

Où $Data_{Aggregated}$ représente les données confidentielles agrégées et d_{ij} représente la valeur des données de la $ji\grave{e}me$ position du nœud.

3) Agrégation des données inter-cluster préservant la confidentialité

Comme la composition du cluster et la méthode de remplissage de camouflage préservant la confidentialité sont identiques, la même technique d'agrégation des données peut être utilisée.

Dans l'agrégation préservant la confidentialité entre les clusters, les données agrégées préservant la confidentialité sont ré-agrégées au niveau des CHs relais pour obtenir des résultats globaux agrégés contenant la valeur maximale des données des capteurs de tous les capteurs selon la formule :

$$Data_{reAggregated} = \bigcup_{i=1....l\ j=1....NRP} max(d_{ij}) \quad (10)$$

Où $Data_{reAggregated}$ représente les données ré-agrégées préservant la confidentialité et NRP représente le nombre de paquets reçus des données d'agrégation préservant la confidentialité.

Il est à noter que lorsque l'agrégation des données préservant la confidentialité inter-cluster est effectuée, les CH effectuent seulement une réagrégation simple par eux-mêmes et reçoivent des données intra-cluster préservant la confidentialité sans avoir connaissance de leur contenu détaillé. Ainsi, la confidentialité des données des grands capteurs est garantie.

4) Recouvrement du résultat agrégé

La dernière opération d'agrégation des données est effectuée par le récepteur pendant la réception de tous les paquets des données agrégées de ses CH_{ls}. Ensuite, le récepteur analyse les positions d'index des données et les valeurs maximales sont conservées comme résultat de récupération :

$$Data_{global=} \bigcup_{i=1....l,j=1....N_{Cluster}} d_{ij} \quad (11)$$

Où $Data_{global}$ représente le paquet des données du capteur global préservant la confidentialité, et $N_{Cluster}$ représente le nombre total de clusters de CH_{ls}.

Pour vérifier les performances du schéma proposé en termes de préservation de la confidentialité, le schéma est comparé à travers la simulation aux mécanismes traditionnels d'agrégation des données préservant la confidentialité tels que CPDA (Conflict-free Periodic Data Aggregation) et KIPDA (K-indistinguishable Privacy Preserving Data Aggregation). De plus, la durée de vie du réseau du protocole proposé est simulée afin de vérifier et valider son efficacité.

Les résultats de simulation montrent que la complexité de calcul et les coûts de calcul de Sca-PBDA sont extrêmement inférieurs à ceux de CPDA et KIPDA. Ainsi, l'approche proposée répond aux exigences d'application de la complexité de calcul et de l'évolutivité.

6.6 Technique de fusion des données basée sur le Clustering pour l'analyse Big Data dans un système multi-capteurs sans fil

Les auteurs dans [122] ont proposé une nouvelle technique de fusion des données basée sur un algorithme hybride pour le Clustering et la sélection des membres du cluster dans un système multi-capteurs sans fil (A cluster-based data fusion technique to analyze Big Data in wireless multi-sensor system). De plus, les auteurs utilisent une technique de fusion des données pour partitionner et traiter les données collectées.

Les auteurs ont essentiellement basé leur travail sur les points suivants :

- Développer une architecture hiérarchique pour intégrer l'algorithme de Clustering à la technique de routage.

- Proposer une stratégie de routage optimisée pour permettre aux nœuds déployés d'atteindre facilement le réseau survivant.

1) Architecture du réseau

L'architecture du réseau est basée sur quatre méthodes : le déploiement des nœuds, la formation des clusters, les critères de sélection des nœuds membres et la stratégie de routage.

Dans l'architecture proposée, les nœuds capteurs sont chargés de détecter les données et de les transmettre aux chefs de clusters qui effectuent les calculs et la communication. Ensuite, le cluster permet de transmettre les données traitées à la station de base en utilisant la technique de routage. Un autre rôle des chefs de clusters est de trouver un chemin économe en énergie vers la station de base.

Dans l'organisation du réseau, les nœuds peuvent être déployés de manière aléatoire ou uniforme et commencer à diffuser le paquet de contrôle vers leurs nœuds voisins pour montrer leur existence. Lorsque les nœuds voisins reçoivent le message diffusé, ils configurent leur table voisine pour calculer principalement la qualité de la liaison pour les futures décisions.

Les nœuds sont déployés de manière hiérarchique et les chefs de clusters transmettent les données reçues vers leurs chefs de clusters voisins qui sont près de la station de base. Une rotation des chefs de clusters est effectuée après un intervalle spécifique pour économiser l'énergie des nœuds.

Le modèle du système est évalué par le calcul de la puissance de transmission :

$$Y_{ij} = \frac{W_i}{N_{\|i,j\|^{\propto}}} \tag{12}$$

Où : W_i représente la puissance d'un nœud i, $\|i,j\|^{\propto}\|$est la distance euclidienne entre les nœuds i, j et α représente l'affaiblissement sur le pas de transmission [123].

De plus, le débit des données entre la source et la destination est calculé en utilisant l'équation suivante :

$$C_D(s_i, d_i) = BLog_2\left(1 + Y_{s_i,d_i}\right) \tag{13}$$

Où, B représente la bande passante disponible. De plus, dans le scénario proposé, les auteurs ont utilisé la technique de décodage et retransmission présentée dans [124].

La consommation d'énergie est calculée par la somme globale des sauts de tous les chefs de clusters :

$$W = \sum_{n=1}^{k} W_n = \sum_{n=1}^{k} E_n \left(D_R^C + S_D\right) \tag{14}$$

Où E_n est la quantité d'énergie consommée par un nœud, D_R^C représente la relation entre tous les nœuds et la somme globale des sauts dans un réseau, S_D est la quantité totale des données collectées par un nœud en un cycle et W_n est l'énergie de consommation du nœud i par cycle.

2) La couche de Clustering

La technique proposée vise à minimiser la consommation énergétique lors de la formation du cluster, augmentant ainsi la durée de vie du réseau. Dans cette technique, le nœud voisin favorise un nœud à un autre pour la sélection des différentes positions, l'ID du paquet et le compteur de suffixe représentent les principaux facteurs dans la conception du cluster. De plus, les nœuds de la deuxième couche sélectionnent les nœuds entourés par un nombre maximal de nœuds pour devenir des nœuds de décision.

Les nœuds de la deuxième couche échangent les informations sur leur densité pour participer au calcul permettant d'être élus comme chefs de clusters. Ils mettent leur émetteur-récepteur sous tension grâce à la technique d'accès multiple par répartition dans le temps (TDMA).

Pour sélectionner les nœuds membres dans chaque cluster, le chef de cluster diffuse initialement un message de demande de jointure qui indique la disponibilité du chef de cluster. Ensuite, le nœud qui reçoit le message *Join Request* diffuse en retour un message *Join Accepte* pour devenir membre du cluster. Si plus d'une demande de jointure est reçue par les nœuds, la décision est prise en fonction de la charge sur le chef de cluster.

3) Modèle de fusion des données

Le modèle de fusion des données proposé dans [125] est modifié en fonction des exigences architecturales. La technique de fusion des données proposée est composée de cinq niveaux.

- *Niveau 0 :* Initialement, les données sont reçues par le chef de cluster. Ensuite, les données sont alignées et divisées en sous-blocs où chacun est traité séparément au niveau de chaque chef de cluster.
- *Niveau 1 :* les données sont affinées à chaque chef de cluster et différents types de données sont convertis en une structure cohérente (images, textes…).

- *Niveau 2 :* Une description contextuelle est fournie pour chaque sous-bloc sur la base des données environnementales.

- *Niveau 3 :* chaque chef de cluster identifie les menaces et les susceptibilités futures pour les opérations, en fonction des complexités de calcul et de l'algorithme conçu.

- *Niveau 4 :* Les performances de traitement sont surveillées en continu au niveau de chaque chef de cluster.

Le cycle de vie du modèle de fusion des données dans le système multi-capteurs est défini. Il se compose de quatre attributs interdépendants :

- Les données brutes sont collectées et organisées sous une forme significative.
- Les données indésirables sont supprimées.
- Les blocs de données sont fusionnés et analysés.
- Enfin, les données traitées sont transmises sur le réseau.

Le schéma proposé est implémenté à l'aide de Hadoop et d'itérations Java. Les résultats de la simulation montrent que la technique proposée est économe en énergie sous tous les scénarios proposés.

6.7 Une approche distribuée sans collision pour l'agrégation des données dans les réseaux de capteurs sans fil à large échelle

Les auteurs dans [126] ont proposé une nouvelle approche d'agrégation des données en série, appelée Spreading Aggregation (Spreading Aggregation: A distributed collision-free approach for data aggregation in large-scale wireless sensor networks (SA)). L'objectif de l'approche consiste à raccourcir le chemin de transmission des données et réduire par conséquent le nombre de communications. Pour cela, à chaque démarrage du processus d'agrégation, un nouveau chemin est créé, permettant de réduire la vulnérabilité aux pannes dans les liens et les nœuds, et permettant la gestion des changements dans la topologie du réseau. Aussi, l'approche proposée est localisée et repose uniquement sur la table de routage des voisins à un saut de chaque nœud dans la construction du chemin, ce qui la rend très évolutive. Un autre point important de l'approche est l'agrégation des données simultanée à la construction progressive du chemin ce qui permet de réduire la consommation énergétique des nœuds capteurs.

1) Description de l'approche

Au début de l'algorithme, tous les nœuds du réseau sont considérés comme non visités. Les nœuds non visités sont notés Ω. Les nœuds visités sont notés Γ et leur ensemble est initialement vide.

Au démarrage de l'agrégation en série, une boule roulante ou le paquet d'agrégation est placée par le processus de lancement de l'agrégation nommé APL sur l'une de ses limites. La boule roulante se déplace et visite le réseau en série. La boule roulante se déplace d'un nœud vers un autre et marque ces nœuds comme visités. Afin de sélectionner le prochain saut adéquat, chaque nœud doit connaitre l'état de ses voisins à savoir s'ils ont été visités ou non. Pour cela, tous les voisins d'un nœud écoutent la communication pour savoir si la boule roulante est livrée au saut suivant. Un mécanisme d'écoute à faible puissance (LPL) [127] [128] est utilisé pour permettre aux nœuds d'écouter un paquet même s'il ne leur est pas adressé (écoute inactive).

Lorsque la boule roulante ait parcouru tout le réseau, le dernier nœud du chemin envoie les données agrégées qui forment le résultat obtenu à l'APL en utilisant n'importe quelle stratégie de routage.

2) Mécanisme de fonctionnement

Avant de débuter l'algorithme, tous les nœuds doivent créer des listes de limite qui seront utilisées ultérieurement afin de détecter et supprimer les cycles. Le nœud APL lance une requête en vérifiant s'il s'agit ou non d'un nœud situé en extrémité.

Afin de trouver tous les cycles possibles du réseau, l'APL place une boule roulante sur l'une de ses extrémités et envoie un paquet nommé IBS (Initial-Boundary Scan) pour le marquer. Ensuite, l'APL commence le traitement des requêtes en tournant la boule roulante dans le sens inverse et en sélectionnant le premier voisin à un saut comme saut suivant. Tous les nœuds du réseau répètent la même opération. Les nœuds ont la possibilité de quitter le processus d'interrogation ou de rester pour assurer la connectivité réseau. Aussi, les nœuds vérifient pendant la réception s'ils possèdent une extrémité non marquée, afin d'éviter de boucler autour des trous de communication. Dans le cas où cela se produit, un cycle est détecté et le nœud possédant la boule roulante, qui est appelé le nœud portail, doit supprimer le cycle en coupant certains liens spécifiques avec ses voisins à un saut et en envoyant un paquet LC (Link-Cut) aux autres voisins pour qu'ils exécutent la même opération. Pour cela, le nœud portail envoi un paquet appelé DBS (Disjoint-Boundary Scan) pour marquer l'extrémité disjointe.

L'APL qui n'est pas une extrémité ne peut pas lancer la boule roulante sur un rayon $\frac{R}{2}$. Pour remédier à ce problème, le nœud place une boule roulante vide plus petite que la boule roulante utilisée et la lance dans le réseau. Cette boule procède de la même manière que la boule roulante ordinaire mais en étalant progressivement la région visitée. A chaque nouveau saut, la boule est agrandie. Une fois le rayon optimal de la boule roulante est atteint, le processus de traversée se poursuit sur l'extrémité de la région non visitée. L'algorithme se termine à un nœud lorsque ses voisins ont complété le processus de traversée.

L'approche proposée rencontre deux problèmes majeurs :

- Le premier problème concerne la manière d'assurer la connectivité des nœuds non visités lors de la traversée du réseau. En effet, l'agrégation en série échoue lorsque les nœuds non visités se marquent comme visités et ne participent pas par conséquent à la traversée. Pour remédier à ce problème un nouvel état qui est le nœud de liaison est introduit. Un nœud est considéré comme un nœud de liaison lorsqu'au moins deux de ses voisins à un saut ne sont pas en mesure communiquer sans son aide. Ainsi, et afin d'assurer l'exhaustivité de l'agrégation, les nœuds de liaison doivent rester impliqués dans la traversée.

- Le deuxième problème concerne le lancement de l'agrégation à partir d'un APL qui n'est pas une extrémité qui, par sa position, ne peut pas définir un roulement valide et ne peut donc pas lancer le processus d'agrégation. Pour remédier à ce problème, L'APL qui n'est pas une extrémité créé une boule roulante rétrécie qui sera centrée à l'emplacement de l'APL. Son rayon d est égal à la distance entre l'APL et son voisin le plus proche. Le seul nœud situé à l'intérieur de la boule roulante rétrécie est un nœud visité, et ne peut pas être un nœud de liaison car il s'agit d'un nœud qui n'est pas une extrémité. Dans certains cas, la distance entre l'APL et son voisin le plus proche peut dépasser le rayon optimal $\frac{R}{2}$ de la balle. Dans ce cas, l'APL doit ajuster la boule à sa forme optimale.

L'approche proposée est évaluée et les résultats expérimentaux ont démontré son évolutivité ainsi que son efficacité en termes de réduction de la consommation énergétique et du temps de réponse. Aussi, d'après la théorie, l'approche proposée visitait tous les nœuds connectés du réseau sans bouclage.

6.8 Une approche d'agrégation des données efficace pour les réseaux de capteurs sans fil à large échelle

Les auteurs dans [129] ont proposé une nouvelle approche d'agrégation des données efficace pour les réseaux de capteurs sans fil à large échelle (An Efficient Data Aggregation Approach for Large Scale Wireless Sensor Networks), qui considère le compromis entre l'efficacité énergétique, la tolérance aux pannes et le délai de bout en bout. Pour cela, l'approche est basée sur l'utilisation de dispositifs d'alimentation fixes dans les déploiements à large échelle. De plus, l'approche permet aux utilisateurs finaux de contrôler dynamiquement la méthode d'agrégation des données proposée par la surveillance statique des données redondantes en fonction du temps accordé à l'utilisateur.

L'approche d'agrégation proposée comprend trois réseaux : réseau de capteurs sans fil, Wi-Fi et Wi-Max. Les utilisateurs peuvent accéder à distance aux données collectées dans chaque cluster à travers un serveur central lié à une interface utilisateur commune.

1) Création de zone et sélection du nœud actif

A un temps donné, le réseau de capteurs sans fil est organisé en plusieurs zones dont chacune possède un ou plusieurs nœuds actifs et plusieurs nœuds alternatifs qui sont en mode veille. La sélection des nœuds actifs est réalisée de manière à ce que ces derniers doivent se trouver dans la plage de transmission de deux nœuds actifs voisins. Les nœuds forment le chemin le plus court pour transmettre des données au chef de cluster compatible Wi-Fi (CH) en se basant sur les positions des uns et des autres. Les nœuds actifs des zones les plus éloignées du chef de cluster envoient leurs données en les agrégeant au niveau des nœuds actifs des zones intermédiaires. Ensuite, le chef de cluster envoi les données à un nœud Wi-Fi qui sert comme une interface entre chaque cluster et les réseaux maillés. D'une autre part, les nœuds des

autres réseaux comme Wi-Max ou GPRS sont chargés de collecter et d'agréger les données provenant des nœuds Wi-Fi. Ces derniers sont considérés comme de simples émetteurs-récepteurs passifs dont le fonctionnement consiste seulement à envoyer les données à un serveur central situé dans le réseau câblé ou Internet.

En se basant sur la plage de détection et de communication R_s et R_c le chef de cluster (CH) sélectionne un ensemble contenant un nombre minimum de nœuds actifs, ce qui permet d'éliminer le trou de détection. Les nœuds sont sélectionnés en fonction de leur énergie résiduelle, leur distance, le nombre de tours de sommeil ainsi que leur espace mémoire. Chaque zone est divisée en plusieurs petites grilles carrées ou hexagonales. Chaque côté d'une grille carrée $\cong\ 2\ *\ R_S$. Tous les autres nœuds restent en mode veille. Chaque nœud actif ou en veille d'une zone est placé dans la plage de transmission d'au moins deux nœuds des zones voisines pour faire face au problème de tolérance aux pannes.

2) Distribution des nœuds et établissement du chemin

Dans l'approche proposée, le CH compatible Wi-Fi est placé à l'extérieur de la zone du réseau. Aussi des nœuds supplémentaires sont distribués sur les zones les plus proches du CH. La probabilité que les données collectées par les nœuds des zones éloignées passent par les nœuds des zones les plus proches du CH, est très élevée. Ainsi, les nœuds présents dans les zones les plus proches du CH consomment plus d'énergie et ont par conséquent une durée de vie plus courte. Pour garantir une durée de vie du réseau plus longue, un plus grand nombre de nœuds alternatifs peut être déployé dans ces zones. La probabilité de transmettre des données à travers les nœuds présents dans les zones les plus éloignées du CH est réduite du fait que les nœuds choisissent toujours le nœud actif voisin le plus proche du chef de cluster comme prochain saut, ce qui nécessite moins de puissance de transmission.

Les nœuds actifs se placent à différents niveaux de la hiérarchie virtuelle, et ceci en fonction du nombre de sauts qui les séparent du CH. Les nœuds appartenant à un niveau L_k connaissent la distance totale qui sépare les nœuds voisins au niveau supérieur $L_k - 1$ du chef de cluster. En se basant sur ces informations, les nœuds actifs établissent un chemin de communication avec les nœuds actifs voisins dont la distance totale vers le CH est minimale. Par conséquent, la consommation énergétique sera réduite.

3) Agrégation des données

L'agrégation des données de l'approche proposée est considérée comme dynamique puisque son type dépendra des exigences de l'application. Certaines applications impliquent une collecte de données périodique dans le but d'augmenter la précision des données. Pour cela, les nœuds intermédiaires transmettent uniquement les données non redondantes au CH, ce qui permettra au serveur central de traiter un grand nombre de données avec une précision élevée. Dans certaines applications, seules les données dont la valeur dépasse un seuil donné,

définit par le CH et transmis aux nœuds intermédiaires, sont transmises au CH. Par conséquent, l'agrégation des données au niveau des nœuds d'agrégation des zones supérieures est arrêtée. Par conséquent, si les nœuds relais reçoivent des données dont la valeur dépasse le seuil, définit par le CH, ils les transmettent au saut suivant sans exécuter la fonction d'agrégation. Sinon, les nœuds d'agrégation regroupent les données en fonction des fonctions d'agrégation sélectionnées telles que MAX, MIN et MOYENNE. Aussi, l'utilisateur peut définir une requête afin que la CH divise une requête en ses composants fondamentaux et la transmet vers des nœuds dans d'autres zones à différents niveaux. Les nœuds collectent les données pour ces composants et les agrègent, ce qui permet de réduire la quantité des données transmises au CH.

Les règles d'agrégation qui sont implémentées dans les nœuds peuvent être modifiées à distance par les utilisateurs. Le schéma TDMA est utilisé par les nœuds actifs afin de pouvoir fonctionner de manière circulaire. Les nœuds actifs qui se situent au niveau le plus éloigné L_k sont chargés de la détection et de l'envoi des données aux nœuds actifs se situant au niveau supérieur $L_k - 1$, et ceci à l'intervalle de temps 1. Les nœuds actifs de ce dernier détectent, agrègent, et envoient un accusé de réception aux nœuds du niveau L_k en se basant sur les règles d'agrégation définies. Par la suite, les nœuds envoient les données aux nœuds actifs du niveau $L_k - 2$ et vice versa. La durée des intervalles de temps varie et ceci au fur et à mesure que les nœuds intermédiaires reçoivent, agrègent et envoient les données au niveau supérieur.

L'approche proposée est évaluée et comparée à d'autres approches. Les résultats expérimentaux ont démontré que l'approche offre de meilleurs résultats en termes d'efficacité énergétique, de temps de propagation des données, de délai de bout en bout et de durée de vie du réseau.

7. Conclusion

Nous avons présenté à travers ce chapitre le mécanisme d'agrégation des données dans les RCSF, ses avantages et ses limites, ses différents types, ainsi que les principaux protocoles d'agrégation des données utilisés dans la littérature. Nous avons aussi étudié les mesures de performance de l'agrégation des données qui représentent les éléments clés de l'évaluation de ce processus. Comme notre travail est concentré sur l'agrégation des données dans les RCSF basés sur la technologie Big Data, nous avons survolé les différents protocoles d'agrégation des données proposés dans ce contexte. Les étapes principales de chaque protocole ainsi que leurs mécanismes de fonctionnement et leurs métriques d'évaluation sont illustrés.

Partie II

Contribution

Chapitre 4

Approche proposée

Chapitre 4 : Approche proposée

1. Introduction

L'agrégation des données dans les réseaux de capteurs sans fil et plus particulièrement dans les réseaux de capteurs sans fil hétérogènes représente un paradigme important dont le principal objectif consiste à éliminer le problème de transmission des données redondantes.

L'agrégation des données est l'un des principaux défis de traitement des données dans les réseaux de capteurs sans fil basés sur la technologie Big Data. En effet, les mécanismes d'agrégation des données peuvent représenter une solution efficace pour gérer les grands ensembles de données en combinant des données généralement similaires, ce qui élimine le problème de redondance des données et réduit par conséquent les quantités des données ainsi que la consommation des ressources des données.

Dans ce chapitre, nous proposons un mécanisme d'agrégation des données dans les réseaux de capteurs sans fil hétérogènes basés sur la technologie Big Data. Notre mécanisme nommé EEMR (Energy Efficient Mark Reduce Protocol) est inspiré des outils technologiques de traitement des données Big Data.

2. Problématique

Dans les réseaux de capteurs sans fil basés Big data, les données générées par les capteurs croissent de façon exponentielle. Les technologies de l'information conventionnelles, devant assurer le traitement et le stockage des données, ne peuvent généralement pas faire face aux besoins de traitement requis. De plus, la plupart des données recueillies à des intervalles réguliers sont en grande partie redondantes, entraînant une perte considérable de ressources de stockage de données et d'énergie de communication au niveau des nœuds. Cela implique qu'une grande partie de ces données sont sans intérêt.

Contrairement aux réseaux de capteurs sans fil typiques, il est essentiel de collecter et de transmettre une grande quantité de données tout en minimisant la latence des données dans les RCSF basés Big Data. De plus, il est nécessaire d'éliminer efficacement la redondance des données et d'améliorer l'efficacité énergétique. La fusion des systèmes Big Data et RCSF réside dans l'utilisation de techniques de traitement de données pouvant répondre aux besoins et aux challenges de ces deux technologies. Du côté des RCSF, cela permettrait d'économiser leurs ressources limitées, et recevoir des données non redondantes et pertinentes réduisant par conséquent les volumes des données excessifs du côté Big Data.

Afin de pouvoir faire face à ces problèmes, nous proposons un mécanisme d'agrégation des données inspiré de l'algorithme MapReduce de la technologie Big Data. Le mécanisme

proposé est nommé EEMR (Energy Efficient Mark Reduce), et est dédié aux réseaux de capteurs sans fil hétérogènes basés sur la technologie Big Data. Le mécanisme proposé est le résultat de la combinaison de trois challenges majeurs des réseaux de capteurs sans fils basés Big Data, à savoir le Clustering, le traitement des données et l'économie d'énergie.

Comme dans l'algorithme MapReduce, EEMR est basé sur l'utilisation de fonctions qui visent à assurer une agrégation optimale des grandes quantités de données en équilibrant les charges des données sur les nœuds de traitement tout en économisant l'énergie et en maximisant par conséquent la durée de vie du réseau.

L'objectif de l'équilibrage de charge est de tirer le meilleur parti des opportunités d'utilisation des ressources. Il vise à garantir qu'un nœud, dont les capacités ont atteint un seuil minimal, ne sera pas surchargé ce qui signifie que la charge de travail sera répartie de manière dynamique et adaptative entre les différents nœuds de traitement.

Le mécanisme proposé permet également de maintenir un délai tolérable. En conséquence, le réseau atteindra un débit maximal, le temps de réponse sera réduit et l'utilisation des ressources sera optimisée.

En résumé, EEMR vise à assurer les points suivants :

- Assurer l'agrégation des ensembles de données tout en optimisant la consommation énergétique et en maximisant par conséquent la durée de vie du réseau. Pour cela, le mécanisme proposé est basé sur l'utilisation de métriques à travers lesquelles les charges des données sur les nœuds sont équilibrées par la sélection dynamique des nœuds de traitement. En effet, lorsque les nœuds de traitement recevront les paquets de données principalement en fonction de leurs réserves d'énergie, leurs charges seront redistribuées vers d'autres nœuds avec plus de capacités, équilibrant ainsi la consommation énergétique sur les nœuds et maximisant par conséquent la durée de vie de l'ensemble du réseau.

- Notre mécanisme proposé est également amélioré en introduisant le mécanisme du feedback control afin de renforcer l'équilibre des charges de données sur les nœuds et de traiter efficacement le problème de planification de l'agrégation des données. En effet, le mécanisme du feedback control ou système de rétroaction permet d'ajuster les paramètres d'agrégation des données, ce qui adapte en conséquence les flux de données envoyés aux nœuds d'agrégation en fonction de leurs capacités de traitement à un instant donné.

- Le mécanisme proposé répond également au problème d'organisation des réseaux de capteurs sans fil. En effet, l'équilibre des charges de données sur les nœuds et l'ajustement des paramètres d'agrégation des données permettent de générer un délai tolérable pour une précision d’agrégation élevée.

3. Concepts de base

L'approche EEMR est basée sur différentes métriques de traitement utilisées dans les réseaux de capteurs sans fil ainsi que dans la technologie Big Data. Dans ce qui suit, nous présentons les mécanismes entrant dans la conception de notre approche :

3.1 L'algorithme K-means

L'algorithme K-means [130] [131] est une méthode de Clustering qui consiste à regrouper des observations en un nombre spécifique de clusters disjoints. Le nombre K fait référence au nombre de clusters groupés. L'objectif principal de l'algorithme consiste à minimiser la distance entre le centroide et les observations et ceci par l'ajout itératif des observations au cluster. Les itérations prennent fin lorsque la mesure de distance la plus basse est atteinte. Afin de déterminer les groupes auxquels les observations doivent être attribuées, plusieurs techniques pour le calcul de la distance peuvent être utilisées. Une observation donnée ne peut se retrouver que dans un cluster à la fois.

L'attribution des observations dans l'algorithme K-means est basée sur leur similarité. Par conséquent, les données similaires sont groupées dans le même cluster. Pour cela, et afin de grouper les données, le degré de similarité des différentes observations est mesuré. Les données ayant un degré élevé de similarité auront une distance réduite, et les données ayant un faible degré de similarité auront une distance plus grande.

Il existe plusieurs mesures pour le calcul de la distance, les plus utilisées dans le Clustering sont la distance Euclidienne et la distance de Manhattan.

- La mesure euclidienne [132], appelée aussi distance à vol d'oiseau, correspond à la distance géométrique la plus courte entre deux observations. Elle est calculée par la formule suivante :

$$d(x_1, x_2) = \sqrt{\sum_{j=1}^{n}(x_{1n} - x_{2n})^2} \qquad (2.1.1)$$

  Où : $d$représente la distance Euclidienne, $x_1 et\ x_2$ représentent les observations pour lesquelles d est calculée, n est le nombre total des observations.

- La distance de Manhattan [133] correspond à la somme des valeurs absolues des différences de coordonnées entre deux observations :

$$m(x, y) = \sum_{i=1}^{n} x_{i-}y_i \qquad (2.1.2)$$

Où : m représente la distance de Manhattan, x et y représentent les observations pour lesquelles m est calculée, n est le nombre total des observations.

Le choix du nombre K de clusters est généralement aléatoire mais doit être optimal. En effet, si K est très grand, le partitionnement des données risque d'être trop fragmenté. Si K est trop petit, les clusters formés peuvent contenir beaucoup de données. Ainsi, le problème majeur de l'algorithme K-means consiste en la difficulté à choisir un nombre de clusters K qui soit optimal. Jusqu'à ce jour, il n'existe pas de technique automatisée permettant de calculer le nombre optimal de clusters.

La méthode la plus utilisée pour le choix du nombre K consiste à lancer l'algorithme K-means en utilisant différentes valeurs de K et de calculer par la suite la variance des clusters formés :

$$V = \sum_j \sum_{x_i \to c_j} D\left(c_j, x_i\right)^2 \qquad (2.1.3)$$

Où : c_j représente le centroide, x_i représente la $i^{ème}$ observation liée au centroide c_j, et $d(c_j, x_i)$ représente la distance entre le centroide et l'observation.

Le calcul de la variance permet de minimiser la distance entre les centroides des clusters et les observations qui leurs sont liées.

3.1.1 Principe de fonctionnement de l'algorithme K-means

L'algorithme K-means est basé sur le calcul des distances entre chaque observation et le centroide. Il est à noter que le choix initial des centroides affecte le résultat final obtenu. L'algorithme K-means fonctionne comme suit :

1. Sélection d'un nombre aléatoire K qui représente le nombre de centroides.
2. Attribution de chaque observation au centroide correspondant en se basant sur le calcul de la distance et en sélectionnant le centroide le plus proche.
3. Recalcule des centroides de chaque cluster.
4. Réattribution des observations aux nouveaux centroides et itération jusqu'à convergence. La convergence de l'algorithme K-means est liée à l'une des conditions suivantes :
   - Soit le nombre d'itérations à effectuer est fixé à l'avance, donc l'algorithme K-means s'arrêtera au nombre fixé des itérations peu importe l'état des clusters formés.
   - Soit le nombre de centroides des clusters ne change pas lors des itérations.

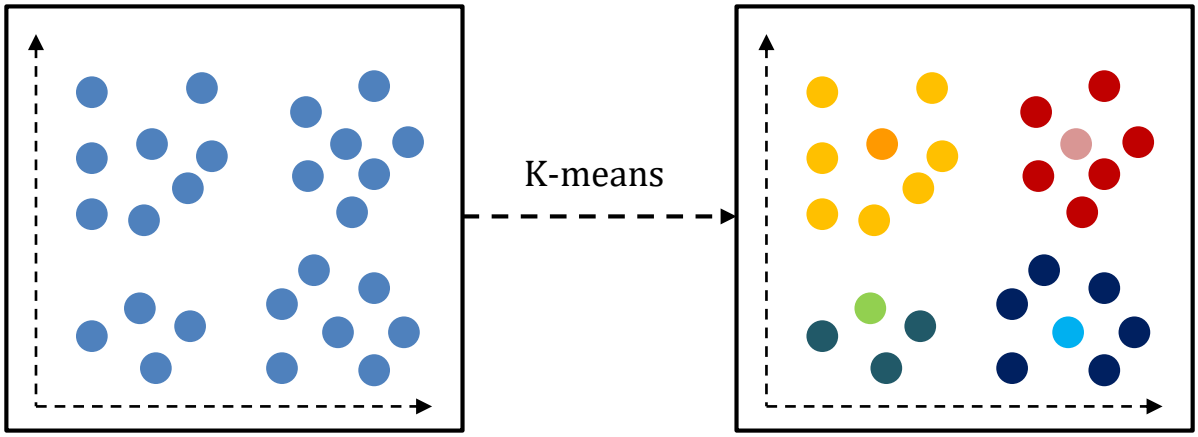


Figure 4-1. Algorithme K-means

3.1.2 Avantages et inconvénients de l'algorithme K-means

L'algorithme K-means présente des avantages considérables qui incitent à son utilisation :

- *Simplicité* **:** L'algorithme K-means est facile à implémenter et les résultats sont obtenus d'une manière rapide.
- *Flexibilité* **:** Correspond à l'adaptabilité de l'algorithme K-means aux différents changements effectués sur les données.
- *Big Data* **:** Une des caractéristiques importantes de K-means est que ses calculs sont très rapides lorsque de grands volumes de données sont utilisés.
- *Efficacité* **:** L'algorithme K-means est très efficace et permet de partitionner de grands ensembles de données en plusieurs clusters.
- *Précision*
- *Faible coût de calcul*
- *Facilité d'interprétation*

Toutefois, l'algorithme K-means peut aussi présenter quelques inconvénients :

- Le premier problème rencontré avec l'algorithme K-means est qu'il ne permet pas de créer des ensembles optimaux de clusters. En effet, les clusters doivent être choisis en premier lieu avant le démarrage de l'algorithme.
- Les résultats obtenus varient d'une exécution à l'autre, ce qui entraîne une incohérence de l'algorithme.
- Comme le choix du nombre de clusters se fait aléatoirement, Il est difficile de prévoir les valeurs de K qui doivent être spécifiées au début de l'algorithme.

3.2 L'algorithme MapReduce

Le paradigme du Big Data est basé sur des techniques analytiques qui sont principalement basées sur des architectures qui implémentent des algorithmes dédiés principalement pour le traitement des données volumineuses.

Hadoop [134] est une plateforme de traitement des données open source largement utilisée dans les applications exhaustives des données comme l'analyse des données Big Data. Il offre un environnement de traitement parallèle et distribué flexible et tolérant aux pannes. L'écosystème Hadoop utilise des modèles de programmation simples pour la gestion distribuée du Big Data. Il est basé sur quatre modules de traitement principaux :

- Le Hadoop Common composé d'un ensemble d'utilitaires et de bibliothèques prenant en charge les modules Hadoop,
- Le Hadoop Distributed File System (HDFS) qui représente la couche de stockage Hadoop prenant en charge le stockage des volumes importants de données non structurées,
- Le Hadoop YARN (Yet Another Resource Negotiator) responsable de la gestion des ressources des clusters, de la planification des tâches et de la surveillance des opérations de traitement des nœuds des clusters individuels,
- Le Hadoop MapReduce, qui implémente l'algorithme MapReduce.

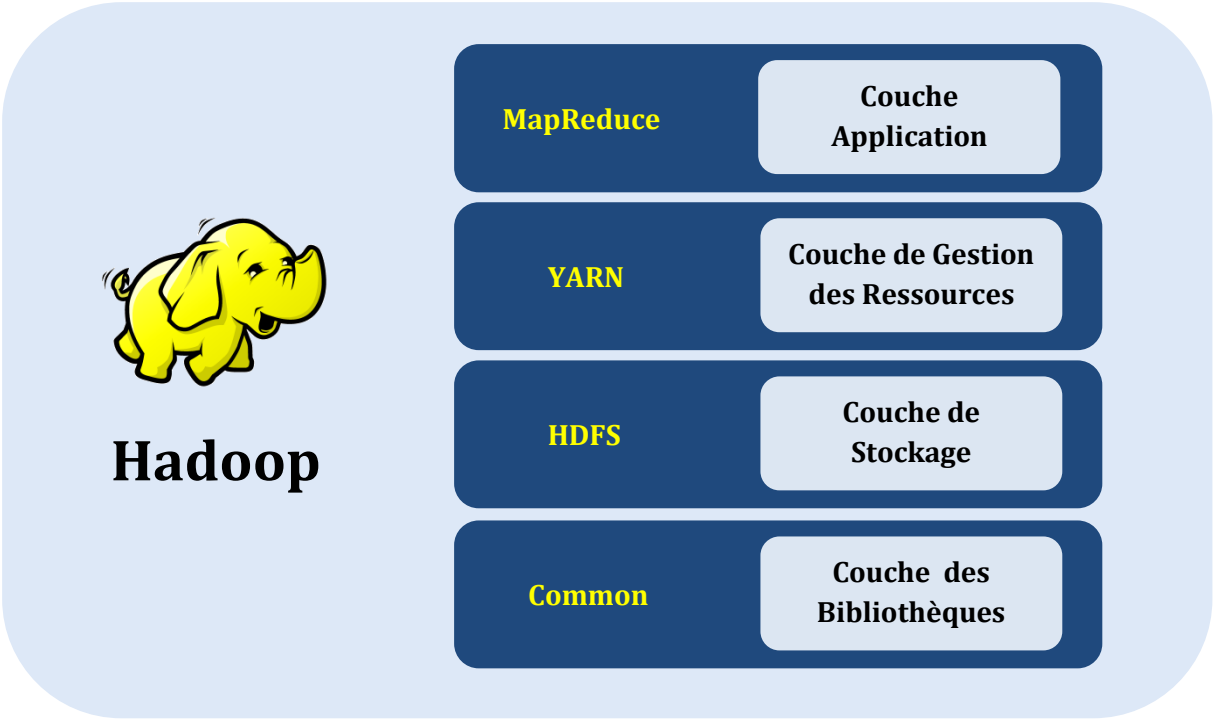


Figure 4-2. Architecture de l'écosystème Hadoop

La couche application de l'écosystème Hadoop implémente principalement l'algorithme MapReduce. MapReduce [135] est un algorithme de traitement des données qui représente le Framework le plus utilisé pour l'analyse des données Big Data. Il représente un modèle de programmation pour le traitement distribué de grands ensembles de données sur des clusters d'ordinateurs [136] [137]. MapReduce offre quantifiabilité, tolérance aux pannes, programmation facile, et adaptabilité.

MapReduce utilise un algorithme de traitement des données [138] composé de deux directives : Map and Reduce principalement utilisées pour traiter les données volumineuses dans les environnements distribués, en se basant sur les opérations suivantes :

- Itération sur un ensemble de données d'entrée,
- Création de paires clé/valeur par registre,
- Regroupement et enregistrement des résultats,
- Réduction de chaque groupe.

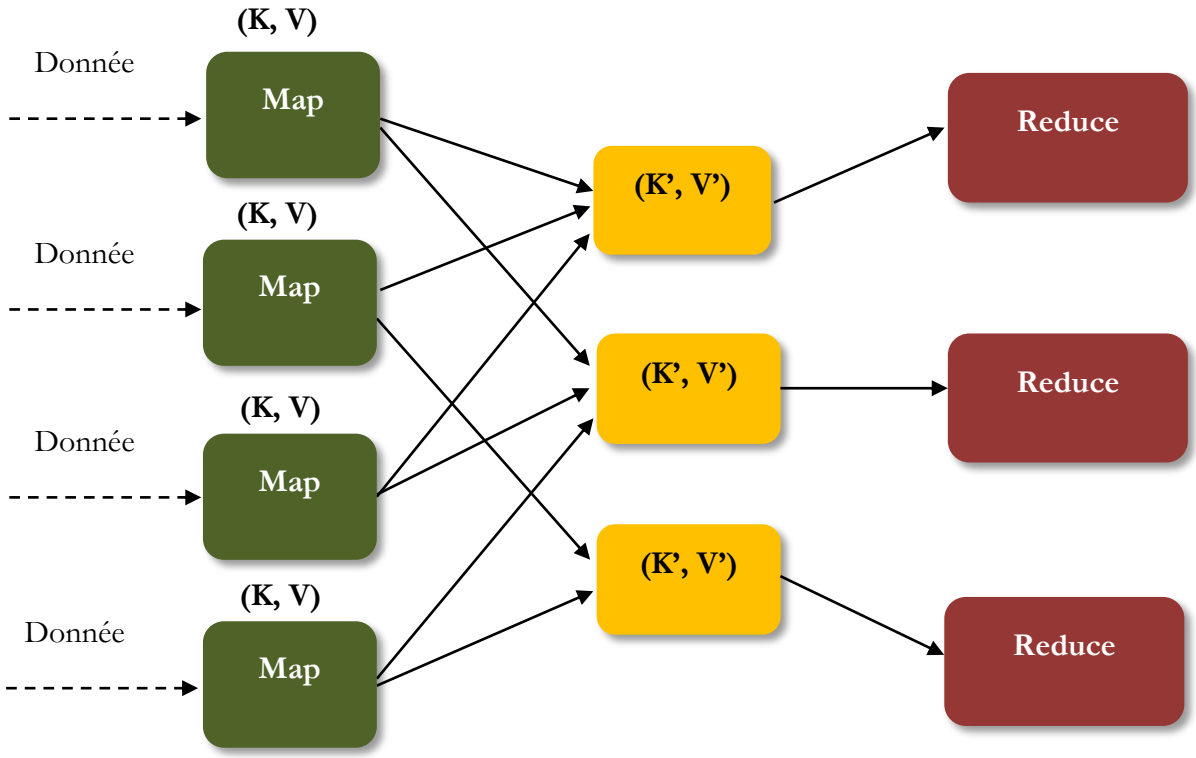


Figure 4-3. Algorithme MapReduce

La directive Map permet de répartir la charge de travail sur les différents nœuds du cluster. Elle crée plusieurs petits morceaux de données à partir des données d'entrée stockées dans le HDFS et reçues ligne par ligne. Pour cela, la fonction Map transforme les données reçues en paires <clé, valeur>, les traite et génère un autre ensemble de paires <clé, valeur> intermédiaires à la sortie.

Les données sont ensuite transmises à la directive Reduce pour produire un nouvel ensemble de sortie stocké dans le HDFS. Ainsi, les résultats fournis par chaque nœud sont organisés et réduits en une seule réponse cohérente à une requête.

Les directives Map et Reduce sont envoyées par Hadoop aux serveurs appropriés du cluster. Le système vérifie l'achèvement de la tâche de données et copie les données autour du cluster entre les nœuds. Ensuite, les données sont collectées, agrégées et renvoyées au serveur Hadoop.

4. Approche proposée

4.1 Le Clustering du réseau

La première étape dans l'approche proposée consiste à organiser les nœuds du réseau. Nous considérons un réseau de capteurs sans fil hétérogène densément distribué, composé d'une station de base et de plusieurs nœuds homogènes et hétérogènes. Les nœuds homogènes sont plus nombreux et possèdent des capacités de traitement et de stockage limitées, tandis que les nœuds hétérogènes sont plus puissants avec des capacités plus importantes.

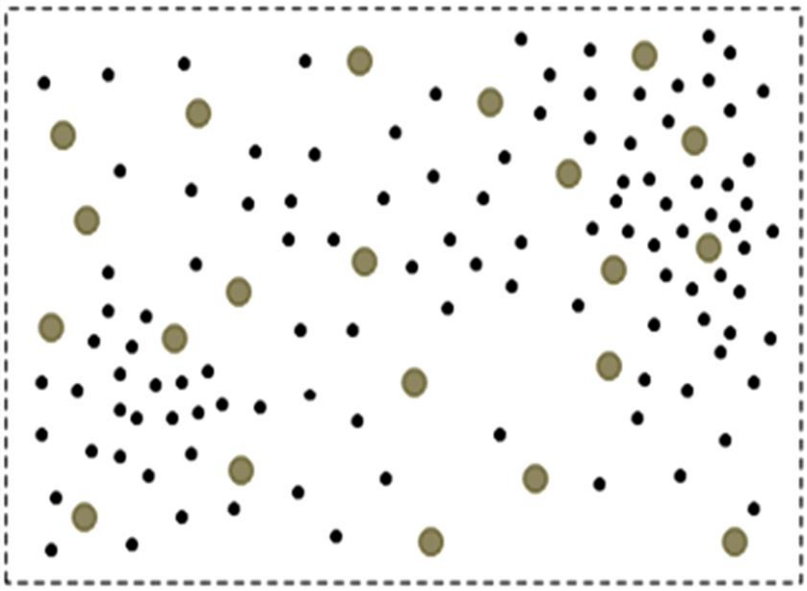


Figure 4-4. Déploiement des nœuds du réseau

Le Clustering du réseau représente la première et la principale étape dans la hiérarchie de classification des challenges Big Data dans les réseaux de capteurs sans fil. Le Clustering permet de déterminer l'organisation des nœuds dans le réseau, leur positionnement par rapport aux autres nœuds et par rapport à la station de base. Il détermine également les chemins et l'ordre dans lesquels les données sont transmises, la manière dont elles sont transmises et les stratégies utilisées pour leur transmission. Le Clustering permet également de définir les stratégies de communication entre les nœuds du réseau.

4.1.1 Matrice des nœuds

Avant de procéder au Clustering, le réseau est d'abords représenté comme une matrice virtuelle partitionnée en plusieurs cellules de nœuds. L'objectif étant en premier lieu d'équilibrer la distribution des nœuds en clusters. Aussi, le partitionnement permettra l'exécution des techniques de Clustering et de traitement d'une manière indépendante dans chaque cellule, assurant par conséquent une diminution du temps d'exécution global et de la consommation énergétique, ce qui rend les techniques utilisées dans cette approche plus adaptées au Big Data dans les réseaux de capteurs sans fil.

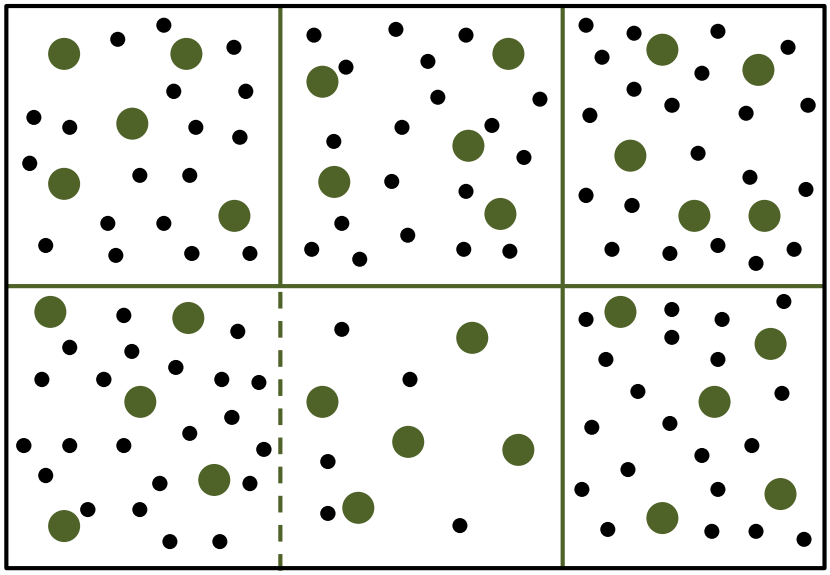


Figure 4-5. Matrice des nœuds

Le partitionnement du réseau est exécuté par la station de base qui détient les informations de localisation des nœuds, et divise par conséquent la matrice des nœuds. Le nombre de cellules dépend principalement du nombre de nœuds hétérogènes qui seront déployés dans le réseau. Pour une distribution équitable des nœuds hétérogènes dans chaque cellule, nous supposons que la création de la matrice des nœuds est effectuée avant le déploiement de ces derniers. Le nombre de cellules est calculé par l'équation suivante :

$$Nb_{Cells} = \left\lfloor {}^{\sqrt[2]{H}}/_{2} \right\rfloor \quad (1)$$

Où : H représente le nombre de nœuds hétérogènes.

Ainsi, pour un réseau contenant 150 nœuds hétérogènes, la matrice sera partitionnée en 6 cellules composée chacune de 25 nœuds hétérogènes, ce qui représente une valeur tolérable.

Ensuite, la station de base qui connaît la position de chaque nœud dans le réseau, ainsi que la taille de chaque zone qui est déterminée en fonction de la taille du réseau ainsi que le nombre

de cellules, crée la matrice en déterminant à quelle cellule appartient chaque nœud dans le réseau.

Comme les nœuds homogènes sont déployés aléatoirement, il peut y avoir des cellules très denses ce qui peut entraîner des charges élevées sur les nœuds de traitement. Inversement, il peut y avoir des cellules largement moins denses ce qui peut entrainer une dissipation énergétique non optimale.

Pour résoudre ce problème, la station de base procède à une fusion des cellules dans la matrice. En effet, la station de base vérifie la densité des nœuds homogènes dans chaque cellule. Si une cellule est très dense, la station de base la fusionne avec la cellule voisine qui possède le plus petit nombre de nœuds homogènes comparée aux autres cellules voisines. En conséquence, le nombre de nœuds hétérogènes dans la cellule résultant de la fusion sera augmenté en parallèle avec le nombre de nœuds homogènes réduisant par conséquent les charges des données sur ces nœuds.

4.1.2 Clustering intra-cellules

Après le partitionnement de la matrice des nœuds, la phase de Clustering intra-cellules débute. Le schéma de Clustering doit répondre à un critère important, qui consiste à diviser les nœuds en clusters optimaux en termes de réduction de la consommation énergétique et d'équilibre dans la répartition des nœuds. Pour cela, nous adoptons une approche de Clustering hybride dans laquelle l'organisation des clusters dans le réseau est assurée par la station de base ainsi que les nœuds hétérogènes.

Une fois les nœuds déployés dans le réseau, la phase de Clustering prend lieu dans chaque cellule de nœuds afin de décider de l'emplacement où les données seront collectées et traitées. Le Clustering est basé sur deux paramètres importants :

- Les réserves énergétiques des nœuds hétérogènes.

- Leurs positions par rapport à la station de base.

1. La première étape de la phase de Clustering correspond à la sélection des CHs dans chaque cellule. Le choix du nombre de CHs ainsi que leur sélection représente une étape cruciale. Initialement, tous les nœuds hétérogènes représentent des chefs de clusters potentiels. La sélection des CHs initiaux peut être comparée à la première étape de l'algorithme de Clustering K-means. Ainsi, comme dans K-means, la station de base sélectionne aléatoirement C nœuds, qui représenteront les CHs initiaux et par conséquent le nombre de clusters du cycle en cours. Contrairement à l'algorithme K-means dans lequel le nombre K est complètement aléatoire, dans l'approche proposée le nombre de clusters est étroitement lié au nombre de nœuds présents dans la cellule :

$$C = \left\lfloor \sqrt[2]{Hc + Sc} / L \right\rfloor \quad (2)$$

Où Hc représente les nœuds hétérogènes dans la cellule, Sc représente les nœuds homogènes, et L est le nombre de niveaux dans le cluster qu'on détaillera par la suite.

Après la sélection initiale des chefs de clusters CHs, ces derniers diffusent un message dans la cellule contenant leurs positions. En se basant sur la position des CHs, chaque nœud hétérogène s'affecte au CH le plus proche et diffuse un message contenant sa position ainsi que ses réserves énergétiques. Les réserves d'énergie E_{res} de chaque nœud hétérogène sont calculées en utilisant l'équation suivante :

$$E_{res} = E_{process} + E_{trans} \quad (3)$$

Où $E_{process}$ représente l'énergie de traitement du nœud et E_{trans} son énergie de communication.

2. Pour équilibrer les charges de traitement, le nombre de nœuds hétérogènes que doit contenir chaque cluster est calculé par l'équation suivante:

$$Nbr_H = H/C \pm \omega \quad (4)$$

Où ω représente le seuil minimal ou maximal fixé pour le nombre de nœuds hétérogènes que doit contenir un cluster.

A la fin de l'étape précédente, chaque CH vérifie le nombre de nœuds hétérogènes qui lui sont affectés. Lorsque ce nombre est inférieur au seuil fixé, le CH diffuse un message informatif. A la réception de ce message, chaque CH possédant un nombre de membres supérieur au seuil fixé retient les nœuds qui lui sont les plus proches et diffuse un message dans son cluster contenant leurs identificateurs. Les nœuds écartés s'affectent au prochain CH le plus proche possédant un nombre de nœuds hétérogènes inférieur au seuil fixé. Cette étape est répétée jusqu'à ce que le nombre de nœuds soit équitable dans chaque cluster.

3. Après la formation des clusters initiaux, Le CH va procéder à une sélection des nœuds qui peuvent être candidats pour devenir des CHs. En effet, seuls les nœuds possédant des réserves énergétiques supérieures à un seuil donné peuvent devenir de nouveaux chefs de clusters. Pour cela, chaque CH calcule la moyenne E_{moy} de toute l'énergie des nœuds de son groupe en utilisant la formule suivante :

$$E_{moy} = \sum_{i=1}^{N} \frac{E_{reserves(i)}}{N} \quad (5)$$

Où : N représente les nœuds hétérogènes du cluster.

Tous les nœuds hétérogènes dont les réserves énergétiques sont supérieures ou égales à E_{moy} représentent de nouveaux CHs potentiels. Ainsi, chaque CH envoie un message à la station de base contenant les identificateurs des nœuds hétérogènes éligibles pour être sélectionnés comme nouveaux CHs.

En se basant sur les informations envoyées par les CHs, la station de base sélectionne, pour chaque cluster, le meilleur nœud parmi les nœuds candidats pour devenir des CHs. Le critère de la sélection est la position des nœuds par rapport à la station de base. En effet, la station sélectionne les nœuds qui lui sont les plus proches parmi les nœuds candidats pour devenir des CHs.

4. La station de base rediffuse un message résultat dans les clusters pour informer les nœuds de leurs nouveaux CHs.

Après l'étape de Clustering, chaque cluster du réseau sera divisé dynamiquement en quatre niveaux selon le type de nœuds, leur position, et leurs réserves d'énergie comme suit :

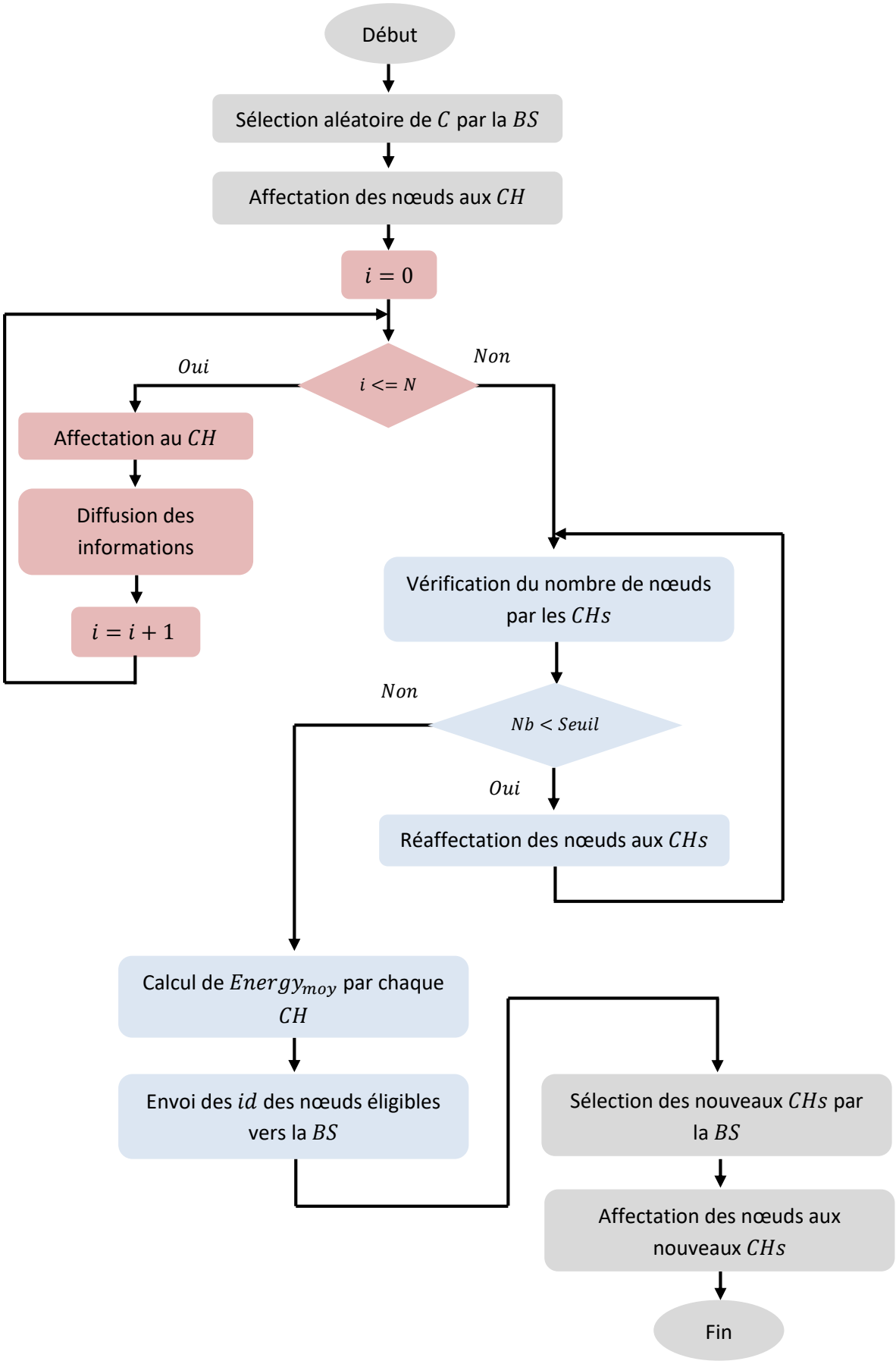
- $Niveau\ 0$: Représente le niveau principal du cluster. Comme décrit précédemment, ce niveau contient le CH qui est responsable de l'agrégation des données et du routage des paquets de données vers la station de base.

- $Niveau\ 1$: Il contient des nœuds homogènes de capacités de traitement et de stockage limitées. Les nœuds de ce niveau sont chargés de collecter les données de l'environnement et de les transmettre au niveau suivant. Les nœuds de ce niveau sont appelés les *Collecteurs* (Collectors).

- *Niveau 2* : Ce niveau est formé de nœuds hétérogènes appelés les *Marqueurs* (Markers), qui permettent d'aiguiller les paquets de données reçus du niveau inférieur vers les nœuds correspondants dans le niveau supérieur.

- *Niveau 3* : Le niveau 3 est formé de nœuds hétérogènes appelés *Réducteurs* ou *Agrégateurs* (Reducers or Aggregators), qui sont chargés du traitement des données et de la transmission des résultats vers le CH.

La répartition des nœuds des clusters est détaillée ultérieurement dans la phase de traitement.



Organigramme 4-1. Sélection des CHs et formation des clusters

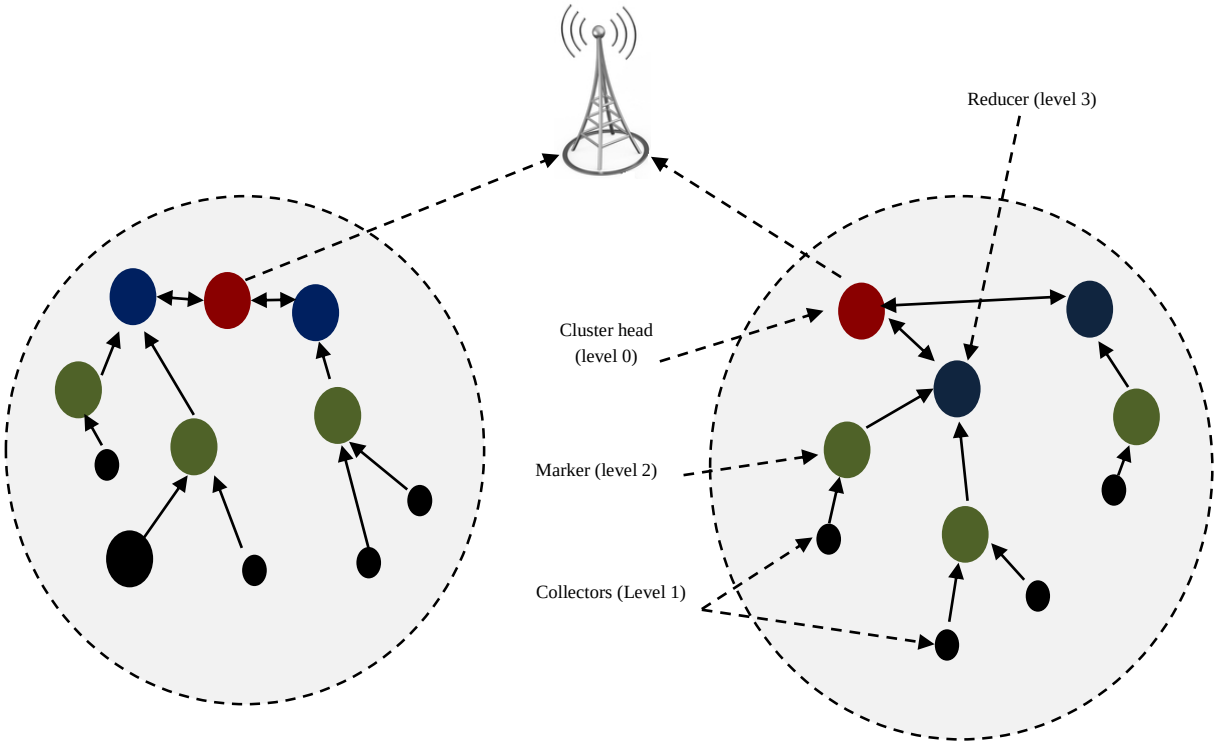


Figure 4-6. Le cluster formé

4.2 Le traitement des données

Après la sélection des CHs, la phase de traitement prend place, et dans laquelle le schéma d'agrégation des données écoénergétique EEMR est illustré. Dans le schéma proposé et comme indiqué précédemment, chaque cluster est divisé en quatre niveaux, dans lesquels les nœuds possèdent des fonctionnalités spécifiques. Comme dans l'algorithme MapReduce, deux directives sont utilisées aux niveaux supérieurs du cluster :

La première directive de l'approche est appelée *«la directive de marquage»* ou *«Mark function»* qui peut être comparée à la fonction *Map* de l'algorithme MapReduce. Cette étape vise à organiser efficacement les nœuds hétérogènes et les paquets de données reçus. Selon l'organisation, les nœuds Marqueurs aiguillent les paquets de données aux nœuds de traitement correspondants. La directive de marquage vise aussi à optimiser les nœuds hétérogènes des niveaux 2 et 3 en fonction de leurs réserves d'énergie et des paquets de données échangés. Pour cela, elle comporte une fonction nommée *« la fonction de réglage ou d'optimisation»*.

La deuxième directive est appelée *« la fonction de réduction ou d'agrégation»* ou *(Reduce function)*, dans laquelle les paquets de données sont agrégés à deux niveaux : les Réducteurs et le CH.

Le schéma de traitement est détaillé comme suit :

Le processus de collecte des données est effectué au bas niveau du réseau par les nœuds Collecteurs qui sont des nœuds homogènes. Les Collecteurs étant des nœuds à faibles capacités et réserves énergétiques, leur rôle consiste à recevoir les données collectées de l'environnement et à les transmettre directement aux nœuds Marqueurs désignés.

4.2.1 La directive Mark

Le processus de traitement vise en premier lieu à proposer un mécanisme qui équilibre et adapte dynamiquement les charges des données sur les nœuds responsables de traitement. Pour cela, le processus de marquage est divisé en plusieurs étapes permettant d'organiser initialement les niveaux élevés de chaque cluster, qui se composent de Marqueurs et de Réducteurs, avant de procéder à la transmission des données. Ensuite, l'organisation des nœuds est optimisée périodiquement.

4.2.1.1 Organisation des nœuds hétérogènes

Avant de démarrer le processus de traitement, chaque cluster est divisé dynamiquement en plusieurs groupes de nœuds Réducteurs/Marqueurs. L'objectif étant d'équilibrer la distribution des paquets de données ainsi que leurs charges de traitement sur les nœuds.

L'organisation initiale des nœuds est inspirée principalement de l'algorithme de Clustering proposé. L'algorithme fonctionne comme suit :

Etant donné l'ensemble $H = \{H_1, H_2, \dots, H_x\}$ des nœuds hétérogènes subsistants après la sélection des CHs, la première étape de l'organisation des nœuds consiste à sélectionner les nœuds qui représenteront les Réducteurs. Pour cela, le CH sélectionne R nœuds hétérogènes parmi les nœuds ayant les plus grandes réserves énergétiques selon l'équation suivante :

$$R = \lfloor {}^{H}/_{3} \rfloor \qquad (6)$$

Où le nombre 3 représente le nombre minimum de nœuds hétérogènes de traitement qui peuvent être présents dans un groupe Réducteur/Marqueurs. En effet, pour chaque Réducteur, il peut y avoir au minimum deux Marqueurs afin de distribuer équitablement les données.

L'ensemble des Réducteurs $R_K = \{R_1, R_2, \dots R_K\}$ sélectionnés représente la liste des clés d'entrée $List_{K1}$ de la fonction d'organisation des groupes. L'ensemble $M_V = \{M_1, M_2, \dots M_V\}$ des nœuds hétérogènes restants représente la liste de valeurs initiale $List_{V1}$ de cette fonction.

$$EEMR(List_{K1}, List_{V1}) \equiv MapReduce(K, V)$$

Une fois la paire $(List_{K1}, List_{V1})$ est formée, l'étape suivante consiste à affecter chaque nœud Marqueur de la liste des valeurs $List_{V1}$ à son Réducteur correspondant dans la liste des clés

$List_{K1}$. Pour cela, la même étape que dans l'étape de Clustering est exécutée. En effet, chaque Réducteur diffuse un message dans le cluster et chaque nœud hétérogène s'affecte au Réducteur le plus proche et diffuse un message contenant sa position ainsi que son.

Ensuite, Chaque Réducteur dont le nombre de Marqueurs est inférieur au seuil fixé diffuse un message informatif. A la réception de ce message, les Réducteurs possédant un nombre de Marqueurs supérieur au seuil fixé retiennent les nœuds qui leurs sont les plus proches et diffusent un message dans leurs groupes contenant leurs identificateurs. Les nœuds écartés s'affectent au prochain Réducteur le plus proche possédant un nombre de Marqueurs inférieur au seuil fixé.

L'ensemble des Réducteurs $R_{K'} = \{R_{K'1}, R_{K'2}, \dots\}$ finaux représente la liste des clés de sortie $List_{K2}$ de la phase de marquage. L'ensemble $M_{V'} = \left\{M_{(V'_1,K'_x)}, M_{(V'_2,K'_x)} \dots.\right\}$ des Marqueurs finaux attribués aux nouveaux Réducteurs sélectionnés représente la liste des valeurs de sortie $List_{V2}$ de la phase de marquage.

4.2.1.2 Transmission des données

Une fois que les groupes finaux sont formés, les nœuds commencent à envoyer leurs paquets de données vers les nœuds correspondants. Lors de l'étape de formation des groupes, les Marqueurs diffusent un message contenant leur position et leur statut. Par conséquent, les nœuds homogènes envoient directement leurs paquets de données vers les Marqueurs qui leurs sont les plus proches. Ces derniers se chargent de les diriger vers le Réducteur correspondant. Ce schéma de transmission permet d'aguiller les paquets de données en empruntant le chemin le plus court pour chaque nœud homogène. D'une autre part, il permet de prévenir le problème de congestion résultant de la surcharge des Réducteurs dans le cas où les nœuds homogènes envoient leurs paquets de données directement vers ces derniers.

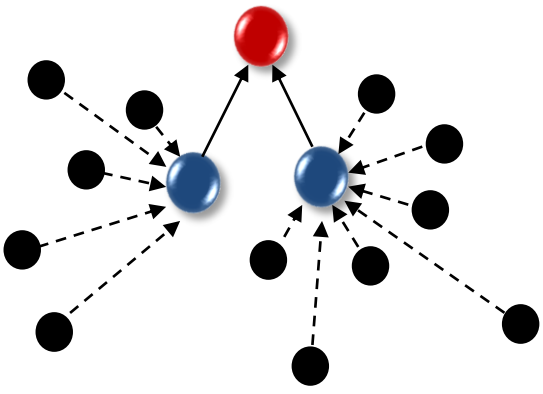


Figure 4-7. Schéma de transmission des données

4.2.2 La directive de réduction ou d'agrégation

La fonction de réduction est exécutée à deux niveaux. D'abord au niveau des Réducteurs qui agrègent les paquets de données qu'ils reçoivent des Marqueurs, et envoient les résultats

agrégés au CH. Pour une agrégation des données et une consommation d'énergie optimales, le CH effectue une deuxième agrégation des paquets de données reçus de ses Réducteurs et envoie les résultats vers la station de base.

La fonction de réduction représente un élément essentiel dans l'approche proposée en termes d'amélioration de l'efficacité énergétique, car les Réducteurs et le CH fusionnent les données sur leur chemin vers la station de base, réduisant par conséquent le nombre de paquets à transmettre.

La phase de réduction est contrôlée par le paramètre de temps d'attente d'agrégation. En effet, chaque nœud se voit attribuer un temps d'attente des paquets de données avant de procéder à leur agrégation. Généralement, plus le temps d'attente est long, plus l'agrégation des données est précise. Dans la section suivante, un mécanisme d'adaptation du temps d'attente d'agrégation des Réducteurs sera défini.

Algorithme 1 Réduction ou Agrégation

1: ET_R: $Seuil\ d'énergie\ des\ Réducteurs$
2: Démarrage de l'algorithme avec les paquets de données $(R_{id}, List(Data))$
3: **BEGIN**
4: $Réducteur < - Marqueur(R_{id}, List(Data))$
5: **If** $(AWT\ expire)\&(E_R > ET_R)$
6: $Réducteur(Data_{agg})$
7: $CH < -Data_{agg}$
8: **End if**
9: $CH(Data_{agg'})$
10: $BS < -Data_{agg'}$
11: **END**

4.2.3 La fonction de réglage (optimisation)

1. La fonction de réglage est exécutée périodiquement afin d'optimiser les groupes Réducteurs/Marqueurs en les réorganisant en fonction de leurs réserves d'énergie.

Le couple $(List_{K'}, List_{V'})$ issu de la formation des groupes de nœuds représente l'entrée de la fonction d'optimisation des nœuds.

Durant la phase de traitement, les Réducteurs vérifient périodiquement leurs réserves énergétiques. Lorsque les réserves d'énergie d'un nœud atteignent leur seuil minimal, le nœud diffuse un message dans son groupe afin d'informer ses Marqueurs. Les Marqueurs se communiquent leurs réserves énergétiques, et le Marqueur ayant les plus grandes

réserves, est élu comme nouveau Réducteur. Si plusieurs Marqueurs possèdent les mêmes réserves énergétiques, le Marqueur le plus proche du CH sera élu.

La fonction de réglage pour l'organisation des nœuds peut être comparée à la fonction Reduce de l'algorithme MapReduce, à la différence que dans la fonction proposée il n y a pas d'opération de réduction pour les nœuds mais une réorganisation de ces derniers.

A l'issue de cette fonction, de nouvelles paires $(List_{K"}, List_{V"})$ sont produites à la sortie et qui correspondent respectivement aux nouveaux Réducteurs et Marqueurs correspondants.

Algorithme 2 Regroupement des nœuds

1: ET_R : $Seuil\ d'énergie\ pour\ les\ Réducteurs$
2: **BEGIN**
3: Pour les Réducteurs
4: **FOR** $(i = 1, R_{K'}, i++)$
5: **IF** $(E_{R_{K'i}} \leq ET_R)$
6: Diffusion d'un message dans le groupe.
7: Communication des réserves énergétiques.
8: Sélection du nouveau Réducteur.
9: Diffusion d'un message dans le cluster.
10: **END**

2. La fonction de réglage permet aussi d'équilibrer les charges des données sur les nœuds Réducteurs en fonction de leur temps d'attente d'agrégation et de leurs capacités de file d'attente des données. Pour cela, chaque nœud Réducteur, vérifie périodiquement ces paramètres. Des scénarios sont envisagés :

Scénario 1 :

Si la file d'attente (FA) des données d'un nœud Réducteur est pleine, une réorganisation temporaire du groupe auquel appartient le Réducteur est effectuée. En effet, les Marqueurs du groupe aiguillent les paquets de données reçus vers les Réducteurs qui leur sont les plus proches, ayant une file d'attente non pleine et dont le temps d'attente d'agrégation n'est pas expiré ou est sur le point d'expirer. Les Marqueur marquent ainsi leurs paquets de données avec l'ID des nouveaux Réducteurs. L'affectation temporaire prend fin lorsque les files d'attentes des nouveaux Réducteurs sont pleines, leur temps d'attente d'agrégation expire, ou la file d'attente du Réducteur concerné n'est plus pleine.

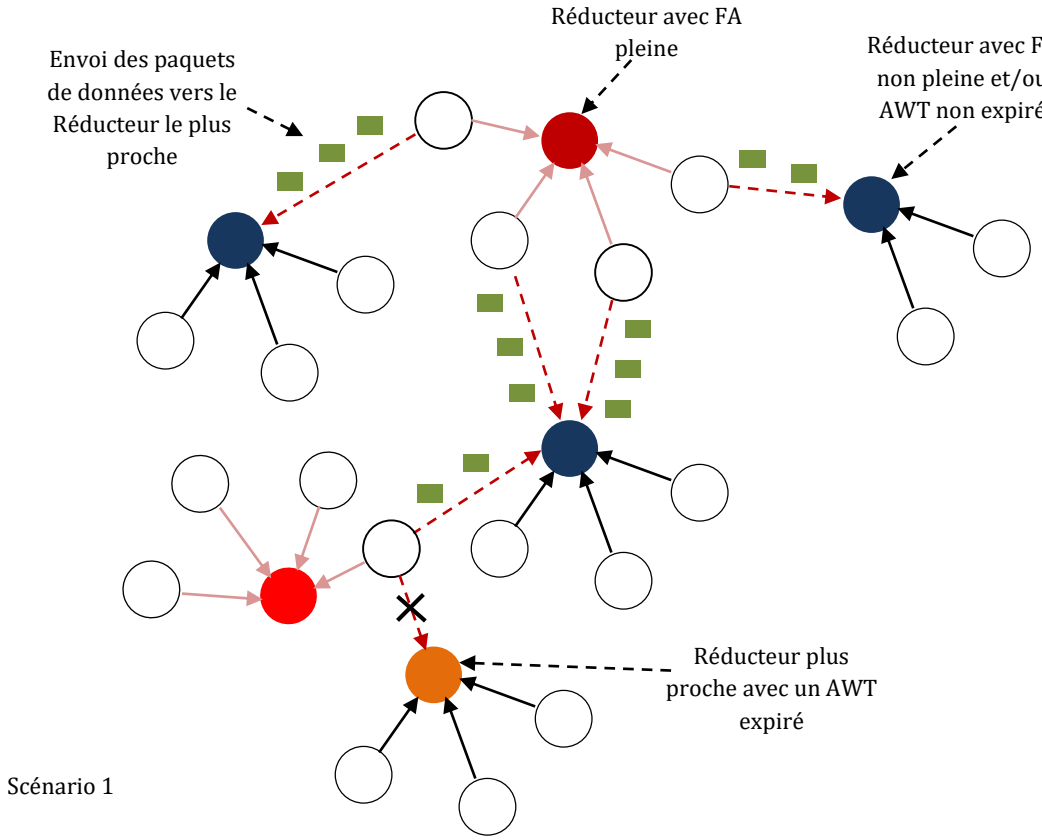
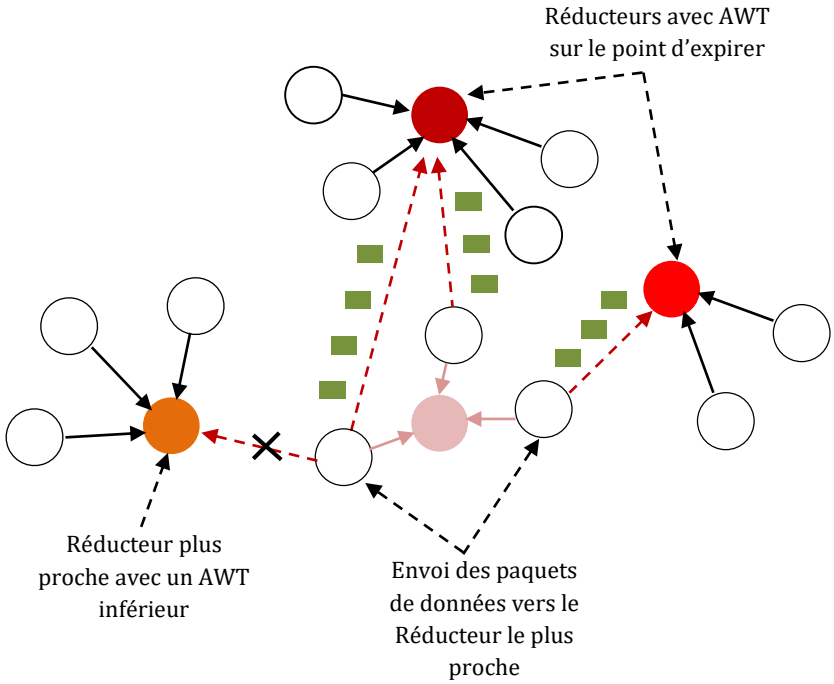


Figure 4-8. Réglage d'envoi des paquets de données (scénario 1)

Scénario 2 :

Si le temps d'attente d'agrégation d'un Réducteur est sur le point d'expirer et que la file d'attente (FA) des données du nœud n'est pas pleine, une réorganisation temporaire des Marqueurs des groupes avoisinants est effectuée. L'organisation temporaire des groupes repose sur deux paramètres : la distance entre les nœuds et l'état des Réducteurs. En effet, les Marqueurs voisins dont le temps d'attente des Réducteurs est supérieur au temps d'attente du Réducteur concerné, aiguillent temporairement leurs paquets de données vers ce Réducteur.

Lorsque le temps d'attente d'agrégation expire ou que la file d'attente de données est pleine, les Marqueurs aiguillent les paquets de données aux Réducteurs auxquels ils sont initialement affectés. Cette technique permet aux Réducteurs d'agréger un maximum de données dans leur temps d'attente d'agrégation augmentant par conséquent la précision d'agrégation des données.



Scénario 2

Figure 4-9. Réglage d'envoi des paquets de données (scénario 2)

Algorithme 3 Réglage d'envoi des paquets de données selon FA

1 : Exp_{tresh} : Seuil de remplissage pour les Réducteurs.
2: R_{dq}: Nombre de paquets dans la file d'attente des Réducteurs.
3: R_{cap}: Capacité de la file d'attente des Réducteurs.

4: **DEBUT**
5: **FOR** $(i = 1, R_{Ki}, i++)$
6: **SI** $\left(Ri_{cap} = Exp_{tresh}\right)$
7: Le Réducteur diffuse un message.
8 : Les Réducteurs envoient leurs statuts.
9: Les Marqueurs aiguillent leurs paquets de données vers les nouveaux Ri.
10: **FINSI**
11: **SI** $Ri_{dq} = 0$
12: Les Marqueurs marquent leurs paquets de données avec les IDs originaux.
13: **FINSI**
14: **FIN**

Algorithme 4 Réglage d'envoi des paquets de données Selon AWT

1: R_{AWT} : $AWT\ restant$
2 : Exp_{tresh} : Seuil d'expiration pour les Réducteurs.
3: R_{dq}: Nombre de paquets dans la file d'attente des Réducteurs.
4: R_{cap}: Capacité de la file d'attente des Réducteurs.

5: **DEBUT**
6: **FOR** $(i = 1, R_K, i++)$
7: **SI** $(R_{AWT} = Exp_{tresh})$ & $(R_{dq} < R_{cap})$
8: Le Réducteur diffuse un message.
9 : Les Réducteurs envoient leurs statuts.
10: Les Marqueurs aiguillent leurs paquets de données vers les nouveaux R_{Ki}.
11: **FINSI**
12: **SI** $AWT_{RK'i} = 0$
13: Les Marqueurs marquent leurs paquets de données avec les IDs originaux.
14: **FINSI**
15: **FIN**

4.3 Mécanisme du Feedback Control pour le contrôle d'agrégation des données

Dans cette partie, un modèle en boucle fermée du Feedback Control est intégré à notre approche pour améliorer et équilibrer le processus d'agrégation des données. Le système du Feedback Control ou contrôle de rétroaction [139] [140] est basé sur l'utilisation d'un ensemble de variables de contrôle et d'une boucle de rétroaction qui surveille et supervise en continu le comportement du système de contrôle.

Le mécanisme du Feedback Control et plus précisément le mécanisme à boucle fermée [141] [142] représente une base qui permet à différents systèmes de maintenir leur homéostasie en comparant leurs valeurs réelles à celles souhaitées. La boucle fermée compare le comportement réel par rapport à une spécification du comportement attendu, en ajustant en conséquence le système afin d'assurer la conformité aux performances comportementales.

Le mécanisme du Feedback Control a prouvé son efficacité dans la mesure où il est possible de contrôler même si des erreurs ont été commises pendant la phase de modélisation, ou plus lorsque l'environnement change. Un système de Feedback Control à boucle fermée typique est représenté par la figure suivante :

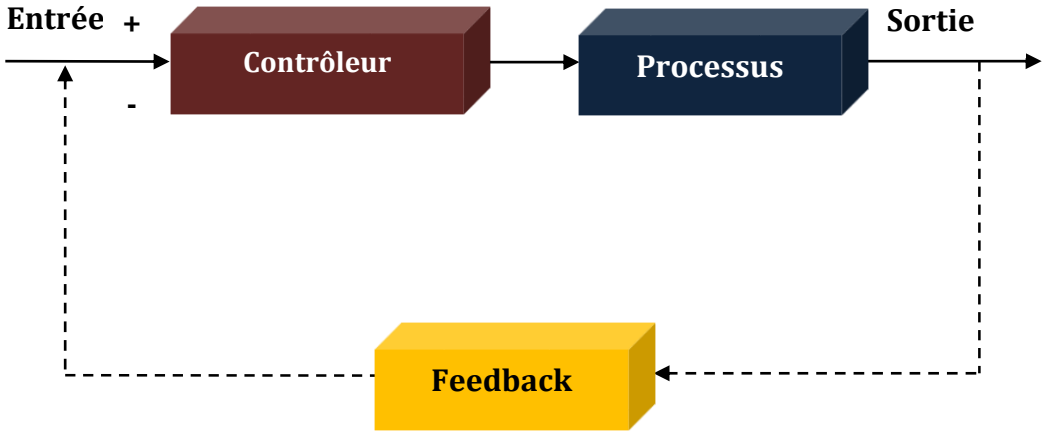


Figure 4-10. Mécanisme du Feedback Control

L'utilisation du mécanisme de contrôle par rétroaction présente de nombreux avantages :

- Permet d'obtenir des performances élevées même en cas d'incertitude ;
- Assure la correction des erreurs grâce au calcul ;
- Permet de modifier la dynamique d'un système ;
- Le contrôle implique des compromis entre robustesse et performance.

L'utilisation du contrôle par rétroaction présente également certains inconvénients :

- Il peut créer des instabilités dynamiques dans les systèmes.
- Il peut introduire un bruit indésirable dans le système nécessitant un filtrage des signaux.

Récemment, le mécanisme du Feedback Control a été introduit pour optimiser dynamiquement le processus d'agrégation des données dans les réseaux de capteurs sans fil. Ce mécanisme a prouvé son efficacité en termes de distribution de l'optimisation des performances de l'agrégation des données et de son adaptation aux changements de l'environnement. Étant donné que le contrôle par rétroaction est un domaine de recherche relativement nouveau pour l'agrégation des données dans les réseaux de capteurs sans fil et que peu de recherches ont utilisé ce mécanisme, dans [143], nous avons souligné ce domaine de recherche en examinant les différentes solutions existantes :

Le protocole proposé dans [144] et nommé AIDA vise à remédier aux limitations de la bande passante et de l'énergie causées par le manque d'adaptation du mécanisme et la dépendance des applications pour prendre des décisions. Le protocole tend à maximiser l'utilisation du canal de communication tout en économisant de l'énergie. Il utilise divers degrés d'agrégation de données (DOA) au niveau des nœuds de transmission conformément aux modèles de trafic locaux actuels. AIDA sépare les décisions d'agrégation des spécificités de l'application en

effectuant une agrégation adaptative dans une couche intermédiaire située entre les protocoles traditionnels de la couche liaison de données et de la couche réseau.

Dans [145] les auteurs ont proposé un protocole basé sur un mécanisme de contrôle de rétroaction simple qui garantit une optimisation distribuée des performances en maintenant des limites de latence acceptables sur la livraison des données avec une consommation d'énergie minimale grâce à un degré d'adaptation d'agrégation des données. Le protocole est conçu de manière à déterminer de manière adaptative à la fois le type et la quantité d'agrégation des données requises afin de répondre aux contraintes de temps. Le système de contrôle décide du type d'agrégation en fonction de la charge du réseau.

Le protocole proposé dans [142] vise principalement à ajuster de manière adaptative le temps d'attente à tous les niveaux de l'arbre d'agrégation des données (DAT) pour enfin optimiser le délai des requête de bout en bout. Pour cela, les auteurs proposent d'utiliser un schéma de contrôle temporel à faible coût pour collecter les données de niveaux faibles dans les plus brefs délais tout en agrégeant les données arrivant du plus grand nombre d'enfants possible.

Le protocole proposé dans [146] vise à résoudre certains problèmes des réseaux de capteurs sans fil centralisés tels que la latence élevée et la précision non optimale, en particulier lorsque de nombreux paquets émergents sont générés en peu de temps. Le protocole propose un modèle de contrôle de rétroaction distingué pour agréger les données détectées sans altérer leur précision. L'objectif est de réduire le délai tout en respectant la précision d'agrégation souhaitée pour les paquets émergents.

Dans [147] les auteurs ont proposé un protocole nommé DRFT (Delay-Ranged Feedback Timing), qui vise à résoudre le problème du temps d'attente des nœuds d'agrégation avant qu'ils ne puissent transmettre leurs données agrégées. De plus, le protocole assure une consommation d'énergie réduite. DRFT est basé sur un réseau de topologie arborescente et est divisé en deux phases : une première phase de calcul de la plage de temps d'attente et une deuxième phase qui ajuste de manière adaptative le temps d'attente.

4.3.1 Modèle proposé

Le modèle du Feedback Control utilisé dans cette approche traite principalement le problème de la planification d'agrégation des données. En effet, malgré le fait que l'agrégation des données soit avantageuse lorsqu'il en résulte moins de paquets de données, permettant par conséquent d'obtenir moins de transmissions sur le réseau, elle est confrontée au problème de l'accumulation des paquets de données au niveau des nœuds d'agrégation avant qu'ils ne procèdent à leur agrégation. Pour cela, le modèle proposé vise à traiter ce problème tout en garantissant un équilibre entre la précision d'agrégation et le délai introduit et ceci par la sélection des nœuds pour lesquels le temps d'attente sera ajusté.

Dans le modèle réseau proposé, lorsque les nœuds Réducteurs reçoivent les paquets de données des Marqueurs correspondants, ils temporisent pour un temps T_i. Une fois le temps expiré, ils agrègent les données et relaient les paquets de données agrégés vers le CH qui performe la même opération avant d'envoyer les données à la station de base.

4.3.1.1 Temps d'attente d'agrégation

La détermination du temps d'attente d'agrégation a lieu après la formation des groupes finaux des Réducteurs/Marqueurs. L'objectif étant d'attribuer des temps d'attente d'agrégation AWT aux nœuds responsables du processus d'agrégation. Le temps d'attente d'agrégation détermine le délai d'attente de chaque réducteur ainsi que du CH avant d'agréger les paquets de données reçus des niveaux inférieurs pour pouvoir les transmettre aux niveaux supérieurs. L'objectif est d'attribuer pour chaque nœud d'agrégation un temps d'attente d'agrégation permettant d'agréger un maximum de paquets de données tout en assurant un délai tolérable pour une précision élevée.

Le calcul du temps d'attente d'agrégation de l'algorithme proposé est donné, pour chaque nœud responsable de l'agrégation, par l'équation suivante :

$$AWT(i) = E(i) \times \left({}^{d(l)}/_{v}\right) \tag{7}$$

Où : $d(l)$ est la distance maximale dont a besoin un paquet de données pour arriver au nœud concerné, v représente la vitesse de propagation (vélocité), et $E(i)$ est l'énergie requise et qui est calculée par l'équation suivante :

$$E(i) = {}^{E(n)}\Big/_{E(T)} \tag{8}$$

Où : $E(n)$ représente l'énergie résiduelle du nœud, et $E(T)$ l'énergie totale.

4.3.1.2 Boucle de contrôle de rétroaction

Le système du Feedback control que nous proposons est basé sur l'utilisation d'une boucle de contrôle de rétroaction permettant l'optimisation du temps d'attente d'agrégation. Dans le modèle de rétroaction proposé, deux paramètres sont impliqués. Ces paramètres sont utilisés afin de contrôler le processus d'agrégation des données :

- Le processus d'agrégation des données est équilibré d'un cycle d'agrégation à un autre par les quantités de paquets de données délivrés aux nœuds d'agrégation, et ceci en fonction de leur degré d'agrégation des données A_{degree}. Ce paramètre affecte directement le temps d'attente d'agrégation.

  Le degré d'agrégation des données A_{degree} est définit par le nombre de paquets de données reçus et agrégés par les nœuds d'agrégation par rapport à la capacité d'agrégation

du nœud. Le degré d'agrégation des données est calculé dans chaque nœud d'agrégation en utilisant l'équation suivante :

$$A_{degree} = {A_{cap}}/{P_{reçus}} \quad (9)$$

Où : A_{cap} représente la capacité d'agrégation du nœud et $P_{reçus}$ représente le nombre de paquets de données reçus et agrégés durant le cycle d'agrégation précédent.

Le mécanisme du Feedback Control contrôle le processus d'agrégation AGG_p à l'instant T_i et peut ainsi augmenter ou diminuer de manière adaptative le temps d'attente d'agrégation AWT en fonction du degré A_{degree}.

- Le modèle de rétroaction proposé tend aussi à contrôler le processus d'agrégation des données en introduisant un autre paramètre qu'on appelle le niveau d'agrégation A_{level} qui affecte le temps d'attente d'agrégation et la précision d'agrégation.

  Nous définissons le niveau d'agrégation par le nombre de paquets de données réellement reçus et agrégés par les nœuds d'agrégation par rapport à une échelle prédéfinie qui dépend de l'application. Le niveau d'agrégation est donné par l'équation suivante :

$$A_{level} = {Exp_p}/{P_{reçus}} \quad (10)$$

  Où : Exp_p représente les paquets de données générés attendus.

  Le processus d'agrégation des données est contrôlé au temps T_i en augmentant ou en diminuant de manière adaptative l'AWT en fonction du niveau d'agrégation des données mesuré.

  En se basant sur les deux paramètres, le mécanisme de fonctionnement de la boucle de contrôle rétroactive est expliqué comme suit :

  Au début de chaque nouveau cycle d'agrégation, une évaluation du temps d'attente d'agrégation du cycle précédent est réalisée. Pour cela, le niveau d'agrégation est calculé en premier lieu en utilisant la formule précédemment définie. Si le nombre de paquets de données réellement reçus est inférieur au nombre considéré, le temps d'attente d'agrégation sera augmenté uniquement au niveau des nœuds réducteurs qui répondront au deuxième paramètre de la boucle :

  Le degré d'agrégation est calculé au niveau de chaque nœud d'agrégation. Si le nombre de paquets de données reçus est inférieur par rapport à la capacité du nœud, le temps d'attente sera augmenté pour ce nœud.

Le degré d'augmentation et de diminution du temps d'attente est variable en fonction de l'application et des caractéristiques du réseau.

La figure suivante illustre le modèle de contrôle par rétroaction proposé :

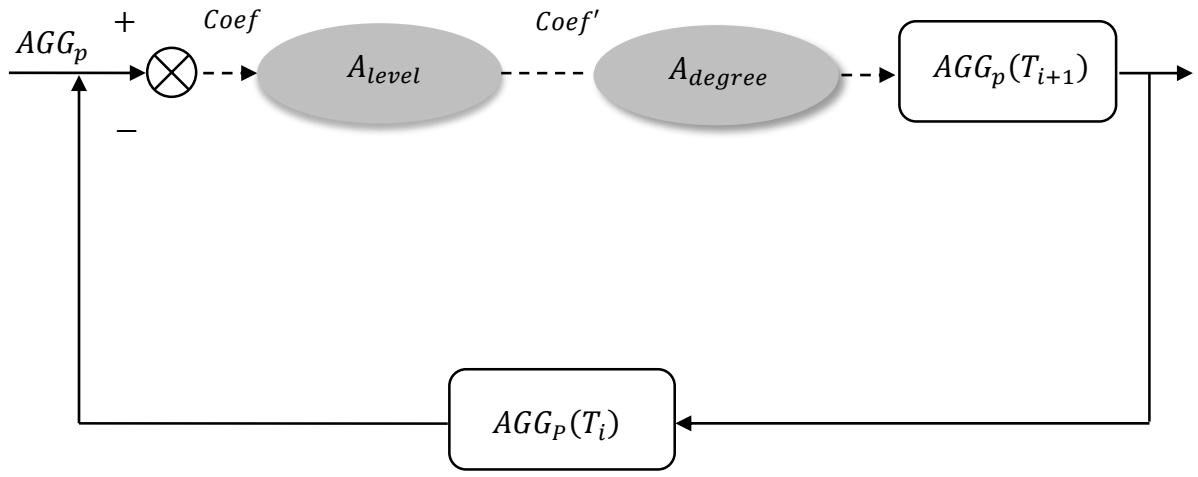


Figure 4-11. Modèle du Feedback Control proposé

Où les coefficients $coef,\ et\ coef'$ représentent l'échelle sur laquelle le contrôleur est basé pour ajuster le temps d'attente d'agrégation AWT.

L'échelle du degré d'agrégation des données est calculée comme suit :

$$coef' = |A_{degree}(T_i) - A_{degree}(T_{i-1})| \quad (11)$$

L'échelle du niveau d'agrégation des données est calculée comme suit :

$$coef = |A_{level}(T_i) - A_{level}(T_{i-1})| \quad (12)$$

Le système converge vers un état stable lorsque les coefficients sont proches ou égaux à 0.

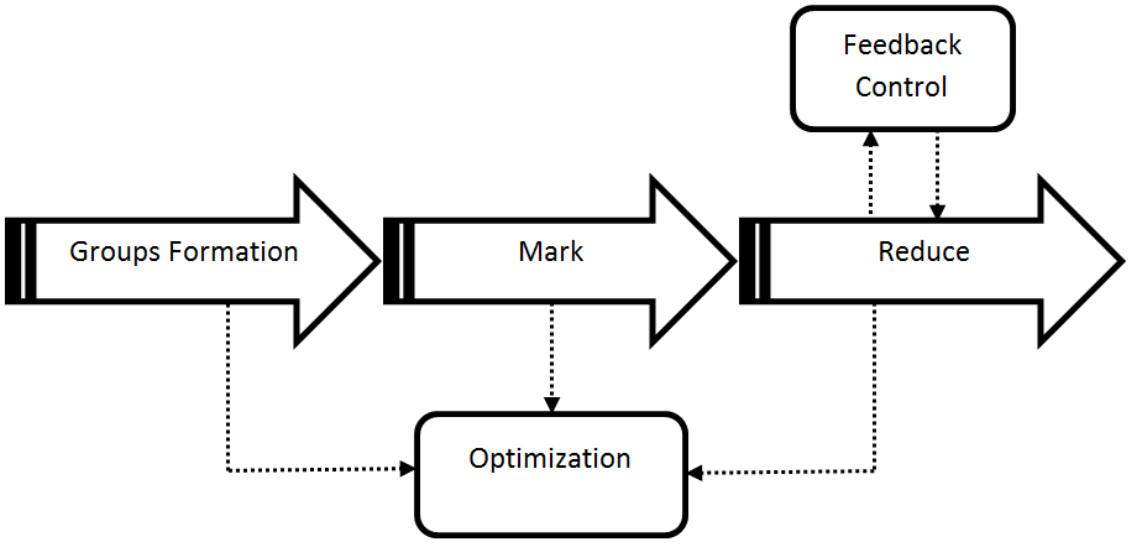


Figure 4-12. Schéma EEMR

5. Conclusion

Nous avons présenté dans ce chapitre un mécanisme d'agrégation des données nommé EEMR (Energy Efficient Mark Reduce mechanism) dont l'objectif principal consiste à prendre en charge le paradigme du Big Data dans les réseaux de capteurs sans fil hétérogènes.

Le mécanisme est basé sur la combinaison de trois principaux défis de la technologie Big Data dans les réseaux de capteurs sans fil qui sont le Clustering, le traitement des données et l'économie d'énergie. Le mécanisme proposé offre des fonctions de traitement importantes qui visent à équilibrer les charges des données sur les nœuds de traitement hétérogènes, et aiguiller les paquets de données reçus vers les nœuds de traitement correspondants, réduisant par conséquent les coûts de l'agrégation des données en termes de consommation énergétique.

L'approche proposée est basée également sur l'utilisation du mécanisme du Feedback Control, qui est introduit afin d'ajuster le temps d'attente d'agrégation des données sur les nœuds hétérogènes d'agrégation en fonction de la capacité du réseau et des nœuds d'agrégation, améliorant par conséquent son efficacité et garantissant un équilibre entre la précision et le délai. Le modèle utilise une boucle de contrôle de rétroaction fermée basée sur des paramètres permettant de contrôler le processus d'agrégation des données, en modifiant d'une manière adaptative le temps d'attente d'agrégation AWT , et par la sélection dynamiques des nœuds pour lesquels le temps d'attente d'agrégation sera ajusté.

Chapitre 5

Evaluation des performances à travers la simulation

Chapitre 5 : Evaluation des performances à travers la simulation

1. Introduction

Dans ce chapitre, les performances du mécanisme EEMR sont évaluées à travers la simulation. Pour cela, EEMR est comparé avec d'autres mécanismes proposés dans la littérature.

Les mesures de performance de notre mécanisme seront basées sur des critères essentiels comme la consommation énergétique, la durée de vie du réseau, la latence, ainsi que la précision d'agrégation.

Le simulateur Cooja de Contiki est utilisé pour exécuter la simulation qui sera basée sur un modèle de simulation et des paramètres réseau présentés ultérieurement.

2. Le simulateur Cooja

La simulation des réseaux de capteurs sans fil nécessite l'utilisation d'outils qui représentent l'un des aspects les plus importants pour le développement des systèmes. Les outils de simulation permettent l'étude des méthodes et des solutions proposées, et d'évaluer leurs performances.

Il existe différents types de simulateurs pour les réseaux de capteurs sans fil [148]. Ces simulateurs sont classés en différentes catégories selon leurs fonctionnalités.

Le simulateur Cooja [149] est l'un des simulateurs dédiés pour les réseaux de capteurs sans fil. Cooja est un simulateur open source, flexible et extensible à tous les niveaux. Il est écrit en langage java, et permet la simulation de capteurs sans fil qui tournent sur les systèmes d'exploitation Contiki [150] et Tinyos [27]. Cooja est le simulateur par défaut du système d'exploitation Contiki. Il utilise différentes plates-formes et possède la possibilité de simuler le niveau de code en utilisant l'émulateur extensible pour les capteurs MSP430, permettant par conséquent l'utilisation de différentes approches de programmation.

Cooja de Contiki est un simulateur de niveau croisé qui représente actuellement une classe très importante de simulateurs des réseaux de capteurs sans fil. Les simulateurs croisés fonctionnent généralement sur trois niveaux d'abstraction :

- Le niveau du réseau ;
- Le niveau du système d'exploitation ;
- Le niveau du jeu d'instructions du code machine.

Cooja offre trois caractéristiques principales :

1. Une interface graphique GUI, basée sur la norme Java Swing toolkit;
2. La possibilité de simuler le support radio sous-jacent aux communications sans fil ;
3. Une architecture extensible, permettant de fournir des fonctionnalités supplémentaires.

Cooja fournit un environnement réel pour la construction des réseaux de capteurs sans fil avec différents types de capteurs. Les solutions proposées peuvent être testées avec différents capteurs tels que Tmote Sky, MicaZ et autres. Le code peut être chargé directement sur les capteurs sans avoir besoin de le modifier le code. De plus, Cooja dispose de différents plug-ins permettant d'obtenir des résultats pertinents et étendre ses fonctionnalités.

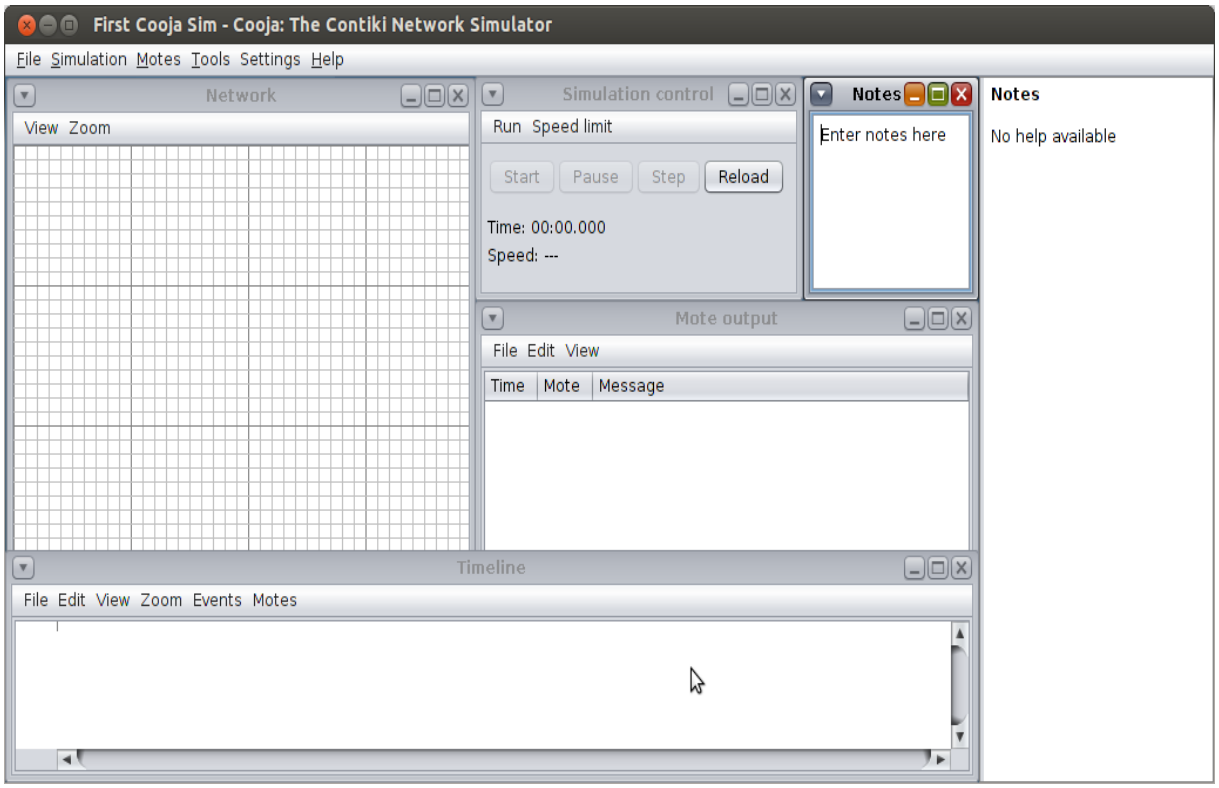


Figure 5-1. Le simulateur Cooja

a. Avantages de Cooja

Cooja est un simulateur qui fonctionne sous Contiki, qui représente un système d'exploitation léger développé essentiellement pour les capteurs sans fil. Cooja présente plusieurs avantages qui incitent à son utilisation :

- Le simulateur Cooja est plus flexible, ce qui signifie que plusieurs parties du simulateur, comme le matériel des capteurs simulés ainsi que les plug-ins, sont remplaçables ;
- Scalabilité ;
- Extensibilité ;

- Efficacité ;
- Flexibilité.

3. Environnement de simulation

La simulation est basée sur l'utilisation en premier lieu de 100 nœuds capteurs déployés de manière aléatoire dans le champ du réseau 100x100 mètres. Le nombre de nœuds est ensuite variable en fonction des protocoles auxquels notre mécanisme est comparé. Initialement, les nœuds déployés sont divisés en deux types de nœuds: 80 nœuds capteurs homogènes à énergie limitée de 2J, des capacités mémoire de 128 Ko et de calcul CPU de 4 bits à 8 MHz, et 20 nœuds hétérogènes avec des capacités énergétiques supérieures (20 J), une mémoire (1 Mo) et des capacités de calcul (processeur 1 GHz 32 bits). Le tableau suivant résume les principaux paramètres de simulation utilisés :

Paramètres	Valeurs
Surface initiale du réseau	100 x 100 m
Nombre initial de nœuds hétérogènes	20
Nombre initial de nœuds homogènes	80
Energie initiale des nœuds hétérogènes	20J
Energie initiale des nœuds homogènes	2J
Taille des paquets de données homogènes	64 bytes
Taille des paquets de données hétérogènes	512 bytes
E_{el} (énergie de calcul)	50nJ/bit
Efs (énergie de propagation)	10nJ/bit/m²
E_{DA} (énergie d'agrégation)	5nJ/bit/signal
Modèle de propagation radio	espace libre

Tableau 5-1. Paramètres de simulation

Pour évaluer les performances du mécanisme proposé, les principales métriques d'évaluation dans la technologie Big Data ainsi que les réseaux de capteurs sans fil sont utilisées et les résultats obtenus avec EEMR sont comparés à d'autres mécanismes d'agrégation des données. Le temps de simulation est fixé à 1500 secondes.

La simulation démarre avec 100 nœuds dans le réseau, composés de 20 nœuds hétérogènes et 80 nœuds homogènes. Les nœuds homogènes représentent 80% de la totalité des nœuds, et les nœuds hétérogènes représentent 20% de la totalité des nœuds. Le nombre de nœuds est ensuite augmenté graduellement.

4. Résultats obtenus

4.1 Evaluation de la consommation énergétique

La consommation énergétique est la première mesure d'évaluation et la plus importante, du fait que notre approche proposée vise à être économe en termes de réduction de la consommation énergétique.

La consommation énergétique est directement liée aux processus de Clustering et de traitement des données, ainsi qu'à la transmission des données entre les nœuds. L'optimisation énergétique implique non seulement de réduire la consommation d'énergie, mais également de prolonger au maximum la durée de vie du réseau.

La division du réseau en matrice de nœuds joue un rôle fondamental dans le processus de réduction de la consommation énergétique. En effet le partitionnement de réseau en plusieurs cellules de nœuds représente un élément important influant les distance des transmissions dans le réseau. Aussi, l'organisation des nœuds hétérogènes en groupes de Réducteurs/Marqueurs est directement impliquée dans la réduction de la consommation énergétique du réseau. En effet, le groupement des nœuds permet de réduire considérablement les distances de transmission entre les nœuds homogènes et le reste des nœuds réduisant par conséquent la consommation énergétique impliquée dans la transmission des données.

De plus, les nœuds Marqueurs jouent le rôle de relieurs entre les niveaux bas et haut du cluster, ce qui permet d'équilibrer la distribution des charges de données sur les nœuds, leur permettant par conséquent de consommer un degré d'énergie équilibré. D'un autre point, et comme le nombre de paquets échangés entre les nœuds d'agrégation et la station de base est réduit grâce à la fonction de réduction, qui est exécutée à deux niveaux, le mécanisme EEMR permet de réduire considérablement la consommation d'énergie.

L'énergie totale consommée est évaluée par l'équation suivante :

$$E_{Total} = \sum_{i=1}^{n} (di^2N * di^2BS) * \varepsilon fs * P_e) \quad (13)$$

Où : di^2N est la distance entre les nœuds, di^2BS est la distance entre les nœuds d'agrégation et la station de base, εfs est l'énergie de propagation qui dépend du modèle d'amplificateur de l'émetteur, n représente le nombre de nœuds, et P_e est l'énergie de traitement qui est calculée par l'équation suivante :

$$P_e = A_e * \sum_{i=1}^{n} (M_{ei} * Opt_{ei}) * \sum_{j=1}^{k} R_{ei} \quad (14)$$

Où : M_{ei} est l'énergie consommée lors de l'étape de marquage, Opt_{ei} est l'énergie consommée lors de l'étape de réglage, R_{ei} est l'énergie consommée pendant l'étape de

réduction, n représente le nombre de nœuds hétérogènes dans les groupes Marqueurs/Réducteurs, k est le nombre de nœuds d'agrégation, et A_e est l'énergie d'agrégation calculée par l'équation suivante :

$$A_e = \left(\sum_{i=1}^{k} RA_{ei}\right) + CHA_e \quad (15)$$

Où RA_{ei} représente l'énergie d'agrégation consommée par les Réducteurs, CHA_ereprésente l'énergie d'agrégation consommée par le CH, et k est le nombre de nœuds d'agrégation.

Pour démontrer l'efficacité de notre approche en termes de réduction de la consommation énergétique, EEMR est comparé à d'autres approches proposées dans la littérature [119] [126] [151] [152] [153] [154]. Les figures ci-dessous montrent les résultats de la simulation.

Le protocole EEMR donne de meilleurs résultats en termes d'efficacité énergétique. En effet, EEMR offre une meilleure gestion d'énergie et introduit par conséquent une faible consommation énergétique comparé aux différents protocoles auxquels il est comparé. Ceci revient au fait que le mécanisme proposé est basé sur l'utilisation de fonctions à travers lesquelles les charges importantes des données sont réparties sur les nœuds hétérogènes d'une manière équilibrée, et ceci par la sélection dynamique des nœuds responsables du traitement. En effet, lorsque ces nœuds reçoivent les paquets de données en fonction de leurs réserves énergétiques, ces dernières sont contrôlées périodiquement et leurs charges seront redistribuées vers d'autres nœuds avec plus de capacités équilibrant ainsi la consommation d'énergie sur les nœuds. Aussi, l'application sur deux niveaux de la fonction d'agrégation permet de réduire le nombre important de paquets transmis et par conséquent la consommation d'énergie est réduite.

Afin de confirmer les résultats obtenus, nous avons mené plusieurs simulations, en augmentant graduellement le nombre de nœuds dans le réseau.

En observant les résultats de simulation de la figure 5-2, nous pouvons constater que le protocole EEMR permet une amélioration de la consommation énergétique d'environ 8% comparé au protocole SA lorsque le réseau contient 100 nœuds, et d'environ 32,2% lorsque le réseau contient 700 nœuds.

Les résultats de simulation de la figure 5-3 montrent également la présence d'un écart important dans la consommation énergétique entre le protocole EEMR et le protocole PC.

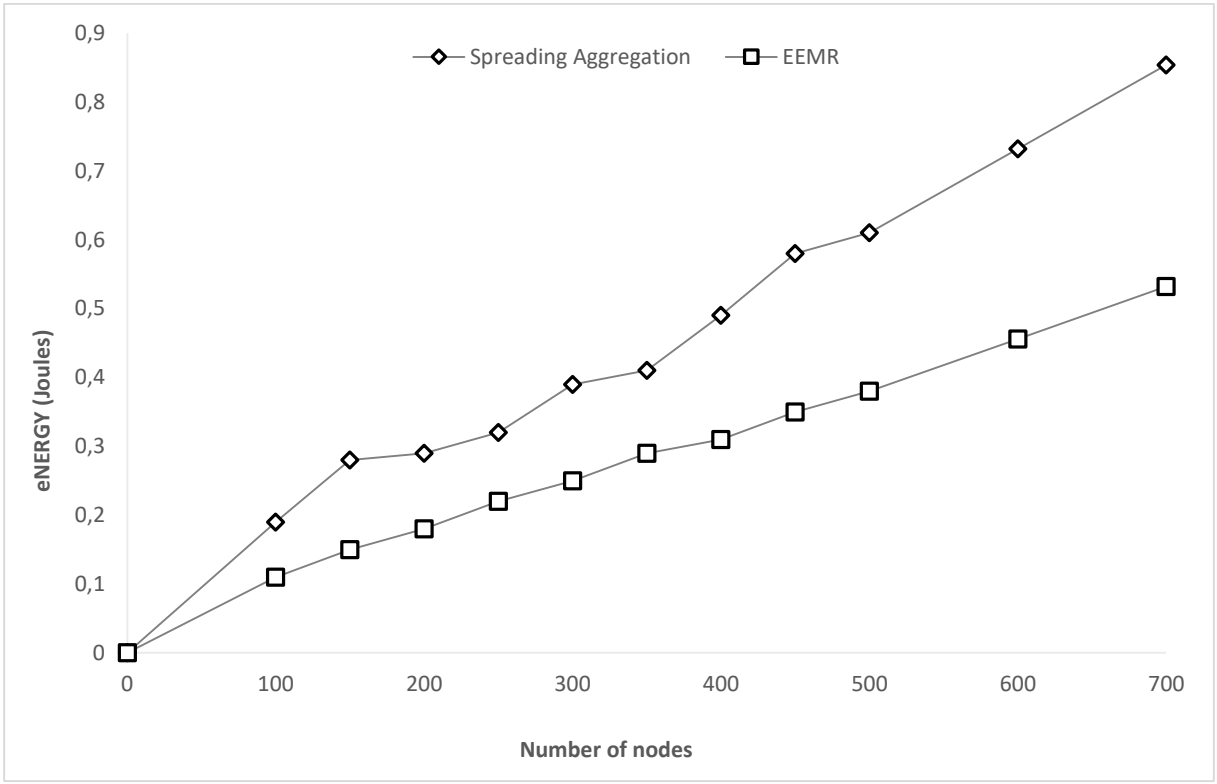


Figure 5-2. Evaluation de la dissipation énergétique : EEMR Vs SA

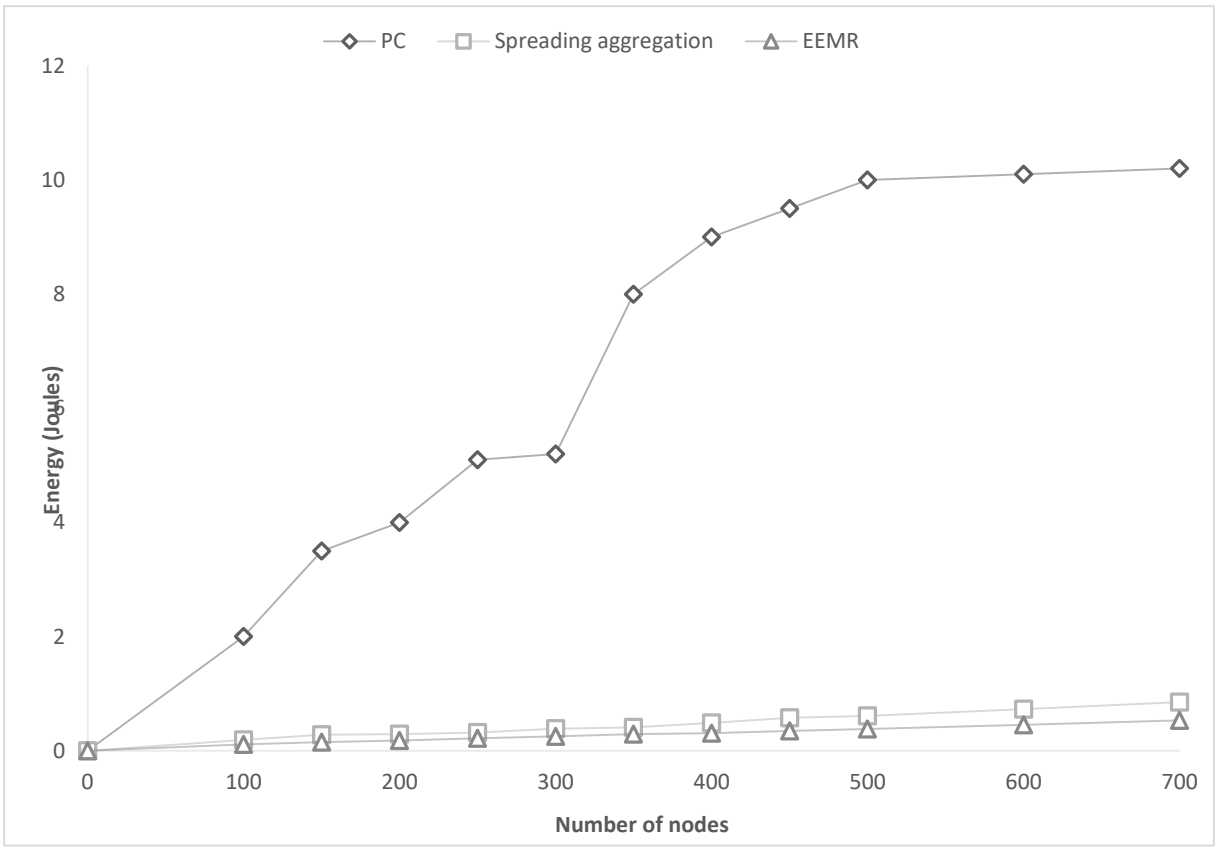


Figure 5-3. Evaluation de la dissipation énergétique : EEMR Vs SA Vs PC

Dans l'objectif de démontrer l'efficacité de l'approche proposée en termes de réduction de la consommation énergétique, nous avons mené plusieurs tours de simulation, et comparé les résultats avec d'autres protocoles. Les résultats obtenus, observés dans la figure 5-4, montrent l'écart enregistré entre EEMR et les autres protocoles. En effet, EEMR présente une meilleure réduction de la consommation d'énergie comparé aux autres protocoles présents dans la littérature.

En effet, dans EEMR la formation des groupes Marqueurs/Réducteur est essentiellement basée sur la position des nœuds. Plus les nœuds sont proches, plus le degré de consommation d'énergie dans le processus de transmission des données est faible. Aussi, la sélection dynamique des nœuds d'agrégation contribue largement à la réduction de la consommation énergétique. D'une autre part, la double agrégation des données au niveau des Réducteurs ainsi que du CH engendre moins de transmissions des données et par conséquent moins de consommation énergétique.

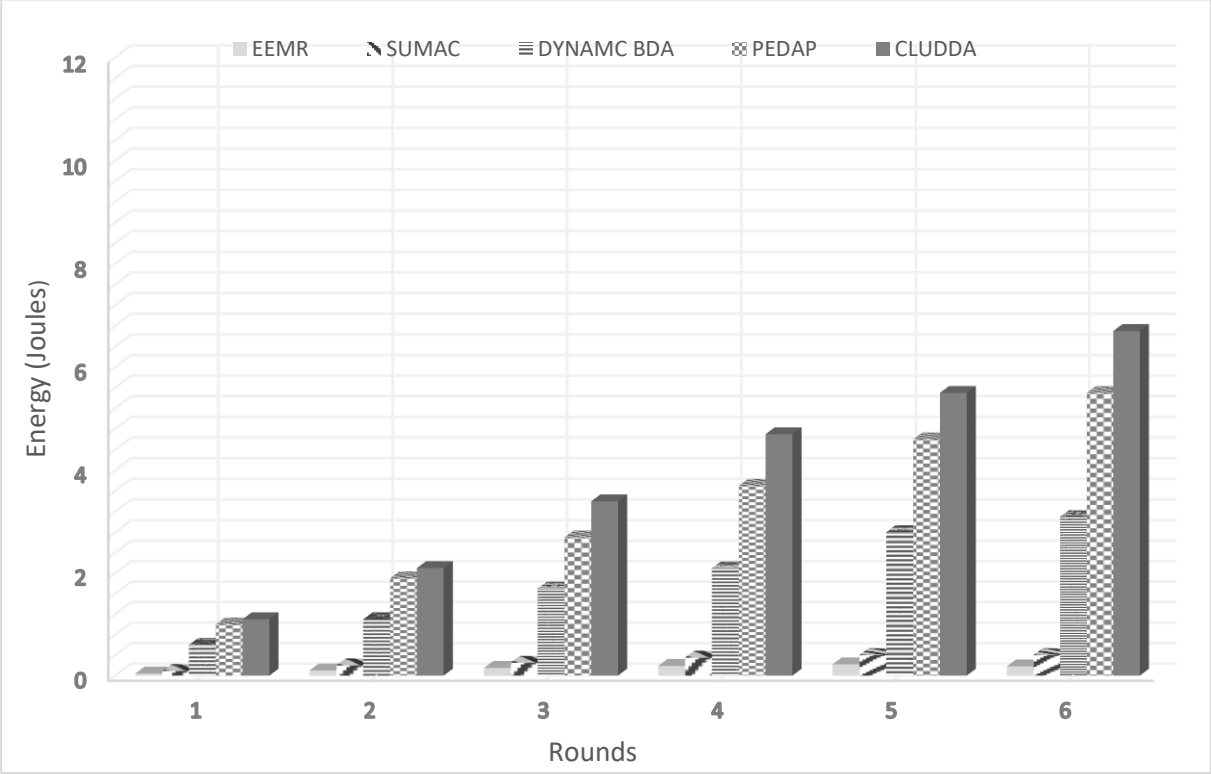


Figure 5-4. Evaluation de la dissipation énergétique : EEMR Vs Dynamic BDA, SUMAC, PUDAP, CLUDDA

4.2 Evaluation de la durée de vie du réseau

La durée de vie du réseau représente une métrique importante dans l'évaluation des performances de l'approche proposée. La durée de vie du réseau est directement liée à la consommation énergétique des nœuds et conséquemment leur durée de vie. En effet, plus la

consommation d'énergie, des nœuds du réseau, diminue plus la durée de vie des nœuds et par conséquent celle du réseau augmente.

Dans notre simulation, nous avons mesuré la durée de vie du réseau, tout au long de la période de simulation en comparaison avec d'autres approches. La figure suivante illustre les résultats obtenus.

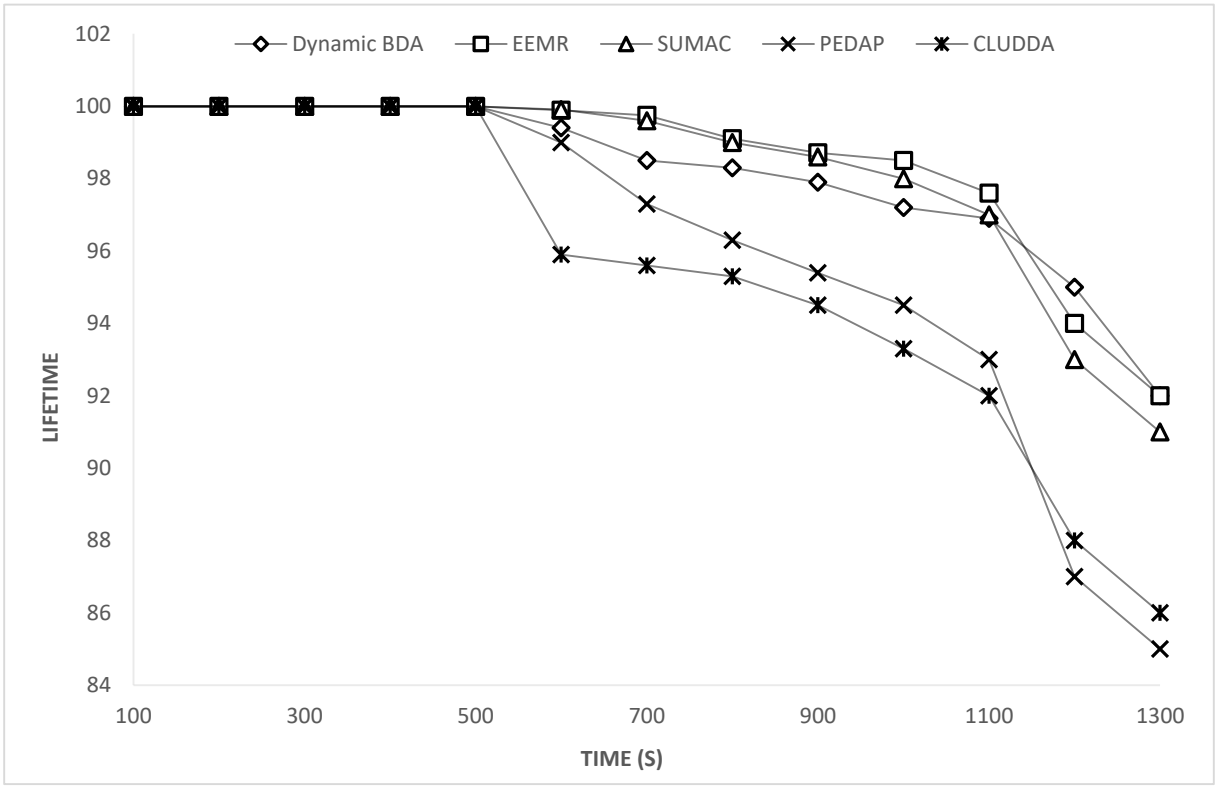


Figure 5-5. Evaluation de la durée de vie du réseau : EEMR Vs Dynamic BDA, SUMAC, PUDAP, CLUDDA

La figure montre clairement que la durée de vie de l'approche EEMR est nettement meilleure que les autres approches auxquelles elle est comparée, et ceci au fur et à mesure que le temps de simulation progresse. Ceci est directement lié à la consommation énergétique réduite des nœuds dans notre approche en raison de la distribution dynamique et contrôlée des paquets de données sur les nœuds responsables du traitement, du mécanisme dynamique de formation des groupes Marqueurs/Réducteurs, et du processus d'agrégation des données à deux niveaux.

4.3 Evaluation de la latence

Une autre métrique importante d'évaluation de l'approche proposée EEMR est l'analyse de la latence ou le délai de bout en bout qui peut être introduit. En effet, la minimisation du délai représente l'un des défis importants dans les réseaux de capteurs sans fil. Le délai et la

précision sont deux métriques directement corrélées. En effet, la minimisation de l'une d'entre elles engendre l'augmentation de la deuxième. La latence affecte directement la transmission de bout en bout et influence le processus d'agrégation des données, car un niveau de latence élevé peut réduire la précision d'agrégation des données.

L'objectif de l'approche EEMR est d'assurer une précision d'agrégation des données élevée tout en garantissant un délai d'agrégation des données tolérable.

Définition

La latence ou délai de bout en bout définie le temps requis pour un nœud pour la transmission d'un paquet de données jusqu'à la station de base. La latence comprend le temps d'attente du nœud avant de procéder à l'envoi du paquet de données et le temps nécessaire pour l'arrivée du paquet à la station de base.

La latence introduite par les nœuds d'agrégation dans l'approche EEMR est calculée par l'équation suivante :

$$Latence(L) = \sum_{i=1}^{n} (AWT_i + T_i) \quad (16)$$

Où :n représente le nombre de nœuds, et T_i représente le temps de transmission.

Pour évaluer la latence introduite dans l'approche EEMR, nous avons mené plusieurs simulations sur des intervalles de temps réguliers. Les résultats obtenus sont représentés dans la figure 5-6.

Selon les résultats de simulation, nous pouvons observer que l'approche EEMR offre de meilleurs résultats en termes de délai de transmission de bout en bout en comparaison avec les autres approches proposées dans la littérature au fur et à mesure que le temps de simulation progresse, en garantissant parallèlement une dépense énergétique minimale. En effet, le mécanisme du feedback control utilisé permet d'adapter et de réguler le temps d'attente d'agrégation des nœuds, qui représente un paramètre important dans l'estimation de la latence, selon le nombre de paquets transmis dans le réseau et reçus par les nœuds d'agrégation.

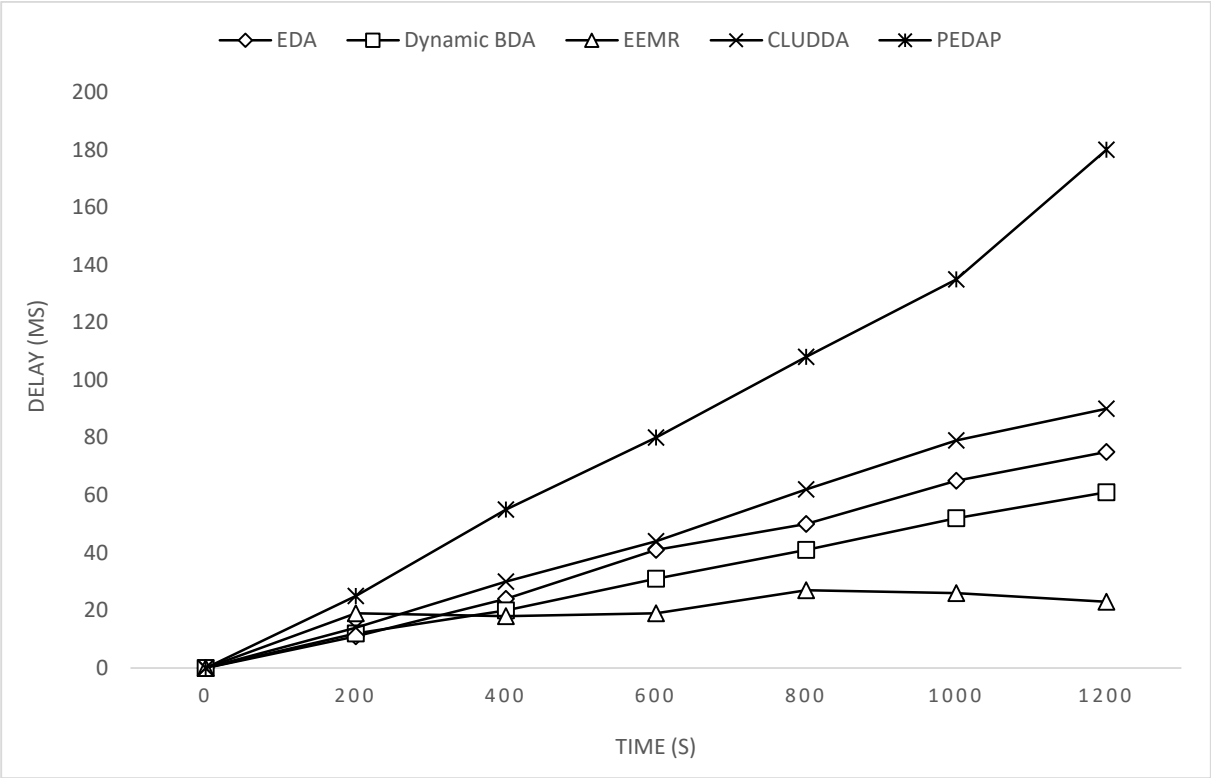


Figure 5-6. Evaluation de la latence

4.4 Evaluation de la précision d'agrégation

Dans la dernière simulation, nous abordons le problème de la précision d'agrégation (Accuracy). La précision d'agrégation est une métrique d'évaluation importante dans le processus d'agrégation des données. La définition de la précision des données dépend de l'application spécifique pour laquelle le réseau de capteurs est conçu. Dans notre approche, la précision d'agrégation représente le nombre de paquets de données agrégés reçus par la station de base par rapport au nombre de paquets de données envoyés à partir des nœuds du réseau.

Lorsque la précision d'agrégation des données est égale à 1, cela signifie que toutes les données détectées sont agrégées. Dans ce cas, le débit du réseau de communication est considérablement réduit, ce qui permet d'optimiser la consommation d'énergie et la bande passante de transmission.

Pour démontrer l'efficacité de notre approche en termes de précision et d'optimisation du processus d'agrégation des données, nous avons comparé EEMR à d'autres protocoles. Le résultat est montré dans la figure suivante :

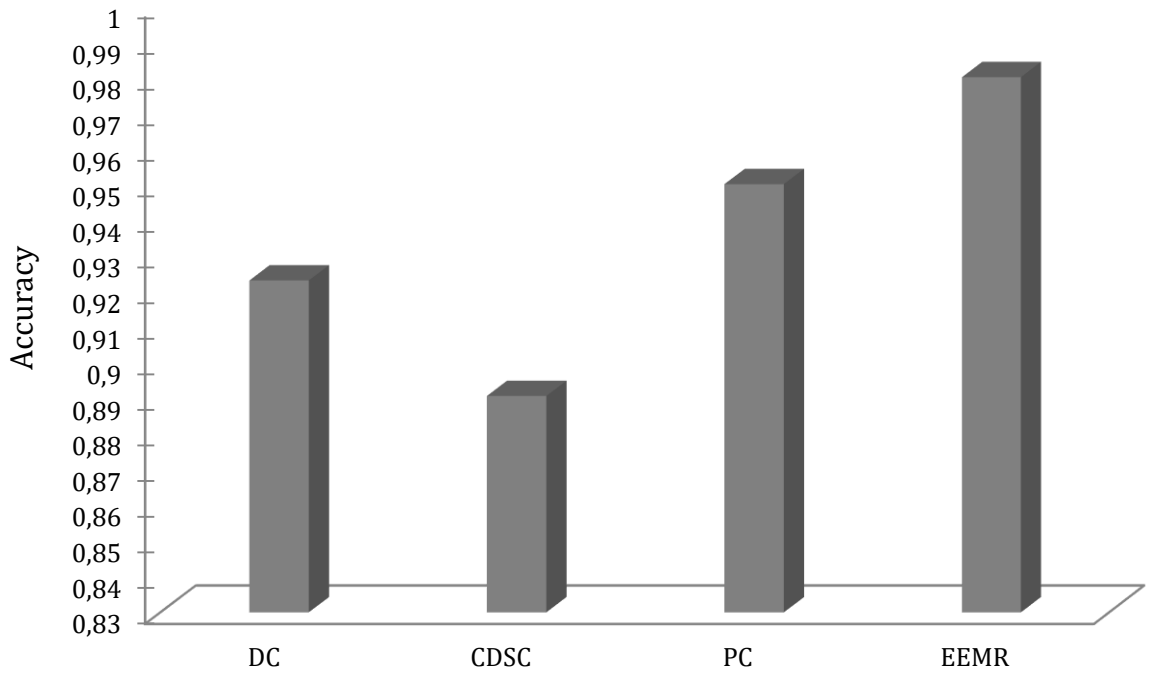


Figure 5-7. Evaluation de la précision d'agrégation

Selon les résultats présentés dans la figure 5-7, nous pouvons constater que notre approche permet d'optimiser la précision de l'agrégation des données. En effet, la précision d'agrégation assurée par EEMR atteint 0.98 comparé aux autres protocoles dont la précision d'agrégation varie entre 0.89 et 0.95. En conséquence, le mécanisme de répartition dynamique des paquets de données sur les nœuds d'agrégation permet non seulement d'assurer une précision élevée de l'agrégation des données, mais aussi de créer une balance entre les différents paramètres d'évaluation des performances de l'approche proposée.

5. Conclusion

Nous avons, à travers ce chapitre, évalué à travers la simulation les performances de notre approche EEMR. Les résultats de simulation montrent que notre approche garantie un équilibre entre les paramètres d'évaluation. En effet, EEMR est économe en énergie et permet de prolonger significativement la durée de vie du réseau, tout en assurant une latence tolérable et une précision d'agrégation élevée. Ceci est justifié par l'utilisation des différentes fonctions de traitement permettant de gérer efficacement l'utilisation des nœuds d'agrégation par la sélection dynamique de ces derniers, ainsi que répartition dynamique des paquets de données sur les nœuds d'agrégation ce qui permet de donner un équilibre à l'ensemble du réseau et d'améliorer par conséquent ses performances.

Conclusion générale

Les réseaux de capteurs sans fil hétérogènes représentent une classe technologique qui prend de plus en plus d'ampleur et qui vise à faire face aux limites des réseaux de capteurs sans fil classiques. En effet, les grandes capacités des nœuds hétérogènes peuvent dépasser de loin celles des nœuds classiques, ce qui fait que cette technologie soit de plus en plus utilisée vus les avantages considérables qu'elle présente comparée à la technologie classique des réseaux de capteurs sans fil.

Les réseaux de capteurs sans fil hétérogènes se sont développés rapidement ces dernières années et leur déploiement représente un avantage pour de nouvelles applications. La grande utilisation des applications des réseaux de capteurs sans fil et particulièrement l'hétérogénéité des nœuds et la diversité des domaines concernés, ont contribué à augmenter le volume des données collectées et traitées. En effet, lorsque les réseaux grandissent et gagnent en volume et en espace de déploiement, les données collectées et traitées croissent de façon exponentielle nécessitant ainsi un traitement efficace, et rendant par conséquent les méthodes de traitement des données traditionnelles difficiles à utiliser.

La technologie Big Data représente une solution efficace pour collecter, analyser, stocker et transmettre des données dans de larges réseaux de capteurs sans fil. En effet, comme les applications de ces réseaux augmentent massivement, les capteurs déployés sont chargés de produire les données en grands volumes, faisant des réseaux de capteurs sans fil des contributeurs clés à la technologie Big Data.

Dans ce travail, nous avons introduit le paradigme du Big Data dans les réseaux de capteurs sans fil, en proposant en premier lieu une nouvelle classification qui combine les défis de ces deux grandes technologies. Notre classification repose sur quatre axes clés qui représentent les principaux piliers considérés dans les réseaux de capteurs sans fil basés sur la technologie Big data. Le Clustering du réseau est le premier pilier dans notre classification et l'étape principale dans la hiérarchie de classification. Le Clustering permet de déterminer principalement l'organisation des nœuds dans le réseau, qui représente un facteur important dans l'application pour laquelle le réseau est dédié. Le deuxième pilier dans notre classification est le traitement des données qui représente un défi crucial nécessitant des stratégies efficaces pour la réalisation des différentes étapes de traitement comme la collecte des données, leur analyse, ainsi que leur stockage. La sécurité, qui représente le troisième pilier dans la classification proposée, joue un rôle indispensable dans les réseaux de capteurs sans fil basés sur la technologie Big Data. Pour cela, plusieurs mécanismes de sécurité adaptés peuvent être introduits afin de protéger les données à tous les niveaux du réseau. Le dernier pilier de la classification proposée est la consommation d'énergie qui représente une métrique importante dans l'évaluation des performances du réseau. Cette métrique est étroitement

liée à tous les piliers de la classification. En effet, un groupement efficace des nœuds ainsi que le déploiement de stratégies adaptées pour le traitement et la sécurité des données permettent de contribuer à la réduction de la consommation énergétique des nœuds du réseau.

Nous nous sommes intéressés par la suite au processus d'agrégation des données, qui représente l'un des principaux défis de traitement des données dans les réseaux de capteurs sans fil basés Big Data. En effet, l'agrégation des données est considérée comme une solution efficace permettant de gérer les ensembles volumineux des données en les combinant, éliminant ainsi le problème de redondance des données et réduisant par conséquent les quantités des données ainsi que la consommation des ressources dans le réseau. Pour cela, nous avons proposé une approche d'agrégation des données nommée EEMR (Energy Efficient Mark Reduce Protocol), inspirée des outils technologiques de traitement des données Big Data. L'approche EEMR implique les principaux challenges de la classification proposée, et vise à optimiser le processus d'agrégation des données volumineuses en réduisant la consommation énergétique du réseau tout en garantissant un délai minimal pour une précision élevée. Pour cela, l'approche EEMR est basée sur l'utilisation de plusieurs fonctions de traitement dont l'objectif est d'optimiser l'organisation des nœuds hétérogènes du réseau par la sélection dynamique des nœuds de traitement, et d'aiguiller les paquets de données sur les nœuds afin d'équilibrer les charges de traitement sur les nœuds hétérogènes.

Nous avons par la suite abordé le problème de planification de l'agrégation des données en proposant un modèle basé sur le mécanisme du Feedback control. Le modèle proposé utilise une boucle de contrôle de rétroaction fermée dans laquelle deux paramètres importants sont impliqués. Ces paramètres visent à équilibrer la charge d'agrégation des données au niveau des nœuds d'agrégation, en contrôlant le processus d'agrégation des données par la modification adaptative du temps d'attente d'agrégation, ce qui contribue largement à la minimisation du délai, et l'augmentation significative de la précision d'agrégation. Les résultats de simulation ont démontré l'efficacité de l'approche proposée en comparaison avec d'autres protocoles proposés dans la littérature.

Comme futurs travaux, nous visons à améliorer notre approche en exploitant la couche MAC, l'objectif étant de réduire davantage le niveau de consommation énergétique du réseau. En effet, une étude des différents protocoles d'accès au média de transmission sera réalisée afin de pouvoir proposer une nouvelle approche permettant de déterminer les périodes de sommeil et d'activation des nœuds afin de les appliquer sur notre approche.

D'un autre côté, d'autres points seront pris en considération, comme la réduction du délai et du temps d'agrégation en respectant la précision d'agrégation souhaitée pour les paquets émergents lorsque de nouveaux événements se produisent.

Aussi, l'approche proposée sera étudiée dans le contexte des réseaux de capteurs sans fil hétérogènes mobiles.

Comme nous avons proposé une nouvelle classification des challenges Big Data dans les réseaux de capteurs sans fil, et que la sécurité représente l'un de ses principaux défis, nous visons à proposer un mécanisme de sécurité pour notre approche.

Une autre perspective importante est l'implémentation de l'approche EEMR sur des capteurs réels, ce qui permettra d'évaluer ses performances dans le monde réel.

Bibliographie

[1] Hailing, C. L. J., Yong, M., Tianpu, L., Wei, L., &Ze, Z. “Overview of Wireless Sensor Networks [J]”. Journal of Computer Research and Development, 2005, vol. 1, p. 021.

[2] Raghavendra, Cauligi S., Sivalingam, Krishna M., et Znati, Taieb (ed.). “Wireless sensor networks”. Springer, 2006.

[3] RAJARAVIVARMA, V., YANG, Yi, et YANG, Teng. “An overview of wireless sensor network and applications”.In: Proceedings of the 35th Southeastern Symposium on System Theory, 2003. IEEE, 2003. p. 432-436.

[4] Matin, M. A., & Islam, M. M. “Overview of wireless sensor network”. Wireless Sensor Networks-Technology and Proto-cols, 2012, p. 1-3.

[5] 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.

[6] Pottie, Gregory J. “Wireless sensor networks”. In : 1998 Information Theory Workshop (Cat. No. 98EX131). IEEE, p. 139-140, 1998.

[7] A. Boukerche. “Heterogeneous Wireless Sensor Networks”, in Algorithms and Protocols for Wireless Sensor Networks. Ottawa, Canada

[8] Ahmed, E., Yaqoob, I., Hashem, I. A. T., Khan, I., Ahmed, A. I. A., Imran, M., &Vasilakos, A. V. “The role of big data analytics in Internet of Things”. Computer Networks, 2017, vol. 129, p. 459-471.

[9] Gandomi, A., &Haider, M. “Beyond the hype: Big data concepts, methods, and analytics”.International journal of information management, 2015, vol. 35, no 2, p. 137-144.

[10] Zikopoulos, P., & Eaton, C. “Understanding big data: Analytics for enterprise class hadoop and streaming data”. McGraw-Hill Osborne Media, 2011.

[11] Harb, H., Idrees, A. K., Jaber, A., Makhoul, A., Zahwe, O., &Taam, M. A. “Wireless sensor networks: A big data source in Internet of Things”. International Journal of Sensors Wireless Communications and Control, 2017, vol. 7, no 2, p. 93-109.

[12] Boubiche, S., Boubiche, D. E., &Azzedine, B. “Integrating Big data paradigm in WSNs”. In: Proceedings of the International Conference on Big Data and Advanced Wireless Technologies. 2016. p. 1-4.

[13] Sundaramurthy, A., &Chitra, V. “Big Data Gathering in Wireless Sensor Network Using Hybrid Dynamic EnergyRouting Protocol”. BEST: International Journal of Management, Information Technology and Engineering (BEST: IJMITE), 2016, vol. 4, no 4, p. 59-68.

[14] Ang, K. L. M., Seng, J. K. P., &Zungeru, A. M. “Optimizing energy consumption for big data collection in large-scale wireless sensor networks with mobile collectors”. IEEE Systems Journal, 2017, vol. 12, no 1, p. 616-626.

[15] Krishnamachari, L., Estrin, Deborah, & Wicker, Stephen. “The impact of data aggregation in wireless sensor networks”. In : Proceedings 22nd international conference on distributed computing systems workshops. IEEE, 2002. p. 575-578.

[16] Callaway JR, Edgar H. “Wireless sensor networks: architectures and protocols”. CRC press, 2003.

[17] Li, Yingshu et Thai, My T. (ed.). “Wireless sensor networks and applications”. Springer Science & Business Media, 2008.

[18] Vujović, Vladimir et Maksimović, Mirjana. “Raspberry Pi as a Wireless Sensor node: Performances and constraints”. In : 2014 37th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO). IEEE, 2014. p. 1013-1018.

[19] VIEIRA, Marcos Augusto M., COELHO, Claudionor N., DA SILVA, D. C., et al. “Survey on wireless sensor network devices”. In : EFTA 2003. 2003 IEEE Conference on Emerging Technologies and Factory Automation. Proceedings (Cat. No. 03TH8696). IEEE, 2003. p. 537-544.

[20] http://www-igm.univ-mlv.fr

[21] Hawi, R. “Wireless Sensor Networks--Sensor Node Architecture and Design Challenges”. International Journal of Advanced Research in Computer Science, 2014, vol. 5, no 1.

[22] Karray, Fatma, Jmal, Mohamed Wassim, Abid, Mohamed, et al. “A review on wireless sensor node architectures”. In : 2014 9th International Symposium on Reconfigurable and Communication-Centric Systems-on-Chip (ReCoSoC). IEEE, 2014. p. 1-8.

[23] Sudevalayam, Sujesha et Kulkarni, Purushottam. “Energy harvesting sensor nodes: Survey and implications”. IEEE Communications Surveys & Tutorials, 2010, vol. 13, no 3, p. 443-461.

[24] Yamada, Toshimi. “Analog-digital converter circuit”. U.S. Patent No 7,372,390, 13 mai 2008.

[25] Xijun, Chen, Meng, MQ-H., et Hongliang, Ren. “Design of sensor node platform for wireless biomedical sensor networks”. In : 2005 IEEE Engineering in Medicine and Biology 27th Annual Conference. IEEE, 2006. p. 4662-4665.

[26] Liu, Xiaolu et Zhou, Shumin. “Evaluation of several time synchronization protocols in WSN”. In : 2010 International Conference of Information Science and Management Engineering. IEEE, 2010. p. 488-491.

[27] Levis, Philip, Madden, Samuel, Polastre, Joseph, et al. “TinyOS: An operating system for sensor networks”. In : Ambient intelligence. Springer, Berlin, Heidelberg, 2005. p. 115-148.

[28] Prasanna, Srinivasa et Rao, Srinivasa. “An overview of wireless sensor networks applications and security”. International Journal of Soft Computing and Engineering (IJSCE), ISSN, 2012, vol. 2231, p. 2307.

[29] Rawat, Priyanka, Singh, Kamal Deep, Chaouchi, Hakima, et al. “Wireless sensor networks: a survey on recent developments and potential synergies”. The Journal of supercomputing, 2014, vol. 68, no 1, p. 1-48.

[30] Sharma, Sukhwinder, Bansal, Rakesh Kumar, et Bansal, Savina. “Issues and challenges in wireless sensor networks”. In : 2013 International Conference on Machine Intelligence and Research Advancement. IEEE, 2013. p. 58-62.

[31] Wu, Chun-Hsien et Chung, Yeh-Ching. “Heterogeneous wireless sensor network deployment and topology control based on irregular sensor model”. In : International Conference on Grid and Pervasive Computing. Springer, Berlin, Heidelberg, 2007. p. 78-88.

[32] Liyang Yu, Neng Wang, Wei Zhang and Chunlei Zheng, “Deploying a Heterogeneous Wireless Sensor Network”, International Conference on Wireless Communications, Networking and Mobile Computing, WiCom 2007, vol., no., 21-25 Sept. 2007, 2588-2591.

[33] S. Rhee, D. Seetharam, and S. Liu, “Techniques for Minimizing Power Consumption in Low Data-Rate Wireless Sensor Networks”, in Proc. of IEEE Wireless Communications and Networking Conference, Atlanta, GA, March, 2004.

[34] Bilel Romdhani. « Exploitation de l’hétérogénéité des réseaux de capteurs et d’actionneurs dans la conception des protocoles d’auto-organisation et de routage”. INSA de Lyon, 2012.

[35] BOUBICHE, Sabrina, BOUBICHE, Djallel Eddine, BILAMI, Azzedine, et al. An outline of data aggregation security in heterogeneous wireless sensor networks. Sensors, 2016, vol. 16, no 4, p. 525.

[36] NAIDJA, Miloud et BILAMI, Azeddine. A dynamic self-organising heterogeneous routing protocol for clustered WSNs. International Journal of Wireless and Mobile Computing, 2017, vol. 12, no 2, p. 131-141.

[37] Syrotiuk, Violet R., Li, Bing, et Mielke, Angela M. “Heterogeneous Wireless Sensor Networks”. 2008.

[38] Zheng, Jun et Jamalipour, Abbas. “Wireless sensor networks: a networking perspective”. John Wiley & Sons, 2009.

[39] Zhuang, Li Qun, Zhang, Jing Bing, Zhang, Dan Hong, et al. “Data management for wireless sensor networks: research issues and challenges”. In : 2005 International Conference on Control and Automation. IEEE, 2005. p. 208-213.

[40] Sagiroglu, Seref et Sinanc, Duygu. “Big data: A review”. In : 2013 international conference on collaboration technologies and systems (CTS). IEEE, 2013. p. 42-47.

[41] Kitchin, Rob. “The data revolution: Big data, open data, data infrastructures and their consequences”. Sage, 2014.

[42] M. Chen, S. Mao, and Y. Liu, “Big Data: A Survey”, Mob. Netw Appl, vol. 19, no. 2, pp. 1–39, 2014.

[43] Jaseena K. U. and Julie M. David, “ISSUES, CHALLENGES, AND SOLUTIONS: BIG DATA MINING”, International journal of Computer Science & Information Technology (CS & IT), Vol. 4, pp. 131–140, 2014.

[44] Halde, S. et Khot, S. “Big data in wireless sensor network: issues & challenges”. Int. J. Adv. Eng. Manag. Sci, 2016, vol. 2, p. 1618-1621.
[45] Ward, Jonathan Stuart et Barker, Adam. “Undefined by data: a survey of big data definitions”. arXiv preprint arXiv:1309.5821, 2013.
[46] Ylijoki, Ossi et Porras, Jari. “Perspectives to definition of big data: a mapping study and discussion”. Journal of Innovation Management, 2016, vol. 4, no 1, p. 69-91.
[47] De Mauro, Andrea, Greco, Marco, et Grimaldi, Michele. “A formal definition of Big Data based on its essential features”. Library Review, 2016.
[48] Dewitt, David J. et Hawthorn, Paula B. “A performance evaluation of database machine architectures”. 1981.
[49] Dewitt, David et Gray, Jim. “Parallel database systems: the future of high performance database systems”. Communications of the ACM, 1992, vol. 35, no 6, p. 85-98.
[50] Walter, T. “Teradata past, present, and future”. UCI ISG lecture series on scalable data management, 2009, vol. 1, no 1, p. 44-48.
[51] Ghemawat, Sanjay, Gobioff, Howard, et Leung, Shun-Tak. “The Google file system”. In : Proceedings of the nineteenth ACM symposium on Operating systems principles. 2003. p. 29-43.
[52] Dean, Jeffrey et Ghemawat, Sanjay. “MapReduce: simplified data processing on large clusters”. Communications of the ACM, 2008, vol. 51, no 1, p. 107-113.
[53] Gantz, John et Reinsel, David. “Extracting value from chaos”. IDC iview, 2011, vol. 1142, no 2011, p. 1-12.
[54] Emmanuel, Isitor et Stanier, Clare. “Defining big data”. In : Proceedings of the International Conference on Big Data and Advanced Wireless Technologies. 2016. p. 1-6.
[55] Beulke, Dave, et al. “Big data impacts data management: The 5 vs of big data”. Available from: Big Data Impacts Data Management: The 5Vs of Big Data, accessed, 2011, vol. 21.
[56] Zhang, Jinson et Huang, Mao Lin. “5Ws model for big data analysis and visualization”. In: 2013 IEEE 16th International Conference on Computational Science and Engineering. IEEE, 2013. p. 1021-1028.
[57] Bhadani, Abhay Kumar et Jothimani, Dhanya. “Big data: challenges, opportunities, and realities”. In : Effective Big Data management and opportunities for implementation. IGI Global, 2016. p. 1-24.
[58] “Welcome to Apache™ Hadoop®!” [Online]. Available: http://hadoop.apache.org/. [Accessed: 29-Dec-2014].
[59] Vavilapalli, Vinod Kumar, Murthy, Arun C., Douglas, Chris, et al. “Apache hadoop yarn: Yet another resource negotiator”. In : Proceedings of the 4th annual Symposium on Cloud Computing. 2013. p. 1-16.
[60] SPARK, Apache. Apache spark. Retrieved January, 2018, vol. 17, p. 2018.
[61] Zaharia, Matei, Xin, Reynold S., Wendell, Patrick, et al. “Apache spark: a unified engine for big data processing”. Communications of the ACM, 2016, vol. 59, no 11, p. 56-65.

[62] Odersky, Martin, Spoon, Lex, et Venners, Bill. “Programming in scala”. Artima Inc, 2008.
[63] Li, Yuliang, Miao, Rui, Liu, Hongqiang Harry, et al. “HPCC: High precision congestion control”. In : Proceedings of the ACM Special Interest Group on Data Communication. 2019. p. 44-58.
[64] Iqbal, Muhammad Hussain et Soomro, Tariq Rahim. “Big data analysis: Apache storm perspective”. International journal of computer trends and technology, 2015, vol. 19, no 1, p. 9-14.
[65] CASSANDRA, Apache. Apache cassandra. Website. Available online at http://planetcassandra. org/what-is-apache-cassandra, 2014, vol. 13.
[66] Huai, Yin, Chauhan, Ashutosh, Gates, Alan, et al. “Major technical advancements in apache hive”. In : Proceedings of the 2014 ACM SIGMOD international conference on Management of data. 2014. p. 1235-1246.
[67] Morzy, Tadeusz et Zakrzewicz, Maciej. “SQL-Like Language for Database Mining”. In : ADBIS. 1997. p. 311-317.
[68] Carbone, Paris, Katsifodimos, Asterios, Ewen, Stephan, et al. “Apache flink: Stream and batch processing in a single engine”. Bulletin of the IEEE Computer Society Technical Committee on Data Engineering, 2015, vol. 36, no 4.
[69] Dillon, Tharam, Wu, Chen, et Chang, Elizabeth. “Cloud computing: issues and challenges”. In : 2010 24th IEEE international conference on advanced information networking and applications. Ieee, 2010. p. 27-33.
[70] Yang, Chaowei, Huang, Qunying, LI, Zhenlong, et al. “Big Data and cloud computing: innovation opportunities and challenges”. International Journal of Digital Earth, 2017, vol. 10, no 1, p. 13-53.
[71] Atzori, Luigi, Iera, Antonio, et Morabito, Giacomo. “The internet of things: A survey”. Computer networks, 2010, vol. 54, no 15, p. 2787-2805.
[72] Dey, Nilanjan, Hassanien, Aboul Ella, Bhatt, Chintan, et al. (ed.). “Internet of things and big data analytics toward next-generation intelligence”. Berlin : Springer, 2018.
[73] Möller, Johanna et Rimscha, M. “(De) centralization of the global informational ecosystem”. Media and Communication, 2017, vol. 5, no 3, p. 37-48.
[74] Landset, Sara, Khoshgoftaar, Taghi M., Richter, Aaron N., et al. “A survey of open source tools for machine learning with big data in the Hadoop ecosystem”. Journal of Big Data, 2015, vol. 2, no 1, p. 24.
[75] Kim, Beom-Su, Kim, Ki-Il, Shah, Babar, et al. “Wireless sensor networks for big data systems”. Sensors, 2019, vol. 19, no 7, p. 1565.
[76] Sivarajah, Uthayasankar, Kamal, Muhammad Mustafa, Irani, Zahir, et al. “Critical analysis of Big Data challenges and analytical methods”. Journal of Business Research, 2017, vol. 70, p. 263-286.
[77] Labrinidis, Alexandros et Jagadish, Hosagrahar V. “Challenges and opportunities with big data”. Proceedings of the VLDB Endowment, 2012, vol. 5, no 12, p. 2032-2033.

[78] Chen, CL Philip et Zhang, Chun-Yang. “Data-intensive applications, challenges, techniques and technologies: A survey on Big Data”. Information sciences, 2014, vol. 275, p. 314-347.

[79] Boubiche, Sabrina, Boubiche, Djallel Eddine, Bilami, Azeddine, et al. “Big data challenges and data aggregation strategies in wireless sensor networks”. IEEE Access, 2018, vol. 6, p. 20558-20571.

[80] Jain, Anil K., Murty, M. Narasimha, et Flynn, Patrick J. “Data clustering: a review”. ACM computing surveys (CSUR), 1999, vol. 31, no 3, p. 264-323.

[81] Harb, Hassan et Abou Jaoude, Chady. “Combining compression and clustering techniques to handle big data collected in sensor networks”. In : 2018 IEEE Middle East and North Africa Communications Conference (MENACOMM). IEEE, 2018. p. 1-6.

[82] Lindsey, Stephanie, Raghavendra, Cauligi, et Sivalingam, Krishna M. “Data gathering algorithms in sensor networks using energy metrics”. IEEE Transactions on parallel and distributed systems, 2002, vol. 13, no 9, p. 924-935.

[83] Russom, Philip, et al. “Big data analytics”. TDWI best practices report, fourth quarter, 2011, vol. 19, no 4, p. 1-34.

[84] Xu, Jinhai, Guo, Songtao, Xiao, Bin, et al. “Energy - efficient big data storage and retrieval for wireless sensor networks with nonuniform node distribution”. Concurrency and Computation: Practice and Experience, 2015, vol. 27, no 18, p. 5765-5779.

[85] Meng, Weizhi, Li, Wenjuan, Su, Chunhua, et al. “Enhancing trust management for wireless intrusion detection via traffic sampling in the era of big data”. Ieee Access, 2017, vol. 6, p. 7234-7243.

[86] Dr. T. AbdulRazak, R. Rajakumar, and M. Rameeja, “Improving Wireless Sensor Network Performances Using Big Data And Clustering Approach”, International Journal of Scientific and Research Publications, vol. 4, pp. 1-7, Aug. 2014.

[87] G. S. Kunal, and Manasa,“An Efficient EM-algorithm for Big data in Wireless Sensor Network using Mobile Sink”, International Journal of Computer Science and Information Technologies (IJCSIT), vol. 7, no. 5, pp. 2201-2205, 2016.

[88] J. Zhou, Y. Zhang, Y. Jiang, C. L. P. Chen, and L. Chen,“A distributed k-means clustering algorithm in wireless sensor networks‟“, in Proc. Int. Conf. Informat. Cybern.Comput. Social Syst (ICCSS), pp. 26-30, 2015.

[89] Doreswamy, G. S. Kunal,“DGC-SOM Clustering algorithm for efficient big data gathering in densely distributed wireless sensor network”, International Journal of Latest Trends in Engineering and Technology Special Issue SACAIM 2017, pp. 040-047, 2017.

[90] L. D. Alahakoon, S. K. Halgamuge, and B. Srinivasan, “A structure adapting feature map for optimal cluster representation,” in Proc. Int. Conf. Neural Information Processing, pp. 809–812, 1998.

[91] D. Alahakoon, “A self-growing cluster development approach to data mining”, in Proc. IEEE Conf. Systems, Man, and Cybernetics, vol. 3, pp. 2901–2906, 1998.

[92] A. S. Pattanshett, and Mr. N D. Kale, “A Survey on Big-Data Gathering using Mobile Collector in Densely Deployed Wireless Sensor Network”, International Journal of Engineering Research & Technology (IJERT), Vol. 3, no. 6, no. 12, Dec. 2014.

[93] D. Takaishi,, et al., “Towards Energy Efficient Big Data Gathering in Densely Distributed Sensor Networks”, Emerging Topics in Computing, IEEE Transactions on, vol. 2, no. 3, pp. 388-397, 2014.

[94] M. Wu, L. Tan, and N. Xiong,“A structure fidelity approach for big data collection in wireless sensor networks,” Sensors, vol. 15, no. 1, pp. 248–273, 2015.

[95] S. Arivoli, and V. Chitra, “Big data gathering in wireless sensor network using hybrid dynamic energy routing protocol”, International Journal of Management, Information Technology and Engineering (BEST: IJMITE), vol. 4, no. 4, pp. 59 – 68, Apr. 2016.

[96] B. Saneja, and R. Rani, “An efficient approach for outlier detection in big sensor data of health care”, International Journal of Communication Systems, vol. 30, no. 17, Nov. 2017.

[97] B. Liu, J. Cao, J. Yin, W. Yu, B. Liu, and X. Fu, “Disjoint multi mobile agent itinerary planning for big data analytics”,EURASIP Journal on Wireless Communications and Networking, vol. 2016, no 1, pp. 99, 2016.

[98] L.Singh, and D.Kumar, “A Big Data Analysis for Risk Identification on Wireless Sensor Network”, World Wide Journal of Multidisciplinary Research and Development, vol. 3, no. 8, pp. 337-343, 2017.

[99] O. Younis, and S. Fahmy,“HEED: A Hybrid, Energy-Efficient, Distributedclustering approach for Ad Hoc sensor networks”, IEEE Transactions on Mobile Computing, vol. 3, no. 4, pp. 366–379, 2004.

[100] H.Kaur, and R.Rjput, “Big Data Analysis on WSN for Risk Analysis on Different Data”, World Wide Journal of Multidisciplinary Research and Development, vol. 3, no. 7, pp. 143-148, 2017.

[101] J.Xu, S. Guo, B. Xiao, and J.He,“Energy-efficient big data storage and retrieval forwireless sensor networks with nonuniform node distribution”, Concurrency and Computation Practice and Experience, vol. 27, no. 18, pp. 5765-5779, 2015.

[102] Al-Karaki, Jamal N., Ul-Mustafa, Raza, et Kamal, Ahmed E. “Data aggregation in wireless sensor networks-exact and approximate algorithms”. In : 2004 Workshop on High Performance Switching and Routing, 2004. HPSR. IEEE, 2004. p. 241-245.

[103] Randhawa, Sukhchandan et Jain, Sushma. “Data aggregation in wireless sensor networks: Previous research, current status and future directions”. Wireless Personal Communications, 2017, vol. 97, no 3, p. 3355-3425.

[104] Handy, M. J., Haase, Marc, et Timmermann, Dirk. “Low energy adaptive clustering hierarchy with deterministic cluster-head selection”. In : 4th international workshop on mobile and wireless communications network. IEEE, 2002. p. 368-372.

[105] Lindsey, Stephanie et Raghavendra, Cauligi S. “PEGASIS: Power-efficient gathering in sensor information systems”. In : Proceedings, IEEE aerospace conference. IEEE, 2002. p. 3-3.

[106] Manjeshwar, Arati et Agrawal, Dharma P. “TEEN: ARouting Protocol for Enhanced Efficiency in Wireless Sensor Networks”. In : ipdps. 2001. p. 189.

[107] Yao, Yong et Gehrke, Johannes. “The cougar approach to in-network query processing in sensor networks”. ACM Sigmod record, 2002, vol. 31, no 3, p. 9-18.

[108] Madden, Samuel, Franklin, Michael J., Hellerstein, Joseph M., et al. “TAG: A tiny aggregation service for ad-hoc sensor networks”. ACM SIGOPS Operating Systems Review, 2002, vol. 36, no SI, p. 131-146.

[109] Beaver, Jonathan, Sharaf, Mohamed A., Labrinidis, Alexandros, et al. “Power-aware in-network query processing for sensor data”. In : Proc. of the 2nd Hellenic Data Management Symposium. 2003.

[110] Yang, Hua, Ye, Fengji, et Sikdar, Biplab. “A dynamic query-tree energy balancing protocol for sensor networks”. In : 2004 IEEE Wireless Communications and Networking Conference (IEEE Cat. No. 04TH8733). IEEE, 2004. p. 1715-1720.

[111] Boubiche, Djallel Eddine et Bilami, Azeddine. “HEEP (Hybrid Energy Efficiency Protocol) based on chain clustering”. International Journal of Sensor Networks, 2011, vol. 10, no 1-2, p. 25-35.

[112] T. Tsai, W.Lan, C. Liu, and M. Sun, “Distributed Compressive Data Aggregation in Large-Scale Wireless Sensor Networks”, Journal of Advances in Computer Networks, vol. 1, no. 4, Dec. 2013.

[113] L. Karim, and M. S. Al-kahtani, “Sensor Data Aggregation in a Multi-layer Big Data Framework”, Information Technology, Electronics and Mobile Communication Conference (IEMCON), pp. 1-7, 13-15 Oct. 2016.

[114] M. S. Al-kahtani,“Efficient Cluster-Based Sleep Scheduling for M2M Communication Network”, Arabian Journal for Science and Engineering, vol. 40, no. 8, pp. 2361-2373, Aug. 2015.

[115] L. Karim, N. Nasser, and T. Salti,“Routing on Mini-Gabriel Graphs in Wireless Sensor Networks”, IEEE WiMob. China, pp. 105-110, Oct. 2011.

[116] L. Karim, N. Nasser, and T. Sheltami, “A fault-tolerant energy efficient clustering protocol of a wireless sensor network”, Wireless Communication and Mobile Computing, vol. 14, no. 2, pp. 175-185, 2014.

[117] W. Heinzelman, A. Chandrakasan, H. Balakrishnan, “Energy-Efficient Communication Protocol for Wireless Microsensor Networks”, Proceedings of the 33rd Hawaii International Conference on System Sciences, vol. 2, pp. 10, Jan. 2000.

[118] L. Cheng, S. Guo, Y. Wang, and Y. Yang,“Lifting Wavelet Compression Based Data Aggregation in Big Data Wireless Sensor Networks”, IEEE 22nd International Conference on Parallel and Distributed Systems China, pp. 561-568, Dec. 2016.

[119] J. Li, S. Guo, Y. Yang, and J. He, “Data Aggregation with Principal Component Analysis in Big Data Wireless Sensor Networks”, 12th International Conference on Mobile Ad-Hoc and Sensor Networks, pp. 45-51, Dec. 2016.

[120] L. G. Rios and J. A. I. Diguez, “Big data infrastructure for analyzing data generated by wireless sensor networks,” In 2014 IEEE International Congress on Big Data, pp. 816–823, Jun. 2014.

[121] D.Wu, B. Yang, and R. Wang,“Scalable privacy-preserving big data aggregation mechanism”,Digital Communications and Networks, vol. 2, no. 3, pp. 122–129, 2016.

[122] S. Din, A. Ahmad, A. Paul, M. M. Ullah Rathore, and J. Gwanggil, “A Cluster-based Data Fusion Technique to Analyze Big Data in Wireless Multi-Sensor System,” IEEE Access, vol.5, pp. 5069-5083, 2017.

[123] D. Bol et al., “Green SoCs for a sustainable Internet-of-Things,” in Proc. IEEE Faible Tension Faible Consommation (FTFC), pp. 1–4, Jun. 2013.

[124] T. Han and N. Ansari, “Heuristic relay assignments for green relay assisted device to device communications,” in Proc. IEEE Global Commun. Conf. (GLOBECOM), pp. 468–473, Dec. 2013.

[125] E. F. Nakamura, A. A. Loureiro, and A. C. Loureiro, “Information fusion for wireless sensor networks: Methods, models, and classifications,” ACM Comput. Surv, vol. 39, no. 3, pp. 9. , 2007.

[126] Merzoug, Mohammed Amine, Boukerche, Azzedine, Mostefaoui, Ahmed, et al. “Spreading Aggregation: A distributed collision-free approach for data aggregation in large-scale wireless sensor networks”. Journal of Parallel and Distributed Computing, 2019, vol. 125, p. 121-134.

[127] Polastre, Joseph, Hill, Jason, et Culler, David. “Versatile low power media access for wireless sensor networks”. In : Proceedings of the 2nd international conference on Embedded networked sensor systems. 2004. p. 95-107.

[128] Ye, Wei, Heidemann, John, et Estrin, Deborah. “Medium access control with coordinated adaptive sleeping for wireless sensor networks”. IEEE/ACM Transactions on networking, 2004, vol. 12, no 3, p. 493-506.

[129] Karim, Lutful, Nasser, Nidal, Abdulsalam, Hanady, et al. “An efficient data aggregation approach for large scale wireless sensor networks”. In: 2010 IEEE Global Telecommunications Conference GLOBECOM 2010. IEEE, 2010. p. 1-6.

[130] Likas, Aristidis, Vlassis, Nikos, et Verbeek, Jakob J. “The global k-means clustering algorithm”. Pattern recognition, 2003, vol. 36, no 2, p. 451-461.

[131] Jain, Anil K. “Data clustering: 50 years beyond K-means”. Pattern recognition letters, 2010, vol. 31, no 8, p. 651-666.

[132] Danielsson, Per-Erik. “Euclidean distance mapping. Computer Graphics and image processing”, 1980, vol. 14, no 3, p. 227-248.

[133] Chang, Dar-Jen, Desoky, Ahmed H., Ouyang, Ming, et al. “Compute pairwise manhattan distance and pearson correlation coefficient of data points with gpu”. In : 2009 10th ACIS International Conference on Software Engineering, Artificial Intelligences, Networking and Parallel/Distributed Computing. IEEE, 2009. p. 501-506.
[134] White, Tom. ‘’Hadoop: The definitive guide’’. " O'Reilly Media, Inc.", 2012.
[135] Shim, Kyuseok. “MapReduce algorithms for big data analysis”. Proceedings of the VLDB Endowment, 2012, vol. 5, no 12, p. 2016-2017.
[136] Dean, J., &Ghemawat, S. “MapReduce: a flexible data processing tool”. Communications of the ACM, 2010, vol. 53, no 1, p. 72-77.
[137] Jung, I. Y., Kim, K. H., Han, B. J., &Jeong, C. S. “Hadoop-based distributed sensor node management system”. International Journal of Distributed Sensor Networks, 2014, vol. 10, no 3, p. 601868.
[138] R. Lammel, “Google‟s mapreduce programming model – revisited”, vol. 70, no. 1, pp. 1-30. 2008.
[139] Franklin, G. F., Powell, J. D., Emami-Naeini, A., & Powell, J. D. “Feedback control of dynamic systems”. Reading, MA : Addison-Wesley, 1994.
[140] Dorf, R. C., & Bishop, R. H. “Modern control systems”. Pearson, 2011.
[141] Donald Christiansen, Ronald K. Jurgen, and Donald G. Fink.” CONTROL SYSTEMS”. The Electronics Engineers' Handbook, 5th Edition McGraw-Hill, Section 19, pp. 19.1-19.30, 2005.
[142] H. Fei, C. May, and C. Xiaojun “Data Aggregation in Distributed Sensor Networks: Towards An Adaptive Timing Control”. In Proceeding of the 3rd International Conference on Information Technology: New Generations (ITNG’06), Las Vegas, NV, Apr. 2006, pp. 256-261.
[143] Boubiche, D.E., D Boubiche, S., Bilami, A. and Toral H., “Feedback Control for Data Aggregation in Wireless Sensor Networks: A Survey”, Networking and Electronic Commerce Research Conference (NAEC 2014), pp.155-162, Trieste, Italy, 21-24 August 2014.
[144] Tian He, Brian M. Blum, John A. Stankovic and Tarek Abdelzaher. “AIDA: Adaptive Application-Independent Data Aggregation in Wireless Sensor Networks”. ACM Transactions on Embedded Computing Systems, Vol. 3, No. 2, May 2004, Pages 426–457.
[145] Tarek Abdelzaher, Tian He, John Stankovic. “Feedback Control of Data Aggregation in Sensor Networks”. 43rd IEEE Conference on Decision and Control December 14-17, 2004 Atlantis, Paradise Island, Bahamas.
[146] Peng Shao-liang, Li Shan-shan, Peng Yu-xing, Zhu Pei-dong, and Xiao Nong. “A Delay Sensitive Feedback Control Data aggregation Approach in Wireless Sensor Network”. ICCS 2007, Part IV, LNCS 4490, pp. 393-400, 2007. Springer-Verlag Berlin Heidelberg 2007.

[147] Zhikui Chen, Song Yang, Xiaodi Huang, Yang Liu. “An adaptive Feedback Timing Control Algorithm of delay-constrained Data Aggregation in Wireless Sensor Networks”. Cyber-Enabled Distributed Computing and Knowledge Discovery, 2009. CyberC '09. International Conference.

[148] Imran, Muhammad, Said, Abas Md, et Hasbullah, Halabi. “A survey of simulators, emulators and testbeds for wireless sensor networks”. In: 2010 International Symposium on Information Technology. IEEE, 2010. p. 897-902.

[149] Velinov, Aleksandar et Mileva, Aleksandra. “Running and testing applications for Contiki OS using Cooja simulator”. 2016.

[150] Dunkels, Adam, Gronvall, Bjorn, et Voigt, Thiemo. “Contiki-a lightweight and flexible operating system for tiny networked sensors”. In: 29th annual IEEE international conference on local computer networks. IEEE, 2004. p. 455-462.

[151] Al-Kahtani, Mohammed S. et Karim, Lutful. “Dynamic Data Aggregation Approach for Sensor-Based Big Data”. International Journal of Advanced Computer Science and Applications, 2018, vol. 9, no 7, p. 62-72.

[152] Jurdak, R., Nafaa, A., & Barbirato, A. “Large scale environmental monitoring through integration of sensor and mesh networks”. Sensors, 2008, vol. 8, no 11, p. 7493-7517.

[153] Tan, Hüseyin Özgür et Körpeoğlu, Ibrahim. “Power efficient data gathering and aggregation in wireless sensor networks”. ACM Sigmod Record, 2003, vol. 32, no 4, p. 66-71.

[154] Chatterjea, Supriyo et Havinga, Paul. “A dynamic data aggregation scheme for wireless sensor networks”. Proc. Program for Research on Integrated Systems and Circuits, Veldhoven, The Netherlands, 2003, p. 924-935.