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

**Université de Batna**
**Faculté des Sciences**

# Thèse

*En vue de l'obtention du diplôme de*
**Doctorat en Sciences en Informatique**

## Une approche Inter-Couches (cross-layer) pour la Sécurité dans les R.C.S.F

*Présentée Par*
**Boubiche Djallel Eddine**

**Membres du jury :**

| | | |
|---|---|---|
| *Président :* | *ZIDANI Abdelmadjid,* | *Prof. Université de Batna.* |
| *Rapporteur :* | *BILAMI Azeddine,* | *Prof. Université de Batna.* |
| *Examinateurs :* | *BENMOHAMMED Mohamed,* | *Prof. Université de Constantine.* |
| | *CHAOUI Allaoua,* | *Prof. Université de Constantine.* |
| | *CHIKHI Salim,* | *Prof. Université de Constantine.* |

# *Remerciements*

Je souhaite remercier en premier lieu mon directeur de thèse, *Pr. Bilami Azeddine*, Professeur au département d'informatique de l'université de Batna et dirigeant du laboratoire Informatique LaSTIC pour m'avoir fait l'honneur de m'encadrer durant la réalisation de cette thèse. Je lui suis également reconnaissant pour le temps conséquent qu'il m'a accordé, ses qualités pédagogiques et scientifiques, sa franchise et sa sympathie. J'ai beaucoup appris à ses côtés et je lui adresse ma gratitude pour tout cela.

Je voudrais remercier les membres de jury : *Pr.ZIDANI Abdelmadjid, Pr. BENMOHAMMED Mohamed, Pr.CHAOUI Allaoua, et Pr.CHIKHI Salim,* pour l'intérêt qu'ils ont porté à mon travail.

Je tiens à remercier mes parents et tous les membres de ma famille qui m'ont apporté le soutien et le confort qui m'ont accompagné durant ce parcours.

Ce travail n'a pu atteindre ses objectifs sans la contribution de près ou de loin de plusieurs personnes auxquelles j'adresse mes chaleureux remerciements.

## *Résumé*

Les réseaux de capteurs sans fil *(RCSF)* représentent une technologie émergente qui vise à offrir des capacités innovantes. Leur utilisation ne devrait cesser d'augmenter et ceci dans de nombreux domaines qu'ils soient scientifiques, logistiques, militaires ou encore sanitaires. Cependant, la limitation de ressources des nœuds capteurs constitue une contrainte importante, principalement en termes de sécurité et d'autonomie d'énergie. C'est pourquoi de nombreux travaux portent sur la conception des protocoles de sécurité en faisant un compromis entre le niveau de sécurité et la consommation en ressources, et le développement de protocoles de communication avec une gestion efficace d'énergie. La plupart de ces travaux se basent sur des approches *mono couche*, qui traitent le problème de sécurité et d'économie d'énergie au niveau d'une seule couche du modèle OSI. Ainsi, dans ce travail nous avons exploré les bénéfices de l'approche *Cross-layer* afin de remédier aux limitations des protocoles mono couche. C'est dans ce but que nous avons proposé deux protocoles à base d'architecture Cross-layer, dont le premier consiste en un protocole de communication économique en énergie, tandis que le deuxième est un système de détection d'intrusions à faible consommation d'énergie.

**Mots clés:** RCSF, sécurité, économie d'énergie, approche Cross-layer, système de détection d'intrusions, HEEP.

## *Abstract*

Wireless sensor networks *(WSN)* represent an emergent technology which aims to offer innovating capacities. Their use should not cease increasing and this in many fields such as scientist, logistic, military or medical. One of the weaknesses of wireless sensor networks is the resources limitation which constrains security and energy efficiency. Therefore, many researches focus on designing security protocols by making a trade off between security level and resource consumption, and develop energy efficient communication protocols. However, Most of these works are based on layered approaches, which address security and energy efficiency problems at a single layer of the OSI model. To overcome layered protocols limitations, we have explored in this work the benefits of the *Cross-layer* approach. As results we have proposed two protocols based on Cross-layer architecture, which consist of energy efficient communication protocol, and intrusion detection system with low energy consumption.

**Keywords:** WSN, security, energy efficiency, Cross-layer approach, intrusion detection system, HEEP.

## ملخص

تعتبر شبكات الاستشعار اللاسلكية ( ش.إ.ل) كتكنولوجيا ناشئة تهدف إلى توفير القدرات الابتكارية. وتشهد هذه التكنولوجيا تطورا سريعا بالإضافة إلى استخدامها في العديد من المجالات العلمية, اللوجستية, العسكرية و الصحة. غير أن الموارد المحدودة لهذا النوع من الشبكات تشكل عقبة رئيسية، خاصة في مجال الأمن و الطاقة. لهذا السبب العديد من البحوث تركز على تصميم بروتوكولات أمنية تقترح حلا وسطا بين مستوى الأمن واستهلاك الموارد، ووضع بروتوكولات للاتصال ذات الإدارة الكفؤة للطاقة. بيد أن معظم هذه الأعمال أحادية الطبقة, حيث توفر حلول لطبقة واحدة من طبقات المودي ل OSI. لمعالجة أوجه القصور في البروتوكولات أحادية الطبقة، قمنا في هذا العمل بدراسة فوائد التواصل بين طبقات المودلِ OSI, و اقترحنا بروتوكولين مبنيين على خاصية التواصل بين الطبقات ، حيث أن الأول يتمثل في بروتوكول للاتصال بخاصية التوفير في الطاقة، والثاني هو نظام حماية خاص بكشف التسللات باستهلاك منخفض للطاقة.

**كلمات البحث**: شبكات الاستشعار اللاسلكية, الأمن, الطاقة, خاصية التواصل بين الطبقات, أنظمة كشف التسللات HEEP,.

# Table des matières

**Chapitre 3 :** « ***Les systèmes de détection d'intrusions*** »

# **Partie 2 :** Contributions

## **Chapitre 6 :** « *Le protocole de communication Cross-layer proposé* »



## **Chapitre 7 :** « *Le système de détection d'intrusions Cross-laye proposé* »

**Chapitre 8 :** « ***Évaluation des performances à travers la simulation*** »

# Liste des figures

# Liste des tableaux

# Glossaire des acronymes

- **AODV:** Ad hoc On-Demand Distance Vector
- **AREA-MAC**: An Asynchronous Real-time Energy-efficient and Adaptive MAC
- **AROS:** Asymmetric communication and Routing in Sensor networks
- **ARQ:** Automatic Repeat Request
- **B-MAC**: Berkeley-MAC
- **BS** : Base station
- **CCA:** Clear Channel Assessment
- **CCS:** Concentric Clustering Scheme
- **CDMA** : Code division multiple access
- **CH** : Cluster Head
- **CLEEP**: Cross-Layer Energy-Efficient Protocol
- **CLEOP:** Cross Layer Energy Optimisation Protocol
- **CLIDS:** Cross Layer Itrusion Detection System
- **CLOA:** Cross-layer optimisation agent
- **CoLaNet**: A Cross-Layer Design of Energy-Efficient Wireless Sensor Networks
- **CTS:** Clear To Send
- **D-MAC:** Data gathering MAC
- **DoS:** Denial of Service
- **DRAND**: Distributed Randomized TDMA Scheduling For Wireless Adhoc Networks
- **DS-PEGASIS:** Diamond-Shaped PEGASIS
- **DSR:** Dynamic Source Routing
- **DS-SS** : Direct-sequence spread spectrum
- **ECC**: Elliptic Curve Cryptography
- **ECSA:** efficient Cluster-based Self-organization Algorithm
- **eHIP**: Energy-efficient Hybrid Intrusion Prohibition system
- **E-MAC:** Event MAC
- **FHSS:** Frequency-hopping spread spectrum
- **FRTS:** Future Request To Send
- **GAF** : Geographic adaptive fidelity
- **G-CLIDA:** Global Cross-Layer Intrusion Detection Agent
- **GEAR** : Geographic and energy-aware routing
- **GPS** : système de positionnement global
- **HCR :** Hierarchical cluster-based routing technique
- **HEEP:** Hybrid Energy Efficiency Protocol
- **HID:** A host-based intrusion detection
- **HSRBH**: Hierarchical secure routing protocol called
- **L-CLIDA:** Local Cross-Layer Intrusion Detection Agent
- **LEA:** Low Entropy Authentication
- **LEACH** : Low Energy Adaptive Clustering Hierarchy
- **LEACH-C** : LEACH-Centralisé

- **LEACH-F** : LEACH avec grappes fixes
- **LEAP:** Localized Encryption and Authentication Protocol
- **LIDC:** Intrusion Detection Component
- **LMA:** Local Mean Algorithm
- **L-MAC**: Lightweight-MAC
- **LMN:** Local Mean of Neighbours Algorithm
- **LPL:** Low Power Listening
- **MECN** : Minimum energy communication network
- **MiniSec:** Mini security
- **MTRP**: Multicast Tree Assisted Random Propagation
- **NAV** : Network Allocation Vecteur
- **NS** : Network simulator
- **OFDM**: Orthogonal Frequency Division Multiplexing
- **PARS:** Power Aware Random Scheduling
- **PEGASIS** : Power-Efficient Gathering in Sensor Information Systems
- **PET:** Personalized Trust model
- **PIDC:** Packet based Intrusion Detection Component
- **P-MAC:** Pipeline Mac
- **Power-aware** : Consommation d'énergie minime
- **RCSF** : Réseau de capteurs sans fil
- **RPT:** Regular and Predictable Times
- **RSSI :** Received Signal Strength Indicator
- **RTS:** Request To Send
- **SDI :** Système de Détection d'Intrusion
- **SHELL**: Scalable, Hierarchical, Efficient, Location aware, and Light-weight
- **S-MAC:** Sensor-MAC
- **SMECN** : Small MECN
- **SPEAR:** Sensor Protocol for Energy Aware Routing
- **SPIN**: Protocoles de capteurs pour l'information par négociation
- **SPIN:** Security Protocols for Sensor Networks
- **STEM:** Sparse Topology and Energy Management
- **TCL** : Tool Command Language
- **TDMA** : Time division multiplexed access
- **TID :** Table d'Informations de Détection
- **TinySec:** Tiny security
- **T-MAC**: Timeout-MAC
- **WEP**: Wired Equivalent Privacy
- **WSN** : Wireless sensors networks
- **Z-MAC**: Zebra-MAC
- **µTESLA:** µTimed Efficient Streaming Loss-tolerant Authentication

# INTRODUCTION GENERALE

Le besoin effréné d'informations et l'évolution rapide de la micro-électronique et des technologies sans fil, ont permis la création de petits appareils électroniques avec un coût très réduit *(ressources limitées)*, capables de collecter et de traiter l'information d'une manière autonome et flexible. Ces appareils peuvent être interconnectés et déployés à grande échelle, donnant naissance à un nouveau type de réseaux nommé réseau de capteurs sans fil *(RCSF)*. Le développement des RCSFs était originalement motivé par les applications militaires *(surveillance des champs de bataille, localisation de l'ennemie…).* Néanmoins, leurs performances remarquables en termes de fiabilité et de faible coût ont permis de proliférer leur utilisation dans le domaine d'application civil *(surveillance d'environnement, l'industrie, la domotique, la santé…)*.

Les réseaux de capteurs sans fil sont conçus pour fonctionner en groupe et coopérer afin de transmettre les données collectées à un point central appelé station de base ou sink. Chaque nœud capteur est équipé d'un microprocesseur à faible puissance de calcul, d'une petite batterie, d'une antenne radio et d'un ou de plusieurs capteurs. Ainsi, les RCSFs doivent opérer en prenant toujours en compte leur limitation de ressources. Ces derniers sont le plus souvent déployés aléatoirement dans des zones hostiles et inexplorées, et doivent s'auto-organiser à l'aide des communications sans fil. La station de base est le seul lien avec le monde extérieur et dispose de plus de ressources par rapport aux nœuds capteurs. Le réseau de capteurs joue le rôle d'un pont entre le monde physique et le système informatique, en fournissant des mesures et des propriétés physiques du monde réel.

L'économie d'énergie représente l'un des grands défis à soulever pour le bon fonctionnement des réseaux de capteurs. En effet, les nœuds capteurs sont généralement alimentés au moyen d'une petite batterie limitée en puissance, et le replacement de celle-ci est une tâche très difficile voire impossible. Par conséquent, l'épuisement des réserves d'énergie des nœuds capteurs implique la mise hors service du réseau tout entier. La sécurité représente un autre défi très important pour les RCSFs, étant donné que des décisions stratégiques peuvent être prises en se basant sur les informations reçues par les nœuds capteurs. Comme la plupart des réseaux distribués, les RCSFs sont exposés aux menaces de sécurité. En outre, leurs caractéristiques spéciales les rendent très vulnérables aux attaques malicieuses. En effet, les RCSFs sont généralement déployés dans des zones inconnues sans aucune protection physique, ce qui facilite leur capture et compromission. De plus, l'environnement de communication sans fil permet d'écouter et d'espionner le trafic échangé dans le réseau, ce qui ouvre l'horizon pour lancer plusieurs types d'attaques. De l'autre côté, la limitation des ressources des nœuds capteurs rend inappropriée l'application des solutions de sécurité classiques. Ainsi, en plus d'offrir un bon niveau de sécurité, les protocoles de sécurité dédiés aux RCSFs doivent respecter les contraintes de ressources de ces derniers.

L'économie d'énergie est considérée comme la principale contrainte à prendre en compte, et elle est étroitement liée au concept de sécurité. En effet, le concept de sécurité peut être formulé d'une

autre manière dans les RCSFs, dans lequel on doit sécuriser le réseau contre les attaques externes et internes, et contre les défaillances des ressources *(épuisement des ressources d'énergie)*. Ainsi, on doit concevoir des protocoles de sécurité en faisant un compromis entre le niveau de sécurité et la consommation en ressources *(sécurité contre les attaques internes et externes)*, et proposer des protocoles de communication avec une gestion efficace d'énergie *(sécurité des ressources)*. Plusieurs recherches ont été conduites afin de proposer des protocoles de sécurité avec une gestion efficace d'énergie. Ces derniers consistent le plus souvent à offrir des solutions mono couche, qui traitent le problème de sécurité et d'économie d'énergie au niveau d'une seule couche du modèle OSI. En conséquent, une solution dédiée par exemple à la couche réseau, ne peut pas protéger le réseau contre les menaces qui opèrent au niveau de la couche liaison. Cela rend ces solutions moins efficaces, d'où la nécessité de les combiner afin d'avoir une solution optimale. Cependant, le problème de consommation d'énergie va en ressurgir à nouveau, étant donné que les solutions combinées opèrent d'une manière indépendante. Par conséquent, nous sommes contraints à explorer de nouveaux horizons de recherche.

Nous considérons que l'approche Cross-layer (inter couches) est l'une des solutions les plus prometteuses en termes de sécurité et d'économie d'énergie. Cette approche consiste à concevoir des protocoles à base d'architecture Cross-layer, faisant interagir plusieurs couches de la pile protocolaire. En se basant sur ce principe, on peut développer des protocoles Cross-layer qui traitent le problème de sécurité et d'économie d'énergie au niveau de différentes couches, tout en respectant les contraintes imposées par les RCSFs. Ainsi, dans le cadre de notre travail, nous avons proposé un nouveau protocole de communication Cross-layer qui se base sur l'interaction des trois couches adjacentes réseau, liaison et physique. De plus, un nouveau type de systèmes de détection d'intrusions a été introduit afin de sécuriser le réseau contre les attaques malicieuses qui peuvent cibler plusieurs couches du modèle OSI.

Notre travail est organisé en deux parties, la première consiste à présenter le domaine de recherche, tandis que la deuxième est consacrée à nos contributions en termes d'économie d'énergie et de sécurité. La première partie est structurée en cinq chapitres. Le premier chapitre est destiné à introduire les réseaux de capteurs sans fil. Ensuite, le problème de sécurité va être abordé dans le deuxième chapitre. Le troisième chapitre met le point sur les systèmes de détection d'intrusions dédiés aux réseaux de capteurs. Dans le quatrième chapitre, nous discutons le problème d'économie d'énergie. Enfin, le concept d'architecture Cross-layer sera présenté dans le chapitre cinq. La deuxième partie de notre travail est aussi divisée en trois chapitres, dont les deux premiers chapitres consistent à présenter respectivement notre protocole de communication Cross-layer, ainsi que le système de détection d'intrusions que nous avons proposé. Le dernier chapitre est consacré à notre étude expérimentale, dans le but d'évaluer les performances de nos contributions.

# Partie 1

## *Introduction sur le domaine de recherche*

# Chapitre 1

## *Généralités sur les réseaux de capteurs sans fil*

# 1. INTRODUCTION

Les réseaux de capteurs sans fil [1] *(RCSFs ou WSNs : Wireless sensor networks)* sont devenus de plus en plus omniprésents. Les milieux scientifiques et industriels leurs prêtent de en plus d'attention du fait de leurs riches applications dans les domaines : médical, commercial et militaire. Selon MIT's Technology Review, il s'agit de l'une des dix nouvelles technologies qui vont influer sur notre manière de vivre et de travailler. Les RCSFs sont des réseaux de nœuds sans fil dédiés à des applications spécifiques. Ils sont considérés comme un type particulier des réseaux Ad-hoc, dans lesquels les nœuds sont très simples et limités en ressources. Les RCSFs sont composés d'un nombre potentiellement très grand de capteurs qui se communiquent selon un modèle de communication « sources multiples - destination unique », déployés dans la zone à couvrir. Chaque capteur est capable d'effectuer d'une manière autonome trois tâches complémentaires, à savoir: mesure d'une valeur physique**,** traitement de ses mesures, et communication par voie hertzienne.

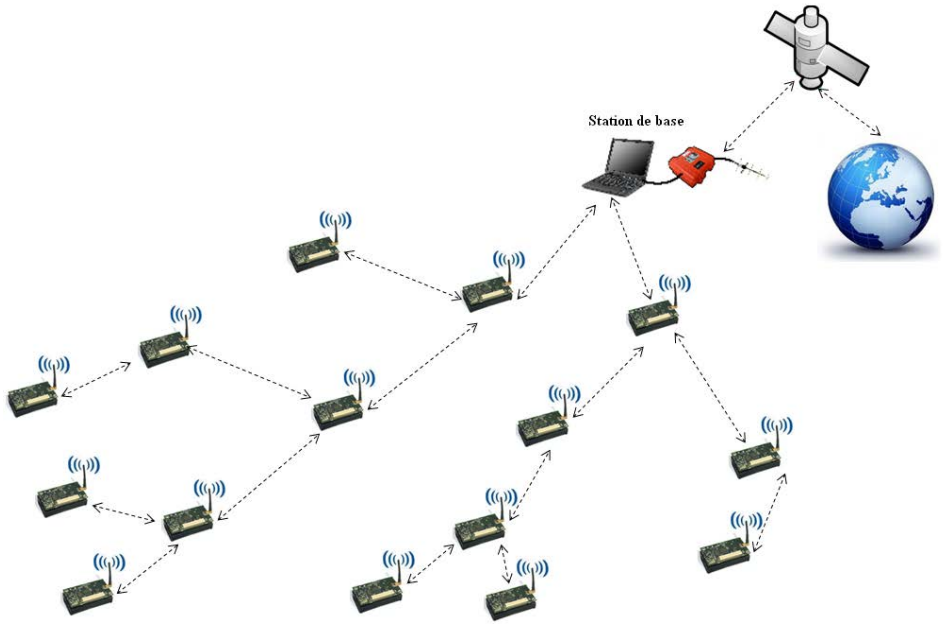



**Figure 1** : Réseau de capteur sans fil (WSN)

Les capteurs sont définis comme étant de petits dispositifs déployés aléatoirement dans une zone géographique appelée champ de captage. Ces derniers sont de capacité de calcul, d'énergie et de bande passante limitées. Le champ de captage définit le terrain d'intérêt pour le phénomène capté. Les données captées sont acheminées grâce à un routage multi-sauts vers un point de collecte appelé nœud puits (sink, ou station de base). La station de base est liée à l'utilisateur du réseau via Internet ou un satellite. Elle permet à l'utilisateur d'établir des requêtes aux autres nœuds du réseau en fournissant le type de données requises et en récoltant les données environnementales captées.

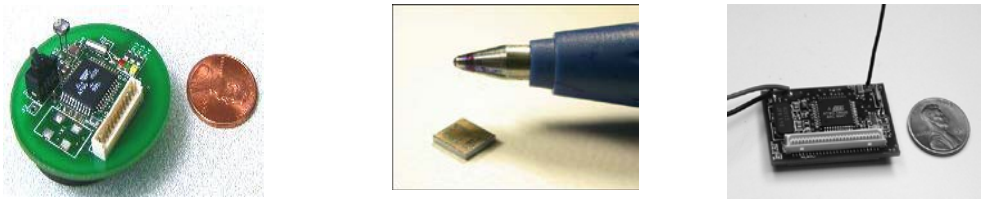


**Figure 2** : Exemple de capteur sans fil.

Les réseaux de capteurs sans fil sont typiquement employés dans les environnements fortement dynamiques et hostiles sans existence humaine (à la différence des réseaux informatiques conventionnels), et donc, ils doivent être tolérants à l'échec (avec une participation humaine minime) et à la perte de connectivité.

## 2. ARCHITECTURE D'UN MICRO CAPTEUR

L'architecture de base d'un nœud capteur est composée de quatre unités à savoir : l'unité de captage, l'unité de traitement, l'unité de transmission, et l'unité de contrôle d'énergie. Selon le domaine d'application, cette architecture peut également contenir d'autres modules, tels qu'un système de localisation *(GPS)*, ou bien un système générateur d'énergie *(cellule solaire)*. Elle peut aussi inclure un système de mobilité chargé de déplacer le micro-capteur dans l'environnement de captage.

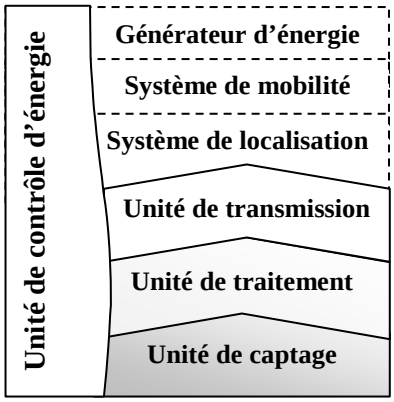



**Figure 3** : Architecture d'un micro capteur.

**2.1. L'unité de captage:** Prend en charge le captage et l'acquisition des données à partir de l'environnement surveillé de l'unité. Elle comprend généralement le dispositif capteur et un convertisseur Analogique/Numérique.

**2.2. L'unité de traitement:** Comme son nom l'indique, l'unité de traitement est responsable de toutes les opérations de calcul dans un nœud capteur. Elle comprend un simple processeur associé à une mémoire de stockage limitée, et fonctionne à l'aide d'un système d'exploitation dédié aux micro-capteurs *(TinyOS par exemple)*.

**2.3. L'unité de transmission:** Cette unité se charge de transmettre ou recevoir les paquets de données en utilisant un dispositif de communication sans fil. Ce dernier constitue généralement une antenne radio à faible portée afin d'économiser l'énergie des nœuds capteurs. L'unité de transmission doit aussi contenir des circuits de modulation, démodulation, filtrage et multiplexage, pour le traitement du signal radio.

**2.4. L'unité de contrôle d'énergie:** A cause de leur taille très réduite et leur faible coût de fabrication, les nœuds capteurs doivent être équipés d'une ressource énergétique à faible autonomie (*généralement une batterie*). De plus, les caractéristiques hostiles de l'environnement du dépoilement rendent généralement ces ressources d'énergie irremplaçables. Dès lors, l'énergie est la ressource la plus précieuse dans un réseau de capteurs puisque elle influe directement sur la durée de vie des micro-capteurs, voire du réseau en entier. Ainsi, l'unité de contrôle d'énergie représente l'une des unités les plus importantes dans un nœud capteur. En effet, celle-ci se charge d'alimenter les autres unités en énergie tout en minimisant la consommation énergétique du nœud capteur. Ainsi, cette unité peut mettre en veille les composants inactifs par exemple, afin de réduire le gaspillage d'énergie.

## 3. LA PILE PROTOCOLAIRE ADOPTEE PAR LES RCSFs

Contrairement aux réseaux Ad-hoc, les réseaux de capteurs sans fil imposent des contraintes supplémentaires aux protocoles de communication. Par conséquent, le modèle traditionnel en couches (modèle OSI), ne répond pas aux exigences de ce type particulier de réseaux. En effet, les RCSFs adoptent une version simplifiée du modèle OSI, à laquelle sont ajoutées de nouvelles couches afin de remédier aux contraintes et aux limitations imposées. Ainsi, le nouveau modèle se compose de 5 couches similaires à celles du modèle OSI (physique, liaison, réseau, transport et application), et trois plans de gestion dédiés pour le contrôle d'énergie, de mobilité et des tâches particulières.

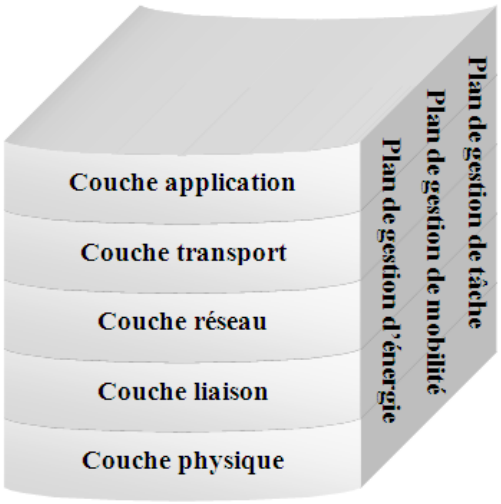



**Figure 4 :** La pile protocolaire dans les RCSFs.

- **La couche physique :** Comme celle du modèle OSI, cette couche est responsable de la modulation, la détection du signal et la sélection des fréquences porteuses.

- **La couche liaison :** Cette couche est chargée du contrôle d'erreurs, du multiplexage des flux de données, et le contrôle d'accès au média de transmission.

- **La couche réseau :** L'objectif de cette couche est de trouver des chemins de routage à faible coût d'énergie pour transmettre les données captées vers la station de base. Ainsi, les protocoles de cette couche doivent toujours prendre en compte les limitations en ressources des nœuds capteurs.

- **La couche transport :** Son rôle est le contrôle du flux, le découpage, l'ordonnancement et le transport des paquets de données, et la gestion des erreurs de transmission.

- **La couche application :** Afin de fournir une interface d'interaction avec l'utilisateur humain, les nœuds capteurs peuvent être dotés d'une couche application, dont le rôle est d'implémenter l'ensemble d'applications et de logiciels d'interaction.

- **Le plan de gestion d'énergie :** Les nœuds capteurs sont sévèrement limités en ressources d'énergie, qui influence directement sur la durée de vie du réseau. Ainsi, le plan de gestion d'énergie doit fournir des mécanismes de gestion efficaces pour réduire le degré de consommation d'énergie, et éliminer les sources de gaspillage de celle-ci.

- **Le plan de gestion de mobilité :** Ce plan est responsable du contrôle du mouvement des nœuds capteurs dans le cas où ils sont mobiles. Il peut par exemple enregistrer les trajectoires d'un nœud capteur afin de l'aider à se localiser.

- **Le plan de gestion de tâche :** Dans un réseau de capteurs, les nœuds peuvent effectuer des tâches qui se diffèrent en termes de consommation de ressources. Ainsi, un plan de gestion de tâche est souvent nécessaire afin de répartir d'une manière équitable les tâches sur les nœuds capteurs, et offrir ainsi une gestion efficace des ressources disponibles.

## 4. DOMAINES D'APPLICATIONS DES RESEAUX DE CAPTEURS

Les caractéristiques particulières des nœuds capteurs *(faible coût, petite taille, communication sans fil…),* ont permit d'étendre rapidement leurs domaines d'application. Parmi les domaines où ces réseaux peuvent offrir les meilleures contributions, nous citons les domaines : militaire, environnemental, domestique, santé, sécurité, etc.

**4.1. Applications militaires :** Les RCSFs ont été initialement conçus pour des projets d'application militaire. En effet, le faible coût, la tolérance aux pannes, l'organisation autonome, et le déploiement rapide, représentent des caractéristiques très attirantes pour ce domaine d'application. Par exemple, les RCSFs, peuvent êtres déployés afin de surveiller les activités des forces ennemies, ou d'analyser le terrain avant d'y envoyer des troupes *(détection d'agents chimiques, biologiques ou de radiations).*

**Figure 5 :** domaines d'application des réseaux de capteurs.

**4.2. Applications à la sécurité :** La sécurité représente un domaine d'application très important pour les RCSFs. En effet, des capteurs peuvent êtres dans les bâtiments afin de détecter les altérations dans leur structure. En outre, Un réseau de capteurs peut constituer un système d'alarme distribué, qui servira à détecter les intrusions sur un large secteur. Un tel système de sécurité sera très robuste étant donné qu'il ne contient pas de point critique pour le déconnecté. Parmi les autres applications de sécurité, on peut citer la surveillance de voies ferrées, pour prévenir des accidents avec des animaux et des êtres humains, ou la détection de fuites d'eau dans les barrages afin d'éviter les dégâts éventuels.

**4.3. Applications environnementales :** Les réseaux de capteurs sans fil peuvent être utilisés afin de surveiller des phénomènes environnementaux. Ainsi, ils sont déployés dans les forêts afin de détecter et de signaler un éventuel début d'incendie. Les capteurs peuvent aussi être semés avec les graines, afin de contrôler l'arrosage des plantes. Dans le domaine industriel, les capteurs sont généralement utilisés afin de détecter des fuites de produits toxiques, ou pour la surveillance des paramètres critiques tels que la température d'un réacteur nucléaire.

**4.4. Applications médicales :** La surveillance des fonctions vitales de l'être humain peut être effectuée avec des micros-capteurs avalés ou implantés sous la peau des malades. Des capteurs peuvent être implantés à l'intérieur du corps humain pour traiter certains types de maladies *(tel que la détection de cancers)* ou pour collecter des informations physiologiques *(tel que la surveillance du niveau de glucose)*, ou encore pour le monitoring des organes vitaux.

**4.5. Applications commerciales :** Des micros-capteurs peuvent être installés dans les produits commerciaux afin de traquer le processus de stockage et de livraison de ces derniers. Dans les immeubles, le système de climatisation peut être conçu en intégrant plusieurs micro-capteurs dans les tuiles du plancher et les meubles. Ainsi, La climatisation pourra être déclenchée seulement aux endroits où il y a des personnes présentes et seulement si c'est nécessaire.

## 5. *BESOINS ET FACTEURS DE CONCEPTION DANS UN RCSF*

Dans ce qui suit, on présentera les besoins de base et les facteurs de conception des réseaux de capteurs sans fil, qui servent de directives au développement des protocoles et des algorithmes dédiés à ce genre particulier de réseau.

**5.1. La tolérance aux fautes, adaptabilité et fiabilité :** les réseaux de capteurs sont requis pour fonctionner en s'adaptant aux changements environnementaux que les capteurs contrôlent. La fiabilité est la capacité de maintenir les fonctionnalités de réseau de capteurs sans la moindre interruption qui sera due à l'échec du noeud capteur. Ce dernier peut échouer en raison du manque d'énergie, de dommages physiques, de problèmes de communication, d'inactivité, ou d'interférence environnementale. Le réseau devrait pouvoir détecter l'échec d'un nœud et *s'organiser*, *se reconfigurer* et *récupérer* des échecs de noeud sans desserrer aucune information.

**5.2. La gestion et consommation d'énergie:** La source d'énergie est l'un des composants les plus importants d'un nœud capteur. Elle peut être généralement représentée par une simple batterie à faible autonomie d'énergie. Au-delà de l'endroit inaccessible avec moins de contrôle et d'existence humaine, les sources d'énergie jouent un rôle critique dans la survie des noeuds capteurs. Ainsi, l'énergie devrait être intelligemment divisée selon le besoin, sur les tâches de captage, de calcul, et de communication. Les capteurs peuvent être mis en veille lorsqu'ils sont inactifs. Un bon nombre de recherches courantes se concentrent sur la conception de protocoles et d'algorithmes power-aware *(consommation d'énergie minime)* pour les réseaux de capteurs sans fil.

**5.3. L'agrégation de données :** Diffuser de grandes quantités de données sur le réseau peut facilement encombrer ce dernier. L'agrégation intelligente des données captées et l'élimination de l'information non désirée et redondante, peut être une solution pour l'utilisation efficace de ressources et d'énergie et l'évitement de congestion.

**5.4. Le routage Intelligent :** Les protocoles de routage doivent être adaptatifs à la flexibilité des RCSFs *(auto-configurant)*. L'information devrait être persistante malgré les changements des nœuds du réseau. En outre, les algorithmes de routage devraient être intelligents pour choisir les sauts et les pats de distance miniums pour le transfert des données avec un faible coût d'énergie.

**5.5. La sécurité :** Pour les applications qui exigent un niveau de sécurité assez élevé telles que les applications militaires, des mécanismes d'authentification, de confidentialité, et d'intégrité doivent être mis en place au sein du réseau. Les algorithmes de cryptographie conçus pour les réseaux de capteurs doivent tenir compte des ressources limitées de ces derniers. De plus, l'absence d'une protection physique des nœuds capteurs ainsi que la nature des liens sans fil, rend le réseau vulnérable aux attaques malveillantes.

**5.6. Le coût de fabrication :** A cause de leur grande échelle, le cout de fabrication d'un nœud capteur doit être très réduit. Ainsi, le coût global du réseau ne doit pas être supérieur à celui d'un réseau classique afin de pouvoir justifier son intérêt.

**5.7. La grande échelle :** Les RCSFs sont généralement déployés avec un grand nombre de capteurs qui peut atteindre le million. Ceci peut engendrer des problèmes de communication et de contrôle qui nécessitent des protocoles capables de les gérer.

## 6. CONCLUSION

Les réseaux de capteurs sans fil présentent un intérêt considérable et une nouvelle étape dans l'évolution des technologies de l'information et de la communication. La flexibilité, la tolérance aux fautes, le prix réduit et les caractéristiques rapides de déploiement des réseaux de capteurs offrent des possibilités infinies de développement dans tous les domaines d'application. Ceci nous permet de penser que les réseaux de capteurs feront bientôt partie intégrante de nos vies et satisferont sûrement les plus grands projets. Nous avons essayé à travers ce chapitre de mettre le point sur l'architecture des RCSFs, ainsi que leurs principaux domaines d'application et leur facteur et défi de conception. Dans le chapitre suivant, nous allons aborder le concept de sécurité dans RCSFs, qui constitue l'une des contraintes majeures qui confrontent le bon fonctionnement de ces derniers.

# Chapitre 2

## *La sécurité dans les RCSFs*

# 1. INTRODUCTION

Les réseaux de capteurs sans fil sont déployés sans infrastructure prédéfinie et laissés généralement sans surveillance. Les caractéristiques inhérentes des réseaux de capteurs sans fil les rendent particulièrement vulnérables aux attaques. Comme les données sont transmises par voie hertzienne, il est extrêmement facile pour un adversaire d'espionner le trafic. Afin de répondre aux strictes exigences budgétaires, les nœuds capteurs ont tendance à ne pas être inviolables et n'offrent ainsi aucune protection contre les attaques de sécurité. En effet, les réseaux de capteurs ne peuvent pas compter sur une intervention humaine pour faire face aux adversaires qui tentent de compromettre le réseau ou d'entraver son bon fonctionnement. De plus, ils ne peuvent pas recourir à des mécanismes de sécurité existants, qui s'avèrent coûteux en calcul. Les objectifs de n'importe quel système de sécurité consistent à assurer la confidentialité, l'intégrité et la disponibilité des données. De plus, l'information doit être protégée contre les accès non autorisés, la divulgation, la perturbation, la modification ou la destruction. Les systèmes à base de fonctions cryptographiques sont généralement utilisés pour assurer la sécurité. Cependant, en raison du manque de mémoire et de puissance *(faible puissance de calcul, réserves énergétiques limitées)* des nœuds capteurs, la plupart de ces approches ne peuvent pas être utilisées directement.

Dans ce chapitre, nous allons donner un aperçu sur la sécurité dans les réseaux de capteurs sans fil. Nous présenterons d'abord les limites des réseaux de capteurs qui rendent la sécurité des données un véritable défi. Ensuite, nous allons identifier les différents types d'attaques qui peuvent cibler ce type particulier de réseaux sans fil. Après, on va discuter les principaux mécanismes de sécurité dédiés aux RCSFs. Enfin, un état de l'art sur les stratégies de défense proposées pour les RCSFs sera présenté.

# 2. LES OBJECTIFS DE SECURITE

La sécurité peut être définie comme la gestion du risque qui menace la confidentialité, l'intégrité, la fraicheur et la disponibilité des données.

**2.1 La confidentialité des données :** la confidentialité constitue l'un des objectifs de sécurité les plus importants dans les réseaux de capteurs. Ce service désigne la garantie que l'information n'a pas été divulguée, et que les données ne sont compréhensibles que par les entités qui partagent le même secret. L'approche standard pour sécuriser l'intégrité des données sensibles consiste à les chiffrer avec une clé publique. Cependant, cette méthode est trop coûteuse pour être utilisée dans les réseaux de capteurs *(contraintes de ressources)*.Par conséquent, la plupart des protocoles de sécurité proposés pour les RCSFs utilisent des méthodes de chiffrement basées sur l'utilisation de clé symétrique [2, 3, 4, 5, 6].

**2.2 L'intégrité :** elle garantit que les données reçues n'ont pas été altérées durant leur transit dans le réseau de manière volontaire ou accidentelle. En fait, on veut éviter qu'un intrus puisse modifier cette information pour en tirer certains avantages. On veut également protéger cette information contre les modifications accidentelles des intervenants légaux car ceci pourrait entrainer plusieurs complications.

**2.3 L'authentification :** consiste à vérifier l'identité authentique des nœuds. En effet, on ne peut assurer la confidentialité et l'intégrité des messages échangés si, dès le départ, on n'est pas sûr de communiquer avec le bon nœud. Si l'authentification est mal gérée, un attaquant peut se joindre au réseau et injecter des messages erronés. En raison de la nature du support sans fil et la nature des réseaux de capteurs *(déployés dans des zones hostiles et sans surveillance)*, il est extrêmement difficile d'assurer l'authentification.

**2.4 La disponibilité :** représente la propriété d'un système d'être accessible par une entité autorisée dans les limites spécifiées [7]. La disponibilité reste difficile à assurer dans les RCSFs. En effet, un nœud peut ne pas servir des informations afin de ne pas épuiser ses ressources d'énergie, de mémoire et de calcul.

**2.5 La fraîcheur des données :** implique que les données sont récentes, et que l'attaquant n'a pas retransmis d'anciens messages. Pour résoudre ce problème, un numéro de séquence peut être ajouté aux paquets de données pour filtrer les anciens messages.

**2.6 Le non répudiation :** la répudiation [7, 8] est signalée dans deux cas différents: dans le premier cas, le nœud récepteur affirme que les données n'ont jamais été reçues, même si elles ont été correctement reçues. Par contre, dans le deuxième cas c'est le nœud expéditeur qui affirme qu'il n'a jamais envoyé les données, même si le message a été correctement délivré au destinataire. Un système de sécurité doit interdire la répudiation afin d'améliorer la traçabilité des messages dans le réseau.

**2.7 La sécurité de localisation :** la localisation est un facteur très important pour la fiabilité de fonctionnement des RCSFs. En effet, un réseau de capteurs doit être capable de localiser automatiquement chaque capteur dans le réseau. Ainsi, un réseau de capteurs conçu pour localiser des événements aura besoin d'informations précises sur la localisation afin de repérer la position exacte de ces derniers. Un nœud malveillant peut essayer de compromettre les informations de localisation afin de déstabiliser le fonctionnement du réseau, ce qui rend la sécurité de localisation un objectif très important pour les systèmes de sécurité.

## 3. CONTRAINTES DE SECURITE DANS LES RCSFs

Les RCSFs possèdent plusieurs contraintes qui rendent les mécanismes de sécurité proposés pour les réseaux Ad-hoc inapplicables à leur niveau. En conséquence, ces mécanismes doivent être adaptés aux caractéristiques de ce type particulier de réseaux sans fil. En effet, le développement de mécanismes de sécurité fiables nécessite une connaissance approfondie des contraintes mentionnées ci-dessous.

### 3.1 La communication sans fil:

Contrairement aux réseaux câblés, où un dispositif doit être connecté physiquement, le milieu sans fil est ouvert et accessible à tout le monde. Par conséquent, toute transmission peut facilement être interceptée, altérée, ou retransmise par un attaquant. De plus, le nœud malveillant peut endommager les paquets de données en provoquant des collisions et des interférences dans le

canal de transmission. D'un autre point, la communication sans fil est particulièrement coûteuse d'un point de vu énergétique (un bit transmis est équivalent à environ un millier d'opérations CPU [9]). C'est pourquoi on ne peut pas utiliser des mécanismes de sécurité compliqués, impliquant l'échange d'un grand nombre de messages entre les nœuds capteurs.

### 3.2 La limitation de ressources :

Les mécanismes de sécurité connus pour leur grande consommation de ressources, sont confrontés à l'extrême limitation des nœuds capteurs. En effet, pour augmenter la durée de vie des capteurs, on a tendance à diminuer leur mémoire, CPU et leur bande passante radio. Ces limitations physiques imposent la conception de mécanismes de sécurité à faible consommation énergétique, exigeant moins de puissance de calcul, d'espace mémoire et de bande passante. Le tableau suivant résume les caractéristiques physiques limitées de la majorité des nœuds capteurs disponibles sur le marché.

| Capteur | Circuit radio | CPU | RAM | Mémoire de stockage |
|---|---|---|---|---|
| MICA2 | CC1000 | ATMega128 | 4 KB | 128 à 512 KB |
| MICAZ | CC2420 | ATMega128 | 4 KB | 128 à 512 KB |
| TelosA | CC2420 | TI MSP 430 | 2 KB | 60 -512 KB |
| TelosB | CC2420 | TI MSP 430 | 10 KB | 48 KB - 1 MB |
| BTnode3 | CC1000/Bluth | ATMega128 | 64 KB | 128 - 180 KB |
| XYZ | CC2420 | ARM 7 | 32 KB | 256 - 256 KB |
| Shimmer | CC2420/Bluth | TI MSP 430 | 10 KB | 48 KB - Up to 2 GB |
| Cricket | CC1000 | ATMega128 | 4 KB | 128 - 512 KB |
| Imote2 | CC2420 | Intel PXA271 | 256 KB | 32 - MB |

**Tableau 2.1 :** Caractéristiques physiques des nœuds capteurs disponibles sur le marché

Assurément, la défaillance des nœuds capteurs est généralement liée à l'épuisement de leurs batteries. Par conséquent, la limitation énergétique constitue la plus grande contrainte des capteurs sans fil. Ainsi, la sécurisation des opérations de base d'un réseau de capteurs devient une tâche difficile. Les algorithmes à base de clés publiques [10] sont inadaptés aux RCSFs, étant donné que ces derniers sont très gourmands en termes de ressources physiques. Par contre, les algorithmes basés sur le chiffrement symétrique et les fonctions de hachage sont largement utilisés et constituent les outils de base pour la sécurisation des RCSFs. Cependant, ces techniques ne sont pas aussi efficaces que la cryptographie à clé publique, ce qui complique la conception d'applications sécurisées.

### 3.3 L'environnement non surveillé :

Selon l'application, les nœuds peuvent être laissés sans surveillance pendant de longues périodes de temps, ce qui les expose aux attaques physiques. Ainsi, les capteurs peuvent être capturés, compromis ou détruits par des attaquants. Un attaquant peut prendre le contrôle d'un nœud dans le réseau après le déploiement, ce qui permet de l'endommager physiquement, le rendant ainsi non fonctionnel. L'attaquant est capable de modifier les informations captées par le nœud capteur ce qui lui permet d'effectuer une variété d'attaques. De plus, des informations vitales pour la sécurité du réseau (tel que la table de routage, des données et des clés cryptographiques), peuvent être extraites du nœud capturé. L'absence de toute infrastructure fixe augmente la vulnérabilité des

réseaux de capteurs. En effet, il n'existe pas de contrôleur central pour surveiller le fonctionnement du réseau et identifier les tentatives d'intrusions. Alors que la plupart de réseaux de capteurs ont une station de base désignée, néanmoins son rôle est généralement limité à la collecte des données et la distribution de requêtes, et ne comprend pas toute forme de surveillance.

### 3.4 Le déploiement aléatoire et l'utilisation à grande échelle

La capacité d'être déployé dans des grandes surfaces avec un nombre important de nœuds capteurs est l'une des caractéristiques les plus intéressantes des RCSFs. Le déploiement est fait sans aucune connaissance préalable de la position des nœuds. L'environnement du déploiement est généralement dynamique avec des topologies qui changent fréquemment. Par conséquent, les réseaux de capteurs nécessitent des mécanismes de sécurité plus robustes pour faire face à l'instabilité de l'environnement. De plus, ces mécanismes devraient être adaptés de telle sorte que le grand nombre de nœuds n'affectera pas leur l'efficacité.

### 3.5 L'agrégation des données

L'agrégation des données [11] est l'une des techniques d'optimisation du temps de vie des RCSFs. L'idée de base est de réduire la quantité de données transférées vers le nœud puits, en éliminant les données redondantes est inutiles. Toutefois, cela exige que les nœuds intermédiaires accèdent aux données échangées pour effectuer le traitement d'agrégation de données. Conséquemment, la confidentialité des données est non respectée, ce qui pose un autre défi pour les mécanismes de sécurité.

## 4. *LES TYPES D'ATTAQUES*

Les nouvelles contraintes de sécurité imposées aux réseaux de capteurs les rendent vulnérables à divers types d'attaques. Ces dernières adoptent généralement de nouvelles stratégies d'attaque, en se basant sur la nature et les caractéristiques uniques des RCSFs. En effet, les attaques peuvent être classifiées selon la couche protocolaire qu'elles ciblent dans le modèle OSI *(classification basée en couches)*. Une autre méthode de classification catégorise les attaques en se basant sur la nature de l'attaquant. Ainsi, l'attaque peut être classifiée comme interne ou externe, et passive ou active [12].

Les attaques externes sont effectuées par des nœuds qui ne font pas partie du réseau. Par contre, les attaques internes sont lancées par des nœuds légitimes *(appartenant au réseau)* qui se comportent contre leurs cahiers de charges *(compromis)*. Les attaques passives se contentent uniquement de l'analyse du trafic, l'interception et l'espionnage des données. Cependant, les attaques actives manipulent généralement les données *(modification, retransmission, rejet de paquets…)*, compromettent la communication entre les nœuds et affectent la disponibilité de ces derniers.

Les attaques peuvent être aussi classées selon la puissance du nœud malveillant [13]. Ainsi, un attaquant peut utiliser, durant son attaque, des dispositifs similaires à ceux des nœuds capteurs *(mêmes caractéristiques)*. L'attaquant peut aussi avoir des capacités d'un ordinateur portable,

avec des dispositifs plus puissants (en termes de bande passante, vitesse de traitement, capacité de mémorisation, couverture radio et énergie). Enfin, les attaques peuvent être classées selon l'aspect de sécurité qu'elles veulent déstabiliser. On distingue ainsi trois types d'attaques qui ciblent respectivement : la disponibilité, l'intégrité des données, la confidentialité et l'authentification.

### 4.1 Les attaques contre la disponibilité :

Les attaques contre sur la disponibilité ou déni de service *(DoS : Denial of Service)*, font généralement référence à la tentative de perturber, corrompre ou détruire un réseau. Cependant, elles peuvent être n'importe quel événement qui diminue ou élimine la capacité du réseau d'exécuter ses fonctions attendues [14]. Plusieurs techniques ont été proposées pour faire face aux attaques de déni de service. Par contre, la plupart des mécanismes de défense exigent beaucoup de calculs supplémentaires, ce qui ne convient pas aux RCSFs (*ressources limitées)*. Les attaques de déni de service adoptent plusieurs stratégies. Cependant, on peut les classer selon la couche ciblée du modèle OSI.

***4.1.1 Les attaques au niveau de la couche physique :*** *l*es attaques de déni de service qui ciblent cette couche sont généralement divisées en deux types : attaque de brouillage (Jamming) et attaque d'altération (Tampering).

- Attaque de brouillage (Jamming): Dans cette attaque, le nœud malveillant essaye d'interférer avec la fréquence radio utilisée par les nœuds capteurs dans le réseau [14,15]. La source de brouillage peut être assez puissante pour perturber l'ensemble du réseau. Le nœud malveillant peut lancer des attaques de brouillage stratégiques en ciblant des zones sensibles du réseau *(station de base ou chef de cluster)* sans attirer les attentions *(signal de brouillage qui respecte les normes du réseau)*.

- Attaque d'altération (Tampering): Les RCSFs sont généralement déployés dans des zones hostiles et non surveillées. Par conséquent, les nœuds capteurs sont vulnérables aux attaques d'altération physique [16] qui causent généralement des dégâts irréversibles. En effet, un attaquant peut capturer un nœud capteur et extraire les clés cryptographiques, altérer les circuits électroniques, modifier les codes programme ou même remplacer le nœud capteur par un capteur malveillant.

***4.1.2 Les attaques au niveau de la couche liaison :*** les attaques qui ciblent cette couche prennent comme objectif la provocation de collisions, l'épuisement et l'allocation abusive des ressources [14]. En effet, une collision survient lorsque deux nœuds essayent de transmettre simultanément en utilisant la même fréquence radio. La présence de collision nécessite la retransmission des données, ce qui n'est pas souhaitable dans le cas des RCSFs. Ainsi, un attaquant peut violer le protocole de communication afin de générer des collisions avec les nœuds du réseau. De plus, les collisions répétées peuvent provoquer l'épuisement des réserves énergétiques. L'allocation abusive est un autre type d'attaques de déni de service. Un attaquant peut allouer abusivement le canal de transmission, ce qui provoque la dégradation des services offerts par le réseau.

En effet, L'attaque par déni de sommeil (*Denial of sleep attacks*) [17, 18] représente l'une des attaques les plus connues au niveau de la couche liaison. Cette attaque est un type particulier

d'attaques de déni de service, qui cible les réserves énergétiques dans l'effort d'épuiser ses ressources limitées et de réduire la durée de vie du réseau. L'attaque par déni de sommeil essaye de pirater le système de gestion d'énergie des nœuds capteurs afin de réduire les possibilités de transition dans les états à faible puissance (état de sommeil).

Afin de réduire la consommation d'énergie, les protocoles de la sous couche MAC (*SMAC, TMAC, BMAC…*) mettent périodiquement les nœuds capteurs dans un état de sommeil temporaire. Ainsi, le temps de cycle de chaque nœud est divisé en deux phases : une pour l'écoute du trafic et l'autre pour la mise en sommeil temporaire. Durant la phase d'écoute, les nœuds capteurs sont capables de communiquer entre eux. Afin de synchroniser le temps d'activation et de sommeil, les nœuds doivent échanger des paquets de contrôle nommés SYNC. Les nœuds capteurs échangent aussi des paquets RTS (*Request to Send*) et CTS (*Clear to Send*), afin d'éviter les problèmes d'interférences et de collisions. Les nœuds capteurs basés sur les protocoles précédents désactivent leur antenne radio et se mettent dans un état de sommeil qui peut durer 90% du temps alloué au cycle de communication.

Selon la stratégie adoptée, les attaques par déni de sommeil peuvent être classées en six catégories. La figure suivante présente ces types d'attaques :

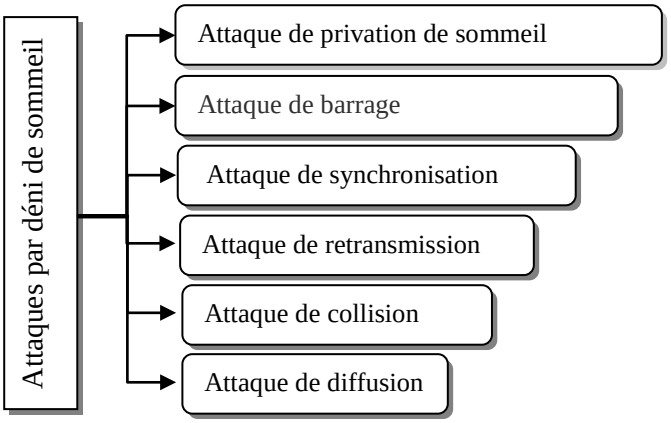



**Figure 2.1 :** Les attaques de déni de sommeil

- Attaque de privation de sommeil (*Sleep deprivation attack*) : L'attaque de privation de sommeil [18] cible tout dispositif avec une ressource d'énergie limitée, qui tente de la préserver en restant aussi longtemps que possible dans un mode en veille à faible puissance. En effet, l'attaquant interagi avec la victime d'une manière qui semble légitime. Cependant, le but de cette interaction est de garder le nœud victime hors de son état de conservation énergétique *(mode de sommeil)*. Par exemple, un nœud malveillant pourrait envoyer périodiquement des demandes d'envoi de données *(paquets RTS)*, forcer le nœud victime à accepter ces demandes *(envoi de paquets CTS)*, et rester éveillé pour attendre les données qui ne seront jamais envoyées par le nœud attaquant. Ainsi, cette attaque peut être utilisée pour réduire considérablement la durée de vie de la victime. En outre, elle est difficile à détecter étant donné qu'elle est réalisée uniquement par le biais d'interactions qui paraissent innocentes.

- Attaque de barrage (*Barrage attack*) : L'attaque de barrage [19] bombarde les nœuds victimes par des messages légitimes. Cependant, le but de ces messages est l'épuisement des réserves énergétiques du nœud victime, en l'amenant à rester en mode actif et d'effectuer des

opérations à forte intensité énergétique. Semblablement à l'attaque de privation de sommeil, l'attaque de barrage empêche la victime d'entrer dans son mode en veille. Par contre, les victimes d'une attaque de barrage vont effectuer des tâches plus coûteuses *(réception de données)*, alors que les victimes d'attaque de privation de sommeil vont être pour la plupart du temps en mode d'écoute.

- Attaque de synchronisation (*Synchronization attack*) : Le but de cette attaque [20] est de provoquer des problèmes de synchronisation au niveau de la sous couche MAC. L'attaque de synchronisation est simple, mais difficile à détecter, étant donné que le nœud malveillant respecte les règles imposées par le protocole de communication. En effet, chaque nœud capteur maintient un calendrier d'activation qui fixe ses périodes d'écoute et de sommeil, et échange périodiquement ce dernier avec ses nœuds voisins afin de synchroniser leurs horloges et former ainsi un cluster virtuel. Cela leur permet de s'activer et de se mettre en mode sommeil en même temps. La mise à jour des calendriers d'activation se fait en échangeant un paquet de synchronisation 'SYNC'. Le paquet SYNC est très court, et comprend l'adresse d'expéditeur et l'heure de son prochain sommeil. Quand un nœud reçoit un paquet SYNC d'un nœud qui appartient au même cluster virtuel, il recalcule son temps de sommeil *(moyenne de son prochain temps de sommeil et le temps de sommeil reçu)* afin de le synchroniser avec le nœud émetteur. Ainsi, L'attaquant peut inciter le nœud ciblé à rester éveillé pendant une fraction supplémentaire du cycle d'écoute en envoyant un message de synchronisation compromis. Le nœud attaqué prolonge donc sa durée d'écoute en se basant sur la durée du sommeil compromis extraite du message de synchronisation reçu.

- Attaque de retransmission (*Replay attack*) : Dans l'attaque de retransmission, les messages échangés entre les nœuds capteurs sont enregistrés et retransmis afin d'épuiser l'énergie des nœuds récepteurs. Avec l'absence de mécanisme d'anti-rejoue, les données retransmises peuvent être diffusées à travers le réseau, ce qui provoque le gaspillage d'énergie de tous les nœuds relais. La retransmission non détectée a l'avantage supplémentaire *(pour l'attaquant)* de détourner le réseau de son but initial. Par exemple, la retransmission du trafic dans un réseau de capteurs militaires, déployés pour capter les mouvements des troupes ennemi, pourrait mal orienter les unités de combat.

- Attaque de collision (*Collision attack*) : L'attaque de collision [21] peut être facilement lancée par un nœud compromis *(ou malveillant)*, qui ne respecte pas les conditions d'accès au média de transmission. L'objectif est de provoquer des collisions avec des transmissions voisines en envoyant périodiquement un simple paquet de bruit. Par exemple, dans le protocole SMAC [22], l'attaquant vérifie le canal de communication afin d'assurer que le support est occupé *(réception des paquets RTS et CTS)*. Si c'est le cas, il envoi des paquets corrompus afin d'entrer en collision avec les paquets échangés dans le réseau.

- Attaque de diffusion (*Broadcas tattack)* : l'attaque de diffusion est présentée dans [18], où l'impact d'un nœud malveillant obéissant aux règles de la sous couche MAC, et diffusant du trafic non authentifié dans le réseau est modélisé. En effet, de longs messages peuvent être diffusés et doivent être entièrement reçus par tous les nœuds du réseau avant que ces derniers soient rejetés à cause de l'échec d'authentification.

***4.1.3 Les attaques au niveau de la couche réseau :*** la couche réseau est la couche la plus ciblée par les attaques de déni de service, vu le rôle important de cette couche dans le bon fonctionnement du réseau. En effet, l'organisation Ad-hoc des RCSFs permet à n'importe quel nœud de devenir un routeur pour transmettre les données d'un nœud à l'autre. Ainsi, les nœuds malveillants peuvent manipuler les informations de routage afin de lancer différents types d'attaques. La section qui suit résume les principales attaques de déni de service au niveau de la couche réseau.

- L'attaque de trou de puits (*Sinkhole attack)*: Dans cette attaque [13, 14,23], l'attaquant essaye de rendre le nœud compromis très attirant par rapport à ses nœuds voisins. Par conséquent, les nœuds voisins vont choisir le nœud compromis comme prochain nœud dans leur chemin de routage afin d'acheminer leurs données captées. Le nœud compromis peut ainsi modifier, supprimer ou même rejeter les paquets reçus, ce qui affecte considérablement le bon fonctionnement du réseau.

- L'attaque de trou noir (Black hole) : Elle représente un cas particulier de l'attaque de trou de puits. En rejetant tous les messages reçus, le nœud malveillant crée une sorte de trou noir qui aspire toutes les données qui lui sont transmises. Le nœud malveillant peut aussi se placer dans un endroit de routage stratégique et supprimer tous les messages qu'il devrait retransmettre, causant la mise hors service de tout le réseau.

- La transmission sélective des paquets (*Selective forwarding*) : Les réseaux de capteurs adoptent généralement des stratégies de transmission multi-sauts, dans lesquelles un nœud capteur transmet ses données captées à son proche voisin dans le chemin de routage. Dans le cas d'une attaque de transmission sélective [13], Un attaquant peut compromettre un nœud de manière à ce qu'il envoi de façon sélective les messages reçus de ses nœuds voisins. Cette attaque peut être combinée à l'attaque de trou de puits, générant ainsi un nouveau type d'attaques appelé : l'attaque du trou gris. En effet, l'attaque du trou gris peut être considérée comme une variante améliorée de l'attaque du trou noir. Contrairement au trou noir, le trou gris retransmet certaines informations. Par exemple, il relaye toutes les informations concernant le routage, et rejette celles qui sont critiques. Ce type d'attaques est ainsi plus difficile à détecter, étant donné que le nœud malicieux ne va pas supprimer tous les messages reçus.

- L'usurpation d'accusés de réception *(Acknowledgment spoofing)*: L'échange d'accusés de réception est très important au bon fonctionnement de la plupart des protocoles de routage dédiés aux RCSFs. A cause de la nature sans fil de ces derniers, le nœud malveillant peut facilement espionner le trafic sur le réseau et imiter les accusés de réception échangés entre les nœuds capteurs [13]. Ainsi, les nœuds malveillants peuvent fournir de fausses informations sur l'état des nœuds et la bonne transmission des données.

- L'usurpation des informations de routage (*Spoofed routing information*) : L'objectif de cette attaque est de déstabiliser le fonctionnement du protocole de routage en ciblant les paquets d'informations de routage qui circulent dans le réseau. En effet, l'attaquant peut usurper, modifier ou rediffuser l'information de routage afin de perturber le fonctionnement du réseau [13]. Ces perturbations entrainent l'empoisonnement des tables de routage, ce qui est en mesure

de créer des boucles de routage, des chemins de routage très coûteux, la congestion et les débordements des tables de routage.

- L'attaque Sybil (*Sybil attack*) : Le principe de cette attaque [23] est la déstabilisation du fonctionnement des protocoles de routage multi-sauts et les mécanismes de maintenance de topologie réseau. Le nœud malveillant peut créer un grand nombre d'identités afin de gagner de l'influence sur les autres nœuds du réseau. Chaque identité (ID) peut être générée aléatoirement ou être dupliquée *(recopiée)* d'une identité légitime qui existe déjà. Ainsi, le nœud attaquant peut profiter de ces multiples identités pour être sélectionné comme chef de groupe *(cluster head)*, ou pour créer des chemins de routage pour son propre intérêt.

- L'attaque de trou de ver (*Worm hole attack*) : Dans cette attaque [13], un nœud malveillant intercepte les paquets de données de ses nœuds adjacents, et les retransmet en utilisant un chemin de routage multi-sauts à un autre nœud malveillant, qui se charge de rediffuser ces paquets. Cela peut déformer les distances entre les nœuds du réseau, et tromper le processus de découverte des nœuds voisins. Par conséquent, un nœud capteur peut sélectionner un nœud éloigné comme son proche voisin et lui transmettre ses données, ce qui entraine l'épuisement rapide des ressources et la réduction du temps de vie du réseau.

- Attaque d'inondation par paquet de Hello (*Hello flood attack*) : L'échange périodique des paquets Hello est très important pour la localisation et la réorganisation de la plupart des protocoles de communication. En effet, un nœud capteur assume que la réception d'un tel paquet implique la présence de l'émetteur dans sa zone de couverture radio. Par conséquent, un attaquant peut utiliser un émetteur de haute puissance pour tromper un grand nombre de nœuds en leur faisant croire qu'ils sont dans son voisinage [13]. En recevant les paquets Hello, les nœuds capteurs tentent de transmettre leurs données au nœud attaquant qui peut être hors de leur portée radio.

- Attaques d'information fabriquées (*Fabricated information Attacks*): L'attaque d'informations fabriquées [16] est réalisée en générant de faux messages de données. Cette attaque est difficile à détecter, puisque le nœud attaquant respecte les conditions imposées par le protocole de routage utilisé. Ainsi, le nœud malveillant pourrait envoyer de fausses valeurs de mesure qui ne reflètent pas la réalité de son environnement. Cela amène à prendre de fausses décisions, et peut être dangereux dans des scénarios tels que la surveillance des champs de bataille et d'environnements hostiles.

***4.1.4 Les attaques au niveau de la couche transport :*** la couche transport gère la connectivité de bout-en-bout entre les nœuds capteurs. Des protocoles de séquençage peuvent être utilisés par cette couche afin d'améliorer la fiabilité de la connexion. Cependant, cela rend la couche transport très vulnérable aux attaques de déni de service. Parmi ces attaques, on peut citer : l'attaque d'inondation et l'attaque de désynchronisation.

- L'attaque d'inondation (*Flooding*): Afin de garantir une connexion fiable, il est nécessaire de maintenir son état à chaque extrémité. Ainsi, le réseau devient vulnérable à l'épuisement de la mémoire par les attaques d'inondation [14]. Un attaquant peut à plusieurs reprises faire de nouvelles demandes de connexion jusqu'à ce que les ressources requises par chaque connexion

soient épuisées ou que la limite maximale soit atteinte. Dans les deux cas, d'autres demandes légitimes seront ignorées.

▪ L'attaque de désynchronisation (*De-synchronization*): L'objectif de cette attaque est l'interruption des connexions existantes [14]. Un attaquant peut par exemple intercepter à plusieurs reprises des messages destinés à un autre nœud dans le réseau, incitant ainsi le nœud récepteur à demander la retransmission des trames manquées. Si l'attaque est lancée au bon moment, l'attaquant peut dégrader ou même empêcher le nœud récepteur de bien échanger les données avec les nœuds émetteurs. Par conséquent, le nœud attaquant pousse sa victime à gaspiller son énergie en tentant de réparer les erreurs de transmission qui n'ont jamais vraiment existé.

### 4.2 Les attaques contre la confidentialité et l'authentification :

Les réseaux de capteurs communiquent généralement des données sensibles sur lesquelles se basent des décisions importantes. Ainsi, le récepteur a besoin de s'assurer que les données sont échangées d'une manière confidentielle et proviennent de la bonne source. Les attaques contre la confidentialité et l'authentification visent à infiltrer le réseau et espionner les messages échangés entre les nœuds capteurs afin de compromettre le bon fonctionnement de ce dernier. On peut classer ces attaques en deux catégories : attaque des nœuds répliqués et attaque contre le secret.

▪ Attaque des nœuds répliqués (*Node replication attack*) : L'objectif de cette attaque est la création de nœuds malveillants avec de fausses identités *(identificateur)*, copiées à partir des nœuds légitimes existants dans le réseau [24]. Le nœud répliqué rejoint le réseau sans attirer les attentions, et peut perturber le routage des données en reliant les paquets à des routes erronées. Les nœuds répliqués peuvent aussi transmettre de fausses lectures ou signaler des événements non existants. L'attaquant peut également placer les nœuds répliqués dans des emplacements stratégiques dans le réseau afin de pouvoir facilement manipuler une partie spécifique du réseau.

▪ Attaque sur la confidentialité (*Attacks on privacy*) : Les RCSFs collectent l'ensemble de leurs données à travers un grand nombre de nœuds capteurs. Cependant, ces nombreuses sources de données peuvent être compromises, ce qui pose un véritable chalenge pour la confidentialité des informations échangées. De plus, la nature sans fil des RCSFs facilite l'interception et l'accès aux données transmises. En effet, L'adversaire n'a pas besoin d'être physiquement présent sur les champs de captage, ce qui diminue le risque d'être détecté et rejeté. La section suivante résume l'essentiel des attaques qui ciblent la confidentialité des données [25, 26].

- L'espionnage et la surveillance passive (*Eaves dropping and passive monitoring*): C'est la forme la plus commune et la plus simple des attaques sur la confidentialité des données. Sans la présence de mécanismes cryptographiques, les nœuds malveillants peuvent facilement écouter et comprendre le contenu des messages échangés dans le réseau. Ainsi, l'attaquant peut espionner et capter des informations stratégiques qui peuvent servir au lancement d'attaques plus dangereuses.

- L'analyse du trafic (*Traffic analysis*): Cette attaque [27] est souvent combinée à l'attaque d'espionnage afin d'augmenter son degré d'efficacité. En analysant le trafic du réseau, l'attaquant peut identifier le rôle et l'importance de quelques nœuds dans le réseau. Par

exemple, une augmentation soudaine du nombre de messages échangés entre les nœuds capteurs, signifie que ces derniers sont des activités et des événements spécifiques à surveiller. En outre, l'attaquant peut identifier les nœuds chefs de groupe (cluster head) sans avoir à comprendre le contenu des messages.

### 4.3 Attaques contre l'intégrité des données :

Les RCSFs sont généralement destinés à surveiller certains environnements et à transmettre des informations souvent sensibles et critiques vers un centre de gestion et de contrôle. Par conséquent, l'intégrité des données est encore plus importante que la confidentialité. En effet, l'attaquant est en mesure d'écouter et de modifier les données recueillies, pour les rendre incomplètes ou incorrectes. Il peut aussi ajouter quelques fragments, ou agréger des données corrompues avec celles reçues, ce qui rend l'attaque très difficile à détecter. L'attaque contre l'intégrité des données cible généralement la couche application, étant donné que celle-ci est responsable des services visibles aux utilisateurs. L'objectif est de manipuler les données afin de changer leur sémantique.

## 5. *LES MECANISMES DE SECURITE*

Le développement d'un mécanisme de sécurité est souvent confronté à la nature limitée des ressources dans les RCSFs. Ainsi, on doit toujours faire un compromis entre la sécurité assurée et le surcoût introduit par la contre-mesure appliquée. Basés sur la puissance et le degré de malveillance de l'attaquant, plusieurs dispositifs de contre-mesure ont été proposés. Cependant, la plupart de ces mécanismes supposent rarement des modèles d'attaquants à grande puissance, d'où la nécessité de les combiner afin de satisfaire toutes les conditions de sécurité. La section suivante présente les différents types des contre-mesures disponibles.

### 5.1 Le cryptage des données :

L'une des premières contre-mesures de sécurité est l'établissement d'un système cryptographique basé sur des clés sécurisées. Ces dernières permettent de chiffrer et d'authentifier les messages envoyés entre les nœuds capteurs. La sélection d'une méthode de chiffrement adaptée aux réseaux de capteurs est une tache vitale et délicate. Ainsi, les algorithmes cryptographiques doivent respecter la limitation en ressources des nœuds capteurs en n'exigeant pas une grande puissance de calcul et une capacité de stockage élevée. De plus, ils doivent être moins énergivores en énergie.

**5.1.1** ***La gestion des clés :*** la gestion des clés est la tâche qui prend en charge l'établissement et le maintien (*sauvegarde, protection, distribution, chargement, utilisation et destruction*) des clés conformément à une politique de sécurité. Afin d'établir une gestion de clés efficace, il faut généralement s'interroger sur quatre préoccupations de base.

**a.** Combien de clés sont nécessaires et comment seront elles distribuées?
- Il s'agit d'un ***problème de déploiement de clé***.

**b.** Comment vont faire les paires ou les groupes de nœuds pour établir une session sécurisée?

- Il s'agit d'un ***problème d'établissement de la clé***.

**c.** Comment ajouter un nouveau nœud de telle sorte qu'il est en mesure d'établir une session sécurisée avec les nœuds existants, tout en n'étant pas en mesure de déchiffrer le trafic dans le réseau ?
- Il s'agit d'un ***problème d'addition de nouveau nœud***.

**d.** Comment peut-on expulser un nœud du réseau, de telle sorte qu'il ne sera pas en mesure d'établir des sessions sécurisées, et de déchiffrer le trafic dans le réseau?
- Il s'agit d'un ***problème d'expulsion de nœud***.

La gestion des clés doit satisfaire la plupart des conditions de sécurité *(l'authenticité, la confidentialité, l'intégrité, la scalabilité et la flexibilité)*. De plus, elle doit respecter toutes les contraintes de limitation de ressources telles que: l'autonomie des batteries, la porté des transmissions radio, la bande passante, la mémoire disponible et le déploiement aléatoire.

- ***L'autonomie de la batterie***: Les nœuds capteurs sont équipés de batteries dont la durée de vie est limitée, ce qui rend l'utilisation des techniques à base de clés asymétriques *(comme la cryptographie à clé publique)* peu pratique. Étant donné que ces techniques consomment beaucoup d'énergie afin d'exécuter leurs calculs mathématiques très complexes, l'utilisation des techniques à base de clés symétriques *(moins énergivores en énergie)* serait plus efficace.

- ***La bande passante limitée:*** Ayant une largeur de bande limitée, les nœuds capteurs ne sont pas censés transmettre des gros blocs de données. Pour compenser, les techniques de gestion des clés doivent utiliser de petits morceaux de données pour transférer leurs informations.

- ***La capacité de mémorisation:*** Un nœud capteur dispose généralement d'un espace de stockage qui varie entre 6 à 8 Kbit, dont la moitié est occupée par le système d'exploitation. Par conséquent, les techniques d'établissement et de maintien des clés doivent utiliser efficacement l'espace de stockage qui reste.

- ***Le déploiement aléatoire:*** Comme les nœuds capteurs sont déployés de façon aléatoire et dynamique, il n'est pas possible de savoir la position de chaque nœud. De ce fait, les méthodes de gestion des clés ne doivent pas exiger la position des nœuds lors de l'initialisation et la création des clés.

La gestion des clés doit aussi répondre à certains critères d'efficacité face aux nœuds adversaires. Ces critères peuvent inclure la résistance, la révocation et la résilience.

- ***La résistance:*** Un attaquant peut capturer et compromettre quelques nœuds, puis reproduire ces derniers afin de les déployer dans le réseau. Ainsi, l'attaquant peut alimenter l'ensemble du réseau avec ses nœuds répliqués et prendre le contrôle du réseau. Une bonne gestion des clés doit résister à la réplication de nœuds pour s'immuniser contre une telle attaque.

- ***La révocation:*** Les techniques de gestion des clés doivent fournir des méthodes efficaces (moins gourmandes en ressources) pour révoquer les nœuds compromis.

- ***La résilience:*** En respectant ce critère d'efficacité, la gestion des clés peut assurer que les informations secrètes sur les autres nœuds, ne seront pas révélées en cas de capture.

***5.1.2 L'établissement des clés :*** traditionnellement, l'établissement des clés s'effectue en se basant sur un système de cryptographie asymétrique *(à base de clé publique)* étant donné que ce dernier offre des mécanismes plus sûrs et stables pour l'authentification et la distribution des clés. Par contre, les RCSFs ne peuvent pas palier toutes les exigences de la cryptographie asymétrique (*capacité de calcul et mémoire de stockage),* ce qui rend son utilisation non appropriée. Néanmoins, il existe quelques recherches [28, 29] qui montrent la possibilité d'appliquer la cryptographie à clé publique, en réduisant le temps de calcul ainsi que la quantité de données transmises et stockées. La cryptographie à base de courbe elliptique (ECC, Elliptic Curve Cryptography) [29,30, 31 et 32], est un bon exemple de ces recherches.

En effet, la cryptographie à base de clés symétriques offre des caractéristiques plus attirantes en termes de vitesse et du faible coût énergétique. Pour cette raison, la plupart des schémas de gestion des clés proposés pour les RCSF sont basés sur la cryptographie symétrique. Il existe beaucoup de recherches qui proposent des mécanismes de cryptographie symétrique, dont les plus populairess'intitulent:RC4 [33], RC5 [34], IDEA [33], SHA-1 [35], and MD5 [33, 36].

La cryptographie à base de clés symétriques se heurte souvent au problème d'échange de clés. Autrement dit, les deux hôtes communicants doivent en quelque sorte connaître la clé partagée avant de pouvoir communiquer en toute sécurité. Donc, le problème qui se pose est de savoir comment faire en sorte que la clé soit partagée entre les deux hôtes qui souhaitent communiquer, et pas entre d'autres hôtes non autorisés qui désirent écouter. La solution commune est d'utiliser une méthode de pré distribution dans laquelle les clés sont chargées dans les nœuds capteurs avant le déploiement.

***5.1.3 La distribution des clés :*** la distribution des clés est l'une des phases cruciales dans le processus de gestion des clés. Cette phase consiste à distribuer les clés cryptographiques d'une manière efficace et sécurisée sur tous les nœuds légitimes appartenant au réseau. En effet, il existe trois modèles essentiels pour la distribution des clés dans les RCSFs :

- ***La distribution à base d'une clé par réseau (Network keying):*** C'est un simple modèle de distribution qui consiste à utiliser une clé unique partagée par tous les nœuds du réseau. L'idée de base est de pré-charger les nœuds, avant le déploiement par une seule clé. Par conséquent, ce modèle utilise très peu de ressources *(stockage d'une seule clé)*, et permet la collaboration facile entre les nœuds voisins. De plus, l'utilisation d'une seule clé offre des possibilités d'évolution et de flexibilité *(ajout de nouveaux nœuds)*. Cependant, ce modèle est très vulnérable et ne présente aucune résilience contre la compromission d'un nœud. Étant donné qu'en capturant un seul nœud capteur, l'attaquant peut procurer la clé de sécurité et compromettre tout le réseau.

- ***La distribution à base d'une clé par paire de nœuds (Pair-wise keying):*** Dans cette solution, une clé sera partagée uniquement entre une paire de nœuds capteurs. Ainsi, chaque nœud est pré-chargé avec N-1 clés secrètes. Chacune de ces clés est connue seulement par ce nœud et un des N-1 autres nœuds *(N étant le nombre de nœuds dans le réseau)*. Ce modèle de distribution permet une résilience parfaite car la compromission d'un nœud n'affecte pas la sécurité des autres nœuds. Cependant, il exige une capacité mémoire importante pour stocker les N-1 clés *(N peut être grand)*. De plus, il ne permet pas l'ajout de nouveaux nœuds, étant donné que les nœuds existants ne possèdent pas les clés de ces nouveaux nœuds.

- ***La distribution à base de groupe (Group keying):*** Ce modèle combine les caractéristiques des deux précédents modèles. Au sein d'un groupe de nœuds *(qui forment un cluster)*, les communications sont effectuées à l'aide d'une seule clé partagée *(distribution à base de clé par réseau)*. Toutefois, les communications entre les groupes vont utiliser des clés partagées entre chaque paire de groupes *(distribution à base de clé par paire de nœuds)*. Ainsi, un certain équilibre entre la robustesse, la résilience, l'évolutivité et le coût en ressources sera maintenu. Cependant, un tel système est difficile à mettre en place.

***5.1.4 Les protocoles de gestion des clés :*** basés sur les trois célèbres modèles de distribution de clés, les chercheurs ont proposé divers protocoles de gestion des clés. Dans cette section, nous allons résumer ceux qui sont les plus utilisés.

- Eschenauer et Gligor [37] ont proposé l'un des premiers protocoles de gestion des clés pour les réseaux de capteurs. Basé sur de simples principes, ce dernier fournit un bon compromis entre la robustesse et l'évolutivité. Initialement, le protocole va générer un grand nombre de clés. Ensuite, des sous-ensembles de clés seront créés aléatoirement, et pré-chargés au niveau de chaque nœud dans le réseau. Lorsque deux nœuds doivent communiquer, ils recherchent la présence d'une clé commune au sein de leur ensemble de clés. Si une telle clé n'existe pas, ils tentent de communiquer par le biais d'un autre nœud intermédiaire, qui est en mesure d'établir des communications avec les deux nœuds. En créant des sous ensembles de clés, ce système utilise moins de mémoire comparé au modèle à base de clé par réseau. Le protocole est également évolutif, car on peut facilement modifier la taille des sous-ensembles de clés. Par conséquent, on peut renforcer la robustesse du protocole en augmentant le nombre de clés dans les sous-ensembles. Cependant, les auteurs ne décrivent pas clairement le processus de révocation ou d'actualisation des clés. En outre, ce protocole ne supporte pas le routage à base de clusters. Il n'est pas garanti que chaque nœud ait une clé commune avec tous ses voisins. Ainsi, il est fort probable que certains nœuds seront inaccessibles. Enfin, si le nombre de nœuds compromis augmente, la sécurité fournie devient insuffisante.

- Du, Deng, Han, et Varshney proposent une version améliorée du protocole précédent [38]. L'idée de base est l'utilisation d'une matrice de clés au lieu d'un simple ensemble de clés symétriques. Les auteurs utilisent ensuite un algorithme à base de clés publiques [39] afin de sélectionner la clé partagée en deux paires de nœuds. Ce protocole est très robuste aux attaques de compromission des nœuds, et offre un degré raisonnable de consommation énergétique. En comparaison avec le protocole d'Eschenauer, les auteurs affirment qu'un

attaquant doit compromettre 5 fois plus de nœuds afin de compromettre tout le réseau. Cependant, la complexité de ce protocole rend son implémentation une tâche ardue. De plus, il n'est pas adapté aux réseaux de capteurs organisés sous forme de clusters. Enfin, la gestion de la clé reste incomplète, étant donné que les opérations de révocation et de rafraichissement des clés ne sont pas supportées.

- Zhu, Setia, et Jajordia introduisent un protocole d'encryptions et d'authentification localisé nommé LEAP *(localized encryption and authentication protocol)* [40]. Le protocole est basé sur une approche de distribution de clé hybride qui utilise quatre types d'établissement de clés : clé Individuelle, clé par-paire, clé de groupe et clé du cluster. La clé individuelle est unique pour chaque nœud et elle est utilisée afin de sécuriser la communication avec la station de base. La clé de groupe est une clé globale qui sécurise la communication entre la station de base et tous les nœuds du réseau. Cette clé est générée par la station de base et diffusée dans tout le réseau. Afin de sécuriser la diffusion de la clé de groupe, LEAP utilise un mécanisme d'authentification nommé µ*TESLA (µTimed Efficient Streaming Loss-tolerant Authentication)* [41]. La clé de cluster est utilisée pour la communication inter-clusters *(entre le cluster head et tous ses nœuds membres)*. Semblablement à la clé de groupe, un deuxième mécanisme d'authentification *(one-way hash-key chain mechanism)* est utilisé afin de sécuriser la diffusion de la clé de cluster vers tous les nœuds membres. Enfin, la clé par paire de nœuds sert à sécuriser l'échange de données entre les nœuds voisins. En utilisant deux mécanismes d'authentification et plusieurs types de pré-distribution de clés, le protocole LEAP garantie un haut niveau de sécurité. Ce dernier permet la révocation et le rafraichissement des clés, ce qui assure une grande résilience à la compromission des nœuds capteurs. LEAP supporte bien les structures de communication hiérarchique, et n'affecte pas l'agrégation de données. La complexité de LEAP est proportionnelle au nombre de nœuds dans le réseau, et sa consommation en ressources est raisonnable. Néanmoins, LEAP assume que la station de base ne peut pas être compromise, ce qui n'est pas toujours vrai.

- Younis, Ghumman, et Eltoweissy présentent un nouveau protocole de gestion des clés [42], qui s'inspire principalement du protocole LEAP. Ce dernier est nommé SHELL *(Scalable, Hierarchical, Efficient, Location aware, and Light-weight)*. Comme LEAP, SHELL adopte une approche de distribution de clé hybride. Ainsi, trois types de clés sont utilisées pour sécuriser respectivement la communication entre le cluster head et ses nœuds membres *(inter cluster)*, la station de base et ses clusters heads, et entre les clusters heads voisins. SHELL assume qu'un attaquant ne peut pas compromettre la station de base. Celle-ci, est chargée de l'établissement et la distribution des trois types de clés, décris précédemment. A l'inverse de LEAP, le cluster head ne va pas générer les clés de ses nœuds membres. En effet, la station de base va désigner deux autres nœuds pour la distribution des clés à l'intérieur du cluster. Le principal avantage de SHELL est qu'il offre une grande robustesse aux attaques de compromission. En effet, la capture des nœuds générateurs des clés inter cluster ne va pas compromettre tout le réseau *(ne révèle pas toutes les clés utilisées dans le réseau)*. Le protocole SHELL garantie l'évolutivité du réseau, puisqu'il permet l'addition et le remplacement des nœuds. De plus, l'organisation hiérarchique *(basée clusters)* est fortement supportée, et l'agrégation des données est bien tolérée. Malgré ses avantages, la complexité de SHELL rend difficile son implémentation.

L'utilisation de plusieurs types de clés augmente significativement la consommation en ressources du protocole SHELL *(surcout de distribution et de stockage des clés)*.

- Panja, Madria, et Bhargava ont proposé un protocole de gestion de clés hiérarchiques basé sur la clé de groupe [43]. L'idée de base est de former les clés secrètes à partir de plusieurs clés partielles. Cela permet d'optimiser l'établissement de clé en ajoutant, en révoquant, ou on modifiant une ou plusieurs clés partielles. Ce protocole est dédié aux réseaux de capteurs hiérarchiques, composés de trois niveaux: nœuds capteurs, clusers heads et station de base. En effet, les auteurs proposent l'utilisation de deux types de clés à savoir : clé inter cluster et intra cluster. Afin de former la clé intra cluster, tous les nœuds du cluster envoient leurs clés partielles à leur cluster head. Ce dernier se charge de former la clé intra cluster *(à partir des clés partielles reçues)*, et la diffuser à tous ses nœuds membres. La clé inter cluster sera formée de la même manière que la clé précédente. Ainsi, les clusters heads vont envoyer leurs clés partielles à la station de base, qui se charge de la formation de la clé inter cluster. Ensuite cette clé sera diffusée à tous les nœuds cluster head. Comparé au protocole SHELL, ce protocole est simple et facile à implémenter. En effet, l'utilisation des clés partielles permet de réduire considérablement la complexité de calcul, la consommation énergétique et l'espace de stockage des clés. Cependant, la simplicité de ce protocole le rend très vulnérable aux attaques de compromission.

Comme synthèse, le tableau suivant résume les avantages et les inconvénients des protocoles présentés précédemment.

| **Protocole** | **Simplicité** | **Scalabilité** | **Robustesse** | **Consommation en ressources** |
|---|---|---|---|---|
| **Eschenauer** | Elevée | Moyenne | Faible | Moyenne/ Elevée |
| **Du** | Faible | Faible | Elevée | Faible/ Moyenne |
| **LEAP** | Moyenne | Moyenne | Moyenne | Moyenne |
| **SHELL** | Faible | Moyenne | Elevée | Faible/ Moyenne |
| **Panja** | Moyenne | Elevée | Moyenne | Moyenne/ Elevée |

**Tableau 2.2 :** les avantages et les inconvénients des protocoles de gestion des clés

### 5.2 L'authentification :

L'authentification est l'un des mécanismes de base pour la sécurité dans les réseaux de capteurs sans fil. Souvent construite autour d'un système cryptographique, l'authentification permet d'assurer au nœud récepteur que les données ou les paquets de contrôle *(les informations de routage, de localisation et de gestion clés)* proviennent bien de la bonne source. Traditionnellement, les protocoles d'authentification dédiés au RCSFs utilisent un mécanisme basé sur la cryptographie symétrique. Ce dernier est nommé MAC *(message authentication code)*. Le code d'authentification de message MAC [33] fait partie des fonctions de hachage à clé symétrique. Ainsi, l'émetteur génère une empreinte ou un code d'authentification en utilisant la clé symétrique partagée avec le nœud récepteur. Ce dernier calcule à son tour le code MAC avec cette même clé et le compare au code qu'il a reçu. S'ils sont bien identiques, alors la source est authentique.

***5.2.1 Protocoles d'authentification :*** plusieurs protocoles ont été proposés afin d'assurer l'authentification dans les réseaux de capteurs sans fil. La plupart de ces protocoles adoptent des mécanismes d'authentification basés sur la génération et l'échange de codes secrets *(MAC)*.

5.2.1.1 Le protocole SPIN *(Security Protocols for Sensor Networks)***:** le protocole SPIN [41] est un ensemble de mécanismes de sécurité qui assurent l'authentification des messages, l'intégrité et la confidentialité des données. SPIN est particulièrement adapté aux réseaux de capteurs hiérarchiques. Ce protocole propose deux mécanismes d'authentification nommés SNEP et µTESLA. Le premier mécanisme permet l'authentification des communications unicast (*entre une paire de nœuds)*, tandis que µTESLA assure l'authentification des communications par diffusion *(entre un groupe de nœuds)*. En effet, les nœuds capteurs ne peuvent pas diffuser des messages d'authentification sans l'assistance de la station de base, ce qui représente l'un des inconvénients du protocole SPIN. De plus, ce dernier ne prend pas en considération les contraintes temps réel, étant donné que le processus d'échange de codes MAC va augmenter les délais de communication. Par exemple, dans µTESLA le nœud récepteur doit attendre un certain temps avant de pouvoir authentifier l'origine des messages reçus.

5.2.1.2 Le protocole RPT *(Regular and Predictable Times)* : afin de remédier aux problèmes de SPIN, un nouveau protocole d'authentification a été introduit dans [44]. Les auteurs ont proposé de modifier le protocole µTESLA afin qu'il tolère les authentifications urgentes et infréquentées. Ainsi, le nouveau protocole *(nommé RPT)* authentifie dans un temps prédictible l'origine des messages reçus. Dans RPT, le nœud émetteur doit calculer un temps 'δ' qui représente la somme du maximum de délai de propagation, et du temps perdu par les erreurs de synchronisation. Pour s'authentifier, le nœud émetteur doit envoyer en premier temps son code MAC, et attendre un temps 'δ' avant d'envoyer ses données et sa clé symétrique. Ensuite le récepteur vérifie d'abord la fraicheur de la clé reçue, et compare le code MAC généré avec celui qu'il a reçu. Malgré l'authentification temps réel offerte par RPT, les analyses ont montré qu'il consomme plus de ressources *(puissance de calcul et réserves énergétiques)* comparé au protocole µTESLA.

5.2.1.3 Le protocole LEA *(Low Entropy Authentication)*: les auteurs d'RPT proposent un deuxième protocole d'authentification nommé LEA *(Low Entropy Authentication)* [44]. Ce dernier utilise un nouveau mécanisme d'authentification basé sur la cryptographie asymétrique. Afin d'être authentifié, le nœud émetteur signe les données à transmettre avec sa clé privée en produisant une signature digitale. Celle–ci sera envoyée avec les données au nœud récepteur. Ce dernier déchiffre la signature avec la clé publique, et la compare avec les données reçues. Dans le cas où elles sont identiques, la signature est validée, et l'émetteur sera authentifié comme nœud légitime. Cependant, LEA peut être très consommateur en espace de stockage, étant donné que la taille de la signature est proportionnelle à la taille des messages envoyés. De plus, LEA exige une clé publique unique pour chaque message envoyé, ce qui impose au récepteur de stocker un grand nombre de clés publiques. Enfin, l'utilisation d'algorithmes à base de cryptographie asymétrique implique une grande puissance de calcul, ce qui n'est pas approprié aux réseaux de capteurs sans fil.

5.2.1.4 Le protocole TinySec *(Tiny security):* TinySec a était proposé dans [45], afin d'assurer l'authenticité, la confidentialité et l'intégrité des données dans un RCSF. L'objectif de base était de fournir un protocole de sécurité qui ne sollicite pas de grandes puissances de calcul, d'espace

de stockage et de bande passante. En effet, les auteurs proposent deux versions du protocole TinySec : TinySec-Auth, dédié uniquement à l'authentification, et TinySec-AE, qui permet le cryptage et l'authentification. Le protocole TinySec est basé sur le mécanisme d'authentification par code *(MAC)*, qui utilise un système de cryptographie symétrique. Cependant, comparé aux protocoles précédents, la taille du code MAC est très réduite *(4 octets au lieu de 8 ou 16 octets)*, ce qui permet de réduire significativement le surcoût de sécurité. Les auteurs estiment que ce code simplifié peut satisfaire toutes les conditions de sécurité, étant donné qu'un attaquant doit essayer $2^{32}$ combinaisons pour trouver le bon code. De plus, avec le débit limité des réseaux de capteurs *(40 tentatives par seconde)*, il faut plus de 2 mois pour envoyer les $2^{32}$ combinaisons. Formellement, le nœud récepteur va consommer toutes ses réserves d'énergie avant de pouvoir réceptionner toutes les combinaisons de code. En utilisant un simple mécanisme de contrôle *(nombre de tentatives d'authentification autorisées)*, les auteurs démontrent que leur protocole peut facilement surmonter les tentatives de déchiffrement du code MAC.

5.2.1.5 Le protocole MiniSec *(Mini security):* Ce protocole [46] a été introduit par la même équipe qui a développé le protocole SPIN. L'idée de base est de proposer un nouveau mécanisme d'authentification qui consomme moins de ressources comparé au protocole TinySec. Pour cela, les auteurs optent pour l'utilisation d'un nouvel algorithme de cryptographie symétrique, à base de chiffrement par bloc. Ce dernier est nommé OCB *(Offset Code Book)* [47]. Le protocole MiniSec utilise deux mécanismes d'authentification : le premier est destiné à l'authentification unicast *(MiniSec-U)*, et l'autre à l'authentification par diffusion *(MiniSec-B)*.Comparé aux protocoles précédents, MiniSec offre un grand niveau d'authentification et de confidentialité, avec moins de consommation en ressources. En effet, le protocole MiniSec consomme le tiers de l'énergie consommée avec le protocole TinySec-AE, ce qui le rend l'un des protocoles d'authentification les plus utilisés dans les RCSFs.

### 5.3 Les modèles de confiance :

Afin de renforcer le niveau de sécurité, plusieurs recherches proposent des mécanismes de sécurité, basés sur l'indice de confiance et de réputation. Ces nouveaux mécanismes permettent de protéger le réseau des attaques très malicieuses, contre lesquelles les solutions cryptographiques ne peuvent rien faire. En effet, les mécanismes à base d'indice de confiance peuvent consommer beaucoup de ressources, puisqu'ils imposent aux nœuds du réseau d'échanger périodiquement un grand nombre de paquets de contrôle. Par conséquent, le développement d'un modèle de confiance avec une basse consommation de ressources devient un véritable chalenge.

- Pirzada et Mcdonald ont proposé une approche d'établissement de relation de confiance entre les différents nœuds du réseau [48]. Dans ce modèle, chaque nœud doit surveiller les paquets reçus ou retransmis par les autres nœuds du réseau. Les activités de réception et de retransmission des paquets sont définies comme des événements. Chaque événement est observé puis quantifié avec un certain poids *(un indice de confiance)*, qui dépend de son type d'application. Ainsi, l'indice de confiance d'un nœud est calculé par l'agrégation de tous les poids d'événements affectés par ces nœuds contrôleurs. Ensuite, les chemins de routage seront établis en se basant sur l'indice de confiance des nœuds dans le réseau.

- Un autre modèle de confiance à base de recommandation a été proposé dans [49]. Afin d'établir une relation de confiance, chaque nœud va calculer l'indice de confiance de son proche voisin, en se basant sur les informations statistiques, la valeur des données et la recommandation de ses autres nœuds voisins. Dans [50], les auteurs proposent une nouvelle technique d'établissement de relation de confiance. L'idée de base est l'utilisation d'un modèle de confiance distribué afin de calculer d'une manière probabiliste l'indice de confiance et de réputation. A l'étape d'initialisation, un nœud négociateur est sélectionné afin de propager et d'initialiser les indices de confiance. Par la suite, une chaine de confiance sera formée entre les nœuds du réseau, afin de sécuriser l'échange de données. Enfin, une approche d'auto organisation est proposée pour réorganiser le réseau conformément aux relations de confiance entre les nœuds du réseau.

- Un nouveau modèle de confiance a été proposé dans [51], afin d'isoler géographiquement les nœuds malicieux. Les auteurs proposent l'utilisation d'une approche cryptographique pour calculer l'indice de confiance de chaque nœud. Ainsi, l'emplacement du nœud est considéré comme non sécurisé *(à éviter)* si son indice de confiance est inférieur à un certain seuil *(fixé par les auteurs)*. Un autre protocole nommé PET *(Personalized Trust model)* a été proposé dans [52]. Ce dernier propose de calculer les indices de confiance en se basant sur l'agrégation de différents paramètres reçus par les nœuds voisins. Les auteurs ont mis le point sur l'influence de la capacité de mémorisation et de calcul pour le bon établissement du modèle de confiance [53]. Pour cela, ils proposent de ne pas retraiter les valeurs redondantes envoyées par différents nœuds voisins. Par conséquent, le protocole PET offre un modèle de confiance assez fiable, avec une consommation raisonnable des ressources.

### 5.4 Les systèmes de détection d'intrusions :

Tous les mécanismes de sécurité présentés précédemment sont mis en œuvre afin d'empêcher les nœuds malveillants de s'infiltrer dans le réseau. Cependant, ces mécanismes à eux seuls ne suffisent pas pour garantir une sécurité optimale du réseau. En effet, un nœud attaquant peut compromettre un nœud légitime afin de s'infiltrer dans le réseau. Le nœud attaquant utilisera par la suite toutes les informations capturées du nœud victime pour surpasser les contrôles d'authentification et décrypter toutes les informations codées. Ainsi, une deuxième ligne de défense est nécessaire afin de garantir un grand niveau de sécurité. Cette deuxième contre mesure représente un système de détection d'intrusions qui s'occupe de la détection et de la prévention des infiltrations mal veilleuses.

Un système de détection d'intrusion *(SDI)* se charge de surveiller les comportements et les activités suspectes des nœuds dans le réseau [54]. Une activité est considérée comme suspecte si elle n'appartient pas à l'ensemble des activités normales et attendues. Les SDIs assument qu'il existe une différence perceptible entre les comportements d'un nœud légitime et un autre malveillant. En se basant sur le modèle d'analyse et de vérification d'intrusions, les SDIs peuvent être classés en : système basé sur les règles de détection [55, 56], et système basé sur l'anomalie de comportement [57]. En effet, les systèmes basés sur les règles de détection sont utilisés pour détecter les modèles d'intrusions connus à l'avance *(répertoriés dans une base de connaissance)*. Par contre, les systèmes basés sur les anomalies de comportement sont utilisés pour la détection

de nouvelles intrusions *(non répertoriées)*. Comparés aux systèmes à base d'anomalie, les systèmes à base de règles possèdent un faible taux de fausse détection. Cependant, les systèmes à base d'anomalie offrent un taux de détection élevé par rapport à ceux basés sur les règles de détection.

Assurément, la mise en œuvre d'un SDI adapté aux contraintes des RCSFs est un véritable défi. Les réseaux de capteurs sont généralement dédiés à des applications spécifiques et ne possèdent pas de comportements prédictibles. Par conséquent, il est impraticable d'installer et d'initialiser un SDI avec un ensemble de pré-connaissances avant le déploiement. En outre, en raison des limitations physiques, les activités d'apprentissage et de détection d'intrusions sont très coûteuses après le déploiement *(puissance de calcul, et consommation énergétique)*. De ce fait, les SDIs existants *(proposés pour les réseaux Ad hoc)* ne sont pas adaptés aux réseaux de capteurs sans fil.

Les propositions d'SDIs pour les réseaux de capteurs restent très préliminaires, ce qui ouvre plusieurs issues de recherche. Dans le chapitre 3, nous allons classifier et détailler les différentes approches utilisées pour la détection des intrusions, et on présentera les défis auxquels elles sont confrontées. De plus, nous allons mettre le point sur les solutions de détection d'intrusions existantes, en présentant leurs avantages et leurs points faibles.

# 6. LES STRATEGIES DE DEFENSE

Dans cette section, nous discuterons la plupart des stratégies de défense proposées pour contrer les différents types d'attaques présentés précédemment. En fonction du niveau de sécurité désiré, ces stratégies peuvent être construites autour d'un ou de plusieurs mécanismes de sécurité *(cryptage, authentification, …)*. Le tableau 2.3 résume la plupart des attaques et leurs stratégies de contre-mesure.

## 6.1 Stratégies de défense contre les attaques de déni de service

### *6.1.1 Les stratégie de défense au niveau de la couche physique :*

L'attaque de brouillage peut être contrée en employant par exemple les méthodes de variation des fréquences de communication. Parmi ces méthodes, nous pouvons citer la méthode de saut de fréquence FHSS *(Frequency-hopping spread spectrum)* [14]. Celle-ci consiste à transmettre les données en changeant plusieurs fois la fréquence de transmission. Les sauts de fréquence s'effectuent en utilisant une suite de fréquences aléatoires, connue par les deux extrémités de transmission. Etant donné que l'attaquant ne peut pas prévoir la séquence de sélection des fréquences, il lui serait difficile de brouiller les transmissions à un temps donné.

Cependant, l'utilisation de plusieurs fréquences de transmission est très coûteuse pour les RCSFs. Afin de remédier à ce problème, d'autres méthodes proposent d'identifier la région responsable du brouillage, et de l'éviter pendant le routage des données. Wood et Stankovic [14] avaient proposé une approche de défense, dans laquelle les nœuds victimes de brouillage signalent leur statut à leurs voisins. Donc, la région touchée par le brouillage est identifiée, et les paquets sont acheminés sans passer par celle-ci.

| Type d'attaque | Attaque | Stratégie de défense |
|---|---|---|
| **Dénie de service (DOS)** | Brouillage et collision | (1) Saut de fréquences, (2) détection et évitement des zones de brouillage. |
| | Epuisement d'énergie | (1) Authentification, (2) contrôle du taux de réception, (3) contrôle du temps de synchronisation, (4) faux plans de transmission, (5) minimisation des temps d'activation. |
| | Allocation abusive des ressources | (1) Authentification, (2) Contrôle du taux d'allocation. |
| | Usurpation des données | (1) Authentification, (2) indice de confiance. |
| | Routage sélectif | (1) Authentification, routage multi-chemins. |
| | Trou noir | (1) Authentification, (2) routage par nœud intermédiaire, (3) processus de découverte multiple du chemin de routage, (4) routage multi-chemins, (5) multiple station de base. |
| | Trou de ver | (1) Authentification, (2) approche de localisation, (3) temps de transmission. |
| | L'attaque Sybil | (1) Authentification, (2) test à base de canaux de transmission aléatoires. |
| | Inondation par paquet HELLO | (1) Authentification, (2) contrôle du taux de réception. |
| | Informations fabriquées | (1) Authentification, (2) nœuds moniteurs, (3) indice de confiance. |
| | La désynchronisation | (1) Authentification, (2) indice de confiance. |
| | L'inondation | (1) mécanisme à base de puzzles, (2) contrôle du taux de demande de connexion. |
| **Confidentialité et Authentification** | Nœuds répliqués | (1) Authentification, (2) indice de confiance, (3) approche de localisation. |
| | Espionnage et analyse du trafic | (1) Cryptage des données, (2) routage multi chemins, (3) chemins de routage virtuels, (4) fausses zones de trafic. |
| **Intégrité** | Altération et modification | (1) Authentification, (2) cryptage des données, (3) nœud moniteurs, (4) tatouage numérique. |

**Tableau 2.3 :** Attaques et leurs stratégies de contre-mesure

### *6.1.2 Les stratégie de défense au niveau de la couche liaison :*

L'une des défenses classiques contre les attaques de collision est l'utilisation d'algorithmes de correction d'erreurs [14]. Cependant, ces algorithmes sont très coûteux *(puissance de calcul)*, et n'offrent pas de bons résultats en cas d'un grand nombre de collisions. En effet, à ce jour il n'existe pas une solution complète afin de prévenir les attaques de collision. En ce qui concerne les attaques d'épuisement d'énergie, quelques recherches proposent de fixer le nombre de réceptions et de retransmissions possibles. Une autre solution est l'utilisation de la stratégie d'accès par slot de temps TDMA *(time division multiple acces)* [14], dans laquelle chaque nœud n'est autorisé à transmettre que durant son slot de temps. Cependant, cette solution est très vulnérable aux collisions. Dans [58], les auteurs proposent de crypter les messages de contrôle en utilisant le protocole WEP (Wired Equivalent Privacy) [59]. Ainsi, chaque nœud peut authentifier l'origine des messages reçus, rejeter ceux en provenance des nœuds malveillants. Le protocole G-MAC a été proposé dans [18], dans le but de prévenir les attaques de privation de sommeil. L'idée de base est de charger le nœud cluster head de l'authentification des messages de diffusion et la retransmission de ces messages aux nœuds membres du cluster. Par conséquent, les nœuds capteurs ne vont pas épuiser leur énergie en recevant de faux messages de diffusion, car ces derniers seront authentifiés par les CHs.

Afin de contrer les attaques de synchronisation, une nouvelle stratégie de défense a été proposée dans [20]. Cette stratégie consiste à ignorer tous les messages de synchronisation dont le temps d'activation est supérieur à un seuil donné. Cependant, cette solution pénalise les communications locales, et augmente la latence des transmissions. Un autre protocole à base de faux plans de transmission, a été proposé dans [60]. Dans ce protocole, chaque nœud capteur échange de faux plans de transmission avec ses nœuds voisins dans le but d'éviter les attaques de collision. En interceptant ces faux plans de transmission, le nœud attaquant croit qu'il connait les périodes d'activation des nœuds capteurs, et essaye donc de créer des collisions avec ces derniers. Cependant, la création et la diffusion des faux plans de transmission introduisent plus de consommation en ressources. Dans [61], les auteurs proposent une combinaison de plusieurs stratégies de défense contre les attaques de privation de sommeil. Cette combinaison comporte un protocole de gestion de clés, un mécanisme d'authentification et un dispositif contre la retransmission et le brouillage des données.

Rainer Falk a proposé un plan d'activation sécurisé, dans lequel les nœuds s'activent uniquement s'ils possèdent un jeton d'activation [62]. Ce dernier sera transmis d'une manière sécurisée en se basant sur un mécanisme d'authentification. L'auteur suppose qu'en utilisant cette approche, le réseau sera sécurisé contre les attaques d'épuisement d'énergie. Un mécanisme d'authentification basé sur la génération et l'échange de codes secrets *(MAC)*, a été proposé dans [63]. En utilisant ce dernier, les nœuds capteurs peuvent authentifier les nœuds malicieux et contrer ainsi les attaques d'épuisement d'énergie.

### *6.1.3 Les stratégie de défense au niveau de la couche réseau :*

Il existe plusieurs mécanismes pour contrer les attaques d'usurpation et d'altération des données. L'utilisation des protocoles d'authentification à base de code MAC, représente l'un de ces mécanismes. En outre, la détection et la prévention des attaques de rediffusion des données peut s'effectuer en introduisant par exemple un simple compteur de messages [41]. Ce dernier doit être en général inférieur à un seuil donné. En ce qui concerne les attaques de routage sélectif, des recherches proposent d'utiliser le routage multi-chemins afin de les contrer [13]. D'autres solutions proposent de détecter le nœud malveillant et de l'exclure définitivement des chemins de routage [64].

Afin de contrer les attaques de trou de ver, plusieurs recherches ont proposé des stratégies de défense basées sur les systèmes de localisation. Dans [65], les auteurs proposent d'utiliser une approche de localisation afin d'estimer les distances entre les nœuds voisins. Ensuite, un schéma virtuel du réseau est construit et analysé afin de détecter d'éventuels trous de ver *(distorsion des distances)*. Un deuxième protocole nommé WODEM a été proposé dans [66], dans le but de remédier aux attaques de trou de ver. L'idée de base est l'utilisation d'un ensemble de nœuds détecteurs, équipés avec des dispositifs de localisation. Ces derniers se chargent de détecter les nœuds trou de ver, et de les signaler aux autres nœuds du réseau. Dans [67], les auteurs proposent une solution originale pour remédier aux attaques de trou de ver. Le principe est d'inclure le temps d'envoi dans les paquets de données, et ce dernier sera comparé au temps de réception. En se basant sur les distances supposées et la durée de transmission, le nœud récepteur peut détecter un éventuel trou de ver. Cependant, cette solution doit assumer que les horloges des nœuds sont synchronisées. De plus, elle n'est pas efficace dans le cas de transmission multi sauts.

L'attaque Sybil représente un autre défi pour la sécurité du réseau. Dans la littérature on trouve quelques recherches qui traitent ce problème, dont la majorité sont basées sur des solutions cryptographiques. Dans [23], les auteurs proposent un test radio, dans lequel chaque nœud assigne à ses voisins plusieurs canaux de transmission. Ce dernier teste l'identité de ses nœuds voisins en envoyant un paquet de contrôle dans l'un des canaux de transmission *(sélectionnés aléatoirement)*. Dans le cas où le nœud récepteur ne détectera pas la transmission de contrôle, il sera identifié comme nœud Sybil. Les auteurs proposent une deuxième solution, basée sur la distribution de clé aléatoire [37, 38]. En utilisant cette technique d'établissement de clé, chaque nœud va associer son identité avec sa clé respective ce qui permet de prévenir la duplication et la fabrication des identités. Un autre protocole a été proposé dans [68], afin de sécuriser le réseau contre les attaques Sybil. Ce protocole consiste à utiliser les certificats d'identité, en se basant sur un système de cryptographie symétrique. Ainsi, chaque nœud doit envoyer avec son identité, un certificat qui prouve l'authenticité de celle-ci. Dans [69], les auteurs proposent une autre approche de sécurité hiérarchique, basée sur le même principe du protocole précédent.

Le problème de sécurité contre les attaques de trou noir, a été le sujet de plusieurs recherches. Dans [70], un mécanisme de sécurité a été proposé pour réduire l'effet de l'attaque de trou noir. En cas de trou noir, un nœud intermédiaire est sélectionné pour retransmettre les données vers la destination. Cependant, ce dernier peut être un nœud malicieux qui va générer un autre trou noir. Pour cela, les auteurs proposent une stratégie de contre mesure, dans laquelle le nœud émetteur envoi une demande de découverte de route pour le deuxième nœud dans le chemin de routage. Par conséquent, si le deuxième nœud confirme la route proposée par le premier nœud, ce dernier sera authentifié comme nœud sain, sinon il sera rejeté et un autre nœud intermédiaire est sélectionné. Un autre mécanisme a été introduit dans [71], dans lequel les nœuds capteurs s'auto surveillent afin de détecter les attaques de trou noir et de routage sélectif. Par contre, ce mécanisme exige l'échange d'un grand nombre de messages de contrôle *(processus de vote)*, ce qui est très coûteux en termes de ressources. Dans [72], les auteurs proposent d'utiliser les techniques de routage multi chemins, afin de contrer les attaques de trou noir. Ainsi, le protocole MTRP *(Multicast Tree Assisted Random Propagation)* a été introduit, et dont le principe consiste à utiliser aléatoirement plusieurs chemins pour le routage de données.

Le protocole HSRBH *(Hierarchical secure routing protocol called)* [73], est un autre protocole de sécurité contre les attaques de trou noir. Pour construire une route sécurisée, HSRBH utilise un système d'authentification à base de clés asymétriques. Par conséquent, les nœuds malveillants seront détectés et écartés définitivement du chemin de routage. Cependant, les algorithmes de cryptographique asymétriques exigent beaucoup de ressources, ce qui n'est pas le cas dans les RCSFs. Une autre solution anti trou noir a été proposée dans [74]. Les auteurs proposent l'utilisation de plusieurs stations de base afin de réduire l'effet des trous noirs dans le réseau. Cependant cette solution est très consommatrice en énergie ; de plus, si le nœud malveillant se positionne au niveau des stations de base, il pourra facilement surmonter cette contre mesure.

En ce qui concerne l'attaque d'inondation par paquets de HELLO, la plupart des recherches proposent des mécanismes de détection et de prévention qui se basent sur l'analyse du trafic. Par exemple, un nœud proche de l'attaquant peut signaler que le nombre de messages reçus de ce

dernier est supérieur à la normale. En recevant le signal d'alarme, les autres nœuds vont rejeter tous les paquets envoyés par le nœud attaquant. Les attaques d'information fabriquées peuvent être mitigées en utilisant des systèmes de détection à base de nœuds moniteurs [75, 76]. En effet, les nœuds victimes peuvent ne pas réussir à router leurs données à cause des fausses informations de routage qu'ils ont reçu. Ainsi, ces anomalies de routage facilitent la détection des attaques d'informations fabriquées. Les mécanismes de sécurité à base d'indice de confiance peuvent être aussi utilisés contre les attaques d'informations fabriquées [77, 78]. En effet, le nœud récepteur décide de prendre ou pas en compte les informations reçues, en fonction de l'indice de confiance du nœud émetteur. Une autre solution plus coûteuse est l'utilisation des mécanismes d'authentification à base de clés secrètes.

#### *6.1.4 Les stratégie de défense au niveau de la couche transport :*

Pour se défendre contre les attaques d'inondation, un mécanisme à base de puzzles à été proposé dans [79]. L'idée de base est que chaque nœud doit résoudre une sorte de puzzle afin de pouvoir établir une connexion avec le nœud destinataire. Ainsi, le nœud attaquant ne peut pas lancer rapidement et successivement plusieurs demandes de connexion dans le but d'épuiser les ressources disponibles. En ce qui concerne les attaques de désynchronisation, la plupart des recherches proposent l'utilisation des mécanismes d'authentification à base de code MAC [14]. En authentifiant les nœuds attaquants, le nœud récepteur évitera de gaspiller son énergie en tentant de recevoir des données altérées, ou de réparer des erreurs de transmission qui n'ont jamais vraiment existé.

### 6.2 Stratégies de défense contre les attaques de confidentialité et d'authentification :

Dans [24], les auteurs ont proposé un mécanisme de détection distribué contre les attaques des nœuds répliqués *(nœuds clonés)*. En utilisant un système à base de clés de groupe, chaque nœud va diffuser les coordonnées de sa position à tous ses nœuds voisins. Dans le cas où ces derniers reçoivent des coordonnées différentes appartenant à un seul nœud, ils signalent la détection d'un nœud répliqué. Les auteurs assument que leur protocole offre un taux de détection de 100 %, dans le cas où la diffusion *(des informations de position)* va atteindre tous les nœuds du réseau. Cependant, le coût total des communications sera $O(n^2)$, ce qui n'est pas adapté aux RCSFs. SET est un autre protocole dédié à la détection des nœuds répliqués [80]. Ce dernier propose de créer des sous-ensembles de nœuds sécurisés en utilisant un protocole d'authentification à base de code MAC. Les sous-ensembles de nœuds effectuent des opérations d'intersection et d'union avec les autres nœuds du réseau afin de détecter les nœuds clonés. Dans [81], les auteurs proposent une stratégie de défense à base de pré-distribution de clés. L'idée de base est de pré charger chaque nœud capteur avec une clé unique. Ensuite, chaque nœud va envoyer périodiquement son identificateur et sa clé à la station de base. Celle-ci va détecter la présence de nœuds répliqués, dans le cas où il y'a une redondance de clé. Bekara et Maknavicius proposent une nouvelle approche pour la détection des nœuds répliqués [82]. Celle-ci consiste à utiliser un modèle de confiance afin de repérer les identités redondantes. Ainsi, le nœud qui possède un identificateur redondant et un faible indice de confiance sera systématiquement rejeté du réseau.

Deng, Han et Mishra ont proposé un mécanisme de défense contre l'attaque d'analyse du trafic [27]. Ce mécanisme combine plusieurs stratégies de défense afin de brouiller et de prévenir les

tentatives d'analyse du trafic. En premier lieu, les auteurs proposent d'utiliser le routage multi chemins afin de compliquer la tâche d'analyse du trafic. De plus, des chemins de routage virtuels sont crées pour empêcher les attaquants de poursuivre les paquets de données durant leur route vers la station de base. Des zones de faux trafic sont aléatoirement créées, dont le but est de cacher les positions des CHs et de la station de base. Malgré le fait que ce mécanisme offre de bons résultats contre l'attaque d'analyse du trafic, il reste plus au moins coûteux en termes de consommation de ressources. D'autres méthodes plus classiques pour la sécurité contre les attaques d'analyse du trafic, et le cryptage des données, par exemple l'utilisation du protocole SPIN, permettent de rendre l'espionnage et l'analyse du trafic des tâches très difficiles. Cependant, l'attaquant peut facilement détecter le rôle et la position des nœuds importants *(CH, BS)*, en traquant les zones de concentration du trafic.

### 6.3 Stratégies de défense contre les attaques d'intégrité des données :

La plupart des mécanismes de sécurité proposés contre les attaques d'intégrité des données, sont basés sur les protocoles de cryptage et d'authentification. Parmi ces protocoles nous pouvons citer les protocoles SPIN [41], TinySec [45], LEA [44]. Etant donné que ces mécanismes sont plus au moins coûteux en termes de ressource, d'autres recherches proposent l'utilisation des algorithmes de signature ou de tatouage numérique *(Watermarking Techniques)* [83, 84]. Ces derniers sont traditionnellement appliqués dans la protection des données multimédia, comme les images et les vidéo clips. Les algorithmes de tatouage numérique sont simples, et n'exigent pas de grandes capacités de calcul, de mémorisation et d'espace de stockage. Le tatouage consiste à intégrer avec les données une information sécrète *(la signature)*, où une simple modification de données implique la corruption de la signature [85, 86]. Les protocoles SGW *(Sliding Group Watermark)* [87] et LWC (*light-weight chained watermarking)*[88], sont un exemple des mécanismes de sécurité à base de tatouage numérique. Les systèmes de détection d'intrusions peuvent être utilisés pour renforcer la sécurité contre les attaques qui ciblent l'intégrité des données. Basés sur les règles de comportement et de contenu, ces mécanisme analysent et signalent toutes modifications, altérations ou suppressions des données.

## 7. CONCLUSION

Dans ce chapitre, nous avons mis le point sur le problème de sécurité dans les RCSFs. En effet, ce type particulier de réseaux impose des contraintes supplémentaires qui rendent la sécurité des réseaux un véritable défi. En plus d'offrir un bon niveau de protection, les protocoles de sécurité dédiés aux RCSFs doivent respecter les limitations de ressources de ces derniers. Cependant, les contraintes de ressources des RCSFs imposent la simplification des mécanismes de sécurité *(cryptographie et authentification)*, ce qui réduit en conséquence leur niveau de sécurité. C'est pour ce fait qu'une deuxième ligne de défense est sollicitée afin de renforcer le niveau de sécurité dans les RCSFs. Cette deuxième ligne consiste en un système de détection d'intrusions capable de détecter et de prévenir les attaques qui peuvent surpasser les premières mesures de sécurité. Le chapitre suivant sera consacré à la présentation des systèmes de détection d'intrusion, et les contraintes qu'ils peuvent confronter lors de leur application dans les RCSFs.

# Chapitre 3

## *Les systèmes de détection d'intrusions*

# 1. INTRODUCTION

La nature inattendue de l'environnement et la limitation des ressources, rendent les réseaux de capteurs sans fil vulnérables à plusieurs types d'attaques. L'authentification et la cryptographie sont les mécanismes les plus connus pour combattre les menaces de sécurité. Cependant, il est devenu clair que nous ne pouvons pas garantir un niveau de sécurité parfait en utilisant uniquement ces deux mécanismes. En effet, ces derniers ne peuvent pas prévoir tous les types d'attaques, d'autant plus que de nouvelles apparaissent constamment. Par conséquent, il est évident d'implémenter une deuxième ligne de défense pour renforcer le niveau de sécurité dans le réseau. Cette ligne de défense représente un système de détection d'intrusions *(SDI)*, capable de détecter les attaques et d'en informer les nœuds du réseau.

Une intrusion peut être définie comme un accès non autorisé à une ressource du système [89]. La détection d'intrusions est utilisée pour identifier ces intrusions afin de rétablir le fonctionnement normal et d'exclure les utilisateurs illégitimes. Un SDI est composé généralement d'agents qui s'exécutent sur certains ou tous les nœuds du réseau. Traditionnellement, les SDIs sont principalement situés sur des ordinateurs très puissants, et sont en mesure de traiter efficacement toutes les données provenant du réseau sur lequel ils opèrent. Malheureusement, il n'existe pas de tels dispositifs dans le cas des réseaux de capteurs *(limitation de ressources)*. Ainsi, il faut savoir tirer avantage des caractéristiques des RCSFs en termes du grand nombre de nœuds, et remédier aux problèmes de faible puissance de calcul et d'énergie.

# 2. LES SYSTEMES DE DETECTION D'INTRUSIONS

La détection d'intrusions consiste à fournir un système de sécurité qui permet d'identifier spontanément la source des attaques, et d'alarmer le réseau sur d'éventuelles intrusions. Ainsi, l'SDI va sélectionner l'action de prévention appropriée selon la nature de l'intrusion détectée. En effet, n'importe quelle action qui conduit à un accès non autorisé ou à une altération des fonctionnalités du réseau, est considérée comme une intrusion. Dans ce cas, un attaquant peut lancer ses attaques sans avoir une autorisation d'accès *(attaquant externe)*, ou être un nœud légitime qui abuse de son privilège d'accès *(attaquant interne)*.

## 2.1 Techniques de détection d'intrusions

Les systèmes de détection d'intrusions doivent être en mesure de faire la distinction entre les activités normales et anormales afin de découvrir avec un coût optimal n'importe quelle tentative malveillante. Le choix d'une technique de détection est une tâche cruciale qui affecte considérablement le niveau de sécurité. Il existe trois techniques principales pour l'analyse et la détection des actions malicieuses, à savoir : technique de détection à base de signature, technique de détection à base d'anomalies et technique de détection à base de spécification.

***2.1.1 Technique de détection à base de signature (Signature-based detection):*** la détection à base de signature ou d'abus *(misuse detection or signature-based detection)* [55, 56], consiste à comparer le comportement observé avec le comportement des attaques connues *(signatures)*. Par

conséquent, les modèles de comportement qui peuvent constituer une menace pour la sécurité doivent être définis et répertoriés par le système de détection *(base de signature)*. Dans le cas d'une intrusion, l'SDI recherche les occurrences des signatures dans le flux d'événements; s'il trouve une concordance, l'alerte d'intrusion est lancée. Ainsi, tout ce qui n'est pas explicitement défini est autorisé, et tout ce qui est explicitement défini est interdit. Cette technique offre un taux réduit de fausses détections, mais ne donne pas de meilleurs résultats face aux attaques inconnues [90]. De plus, la faible capacité de mémorisation dans les RCSFs rend inefficace l'implémentation de cette technique *(sauvegarde et mise à jour des bases de signature)*.

***2.1.2 Technique de détection à base d'anomalies (Anomaly-based detection):*** contrairement à la technique précédente, l'approche comportementale ou approche par détection d'anomalies [57] se focalise sur les événements normaux au lieu de ceux anormaux. Ainsi, cette technique s'attache à définir le comportement normal du système, et considère comme étant une intrusion, toute déviation par rapport à ce comportement de référence. Donc, tout ce qui n'est pas explicitement défini est interdit, et tout ce qui est explicitement défini est autorisé. Comparée à la détection à base de signature, cette technique présente l'avantage de détecter les attaques inconnues. Cependant, elle est fortement confrontée aux problèmes de fausse détection positive et négative. Dans le cas d'une fausse détection positive, la détection à base d'anomalies détecte une intrusion alors qu'en réalité ce n'en n'est pas une. Ce problème peut être généré dans le cas où la définition de l'ensemble des comportements normaux est incomplète. Le problème de fausse détection négative est posé lorsqu'une intrusion qui aurait due être détectée ne l'a pas été. La détection à base d'anomalies peut provoquer ce problème si le modèle de comportement est construit à l'aide d'un algorithme d'apprentissage, et que durant la phase d'apprentissage un comportement malveillant se produit, ce dernier sera automatiquement inclus dans le modèle et ne déclenchera pas d'alerte durant la phase de recherche d'attaque. En général, on doit faire des précautions lorsqu'on applique la détection à base d'anomalies dans les RCSFs. En effet, il est extrêmement difficile de définir ce qui est un comportement normal dans ce type particulier de réseau. De plus, à cause de leurs limitations énergétiques, les nœuds capteurs ne peuvent pas supporter le coût de l'apprentissage automatique.

***2.1.3 Technique de détection à base de spécification (Specification-based detection):*** cette technique [91, 92] adopte le même principe que la détection à base d'anomalies, selon lequel toute déviation du comportement normal est considérée comme une intrusion. Cependant, la définition du modèle de comportement est réalisée d'une manière manuelle et non pas automatiquement à l'aide d'un algorithme d'apprentissage. Cela permet de simplifier le système de détection, et réduit significativement le taux de fausses détections négatives. Comparée à la détection à base d'anomalies, cette technique semble être la mieux appropriée aux limitations des réseaux de capteurs.

## 2.2 Architecture d'un système de détection d'intrusions

Un système de détection d'intrusions peut être implémenté d'une manière décentralisée sur tous les nœuds du réseau, ou d'une manière centralisée sur un nœud puits qui représente la station de base. Une architecture hybride peut être aussi utilisée, en combinant les deux architectures précédentes.

***2.2.1 SDI à base d'architecture décentralisée :*** avec cette architecture, les intrusions sont détectées localement au niveau de chaque nœud du réseau. Connue aussi sous le nom d'architecture HID *(A host-based intrusion detection)*, elle est considérée comme la première architecture à être exploitée pour la détection d'intrusions. L'architecture décentralisée est très simple, et offre une grande résilience aux attaques de compromission. Cependant, certaines attaques ne peuvent être détectées, étant donné que chaque nœud ne possède qu'une vision partielle du réseau. En outre, la limitation des nœuds capteurs ne permet pas l'utilisation des puissantes méthodes d'analyse et de détection.

***2.2.2 SDI à base d'architecture centralisée :*** dans l'architecture centralisée, toutes les informations pertinentes pour la détection d'intrusions doivent être transférées vers un seul point du réseau, en général la station de base. Contrairement à l'approche décentralisée, la détection d'intrusions est effectuée uniquement sur la station de base, qui est supposée être la plus puissante en termes de mémoire, d'énergie et de puissance de calcul. Par conséquence, cela permet l'utilisation de méthodes de détection plus sophistiquées, et fiabilise le niveau de sécurité. De plus, les systèmes à base d'architectures centralisées disposent d'une vision globale du réseau, offrant la possibilité de détecter des attaques qui seraient restées indétectables dans les architectures décentralisées. Toutefois, le transfert périodique des informations de détection va consommer plus de ressources principalement en matière d'énergie. En outre, cette architecture est très vulnérable aux attaques de compromission, étant donné que la compromission de la station de base met hors service tout le système de détection.

***2.2.3 SDI à base d'architecture hybride :*** l'architecture de détection hybride est une combinaison d'architectures centralisées et décentralisées. Par conséquent, la détection d'intrusions est réalisée à la fois localement et globalement dans le réseau. En effet, chaque nœud dispose d'un agent de détection d'intrusions. Ce dernier transfert ses alarmes à un agent central plus puissant afin de vérifier et confirmer l'intrusion détectée. L'architecture de détection hybride offre un bon compromis entre les deux architectures précédentes en se servant de leurs avantages tout en minimisant leurs inconvénients.

## 2.3 Approches de prise de décision

Dans un système de détection d'intrusions, on doit déterminer la partie responsable des décisions d'intrusions et les actions de contremesure. Pour cela, deux principales approches ont été proposées à savoir : approche de prise de décision indépendante, et approche de prise de décision coopérative.

***2.3.1 Approche de prise de décision indépendante :*** dans une prise de décision indépendante, certains nœuds sont chargés d'effectuer la prise de décision en fonction des événements observés. Ainsi, ces derniers recueillent les informations et les preuves d'activités anormales des autres nœuds, et prennent individuellement les décisions au sujet des intrusions dans le réseau. Cette approche est généralement utilisée dans les SDIs à base d'architectures centralisées, dans lesquelles la station de base se charge de prendre toutes les décisions d'intrusions.

***2.3.2 Approche de prise de décision coopérative :*** dans cette approche, chaque nœud participe à la détection et la prise de décision d'intrusions. Ainsi, tous les nœuds sont chargés de détecter les tentatives d'intrusions au niveau local. Si un nœud n'est pas sûr de l'anomalie qu'il a détectée, ou si la preuve d'intrusion est non concluante, alors un mécanisme de coopération est initié avec ses nœuds voisins afin de collaborer sur la prise de décision d'intrusions. Lors de la conception d'un mécanisme de prise de décision coopérative dans les réseaux de capteurs sans fil, il faut tenir compte du fait qu'un nœud peut être compromis. Par conséquent, ce dernier peut envoyer des données falsifiées vers ses voisins afin d'affecter leur prise de décision. En outre, un mécanisme de décision coopératif doit respecter les limitations en ressources des nœuds capteurs. Les nœuds ne peuvent pas échanger des données de sécurité et des alertes d'intrusion sans considérer l'énergie qui doit être consommée pour l'envoi, la réception et le traitement de ces messages.

### 2.4 La signalisation d'intrusions et démarche de contremesure

Une fois qu'une attaque est détectée, les informations d'intrusions *(alarme)* doivent être transmises aux nœuds de relai et nœuds victimes. Les nœuds de relai sont censés transmettre l'alerte à la station de base ou aux nœuds administrateurs *(par exemple le cluster head)*, qui vont décider de la démarche possible de contremesure. Celle-ci peut consister par exemple à révoquer le nœud malveillant du réseau. Après avoir reçu l'alerte, chaque nœud victime vérifiera sa situation de voisinage et reconstruit sa table de routage pour contourner si nécessaire le nœud malveillant.

### 2.5 Evaluation des performances d'un SDI

Afin d'évaluer les performances d'un système de détection d'intrusions, deux paramètres de mesure sont généralement utilisés, à savoir le taux de faux positifs *(FP)*, et le taux de faux négatifs *(FN)*. En effet, le taux de faux positifs est défini comme l'ensemble d'événements normaux qui sont classifiés à tort comme des événements anormaux. Par contre, le taux de faux négatifs est défini comme l'ensemble d'événements anormaux qui sont classifiés par erreur comme des événements normaux. De toute évidence, un système de détection d'intrusions devrait avoir un faible taux de faux positifs et négatifs. Cependant, un compromis est généralement à faire entre FP et FN, compte tenu que la réduction de l'un va engendrer l'augmentation de l'autre.

Dans les RCSFs, d'autres paramètres de performance doivent être considérés. En effet, la surcharge causée par le processus de détection d'intrusions représente un paramètre qui affecte significativement les performances d'un SDI. De plus, l'extrême limitation des ressources rend la faible consommation *(énergie, puissance de calcul, bande passante, capacité de stockage…)* un des paramètres crucials dans l'évaluation des performances.

## *3. SYSTEME DE DETECTION D'INTRUSIONS POUR LES RCSFs*

L'implémentation d'un mécanisme de détection d'intrusions dans les RCSFs, constitue un vrai chalenge. En effet, la plupart des communications s'effectuent d'une manière multi-sauts, ce qui facilite l'infiltration des nœuds malicieux. En outre, la topologie dynamique et le déploiement

aléatoire du réseau, permet aux attaquants de s'injecter dans le réseau. Par conséquent, il est extrêmement difficile de définir ce qui est un comportement normal, dans ce type particulier de réseaux. La limitation des ressources des nœuds capteurs est une autre contrainte imposée aux systèmes de détection d'intrusions. Cette contrainte rend difficile voire impossible, l'utilisation des SDIs traditionnels dans les RCSFs. Donc, un système de détection d'intrusions dédié aux RCSFs, doit généralement satisfaire les propriétés suivantes :

- ***Faible consommation en ressources* :** un SDI pour les réseaux de capteurs doit gérer efficacement les ressources disponibles dans le réseau. Ainsi, la tâche de détection ne doit pas exiger l'échange d'un grand nombre de messages entre les nœuds capteurs, étant donné que celle-ci va consommer beaucoup d'énergie et de bande passante. De plus, les techniques d'analyse d'intrusions doivent être simples et adaptées aux faibles puissances de calcul des nœuds capteurs.

- ***La distributivité* :** Le processus de collecte et d'analyse des données doit être effectué sur un certain nombre de destinations afin de répartir la charge de détection d'intrusions.

- ***Méfiance des autres nœuds* :** Dans les SDI collaboratifs, les nœuds capteurs doivent assumer l'existence probable des nœuds malveillants dans l'ensemble des nœuds détecteurs. Contrairement aux réseaux Ad-hoc, les nœuds capteurs peuvent être facilement compromis. Ces nœuds peuvent se comporter normalement en respectant les conditions d'acheminement d'information, afin d'éviter d'être détectés par les SDIs. Cependant, ils peuvent entraver la détection d'un autre nœud intrus. Par conséquent, chaque nœud détecteur ne doit pas faire totalement confiance durant sa collaboration avec les autres nœuds dans le réseau.

- ***L'ajout de nouveaux nœuds* :** l'ajout de nouveaux nœuds après le déploiement est une opération fréquente dans un réseau de capteurs. Ainsi, un SDI doit être en mesure de supporter cette opération et de la distinguer des attaques d'infiltration des nœuds malveillants.

- ***L'auto défense*:** Un SDI doit s'auto protéger contre les attaques hostiles le ciblant. La compromission d'un nœud de surveillance ne doit pas permettre à l'attaquant de révoquer un nœud légitime du réseau ou de perturber la détection des nœuds intrus.

## 4. ETAT DE L'ART SUR LES SDIs PROPOSES POUR LES RCSFs

Le développement et l'implémentation des systèmes de détection d'intrusions dédiés aux RCSFs, constitue un domaine de recherche très récent. En effet, les recherches sont encore dans leurs états primitifs, et ne traitent pas tous les aspects du problème de détection. Les techniques proposées pour l'analyse et la détection d'intrusions, sont généralement des adaptations de celles proposées pour les réseaux Ad-hoc. De plus, la plupart des propositions se concentrent sur la détection d'un genre particulier d'attaques, ce qui les rend moins efficaces. Dans la section

suivante, on va résumer la majorité des solutions proposées pour la détection d'intrusions dans les RCSFs.

La tentative d'application de la technique de détection à base d'anomalies dans les RCSFs, a été présentée pour la première fois dans [93]. Les auteurs assument la présence de quelques nœuds moniteurs dans le réseau, qui sont responsables de la surveillance de leurs nœuds voisins. Ces derniers écoutent et analysent tous les messages qui circulent dans leur zone de couverture radio, afin de s'informer sur des nœuds surveillés. L'algorithme de détection est basé sur un ensemble de règles de comportement, à savoir:

- Le taux d'envoi de messages doit respecter un certain seuil.
- Les messages relayés ne doivent pas être altérés
- La retransmission des messages est bornée par un temps donné
- La transmission d'un même message ne doit pas être effectuée plusieurs fois

Basé sur les règles de comportement normal, l'SDI proposé essaie de détecter certains types d'attaques, comme la répétition de transmission, l'altération des données, le trou noir et le renvoi sélectif. D'après les résultats expérimentaux présentés par les auteurs, on peut conclure que la capacité de mémorisation *(pour stocker les messages surveillés)* est un facteur important, qui influence significativement le nombre de fausses détections positives. Par conséquent, la limitation de mémoire des nœuds capteurs réduit l'efficacité de détection de l'SDI proposée *(réduire le nombre de messages stockés va minimiser les capacités de détection d'intrusions)*.

Onat et Miri proposent un autre système de détection pour les RCSFs [94], basé sur une approche similaire à celle de l'SDI précédent [93]. Dans cette proposition, chaque nœud enregistre dans un baffer de taille fixe, tous les paquets reçus par ses voisins ainsi que le temps d'arrivée et la puissance de réception correspondants à ces paquets. Ainsi, une intrusion est signalée si le taux de paquets reçus et le temps et la puissance de réception des paquets dépassent les seuils de sécurité. Deux autres SDIs dédiés aux attaques de routage dans les réseaux de capteurs ont été décrits dans [95] et [96]. Ces derniers supposent que les protocoles de routage pour les réseaux ad hoc peuvent également être appliqués aux réseaux de capteurs. Ainsi, les auteurs du premier SDI [95], proposent une stratégie de détection basée sur le protocole AODV *(Ad hoc On-Demand Distance Vector)*, tandis que l'SDI proposé par Bhuse et Gupta [96] utilise les protocoles DSDV et DSR. Le point commun de ces deux SDIs est qu'ils proposent une stratégie de détection basée sur les caractéristiques spécifiques du protocole de routage, comme le nombre reçu de demandes de route. Cependant, l'application des protocoles de routage pour les réseaux Ad-hoc dans les réseaux de capteurs est un cas très rare, ce qui rend les SDIs proposés moins attrayants.

La recherche présentée dans [97], propose une approche de détection d'intrusions basée sur les signatures. Cette approche est basée sur une architecture de détection centralisée, dans laquelle un seul nœud est chargé de la surveillance et la détection d'intrusions. Cependant, l'efficacité de l'SDI est fortement liée au bon choix du nœud hôte *(le nœud sur lequel s'installe l'SDI)*. Ce dernier doit permettre que tous les paquets soient contrôlés au moins une fois. De plus, il doit offrir une bonne résilience aux attaques de compromission.

Un système de détection d'intrusions à base d'une architecture distribuée a été proposé dans [98]. Dans cette proposition, chaque nœud capteur possède un agent de détection. Ce dernier peut être un agent local ou global. Les agents locaux sont chargés seulement de surveiller et d'analyser les sources d'informations locales afin de détecter les attaques d'intrusions au niveau de chaque nœud. Par contre, les agents globaux sont actifs uniquement sur un sous-ensemble de nœuds, et sont en charge d'analyser les paquets de leur voisinage. Afin de réduire le coût de surveillance, une nouvelle approche de détection basée sur les clusters a été proposée dans [99]. L'idée de base est de diviser le réseau en clusters, dans lesquels un chef de cluster *(cluster head)* est sélectionné pour surveiller ses nœuds membres. Les nœuds membres d'un cluster sont divisés à leur tour en groupes qui se relaient pour surveiller le cluster head. Ainsi, cette approche permet de distribuer la charge de surveillance sur le réseau, ce qui réduit en conséquence la consommation des ressources.

Dans [100], les auteurs proposent un système de détection d'intrusions dédié aux attaques de trou de puits *(Sinkhole)*. La première étape de détection consiste à trouver une liste de nœuds suspects, grâce à l'estimation de la zone attaquée. Ainsi, les auteurs supposent que la station de base a une connaissance approximative de l'emplacement de tous les nœuds capteurs. En outre, la station de base utilise une méthode d'analyse statistique afin de détecter les incohérences des données. Par conséquent, un nœud est signalé comme suspect si le taux d'incohérences est supérieur à un seuil donné. Après la détection des nœuds suspects, la station de base va identifier la zone contenant ces derniers comme une zone potentiellement attaquée. Le nœud intrus sera identifié grâce à l'analyse du routage dans la zone attaquée. Pour cela, un message de requête contenant les identifiants de tous les nœuds suspectés sera diffusé par la station de base. Ce dernier est signé avec la clé privée de la station de base. En recevant le message de contrôle, les nœuds concernés répondent en envoyant leurs propres identifiants, l'id du nœud suivant dans leur chemin de routage et le coût de routage *(nombre de sauts)*. En se basant sur les messages de contrôle reçus, la station de base va reconstruire l'arbre de routage afin d'analyser les incohérences de routage. Ainsi, l'intrus sera rapidement identifié, étant donné que tout le trafic dans la zone attaquée circule vers lui.

Une approche pour la détection d'attaque de routage a été présentée dans [101]. L'idée de base est l'utilisation d'un algorithme basé sur les clusters, afin de construire un modèle de comportement normal du trafic. Ainsi, cette approche adopte une technique de détection par anomalie, pour détecter les intrusions possibles dans le réseau. Cela permet de détecter un large éventail d'attaques de routage, sans pour autant être obligé de définir la signature de ces dernières *(détection des attaques inconnues)*. Les auteurs assument que les décisions d'intrusions se font d'une manière indépendante, et ne nécessitent pas de communications entre les nœuds capteurs, ce qui réduit significativement la consommation d'énergie. L'algorithme d'analyse et de détection d'intrusions se base uniquement sur les informations locales du nœud, tel que la table de routage et les paquets reçus par ce dernier.

Un système de détection et de prévention d'intrusions hybride à faible consommation énergétique a été introduit dans [102]. Ce dernier est nommé eHIP *(Energy-efficient Hybrid Intrusion Prohibition system)*. Les auteurs supposent une topologie de routage à base de clusters, dans laquelle les données sont acheminées à la station de base à travers les têtes de clusters. eHIP propose l'utilisation de deux mécanismes d'authentification afin d'empêcher toutes les tentatives

d'intrusions. Le premier mécanisme est utilisé pour authentifier l'origine des messages de contrôle, tandis que le deuxième est destiné à l'authentification des paquets de données. Pour détecter les attaques d'intrusions, les auteurs mettent en œuvre un système de détection collaboratif pour surveiller les clusters heads ainsi que leurs nœuds membres. Ainsi, les nœuds membres coopèrent afin de détecter les anomalies dans leur cluster head, alors que ces derniers sont chargés de surveiller les incohérences de leurs nœuds membres. Les auteurs affirment que leur SDI est capable de détecter les attaques d'altération, de duplication et de brouillage de paquets, mais aucune précision n'a été donnée. Leur simulation met l'accent sur la gestion efficace d'énergie par eHIP, et ne présente pas de résultats sur les performances de détection.

En n'utilisant que les informations partielles et locales, Krontiris, Dimitriou et Felix ont proposé une nouvelle approche de sécurité pour la détection des attaques de trou noir et de renvoi sélectif [71]. Dans cette approche, chaque nœud surveille son voisinage et collabore avec ses voisins les plus proches afin de détecter les anomalies d'intrusions. Ces dernières sont signalées dans le cas où il existe des incohérences par rapport au comportement normal, initialement définies à l'aide d'un ensemble de règles *(i.e. taux de messages rejetés)*. Ainsi, en suivant une approche à base de règles de spécification, les auteurs assument que le nœud attaquant sera identifié, si plus de la moitié des nœuds de surveillance déclenchent une alerte d'intrusions pour ce nœud. Une version étendue de cette approche de détection a été proposée dans [103], afin de détecter les attaques de trou de puits.

Fang, Xiuzhen et Dechang ont proposé un algorithme de détection d'intrusions externe, basé sur les informations locales de comportement [104]. Afin de détecter les intrusions, l'algorithme proposé explore l'existence de corrélations spatiales entre les nœuds capteurs. De plus, la charge de calcul et de communication devrait être équivalente entre les nœuds voisins. Par conséquent, un comportement malveillant est indiqué par dérogation aux caractéristiques précédentes (*corrélation spatiale et déséquilibre de répartition de charge)*. Les auteurs adoptent une architecture de détection distribuée, dans laquelle chaque nœud capteur est chargé de surveiller le comportement de ses proches voisins. En effet, l'algorithme de détection est composé de quatre phases à savoir: la collecte d'informations locales, le filtrage des données recueillies, l'identification des valeurs aberrantes, et enfin, l'application du vote majoritaire pour obtenir une liste définitive des nœuds intrus. Afin de supprimer des informations falsifiées par les attaquants, les auteurs proposent de filtrer les données recueillies. Ainsi, une valeur de confiance est attribuée à chaque voisin dans l'intervalle [0, 1], où une valeur proche de 1 indique un risque d'avoir un nœud intrus. Cette valeur de confiance est attribuée en fonction du degré de déviation du comportement du nœud, par rapport au comportement de ses nœuds voisins.

Dans [105], un système d'apprentissage automatique a été proposé pour les systèmes de détection d'intrusions. Ce dernier consiste à mettre en œuvre un agent de détection d'intrusions au niveau de chaque nœud dans le réseau. Pour détecter une intrusion, les nœuds capteurs surveillent continuellement le trafic de leurs voisins. En outre, les décisions d'intrusions se font d'une manière indépendante *(il n'y a pas de coopération entre les nœuds moniteurs)*. L'agent de détection d'intrusions commence le processus de détection en repérant si le nœud lui-même est attaqué. Pour cela, un composant de détection d'intrusions local *(LIDC: Intrusion Detection Component)* a été proposé afin d'analyser les paramètres locaux de sécurité tel que : le taux de collision, temps de latence, le taux des paquets RTS *(Request to send)*, nombre des nœuds voisins,

le coût de routage et le taux de consommation d'énergie. Pour surveiller et détecter les nœuds intrus parmi ses nœuds voisins, le nœud capteur utilise un deuxième composant de détection d'intrusions basé sur les paquets surveillés *(PIDC : Packet based Intrusion Detection Component)*. Ce dernier consiste à analyser les paramètres des paquets surveillés, à savoir : la force du signal du paquet reçu, le taux de réception des paquets, taux de rejet et de retransmission de paquets. Les auteurs adoptent l'utilisation de la méthode d'apprentissage *SLIPPER* [106], afin de construire le modèle de comportement normal de leur SDI.

Une technique de détection d'intrusions collaborative et généraliste a été proposée dans [107]. L'approche proposée ne se concentre pas sur la détection d'attaques spécifiques, mais plutôt sur les techniques de coopération. Le problème de la détection d'intrusions est formellement défini et des conditions nécessaires et suffisantes pour la détection d'intrusions coopératives sont identifiées. Les auteurs proposent un algorithme de détection assez léger, qui peut fonctionner sur un seul nœud capteur. Cependant, la technique de détection proposée a été expérimentée sur un seul type d'attaquant, ce qui la rend moins crédible.

Dimitriou et Giannetsos proposent un algorithme basé sur la localisation afin de détecter les attaques de trou de ver [108]. L'idée de base consiste à utiliser un graphe de communication sous-jacente afin de détecter les incohérences de connectivité entre les nœuds du réseau. Les auteurs supposent l'existence d'un intervalle de temps sécurisé *(sans attaques)*, dans lequel les nœuds vont collecter leurs informations de voisinage afin d'établir un graphe de communication sous-jacente. Un test de vérification doit être exécuté, dans le cas où un nœud veut ajouter un nouveau nœud à sa table de routage. Ce dernier consiste à trouver un chemin de routage sécurisé *(en utilisant le graphe de communication sous-jacente)*, vers le nouveau nœud. Dans [109] les auteurs présentent un système de détection d'intrusions hybride, basé sur la combinaison d'architectures de détection centralisée et décentralisée. Dans cette architecture, la détection d'intrusions est réalisée à l'aide de deux types d'agents. Le premier type d'agents opère localement au niveau de chaque nœud dans le réseau, tandis que le deuxième est un agent qui opère globalement et se situe uniquement sur un seul nœud central *(souvent la station de base)*. L'agent local est chargé d'analyser les paquets routés à travers les nœuds capteurs, afin de détecter les anomalies d'intrusions. Ces anomalies sont signalées par la suite à l'agent central, qui se charge de les vérifier en utilisant un modèle. Cependant, les auteurs ne donnent pas de détails sur la façon selon laquelle les anomalies sont détectées.

Un système de détection d'intrusions dédié pour les réseaux de capteurs a été proposé dans [110], afin de détecter les attaques de routage dans les topologies d'organisation en clusters. Les auteurs proposent une architecture de détection hybride, dans laquelle un agent de surveillance locale est installé sur tous les nœuds du cluster, tandis qu'un agent de surveillance globale est implémenté dans le nœud cluster head. Ainsi, tous les paquets envoyés et reçus par un nœud capteur sont analysés par l'agent local. De plus, toutes les communications entre les nœuds du même cluster seront contrôlées par l'agent global. La décision d'intrusions est effectuée par le nœud cluster head en fonction du nombre d'alertes reçues par ses nœuds membres *(le nombre d'alertes concernant un nœud spécifique est supérieur à un certain seuil donné)*. Dans ce cas, une liste noire est mise à jour et transférée à tous les nœuds du cluster, afin d'isoler le nœud attaquant. Les auteurs supposent que leur SDI est capable de détecter les attaques de renvoi sélectif, de trou de ver, de trou de puits et d'inondations par paquet Hello.

## 5. DISCUSSION

Le problème de détection d'intrusions dans les réseaux de capteurs sans fil est fortement lié au respect des limitations des ressources de ces derniers. En effet, un SDI dédié aux RCSFs doit offrir un bon niveau de sécurité tout en garantissant une gestion efficace des ressources disponibles. Cependant, le niveau de sécurité est inversement proportionnel au degré de consommation. Par conséquent, une optimisation du niveau de sécurité va engendrer, dans la plupart des cas, une augmentation significative de la consommation en ressources. Ainsi, un compromis doit être fait entre ces deux paramètres l'or du développement d'un système de détection d'intrusions pour les RCSFs. Les systèmes de détection d'intrusions présentés précédemment tentent d'établir ce compromis en se basant sur différentes techniques et stratégies de détection. Néanmoins, la plupart de ces SDIs proposent des solutions de sécurité contre un seul type d'attaques, ou un ensemble d'attaques qui cible une couche spécifique dans le modèle OSI. Pour cela, ces derniers sont considérés comme des systèmes mono couche, qui opèrent au niveau d'une seule couche du réseau, sans prendre en compte les attaques qui peuvent cibler les autres couches adjacentes. Ainsi, ces systèmes de sécurité restent très vulnérables, et n'offrent pas un niveau de sécurité optimal.

Afin de résoudre ce problème, des recherches proposent de combiner différents SDIs mono couche pour détecter les attaques d'intrusions au niveau de plusieurs couches du modèle OSI. Cependant, cette solution risque d'épuiser rapidement les ressources disponibles dans le réseau, ce qui n'est pas souhaitable pour les RCSFs. En effet, il serait judicieux de bénéficier des caractéristiques de l'approche Cross-layer *(inter couches)*, afin de résoudre le compromis entre les niveaux de sécurité et le degré de consommation en ressource. Notre point de vu consiste à proposer une nouvelle approche de détection d'intrusions, basée sur l'interaction Cross-layer entre les différentes couches du modèle OSI. Par conséquent, cette approche permet d'ouvrir une nouvelle issue de recherche pour la sécurité dans les réseaux de capteurs sans fil.

## 6. CONCLUSION

Dans ce chapitre, nous avons essayé de mettre le point sur le problème de détection d'intrusions dans les RCSFs. Ce dernier représente un domaine de recherche très récent, dont la plupart des recherches sont encore à leurs stades théoriques. Comparées à celles des réseaux classiques, les techniques de détection d'intrusions dédiées aux RCSFs sont confrontées à des contraintes supplémentaires. De plus, les performances de ces techniques sont non seulement évaluées selon leur niveau de sécurité, mais aussi selon leur degré de consommation en ressources. En effet, les systèmes de détection d'intrusions mono couche, ont démontré leur inefficacité en termes de sécurité et de consommation des ressources. Par conséquent, nous sommes contraints à explorer de nouveaux horizons de recherche. Nous avons constaté qu'on peut profiter des caractéristiques d'architectures Cross-layer afin de développer des systèmes de détection adaptés aux particularités des RCSFs. Cette proposition permet la création d'une nouvelle approche de détection d'intrusions basée sur l'interaction Cross-layer entre plusieurs couches du modèle OSI. Dans le chapitre suivant, nous allons aborder un autre aspect de sécurité dans les RCSFs, dans lequel on doit sécuriser le réseau contre les défaillances de ressources qui peuvent surgir. Ce nouvel aspect de sécurité est intitulé sécurité de ressources dont l'économie d'énergie représente l'axe de recherche le plus important.

# Chapitre 4

## *L'économie d'énergie dans les RCSFs*

# 1. INTRODUCTION

Dans les chapitres précédents, nous avons présenté le concept de sécurité dans les RCSFs dans lequel on doit sécuriser le réseau contre des nœuds attaquants qui peuvent être internes ou externes par rapport à ce dernier. Cependant, le problème de sécurité peut aussi être formulé d'une autre manière, si on prend en considération la limitation des ressources des nœuds capteurs. En conséquence, on peut considérer la mauvaise gestion des ressources comme une menace de sécurité qui peut entraver le bon fonctionnement du réseau. Ainsi, le concept de sécurité dans les RCSFs consiste à sécuriser le réseau contre les attaques externes et internes et contre le réseau lui-même *(épuisement des ressources)*. De notre point de vue, en plus d'offrir des mécanismes de sécurité, les solutions de sécurité doivent aussi proposer des protocoles de communication avec une gestion efficace des ressources. Nous désignons cette nouvelle forme de sécurité par sécurité de ressources, qui consiste à gérer efficacement les ressources disponibles dans le réseau.

Etant donné que l'énergie est la ressource la plus précieuse dans les RCSFs, les protocoles de communication dédiés pour l'économie d'énergie représentent l'axe de recherche le plus important dans le domaine de sécurité de ressources. En effet, les nœuds capteurs sont alimentés par de petites batteries à faible capacité, et le changement ou le rechargement de ces dernières est une tâche difficile *(à cause du nombre très grand de nœuds capteurs)* voire même impossible *(due au déploiement dans les zones hostiles)*. Ainsi, l'épuisement d'énergie des nœuds capteurs implique le plus souvent la mise hors service du réseau tout entier. Par conséquent, les protocoles de communication dédiés aux RCSFs doivent prendre en compte le degré de consommation d'énergie afin d'offrir une gestion efficace de cette ressource très précieuse. Dans ce chapitre, nous allons aborder le problème d'économie d'énergie en mettant le point sur les formes de consommation et les sources de gaspillage d'énergie. De plus, un état de l'art sur les principaux protocoles de communication à base d'économie d'énergie sera présenté.

# 2. LA CONSOMMATION D'ENERGIE DANS UN NŒUD CAPTEUR

Les nœuds capteurs sont généralement destinés à surveiller leur environnement, collecter des informations sur ce dernier, et traiter et transmettre ces informations à un point central dans le réseau *(BS)*. Ainsi, on peut distinguer différentes sources de consommation d'énergie à savoir : la collecte, le traitement et la transmission des données.

**2.1 Énergie consommée durant la collecte des données:** l'acquisition des données se fait généralement à travers un ou plusieurs capteurs. Par conséquent, l'énergie consommée durant cette phase peut être générée par l'échantillonnage et la conversion des signaux physiques en signaux électriques, le conditionnement des signaux et la conversion analogique-numérique. Étant donné la diversité des capteurs, il n'y a pas de valeurs typiques de l'énergie consommée. Cependant, le niveau de consommation est en général très faible comparé aux autres sources de consommation d'énergie.

**2.2 Énergie consommée durant le traitement des données:** l'énergie du traitement est l'énergie consommée par le processeur afin d'effectuer les calculs nécessaires sur les données

collectées. Dans le but de réduire leur consommation d'énergie, les processeurs des nœuds capteurs doivent passer de l'état actif à l'état d'écoute *(idle)* ou de sommeil, dans le cas où il n'y a pas de données à traiter. Bien évidemment, la consommation d'énergie dans ces deux états est très réduite par rapport à l'état actif. Le degré de consommation d'énergie varie d'un processeur à l'autre ; par exemple le processeur MSP430 *(embarqué dans les capteurs tmoteSky)* consomme 3 mW en mode actif, 98 µW en mode idle et seulement 15 µW en mode sommeil.

**2.3 Énergie consommée durant la transmission des données:** la transmission des données constitue la source de consommation d'énergie la plus importante, étant donné que celle-ci est basée sur l'utilisation des antennes radio. Comme le processeur, l'antenne radio possède plusieurs modes de fonctionnement afin de gérer efficacement sa consommation énergétique. Ainsi, selon le niveau de dissipation d'énergie, l'antenne radio peut être en mode sommeil, idle, réception ou transmission. En outre, le passage d'un mode à l'autre engendre une dissipation d'énergie, puisque cela sollicitera une activité importante des circuits électroniques.

## 3. LES SOURCES DE GASPILLAGE D'ENERGIE

La surconsommation d'énergie est toute énergie consommée au-delà du seuil normal. Celle-ci peut être causée par différents phénomènes qui engendrent le gaspillage d'énergie des nœuds capteurs.

**3.1 L'écoute passive *(idle)*:** les nœuds capteurs sortent périodiquement de leur état de sommeil afin d'écouter le trafic dans le réseau. Cette écoute permet de savoir s'il y'a des données à recevoir ou à relier dans le réseau. Cependant, cette mise en écoute peut être très coûteuse dans le cas de réseau à faible trafic. En effet, le nœud capteur peut écouter le trafic sans pour autant recevoir ou transmettre des données, ce qui va gaspiller inutilement ses réserves d'énergie. De plus, la transition périodique entre le sommeil et l'écoute engendrera une autre source de surconsommation d'énergie, surtout dans le cas où celle-ci n'est pas contrôlée.

**3.2 Les collisions:** à cause de leur environnement de communication sans fil, les réseaux de capteurs sont fortement exposés aux interférences et aux collisions. Ces dernières sont générées lorsque deux ou plusieurs nœuds adjacents transmettent leurs données en même temps. En effet, les collisions sont considérées comme étant la source de gaspillage d'énergie la plus importante, étant donné que celles-ci provoquent la retransmission des paquets en collisions, ce qui est très coûteux en énergie.

**3.3 La puissance de transmission:** la portée des antennes radio est directement liée à la puissance de transmission utilisée. La plupart des nœuds capteurs possèdent des antennes à portée statique dans lesquels la puissance de transmission est fixée précédemment par les concepteurs. Par conséquent, un nœud capteur peut gaspiller pas mal d'énergie en utilisant une grande puissance de transmission afin de communiquer avec un nœud très proche de lui.

**3.4 Les distances de transmission:** afin d'économiser la consommation d'énergie, il est préférable de réduire les distances de transmission entre les nœuds capteurs. Ainsi, la communication multi sauts est souvent sollicitée dans les réseaux de capteurs sans fil,

contrairement à celle basée sur un seul saut dans laquelle la dissipation d'énergie est très élevée.

**3.5 L'écoute abusive *(Overhearing)*:** le nœud capteur peut recevoir toutes les données échangées entre ses nœuds voisins, même si ces dernières ne lui sont pas destinées. L'intensité de cette écoute abusive est proportionnelle à la densité du réseau. Ainsi, cela peut engendrer un grand gaspillage d'énergie, vu que la majorité des RCSFs sont déployés à grande échelle.

**3.6 Le surcoût des paquets de contrôle *(Overhead)*:** l'échange des paquets de contrôle peut être une autre source de gaspillage d'énergie, principalement si le nombre de paquets de contrôle est inutilement élevé.

## 4. LES PROTOCOLES D'ECONOMIE D'ENERGIE

L'économie d'énergie est autant considérée comme un paramètre fondamental dans le contexte de robustesse et de sécurité des réseaux de capteurs sans fil. Malgré les progrès qui ont été réalisés, le prolongement de la durée de vie des nœuds capteurs continue d'être un défi majeur et un facteur clé, exigeant davantage de recherches sur l'efficacité énergétique des plates-formes et des protocoles de communication. Dans cette section, nous présentons un état de l'art sur les recherches proposées dans le domaine d'économie d'énergie dans les RCSFs. En effet, la plus grande partie ces recherches est réservée aux protocoles mono couche. Qui traitent le problème d'économie d'énergie au niveau d'une seule couche du modèle OSI.

### 4.1 Protocoles dédiés à la couche réseau :

Les protocoles de routage destinés pour la gestion efficace d'énergie, représentent la majorité des recherches proposées au niveau de la couche réseau. L'idée de base de cette catégorie de protocoles est la construction des chemins de routage à faible consommation énergétique *(energy aware routing path).* De nombreuses stratégies de routage ont été créées pour les réseaux de capteurs sans fil. Certaines sont des adaptations de stratégies qui existaient pour d'autres types de réseaux *(principalement pour les réseaux sans fil au sens le plus large)*, tandis que d'autres ont été conçues spécialement pour les réseaux de capteurs sans fil. Les algorithmes de routage sont en fait découpés en trois familles [111] : les algorithmes de routage centrés données, hiérarchiques ou géographiques**.**

***4.1.1 Les protocoles de routage centrés données (Data-centric protocols)***: le routage centré sur les données est le modèle le plus simple où chaque nœud dans le réseau transmet ses données à la station de base. Chaque nœud joue typiquement le même rôle et les nœuds capteurs collaborent pour accomplir la tâche de captage. La station de base envoie des requêtes à certaines régions et se met en attente des données des capteurs situés dans les régions choisies. On peut citer comme exemples : les protocoles de propagation *(flooding)* et discussion *(gossiping)* proposés dans [112], le protocole de routage par négociation SPIN [113] et le protocole de routage par diffusion dirigée [114].

***4.1.2 Les protocoles de routage basés sur la localisation (géographique) :*** dans ce type de routage [111], les nœuds capteurs sont adressés en fonction de leurs localisations. La distance

entre les nœuds voisins peut être estimée sur la base des forces entrantes du signal. Des coordonnées relatives des nœuds voisins peuvent être obtenues en échangeant une telle information entre les voisins. Alternativement, la location des nœuds peut être disponible directement en communiquant avec un satellite en utilisant GPS *(système de positionnement global)*. Dans la plupart des protocoles de routage, l'information sur la localisation des nœuds est nécessaire afin de calculer la distance entre deux nœuds particuliers de sorte que la consommation d'énergie puisse être estimée. En effet, le routage géographique suppose que tous les nœuds connaissent leur position. Néanmoins, une solution basée sur le GPS peut être trop coûteuse, d'autant plus que le nombre de nœuds à équiper est très grand. Parmi ces protocoles géographiques, on peut mentionner les protocoles MECN [115], SMECN [115] , GAF [116] et GEAR [111] .

***4.1.3 Les protocoles hiérarchiques :*** le routage hiérarchique [111, 115, 117] est considéré comme étant l'approche la plus favorable en termes d'efficacité énergétique. Il se base sur le concept « nœud standard – nœud maître » où les nœuds standards acheminent leurs messages à leur maître, lequel les achemine ensuite dans le réseau tout entier via d'autres nœuds maîtres jusqu'à la station de base *(sink)*.

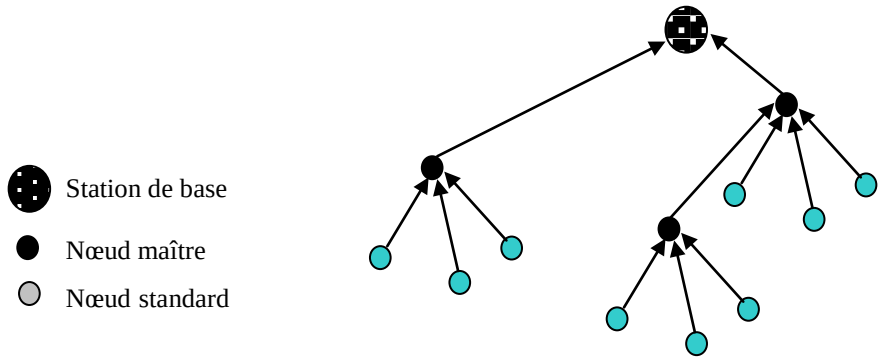



**Figure 4.1** : Le routage hiérarchique

Le point fort de ce type de protocoles est l'agrégation et la fusion des données [11] afin de diminuer le nombre de messages transmis au sink, ce qui implique une meilleure économie d'énergie. En fait, deux grandes approches sont dérivées de ce type de protocoles à savoir : *l'*approche chaînée *(chaîne-based approach)* [118, 119, 120] et l'approche à grappe *(cluster-based approach)* [121, 122, 123, 124].

#### 4.1.3.1 L'approche à grappe (Cluster-based approach)

Elle consiste, de façon similaire aux réseaux téléphoniques cellulaires, à partitionner le réseau en groupes *(clusters)* où dans chacun d'entre eux, un seul capteur est sélectionné comme leader *(Cluster Head)* pour jouer le rôle spécial de point de transfert. D'ailleurs, chaque CH créera un plan de transmission pour tous les capteurs du cluster, ce qui permet aux antennes radio de chaque nœud non-CH d'être éteintes périodiquement, excepté pendant le temps de transmission. Les nœuds dont l'énergie est la plus élevée peuvent être employés dans le traitement et l'envoi d'informations tandis que ceux de basse énergie peuvent être employés dans la collection de données.

L'agrégation des capteurs en grappes permet de réduire la complexité des algorithmes de routage, d'optimiser la ressource médium en la faisant gérer localement par un chef de grappe, de

faciliter l'agrégation des données, de simplifier la gestion du réseau, en particulier l'affectation d'adresses, d'optimiser les dépenses d'énergie, et enfin de rendre le réseau plus évolutif *(scalable)*. L'utilisation des grappes permet aussi aux nœuds d'effectuer des communications avec leurs CHs, sur des petites distances.

La rotation des CHs s'avère également un facteur important pour l'organisation des réseaux de capteurs hiérarchiques. Puisque la BS *(station de base ou sink)* est généralement loin du champ des capteurs, les CHs diffusent une quantité plus importante d'énergie pour la transmission de données à la BS. Par conséquent, les CHs mourront rapidement si le même nœud fonctionne continuellement comme un CH. Ainsi, pour ne pas épuiser la batterie d'un capteur simple, la plupart des algorithmes basés sur le clustering adoptent le concept de la rotation périodique des CHs.

- ***Le protocole LEACH (Low Energy Adaptive Clustering Hierarchy)***

LEACH [125, 126] est un protocole auto-organisateur basé sur le clustering adaptatif, qui utilise la rotation randomisée des têtes de cluster pour distribuer équitablement la charge d'énergie entre les nœuds capteurs dans le réseau. Ce dernier est considéré comme étant l'une des premières approches de routage hiérarchique basées sur le clustering. LEACH est fondé sur deux hypothèses de base dans lesquelles la station de base est fixe et est placée loin des capteurs. De plus, tous les nœuds du réseau sont supposés homogènes et limités en énergie.

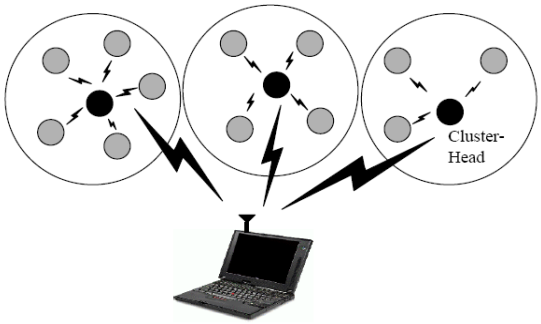



**Figure 4.2**: Algorithme de routage LEACH

L'idée derrière LEACH est de former des clusters de nœuds capteurs selon la force reçue du signal, et d'utiliser les chefs des clusters comme des routeurs pour transférer les données à la station de base. Les dispositifs principaux de LEACH sont :

- *La coordination et le contrôle localisés :* pour l'initialisation et le traitement de grappe.
- *La rotation randomisée du rôle du CH :* effectuée par la station de base ou les têtes de cluster.
- Compression locale *(agrégation) :* les nœuds CH compressent les données arrivant des nœuds appartenant à leurs grappes respectives, et envoient un paquet d'agrégation à la station de base, afin de réduire la quantité d'information qui doit être transmise.

Dans LEACH, le traitement est séparé dans des cycles de longueur constante, où chaque cycle commence par une phase d'initialisation suivie d'une phase de transmission. Dans la première

phase, les clusters sont organisés et les CHs sont sélectionnés. Cette élection est basée sur le pourcentage désiré de CHs et le nombre d'itérations au cours desquelles un nœud a pris le rôle de CH. Ainsi, un nœud n prend une valeur aléatoire entre 0 et 1. Si cette valeur est inférieure au seuil T (n), le nœud se déclare CH.

$$T(n) = \begin{cases} \dfrac{P}{1 - P\left(r \bmod \frac{1}{P}\right)} & Si \quad n \in G \\ 0 & Sinon \end{cases} \qquad (1)$$

Avec :

- $P$ : pourcentage désiré de CHs.
- $r$ : itération actuelle.
- $G$ : ensemble des nœuds qui ont été sélectionnés comme CH durant les dernières (1/ $P$) itérations.

Chaque CH élu émet un message de signalisation au reste des nœuds dans le réseau afin de les informer de son élection comme nouveau CH. Tous les nœuds non CH, et après avoir reçu ce message, décident du cluster auquel ils veulent appartenir. Cette décision est basée sur la force du signal du message reçu. Les nœuds non CH informent les CH appropriés qu'ils seront membres de leurs clusters. Après la réception de tous les messages des nœuds qui voudraient être inclus dans la grappe, et basé sur le nombre de nœuds dans celle-ci, le nœud CH créé un plan de transmission et assigne à chaque nœud un slot de temps durant lequel il peut transmettre. Ce plan est émis à tous les nœuds dans la grappe. L'algorithme LEACH utilise la technique de multiplexage temporel TDMA *(Time division multiplexed access)* comme méthode d'accès au médium. Chaque nœud utilise la totalité de la bande passante allouée par le système de transmission durant son slot de temps. De plus, les nœuds peuvent passer à l'état "endormi" durant les slots inactifs. Ainsi, la perte d'énergie due aux états de sur-écoute *(overhearing)* et d'écoute passive (idle) est évitée.

Dans la deuxième phase, le transfert des données collectées à la station de base a lieu. La durée de la deuxième phase est plus longue que celle de la première phase afin de réduire au minimum les problèmes d'oveheading. Cependant, la collection de données est centralisée et est exécutée périodiquement. Par conséquent, ce protocole s'avère le plus approprié quand on constate un besoin de surveillance de constante par le réseau de capteurs. Après un intervalle de temps donné, une rotation randomisée du rôle du CH est conduite de sorte que la dissipation uniforme d'énergie dans le réseau de capteurs soit obtenue. Les auteurs ont trouvé, en se basant sur leur modèle de simulation, que seulement 5% des nœuds ont besoin d'agir comme leaders.

Etant donné que cet algorithme ne garantit pas une distribution équitable de la dissipation d'énergie, une version centralisée de l'algorithme *(LEACH-C)* [125] est donc proposée. Cette dernière permet de déterminer, à partir de la position exacte des nœuds, la configuration optimale pour minimiser l'énergie dépensée. LEACH-C *(LEACH-CENTRALISE)* est une variante de LEACH où les grappes sont formées d'une manière centralisée par la station de base. LEACH-C utilise la même étape de transmission que LEACH. Durant la phase d'initialisation, la BS reçoit de chaque nœud des informations concernant leur localisation et leur réserve d'énergie. Ensuite, elle exécute un algorithme de formation de grappes centralisé afin de former

les grappes et sélectionner leurs CHs. LEACH-C utilise l'algorithme de la réussite simulée [127] pour obtenir des grappes optimales. Dès que les clusters sont formés, la station de base envoie ces informations à tous les nœuds du réseau. Cependant, la version centralisée de LEACH n'est pas adaptée aux réseaux de grande dimension.

Une autre version a été proposée dans [125], nommée LEACH-F *(LEACH avec des grappes fixes)*. LEACH-F est basé sur le clustering statique qui consiste à former et fixer les clusters une seule fois tout au long de la durée de vie du réseau. L'avantage est que, une fois que les grappes sont formées, aucune autre phase d'initialisation n'aura lieu, ce qui permet de réduire les problèmes d'overheading et d'économiser plus d'énergie. Cependant, les clusters fixes dans LEACH-F ne permettent pas à de nouveaux nœuds d'être ajoutés au système et n'ajustent pas leur comportement basé sur la mort des nœuds, ce qui rend le réseau moins flexible.

Malgré le fait que LEACH permet de réduire la dissipation d'énergie, d'optimiser l'utilisation de largeur de bande et de ne pas gaspiller les réserves d'énergie, un nombre d'inconvénients restent plus au moins apparents. Nous citons parmi eux :

- L'idée de formation de grappes augmente le nombre de messages échangés entre les nœuds, qui peuvent diminuer le gain dans la consommation d'énergie ;
- Les nœuds les plus éloignés du CH meurent rapidement par rapport à ceux plus proches *(communication à un seul saut);*
- Le nombre de messages reçus par les CHs équivaut, approximativement, celui des nœuds gérés. Ceci mène à l'épuisement rapide de leur réserve d'énergie ;
- L'utilisation de single-hop (un saut) au lieu de multi-hop (multi sauts) épuise rapidement l'énergie des nœuds et consomme d'avantage de largeur de bande.
- LEACH suppose que tous les nœuds peuvent transmettre avec une énergie assez suffisante pour atteindre la BS si nécessaire. Ce qui n'est pas toujours le cas.
- Il ne peut pas être appliqué à des applications avec une contrainte de temps du fait qu'il résulte en une longue latence.

Etant donné que LEACH est considéré comme étant le fondateur de l'approche de routage hiérarchique basée sur le clustering, plusieurs recherches ont été proposées afin d'améliorer ce dernier. Parmi ces recherches on peut citer les protocoles suivants : AROS [128], HCR [129], SPEAR [130], Energy-LEACH et MultiHop-LEACH [131], ECSA [132], E-LEACH [133], Improved-LEACH [134], STATIC-LEACH [135], Q-LEACH [136] et LEACH-TLCH [137].

#### 4.1.3.2. L'approche à chaîne (Chain-based approach)

Dans cette approche [138], le principe du clustering est abandonné. Les nœuds du réseau sont organisés de façon à former une grande chaîne de proches voisins, où un seul nœud est sélectionné pour transmettre au sink. En fait, l'idée de formation de chaîne a été proposée pour la première fois dans l'algorithme PEGASIS [118, 119] *(Power-Efficient Gathering in Sensor Information Systems)*.

- ***Le protocole PEGASIS (Power-Efficient Gathering in Sensor Information Systems)***

L'idée de base du protocole PEGASIS est que, dans le but de prolonger la durée de vie du réseau, les nœuds vont être organisés de telle sorte à ce qu'ils forment une chaîne, et n'auront ainsi besoin de communiquer qu'avec seulement leurs voisins les plus proches, et se relaient dans la communication avec la station de base. Ceci réduit l'énergie exigée pour transmettre des données par cycle du moment que la dissipation d'énergie est diffusée uniformément sur tous les nœuds. Par conséquent, PEGASIS a deux principaux objectifs. D'abord, augmenter la durée de vie de chaque nœud en employant les techniques de collaboration, ce qui permet d'augmenter par conséquent la durée de vie du réseau. En second lieu, permettre seulement la coordination locale entre les nœuds voisins de sorte que la largeur de bande consommée dans la communication soit réduite.

À la différence de LEACH, PEGASIS évite la formation des clusters et n'utilise qu'un seul nœud dans une chaîne afin de transmettre à la BS, au lieu d'en utiliser plusieurs. La forme agréée des données sera envoyée à la station de base par n'importe quel nœud dans la chaîne, et les nœuds dans cette dernière vont se relayer pour la transmission à la station de base. Les nœuds vont être organisés de sorte qu'ils forment une chaîne qui peut être soit calculée d'une façon centralisée par la BS et émise à tous les nœuds, ou accomplie par les nœuds capteurs eux-mêmes en employant un algorithme spécial *(greedy algorithme)*. Dans le cas où la chaîne est calculée par les nœuds capteurs, ils doivent d'abord obtenir toutes les données sur l'emplacement des nœuds capteurs, et forment ensuite localement la chaîne en se basant sur les informations de localisation. Puisque tous les nœuds ont les mêmes données d'emplacement et exécutent le même algorithme, ils vont tous produire le même résultat.

Pour construire la chaîne, PEGASIS commence avec le nœud le plus éloigné de la BS. Le voisin le plus proche de ce nœud sera le nœud suivant dans la chaîne. Les voisins successifs sont sélectionnés de cette manière parmi les nœuds non visités afin de former la chaîne de nœuds. La figure 4.3 montre le nœud c0 se reliant au nœud c3, le nœud c3 se reliant au nœud c1, et le nœud c1 se reliant au nœud c2, dans cet ordre. Quand un nœud meure, la chaîne est reconstruite de la même manière pour dévier le nœud mort.

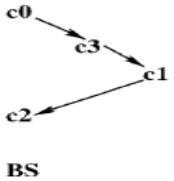



**Figure 4.3 :** Construction de la chaîne de transmission.

Pour collecter les données des nœuds capteurs, le nœud capteur i reçoit les données du nœud voisin (i-1), les fusionne avec les siennes, et transmet à un autre nœud voisin (i+1) dans la chaîne. Les nœuds se relient dans la transmission à la BS, par conséquent, le CH dans chaque cycle de communication sera à une position aléatoire sur la chaîne, ce qui permet aux nœuds de mourir à des localisations aléatoire *(rendre le réseau robuste aux échecs)*. Chaque cycle de collecte des données peut être lancé par la BS, avec un signal de balise qui synchronisera tous les nœuds capteurs. Puisque tous les nœuds connaissent leurs positions sur la chaîne, PEGASIS peut employer une approche de slot de temps (TDMA) pour la transmission des données. Ainsi,

Le nœud $c_0$ transmettra ses données au nœud $c_1$ dans le premier slot, $c_1$ fusionne et transmet ces données dans le deuxième slot, et ainsi de suite jusqu'à ce que le nœud CH soit atteint. Finalement, dans le $n^{ième}$ slot, le leader transmet les données à la BS.

PEGASIS exécute la fusion des données à chaque nœud excepté les nœuds de fin de chaîne. Chaque nœud va fusionner les données de ses voisins avec les siennes, afin de générer un paquet simple de la même longueur et les transmet par la suite à son autre voisin *(S'il en a deux)*. Dans la formation de la chaîne, il est possible que certains nœuds puissent relativement avoir des voisins distants le long de cette dernière. De tels nœuds vont dissiper plus d'énergie dans chaque cycle, comparé à d'autres nœuds capteurs. Pour remédier à cela, les auteurs ont placé un seuil de distance entre les nœuds voisins, qui ne permet pas aux nœuds éloignés de devenir leaders.

Par rapport à LEACH, le protocole PEGASIS offre des améliorations significatives en termes d'économie d'énergie. En premier lieu, dans la collecte des données, les distances que la plupart des nœuds transmettent sont beaucoup plus réduites par rapport à la transmission au CH dans LEACH. En second lieu, la quantité de données à recevoir par le CH est au plus deux messages au lieu de n *(n est le nombre de nœuds dans le cluster)*. Finalement, seul un nœud transmet à la BS dans chaque cycle de communication *(économiser plus d'énergie)*. Bien que le *Clustering* soit évité, PEGASIS *(l'approche à chaîne)* présente un retard excessif pour le nœud le plus éloigné sur la chaîne. PEGASIS suppose que chaque nœud capteur doit pouvoir communiquer directement avec la BS. Dans des cas pratiques, les nœuds capteurs utilisent la communication multi sauts pour atteindre la station de base. En outre, PEGASIS suppose que tous les nœuds maintiennent une base de données complète à propos de la location de tous les autres nœuds dans le réseau. La méthode par laquelle les nœuds sont localisés n'est pas décrite. En outre, PEGASIS suppose que les nœuds capteurs ont tous le même niveau d'énergie et qu'ils sont susceptibles de mourir en même temps. Ceci ne peut pas être appliqué à un réseau de capteurs dans lequel il n'est pas facile d'obtenir une connaissance globale du réseau *(position de tous les nœuds)*.

Une amélioration de PEGASIS, appelée H-PEGASIS *(PEGASIS hiérarchique)* a été présentée dans [111]. H-PEGASIS vise à diminuer le retard encouru pour les paquets pendant la transmission à la station de base, à travers les transmissions simultanées des messages de données. Pour éviter les collisions et les interférences possibles du signal le long des capteurs, deux approches ont été investiguées. La première approche incorpore le codage du signal, par exemple CDMA. Dans la deuxième approche, seuls les nœuds spatialement séparés ont la permission de transmettre en même temps. D'autres recherches ont été proposées afin d'améliorer le protocole PEGASIS. Parmi ces recherches, on peut citer les protocoles suivants : CCS [139] et DS-PEGASIS [140].

### 4.2 Protocoles dédiés à la couche liaison (sous couche MAC)

Les sources de consommation d'énergie dans un nœud capteur sont le module radio, le microprocesseur et le capteur. La communication radio est considérée comme la source la plus consommatrice parmi les trois précédentes. Etant donné que la sous couche MAC est principalement concernée par l'utilisation du module radio, plusieurs recherches ont été proposées afin de gérer efficacement cette ressource et optimiser ainsi la consommation énergétique. En

effet, un protocole MAC économe en énergie essaie d'utiliser le module radio le moins souvent possible. L'utilisation inutile du module provient de 5 sources essentielles : l'overhearing, les collisions, l'Idle, les envois infructueux et les messages de contrôle.

Le moyen le plus efficace pour conserver l'énergie est de mettre la radio du nœud capteur en mode veille *(low-power)* à chaque fois que la communication n'est pas nécessaire. Idéalement, la radio doit être éteinte dès qu'il n'y a plus de données à envoyer et ou à recevoir, et devrait être prête dès qu'un nouveau paquet de données doit être envoyé ou reçu. Ainsi, les nœuds alternent entre périodes actives et sommeil en fonction de l'activité du réseau. Ce comportement est généralement dénommé *Duty-cycling ou wake-up scheme*. Le mécanisme du Duty-cycling permet de réduire le temps qu'un nœud passe dans l'état d'écoute passive (*Idle listening* ), l'overhearing et d'autres activités inutiles en mettant le nœud dans l'état de sommeil.

En effet, de nombreux protocoles MAC introduisent le mécanisme du Duty-cycling dans leurs conceptions afin de réaliser une faible consommation énergétique. Nous classons ces protocoles en deux grandes classes: protocoles synchrones et asynchrones. Les protocoles basés sur le Duty-cycling synchrones sont typiquement équipés avec des périodes d'activation (*wake-up schedules*) prédéterminées. Chaque période est divisée en une phase de sommeil '*Tsleep*' et une phase d'activité '*Tactive*'. La synchronisation est effectuée par la transmission fréquente de petites trames de contrôle. Chaque nœud diffuse des trames d'activation une fois qu'il entre dans sa période d'éveil. Ainsi il réveille tous ses nœuds voisins pour une éventuelle communication.

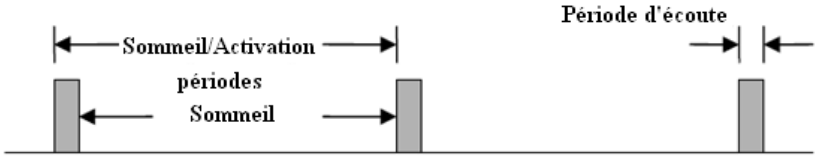



**Figure 4.4:** le mécanisme du Duty-cycling

En ce qui concerne les protocoles basés sur le Duty-cycling asynchrones, on n'est pas obligé de synchroniser les horloges des nœuds capteurs. Dans ce cas, chaque nœud suit un plan d'activation afin de se mettre en état d'activité ou de sommeil. Ainsi, le plan d'activation arrange les périodes d'activation des nœuds capteurs, de telle sorte à garantir que les nœuds émetteurs et récepteurs s'activent au même instant afin de communiquer leurs données.

### *4.2.1 Sensor-MAC (S-MAC):*

Sensor-MAC [22] est conçu pour assurer une méthode d'accès économe en énergie pour les réseaux de capteurs sans fil. Pour ce faire, les nœuds se mettent en mode sommeil pendant une certaine durée et se réveillent pour écouter le médium pendant une autre durée. Les nœuds échangent leur calendrier de périodes d'écoute en le diffusant à leurs voisins à un saut. Ainsi, chaque nœud connait le calendrier de ses voisins et sait quand il faut se réveiller pour communiquer avec un nœud à sa portée. Plusieurs nœuds peuvent avoir le même intervalle de temps comme période d'écoute. Les nœuds accèdent au médium en utilisant le CSMA/CA *(IEEE 802.11)* avec le mécanisme RTS/CTS. En outre, un champ supplémentaire est ajouté à tous les messages *(y compris les messages RTS/CTS et les acquittements)* indiquant la durée de l'échange, ce qui permet aux nœuds non concernés de dormir pendant cette durée.

Pour maintenir une synchronisation des horloges, les nœuds émetteurs envoient des messages de synchronisation SYNC au début de la période d'écoute de leurs voisins. Dans [141], les auteurs améliorent le fonctionnement de S-MAC en minimisant le délai de bout-en bout. Pour ce faire, ils obligent les nœuds, après avoir reçu un RTS ou un CTS qui ne les concerne pas, à se réveiller pour une courte durée après la fin de la transmission pour vérifier s'ils sont la prochaine destination et recevoir la trame si c'est le cas.

La figure 2.5 montre le séquencèrent des périodes d'écoute et de sommeil des nœuds avec le découpage en deux parties de la période d'écoute. Les émetteurs B et C, qui souhaitent communiquer avec le récepteur A, connaissent la période d'écoute de A grâce aux messages SYNC envoyés par A.

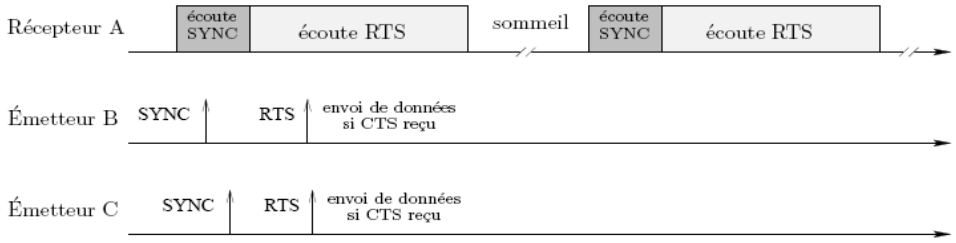



**Figure 4.5:** Séquencèrent des périodes d'écoute et de sommeil dans S-MAC.

S-MAC apporte une amélioration par rapport au CSMA/CA de la norme 802.11 en termes d'économie d'énergie. Cependant, les messages de synchronisation et les messages RTS/CTS génèrent une surcharge du réseau.

Plusieurs recherches basées sur le mécanisme du Duty-cycling ont été proposées afin d'améliorer le protocole S-MAC. Parmi ces recherches on peut citer les protocoles suivants : T-MAC [142], D-MAC [143], B-MAC [144], WiseMAC [145], X-MAC [146], Z-MAC [147, 148], E-MAC [149], L-MAC [150].

### *4.2.2 STEM (Sparse Topology and Energy Management):*

La capacité d'économie d'énergie ainsi que la latence générée par le réveil des nœuds peuvent être améliorées en utilisant un canal de réveil supplémentaire. Les recherches proposées dans [151], [152] et [153] supposent qu'il y a une antenne radio à faible puissance en plus de l'antenne radio principale. La nouvelle antenne radio (*Wake up radio*) est destinée pour le réveil des nœuds capteurs. Elle possède une faible portée radio et consomme beaucoup moins d'énergie comparé aux antennes radio habituelles. Par conséquent, cette radio réveil peut rester activée pendant tout le temps, en consommant peu d'énergie.

Le protocole STEM [154] utilise également deux antennes radio, où l'une fonctionne comme une radio de réveil (*Wake up radio*) et l'autre est utilisée pour la transmission des données. Dans le protocole STEM, chaque nœud active sa radio de réveil pour une durée $T_{dstem}$ à chaque période T, où $T/T_{dstem}$ définie le duty cycle de la radio de réveil. Dans le cas où un nœud veut transmettre, il envoi une trame d'activation au nœud récepteur en utilisant sa radio de réveil. L'émetteur continu à envoyer des trames d'activation jusqu'à ce qu'il reçoit une trame d'acquittement du nœud récepteur. Après que le nœud récepteur soit activé, la transmission des

données peut commencer entre les deux nœuds en utilisant les antennes principales. Afin d'éviter les interférences entre les deux antennes, STEM associe à chacune une fréquence radio différente. La figure suivante présente les différents modes de transition d'antennes radio pour un nœud donné.

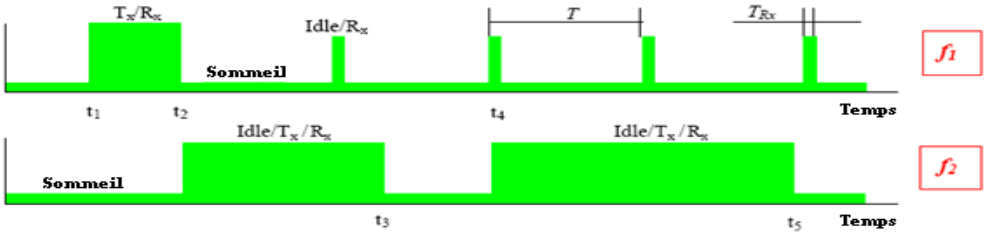



**Figure 4.6 :** Les différents modes de transition d'antennes radio dans le protocole STEM

Au temps t1, le nœud capteur veut activer son nœud voisin afin de lui transmettre ses données. Donc, il commence à envoyer des trames d'activation en utilisant sa radio de réveil jusqu'à ce qu'il reçoit une trame d'acquittement, ce qui surviendra dans le temps t2. Après, les émetteurs et récepteurs activent leurs antennes principales pour la transmission des données. De plus, la radio de réveil continue d'être activée périodiquement pour écouter le trafic sur le réseau. Après la fin de la transmission des données, le nœud met en état de veille son antenne principale au temps t3. Au temps t4, le nœud capteur reçoit une trame d'activation d'un autre nœud et répond par l'activation de son antenne principale.

## 5. CONCLUSION

L'économie d'énergie constitue un défi très important pour la conception d'RCSFs robustes et sécurisés contre les défaillances de ressources. Vu que la disponibilité de ces derniers est directement liée à la durée de vie de leurs batteries, l'application des protocoles de communication à faible consommation d'énergie est plus que nécessaire. Les protocoles mono couche ont prouvé leur efficacité en termes d'économie d'énergie au niveau de leurs couches respectives. Cependant, l'utilisation de l'un de ces protocoles au niveau de chaque couche de la plie protocolaire engendre une mauvaise gestion d'énergie. Les protocoles de communication basés sur l'architecture Cross-layer surpassent ce problème, en faisant interagir plusieurs couches du modèle OSI. Ainsi, au lieu d'utiliser un protocole d'économie d'énergie au niveau de chaque couche, un seul protocole sera appliqué afin de gérer efficacement les réserves d'énergie dans plusieurs couches du modèle OSI. Néanmoins, les protocoles à base d'architecture Cross-layer sont encore dans leur stade primitif, ce qui ouvre une nouvelle issue de recherche dans le domaine d'économie d'énergie. Le chapitre suivant sera consacré à la présentation du concept d'architecture Cross-layer, et son application dans les réseaux de capteurs sans fil.

# Chapitre 5

## *Le concept d'architecture Cross-layer*

# 1. INTRODUCTION

Récemment, le design Cross-layer est apparu comme une approche intéressante pour l'amélioration des performances des réseaux sans fil. Cette approche consiste à concevoir des protocoles à travers l'exploitation des dépendances entre différentes couches, dont le but est d'obtenir un gain en performance. Le design Cross-layer est aussi défini par l'exploitation de l'interaction, et l'exécution d'optimisation conjointe sur plusieurs couches, sous contraintes prédéterminées de ressources. Le design Cross-layer permet aux différentes couches de la pile protocolaire de partager des informations d'état ou de coordonner leurs actions, ce qui rend cette approche particulièrement souhaitable pour les RCSFs. En effet, la limitation des ressources des nœuds capteurs et la surcharge significative générée par l'utilisation des protocoles mono couche nécessitent une telle approche de conception.

Dans ce chapitre, nous allons présenter le concept du design Cross-layer ainsi que les facteurs qui ont motivé l'apparition de ce dernier. Ensuite, nous allons mettre le point sur les différents types d'architectures Cross-layer. Enfin, un état de l'art sur les principaux protocoles Cross-layer proposés pour les RCSFs sera présenté.

# 2. LE DESIGN CROSS-LAYER

L'architecture en couches consiste à diviser les fonctionnalités du réseau en couches, et définir une hiérarchie de services *(protocoles mono couche)* à fournir par ces différentes couches. Ainsi, l'architecture en couches interdit la communication directe entre les couches non adjacentes, tandis que la communication entre les couches adjacentes est limitée aux appels de procédures. Par contre, l'architecture Cross-layer permet de concevoir des protocoles qui ne respectent pas les règles de l'approche en couches. Le concept de base est de permettre le partage de l'information entre les différentes couches du protocole afin d'améliorer la flexibilité et d'accroître les interactions inter-couches. En effet, le design Cross-layer peut être défini par le processus de conception de protocoles par violation de l'architecture en couches [155]. Cette violation implique d'abandonner le luxe de la conception indépendante des protocoles à différentes couches. Ainsi, les protocoles conçus imposent certaines conditions de traitement aux autres couches.

Le concept Cross-layer a été proposé en premier temps pour les réseaux sans fil à base d'architecture TCP/IP [156]. Cela est motivé par la perte de performance générée par l'utilisation de l'architecture TCP/IP, étant donné que celle-ci était originalement proposée pour les réseaux filaires. En effet, il y a une interdépendance étroite entre les couches protocolaires des réseaux sans fil. L'application du design Cross-layer peut aider à exploiter cette interdépendance, et favorise l'adaptabilité de celle-ci sur la base des informations échangées. Cependant, un tel processus de conception doit être soigneusement coordonné, puisque il est difficile de caractériser les interactions entre les différentes couches de protocoles. De plus, l'optimisation conjointe des couches peut conduire à des algorithmes complexes, qui engendreront en conséquence des problèmes de mise en œuvre, de débogage, de mise à jour et de standardisation. En effet, il n'existe pas de standard pour la normalisation de la modélisation protocolaire et la méthodologie d'échanges entre les couches. Cependant, plusieurs plans de gestion ont été proposés afin de faciliter la conception de protocoles Cross-layer.

## 3. MOTIVATIONS DE BASE POUR LE DESIGN CROSS-LAYER

Les caractéristiques particulières de l'environnement de communication sans fil étaient la principale cause qui a poussé les concepteurs de protocoles à violer les normes de l'architecture en couches. En effet, les liens de communication sans fil ont créé plusieurs nouveaux problèmes qui ne peuvent pas être traités par l'architecture classique en couches. Comme exemple de ces problèmes la fausse détection de congestion au niveau du protocole de la couche transport [157]. Cela survient lorsqu'un paquet est perdu lors de sa transmission. Cependant, dans un réseau sans fil les paquets de données peuvent bien ne pas atteindre leur destination à cause des interférences dans le canal de transmission. Ainsi, l'approche Cross-layer peut remédier à ce problème en faisant interagir les deux couches liaison *(sous couche Mac)* et transport.

En plus de son incapacité de résoudre les problèmes engendrés par la communication sans fil, l'approche en couches ne permet pas de s'adapter au changement dynamique du réseau *(ajustement des paramètres de communication par rapport aux variations de la qualité du canal de transmission)*. La mauvaise gestion des ressources représente une autre motivation pour l'utilisation des approches Cross-layer. En effet, l'architecture en couches génère une certaine forme de redondance, qui peut gaspiller les ressources disponibles dans le réseau. De plus, l'allocation indépendante des ressources par les différentes couches du réseau engendre une mauvaise utilisation de ces dernières.

## 4. LES TYPES D'ARCHITECTURES CROSS-LAYER

Selon les recherches proposées dans [156] et [158], les architectures Cross-layer peuvent être classées en trois types de base: architecture à base de communication directe, architecture à base de communication indirecte et architecture à base de nouvelles abstractions.

### 4.1 Architecture Cross-layer à base de communication directe

Le design Cross-layer à base de communication directe consiste à permettre la communication directe *(sans intermédiaire)* entre les protocoles au niveau des couches adjacentes et non adjacentes. Ainsi, les protocoles mono couche proposés pour les architectures en couches doivent être redéfinis, et de nouvelles routines sont à intégrer. Cette redéfinition permet de pouvoir manipuler les données Cross-layer échangées entre les différentes couches. L'orientation des échanges inter-couche, ainsi que le choix des couches impliquées dans cet échange ne sont pas normalisés. Ils dépendent de l'objectif d'optimisation à atteindre. Nous pouvons affirmer que selon ce principe, il y'a autant de variantes d'architectures que de protocoles et de buts à satisfaire.

### 4.2 Architecture Cross-layer à base de communication indirecte

Dans cette architecture, une entité intermédiaire se charge des communications entre les différentes couches protocolaires. Par conséquence, le fonctionnement normal de la pile protocolaire est conservé, ce qui permet de ne pas redéfinir les protocoles existants *(architecture en couches)*. La dénomination et les fonctionnalités de l'entité intermédiaire varient selon l'architecture Cross-layer [159,160]. L'utilisation d'une interface de communication permet de:

• Maintenir l'architecture classique en couches, étant donné qu'il n'y a pas de modification sur ses fonctionnalités de base *(compatibilité complète)*;

• Profiter des avantages de la conception modulaire de l'architecture en couches, ce qui facilite le processus de mise à jour. Ainsi, l'addition et/ou la suppression des protocoles ne va pas engendrer des modifications dans les autres couches;

• Mettre à jour l'entité cross-layer sans avoir à gêner les protocoles relatifs aux couches;

### 4.3 Architecture Cross-layer à base de nouvelles abstractions

La troisième catégorie d'architectures Cross-layer est particulièrement distincte des deux autres car elle présente des abstractions complètement nouvelles [161]. Dans cette approche le concept d'architecture en couches est totalement abandonné, puisque plusieurs couches peuvent être couplées ensembles afin de construire une sorte de super couche, qui peut gérer différentes tâches dans le réseau. Cette nouvelle approche Cross-layer permet d'offrir une forte flexibilité avec un minimum de problèmes.

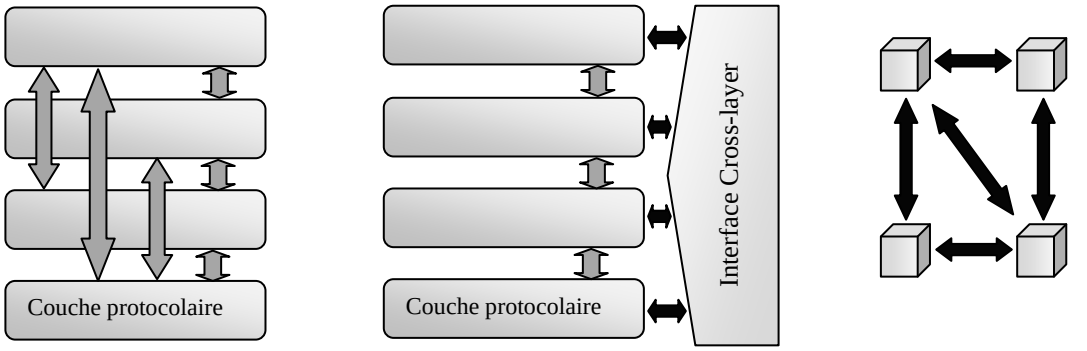



**Figure 5.1:** Classification des architectures cross-layer

## 5. PROTOCOLES CROSS-LAYER DEDIES AUX RCSFs

A cause de leur limitation en ressources, les réseaux de capteurs sans fil constituent l'un des domaines d'application les plus intéressants pour les approches de conception Cross-layer. Dans les dernières années, un ensemble de recherches a été conduit pour l'intégration du design Cross-layer dans les RCSFs. L'objectif de ces recherches est de proposer des protocoles à base d'architectures Cross-layer, dont le but est de gérer les fonctionnalités du réseau tout en

optimisant les ressources disponibles. Néanmoins, ces recherches sont encore dans leur stade primitif et n'ont pas été testées et déployées dans le monde réel. En effet, la plupart des protocoles proposés se concentrent sur les problèmes d'économie d'énergie et de sécurité dans les RCSFs.

### 5.1. Protocoles d'économie d'énergie à base d'architecture Cross Layer

Etant donné que la gestion efficace des ressources est l'une de principales motivations du design Cross-layer, les protocoles Cross-layer dédiés pour l'économie d'énergie constituent la plus grande partie des recherches proposées dans le domaine en question. Ces protocoles peuvent être classés en trois types essentiels, à savoir : Protocoles basés sur une architecture cross layer exploitant l'interaction entre les couches Mac et Physique. Le deuxième type exploite l'interaction entre les couches Réseau et Mac. Enfin, le dernier type se focalise sur l'interaction entre les couches Réseau et Physique.

Les recherches proposées en [162], présentent une nouvelle stratégie de transmission dédiée aux protocoles de routage géographique. Les auteurs ont mis au point une expression qui calcule la distance de transmission optimale, basée sur le principe de répétition automatique des requêtes de transmission *(automatic repeat request, ARQ)*. Ce principe est utilisé au niveau de la couche physique pour établir le taux de réception des nœuds et ainsi les distances de transmission optimale. Au niveau de la couche réseau, l'algorithme de routage utilise le taux de réception de chaque nœud adjacent, afin de déterminer le saut suivant dans le chemin de routage. Une optimisation Cross-layer dédiée aux protocoles de communication multi sauts a été introduite dans [163]. Les auteurs divisent le problème d'optimisation d'énergie en deux sous problèmes, à savoir : Le routage multi sauts à faible consommation énergétique au niveau de la couche réseau, et l'allocation d'énergie de transmission optimale au niveau de la couche physique. Le saut suivant dans le chemin de routage est sélectionné en se basant sur la qualité optimale du lien de transmission. Ce dernier est lié au plus faible niveau d'énergie radio pour qu'un nœud reçoive un paquet de données. Basée sur cette idée, une solution utilisant le principe CDMA *(Code Division Multiple Access)* et OFDM *(Orthogonal Frequency Division Multiplexing)* a été développée pour que le contrôle d'énergie de transmission et le routage des données soient établis d'une manière distribuée.

Deux algorithmes pour l'ajustement d'énergie de transmission ont été proposés dans [164]. Ces derniers sont intitulés *LMA (Local Mean Algorithm)* et *LMN (Local Mean of Neighbours Algorithm)*. Le principal but des ces deux algorithmes est l'établissement d'une table de nœuds voisins au niveau de la couche physique. Celle-ci est ensuite utilisée au niveau de la couche réseau pour la construction des chemins de routage. Dans l'algorithme LMA, chaque nœud transmetteur diffuse périodiquement un message de contrôle contenant son adresse pour connaitre ses nœuds voisins. Après la réception réussie de ce message, chaque nœud récepteur va répondre par l'envoi d'un message d'acquittement contenant l'adresse de ce dernier. Avant de transmettre le message de contrôle, le nœud transmetteur initialise l'énergie de transmission au plus bas niveau. Ainsi, la portée de son antenne radio est réduite au maximum. Dans le cas où aucun message d'acquittement n'est reçu, le nœud transmetteur augmente l'énergie de transmission et envoi à nouveau un autre message de contrôle. L'opération sera répétée jusqu'à ce que le nœud le plus proche soit détecté. Ainsi, une table de nœuds voisins sera établie afin d'être utilisée au niveau de la couche réseau.

L'algorithme LMN, est basé sur le même principe que l'algorithme LMA. Cependant, l'algorithme LMN introduit le nombre de voisins du nœud récepteur dans le choix de ce dernier. En plus de son adresse, le nœud récepteur ajoute le nombre de ses voisins dans le paquet d'acquittement. Après que le nœud transmetteur reçoit des paquets d'acquittement de tous ses voisins, il sélectionne le nœud voisin le plus proche et possédant le plus grand nombre de voisins. Ainsi, une meilleure connectivité de réseau est garantie.

Les recherches proposées en [165], présentent une nouvelle stratégie de construction des chemins de routage. Les auteurs proposent une approche Cross-layer qui exploite le taux d'interférence au niveau de la couche Mac afin de sélectionner les nœuds suivants dans le chemin de routage. Ainsi, le problème d'interférence entre les nœuds voisins sera éliminé, ce qui implique une meilleure gestion d'énergie. L'utilisation du mécanisme de *Duty-cycling* dans une architecture Cross-layer combinant les couches réseau et MAC a été considérée dans [166]. Les informations de routage sont utilisées afin de former des plans d'activation distribués pour chaque nœud dans le réseau. Par conséquent, les nœuds capteurs sont éveillés uniquement lorsqu'ils sont impliqués dans la communication. Comme le trafic est périodique, les horaires d'activation sont ensuite entretenus pour favoriser une efficacité maximale d'énergie. Un autre protocole basé sur la même architecture que ce dernier est également proposé dans [167]. Les auteurs proposent de créer un système d'accès au niveau de la couche MAC basé sur la division temporelle *(TDMA)*, où les nœuds sélectionnent distributivement leurs slots de temps appropriés en se référant aux informations de la topologie locale.

Le protocole Cross-layer PARS *(Power Aware Random Scheduling)* a été proposé dans [168]. Ce dernier est un protocole de communication qui combine les deux couches Mac et Réseau. L'objectif principal de ce protocole est la génération des slots d'accès au média de transmission en se basant sur un nombre pseudo-aléatoire calculé au niveau de la couche de routage. Cela évite la surcharge causée par l'utilisation de paquets RTS et CTS. Une extension du protocole PARS a été proposée par les mêmes auteurs nommée EPAR, qui conserve l'énergie en évitant les collisions et retransmissions. Cependant, EPAR exige la synchronisation des horloges entre les nœuds voisins.

Dans [169], les chercheurs ont proposé un protocole Cross-layer appelé MAC-CROSS. Ce dernier consiste à exploiter les informations de routage au niveau de la couche Mac. MAC-CROSS est basé sur le protocole S-MAC qui introduit une période de mise en veille dans le temps d'accès au media, afin de réduire la consommation d'énergie. Ainsi, chaque nœud entre périodiquement dans un état de veille s'il n'a aucune donnée à transmettre ou à recevoir. Dans le cas où le support de transmission est occupé, les nœuds vont mettre à jour leur NAV (*network allocation vecteur)* afin de prolonger leur période de sommeil. Cependant, les nœuds capteurs se réveillent périodiquement pour écouter le trafic sur le réseau, même si ces derniers ne sont pas inclus dans le chemin de routage, ce qui gaspille inutilement leur réserve d'énergie. Le protocole MAC-CROSS surmonte ce problème en utilisant la table de routage de la couche réseau. Chaque nœud n'appartenant pas au chemin de routage est mis dans un état de veille prolongée. MAC-CROSS classe les nœuds capteurs en trois types en fonction de l'état défini par la transmission de données:

- Partie de communication *(Communicating Parties (CP))* : Représente tous les nœuds qui sont inclus dans la communication actuelle *(comme par exemple les nœuds A et B dans la figure 5.3)*.
- Partie prochaine de communication (*Upcoming communicating Parties (UP))* : Représente tous les nœuds qui vont être inclus dans la communication *(le nœud C dans la figure 5.3)*.
- Troisième partie *(Third Parties (TP))* : Nœuds qui ne sont pas concernés par la communication *(les nœuds D et K dans la figure 5.3)*.

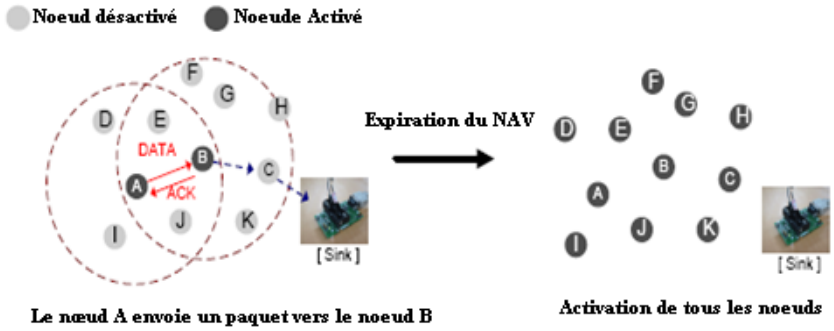



**Figure 5.2:** Le cycle d'activation avec le protocole S-MAC

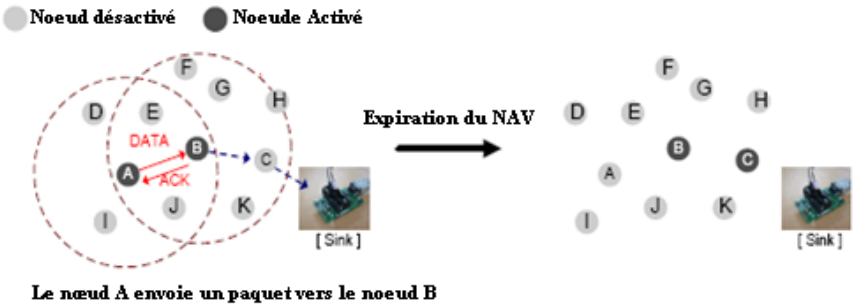



**Figure 5.3:** Le cycle d'activation avec le protocole MAC-CROSS

Le protocole MAC-CROSS modifie les paquets de contrôle RTS et CTS afin d'indiquer aux nœuds capteurs leur changement d'état. La figure suivante compare le format original des paquets RTS et CTS dans le protocole SMAC et les modifications apportées au niveau du protocole MAC-CROSS.

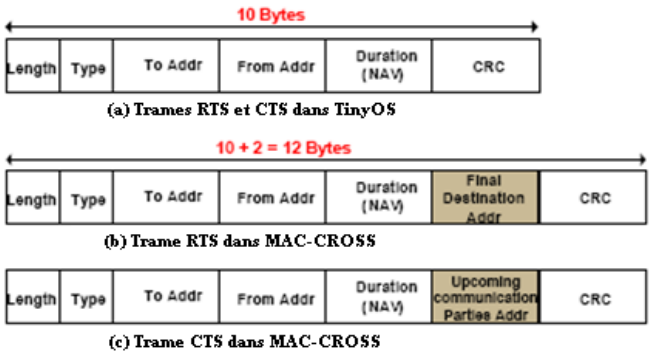



**Figure 5.4:** Les modifications effectuées par MAC-CROSS sur les paquets RTS et CTS.

La première modification effectuée par le protocole MAC-CROSS consiste à ajouter un seul champ au paquet RTS. Ce dernier est nommé '*Final_Destination_Addr*' qui représente la destination finale du paquet de données par lequel le nœud récepteur peut identifier le prochain saut dans le chemin de routage. La deuxième modification consiste à ajouter un autre champ au paquet CTS. Ce nouveau champs est nommé '*UP_Addr*', et informe les nœuds voisins de l'identité du prochain nœud qui va être impliqué dans le routage des données. Revenant à la figure 5.4, lorsque le nœud B reçoit un paquet RTS qui comporte l'adresse de destination finale, son agent de routage se réfère à la table de routage afin d'obtenir l'UP (nœud C). Après, l'agent de la couche MAC du nœud B transmet un paquet CTS comportant les renseignements UP à ses nœuds voisins. A la réception du paquet CTS, le nœud C change son état *(de l'état TP vers l'état UP)* tandis que les autres nœuds prennent conscience du fait que ce sont des nœuds 'TP'. Les nœuds de type 'UP' doivent se réveiller lorsque leur NAV expire pour recevoir les données. Par contre, les nœuds de type 'TP' maintiennent leur état de sommeil même si leur NAV expire, pour économiser l'énergie.

Un autre protocole combinant la couche réseau et Mac a été proposé dans [170]. Ce dernier est nommé CoLaNet *(A Cross-Layer Design of Energy-Efficient Wireless Sensor Networks)*. Les auteurs exploitent l'interaction bidirectionnelle entre les deux couches du modèle OSI. En conséquent, la couche réseau utilise en premier temps les informations locales de la couche Mac pour la construction du chemin de routage. Ensuite, la couche Mac va exploiter la table de routage de la couche réseau afin d'établir le plan de désactivation *(Wake up scheme)* des nœuds capteurs. Le protocole MACRO [171] propose une autre architecture Cross-layer. Dans ce dernier, la décision de routage est effectuée à la suite de compétitions successives au niveau d'accès au support de transmission *(couche Mac)*. Le protocole proposé exige que chaque nœud ne connaisse que ses propres coordonnées et celles du nœud destinataire. De plus, MACRO ne nécessite pas l'échange d'informations de localisation. Afin de sélectionner le nœud relai, un concours est déclenché à chaque saut, de sorte que le nœud de relai les plus économe en énergie soit choisi.

Le protocole XLM [172] procède d'une manière complètement différente comparé au protocole MAC-CROSS. Ce protocole introduit un nouveau concept appelé concept d'initiative. Ce dernier est considéré comme le cœur de son mode de fonctionnement. Dans le cas où un nœud veut transmettre, il annonce à ses voisins qu'il possède un paquet RTS à transmettre. A la réception du paquet RTS, les nœuds voisins décident alors s'ils vont participer à la transmission ou non par la détermination du facteur d'initiative.

Le protocole CLEEP *(Cross-Layer Energy-Efficient Protocol)* [173] introduit un nouveau concept qui consiste à proposer une stratégie Cross-layer basée sur l'interaction des couches Physique, Mac et Réseau. CLEEP obtient en premier temps l'énergie de transmission minimale entre deux nœuds, et maintient par la suite des tables de nœuds voisins au niveau de chaque nœud dans le réseau. Ces tables sont utilisées par la couche réseau afin de construire des chemins de routage économiques en énergie. De plus, le protocole CLEEP utilise les informations de routage de la couche réseau afin de déterminer le plan d'activation des nœuds capteurs au niveau de la couche Mac. Ainsi, la durée de mise en veille des nœuds capteurs peut être considérablement optimisée.

Dans [174], les chercheurs proposent une technique d'optimisation Cross-layer qui représente une extension du traditionnel protocole de routage DSR *(Dynamic Source Routing)*. La nouvelle version improuve la consommation énergétique par la réduction de la fréquence de reconstruction des chemins de routage. Contrairement au protocole DSR, le chemin de routage est reconstruit uniquement s'il y a une erreur dans les liens de transmission. L'architecture Cross-layer proposée combine les deux couches réseau et Mac. Dans les cas où une erreur de communication se déclenche, la couche Mac informe aussitôt la couche réseau que la transmission est interrompue. Ainsi, le chemin de routage est reconstruit afin de contourner le problème de transmission. Cependant, les auteurs utilisent le protocole 802.11 au niveau de la couche Mac. Ce dernier possède plusieurs limites par rapport aux réseaux de capteurs sans fil étant donné qu'il introduit beaucoup d'overhead dans la communication.

Un protocole temps réel asynchrone et basé sur l'économie d'énergie a été proposé dans [175]. Ce dernier est nommé AREA-MAC *(An Asynchronous Real-time Energy-efficient and Adaptive MAC)*. AREA-MAC exploite les deux couches Mac et réseau et interagit en plus avec les couches application et Physique. AREA-MAC est conçu pour fournir une solution adaptée pour les applications temps réel et économes en énergie, dédiées pour les réseaux de capteurs sans fil. De plus, AREA-MAC permet d'offrir un compromis acceptable entre les autres paramètres. Dans AREA-MAC, les auteurs exploitent le mécanisme d'écoute à faible énergie (*low power listening 'LPL'*). Ce mécanisme introduit l'utilisation des messages avec des courts préambules afin de minimiser la latence, la consommation énergétique et l'overhead généré par la communication entre les nœuds capteurs. La technique LPL permet aux nœuds de s'activer pendant une période très courte afin d'écouter le trafic sur le réseau, et d'envoyer leurs données précédées d'une courte trame de contrôle dans le cas où le canal est libre. Ainsi, les nœuds transmetteurs attendent pendant une très court période l'acquittement des données transmises aux nœuds récepteurs. Par contre, si aucun paquet d'acquittement n'est reçu durant cette courte période, les nœuds transmetteurs se mettent en état de veille pendant un temps très réduit avant de s'activer une nouvelle fois. Cela permet de réduire la latence de transmission et d'optimiser la consommation énergétique. Après un nombre maximum de tentatives autorisées pour recevoir un accusé de réception, les nœuds se mettent dans un mode de sommeil prolongé avant de tenter de transmettre une nouvelle fois.

Afin d'améliorer encore la ponctualité d'activation, les nœuds vérifient leurs files d'attente de données après avoir terminé leur réception ou transmission plutôt que d'aller en mode veille directement. Ainsi, s'il existe un ensemble de paquets de données dans leurs files d'attente, les nœuds commencent leur cycle directement à partir de leur période d'activité. Les nœuds utilisant AREA-MAC sont totalement indépendants des horaires de sommeil et de réveil des autres nœuds. Ils ne nécessitent pas un système de synchronisation, qui génère souvent des problèmes d'overhead. Au niveau de la couche réseau, le saut suivant est sélectionné en fonction d'un facteur de progrès pondéré, et la puissance d'émission est augmentée successivement au niveau de la couche physique jusqu'à ce que le nœud le plus efficace soit trouvé. Par ailleurs, un plan d'activation des nœuds capteurs est utilisé au niveau de la couche Mac afin de mettre les capteurs dans un état actif ou en veille, en fonction de la table de routage de la couche réseau. La simulation montre que l'AREA-MAC peut améliorer l'efficacité énergétique, la durée de vie du réseau et conserve un compromis acceptable entre les autres paramètres.

P-MAC (pipeline Mac) [176] est l'un des derniers protocoles proposés pour l'économie d'énergie, et exploitant une architecture Cross-layer basée sur l'interaction des couches réseau et Mac. L'idée de base de ce protocole est de classer les nœuds du réseau en plusieurs niveaux en se basant sur leur distance par rapport à la station de base. Le chemin de routage consiste donc à transmettre les données des nœuds dans un pipeline qui commence par les nœuds les plus éloignés de la BS et se termine par ceux qui sont les plus proches. Au niveau de la couche mac, le réseau sera divisé selon la distance des nœuds par rapport à la BS en plusieurs groupes. Au niveau de chaque groupe, un plan de désactivation différent sera établi où la période de sommeil d'un nœud capteur dépend de la durée d'activité de son nœud voisin dans le groupe de bas niveau. Un plan d'activation à base d'architecture Cross-layer a été proposé dans [177], afin de gérer efficacement la consommation d'énergie dans un RCSF. L'idée de base est d'allouer des slots de temps pour les nœuds capteurs en se basant sur les informations fournies par les deux couches réseau et physique.

## 5.2 Protocoles de sécurité à base d'architecture Cross-Layer

Le niveau de sécurité dans les réseaux de capteurs sans fil est toujours confronté aux contraintes de limitation en ressources. Le design Cross-layer représente une solution intéressante pour remédier au problème de sécurité tout en garantissant un faible taux de consommation en ressources. Cependant, il n'existe pas beaucoup de recherches qui proposent des protocoles de sécurité à base d'architecture Cross-layer pour les RCSFs. La section suivante résume les principaux protocoles et architectures Cross-layer dédiés pour la sécurité dans les RCSFs.

Lazos et Poovendran [178] ont proposé l'une des premières architectures Cross-layer pour la sécurité dans les RCSFs. L'idée de base était de faire collaborer les deux couches physique et réseau afin d'établir des clés secrètes entre les nœuds du réseau. Eschenauer et Gligor [179] proposent un algorithme de gestion de clés Cross-layer qui considère l'interaction entre les trois couches physique, réseau et application. Ainsi, le mécanisme proposé permet l'établissement de clé avec une faible consommation d'énergie, en se basant sur la puissance d'émission des nœuds et l'arbre de routage. Un autre mécanisme basé sur la même architecture Cross-layer précédente [179], a été introduit dans [180] afin d'établir un système de distribution de clés économiques en énergie.

Une nouvelle architecture a été proposée dans [181], faisant interagir les deux couches, physique et application. Les auteurs proposent de combiner les paramètres de sauts de fréquence et de clés secrètes afin de fournir des services de sécurité optimaux pour les réseaux de capteurs. Dans [182], les auteurs proposent un mécanisme de sécurité Cross-layer afin de prévenir les attaques de déni de service dans les RCSFs. Le mécanisme proposé est basé sur un algorithme évolutif qui utilise les propriétés de l'approche Cross-layer afin de s'adapter aux contraintes de ressources dans le réseau. Les auteurs du mécanisme précédent ont aussi proposé un autre algorithme de sécurité basé sur les métas heuristiques [183]. Ce dernier adopte une architecture Cross-layer afin d'intégrer les propriétés physiques du signal dans le processus de sécurité *(couche application)*. Cela permet de préserver les ressources du réseau et d'équilibrer le débit tout en réduisant le temps d'attente des paquets.

Thamilarasu et Sridhar ont abordé la nécessité d'optimisation Cross-layer en matière de sécurité dans les réseaux de capteurs sans fil [184]. Ainsi, les auteurs ont proposé une nouvelle

architecture d'interaction Cross-layer nommée XLSEC, et modélisé différents types d'attaques au niveau des deux couches réseau et application. Les résultats des simulations indiquent les gains apportés par XLSEC en termes de sécurité grâce à ces adaptations Cross-layer. Afin d'offrir un niveau de sécurité optimal, une nouvelle architecture Cross-layer a été proposée dans [185]. Le principe de base de cette architecture est d'offrir plusieurs services de sécurité en exploitant l'interaction entre différentes couches du réseau. L'architecture proposée est classée dans le type d'architectures à base de communication indirecte, où une entité intermédiaire et intelligente nommée ISA est chargée de l'interaction entre les couches protocolaires.

I-Hsun, HSIEH et KUO ont proposé une autre architecture Cross-layer afin de sécuriser le réseau tout en optimisant ses performances [186]. La nouvelle architecture est intitulée CLDNSM, et basée sur l'agrégation de plusieurs paramètres au niveau de différentes couches protocolaires. Le design Cross-layer a été adopté dans [187], afin de concevoir un système de sécurité à base d'indice de confiance. Ce dernier est basé sur l'interaction Cross-layer entre les couches liaison et transport dans l'objectif d'établir des routes sécurisées vers la BS. Un mécanisme de sécurité à base d'architecture Cross-layer a été proposé dans [188], afin de sécuriser le réseau contre différents types d'attaques. L'idée de base est de concevoir un algorithme de sécurité dynamique qui offre le même niveau de sécurité des protocoles mono couche tout en garantissant une faible consommation des ressources.

## 6. CONCLUSION

Le design Cross-layer a prouvé son efficacité par rapport à l'approche classique en couches. L'application de ce nouveau concept dans les RCSFs a engendré plusieurs améliorations, que ce soit au niveau de la gestion efficace des ressources *(économie d'énergie)* ou de sécurité. En effet, l'exploitation de l'interaction entre plusieurs couches protocolaires permet d'éliminer toute forme de redondance et de concevoir des protocoles robustes qui traitent le problème d'économie d'énergie et de sécurité, en prenant en considération différentes couches du modèle OSI. Ce type de protocoles Cross-layer est plus que nécessaire pour les RCSFs, étant donné qu'il permet de remédier à leurs limitations. Néanmoins, l'intégration de l'approche Cross-layer dans la conception des RCSFs reste un domaine très récent. En effet, il n'existe pas suffisamment de recherches dans ce domaine, et toutes les architectures Cross-layer n'ont pas été explorées. De plus, le domaine de sécurité n'a pas été beaucoup sollicité, ce qui ouvre de nouvelles issues pour la recherche scientifique. Le chapitre suivant sera dédié à notre contribution en termes d'économie d'énergie, qui consiste en un protocole de communication Cross-layer à faible consommation énergétique. Ce dernier propose une nouvelle architecture Cross-layer basée sur l'interaction des trois couches réseau, liaison et physique.

# Partie 2

## *Contributions*

# Chapitre 6

## *Le protocole de communication Cross-layer proposé*

# 1. INTRODUCTION

Précédemment, nous avons donné un état de l'art sur les différents protocoles de communication basés sur l'économie d'énergie et dédiés aux réseaux de capteurs sans fil. La recherche dans ce domaine a été très fructueuse ces dernières années avec de nombreuses propositions et améliorations au cours du temps. Ces protocoles peuvent être optimaux dans certaines applications mais pas dans tous les cas. La recherche dans les réseaux de capteurs est ouverte pour de nouvelles idées afin d'optimiser encore les protocoles existants pour obtenir de meilleures performances.

Ce chapitre est dédié à nos contributions de recherche dans le domaine d'économie d'énergie *(sécurité de ressources)*. Il s'agit d'un nouveau protocole de communication nommé CLEOP *(Cross Layer Energy Optimisation Protocol)*, basé sur une architecture Cross-layer et faisant interagir trois couches successives du standard OSI. En effet, notre protocole exploite l'interaction entre les couches Réseau, Liaison (Mac) et Physique dans l'objectif de réduire le plus possible la consommation d'énergie et prolonger la durée de vie du réseau. Ainsi, le protocole CLEOP est composé de trois sous protocoles qui s'auto-complètent : un protocole de routage au niveau de la couche réseau, un protocole d'accès au média au niveau de la couche Mac et un protocole de contrôle d'énergie de transmission au niveau de la couche physique.

Dans ce chapitre, nous allons tout d'abord présenter l'architecture générale du protocole CLEOP. Après, nous allons aborder le protocole de routage proposé au niveau de la couche réseau. Le point suivant qui va être présenté est le protocole d'accès au niveau de la couche Mac. Enfin, nous allons décrire notre protocole de contrôle d'énergie de transmission au niveau de la couche physique.

# 2. LE PROTOCOLE CLEOP

CLEOP *(Cross Layer Energy Optimisation Protocol)*, est un protocole de communication Cross-layer dédié pour les réseaux de capteur sans fil. Ce protocole prend comme objectif principal l'optimisation de la consommation énergétique et le prolongement de la durée de vie du réseau. Contrairement à la plupart des protocoles monocouche, notre protocole traite le problème d'économie d'énergie sur plusieurs couches du modèle OSI. Pour cela, CLEOP propose une architecture Cross-layer basée sur l'interaction et la collaboration des couches réseau, Mac et physique.

Notre démarche consiste à proposer une solution d'optimisation énergétique pour chacune des couches précédentes. Ainsi, nous proposons en premier temps un nouveau protocole de routage hiérarchique au niveau de la couche réseau, ce dernier organise efficacement les nœuds du réseau et établi des chemins de routage économiques en énergie. Au niveau de la couche Mac, notre protocole introduit une nouvelle solution à faible consommation d'énergie pour accéder au média de transmission. Cette solution résout les problèmes majeurs responsables du gaspillage d'énergie dans la couche Mac. En dernier, nous proposons un mécanisme de contrôle d'énergie de

transmission au niveau de la couche physique afin de réduire la perte d'énergie générée par la mauvaise utilisation des antennes radio.

## 2.1 L'architecture Cross-layer du protocole CLEOP

L'architecture Cross-layer proposée dans ce travail conserve la structure traditionnelle en couches et considère l'interaction entre les couches réseau, Mac et physique. Ainsi, cette notion Cross-layer implique 'une cassure' de la notion de 'couches isolées' et un échange d'informations entre toutes les couches, éventuellement non adjacentes.

Tenant compte des challenges de la conception Cross-layer et des problèmes engendrés par une architecture Cross-layer inadaptée, nous avons développé une nouvelle architecture adoptant le principe de communication via une entité intermédiaire nommée CLOA (*Cross-layer optimisation agent*). Ce choix est basé sur la multitude d'avantages offerts par ce type d'architecture à savoir :

- La conservation des avantages de l'architecture en couches, particulièrement en termes de modularité, contribuant ainsi à la compatibilité avec le modèle OSI.
- La présence de l'entité Cross-layer permet une évolution individuelle et continue à la fois pour les couches et pour l'entité elle même sans gêner le système global.
- Un autre avantage est que cette entité dispose d'un libre accès à toutes les couches, ce qui rend ses décisions plus objectives.
- Elle permet également une intégration facile et simple de nouveaux algorithmes et données cross-layer sans avoir à changer le reste de l'architecture.
- Finalement, la présence de cette entité évite toute duplication d'efforts par les différentes couches pour la recherche des informations sur les autres couches non adjacentes ou d'informations sur l'état du canal.

La figure suivante présente l'architecture Cross-layer proposée.

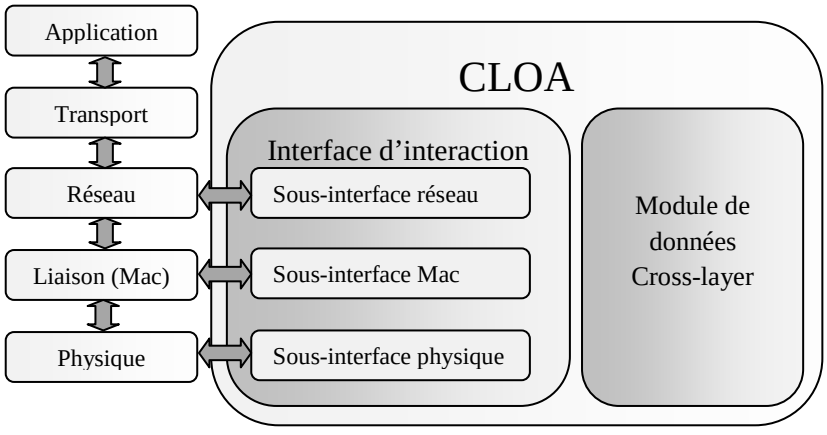



**Figure 6.1 :** L'architecture Cross-layer proposée

L'agent de communication Cross-layer CLOA est l'entité via laquelle les couches et les applications communiquent. Il comporte essentiellement deux parties : l'interface d'interaction et le module de donnée Cross-layer.

***2.1.1 L'interface d'interaction de l'agent CLOA:*** c'est le plan de contact entre les couches et les applications d'une part et l'agent CLOA d'autre part. L'interface d'interaction prend comme objectif principal la gestion des trois sous interfaces qui permettent d'accéder aux couches (IR : Interface réseau, IL : Interface Liaison et IP : Interface physique). Chaque sous interface décrit des méthodes d'écriture et de lecture afin de faciliter la manipulation des paramètres du protocole correspondant. Via ces méthodes se fait la collecte et/ou la mise à jour des données (e.g. l'état de la batterie, l'état du canal, tables de routage, etc.).

***2.1.2 Le module de données Cross-layer de l'agent CLOA:*** le module de données Cross-layer représente les données Cross-layer d'une façon spéciale pour qu'elles soient rapidement accessibles par tous les protocoles des couches. Les données Cross-layer fournies par ce module sont la base de toute adaptation et optimisation Cross-layer. Le module est aussi responsable du maintien à jour de ces données à travers les interfaces d'interaction Cross-layer.

Dans cette nouvelle organisation, les couches Mac et physique sont informées sur les conditions de routage de la couche réseau. De plus, l'interaction entre la couche Mac et physique est aussi fortement exploitée. La figure 6.2 illustre le scénario d'interaction entre les trois couches.

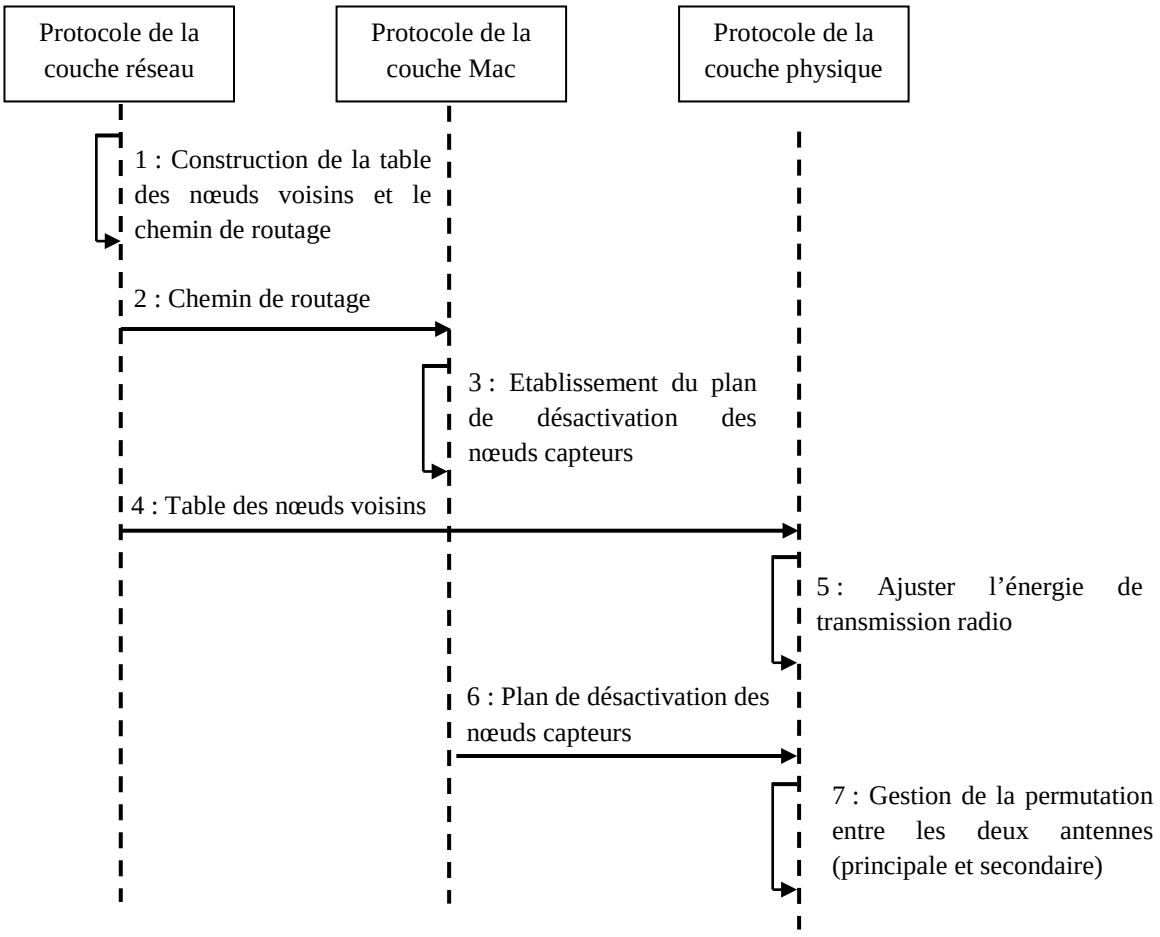



**Figure 6.2 :** Le scenario d'interaction entre les trois couches.

Au niveau de la couche réseau, notre nouveau protocole de routage établi la table des nœuds voisins, organise le réseau sous forme de clusters à chaînes et forme les chemins de routage à faible consommation énergétique.

Dans la couche Mac, notre protocole implémente un plan de désactivation des nœuds capteurs *(Duty-cycling)*, en se basant sur les chemins de routage établis précédemment. Ce plan consiste à mettre dans un état de veille tous les nœuds qui n'appartiennent pas au chemin de routage. Ainsi, le nombre de nœuds activés inutilement est considérablement réduit, ce qui implique une meilleure gestion. CLEOP accorde également une attention aux problèmes de collision, d'écoute passive, d'interférences et d'activation répétitive *(compulsory wake)* qui sont à l'origine de grand gaspillage d'énergie. Pour cela, notre protocole propose d'utiliser une deuxième antenne radio à faible consommation énergétique qui prend en charge la surveillance du trafic sur le réseau et l'activation des nœuds capteurs.

Au niveau de la couche physique CLEOP développe un mécanisme dédié à la gestion et le contrôle efficace d'énergie de transmission radio *(TPC : Transmission Power Control)*. Ce dernier consiste à ajuster l'énergie de transmission radio, en utilisant la table des nœuds voisins établie au niveau de la couche réseau. Ainsi, la portée des antennes radio est ajustée afin d'atteindre uniquement leurs nœuds voisins, ce qui permet de réduire leur consommation d'énergie et prolonger la durée de vie du réseau. De plus, l'interaction avec la couche Mac est exploitée afin de gérer la permutation entre les deux antennes (principale et secondaire).

## 2.2 Protocole de routage au niveau de la couche réseau

Plusieurs protocoles de routage dont l'objectifs commun est l'économie d'énergie ont été proposés pour des les RCSFs. L'étude et l'analyse de ces principaux protocoles et approches de routage nous ont permis de proposer notre propre protocole de routage, dont l'objectif principal est le prolongement du temps de vie du réseau ainsi que la gestion efficace de la consommation énergétique. Ce dernier et nommé HEEP (*Hybrid Energy Efficiency Protocol*) [189, 190]. En effet, l'étude des deux algorithmes LEACH [125] et PEGASIS [119] *(présentés précédemment dans le chapitre 4)*, nous a permis de constater qu'on peut améliorer le premier protocole (LEACH) en appliquant le concept du deuxième protocole (PEGASIS) au niveau de la grappe (cluster), ce qui nous amène à proposer un nouveau protocole hybride combinant les avantages des deux grandes approches, à savoir approche à base de clusters *(Cluster-based approach)* et approche à base de chaînes *(Chain-based approach)*. Le concept de base ainsi que l'architecture de fonctionnement de ce dernier vont être présentés dans ce qui suit.

### *2.2.1 Concept de base du protocole HEEP*

Semblablement au protocole LEACH, notre solution est basée sur l'agrégation des nœuds capteurs sous forme de clusters. Par ailleurs, HEEP applique le principe du protocole PEGASIS à l'intérieur des clusters. Ainsi, dans chaque cluster une chaîne de nœuds adjacents est conçue pour améliorer et réguler la dissipation d'énergie et réduire la charge sur le CH *(cluster-head)*. Contrairement au protocole LEACH, les nœuds communiquent uniquement avec leurs proches voisins et non pas directement avec leur CH, ce qui économise d'avantage d'énergie et offre une meilleure utilisation de la largeur de bande. L'agrégation des données au niveau de chaque nœud

dans la chaîne réduit la quantité de données échangées entre les nœuds et leur CH, ce qui a pour effet de préserver les réserves d'énergie de ces derniers.

La figure 6.3 montre comment les nœuds seront organisés à l'intérieur des clusters, le nœud $N_0$ transmet ses données à son proche voisin $N_1$, $N_1$ quant à lui agrège les données reçues avec les siennes et les transmet à son autre voisin jusqu'à atteindre le CH qui les transmet directement à la BS, ou en utilisant l'approche multi sauts pour préserver d'avantage d'énergie. Donc, dans cette nouvelle organisation (clusters à chaînes), tous les nœuds du cluster vont transmettre leurs données colletées à leurs CHs respectifs en se reliant à travers la chaîne, tandis que chaque CH doit recevoir les données collectées des nœuds entêtes de la chaîne.

A l'inverse de LEACH, le nombre de nœuds qui communiquent avec le CH est considérablement réduit. Ceci implique une meilleure économie d'énergie et prolonge d'avantage le temps de vie des CHs, car si ces derniers meurent (épuisent leur réserve d'énergie), tous les nœuds du cluster vont perdre leur pouvoir de communication avec la BS, et par conséquent le cluster tout entier est considéré comme invalide *(ne communique pas avec la BS)*.

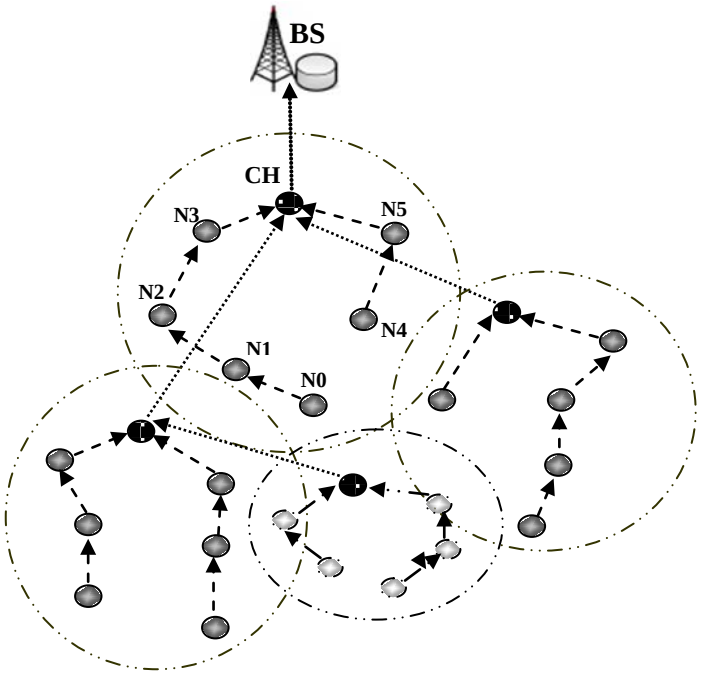



**Figure 6.3 :** Organisation des nœuds dans le réseau.

Dans notre protocole, nous avons adopté le concept de la rotation aléatoire du rôle de CH proposé par LEACH, qui régule la dissipation d'énergie et évite que les nœuds choisis comme CHs meurent plus rapidement. Cependant, par opposition à LEACH, nous avons réutilisé le concept de PEGASIS en organisant les nœuds du cluster sous forme de chaîne, ce qui a pour effet de prévenir que les nœuds les plus éloignés des CHs épuisent leur réserve d'énergie

La nouvelle organisation *(Clusters à chaines)* réduit les distances de transmission ce qui implique une meilleure économie d'énergie et prolonge la durée des nœuds capteurs. En ce qui concerne la latence introduite par la longue chaîne de nœuds dans le protocole PEGASIS, elle est considérablement réduite puisque les chaînes formées dans les clusters sont de petite taille et opèrent simultanément.

### *2.2.2 Les grandes étapes de notre algorithme*

Le déroulement de notre protocole hybride est divisé en plusieurs cycles d'exécution. Chaque cycle commence par une phase d'initialisation dans laquelle les clusters à chaînes sont formés et les CHs sont élus, suivie d'une phase de transmission où les données collectées sont transmises à travers les chaînes aux CHs qui vont à leur tour transmettre à la station de base. Les nœuds doivent être synchronisés de façon à participer à la phase d'initialisation en même temps.

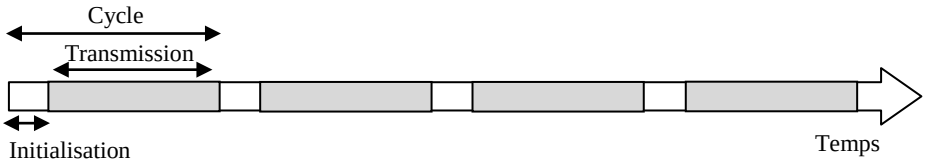



**Figure 6.4** : Etapes d'exécution de notre protocole.

A chaque phase de transmission, tous les nœuds appartenant au même cluster sont délégués pour la tâche de la collecte des données *(tâche à faible consommation énergétique)*, tandis que la tâche la plus coûteuse en énergie *(la tâche de transmission des données vers la BS)* est assignée au nœud possédant la plus grande réserve d'énergie, ce dernier est le CH. Cela signifie que HEEP délègue la tâche la plus coûteuse en énergie à un seul nœud dans le cluster à chaque phase de transmission et assigne la tâche de collecte aux nœuds restants, même s'il y a plusieurs nœuds puissants dans le cluster.

Afin de minimiser les problèmes d'interférence et d'overhead, la durée de la phase d'initialisation est fixée de façon à être beaucoup plus petite par rapport à la phase de transmission. Le coût énergétique de la phase d'initialisation est très faible comparé à la phase de transmission, et dépend du nombre de nœuds dans le réseau et la distance de la BS par rapport au réseau, nous pouvons l'estimer par la règle suivante:

$$\sum_{i=1}^{i=N} q_i \, (E_{elect} + E_{fs} d^2_{toBS}) \qquad (1)$$

Où $q_i$ est la taille d'un paquet de contrôle transmis par un nœud i, $E_{elect}$ est l'énergie consommée par les circuits électroniques *(énergie de calcul)*, $E_{fs}$ est l'énergie perdue dans l'espace de transmission, $d_{toBS}$ est la distance géographique entre un nœud i et la station de base, et N est le nombre de nœuds vivants dans le réseau. Le totale d'énergie initiale dans le réseau est défini par l'équation donnée:

$$E_{totale} = NE_o \qquad (2)$$

Où $E_0$ est l'énergie initiale de chaque nœud dans le réseau et N est le nombre de nœuds du réseau. La durée de vie du réseau est divisée en plusieurs phases de communication *(cycle)*. Nous estimons le nombre de ces dernières par l'équation suivante :

$$Nb_{Ph} = E_{totale} / E_{Ph} \qquad (3)$$

Où $E_{Ph}$ est le total d'énergie consommée durant une phase de communication. Celle-ci est calculée suivant l'équation (4) :

$$E_{Ph} = t\,[2NE_{elect} + NE_{ag} + NE_{fs}d^2_{ns} + N_{CH}E_{mp}d^4_{Bs}] \quad (4)$$

Où $E_{ag}$ est l'énergie consommée durant l'agrégation des données, $d_{Bs}$ est la distance moyenne entre les CHs et la BS, $d_{ns}$ est la distance moyenne entre les nœuds voisins dans la chaine, t est la taille des paquets de données à transmettre, $N_{CH}$ est le nombre de CH dans le réseau $E_{fs}$ et $E_{mp}$ représentent l'énergie perdue dans l'espace de transmission.

Étant donné que les distances de transmission sont réduites, le nombre de phases de communication est impérativement optimisé. Le total des distances de transmission peut être calculé par la formule suivante :

$$d_{Total} = \iint \sqrt{(Xi-Xj)^2 + (Yi-Yj)^2}\, dxdy \quad (5)$$

Où Xj et Yj sont les coordonnées du prochain nœud dans la chaine de transmission.

2.2.2.1 Etape d'initialisation :

L'étape d'initialisation commence par la création des clusters, dans laquelle on adopte la même approche centralisée utilisée dans LEACH-C, et où la station de base utilise la réussite simulée pour former les grappes. Cette approche offre un résultat meilleur, par rapport à l'approche distribuée, utilisée dans LEACH, en termes de formation de grappes et de préservation d'énergie. Après la formation des grappes, les CHs sont choisis d'une manière simplifiée où seul le nœud qui a la plus grande réserve d'énergie, parmi les nœuds de la même grappe, est élu. Ensuite, on aborde la construction des chaînes où l'on suit une méthode centralisée dans laquelle la station de base utilise les informations envoyées par les nœuds pour former les chaînes.

Déléguer la tâche de transmettre des données vers la BS à un seul nœud dans le cluster peut poser des problèmes de perte de données et affecte la fiabilité des données. Le noeud CH peut échouer *(tomber en panne)*, ce qui signifie la défaillance du cluster tout entier. Le protocole HEEP résout ce problème en reléguant la charge de transmission automatiquement au premier nœud puissant *(le plus proche de la station de base)* dans la chaîne de transmission, au cas où le CH cesserait de fonctionner. Le problème de perte de données peut également se produire quand un nœud appartenant à la chaîne de transmission échoue. Pour remédier à ce problème, chaque nœud peut envoyer ses données capturées directement au nœud qui succède le nœud défaillant dans la chaîne de transmission.

2.2.2.2 Etape de transmission :

L'étape de transmission est divisée en plusieurs itérations dans lesquelles les nœuds vont transmettre leurs données colletées à travers la chaîne aux CHs. Dans chaque itération, un nœud transmet au moins un paquet de données. Afin de réduire l'énergie consommée durant la transmission des données, chaque nœud va régler la puissance de son antenne radio de façon à

pouvoir transmettre uniquement à ses proches voisins, à l'inverse du protocole LEACH où les nœuds les plus loin des CHs perdent beaucoup d'énergie afin de pouvoir transmettre leurs données.

Chaque nœud transmet, durant sa période de transmission, les données collectées à son proche voisin dans la chaîne. Comme il est montré dans la figure 6.5, le dernier nœud de la chaîne $n_0$ va transmettre ses données au nœud $n_1$. Ce dernier agrège les données de $n_0$ avec les siennes et transmet à son autre voisin et ainsi de suite jusqu'à ce que l'on atteigne le CH. Chaque cycle de transmission de données est lancé par la BS avec un signal de balise qui synchronisera tous les nœuds capteurs.

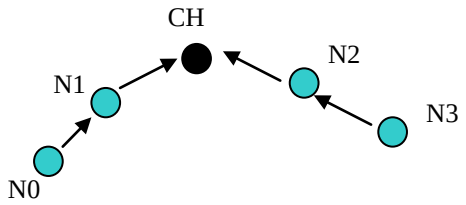



**Figure. 6.5 :** Approche de contrôle de transmission

L'agrégation des données qui permet d'éliminer les informations dupliquées (compression des données) est effectuée au niveau de chaque nœud dans la chaîne contrairement au protocole LEACH où seul le CH est chargé d'agréger les données qui vont être transmises à la station de base. De cette manière, notre protocole hybride réduit la quantité de données envoyées au CH, ce qui implique évidemment le prolongement de la durée de vie de ces derniers.

### *2.2.3 Approche de formation de clusters à chaînes*

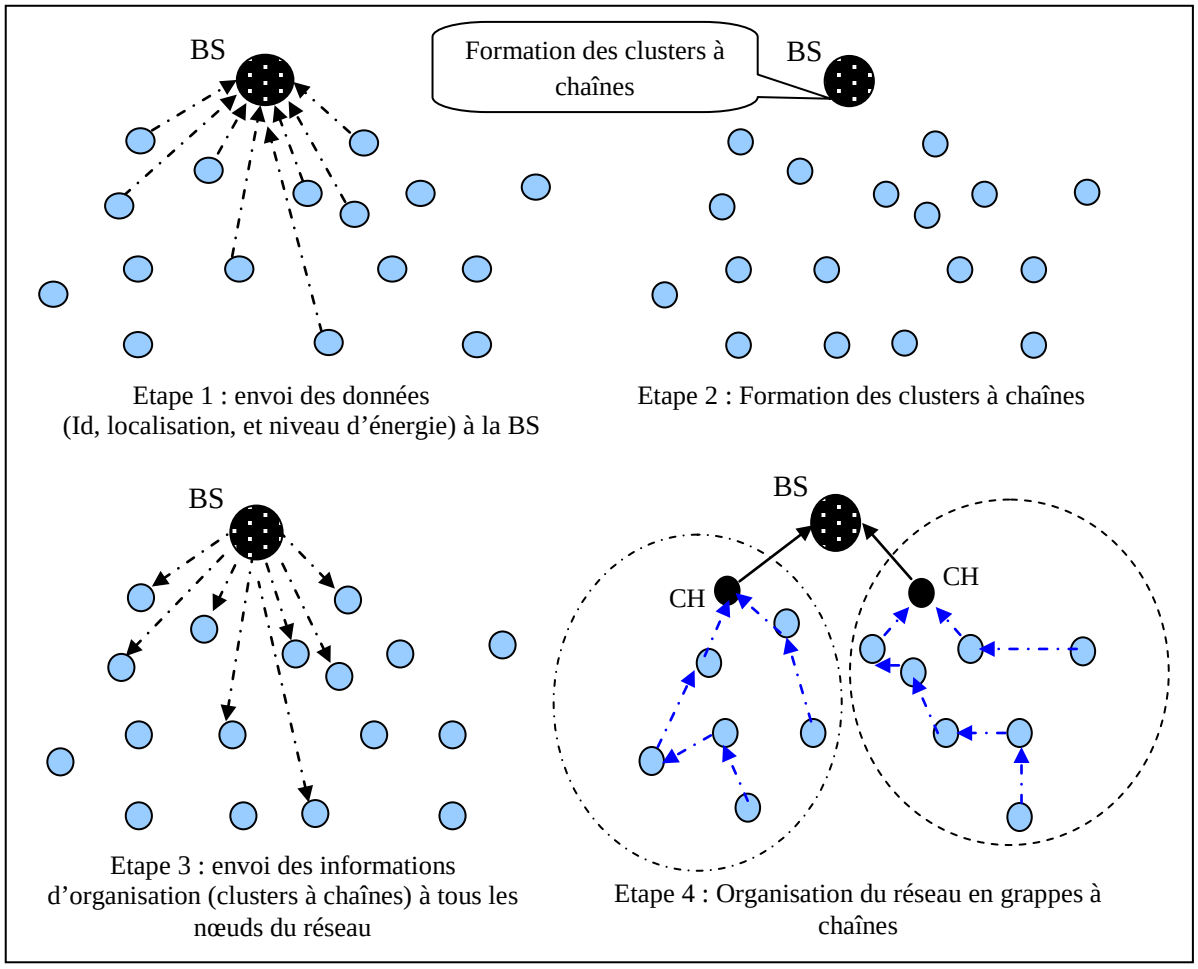
La formation des clusters à chaînes peut être effectuée d'une manière centralisée par la station de base ou d'une manière distribuée par les CHs. Pour obtenir des résultats meilleurs en termes de répartition égale des nœuds entre les clusters, on a choisi l'approche centralisée proposée dans le protocole LEACH-C [125], où chaque nœud envoi un paquet de données à la BS contenant l'identificateur du nœud, la réserve d'énergie et la localisation sur le réseau *(en utilisant par exemple le système de localisation GPS)*. La station de base va exécuter un algorithme d'optimisation afin de former les clusters. L'algorithme de détermination du cluster optimal est algorithme d'optimisation NP-complexe *(NP-HARD)*, comme par exemple l'algorithme de recherche TABOO, qui ne nous donne pas des résultats exactes mais proches de l'optimal. Dans notre algorithme hybride, nous avons opté pour le même algorithme d'optimisation utilisé dans LEACH-C à savoir la réussite simulée (*simulated annealing*) [125].

En effet, cet algorithme est découvert en 1982 par S. Kirkpatrick dans lequel le choix des solutions est basé sur une loi de probabilité (*distribution de Bolztmann*) qui mesure la probabilité $P(X)$ de visiter l'état $X$ en fonction de son énergie $E(X)$ et de la température $T$.

$$P(X) = e^{-\frac{E(X)}{kT}}/N(T) \qquad (6)$$

K : est la constante de Bolztmann.

Donc, cette méthode de formation de grappe centralisée permet de déterminer, à partir de la position exacte des nœuds, la configuration optimale pour minimiser l'énergie dépensée. Dès que les clusters sont formés, la station de base va passer à l'élection des CHs. Ces derniers sont choisis d'une manière très simple où seul le nœud qui a la plus grande réserve d'énergie est éligible de devenir le prochain cluster head. On peut améliorer le choix du CH en ajoutant des critères de choix supplémentaires comme par exemple la position du CH par rapport aux têtes de chaînes, c'est-à-dire choisir comme CH le nœud qui a la position la plus proche pratiquement de tous les nœuds à partir desquels il va recevoir les données.




**Figue 6.6:** Etapes de formation des grappes à chaînes.

En ce qui concerne la formation de chaînes, on a adopté la même idée utilisée dans PEGASIS où les nœuds de la même grappe forment une chaîne de proches voisins. Chaque nœud reçoit des donnée de l'un de ses voisins, fusionne (agrège) les données de ces derniers avec ses propres données et les envois à son tour à son autre voisin dans la chaîne. L'opération d'agrégation est exécutée au niveau de chaque nœud afin d'éliminer les informations redondantes et de réduire la quantité de données échangées pour préserver l'énergie. Afin de réduire le degré de latence introduit par la chaîne, on peut permettre à quelques nœuds de transmettre simultanément. Pour éviter les problèmes d'interférence, on peut codifier les transmissions ou bien permettre aux nœuds séparés géographiquement d'émettre au même temps.

En effet, nous avons proposé deux approches pour la création des clusters à chaines à savoir une approche dynamique et une approche statique.

2.2.3.1 Approche dynamique de formation de clusters à chaînes (HEEP-D)

L'approche dynamique nommée HEEP-D [190] consiste à former les clusters à chaines d'une manière périodique. Ainsi, l'algorithme de formation des clusters à chaines s'exécute à chaque phase d'initialisation. Cela permet une meilleure connectivité du réseau et permet l'addition de nouveaux nœuds *(réseau évolutif)*. Pour construire la chaîne, on a proposé un algorithme simplifié, détaillé dans la figure 6.7 et la figure 6.8. Ce dernier suit le même concept utilisé dans PEGASIS où l'on commence avec le nœud le plus éloigné de le CH *(choisir un noeud aléatoirement s'il y a un ensemble de nœuds possédant la même distance par rapport au CH)*. Ce nœud représente la tête de la chaîne. Ensuite, le nœud le plus proche de la tête de la chaîne est choisi pour être ajouté et devenir ainsi la nouvelle tête de la chaîne. Les voisins successifs sont sélectionnés de cette manière parmi les nœuds non visités. L'opération se répète jusqu'à ce que tous les nœuds soient dans la chaîne.

```
(1)   DEBUT
(2)   Chaine (i)          // la table de chaîne à construire,
(3)   Head                // la tête de la chaîne,
(4)   N                   // ensemble des nœuds appartenant à la même grappe,
(5)   i=1
(6)   Supprimer (N, CH)
(7)   Chaine (i) = {}
(8)   Head = le nœud le plus éloigné du CH
(9)   Supprimer (N, Head)
(10)  Tant que (N ≠ Φ) Faire
(11)   Nearest = le nœud le plus proche de la tête de chaîne
(12)   Si ( distance(Head,CH) > distance(Head, Nearest) )
(13)      Head = Nearest
(14)      Ajouter (Chain(i), Head)
(15)      Supprimer (N, Head)
(16)   Fin Si
(17)   Sinon
(18)      Ajouter (Chain(i), CH)
(19)      i=i+1
(20)      Goto (7)        //division de la chaîne
(21)   Fin Sinon
(22)  Fin Tant que
(23)  Fin
```

**Figure 6.7** : Algorithme de construction des chaînes.

En effet, l'algorithme proposé commence par le nœud le plus lointain pour s'assurer que les nœuds les plus loin du CH ont des voisins proches. Les distances voisines augmenteront graduellement puisque des nœuds déjà présents sur la chaîne ne peuvent pas être revisités. Quand un nœud meure, la chaîne est reconstruite de la même manière pour dévier le nœud mort.

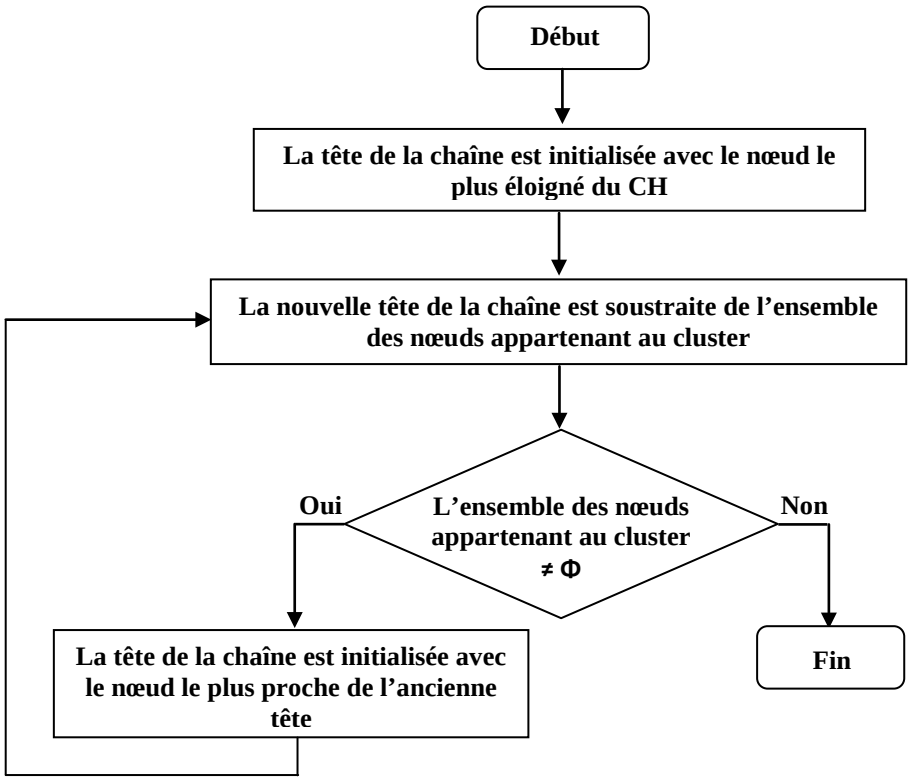



**Figure 6.8** : Organigramme de construction des chaînes.

Contrairement à PEGASIS, notre algorithme de formation de chaîne résout le problème de latence généré par la longueur de chaîne de transmission, et ce en la fragmentant en petites chaînes qui partagent la même tête de chaine à savoir le CH. Ainsi, chaque cluster peut contenir un nombre X de chaînes qui transmettent les données capturées simultanément au CH. Le nombre X dépend de la localisation géographique du CH dans le cluster. La figure 6.9 montre comment sera divisée la chaîne de nœuds.

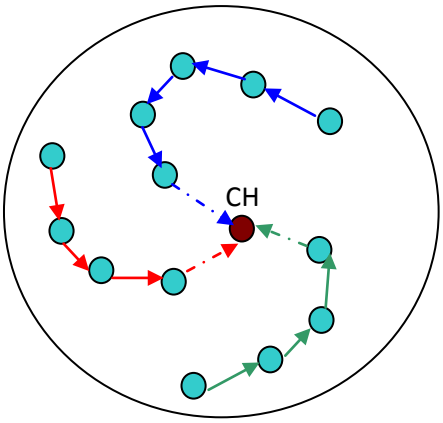



**Figure 6.9:** Division de la chaîne de nœuds.

2.2.3.2 Approche statique de formation de clusters à chaînes (HEEP-S)

L'idée de base de cette approche (HEEP_S) [190] est très simple et basée sur le même principe proposé par l'algorithme LEACH-F *(présenté dans le chapitre 4)*, où la création des clusters à chaînes se fait de manière statique. Dans ce cas, nous n'avons qu'une seule phase d'initialisation dans laquelle les clusters à chaîne sont formés, suivie par plusieurs cycles de transmission de données.

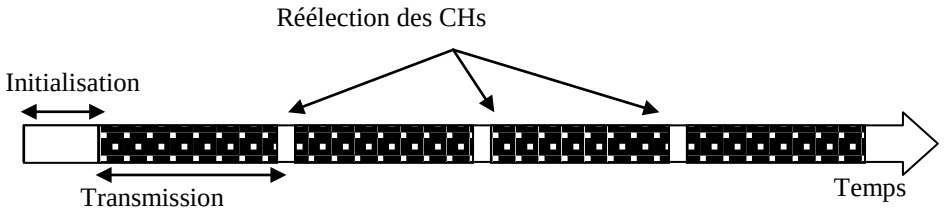



**Figure 6.10 :** Phases d'exécution de l'approche statique

Après plusieurs cycles de transmission, le rôle du cluster head est affecté à un autre nœud capteur dans la grappe, selon un ordre d'élection établi antérieurement par la BS (à la phase d'initialisation). Lorsque les clusters à chaîne sont organisés une seule fois, nous pouvons éviter que les nœuds perdent plus d'énergie due à la transmission périodique des paquets de contrôle vers la BS. Par ailleurs, le surcoût provoqué par cet échange *(problème d'overhead)* est considérablement réduit.

La phase de transmission est la même que celle utilisée par l'approche de formation de clusters à chaine dynamique. Chaque nœud transmet ses données recueillies à son proche voisin dans la chaine et se met dans un état endormi aussitôt après. Étant donné que la phase d'initialisation n'est accomplie qu'une seule fois, la BS devra établir un plan d'élection qui définit l'ordre d'élection du CH à l'intérieur de chaque cluster et à le transmettre à tous les nœuds du réseau. A chaque cycle de transmission, le cluster head sera réélu en fonction de l'ordre d'élection défini précédemment par la BS. L'algorithme qui sélectionne l'ordre d'élection des CHs est détaillé dans les figures 6.11 et 6.12.

```
(1)  DEBUT
(2)  Ordre_CH      // L'ordre d'élection des CHs
(3)  N             // Ensemble de nœuds appartenant au même cluster
(4)  CH            // Le nœud capteur élu comme cluster head
(5)  Ordre_CH = {}
(6)  Tant que (N ≠ Φ) faire
(7)    CH = le nœud capteur qui a la plus grande réserve d'énergie dans le cluster
(8)    Ajouter (Ordre_CH, CH)
(9)    Supprimer (N, CH)
(10) Fin tant que
(11) Fin
```

**Figure 6.11 :** Algorithme d'élection des cluster-heads

Le choix de l'ordre d'élection des CHs est très simple. Il est basé sur le niveau de la réserve énergétique des nœuds capteurs. Dans ce cas, les CHs seront élus dans l'ordre décroissant de leur réserve d'énergie.

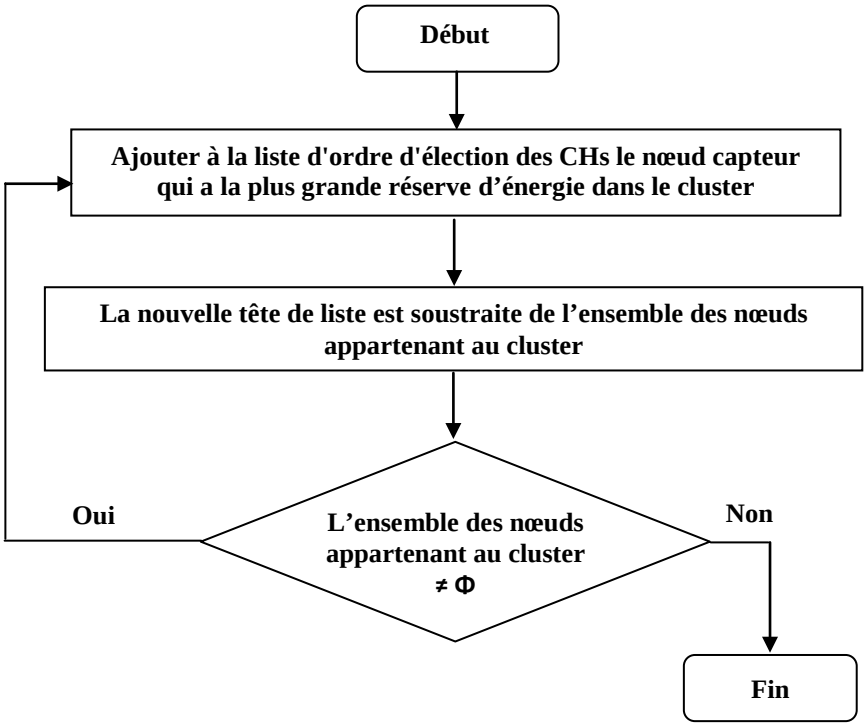



**Figure 6.12** Organigramme d'élection des cluster-heads

L'algorithme de formation de chaînes va être exécuté une seule fois (étape d'initialisation). Donc, la station de base devra construire une chaîne de nœuds pour chaque CH élu précédemment. Les informations des chaînes construites vont être assemblées sous forme de tables indexées par les clusters heads. La figue 6.13 présente un exemple de table de chaînes formées par la BS.

| CHi | Chaîne i |
|---|---|
| CH1 | Chaîne 1 |
| CH2 | Chaîne 2 |
| ⋮ | ⋮ |
| CHn | Chaîne n |

**Figure.6.13 :** Table de chaînes

Les tables de chaînes vont être transmises à tous les nœuds du réseau à la fin de l'étape d'initialisation. Au début de chaque cycle d'exécution, chaque nœud dans le réseau va router ses données collectées, en utilisant la chaîne respective, au cluster head élu précédemment pour ce cycle. Dans le cas où un nœud capteur est mort *(épuise sa réserve d'énergie)* les données sont routées directement à son voisin successif dans la chaîne afin d'éviter des pertes d'informations.

L'algorithme de formation de chaînes utilisé par la BS est détaillé dans la figure 6.14, et en fait une version adaptée de l'algorithme de construction de chaînes déjà proposé dans l'approche dynamique de formation des clusters à chaînes. Pour chaque cluster head élu, l'algorithme va construire une chaîne de noeuds différente afin de garantir que les distances de transmission soient optimales *(minimales)*.

```
(1)  DEBUT
(2)  Chaîne          // la chaîne à construire.
(3)  Tête            // la tête de la chaîne.
(4)  N               // ensemble de nœuds appartenant au même cluster.
(5)  Tab_chaine      // un tableau contenant les chaînes de transmission.
(6)  Ordre_CH        // L'ordre d'élection des CHs.
(7)  i = 1
(8)  Tant que (Ordre_CH ≠ Φ) faire
(9)      CH_i = Ordre_CH (i)          // sélection du prochain CH
(10)     Chaîne = {}
(11)     Tête = le nœud le plus éloigné du CH_i
(12)     N = N – Tête
(13)     Tant que (N ≠ Φ) faire
(14)       Tête = le nœud le plus proche de la tête de la chaîne
(15)       Ajouter (Chaîne, Tête)
(16)       Supprimer (N, Tête)
(17)     Fin tant que
(18)     N = ensemble des nœuds appartenant au même cluster
(19)     Supprimer (Ordre_CH , CHi)   // Mise à jour de la table de chaînes
(20)     Tab_chaine (i) = Chaîne
(21)     i = i+1
(22) Fin tant que
(23) Fin
```

**Figure 6.14 :** Algorithme statique de construction de chaînes

L'approche statique de formation des clusters à chaînes peut optimiser le temps de vie du réseau, ce qui augmente le nombre de phases de transmission de données comparé à l'approche dynamique. Nous pouvons démontrer cette amélioration analytiquement avec les équations suivantes :

$$T_{Ph\_dynamique} = (E_{Totale} - kE_{Phase\_initial})/E_{Ph} \quad (7)$$

Où $T_{Ph\_dynamique}$ est le nombre total de phases de transmission dans l'approche dynamique et k est le nombre de phases d'initialisation exécutées pendant le temps de vie de réseau.

$$T_{Ph\text{-}statique} = (E_{Totale} - E_{Phase\_initial})/E_{Ph} \Rightarrow T_{Ph\text{-}statique} > T_{Ph\_dynamique} \quad (8)$$

L'optimisation dans le nombre de cycles de transmission peut être calculée par :

$$(k\text{-}1)\, E_{Phase\_initial} / E_{Ph} \quad (9)$$

Bien que l'approche statique de construction des clusters à chaînes (HEEP_S) peut offrir une meilleure économie d'énergie comparée à l'approche dynamique (HEEP_D), la flexibilité et l'extensibilité du réseau demeurent limitées *(addition de nouveaux nœuds capteurs au réseau)*. Cependant, ce problème peut être évité en reconstruisant simplement les clusters à chaînes à chaque déploiement de nouveaux nœuds.

### 2.3 Protocole d'accès au média de transmission au niveau de la couche Mac

Afin d'accéder au média de transmission, nous avons proposé une version modifiée du protocole SMAC [22] *(Version Cross layer)*, où les périodes de mise en veille et d'activation des nœuds capteur sont dynamiques et dépendent des événements de transmission ou de réception des données [191]. En exploitant les informations de la couche Réseau, tous les nœuds qui ne sont pas inclus dans le chemin de routage sont mis dans un état de sommeil afin de réduire leur consommation d'énergie.

Afin de désactiver les nœuds inactifs, la plupart des protocoles proposés (MAC-CROSS, PARS, EPAR, CoLaNet…etc.) modifient les paquets de contrôle (RTS et CTS) en ajoutant l'adresse du destinataire final dans ces derniers. En interceptant ces paquets de contrôle, les nœuds capteurs peuvent déduire en utilisant leur table de routage qu'ils ne sont pas inclus dans le chemin de routage et se désactivent donc jusqu'à la fin de la transmission. Ainsi, les nœuds qui n'ont ni à transmettre ni à recevoir des données, ne vont pas être activés périodiquement à la fin de leur phase de sommeil pour écouter le trafic sur le réseau. Par conséquent, le problème d'activation répétitive *(Compulsory Wake up)* qui provoque une grande perte d'énergie est réduit. Cependant, cette approche est fortement conditionnée par la portée des antennes radio. Ainsi, les nœuds éloignés du chemin de routage ne peuvent pas intercepter les paquets RTS et CTS. Donc, ils continuent à s'activer périodiquement, ce qui entraine une mauvaise gestion d'énergie.

Afin de remédier à ce problème, nous avons proposé l'utilisation d'une deuxième antenne radio à très faible consommation d'énergie nommée antenne d'activation *(Wake up radio)*. Cette dernière possède une potée limitée et consomme beaucoup moins d'énergie comparée aux antennes radio classiques. Dans notre proposition, tous les nœuds utilisent en premier temps leur antenne d'activation afin d'écouter le trafic sur le réseau. Dans le cas où un nœud veut transmettre des données, il active automatiquement son antenne principale et envoi ensuite un message de contrôle nommé *Wake up tone* afin d'activer le nœud suivant dans le chemin de routage.

En interceptant ce message, le nœud récepteur active son antenne principale et se synchronise avec le nœud émetteur afin de recevoir les données à transmettre. Après l'acquittement des données transmises, le nœud transmetteur désactive son antenne principale et réactive son antenne d'activation afin d'économiser son énergie. Le récepteur active à son tour son nœud voisin dans la chaine de transmission. L'opération d'activation et de désactivation est répétée jusqu'à ce que l'on atteigne le CH. La figure 6.15 présente le mécanisme d'activation et de désactivation des nœuds capteurs.

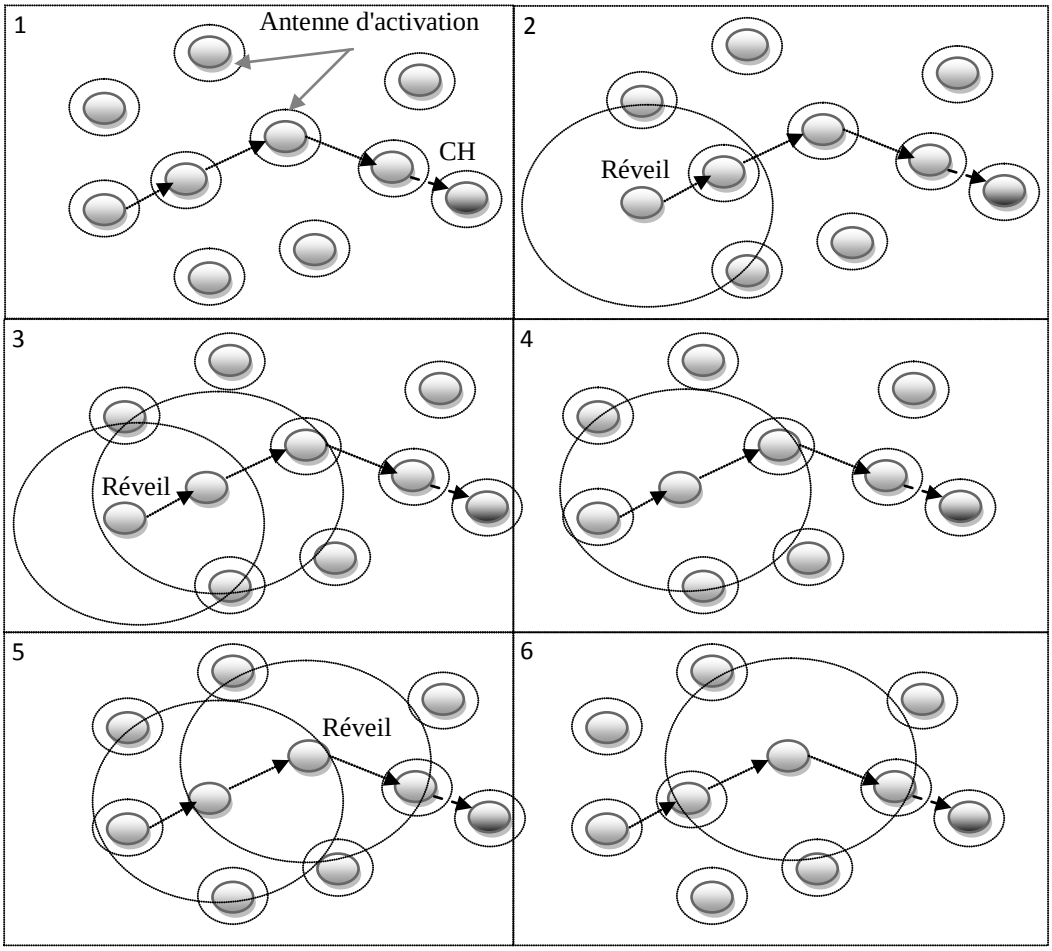



**Figure 6.15 :** Mécanisme d'activation et de désactivation des nœuds capteurs

En utilisant une deuxième antenne radio à faible consommation énergétique, on peut résoudre plusieurs problèmes qui sont à l'origine de beaucoup de perte d'énergie au niveau de la couche Mac. Le premier problème étant l'écoute passive *(idle listening)*. Ce dernier est considérablement réduit parce que les nœuds capteurs utilisent des antennes à très faible consommation énergique pour écouter la porteuse. Le deuxième problème est l'interférence entre les nœuds capteurs : Etant donné que les nœuds non impliqués dans le routage des données utilisent des antennes à faible portée, ils ne peuvent pas par conséquent interférer dans la transmission des autres nœuds. La résolution du problème précédent permet la réduction des problèmes de collision et de retransmission qui engendrent une consommation supplémentaire d'énergie.

En effet, l'utilisation d'une deuxième antenne comme antenne d'activation est déjà proposée dans [154]. Cependant, dans notre proposition, nous utilisons l'antenne principale pour envoyer les paquets d'activation *(wake-up tones)* alors que l'antenne d'activation est utilisée seulement pour l'écoute de la porteuse. Ainsi, nous pouvons réduire considérablement la portée de l'antenne d'activation, ce qui permet d'optimiser l'efficacité énergétique. Étant donné que la consommation d'énergie de l'antenne d'activation est extrêmement réduite, nous pouvons donc laisser l'antenne d'activation en mode d'écoute (Idle) tout le temps avant et après toute réception ou transmission de données, ce qui réduit la latence générée par la transmission des paquets. La Figure.6.16 présente un exemple des différents types de transitions radio pour trois nœuds adjacents sur le chemin de routage.

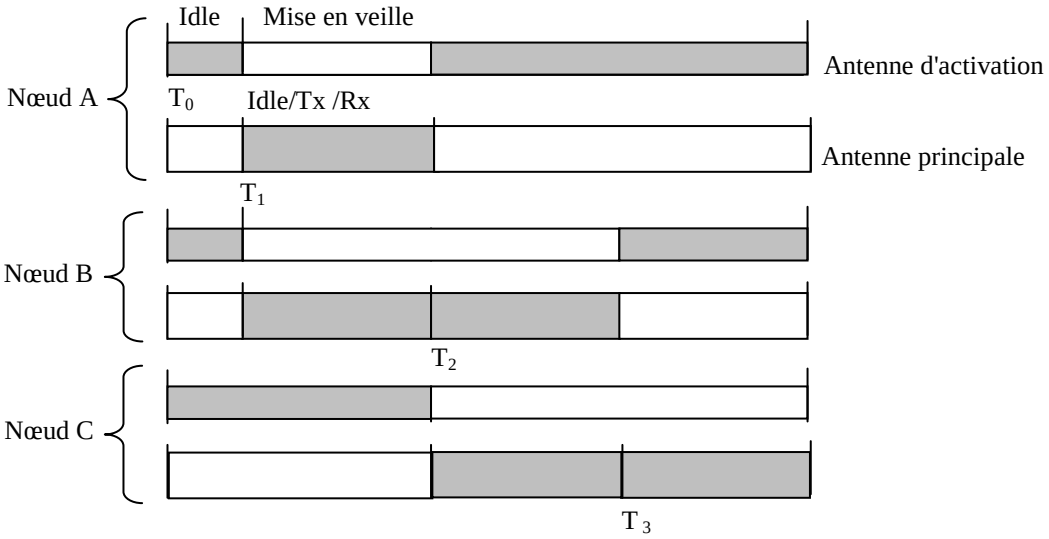



**Figure 6.16 :** Les différents types de transitions radio

Au temps T1, le nœud A bascule sur son antenne principale et réveille le nœud B (*prochain nœud dans le chemin de routage)*, en lui envoyant une balise de réveil *(Wake-up tone)*. Après avoir reçu le message de réveil, le Nœud B bascule sur son antenne principale, acquitte le paquet de réveille et commence à recevoir les données transmises par le nœud A. Au temps T2, le nœud B transmet une balise de réveil pour réveiller le nœud C. En même temps, le nœud A bascule sur son antenne d'activation pour préserver son énergie. De plus, le nœud C bascule sur son antenne principale afin de recevoir les données transmises à partir du nœud B. Enfin, au temps T3 le nœud C commence à transmettre les données reçues vers le nœud suivant alors que le nœud B bascule sur son antenne d'activation.

Plusieurs antennes radio à faible puissance ont été proposées pour les réseaux de capteurs sans fil [192, 193, 194 et 195]. Dans notre proposition, nous pouvons utiliser l'antenne radio TR1000 proposée par RF monolithique [195] comme antenne d'activation. Cette radio de faible puissance a un débit de données de 2,4 Kbps et utilise une modulation de type OOK. Le tableau 2 résume quelques nombres représentatifs pour les différents modes radio de l'antenne TR1000. Cependant, ces paramètres radio peuvent être réglés à un niveau plus bas, parce que la puissance et la portée de transmission sont considérablement réduites.

| Modes Radio | Consommation énergétique (mW) |
|---|---|
| Transmission (Tx) | 14.88 |
| Réception (Rx) | 12.50 |
| Idle | 12.36 |
| Mise en veille | 0.016 |

**Table 6.1 :** Caractéristique énergétique de l'antenne d'activation

Nous avons utilisé les réseaux temporels de pétri (TPN) [196] afin de valider formellement notre mécanisme d'activation et de désactivation des nœuds capteurs. Notre choix est basé sur leur habilité à modeler des contraintes temporelles et l'existence d'un analyseur et un simulateur de TPN nommé TINA. Les TPNs associent à chaque transition un intervalle de temps [Min et Max],

où Min et Max sont les temps minimum et maximum qu'une transition peut attendre avant d'être déclenchée partant du moment où elle est éligible. En accord avec les paramètres mentionnés dans les standards 802.11 [197], nous avons fixé le temps nécessaire pour la transmission ou la réception des paquets RTS, CTS, et Ack à 3 unités de temps. De plus, le temps nécessaire à la transmission d'un paquet de données et fixé à 10 unités de temps.

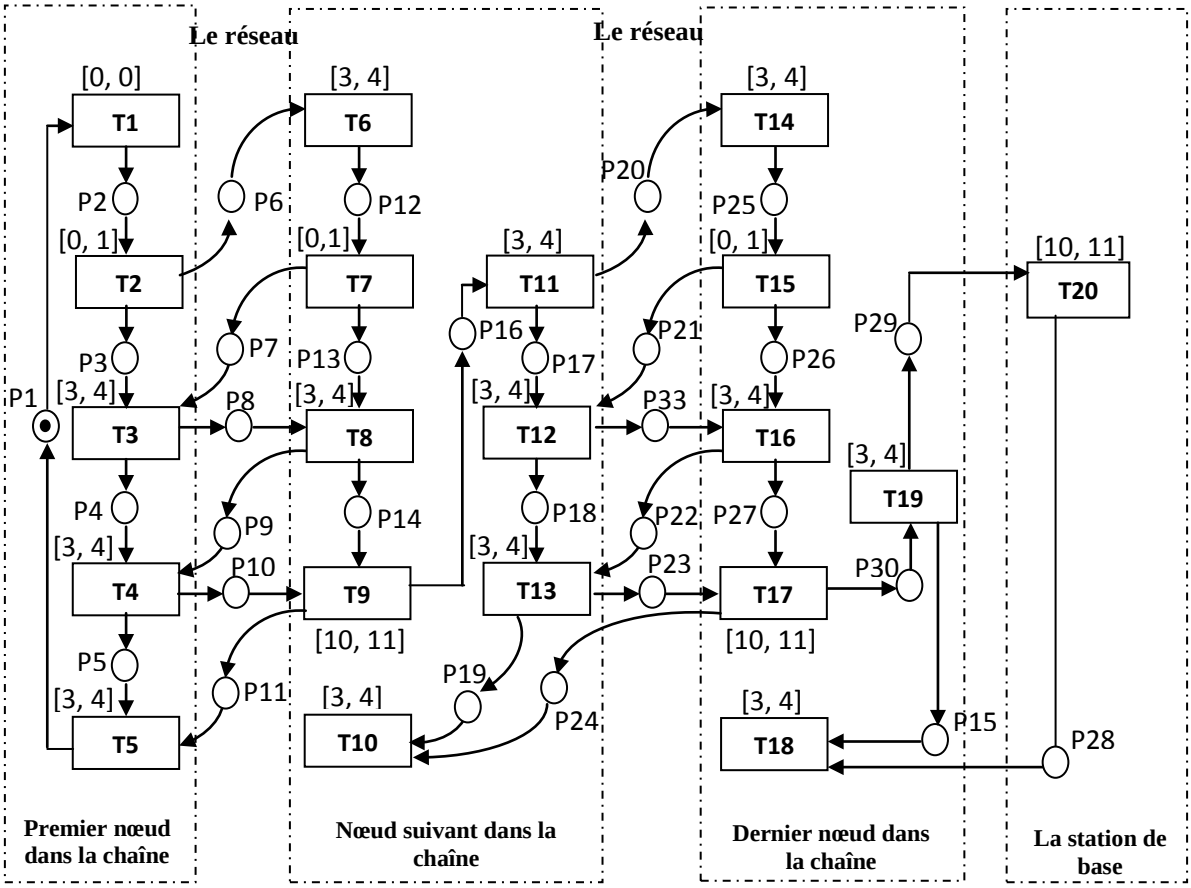



**Figure 6.17** : Modèle du mécanisme d'activation et de désactivation des nœuds capteurs réalisé avec les TPNs

La figure 6.17 représente le modèle du mécanisme d'activation et de désactivation des nœuds capteurs réalisé avec les TPNs. Le tableau qui suit décrit le rôle de chaque transition appartenant à notre modèle.

| Transition | Explication |
|---|---|
| T1, T6 et T14 | Activation de l'antenne principale du nœud capteur |
| T2 et T11 | L'envoi d'un Wake up tone au nœud récepteur |
| T7 et T15 | L'acquittement du Wake up tone |
| T3 et T12 | L'envoi d'un paquet RTS par le nœud émetteur |
| T8 et T16 | L'envoi d'un paquet CTS par le nœud récepteur |
| T4, T13 et T19 | Transmission de la donnée par le nœud émetteur |
| T9, T17 et T20 | L'acquittement des données transmises |
| T5, T10 et T18 | Désactivation de l'antenne principale et activation du Wake up radio |

**Tableau 6.2** : Les transitions du modèle et leur explication

Nous avons validé notre modèle à l'aide du simulateur TINA. Les résultats de l'analyse effectuée sont résumés dans la figure suivante :

```
LIVENESS ANALYSIS ……
Possibly live
0 dead classe(s), 30 live classe(s)
0 dead transition(s), 20 live transition(s)
STRONG CONNECTED COMPONENTS:
0: 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30
```

**Figure 6.18 :** Résultats de l'analyse effectuée sur notre modèle à l'aide du simulateur TINA

### 2.4 Approche d'ajustement d'énergie de transmission au niveau de la couche physique

La plupart des protocoles de communication (MAC CROSS, AREA MAC, XLM, CoLaNet, CLEEP…) utilisent des antennes radio à portée statique qui dépasse souvent les nœuds récepteurs dans le chemin de routage. Cela entraine une mauvaise gestion d'énergie et réduit le temps de vie des nœuds capteurs.

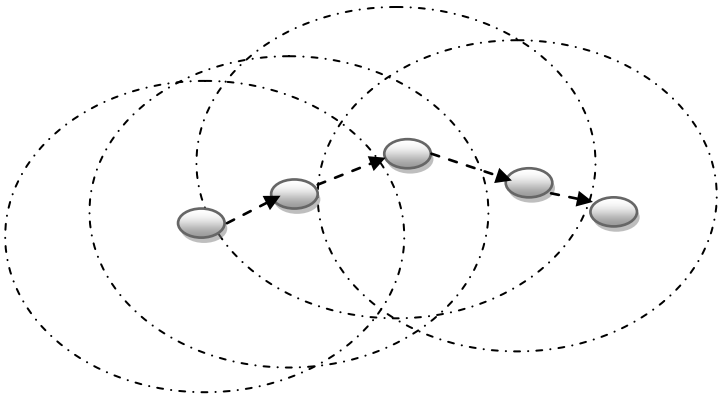


**Figure 6.19 :** Transmission basée sur des antennes radio statiques

Afin de remédier à ce problème, nous avons proposé un mécanisme de contrôle d'énergie de transmission afin d'ajuster dynamiquement la portée des antennes radio [198]. En se basant sur les informations de routage de la couche réseau, chaque nœud calcule l'énergie de transmission nécessaire pour transmettre à son nœud voisin dans le chemin de routage. Ainsi, chaque nœud ajuste la portée de son antenne radio pour atteindre uniquement son nœud voisin dans le chemin de routage, ce qui permet une meilleure économie d'énergie.

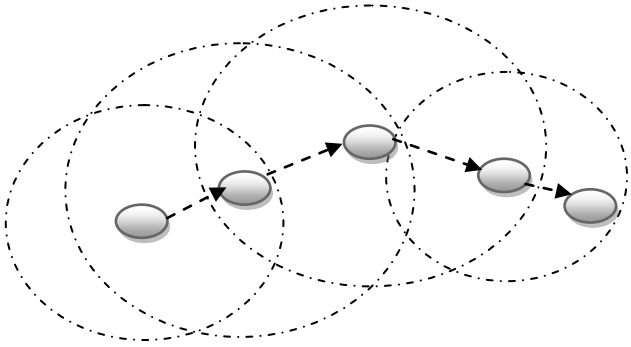


**Figure 6.20** : Transmission basée sur des antennes radio dynamiques (ajustables)

L'ajustement de la portée des antennes radio permet aussi de réduire les problèmes d'interférence qui représente une autre source de perte d'énergie.

Pour calculer l'énergie de transmission nécessaire pour transmettre à un nœud donné, il faut prendre en considération trois facteurs essentiels : la position géographique, la propagation du signal et les paramètres de transmission. Notre chemin de routage est représenté par la formule suivante :

$$Chemin = (N,Ps) \quad (10)$$

Où N est l'ensemble des nœuds voisins et Ps est la position géographique de ces derniers. Les paramètres de transmission d'un nœud sont représentés par le vecteur :

$$P = \{f_1, f_2 \ldots f_n\} \quad avec \quad f_i : N \rightarrow R \quad (11)$$

Où $R_i$ est la valeur ajustée du paramètre de transmission. Ce dernier peut inclure : l'énergie de transmission ($E_T$), la direction des antennes radio ($D_{rec}$), la hauteur des antennes radio ($H_r$), le code d'authentification du signal radio ($C_a$)…etc.

Dans notre proposition, nous supposons que les paramètres de transmission incluent uniquement l'énergie de transmission, ce qui nous donne :

$$P = \{E_T\} \quad (12)$$

La fonction de propagation du signal radio est définie par la formule suivante :

$$P_{rop}\ (Ps_x, Ps_y) \quad avec \quad P_{rop} : Ps\ X\ Ps \rightarrow Z \quad (13)$$

Où $P_{rop}$ retourne l'atténuation du signal radio *(mesurée en décibel 'dB')* provoquée par sa propagation de la position $Ps_x$ à la position $Ps_y$. La réception réussit du signal de transmission dépend de sa propagation, l'énergie de transmission du nœud émetteur et la sensibilité de l'antenne radio du nœud récepteur ($S_{ens}$). Ce dernier est le seuil minimum de la puissance du signal exigée pour la réception.

Nous supposons que la valeur $S_{ens}$ est une constante pour tous les nœuds du réseau. La formule suivante évalue la valeur de ce dernier :

$$E_T - P_{rop}\ (Ps_x, Ps_y) \geq S_{ens} \quad (14)$$

Nous assumons que l'atténuation du signal s'amplifie au fur et à mesure que la distance de propagation augmente. Cela est généralement accepté dans le cas où il n y a pas d'obstacles dans l'espace de transmission ou si ces derniers provoquent la même dégradation du signal sur toutes les directions. Ainsi, nous pouvons déduire que l'énergie de transmission nécessaire pour transmettre à un nœud voisin doit être au moins égale à:

$$E_T \geq P_{rop}\ (Ps_x, Ps_y) + S_{ens} \quad (15)$$

## 3. CONCLUSION

Prenant la sécurité de ressource et l'efficacité énergétique comme objectif principal, nous avons proposé un nouveau protocole de communication Cross-layer dédié pour la gestion efficace d'énergie. Notre idée de base est d'exploiter l'interaction entre les couches réseau, Mac et physique. En effet, le protocole CLEOP combine l'organisation de clusters à chaines, le cycle d'activation *(Duty-cycling)* Cross-layer, la radio d'activation à très faible consommation énergétique et l'ajustement de la portée dynamique des antennes radio, pour optimiser la consommation d'énergie et améliorer la durée de vie du réseau. Par conséquent, notre protocole régule la dissipation d'énergie, optimise les distances de transmission, réduit le nombre de nœuds inutilement activés et minimise la perte d'énergie générée par: l'écoute passive, overhearing, réveils répétitifs *(compulsory wake up)*, les interférences et les problèmes de collision. Dans le chapitre qui suit, nous allons introduire notre deuxième contribution, qui consiste en un système de détection d'intrusions basé sur une architecture d'interaction Cross-layer.

# Chapitre 7

## *Le système de détection d'intrusions Cross-layer proposé*

# 1. INTRODUCTION

Les systèmes de détection d'intrusions constituent une deuxième ligne de défense contre les attaques qui peuvent cibler le bon fonctionnement du réseau. Les réseaux de capteurs sans fil représentent l'un des domaines dans lesquels l'application de ces systèmes est plus que nécessaire. Cependant, les limitations des ressources et les caractéristiques environnementales *(communication sans fil, déploiement aléatoire…)* de ce type particulier de réseau, compliquent l'utilisation des SDIs proposés pour les réseaux classiques. Plusieurs recherches ont proposé des SDIs dédiés aux réseaux de capteurs sans fil. Néanmoins, ces SDIs sont basés sur des architectures en couche qui traitent le problème de sécurité au niveau d'une seule couche du modèle OSI. Ainsi, un SDI mono couche peut par exemple détecter et traiter les attaques malicieuses au niveau de la couche réseau, sans pour autant repérer celles qui opèrent au niveau de la couche liaison.

On peut déduire que ces systèmes de détection restent inefficaces et très vulnérables aux attaques qui peuvent cibler plusieurs couches du modèle OSI. Il parait évident d'utiliser différents SDIs au niveau de chaque couche protocolaire afin de remédier à ce problème de sécurité. Cependant, cela engendrerait une grande consommation de ressources *(puissance de calcul, réserves d'énergie, bande passante…)*, ce qui n'est pas souhaitable pour les RCSFs. Nous considérons que l'approche Cross-layer (inter couche) est l'une des solutions les plus prometteuses en termes de détection d'intrusions dans les réseaux de capteurs sans fil. En se basant sur cette approche, nous avons proposé un nouveau type de systèmes de détection d'intrusions, dans lequel la détection d'intrusions se fait à travers l'interaction de plusieurs couches du modèle OSI. Ainsi, au lieu d'utiliser un SDI au niveau de chaque couche, nous proposons l'application d'un seul SDI Cross-layer qui permet la détection d'intrusions au niveau de plusieurs couches du modèle OSI. Cela permet d'optimiser le niveau de sécurité tout en réduisant le taux de consommation en ressources.

Dans ce chapitre on va présenter notre nouveau système de détection que nous avons nommé CLIDS *(Cross Layer Intrusion Detection System )*, en mettant le point sur son principe de fonctionnement, son architecture, sa technique de détection Cross-layer et l'algorithme de décision d'intrusions.

# 2. CONCEPT DE BASE

Notre système de détection d'intrusions *(CLIDS)* [199, 200 et 201] est basé sur une architecture Cross-layer qui exploite l'interaction et la collaboration de trois couches adjacentes du modèle OSI à savoir : réseau, liaison et physique. Cette interaction permet l'établissement d'un modèle de comportement normal, à travers la combinaison d'informations recueillies au niveau des trois couches précédentes. L'idée de base est de détecter le nœud malveillant lorsqu'il tente d'établir une liaison de communication avec ses nœuds victimes. Cela est effectué au niveau de la couche liaison en se basant sur les informations issues des deux couches adjacentes (réseau et physique).

En effet, le système proposé n'exige pas la collecte de nouvelles informations pour la détection d'intrusions, et se contente des informations existantes. Ces dernières représentent la table de routage au niveau de la couche réseau, et la puissance du signal reçu *(RSSI : received signal strength indicator)* des nœuds voisins au niveau de la couche physique. Ainsi, chaque nœud voulant établir une liaison de communication *(envoi du paquet RTS)*, doit appartenir à la table de routage du nœud destinataire, et avoir une puissance de signal conforme à celle mémorisée par ce dernier.

| **Table d'informations de détection du nœud N2** | |
|---|---|
| ***Nœud voisin dans le chemin de routage*** | |
| ***Identificateur*** | ***RSSI*** |
| id N1 | RSSI N1 |
| id N3 | RSSI N3 |

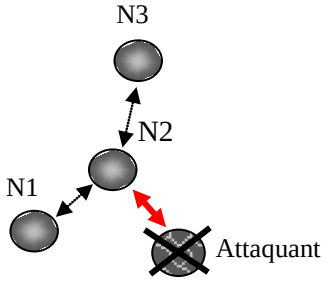



**Figure 7.1 :** Table d'informations de détection

En utilisant les informations de routage au niveau de la couche liaison *(sous couche MAC)*, chaque noud capteur peut prévoir les sources d'informations qui vont être reçues. Par conséquent, tout nœud essayant de communiquer avec un nœud légitime sera automatiquement déclaré comme suspect s'il n'appartient pas au chemin de routage de ce dernier. Cependant, le nœud attaquant peut copier l'identité de l'un des nœuds appartenant à la table de routage du nœud victime. Pour remédier à ce problème, nous avons proposé de combiner la puissance du signal reçu (RSSI) avec l'identificateur du nœud correspondant dans la table de routage. Le résultat de cette combinaison est une table d'informations de détection que nous avons nommée *TID*. Ainsi, chaque nœud mémorise l'RSSI de ses nœuds voisins dans le chemin de routage, ce qui permet d'identifier les nœuds malveillants présentant de fausses identités *(recopiés)*.

## 3. L'ARCHITECTURE CROSS-LAYER PROPOSEE

Il est clair que l'architecture de détection d'intrusions hybride *(centralisée et décentralisée)* est l'architecture la mieux adapté pour les RCSFs. Cependant, nous proposons d'optimiser les avantages de cette architecture en les combinant avec ceux de l'architecture Cross-layer. Par conséquent, notre système de détection d'intrusions est basé sur une architecture de détection Cross-layer et hybride, dans laquelle chaque nœud capteur implémente un agent de détection Cross-layer local nommé L-CLIDA *(Local Cross-Layer Intrusion Detection Agent)*. De plus, un agent de détection Cross-layer global G-CLIDA *(Global Cross-Layer Intrusion Detection Agent)* sera installé au niveau de la station de base.

Les agents L-CLIDA auront pour rôle la détection d'intrusions au niveau local des nœuds capteurs, tandis que l'agent G-CLIDA s'occupera de la détection d'intrusions au niveau de la station de base. La figure 7.2 présente l'architecture proposée.

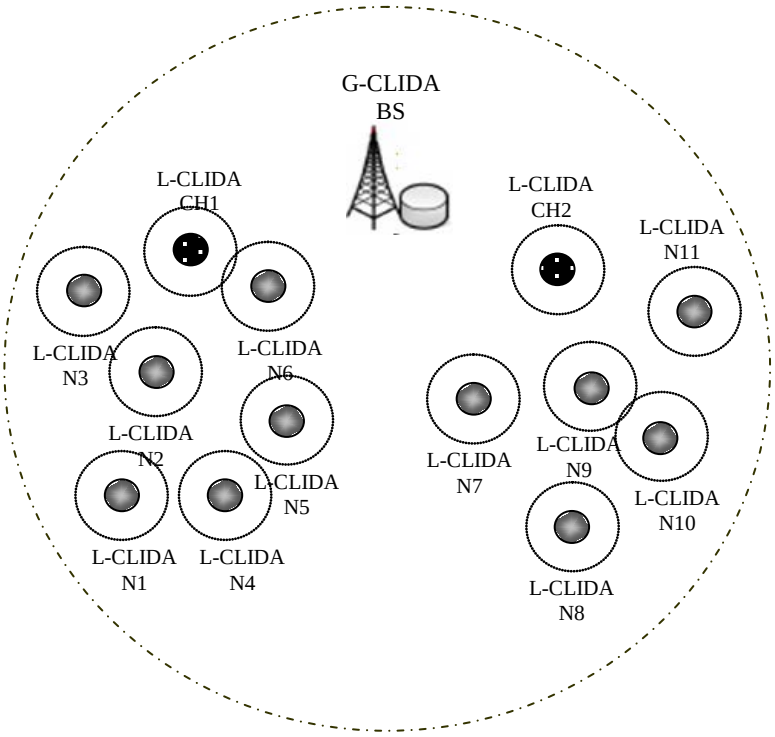



**Figure 7.2 :** L'architecture de détection d'intrusions Cross-layer

### 3.1 Architecture du système de détection local

Nous avons adopté la même architecture Cross-layer proposée précédemment pour le protocole CLEOP. Cette architecture consiste à conserver la structure traditionnelle en couches et considère l'interaction entre les couches réseaux, Mac et physique. Ainsi, les communications inter couches se font à l'aide d'une entité intermédiaire *(L-CLIDA)*. Comme présenté dans le chapitre 5, le choix d'une interaction Cross-layer indirecte offre une multitude d'avantages tel que: la compatibilité avec le modèle OSI, la flexibilité, l'évolutivité et la non redondance. La figure suivante présente l'architecture de détection Cross-layer proposée.

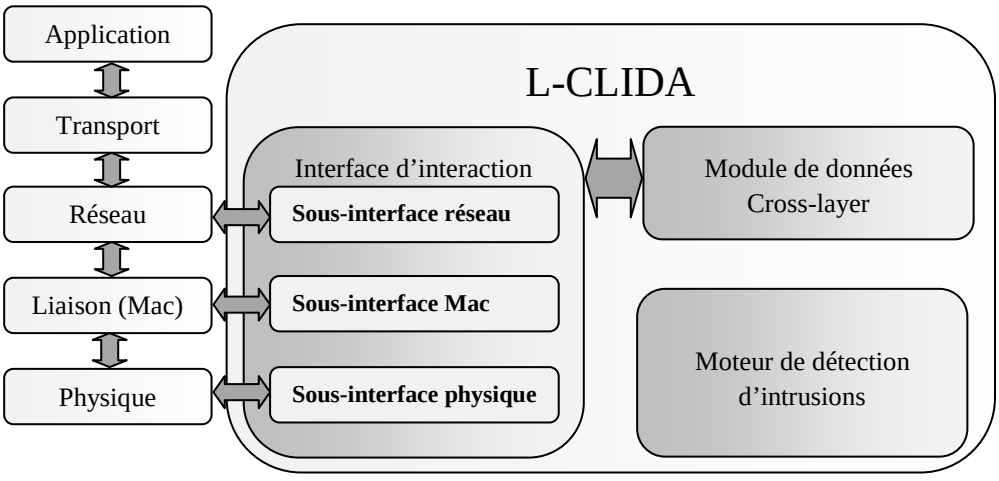



**Figure 7.3 :** L'architecture Cross-layer pour la détection d'intrusion locale

***3.1.1 L'interface d'interaction de l'agent L-CLIDA :*** elle se compose de trois sous interfaces d'interaction *(IR : Interface réseau, IL : Interface Liaison et IP : Interface physique)* dont le rôle est la collecte d'informations de détection *(table de routage, valeur des RSSI)* pour le module de données Cross-layer.

***3.1.2 Le module de donnée Cross-layer de l'agent L-CLIDA :*** toutes les données collectées par l'interface d'interaction seront stockées par la suite dans le module des données Cross-layer. Ce dernier s'occupe de la création de la table d'informations de détection Cross-layer (TID) afin d'être exploitée par le moteur de détection d'intrusions. Le module est aussi responsable du maintien à jour de la table TID à travers les interfaces d'interaction Cross-layer.

***3.1.3 Le moteur Cross-layer de détection d'intrusions :*** le moteur de détection Cross-layer analyse toutes les demandes de connexion reçues *(paquets RTS)*, en se basant sur un modèle de comportement normal créé à partir de la table TID. L'algorithme de détection d'intrusions utilisé est très simple et n'exige pas une grande puissance de calcul, ce qui est adapté pour les RCSFs.

### 3.2 Architecture du système de détection global

Au niveau de la station de base un agent de détection global G-CLIDA sera implémenté, afin de détecter les intrusions au niveau du réseau. L'agent G-CLIDA se base sur la même architecture que celle de l'agent L-CLIDA, néanmoins il utilise une table TID qui combine l'identificateur et l'RSSI de tous les nœuds légitimes appartenant au réseau. La figure suivante présente la table TID utilisée par l'agent G-CLIDA.

| **Table d'informations de détection de la BS** | |
|---|---|
| ***Nœud légitime appartenant au réseau*** | |
| ***Identificateur*** | ***RSSI*** |
| id N1 | RSSI N1 |
| id N2 | RSSI N2 |
| . | . |
| . | . |
| Id Nn | RSSI Nn |

**Figure 7.4 :** Table d'informations de détection de l'agent G-CLIDA

## 4. MODELE DE BASE DU SYSTEME DE DETECTION PROPOSE

Notre système de détection est construit autour d'un modèle de base qui décrit l'ensemble des règles et des assumassions nécessaires pour le bon fonctionnement de ce dernier. Le modèle proposé comprend trois sous modèles à savoir : le modèle de communication, le modèle d'attaquant et le modèle de sécurité.

### 4.1 Le modèle de communication

Nous assumons le même modèle de communication proposé pour le protocole CLEOP, dans lequel une architecture d'interaction Cross-layer est adoptée afin d'organiser la communication des nœuds dans le réseau. Ainsi, le réseau sera organisé sous forme de clusters à chaines, dont chacun possède un cluster head qui se charge de gérer les communications internes et externes du cluster. Toutes les données seront transmises aux cluster heads respectifs en passant par les chaines de nœuds adjacents *(en utilisant le protocole HEEP)*. Ces derniers se chargent de les transmettre par la suite vers la station de base. En effet, l'organisation du réseau se fait d'une manière centralisée, dans laquelle la station de base est responsable de la formation des clusters à chaine et de l'élection des nœuds cluster heads, en se basant sur les informations *(Id, réserves d'énergie, position géographique)* reçues périodiquement *(durant la phase d'initialisation)* par tous les nœuds du réseau. Les données de la nouvelle organisation *(tables de routage, plans d'activation, valeurs de portée radio, RSSIs…)* seront transmises par la suite à tous les nœuds du réseau, afin qu'ils s'auto organisent.

Le protocole SMAC est utilisé au niveau de la couche liaison afin de gérer l'accès au média de transmission. Ce dernier est basé sur l'utilisation du mécanisme de connexion RTS/CTS, dans lequel le nœud émetteur doit avoir l'autorisation du nœud récepteur avant de lui envoyer ses données. Ainsi, le nœud émetteur envoi une demande de connexion RTS *(Request to Send)* au nœud destinataire, qui va répondre par l'envoi d'une acceptation de connexion CTS *(Clear To Send)*. A la réception de celle-ci, le nœud émetteur peut par la suite transmettre ses données au nœud récepteur.

L'ajustement dynamique de la portée des antennes radio est adopté au niveau de la couche physique, en se basant sur les informations de routage établies au niveau de la couche réseau. Nous assumons que les valeurs de portée et d'RSSIs seront calculées de façon centralisée par la station de base durant la phase d'initialisation du protocole CLEOP. En effet, les valeurs d'RSSIs peuvent être affectées par la démunissions de la puissance énergétique des nœuds capteurs. Ainsi, nous assumons que chaque nœud capteur doit avoir un certain seuil d'énergie pour pouvoir transmettre ses données *(stabilité de la valeur RSSI)*.

### 4.2 Le modèle d'attaquant

Le modèle d'attaquant définit les caractéristiques du nœud malveillant ainsi que la stratégie des attaques qui peuvent être lancées par ce dernier. Dans notre modèle, le nœud attaquant est capable de lancer plusieurs types d'attaques au niveau de différentes couches du modèle OSI. Nous assumons que ce dernier dispose de plus de ressources, comparé à celles disponibles pour les nœuds capteurs. Le nœud malveillant peut cibler d'une manière aléatoire ses nœuds victimes, ou d'être plus malicieux en ciblant les nœuds cluster heads. Enfin, nous assumons que l'attaquant passe par une période passive dans laquelle il se contente d'écouter le trafic dans le réseau, ensuite il passe à un état actif durant lequel il lance ses attaques malicieuses.

### 4.3 Le modèle de sécurité

Nous assumons que tous les identificateurs des nœuds capteurs seront mémorisés par la station de base avant le déploiement du réseau. Ainsi, la construction des chemins de routage *(établissement des tables de routage, élection des CHs)* est entièrement sécurisée, étant donné que les nœuds attaquants seront automatiquement détectés et rejetés par la BS.

## 5. *L'APPROCHE DE PRISE DE DECISION*

Le système de détection proposé se base sur une approche de prise de décision indépendante, dans laquelle chaque nœud capteur peut détecter et décider sur la nature de l'intrusion rencontrée. En effet, le nœud capteur n'a pas besoin de collaborer avec d'autres nœuds dans le réseau afin de détecter les intrusions possibles. Cela permet d'offrir une meilleure gestion des ressources, comparée à l'approche de détection collaborative. En outre, les nœuds malveillants ne peuvent pas affecter la décision des nœuds capteurs *(données de détection falsifiées)*, ce qui optimise le niveau de sécurité de notre SDI.

## 6. *TECHNIQUE DE DETECTION D'INTRUSIONS CROSS-LAYER*

Afin de détecter les différents types d'attaques *(connues et inconnues)*, le système CLIDS adopte une technique de détection à base d'anomalie de comportement. Cette technique s'attache à définir le comportement normal du système, et considère comme étant une intrusion, toute déviation par rapport à ce comportement de référence. Notre choix est basé sur l'adaptabilité de la détection à base d'anomalie, aux limitations des ressources des nœuds capteurs.

Notre modèle de comportement normal est construit essentiellement autour de deux règles de comportement, basées sur la table d'informations de détection Cross-layer *(TID)*. Ces règles sont définies comme suit:

- ***Règle d'appartenance à la table d'informations de détection* :**

  « N'importe quel nœud désirant communiquer avec un autre nœud dans le réseau, doit être inclus dans la table d'informations de détection de ce dernier ».

- ***Règle de conformité de l'RSSI :***

  « Les paquets reçus du nœud émetteur doivent avoir une puissance de signal *(RSSI)* conforme à celle mémorisée dans la table TID ».

Le modèle de comportement ainsi définit est très simple, et n'exige pas une phase d'apprentissage souvent complexe et consommatrice en ressources. De plus, il sera établi durant la phase d'initialisation du protocole de communication *(CLEOP)*, en se basant sur les informations existantes *(table de routage et valeur RSSI)*.

# 7. ALGORITHME DE DETECTION D'INTRUSIONS CROSS-LAYER

L'algorithme de détection d'intrusions proposé est divisé en deux phases essentielles : la phase d'initialisation et la phase de détection. La première phase comprend la création et la mise à jour des tables d'informations de détection (TIDs), et l'initialisation des agents de détection au niveau de tous les nœuds du réseau. Dans la phase de détection, les nœuds capteurs vont surveiller tous les paquets reçus afin de détecter d'éventuelles intrusions. Le déroulement de l'algorithme de détection s'effectue d'une manière périodique et en parallèle avec le protocole de communication. Par conséquent, la phase d'initialisation s'exécutera avec celle du protocole de communication (CLEOP), tandis que la phase de détection sera exécutée pendant l'exécution de la phase de transmission. La figure 7.5, illustre les étapes essentielles de notre algorithme.

## 7.1 La phase d'initialisation

Durant la phase d'initialisation, la station de base se charge de la création des tables d'informations de détection en se basant sur les informations collectées du protocole de communication *(tables de routage et valeurs d'RSSI)*. Ensuite, les TIDs seront diffusées à tous les nœuds du réseau afin d'initialiser ou de mettre à jour leurs agents locaux de détection. De plus, l'agent de détection global sera initialisé au niveau de la station de base, en se basant sur les identificateurs de tous les nœuds légitimes mémorisés précédemment *(avant le déploiement)* dans la BS. Ce dernier sera mis à jour dans le cas d'ajout ou de révocation des nœuds capteurs.

## 7.2 La phase de détection

Après avoir mis à jour sa table TID, chaque nœud capteur tentera de détecter localement les nœuds malicieux qui peuvent le cibler. A la réception d'une demande de connexion *(paquet RTS)*, le nœud récepteur consultera sa table TID afin de vérifier l'appartenance du nœud émetteur à sa table de routage. De plus, l'authenticité du nœud émetteur sera identifiée, en comparant la puissance du signal du paquet RTS reçu avec celle sauvegardée dans la table TID. Ainsi, en se basant sur les deux règles de comportement normal, le nœud récepteur peut décider sur la nature du nœud émetteur. Dans le cas où un nœud malicieux sera identifié, une alarme sera envoyée à la BS afin d'entreprendre une action de prévention. Cette action consiste à isoler le nœud intrus, en évitant de l'inclure dans les prochains chemins de routage *(l'exclure des tables TIDs)*. L'action de prévention peut être aussi une action physique, qui consiste à détruire ou enlever le nœud intrus.

L'envoi de l'alarme peut être d'une manière directe (sécurisée) ou en utilisant le chemin de routage multi sauts *(économique en énergie)* établit par le protocole de routage. La transmission directe permet d'éviter que le paquet d'alarme sera intercepté par un autre nœud malveillant. Cependant, cette approche peut être consommatrice en énergie comparée à l'approche de transmission multi sauts *(moins sécurisée)*.

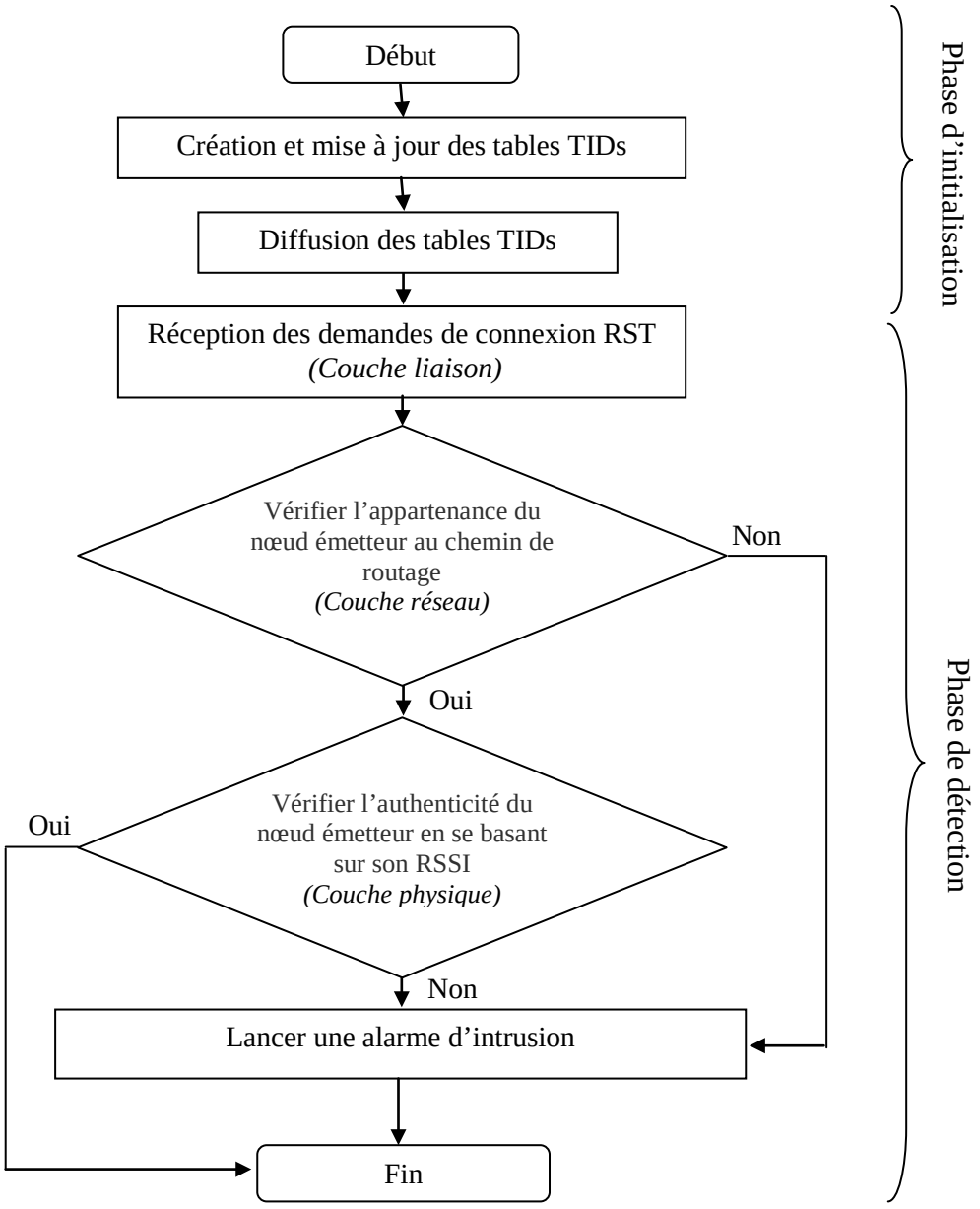



**Figure 7.5 :** Algorithme de détection d'intrusions

## 8. ANALYSE ANALYTIQUE DU SYSTEME PROPOSE

Dans cette section, nous allons analyser analytiquement les performances de notre système de détection d'intrusions Cross-layer, en mettant le point sur sa probabilité de détection et son degré de consommation énergétique.

### 8.1 La probabilité de détection d'intrusions

La probabilité de détection d'un nœud intrus dépend essentiellement de deux facteurs : le nombre de nœuds attaqués par les nœuds malicieux, ainsi que la probabilité de ne pas détecter une intrusion. Nous définissons les deux variables A et $P_{Det}$ qui représentent respectivement le nombre de nœuds attaqués dans le réseau, et la probabilité de détection d'intrusions.

Nous assumons la possibilité qu'un nœud ne détecte pas une intrusion, dans le cas où il ne reçoit pas de messages du nœud intrus *(attaque passive)*. Par conséquent, la probabilité de ne pas détecter une intrusion est équivalente à la probabilité qu'une collision s'effectue dans le lien de transmission. Cette probabilité sera définie par la variable $P_{Col.}$

En se basant sur la loi Binomial, on peut définir la probabilité de détection d'intrusions d'un seul nœud intrus par l'équation suivante :

$$P_{Det=}\binom{A}{1}(1- P_{Col})\, P_{Col}^{A-1} \quad (1)$$

Ainsi, la probabilité de détecter X nœuds intrus sera définit comme suite :

$$P_{Det=}\binom{A}{X}(1- P_{Col})^{X}\, P_{Col}^{A-X} \quad (2)$$

Dans notre DSI, tous les nœuds capteurs peuvent détecter les éventuelles intrusions, et la probabilité de détection augmente graduellement avec l'expansion des nœuds attaquants dans le réseau. Par contre, la majorité des SDIs existants doivent augmenter le nombre de nœuds moniteurs, afin d'améliorer leur capacité de détection, ce qui n'est pas souhaitable pour les RCSFs.

Nous assumons que les nœuds intrus peuvent attaquer tous les nœuds capteurs dans leur champ de couverture radio. Par conséquent, nous pouvons estimer le nombre de nœuds attaqués par l'équation suivante :

$$A= (N-1)\, \pi\, r^2 / a \quad (3)$$

Où a représente la surface de couverture radio, N le nombre de nœuds présents dans la surface de couverture et r est le radius de la surface a.

### 8.2 Le niveau de consommation énergétique

Notre système de détection d'intrusions introduit un faible taux de consommation énergétique, comparé aux systèmes de détection existants. Cette consommation est due à l'échange de messages de contrôle entre la station de base et les nœuds capteurs, ainsi que l'exécution périodique de l'algorithme de détection d'intrusions. Afin d'estimer la consommation énergétique de notre SDI, nous allons calculer d'abord la consommation d'énergie au niveau d'un nœud capteur durant une seule itération de l'algorithme de détection (*phase d'initialisation + phase de détection)*. En effet, la consommation énergétique d'un nœud capteur durant la phase d'initialisation *($E_{init}$)*, équivaut à l'énergie consommée pour recevoir un paquet de contrôle (table TID) de la BS. Cette énergie peut être mesurée par l'équation suivante :

$$E_{init} = T_{Cnt} x E_{elect} \quad (4)$$

Où $T_{Cnt}$ est la taille d'un paquet de contrôle envoyé par la BS, et $E_{elect}$ est l'énergie consommée par les circuits électroniques.

La consommation énergétique de la phase de détection est un peu plus grande à celle de la phase d'initialisation. Celle-ci est proportionnelle au nombre de paquets envoyés par le nœud intrus *($Nbr_{intrus}$)* et le nombre de détections signalées à la BS *($Nbr_{Alarme}$)*.

$$E_{Détecte} = (Nbr_{intrus} x E_{elect}) + (Nbr_{reçu} x E_{calcule}) + (Nbr_{Alarme} x E_{Tx}) \quad (5)$$

Où $Nbr_{reçu}$ représente le nombre total des paquets reçus, $E_{calcule}$ est l'énergie consommée par l'exécution de notre algorithme de détection d'intrusions, et $E_{tx}$ est l'énergie consommée pour la transmission d'une alarme à la BS. Celle-ci sera calculée par l'équation suivante :

$$E_{Tx} = T_{Alarme} x (E_{elect} + E_{fs} d^{2}_{toBS}) \quad (6)$$

Où $T_{Alarme}$ est la taille d'un paquet d'alarme transmis vers la BS, $E_{fs}$ est l'énergie perdue dans l'espace de transmission, $d_{toBS}$ est la distance géographique entre un nœud émetteur et la station de base.

Ainsi, le total d'énergie consommée pendant une seule itération de l'algorithme de détection sera équivalent à :

$$E_{itération} = E_{init} + E_{Détecte} \quad (7)$$

Le nombre d'itérations de l'algorithme de détection d'intrusions est proportionnel à la durée de vie du réseau. Donc, nous pouvons estimer la consommation totale de notre SDI par l'équation 8 :

$$E_{SDI} = \sum_{i=0}^{i=Nbr\_itération} E_{itération} \quad (8)$$

Contrairement aux systèmes de détection d'intrusions existants *(à base de nœuds moniteurs)*, le taux de consommation énergétique de notre SDI décroit avec la démunissions du nombre de nœuds attaquants. Cela permet une meilleure gestion d'énergie et prolonge le temps de vie du réseau.

## 9. DETECTION D'ATTAQUES A TRAVERS LE PROTOCOLE CLIDS

Dans cette section, nous allons présenter le comportement de notre SDI face à plusieurs types d'attaques qui opèrent au niveau de différentes couches protocolaire.

### 9.1 Attaques au niveau de la couche réseau

Les attaques qui opèrent au niveau de la couche réseau peuvent être détectées par notre SDI lorsque les nœuds attaquants essayent de se connecter avec les nœuds victimes *(au niveau de la couche liaison)*. En effet, le nœud attaquant doit informer le nœud ciblé qu'il désire lui envoyer des données *(envoi de paquet RTS)*. Ce dernier accepte de recevoir ces données par l'envoi de paquet CTS. Ainsi, notre SDI permet de vérifier l'identité du nœud attaquant lors de la réception du paquet RTS, et rejeter tout type de communication avec ce dernier. Dans le cas où l'attaquant

ne respecte pas les règles de communication *(envoi directement ses données sans demander la permission du nœud récepteur),* il sera signalé comme un nœud intrus est rejeté du réseau.

Dans l'attaque de trou de puits, l'attaquant essaye de s'intégrer dans le chemin de routage afin de lancer d'autres types d'attaques tel que l'attaque de trou noir ou de routage sélectif. De plus, l'attaquant peut être malicieux en essayant de se faire sélectionner comme cluster head, et causer par conséquence plus de dégâts. Notre SDI permet de détecter facilement cette attaque au niveau de la BS. En effet, l'agent de détection global G-CLIDA vérifie l'identificateur de tous les nœuds qui désirent participer au processus de routage. Par conséquent, tout nouds intrus sera directement identifié, étant donné que son identificateur n'existe pas dans la table TID au niveau de la BS.

L'attaque d'usurpation d'accusés de réception constitue une autre attaque qui peut déstabiliser le fonctionnement du réseau. L'attaquant espionne le trafic sur le réseau et imite les accusés de réception échangés entre les nœuds capteurs. En recevant ces faux accusés de réception, les nœuds émetteurs supposent que leurs données ont été bien reçues. Etant donné que l'accusé de réception contient l'identificateur du nœud malveillant, notre SDI peut reconnaitre l'identité de l'attaquant *(n'appartient pas à la table TID)*. Par conséquent, l'accusé de réception sera rejeté, et le nœud malveillant sera signalé. L'attaquant peut aussi usurper et modifier les informations de routage afin de perturber le fonctionnement du réseau. Cette perturbation peut entrainer l'empoisonnement des tables de routage, ce qui est en mesure de créer des boucles de routage, des chemins de routage très coûteux, la congestion et les débordements des tables de routage. Comme l'attaque d'usurpation d'accusés de réception, notre SDI peut détecter l'identité des nœuds malveillants et rejeter par conséquence toutes les informations qui proviennent de ces derniers.

L'attaque Sybil est l'une des attaques les plus difficiles à détecter, étant donné que l'attaquant peut utiliser des identités recopiées des nœuds légitimes. Cette attaque est généralement combinée avec d'autres types d'attaques tel que le trou de puits ou l'usurpation des données, afin d'augmenter leur degré de malveillance. En effet, l'identificateur de l'attaquant peut appartenir à la table TID, ce qui lui permet de ne pas être détecté. Cependant, en combinant la valeur RSSI à l'identificateur du nœud capteur, notre SDI peut détecter l'identité copiée du nœud Sybil car la puissance du signal reçu ne sera pas conforme à celle du nœud légitime.

Le nœud attaquant peut transmettre un grand nombre de paquets de contrôle *(d'inondation par paquets de Hello)* ou de faux paquets de données *(Attaque d'information fabriquée)*. En utilisant notre SDI, les nœuds capteurs peuvent détecter la nature malveillante des nœuds émetteurs, et rejeter tous les paquets envoyés. En effet, l'identité des nœuds malveillants sera détectée lorsqu'ils essayent d'établir une liaison de communication avec les nœuds légitimes *(par envoi de paquet RTS au niveau de la sous couche Mac).* Ainsi, les nœuds légitimes peuvent rejeter les demandes de connexion *(n'envoient pas de paquet CTS),* et se mettent directement dans un état de sommeil afin de ne pas recevoir les données corrompues.

### 9.2 Attaques au niveau de la couche Mac

CLIDS permet de détecter les attaques au niveau de la sous couche Mac *(couche liaison),* en se basant sur les informations Cross-layer obtenues des deux couches réseau et physique. L'objectif de la plupart des attaques qui ciblent cette couche (Mac), est l'épuisement des réserves d'énergie

des nœuds capteurs en perturbant leur mécanisme de *Duty-cycling*, ou en provoquant des collisions et des retransmissions des données. CLIDS essaye de prévenir ces attaques en rejetant toutes demandes de connexion ou d'informations envoyées par les nœuds malveillants.

Dans l'attaque de privation de sommeil, tout nœud recevant un paquet RTS doit vérifier l'identité du nœud émetteur avant d'envoyer son paquet CTS. Ainsi, il rejette le paquet RTS et entre en mode de veille si le nœud émetteur n'appartient pas à sa table de routage. Dans le cas contraire, il envoi un paquet CTS et prolonge l'état éveillé *(actif)* afin de recevoir les données qui vont être envoyées. La figure 7.6 présente un exemple de l'attaque de privation de sommeil dans laquelle deux nœuds attaquants tentent respectivement d'épuiser l'énergie des nœuds A et B. En outre, le nœud A ne possède aucun mécanisme de sécurité, tandis que le nœud B est sécurisé par notre système de détection d'intrusions.

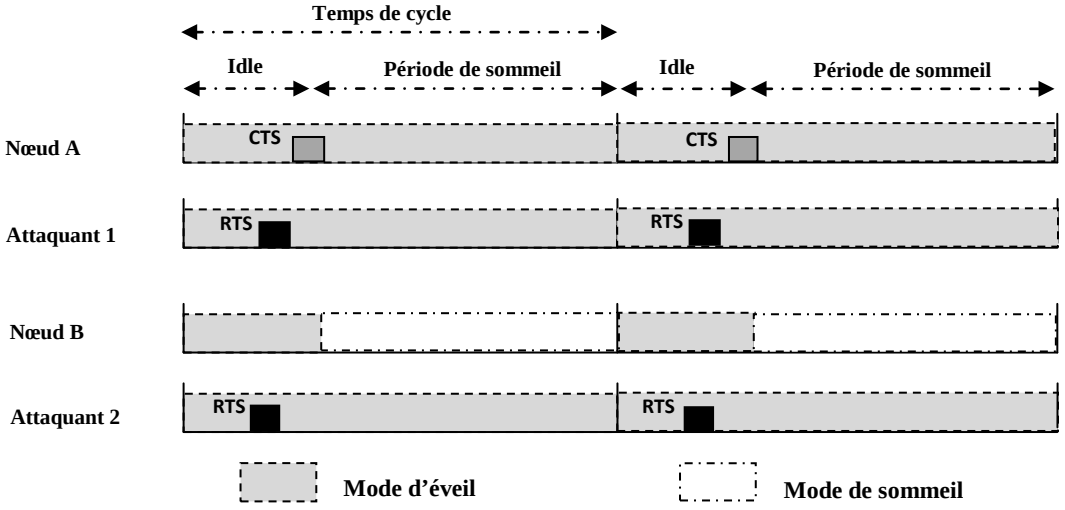



**Figure 7.6 :** Exemple d'attaque de privation de sommeil.

Comme l'attaque de privation de sommeil, l'attaque de barrage force le nœud victime à rester éveillé, mais en plus, elle l'oblige à effectuer des opérations à forte intensité énergétique tel que la réception ou la transmission des données. En rejetant le paquet RTS envoyé à partir du nœud attaquant, CLIDS peut prévenir l'attaque de barrage et empêche les nœuds ciblés de rester éveillés pour accomplir tout genre de tâches exhaustives en énergie. La figure 7.7 présente un exemple du comportement d'un nœud sécurisé avec notre SDI, sous l'attaque de barrage.

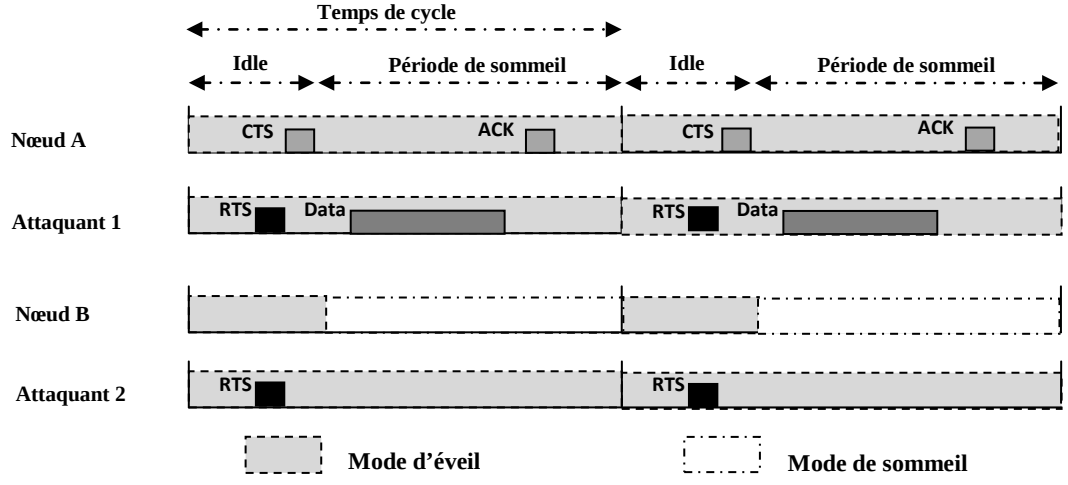



**Figure 7.7 :** Exemple d'attaque de barrage.

Dans l'attaque de synchronisation, le nœud attaquant a connaissance du protocole Mac, et envoi des paquets de synchronisation dont les valeurs temps de sommeil sont corrompues. Ainsi, les faux paquets de synchronisation suffisent à maintenir les nœuds ciblés dans un état éveillé. Etant donné que les messages de synchronisation contiennent l'identifiant de l'expéditeur, le nœud récepteur peut rejeter et ignorer ces paquets si l'expéditeur ne figure pas dans sa table de routage. En effet, notre SDI peut donner le même résultat fourni par le mécanisme proposé en [20], en outre, il est plus économe en énergie par rapport au protocole proposé dans [60], car il n'y a pas de surcharge due à l'échange périodique des faux calendriers de synchronisation.

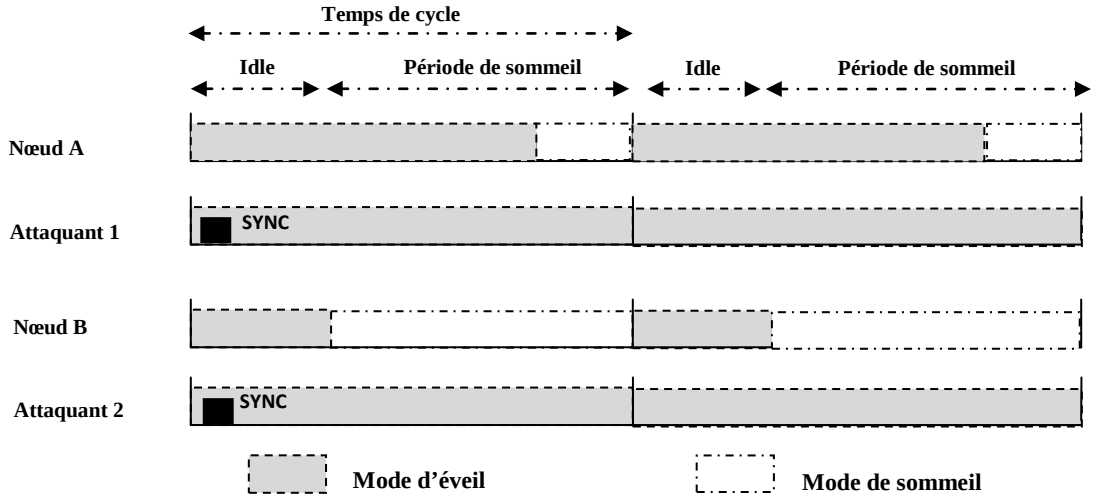



**Figure 7.8 :** Exemple d'attaque de synchronisation.

Dans le cas où le nœud attaquant retransmet un trafic enregistré *(attaque de retransmission)*, il sera difficile de le distinguer de celui du nœud légitime, car ils ont le même identifiant. En combinant la valeur RSSI avec la table de routage *(table TID)*, le mécanisme de sécurité proposé peut détecter et atténuer l'effet des attaques de retransmission.

L'attaque de diffusion consiste à diffuser un long message de données à tous les nœuds qui se situent dans la zone de couverture radio du nœud malveillant. Etant donné qu'il n'y a pas de paquets RTS qui précèdent le message diffusé, le nœud récepteur ne peut pas authentifier la source de ce dernier avant de le recevoir. Par conséquent, les solutions d'authentification et de chiffrement ne peuvent pas empêcher ce genre d'attaques. La solution proposée dans [18] atténue l'attaque de diffusion, mais elle propose de modifier radicalement le protocole de la sous couche Mac. Dans notre mécanisme, le nœud victime ne reçoit que le premier fragment de données et rejette ceux qui restent *(en entrant dans un mode de sommeil)*. En effet, le premier fragment du message de diffusion contient l'identité de l'expéditeur. Ainsi, CLIDS permet de détecter le nœud intrus et rejette ensuite les autres fragments. Par conséquent, l'effet de l'attaque de diffusion est réduit de façon significative car le nœud ciblé reçoit un seul fragment et ne prolonge pas son état de réveil afin de recevoir les fragments restants.

### 9.3 Attaques au niveau de la couche physique

Les attaques au niveau de la couche physique consistent généralement à provoquer des interférences dans le canal de communication. En effet, la source d'interférence peut être assez

puissante pour perturber l'ensemble du réseau. De plus, le nœud malveillant peut lancer des attaques d'interférence stratégiques en ciblant des zones sensibles du réseau *(station de base ou cluster head)*. Notre système de détection peut détecter la zone de brouillage en se basant sur les valeurs RSSIs des paquets échangés dans le réseau. Ainsi, dans le cas où plusieurs perturbations d'RSSIs sont détectées dans le même périmètre, une zone d'interférence sera signalée à la BS. En conséquence, la BS établira le prochain chemin de routage en évitant d'inclure les nœuds appartenant à la zone d'interférence détectée.

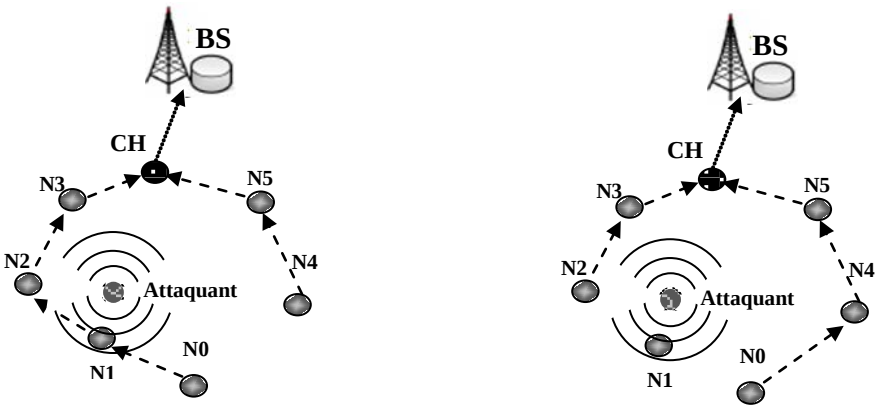



**Figure 7.9 :** Exemple d'attaque de brouillage.

### 9.4 Attaques au niveau de la couche transport

La plupart des attaques au niveau de cette couche sont des attaques de déni de service, dont l'objectif est de perturber le bon fonctionnement du réseau. L'attaque d'inondation et de désynchronisation sont les types les plus connus parmi les attaques qui ciblent la couche transport. CLIDS permet de détecter ces deux attaques l'or de l'établissement de la connexion avec les nœuds victimes *(réception de paquet RTS au niveau de la couche liaison)*. En effet, l'identité du nœud malveillant sera détectée avant qu'il soit autorisé à envoyer ses paquets de données ou de contrôle corrompus. Ainsi, le nœud attaquant ne peut pas épuiser les ressources de connexion des nœuds victimes *(attaque d'inondation)*, ou forcer ces derniers à réparer des erreurs de transmission qui n'ont jamais vraiment existé *(attaque de désynchronisation)*.

## *10. CONCLUSION*

En prenant la sécurité comme principal objectif, nous avons proposé un système de détection d'intrusions dédié pour les réseaux de capteurs sans fil. Dans notre proposition, le problème de détection d'intrusions est abordé avec une nouvelle stratégie dans laquelle l'interaction Cross-layer est pleinement exploitée. Notre approche est de produire un seul système de détection Cross-layer pour la détection d'intrusions au niveau de plusieurs couches du modèle OSI. L'approche proposée ne clame pas d'être immunisée contre tout genre d'attaques malicieuses, mais donne certainement une nouvelle direction aux recherches de sécurité dans les RCSFs. Le chapitre suivant sera consacré à l'évaluation des performances de nos contributions à l'aide d'un ensemble de simulations réalisées avec le simulateur NS 2 *(network simulator)*.

# Chapitre 8

## *Evaluation des performances à travers la simulation*

# 1. INTRODUCTION

Dans ce chapitre, on va évaluer les performances de nos contributions en termes de sécurité et d'économie d'énergie. L'évaluation des performances sera conduite à travers le simulateur réseau NS 2.34 [202], qui est l'un des simulateurs leaders dans le domaine de la simulation réseau. Nous divisons notre étude de performance en deux parties, dont la première est destinée à notre protocole de communication Cross-layer *(CLEOP)*, tandis que la deuxième se focalise sur le système de détection d'intrusions proposé *(CLIDS)*.

# 2. ENVIRONNEMENT DE SIMULATION

L'expérimentation de notre simulation est établie sur 100 nœuds répartis aléatoirement sur une surface carrée de 100 x 100 m² présentée par la figure 8.1.

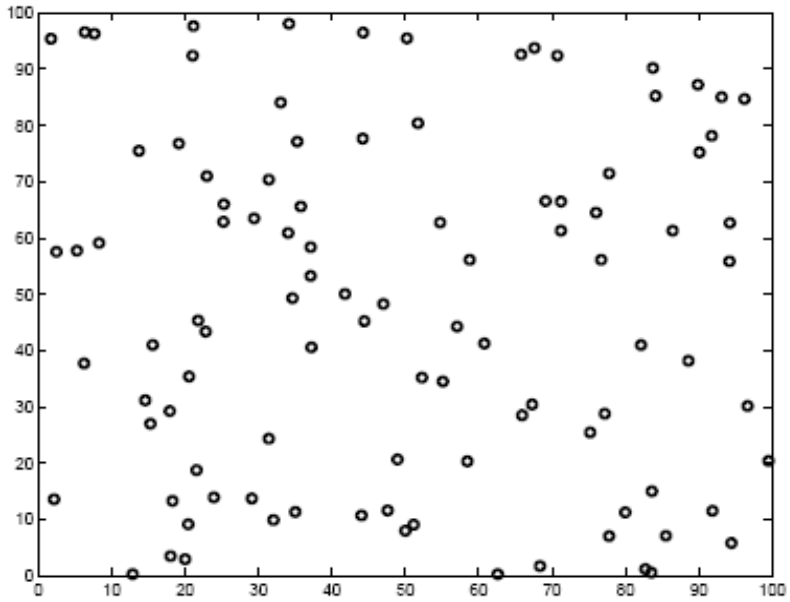



**Figure 8.1:** modèle d'expérimentation

Nous assumons que tous les nœuds ont une position fixe durant toute la période de simulation. De plus, la station de base est positionnée à 75 mètres par rapport au nœud le plus proche (X=50, Y=175). La largeur de bande de transmission est initialisée à 1Mbps, la latence de transmission et de réception d'un paquet de données est égale à 25µs, la taille d'un paquet de données est égale à 500 Bytes, avec une entête de paquet mesurant 25 Bytes. La taille des paquets d'activation est fixée à 144 bites, et la portée des antennes d'activation est initialisée à 2 mètres.

Nous adoptant un modèle simplifié pour la dissipation d'énergie radio, où l'émetteur dissipe de l'énergie durant l'utilisation de son antenne radio pour la transmission des données, et le récepteur consomme également de l'énergie pour recevoir les données via son antenne radio, comme le montre la figure 8.2.

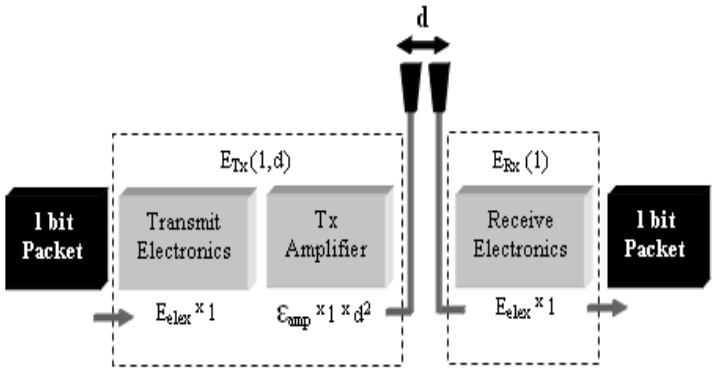



**Figure 8.2 :** modèle de dissipation d'énergie radio proposé

L'énergie des circuits électroniques ($E_{elec}$) dépend de facteurs tels que le codage numérique, la modulation, le filtrage et la propagation du signal, tandis que l'énergie d'amplification ($\varepsilon_{fs}$ pour la transmission directe et $\varepsilon_{mp}$ pour la transmission multi sauts) dépend de la distance vers le récepteur et l'acceptable taux d'erreur binaire.

Pour les simulations décrites dans ce chapitre, les paramètres d'énergie de communication sont fixés comme suit: $E_{elec}$ = 50nJ/bit, εfs = 10pJ/bit/$m^2$, εmp = 0.0013pJ/bit/$m^4$ et l'énergie pour l'agrégation de données est définie comme $E_{DA}$ = 5nJ/bit/ signal. Ainsi, l'énergie nécessaire pour transmettre q bites de donnée sur une distance d, peut être calculée en se basant sur un seuil $d_0$ avec l'équation suivante :

$$E_{tx}(q,d) = \begin{cases} qE_{elect} + q\varepsilon_{fs}d^2 & d<d_0 \\ qE_{elect} + q\varepsilon_{mp}d^4 & d>d_0 \end{cases} \quad (1)$$

Où

$$d_0 = \sqrt{\frac{\varepsilon_{fs}}{\varepsilon_{mp}}} \quad (2)$$

De même, pour recevoir un message de q bits, la radio consomme:

$$E_{Rx}(q) = qE_{elect} \quad (3)$$

Tous les nœuds du réseau commencent la simulation par une énergie initiale égale à 2 J et une quantité de données illimitées à transmettre à la station de base. De plus, l'énergie de la station de base est considérée comme illimitée. Le temps de changement du cluster head, ainsi que la reconstruction de grappe à chaînes et la mise à jour des tables TIDs est fixé à 20 s. Chaque nœud consomme sa réserve d'énergie limitée tout au long de la durée de simulation, ce qui implique l'épuisement de celle-ci. Ainsi, tout nœud qui a épuisé sa réserve d'énergie est considéré comme mort. Par conséquent, il ne peut ni transmettre ni recevoir des données.

Nous assumons qu'il existe dix nœuds attaquants, déployés aléatoirement dans le réseau. Ces derniers passent par une période d'écoute passive *(d'une durée aléatoire)*, et essayent par la suite

de se connecter avec des nœuds victimes choisis d'une manière aléatoire *(attaquant simple)*, ou d'une manière plus spécifique *(attaquant malicieux)*. Ainsi, les nœuds intrus ne vont pas lancer leurs attaques simultanément, mais au fur et à mesure de l'avancement de la simulation. Tous les paramètres de notre simulation sont résumés dans le tableau ci-dessous :

| ***Paramètre*** | ***Valeur*** |
|---|---|
| La surface du réseau | 100 m² |
| La localisation de la BS | (50, 175) |
| Le nombre de nœuds | 100 |
| Le nombre de clusters | 5 |
| L'énergie initiale des nœuds | 2 J |
| Taille du paquet de données | 500 Bytes |
| Taille du paquet d'activation | 144 bits |
| Taille de paquets RTS, CTS et ACK | 30 Bytes |
| $E_{elect}$ | 50nJ/bit |
| Efs | 10nJ/bit/m² |
| Emp | 0.0013pJ/bit/$m^4$ |
| $E_{DA}$ | 5nJ / bit / signal |
| Temps d'exécution de la phase d'initialisation | Chaque 20 s |
| Nombre des nœuds intrus | 10 |

**Tableau 8.1** : Paramètres de simulation

## 3. EVALUATION DES PERFORMANCES DU PROTOCOLE CLEOP

Avant d'évaluer les performances du protocole CLEOP, nous allons d'abord évaluer celles du protocole HEEP, étant donné que ce dernier constitue la base de notre protocole de communication.

### 3.1 Evaluation des performances du protocole de routage HEEP

Notre première étape d'évaluation consiste à analyser la performance énergétique du protocole. Pour cela, nous allons mesurer et comparer la durée de vie du réseau offerte par le protocole HEEP *(HEEP_D et HEEP_S)*, avec celle obtenue en utilisant différents algorithmes de routage existants (LEACH, LEACH-C, PEGASIS). Ainsi, nous avons mesuré l'énergie résiduelle des nœuds capteurs toutes les 10 secondes pendant toute la durée de la simulation, afin de calculer le nombre total des nœuds vivants. De plus, nous avons calculé le pourcentage des nœuds morts *(épuisement de toutes les réserves énergétiques)*, au fur et à mesure de l'avancement de la simulation. Les figures 8.3 et 8.4, présentent les résultats que nous avons obtenus.

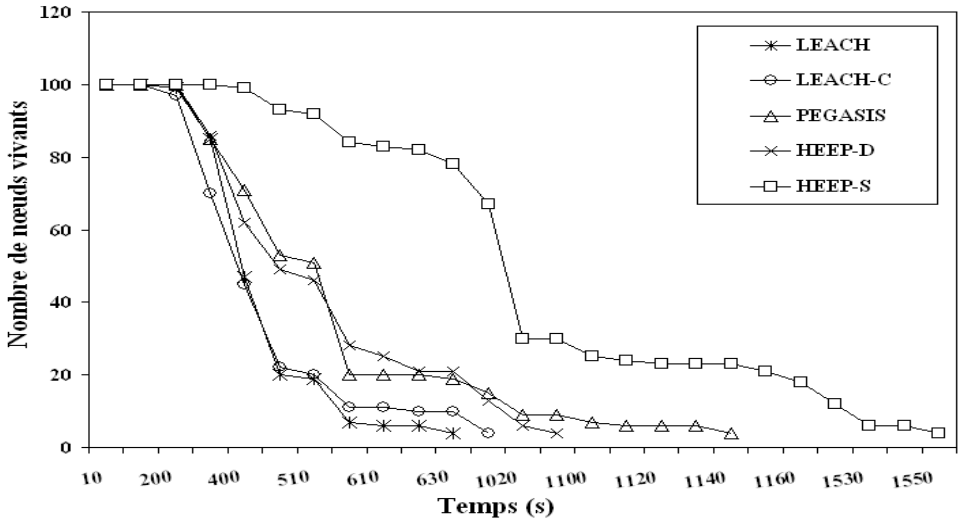



**Figure 8.3** Nombre de nœuds vivants au fil du temps.

En se basant sur les résultats des simulations, nous avons démontré les améliorations apportées par le protocole HEEP, en termes d'économie d'énergie et du temps de vie du réseau. En effet, la version dynamique du protocole HEEP *(HEEP-D)*, prolonge le temps de vie du réseau par 50% à 70% et de 45% à 55%, comparée à la durée de vie obtenu respectivement avec les deux protocoles LEACH et LEACH-C. Nous notons également que la durée de vie du réseau obtenu est très proche de celle fourni avec le protocole PEGASIS (de 7% à 15%).

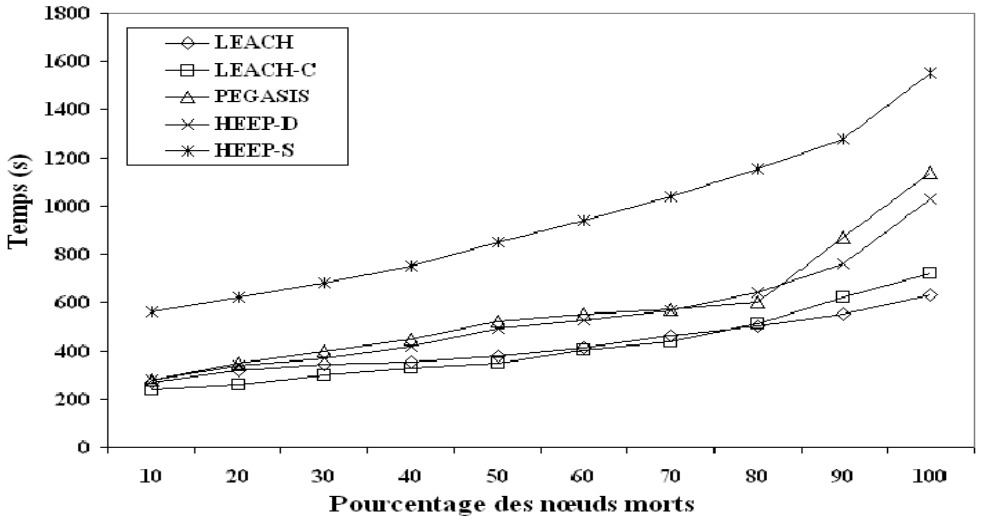



**Figure 8.4 :** Pourcentage des nœuds morts.

En outre, les résultats des simulations montrent que la version statique du protocole HEEP *(HEEP_S)*, offre les meilleurs résultats en termes de gestion d'énergie et augmente la durée de vie du réseau par 48% à 56% comparé au protocole HEEP_D, par 98% à 131% comparé au protocole LEACH, par 93% à 111% comparé au protocole LEACH-C, et par 33% à 49% comparé au protocole de PEGASIS. La deuxième étape d'évaluation, consiste à analyser le taux de parquets de données délivrés à la BS. Les résultats obtenus sont comparés à ceux fournis par les protocoles LEACH, LEACH-C et PEGASIS. La figure 8.5 illustre les résultats obtenus.

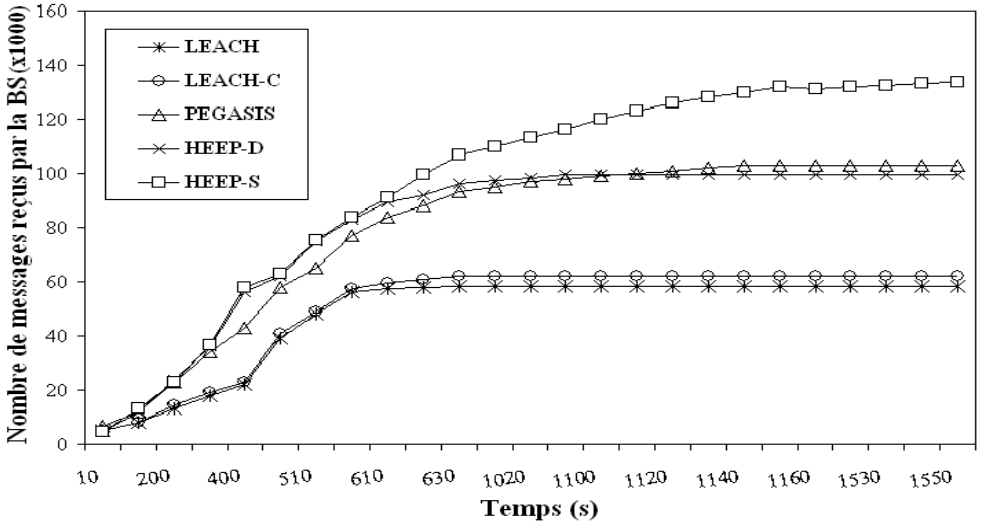



**Figure 8.5 :** Nombre de messages reçus par rapport au temps de vie du réseau

On peut clairement observer l'efficacité du protocole HEEP-S, en termes de taux de réception des messages de données. ce dernier offre des améliorations dans la livraison des paquets de données par des facteurs de 56, 54, 25 et 23 pour cent, respectivement par rapport aux protocoles LEACH, LEACH-C, HEEP-D et PEGASIS. De l'autre côté, le protocole HEEP-D améliore les deux protocoles LEACH et LEACH-C, respectivement par des facteurs de 41 et 38 pour cent. L'étape suivante dans l'évaluation de performances, consiste à mesurer la dissipation énergétique de notre protocole tout au long de la période de simulation. La figure 8.6 résume les résultats obtenus. De toute évidence, notre protocole améliore les deux algorithmes LEACH et LEACH-C, et offre une meilleure dissipation d'énergie. En effet, LEACH et LEACH-C ne garantissent pas une consommation uniforme d'énergie entre les nœuds capteurs. Par conséquent, les nœuds qui sont loin de leur CH vont dissiper une quantité considérable d'énergie comparée à celle dissipée par les nœuds les plus proches.

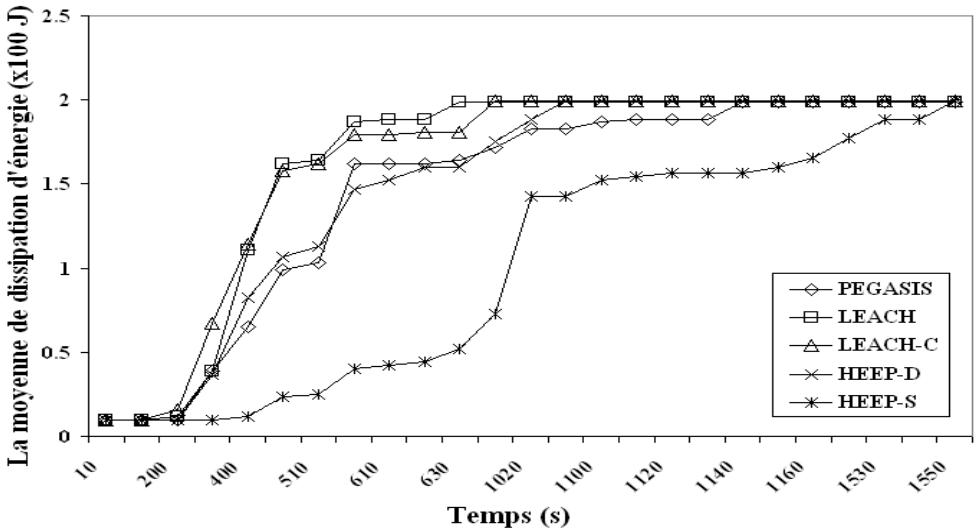



**Figure 8.6 :** Dissipation d'énergie par rapport au temps de vie du réseau

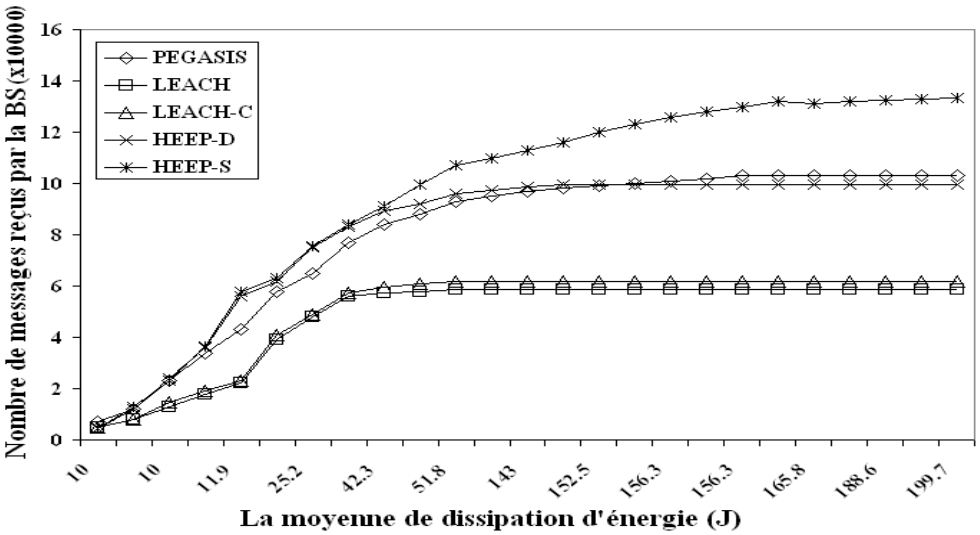



**Figure 8.7 :** Nombre de messages reçus par rapport à la dissipation d'énergie

Le protocole HEEP *(approche statique et dynamique)* permet de résoudre ce problème en réduisant les distances de transmission entre les nœuds et leurs CHs. De plus, l'agrégation des données au niveau des chaînes de transmission, réduit la quantité des données transmises aux CHs, et permet de répartir uniformément la charge entre les nœuds du cluster. Nous remarquons aussi dans la figure 8.6, que la version statique du protocole HEEP améliore significativement la dissipation d'énergie obtenue avec le protocole PEGASIS. HEEP-S réduit donc la consommation moyenne d'énergie par 24%, comparée à celle obtenue avec PEGISIS. En effet, l'amélioration de la dissipation énergétique permet d'augmenter le taux de messages délivrés à la BS. Cela est confirmé par la figure 8.7, qui démontre les performances de notre protocole en termes de taux de messages délivrés par unité d'énergie.

Afin d'évaluer la répartition équilibrée de la consommation d'énergie dans le réseau, nous avons mesuré le niveau de consommation énergétique au niveau de chaque nœud capteur, après 450 secondes de simulation. Les résultats obtenus sont représentés par la figure 8.8 avec le protocole LEACH, et la figure 8.9 avec le protocole HEEP_D. On peut clairement observer que la dissipation d'énergie est plus importante avec le protocole LEACH. De plus, nous constatons que la consommation d'énergie n'est pas répartie uniformément entre les nœuds du cluster. Par contre, la dissipation d'énergie est moins importante et distribuée équitablement tout au long des clusters, dans le cas où on utilise protocole HEEP-D.

La dernière étape dans l'évaluation des performances du protocole HEEP, est l'analyse du délai de bout en bout *(latence)* qui peut être introduit par ce dernier. Pour évaluer ce niveau de latence *(temps moyen consacré à envoyer un paquet avec les données agrégées à la BS)*, nous avons mené plusieurs simulations tout en augmentant le nombre de nœuds dans le réseau. Les résultats obtenus sont représentés par la figure 8.10. Les résultats des simulations montrent que le protocole HEEP_S offre les meilleurs résultats en termes de la latence de livraison. En revanche, la latence introduite par le protocole HEEP_D est inférieure à celle introduite par le protocole PEGASIS *(longue chaine de transmission)*, et reste très proche de celle fournie par le protocole LEACH.

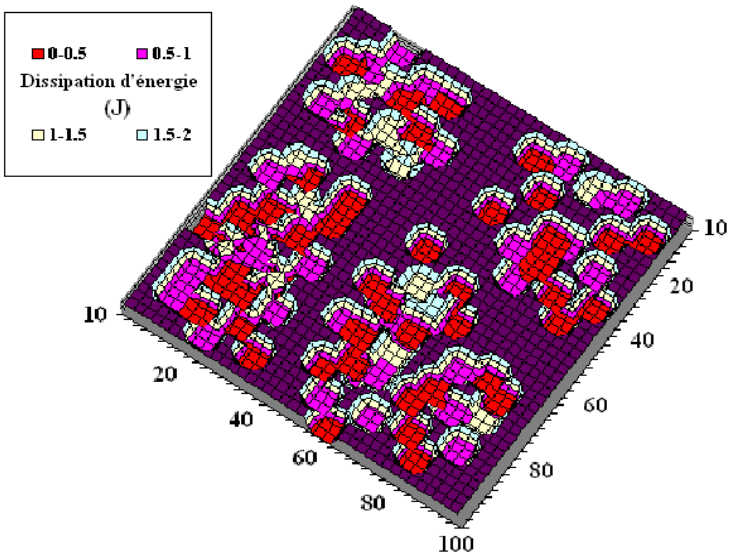



**Figure 8.8 :** Répartition de la dissipation d'énergie avec le protocole LEACH

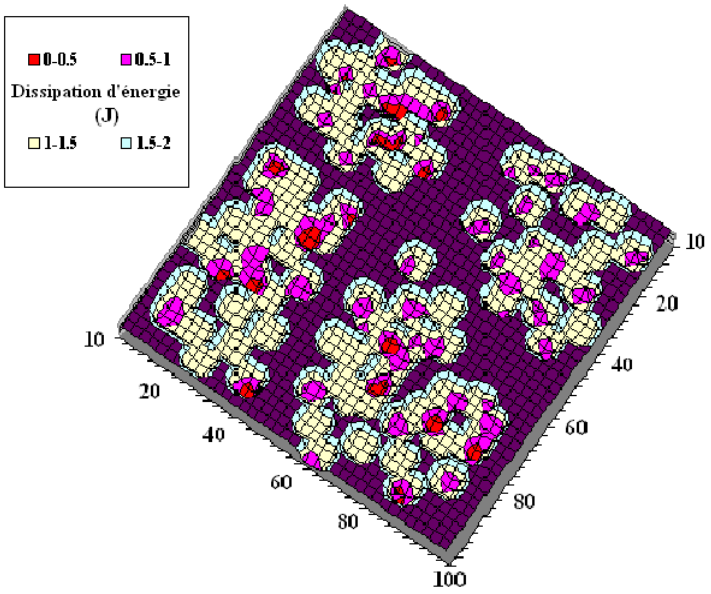



**Figure 8.9 :** Répartition de la dissipation d'énergie avec le protocole HEEP-D

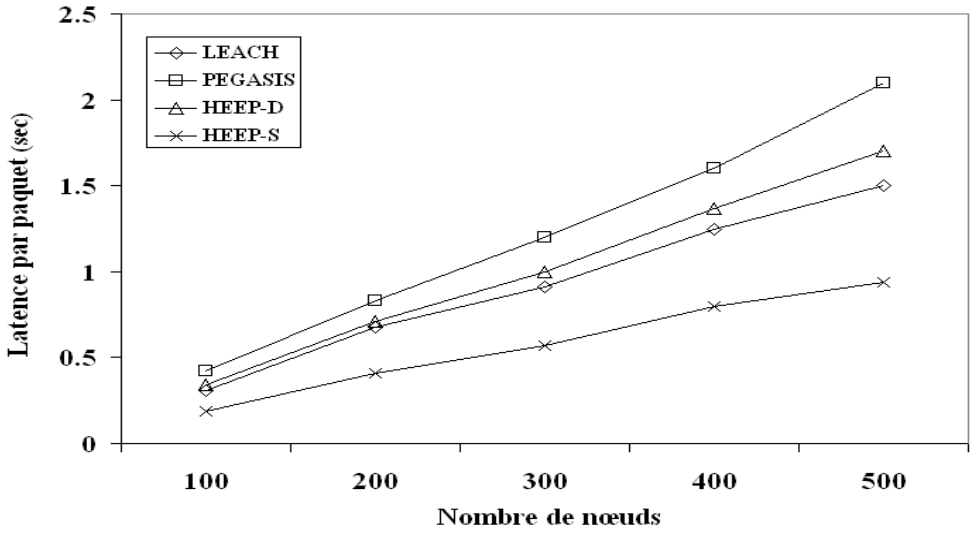



**Figure 8.10 :** Moyenne de latence introduite par rapport au nombre des nœuds

### *3.1.1 Discussion sur les résultats de simulation du protocole HEEP*

Les résultats des simulations confirment que notre protocole apporte des améliorations aux deux protocoles LEACH et PEGASIS. En effet le protocole HEEP *(statique et dynamique)* permet de doubler la durée de vie du réseau par rapport au protocole LEACH, augmente la quantité de données envoyées à la BS, et régule la dissipation d'énergie au sein du cluster. Comparé au protocole PEGASIS, notre protocole donne de meilleurs résultats, étant donné que le temps de latence est réduit à la moitié. Toutefois, les améliorations apportées par notre protocole ne sont pas sans contraintes, la transmission des données à travers des chaines de nœuds voisins augmente la latence de transmission par rapport au protocole LEACH. La version statique de HEEP résout ce problème, mains en revanche elle réduit la flexibilité du réseau *(ajout ou suppression des nœuds capteurs)*.

## 3.2 Evaluation des performances du protocole CLEOP

L'évaluation des performances du protocole CLEOP *(CLEOP-D et CLEOP-S basés respectivement sur les protocoles HEEP-D et HEEP-S)*, est effectuée en utilisant plusieurs métriques de performances, tel que : la durée de vie du réseau, la dissipation d'énergie, le taux de messages délivrés à la BS, le degré de latence introduit et le taux des collisions entre les paquets de données. Les résultats obtenus seront comparés à ceux fournis par d'autres protocoles du domaine, à savoir : S-MAC *(couplé au protocole de routage HEEP-D)*, MAC-CROSS et AREA-MAC.

### *3.2.1 Évaluation de la durée de vie du réseau*

La durée de vie du réseau constitue la métrique la plus importante dans l'évaluation des performances des protocoles de communication dédiés aux RCSFs. Celle-ci est en relation directe avec le nombre de nœuds vivants dans le réseau. Dans notre simulation, nous avons mesuré le nombre de nœuds vivants dans le réseau, tout au long de la période de simulation. La figure 8.11, illustre les résultats que nous avons obtenus.

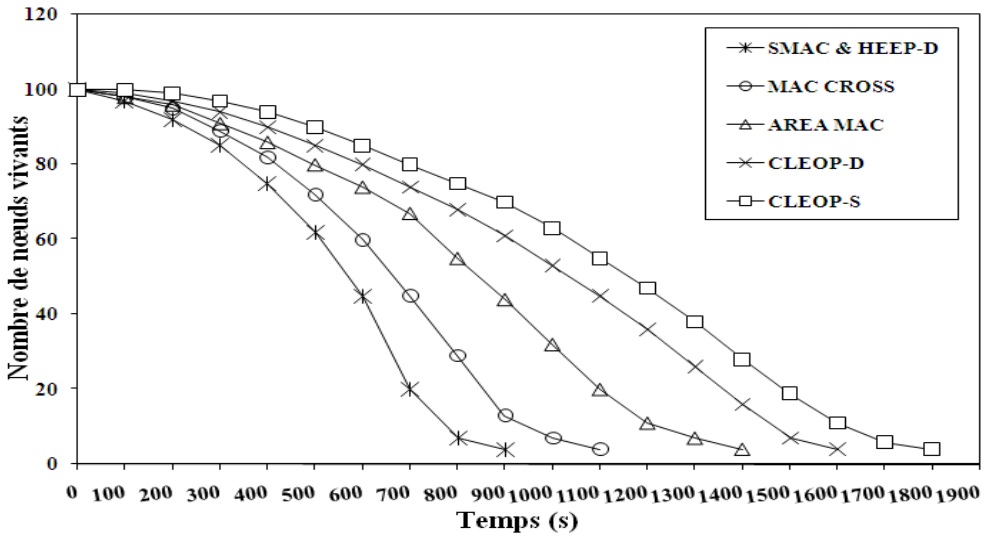



**Figure 8.11 :** Nombre de nœuds vivants dans le réseau.

Nous pouvons observer que le nombre de nœuds morts augmente progressivement avec l'avancement du temps de simulation. Cependant, cette augmentation est plus lente dans notre protocole *(CLEOP-D et S-CLEOP)* par rapport aux protocoles SMAC, MAC-CROSS et AREA-MAC. Cela s'explique par la mise en état de sommeil de tous les nœuds inutiles *(les nœuds qui ne participent pas à l'activité de routage)*, ce qui permet de ne pas gaspiller les réserves d'énergie et optimise la durée de vie du réseau. En outre, l'utilisation d'une antenne d'activation élimine le problème d'activation abusive et minimise l'énergie dépensée durant les périodes d'écoute passive. L'organisation à base de cluster à chaines améliore aussi la durée de vie du réseau, étant donné que la dissipation d'énergie et les distances de transmission sont considérablement réduites.

### *3.2.2 Évaluation de la dissipation d'énergie*

La dissipation d'énergie est relativement liée à la bonne gestion des réserves d'énergie. La plupart des protocoles de communication dédiés aux RCSFs, doivent offrir une bonne gestion d'énergie afin de prolonger la durée de vie du réseau. Dans la simulation suivante, nous avons mesuré le total d'énergie consommée par les nœuds capteurs durant toute la période simulation, ce qui nous a donné comme résultat la figure suivante :

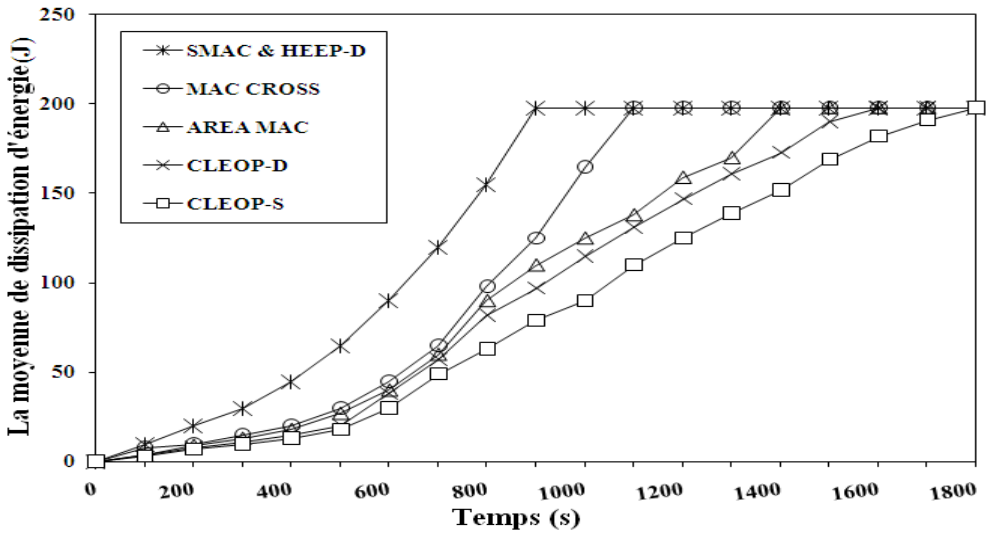



**Figure 8.12 :** Dissipation d'énergie par rapport au temps

La Figure 8.12 montre que le protocole CLEOP *(statique et dynamique)* offre les meilleurs résultats en termes d'efficacité énergétique. Comparé aux protocoles SMAC, MAC-CROSS et AREA-MAC, notre protocole peut atteindre une meilleure dissipation d'énergie, étant donné que la majorité des problèmes de gaspillage d'énergie sont résolus *(activation inutile ou abusive, écoute passive…)*. En effet, l'utilisation d'une deuxième antenne radio permet de réduire aussi les problèmes d'interférences, de collisions et de retransmission des paquets, ce qui optimise la consommation d'énergie. De plus, le contrôle dynamique et Cross-layer de puissances de transmission radio, préserve plus d'énergie et améliore la durée de vie du réseau.

Afin de confirmer les résultats obtenus en termes de dissipation d'énergie, nous avons mesuré le nombre de nœuds activés après 300 secondes du temps de la simulation. La figure 8.13 illustre le résultat obtenu avec le protocole SMAC (combiné au protocole HEEP-D), tandis que la figure

8.14 présente le résultat obtenu avec le protocole CLEOP-D. Nous pouvons clairement constater que le nombre de nœuds activés avec le protocole SMAC est beaucoup plus élevé que celui obtenu avec le protocole CLEOP-D. Cela est dû à l'utilisation des cycles d'activation/sommeil fixés par le protocole SMAC. Ainsi, un nœud doit être réveillé lorsque sa période de sommeil arrive à expiration même si le nœud n'a pas d'activité de routage, entraînant ainsi une consommation inutile d'énergie. En revanche, notre protocole met dans un état de sommeil prolongé tous les nœuds qui ne sont pas inclus dans le processus de communication, ce qui permet de réduire la quantité de nœuds activés inutilement et préserve donc les réserves d'énergie.

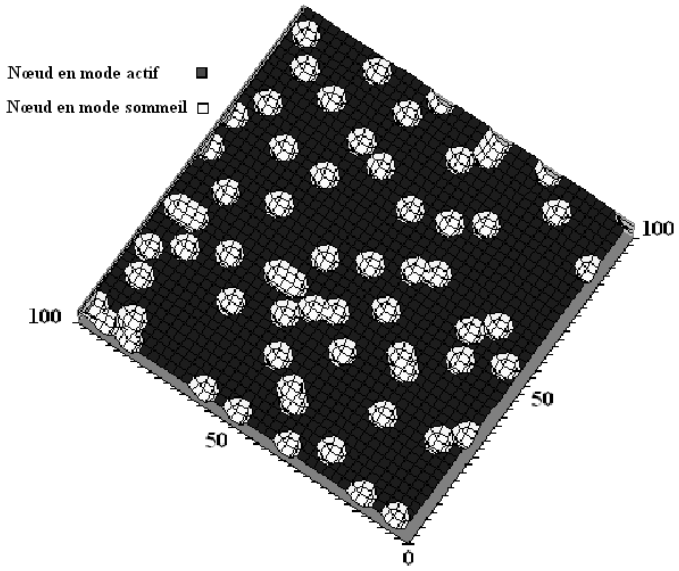



**Figure 8.13 :** Schémas des nœuds activés avec le protocole SMAC

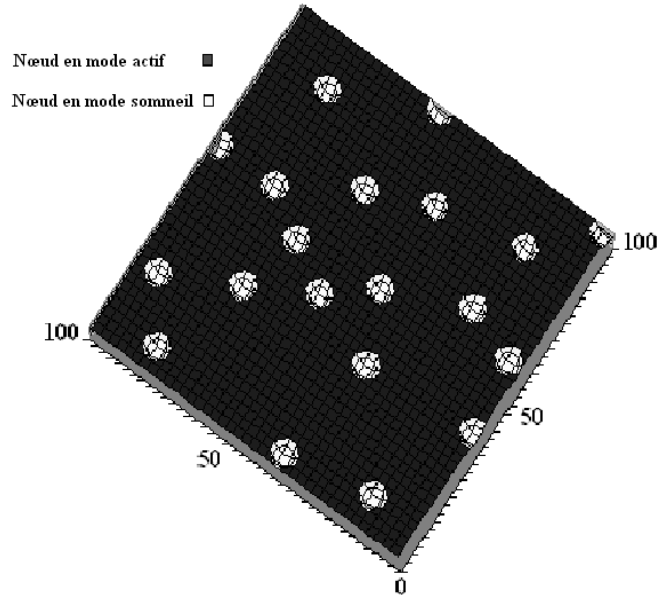



**Figure 8.14 :** Schémas des nœuds activés avec le protocole CLEOP-D

### *3.2.3 Evaluation du taux de paquets délivrés à la BS*

Le taux des paquets reçus par la station de base représente une autre métrique importante dans l'évaluation des performances de notre protocole. Ce taux est directement lié à la durée de vie du réseau et au nombre de nœuds vivants dans le réseau. Les figures 8.15 et 8.16 présentent nos résultats de simulation en termes de nombre de paquets reçus par la BS.

Les résultats de simulation prouvent que notre protocole *(statique et dynamique)* est capable d'offrir des taux optimaux en termes de délivrance de paquets. Cela est principalement justifié par la gestion efficace des réserves d'énergie *(les nœuds capteurs vivent plus longtemps et transmettent plus de paquets)*. De plus, l'ajustement dynamique des portées des antennes radio permet d'offrir une meilleure qualité de liens de transmission, et réduit les problèmes d'interférences et de collision de paquets. La désactivation des nœuds non concernés par le processus de routage permet aussi d'optimiser le nombre de messages transmis vers la BS, étant donné que ces derniers ne vont pas interférer dans la communication des autres nœuds *(qui participent à la transmission des données)*.

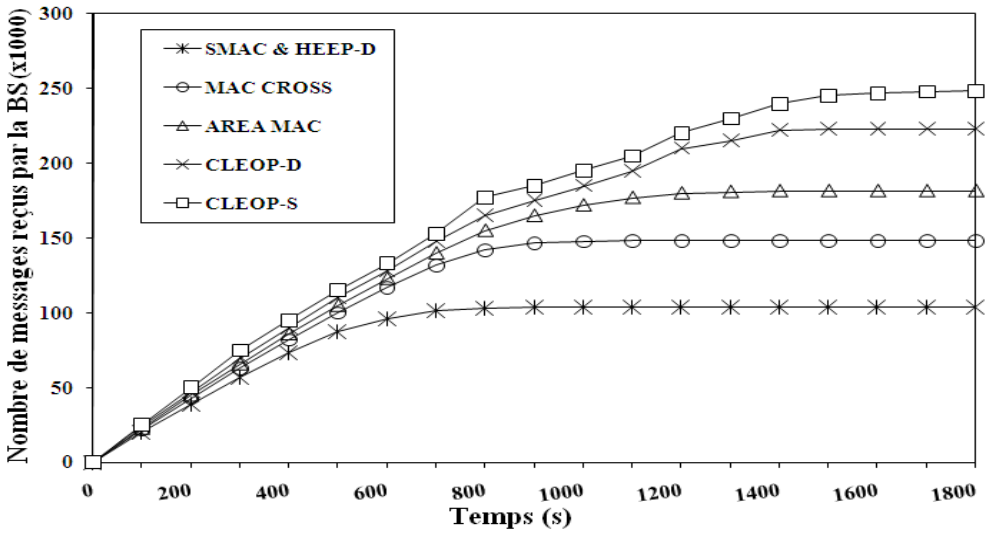



**Figure. 8.15 :** Nombre de parquets reçus par la BS par rapport au temps.

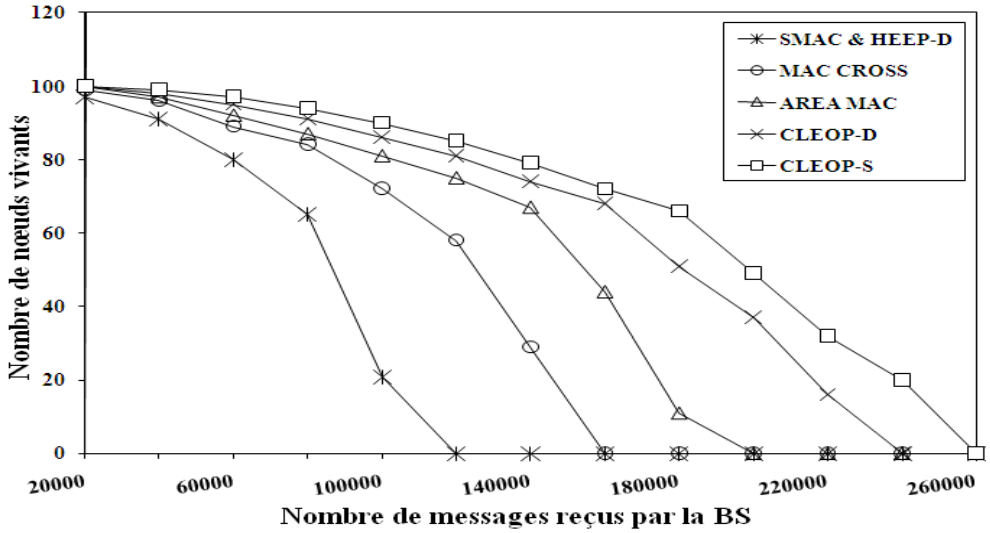



**Fig. 8.16 :** Nombre de nœuds vivants par rapport aux parquets reçus par la BS

Pour montrer l'importance du lien entre l'économie d'énergie et le taux de délivrance des données, nous avons mesuré la moyenne de dissipation d'énergie par rapport au nombre de messages reçus par la station de base. La figure suivante montre que notre protocole offre le meilleur rapport entre la dissipation d'énergie et le nombre de paquets délivrés.

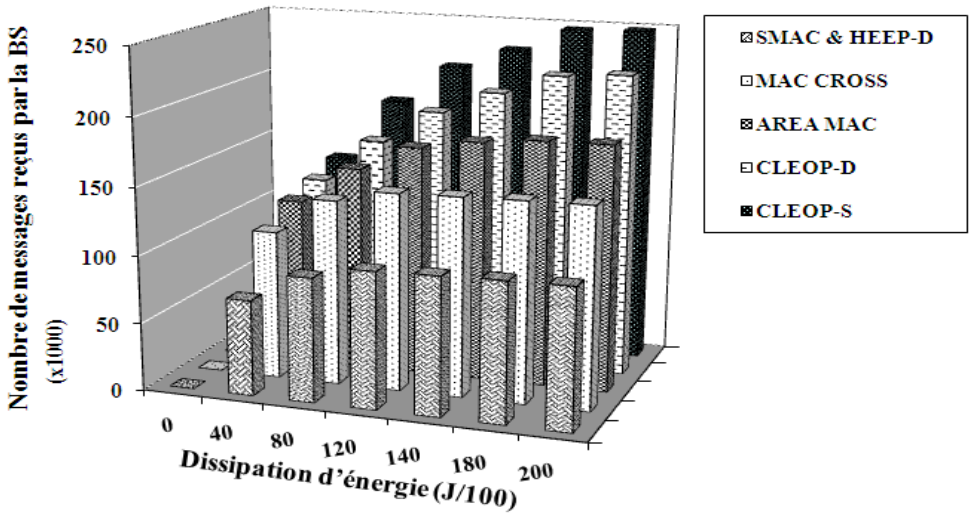



**Figure 8.17 :** Nombre de parquets reçus par la BS par rapport à la dissipation d'énergie.

### *3.2.4 Évaluation du degré de latence introduit*

L'étape suivante dans notre étude de performance consiste à mesurer le degré de latence (*le temps nécessaire pour transmettre un paquet de données à la BS)* introduit par notre protocole. Ainsi, nous avons conduit plusieurs simulations, où dans chacune d'elles le nombre de nœuds dans le réseau est augmenté et le degré de latence introduit par les différents protocoles est mesuré. Les résultats obtenus sont représentés par la figure 8.18.

Nous pouvons observer que le degré de latence est proportionnel au nombre de nœuds dans le réseau. En effet, le protocole S-MAC introduit le plus grand degré de latence, étant donné que les nœuds ne peuvent pas envoyer ou recevoir des données avant l'expiration de leur période de sommeil. Comparé aux autres protocoles de communication, la latence introduite par notre protocole est très minime. Cela est justifié par l'utilisation d'antennes d'activation radio, qui permettent aux nœuds capteurs d'être prêts à tout événement de transmission ou de réception de données.

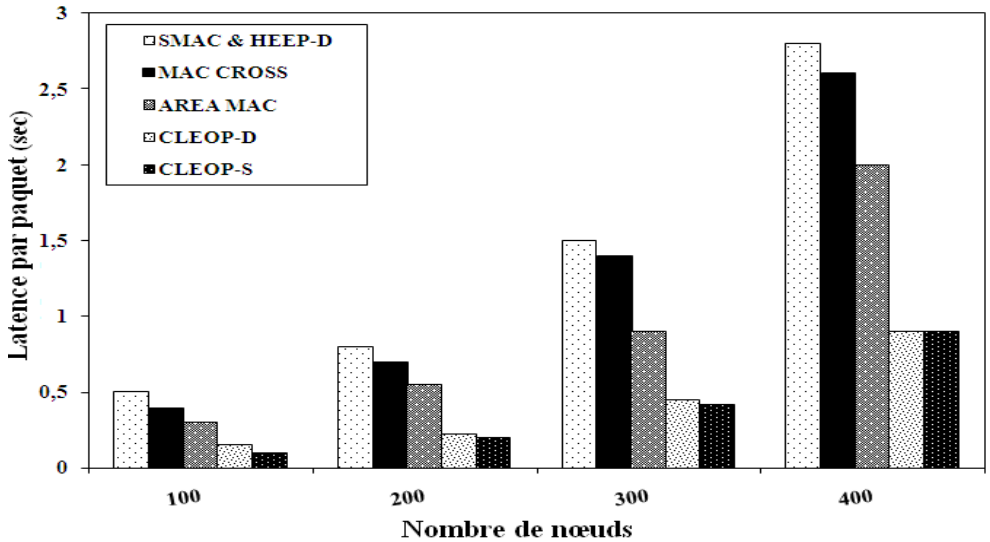



**Figure 8.18 :** La moyenne de latence par rapport au nombre de nœuds dans le réseau.

### *4.2.5 Évaluation du taux de collisions de paquets*

La dernière étape de notre évaluation des performances se focalise sur l'analyse du taux de collisions des paquets générés par notre protocole de communication. Ainsi, nous avons analysé le nombre de paquets en collision par rapport à la densité des nœuds dans le réseau.

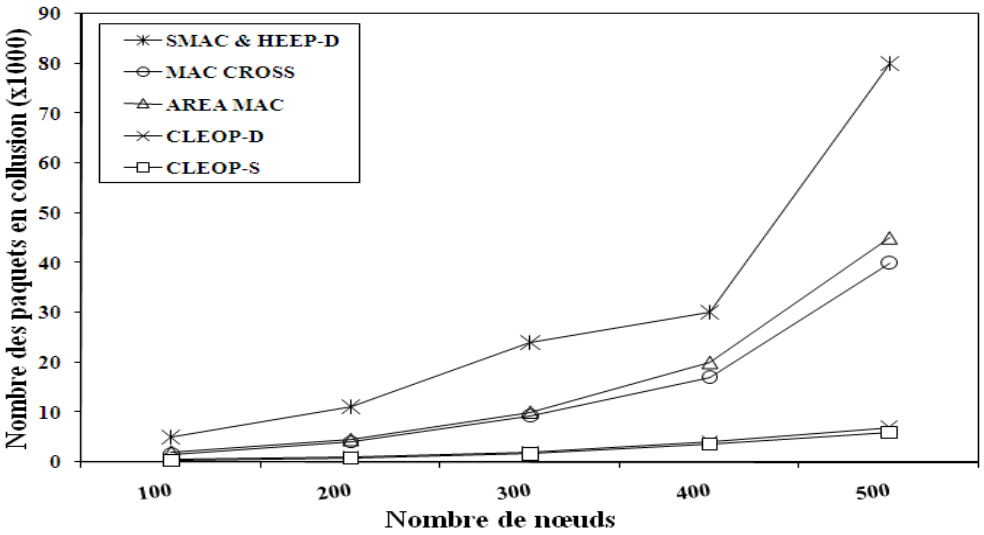



**Figure 8.19** La moyenne des paquets en collision par rapport au nombre de nœuds

Basé sur Figure 8.19, nous avons démontré que notre protocole (CLEOP-D et CLEOP-S) permet de réduire considérablement le nombre de paquets en collision. Cela s'explique par l'ajustement dynamique de la portée des antennes radio et l'utilisation des antennes d'activation qui permettent de minimiser les problèmes d'interférence entre les nœuds du réseau.

## 4. EVALUATION DES PERFORMANCES DU PROTOCOLE CLIDS

La deuxième partie de notre étude expérimentale se focalise sur l'analyse des performances du système de détection d'intrusions Cross-layer CLIDS. Cette analyse consiste à évaluer ses capacités de détection d'intrusions au niveau de différentes couches du modèle OSI. Ainsi, nous avons défini plusieurs métriques pour l'évaluation des performances du protocole CLIDS, tel que : le nombre de nœuds intrus détectés, le taux de paquets délivrés à la BS, la durée de vie du réseau, la dissipation d'énergie, le nombre de paquets de données corrompus et le taux de consommation d'énergie de notre SDI.

### 4.1 Capacité de détection d'intrusions au niveau de la couche réseau

La couche réseau constitue la couche la plus ciblée par les attaques malicieuses dans les RCSFs. Par conséquent, l'évaluation des performances de notre SDI contre les attaques qui ciblent cette couche *(attaque de routage)*, est une étape très importante. Dans notre étude expérimentale, nous avons choisi d'analyser le comportement de notre SDI face aux attaques de routage les plus connues, tel que l'attaque de trou noir, de routage sélectif, de trou de puits, de falsification et d'attaque Sybil. Afin d'approfondir le niveau d'évaluation, nous assumons l'existence de deux types d'attaquants dans le réseau. Le premier type lance des attaques simples qui ciblent

aléatoirement les nœuds dans le réseau, tandis que le deuxième type sont plus malicieux et essayent d'être sélectionnés comme cluster head avant de lancer leurs attaques.

### *4.1.1 Évaluation des performances contre les attaques de trou puits*

L'attaque de trou de puits est l'une des attaques les plus dangereuses, qui peuvent cibler le processus de routage des données. Cette attaque peut être la base pour le lancement d'autres attaques, tel que le trou noir, le routage sélectif et la corruption des données. Dans notre simulation, nous avons proposé d'étudier le comportement de notre SDI contre les attaques de trou de puits simples et malicieuses. Dans les deux types d'attaques, les nœuds intrus essayent de s'insérer dans les chemins de routage afin de pouvoir créer des trous de puits. Ainsi, nous avons mesuré le nombre des trous de puits créés dans le réseau, tout au long de l'avancement de la simulation. Les figures 8.20 et 8.21 présentent les résultats obtenus.

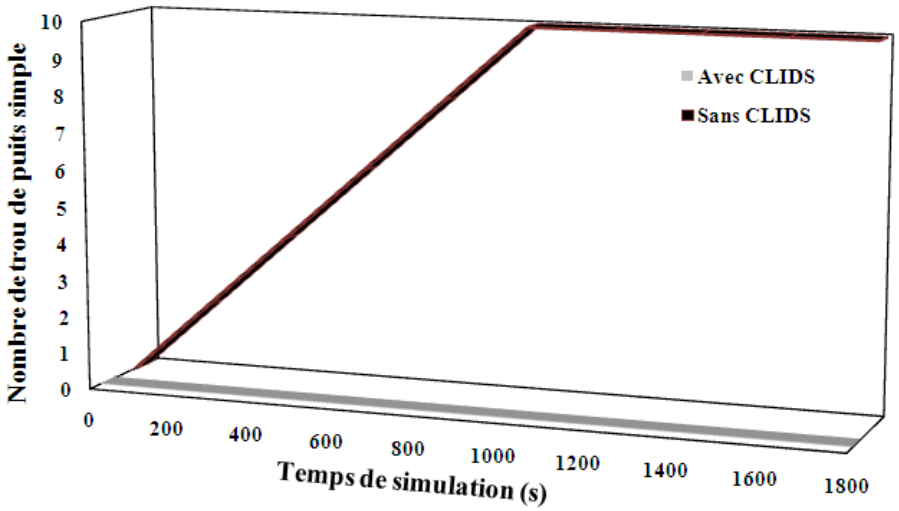



**Figure 8.20 :** Nombre des trous de puits simples dans le réseau

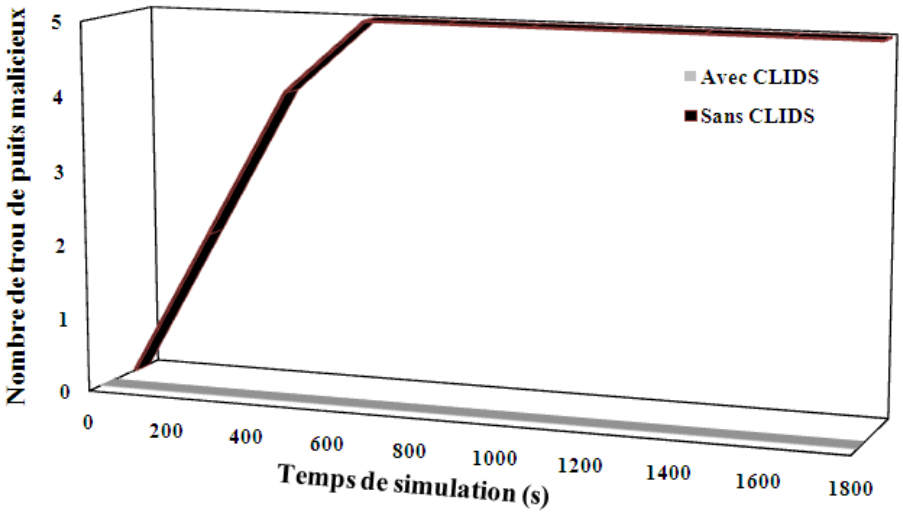



**Figure 8.21** Nombre des trous de puits malicieux dans le réseau

On peut constater qu'aucun trou de puits *(simple et malicieux)* n'a été créé avec notre système de détection CLIDS. Cela est justifié par la détection et la rejection de ces derniers au niveau de la

BS lors de l'établissement des chemins de routage et l'élection des CHs. Par contre, dans le cas non sécurisé *(sans CLIDS)*, tous les CHs élus seront des nœuds intrus *(trou de puits malicieux)* après 800 secondes de la simulation, car il y a plus de 5 nœuds intrus activés *(sortis de leur période d'écoute passive)* dans le réseau.

### *4.1.2 Évaluation des performances contre les attaques de trou noir et de routage sélectif*

Afin d'analyser le comportement de CLIDS contre les attaques de trou noir et de routage sélectif, nous avons mesuré en premier temps le nombre de paquets reçus par la BS. Cela nous permet d'évaluer le taux de paquets rejetés par les attaques en question. Les résultats de notre simulation sont illustrés par les figures suivantes.

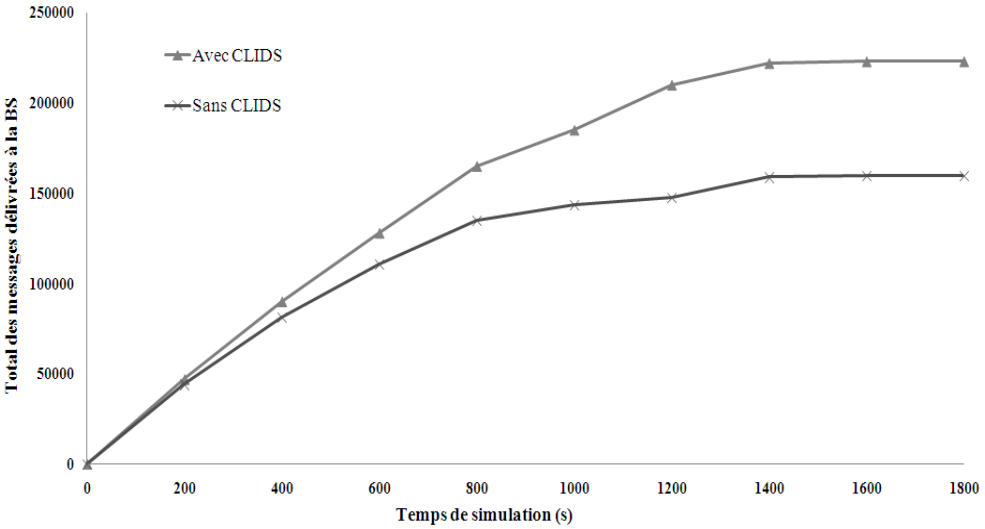



**Figure 8.22** Nombre de paquets délivrés à la BS sous l'attaque de trou noir simple

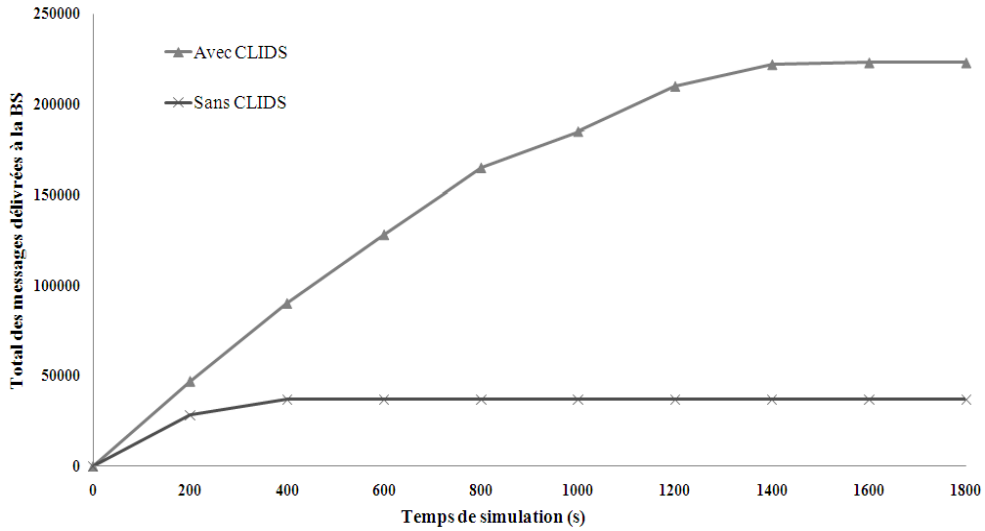



**Figure 8.23** Nombre de paquets délivrés à la BS sous l'attaque de trou noir malicieuse

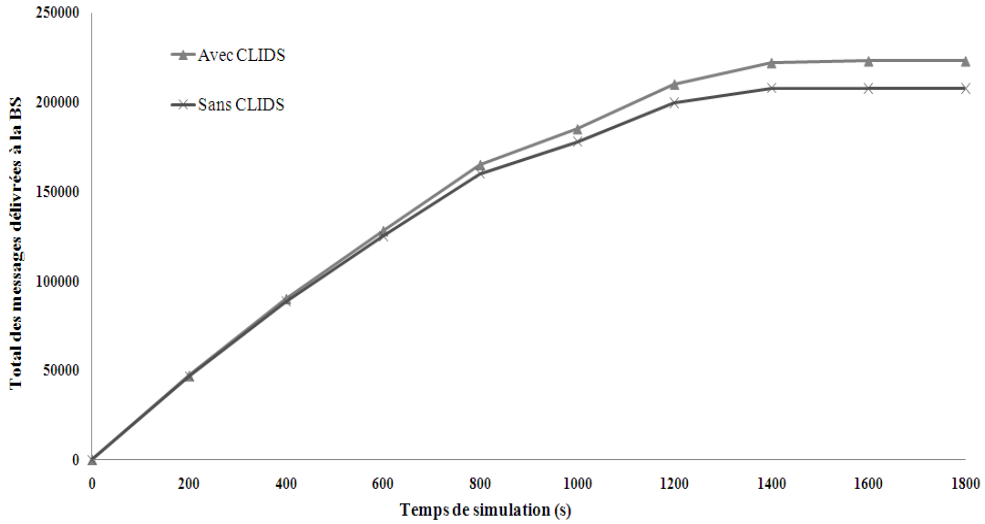



**Figure 8.24** Nombre de paquets délivrés à la BS sous l'attaque de routage sélectif simple

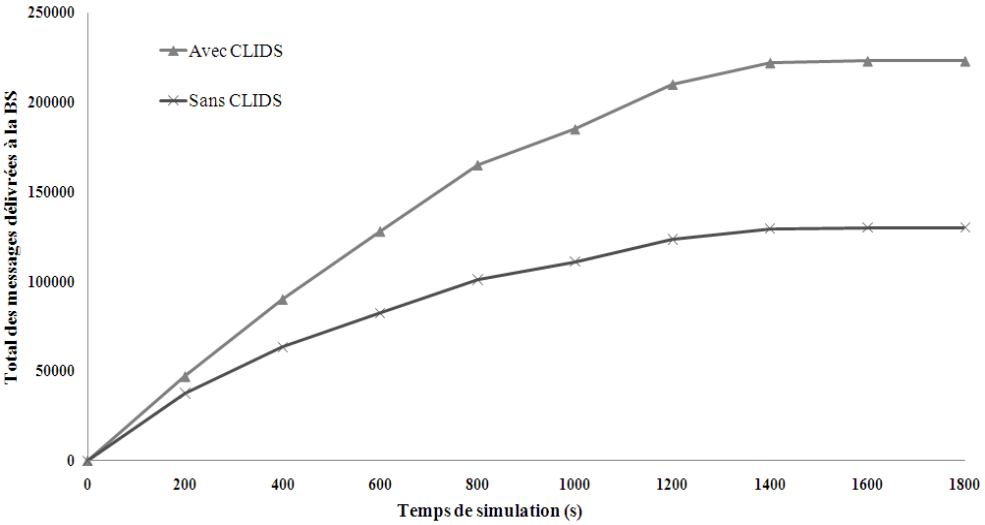



**Figure 8.25** Nombre de paquets délivrés à la BS sous l'attaque de routage sélectif malicieuse

Les résultats de simulation montrent que notre SDI est capable de contrer les attaques de trou noir et de routage sélectif, sous leurs deux formes *(simple et malicieuse)*. En effet, dans notre protocole la sélection des CHs et l'établissement des chemins de routage se font d'une manière centralisée. Par conséquent, les attaques de trou noir et de routage sélectif peuvent êtres facilement détectées. On peut constater dans la figure 23 *(dans le cas non sécurisé)*, qu'aucun message ne sera délivré après 400 secondes de la simulation. Cela est justifié par la réélection répétitive de 5 nœuds intrus comme CH, dont le rôle est de rejeter tous les paquets du cluster *(trou noir)*. Par contre, avec notre SDI, les nœuds intrus seront détectés est rejetés dès qu'ils tentent de s'insérer dans le chemin de routage.

Dans la simulation suivante, nous avons évalué le taux de consommation d'énergie de notre protocole pour la détection des attaques de trou noir et de routage sélectif.

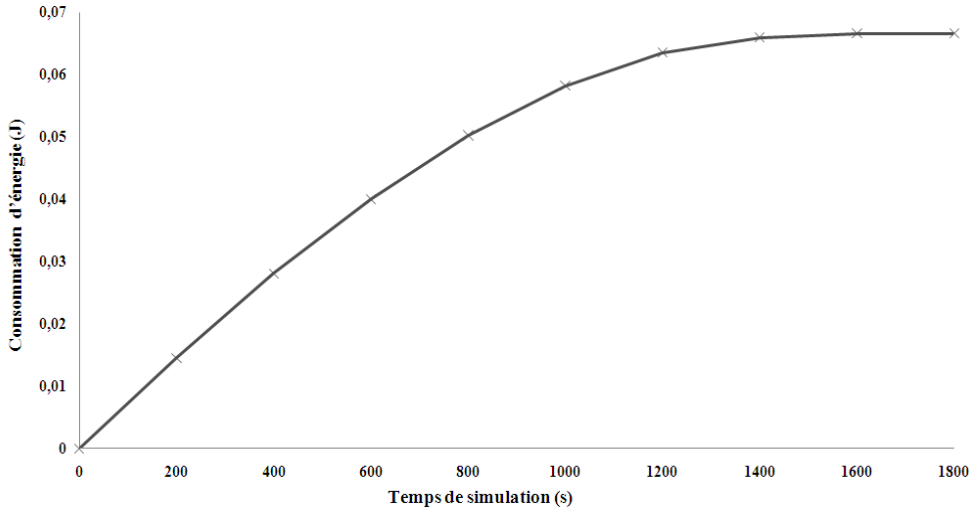



**Figure 8.26 :** Consommation d'énergie du protocole CLIDS par rapport au temps

Comme présenté dans la figure 26, la consommation d'énergie de notre SDI est très minime et ne dépasse pas 0.066 Joule. En effet, les attaques de trou noir et de routage sélectif sont détectées de façon centralisée par la BS, ce qui n'impose pas l'échange de messages d'alarme entre les nœuds capteurs et la BS *(opération coûteuse en énergie)*. Ainsi, notre SDI consomme uniquement 0.03% de l'énergie disponible dans le réseau (200 J) afin de détecter dix nœuds intrus. Cette énergie est consommée durant l'exécution périodique de notre algorithme de détection d'intrusions au niveau de tous les nœuds du réseau. On peut constater aussi que la consommation d'énergie va diminuer au fur et à mesure de l'avancement de la simulation, étant donné que celle ci est proportionnelle au nombre de nœuds vivants et de nœuds intrus *(non détectés)* dans le réseau.

### *4.1.3 Évaluation des performances contre les attaques d'informations fabriquées*

L'attaque d'informations fabriquées consiste à injecter de fausses informations dans le réseau. Dans notre simulation, les nœuds intrus ciblent aléatoirement les nœuds du réseau afin de transmettre des données erronées à la BS. Ainsi, nous avons mesuré le nombre d'informations corrompues reçues par la BS *(avec et sans CLIDS)* et le taux de consommation d'énergie de notre SDI par rapport au temps de la simulation, ce qui nous a donné comme résultat les figures 8.27 et 8.28.

On peut constater qu'aucune fausse information n'a été transmise vers la BS dans le cas d'utilisation du protocole CLIDS. En effet, chaque nœud capteur vérifie l'identité du nœud émetteur avant d'accepter de recevoir ses paquets de données. Par conséquent, tous les messages fabriqués sont aussitôt détectés et rejeté, ce qui permet de ne pas les transmettre par la suite à la BS.

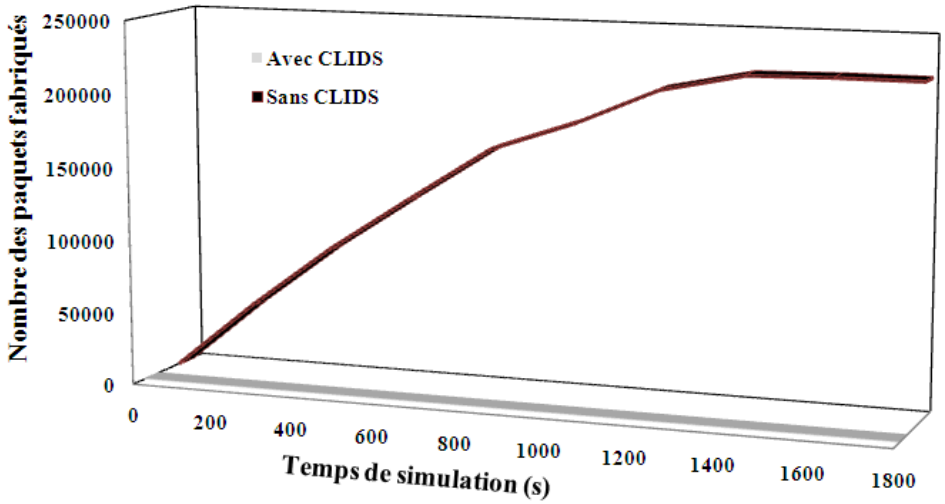



**Figure 8.27** Nombre de paquets fabriqués reçus par la BS

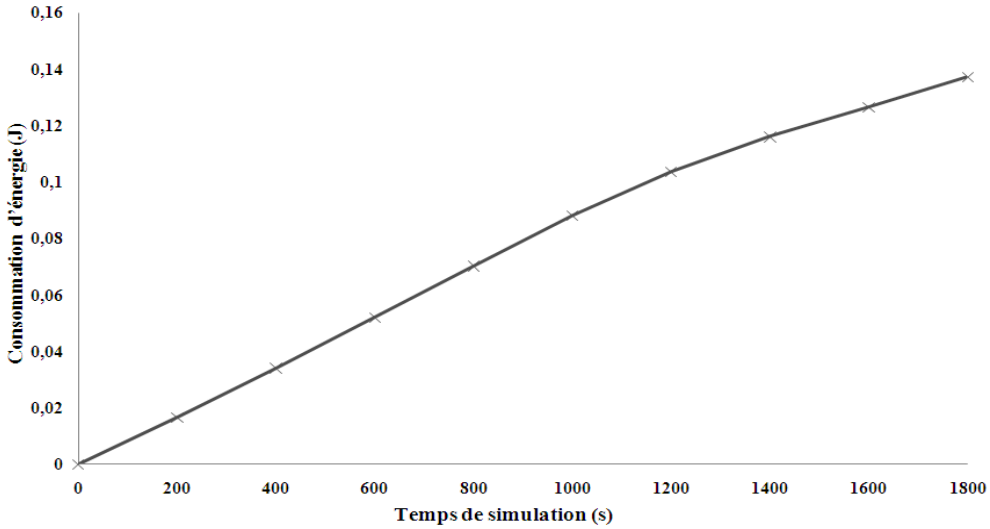



**Figure 8.28 :** Consommation d'énergie pour détecter l'attaque d'informations fabriquées

Afin de détecter les attaques d'information fabriquées *(10 nœuds intrus)*, notre SDI va consommer moins de 0.14 J, qui représente 0.07% de l'ensemble d'énergie disponible dans le réseau. Ce taux de consommation est très réduit voir négligeable comparé à celui qui peut être introduit par les autres solutions *(authentification à base de cryptographie)*. Cependant ce taux de consommation est un peu supérieur à celui qui a été mesuré durant la détection des attaques de trou noir et de routage sélectif (0.03%). Cela est justifié par la transmission des alertes de détection d'intrusions entre les nœuds du réseau la station de base.

### *4.1.4 Évaluation des performances contre les attaques Sybils*

L'attaque par identité multiple ou Sybil est l'une des attaques les plus difficiles à détecter, étant donné que celle-ci peut emprunter les identités des nœuds légitimes dans le réseau. Cette attaque peut être utilisée pour lancer d'autres types d'attaques tels que le trou de puits ou le trou noir. Dans notre simulation, nous assumons que les nœuds intrus présentent l'identité de plusieurs nœuds légitimes dans le réseau afin d'êtres élus comme CH. Après, nous avons mesuré le nombre

de CHs malicieux par rapport au temps et au nombre de nœuds intrus dans le réseau. Les figures 8.29 et 8.30 illustrent les résultats obtenus.

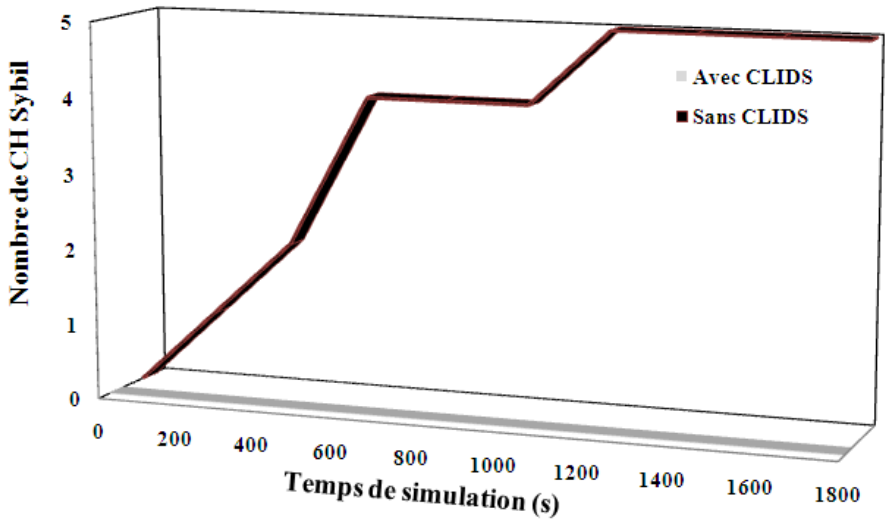



**Figure 8.29 :** Nombre de CHs malicieux par rapport au temps

On voit bien que notre SDI ne permet pas la sélection d'un nœud malicieux comme CH, même si ce dernier peut présenter plusieurs identités légitimes. En effet, la station de base peut détecter la vraie identité des nœuds intrus, étant donné que ces derniers possèdent des valeurs d'RSSIs non conformes à celles des nœuds légitimes. Par contre, dans le cas non sécurisé *(sans CLIDS)* le nombre de nœuds CHs malicieux va augmenter graduellement, avec l'augmentation du nombre de nœuds intrus activés dans le réseau *(plus de chance d'être sélectionné comme CH)*.

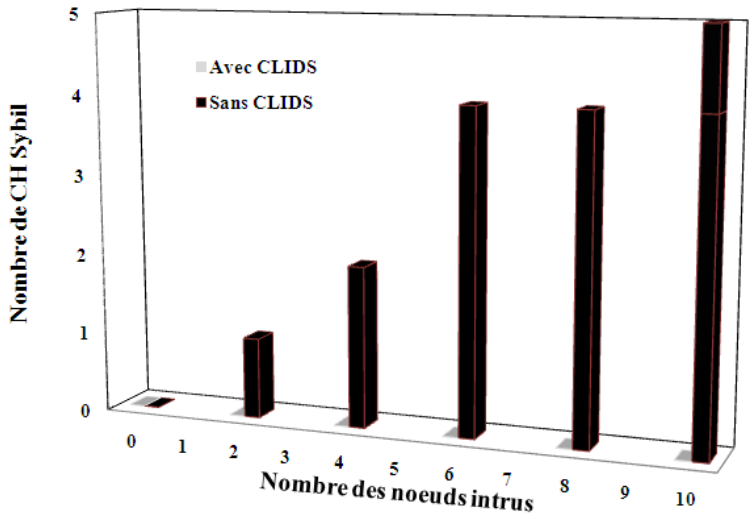



**Figure 8.30** Nombre de CHs malicieux par rapport nombre des nœuds intrus

### 4.2 Capacité de détection d'intrusions au niveau de la couche liaison

La couche liaison peut être aussi ciblée par différents types d'attaques malicieuses, dont la plupart consistent à épuiser les réserves d'énergie des nœuds capteurs. Ainsi, un nœud intrus peut priver les nœuds victimes d'entrer dans leur période de sommeil *(attaque de privation de sommeil)*, de recevoir de grandes quantités de données erronées *(attaques de barrage)*, de prolonger inutilement leur durée d'écoute du trafic *(attaques de désynchronisation)*, ou de

recevoir et rediffuser de grands paquets de données *(attaques par diffusion)*. Dans cette section, nous allons évaluer les performances de notre SDI contre différentes attaques au niveau de la couche liaison. Dans notre simulation, nous avons assumé que tous les nœuds intrus passent par une période d'écoute passive *(100 secondes)*, et décident ensuite de cibler aléatoirement les nœuds du réseau.

### *4.2.1 Évaluation des performances contre les attaques de privation de sommeil*

Afin d'analyser le comportement de notre protocole contre les attaques de privation de sommeil, nous avons mesuré le taux de consommation d'énergie et le nombre de nœuds morts dans le réseau par rapport au temps de simulation. De plus, nous avons assumé que les nœuds attaquants envoient un paquet RTS *(après avoir activé le nœud victime)* à chaque deux cycles d'écoute, afin d'empêcher les nœuds victimes d'entrer dans leur période de sommeil.

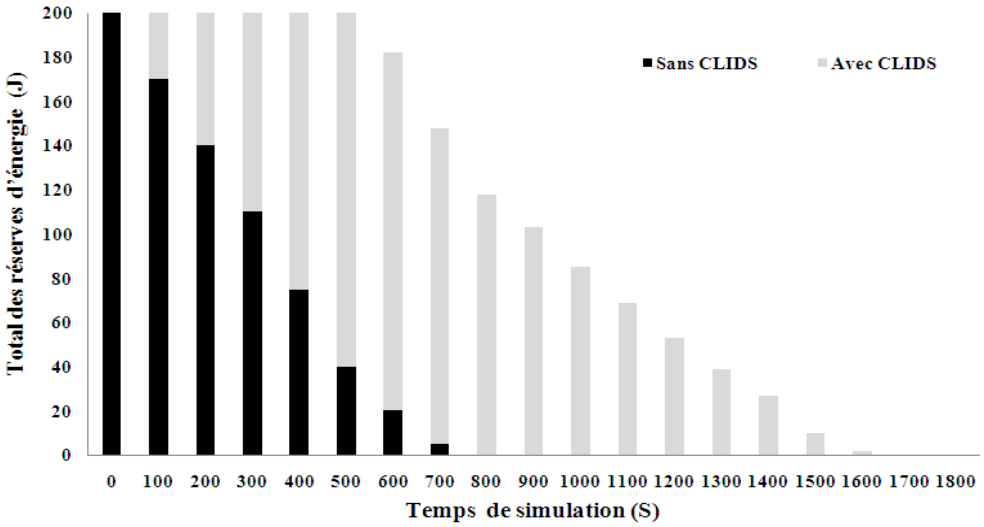



**Figure 8.31 :** Consommation d'énergie sous l'attaque de privation de sommeil

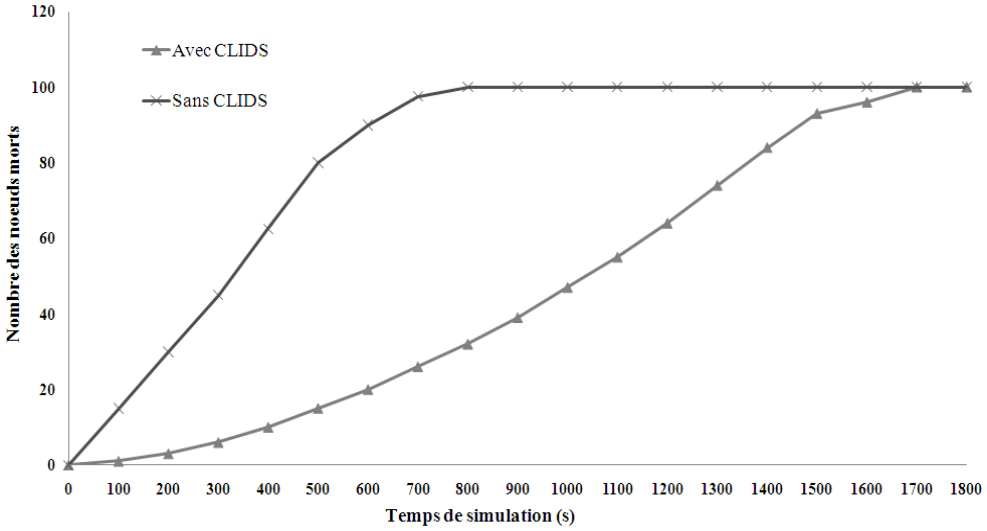



**Figure 8.32 :** Nombre de nœuds morts sous l'attaque de privation de sommeil

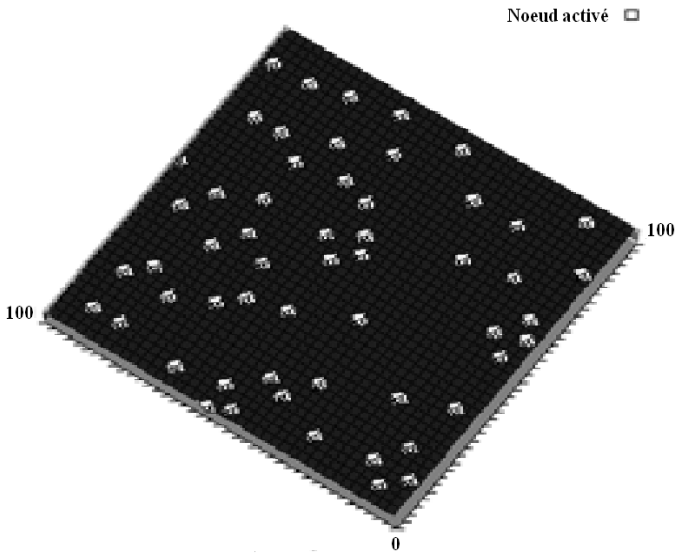



**Figure 8.33 :** Schémas des nœuds activés sous l'attaque de privation de sommeil (Cas sans CLIDS)

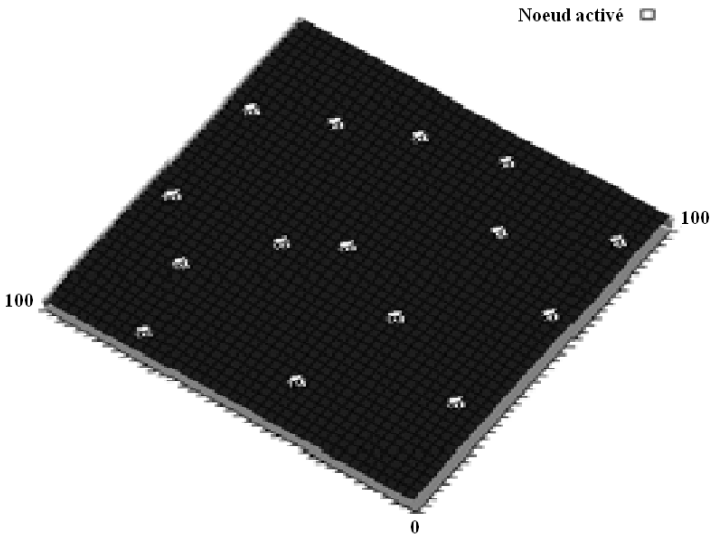



**Figure 8.34 :** Schémas des nœuds activés sous l'attaque de privation de sommeil (Cas avec CLIDS)

Basés sur les résultats de simulation, nous avons démontré que le protocole CLIDS peut prévenir les attaques de privation de sommeil et préserver donc les réserves d'énergie. En effet, avec notre SDI les nœuds du réseau consomment régulièrement leur réserve d'énergie, afin de transmettre les données collectées à la BS. De l'autre côté *(sans CLIDS)*, les nœuds vont épuiser rapidement leur réserve d'énergie *(activation inutile et coûteuse en énergie)*, ce qui réduit considérablement le temps de vie du réseau. En outre, nous pouvons clairement constater dans les figures 8.33 et 8.34, que le nombre de nœuds activés sous l'attaque de privation de sommeil est très élevé dans le cas non sécurisé *(sans CLIDS)*.

### *4.2.2 Évaluation des performances contre les attaques de barrage*

Comme l'attaque de privation sommeil, l'attaque de barrage prive le nœud victime de sa période de sommeil. De plus, ce dernier doit effectuer des taches coûteuses en énergie tel que la réception des données. Dans notre simulation, les nœuds intrus vont envoyer des paquets de données

erronés, après l'envoi de leurs paquets RTS. Les figures suivantes présentent les résultats de simulation obtenus, en termes de consommation d'énergie et du nombre de nœuds morts.

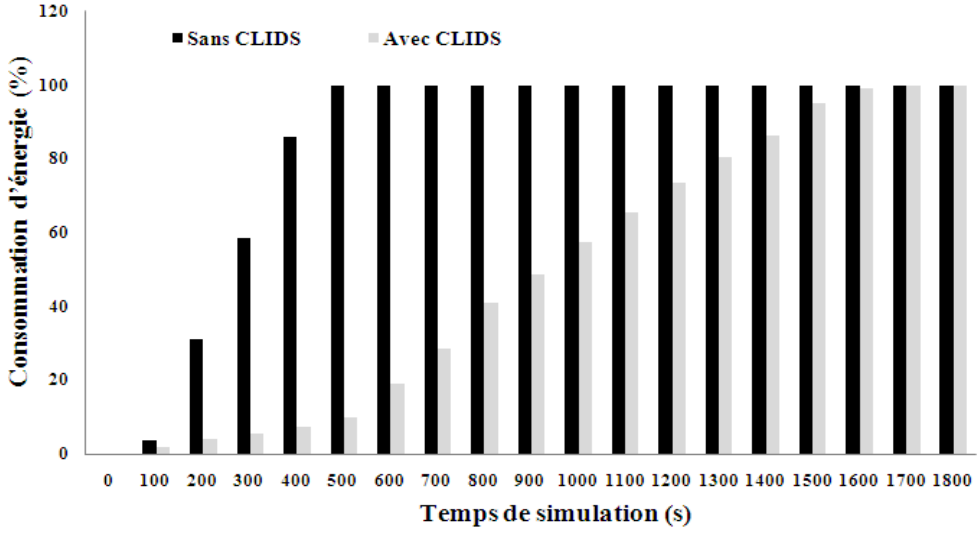



**Figure. 8.35 :** Consommation d'énergie sous l'attaque de barrage

Comme le montrent les figures 8.35 et 8.36, notre mécanisme de sécurité permet de protéger les nœuds du réseau contre les attaques de barrage, et de prolonger la durée de vie du réseau par 220% comparé à celle obtenue sans le protocole CLIDS.

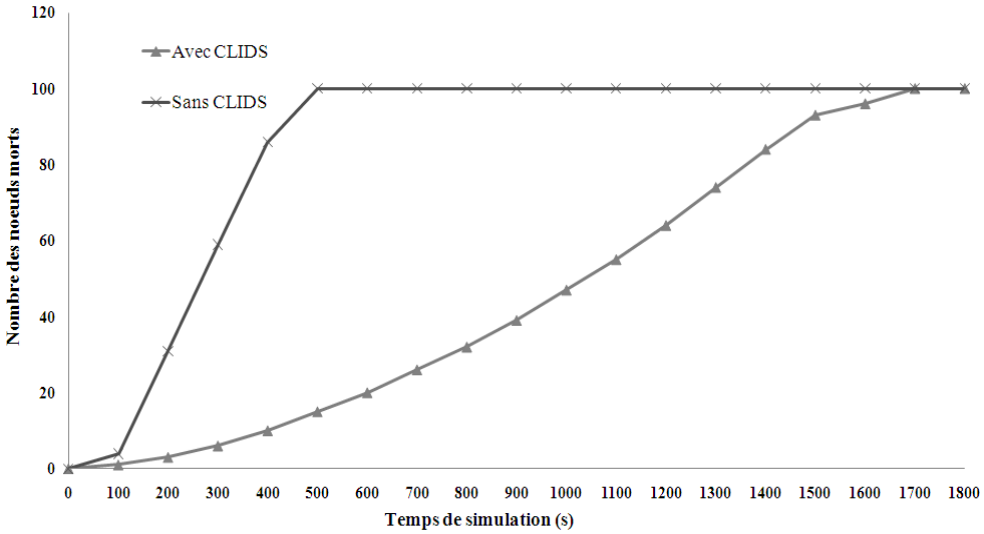



**Figure 8.36 :** Nombre de nœuds morts sous l'attaque de barrage

### *4.2.3 Évaluation des performances contre les attaques de synchronisation*

L'attaque de synchronisation est basée sur le même principe de l'attaque de privation de sommeil, dont lequel la victime est privée de sommeil. Cependant, au lieu d'envoyer des paquets RTS aux nœuds victimes, l'attaquant va transmettre des paquets de synchronisation. En recevant ces derniers, le nœud récepteur va synchroniser son cycle d'activation avec celui du nœud attaquant *(prolonger la durée d'activation)*. Afin d'analyser les performances de notre SDI contre ce type particulier d'attaques, nous avons calculé le pourcentage de la consommation d'énergie des nœuds morts dans le réseau, tout au long de la période de simulation. Ce qui nous a donné comme résultats les figures 8.37 et 8.38.

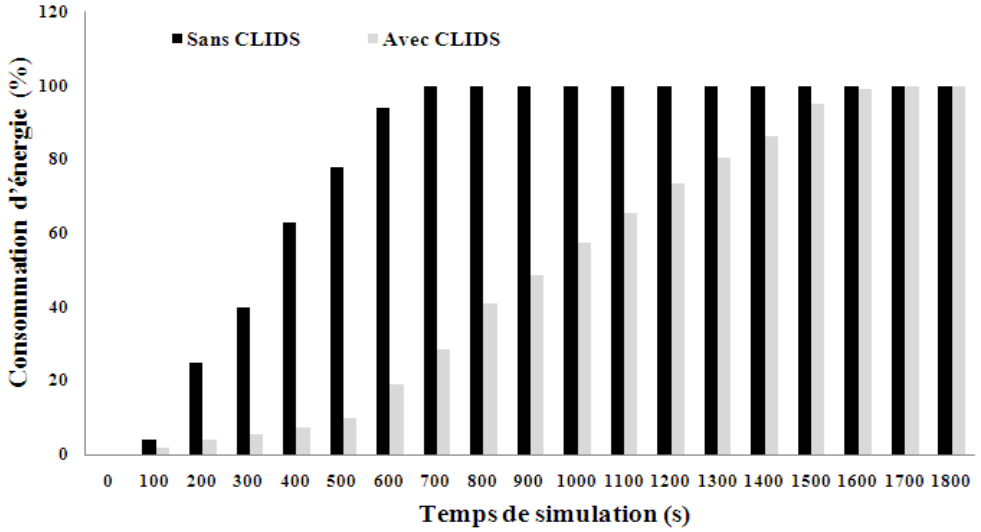



**Figure 8.37 :** Consommation d'énergie sous l'attaque de synchronisation

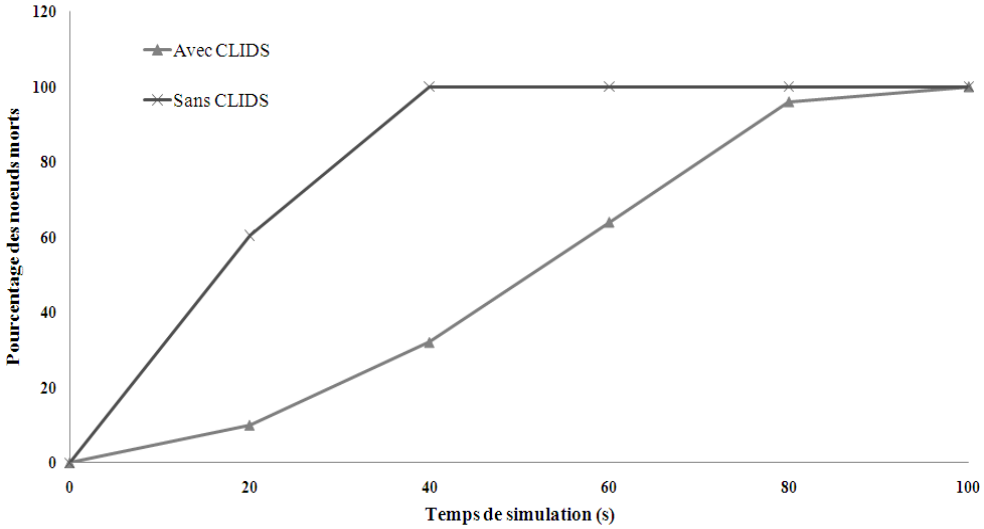



**Figure 8.38 :** Nombre de nœuds morts sous l'attaque de synchronisation

Sans notre SDI, les nœuds capteurs consomment rapidement leur réserve d'énergie, en raison du prolongement de la durée de réveil générée par les faux messages de synchronisation. Par contre, tous ces messages seront rejetés avec le protocole CLIDS, ce qui permet de protéger les nœuds de réseau contre les attaques de synchronisation.

### *4.2.4 Évaluation des performances contre les attaques de diffusion*

La dernière simulation expérimentale consiste à évaluer les performances de notre mécanisme de sécurité contre les attaques de diffusion. Par conséquent, nous supposons les mêmes caractéristiques d'attaquants utilisés dans la simulation de privation de sommeil. De plus, nous assumons que le nœud attaquant peut s'adresser à tous les nœuds dans sa zone de portée radio *(2 mètres)*. En effet, l'attaque par diffusion est difficile à détecter, étant donné qu'elle n'est pas précédée par des messages RTS. Ainsi, les nœuds victimes doivent recevoir les messages de diffusion avant de pouvoir vérifier leurs provenances. Dans notre protocole de communication, les nœuds capteurs peuvent vérifier l'identité du nœud émetteur avant de basculer vers leurs antennes principales. Ainsi, afin de bien évaluer l'effet de cette attaque sur notre protocole de sécurité, nous

assumons que tous les nœuds victimes seront dans un état actif avant la réception des paquets de diffusion *(pas besoin de leur envoyer des paquets d'activation)*. Les figures 8.39 et 8.40 présentent les résultats obtenus.

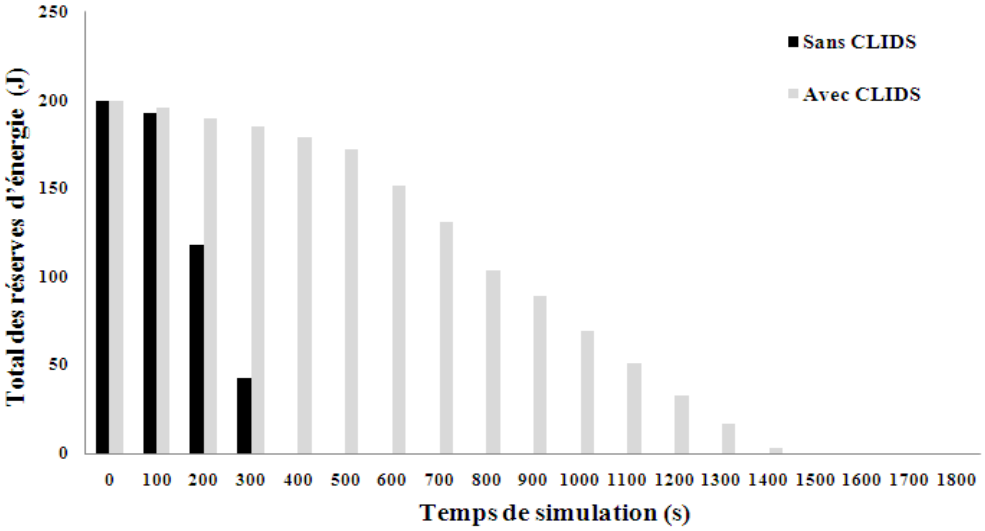



**Figure 8.39 :** Consommation d'énergie sous l'attaque de diffusion

Comme le montrent les figures 8.40 et 8.41, les attaques de diffusion sont les types d'attaques les plus nocives qui affectent les réserves d'énergie des nœuds capteurs. En effet, notre mécanisme de sécurité proposé réduit considérablement l'effet de ces attaques. Cependant, puisque le nœud victime doit recevoir le premier fragment de données avant de pouvoir identifier le nœud attaquant, la durée de vie du réseau obtenue est réduite par rapport aux attaques précédentes. Par contre, notre SDI permet de prolonger par plus de 250% la durée de vie du réseau obtenu dans le cas non sécurisé.

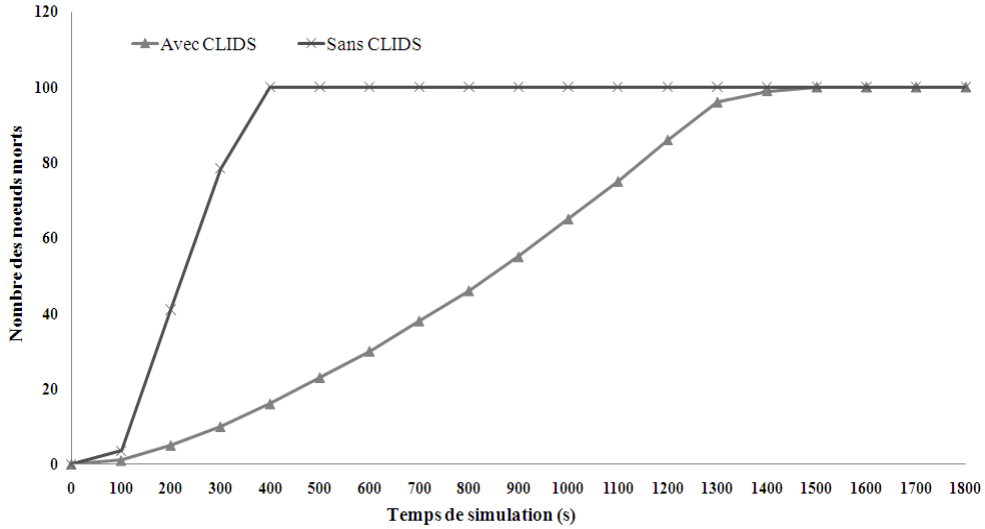



**Figure 8.40 :** Nombre de nœuds morts sous l'attaque de diffusion

Les figures 8.41 et 8.42, représentent le schéma des nœuds éveillés après 300 secondes du temps de simulation. Dans le cas non sécurisé, plus de 70% des nœuds du réseau sont activés afin de recevoir les messages de diffusion envoyés à partir du nœud attaquant d'où l'épuisement rapide

des réserves d'énergie. De l'autre côté, le nombre de nœuds réveillés est considérablement réduit, à cause du niveau de protection offert par notre mécanisme de sécurité.

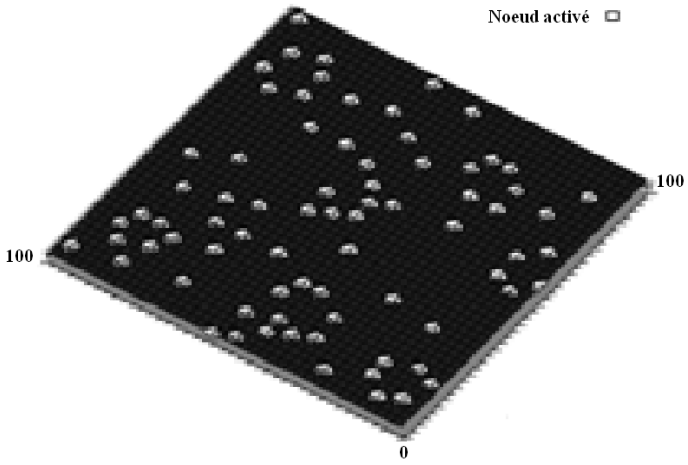



**Figure 8.41 :** Schémas des nœuds activés sous l'attaque de diffusion (Cas sans CLIDS)

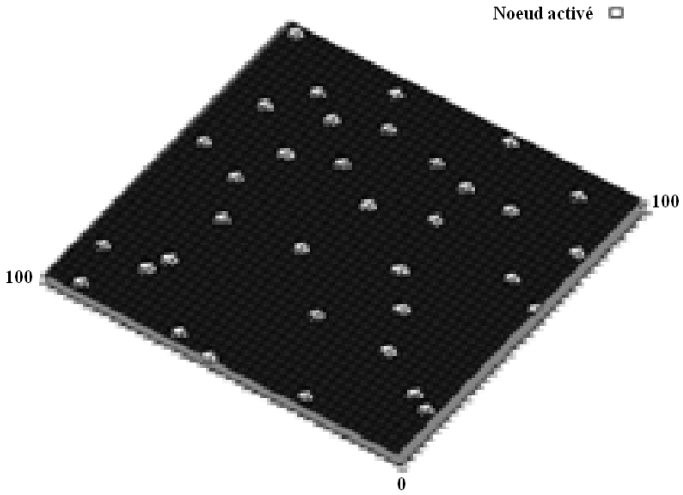



**Figure 8.42 :** Schémas des nœuds activés sous l'attaque de diffusion (Cas avec CLIDS)

## 5. CONCLUSION

Dans ce chapitre, nous avons analysé et évalué les performances de nos contributions en termes de sécurité et d'économie d'énergie. En effet, les résultats de simulation prouvent que notre protocole de communication Cross-layer préserve les réserves d'énergie et prolonge significativement la durée de vie du réseau. Cela est justifié par la gestion efficace des ressources disponibles et le traitement de la majorité des problèmes de gaspillage d'énergie (distance de transmission, écoute passive, activation abusive, interférences, collision et retransmission, puissance de transmission…). De l'autre côté, notre système de détection d'intrusion offre aussi un bon niveau de sécurité, tout en assurant un faible niveau de consommation d'énergie. Les simulations conduites sur ce dernier démontrent son efficacité à contrer plusieurs types d'attaques au niveau de différentes couches du modèle OSI.

# CONCLUSION GENERALE

Le succès des réseaux de capteurs sans fil est fondé sur la simplicité des nœuds capteurs *(faible puissance de calcul, petite batterie, antenne radio à portée limitée…)*. Cependant, ce point fort des RCSFs représente également leur contrainte la plus imposante. En effet, la limitation des ressources des nœuds capteurs engendre plusieurs défis de conception, dont la sécurité et l'économie d'énergie *(sécurité des ressources)* sont les plus importants. L'objectif est de concevoir des protocoles qui offrent un bon niveau de sécurité, tout en respectant les limites de consommation d'énergie. Les approches traditionnelles *(basé sur les architectures en couche)* ont montré leur inefficacité en termes de sécurité et d'économie d'énergie. Subséquemment, dans ce travail nous avons exploré les bénéfices de l'approche Cross-layer, ainsi que son l'application afin de remédier aux limitations des protocoles mono couche.

Dans cette thèse, nous avons proposé deux protocoles à base d'architecture Cross-layer. Le premier consiste en un protocole de communication économique en énergie nommé CLEOP, qui exploite l'interaction des trois couches réseau, Mac et physique. CLEOP combine l'organisation de clusters à chaines, le cycle d'activation *(Duty cycling)* Cross-layer, la radio d'activation à très faible consommation énergétique et l'ajustement Cross-layer de la portée des antennes radio, pour optimiser la consommation d'énergie et améliorer la durée de vie du réseau, ce qui permet de sécuriser le réseau contre les défaillances énergétiques. Ainsi, notre protocole régule la dissipation d'énergie, optimise les distances de transmission, réduit le nombre de nœuds inutilement activés et minimise la perte d'énergie générée par: l'écoute passive, Overhearing, l'activation abusive *(Compulsory wake up)*, les interférences et les problèmes de collision.

Notre deuxième contribution consiste en un système de détection d'intrusions à faible consommation d'énergie nommé CLIDS, qui permet le traitement de différents types d'intrusions au niveau de plusieurs couches du modèle OSI. CLIDS est basé sur une architecture Cross-layer qui exploite l'interaction et la collaboration de trois couches adjacentes du modèle OSI à savoir : réseau, Mac et physique. A base de cette interaction, un modèle de comportement normal sera établi. L'idée de base est de détecter le nœud malveillant lorsqu'il tente de communiquer avec ses nœuds victimes. Cela est effectué au niveau de la couche liaison en se basant sur les informations issues des deux couches adjacentes *(réseau et physique)*. Le système proposé n'exige pas la collecte de nouvelles informations pour la détection d'intrusions, et se contente des informations existantes *(table de routage et valeurs d'RSSI)*. Les résultats expérimentaux démontrent que notre système de détection d'intrusions offre un bon niveau de sécurité, tout en assurant un faible niveau de consommation d'énergie.

Les bons résultats de simulation ne signifient pas que nos deux protocoles sont optimaux. En effet, il existe plusieurs points à améliorer en perspective. L'utilisation d'une deuxième antenne radio dans le protocole CLEOP peut compliquer la conception des nœuds capteurs. L'ajustement des antennes radio doit prendre en compte d'autres paramètres tel que la présence d'obstacle entre les nœuds capteurs. Le protocole CLEOP doit être testé dans le cas où les nœuds capteurs sont

mobiles. En outre, les performances de CLEOP n'ont pas été évaluées dans le cas des réseaux à grande échelle.

De l'autre côté, notre système de détection d'intrusions est dédié aux réseaux à base de topologie en clusters, dans lequel la station de base est responsable de l'établissement du chemin de routage et l'organisation du réseau. Par conséquence, CLIDS doit être amélioré pour s'adapter aux autres types de réseaux. D'autres simulations doivent être conduites afin de tester notre SDI face au plus grand nombre possible d'attaques. L'attaque de compromission des nœuds capteurs reste non détectée par notre SDI, étant donné que le nœud compromis présente un identificateur et un RSSI valides. L'existence d'obstacles après le déploiement des nœuds capteurs peuvent perturber les valeurs d'RSSIs, ce qui entraine de fausses détections positives.

L'implémentation de nos protocoles sur des capteurs *(tels que : Tmote Sky, MICA, Imote ou BTnode)* représente une autre perspective de notre travail, qui permettra d'évaluer les performances de nos contributions dans le mode réel.

# Bibliographie


**[1]** F. Akyildiz, W. S. Sankarasubramaniam, E. Cayirci, "A survey on sensor networks," *IEEE Communications*, Aug 2002.

**[2]** S. Bala, G. Sharma, et A. K. Verma, **"**A survey and taxonomy of symmetric key management schemes for wireless sensor networks," *CUBE '12 Proceedings of the CUBE International Information*, pp. 585-592, 2012.

**[3]** T. Dimitriou, "Efficient mechanisms for secure inter-node and aggregation processing in sensor networks," *Ad-Hoc Networks and Wireless*, pp. 18-31, 2005.

**[4]** H. Soroush, M. Salajegheh, et T. Dimitriou, "Providing transparent security services to sensor networks.," *ICC*, pp. 3431-3436, IEEE, 2007.

**[5]** A. Boukerche, Y. Ren, et L. Mokdad, "Applying symmetric and asymmetric key algorithms for the security in wireless networks: proof of correctness," *Proceedings of the 6th ACM workshop on QoS and security for wireless and mobile networks, Q2SWinet '10*, (New York, NY, USA), pp. 33-40, ACM, 2010.

**[6]** A. K. Das, "An improved efficient key distribution mechanism for large-scale heterogeneous mobile sensor networks," *International Journal of Information Processing*, vol. 2, no. 3, 2011.

**[7]** W. Stallings, "Network Security Essentials *(2nd Edition),*" Pearson Education, 2nd edition, 2003.

**[8]** R. Shirey, "Internet Security Glossary (RFC 2828)," The Internet Society, May 2000.

**[9]** J. Hill, R. Szewczyk, A. Woo, S. Hollar, D. Culler, et K. Pister. "System architecture directions for networked sensors". ACM SIGP LAN Notices, vol. 35(11):pp. 93-104, 2000.

**[10]** T. El Gamal, "A public key cryptosystem and a signature scheme based on discrete logarithms," *in Proceedings of CRYPTO 84 on Advances in cryptology*, (New York, NY, USA), Springer-Verlag New York, Inc,pp. 10-18, 1985.

**[11]** N.J. Al-Karaki, U.M. Raza, et E. K. Ahmed, "Data Aggregation in Wireless Sensor Networks-Exact and Approximate Algorithms," *Proceedings of IEEE Workshop on High Performance Switching and Routing (HPSR)*, pp 18-21, Phoenix Arizona, USA. Avril 2004.

**[12]** T. Roosta, S. Pai, P. Chen, S. Sastry, et S. Wicker, "Inherent Security of Routing Protocols in Ad-Hoc and Sensor Networks," *Global Telecommunications Conference, 2007 GLOBECOM '07, IEEE*, pp.1273-1278, 26-30 Nov 2007.

**[13]** C. Karlof et D. Wagner, "Secure routing in wireless sensor networks: Attacks and countermeasures", In *Proceedings of the 1st IEEE International Workshop on Sensor Network Protocols and Applications*, pp. 113-127, Mai 2003.

**[14]** A.D. Wood et J.A. Stankovic, "Denial of service in sensor networks," *IEEE Computer*, Vol. 35, No. 10, pp. 54-62, 2002.

**[15]** E. Shi et A. Perrig, "Designing secure sensor networks," *Wireless Communication Magazine*, Vol. 11, No. 6, pp. 38-43, Décembre 2004.

**[16]** C. Hartung, J. Balasalle, et R. Han, “Node compromise in sensor networks: The need for secure systems,” Technical Report CU-CS-988-04, Department of Computer Science, University of Colorado at Boulder, 2004.

**[17]** Stajano et R. Anderson, “The Resurrecting Duckling: Security Issues for Ad-hoc Wireless Networks,” *In ICISC*, Springer-Verlag, 2000.

**[18]** M. Brownfield, Y. Gupta, et N. Davis, ”Wireless sensor network denial of sleep attack,” *In Proc. 6th Annu. IEEE SMC Inf. Assurance Workshop*, pp. 356–364. 2005.

**[19]** M. Pirretti, S. Zhu, N. Vijaykrishnan, P. Mcdaniel, et M. Kandemir, “The Sleep Deprivation Attack in Sensor Networks: Analysis and Methods of Defense,” *International Journal of Distributed Sensor Networks*, Vol.2, pp. 267–287, 2006.

**[20]** L. Xiaoming, M. Spear, K. Levitt, N.S. Matloff, et S.F. Wu, “A Synchronization Attack and Defense in Energy-Efficient Listen-Sleep Slotted MAC Protocols,” *ECURWARE '08. Second International Conference on Emerging Security Information, Systems and Technologies*, pp. 403-411, 2008.

**[21]** Y.l. Law, “Link-layer Jamming Attacks on SMAC,” Technical Paper, Univ. of Twente, NL, 2005.

**[22]** W. Ye, J. Heidemann, et D. Estrin, “An energy-efficient MAC protocol for wireless sensor networks,” *in IEEE Infocom*, pp. 1567–1576, 2002.

**[23]** J. Newsome, E. Shi, D. Song, et A. Perrig, “The Sybil attack in sensor networks: Analysis and defenses,” *In Proceedings of the 3rd International Symposium on Information Processing in Sensor Networks*, pp. 259-268, ACM Press 2004.

**[24]** B. Parno, A. Perrig, et V. Gligor, “Distributed detection of node replication attacks in sensor networks,” *In Proceedings of IEEE Symposium on Security and Privacy*, Mai 2005.

**[25]** M. Gruteser, G. Schelle, A. Jain, R. Han, et D. Grunwald, “Privacy-aware location sensor networks”, *In Proceedings of the 9th USENIX Workshop on Hot Topics in Operating Systems, (HotOS IX)*, 2003.

**[26]** H. Chan et A. Perrig, “Security and privacy in sensor networks,” *IEEE Computer Magazine*, pp. 103-105, 2003.

**[27]** J. Deng, R. Han, et S. Mishra, “Countermeasures against traffic analysis in wireless sensor networks,” Technical Report CU-CS-987-04, University of Colorado at Boulder, 2004.

**[28]** G. Gaubatz, et al, "Public Key Cryptography in Sensor Networks-Revisited," *ESAS '04 : 1st European Wksp, Security in Ad-Hoc and Sensor Networks*, 2004.

**[29]** K. Piotrowski, P. Langendoerfer, et S. Peter, "How Public Key Cryptography incense Wireless Sensor Node Lifetime," *SASN'06, Alexandria, Virginia*, USA, Octobre 2006.

**[30]** A.S. Wander, N. Gura, H. Eberle, V. Gupta, et S.C. Shantz, "Energy Analysis of Public-Key Cryptography for Wireless Sensor Networks", *PerCom '05*, Mars 2005.

**[31]** N. Gura, A. Patel, A. Wander, H. Eberle, et S.C. Shantz, "Comparing Elliptic Curve Cryptography and RSA on 8-bit CPUs", *Boston, Massachusetts: 6th International Workshop on Cryptographic Hardware and Embedded Systems*, Aout 2004.

**[32]** F. B. Ian, S. Gadiel, et P.S. Nigel, "Advances in Elliptic Curve Cryptography", London Mathematical Society Lecture Note Series (No. 317), Avril 2005.

**[33]** A. J. Menezes, S. A. Vanstone, et P. C. V. Oorschot, “Handbook of Applied Cryptography”, Boca Raton, FL: CRC Press,1996.

**[34]** R. L. Rivest, “The RC5 Encryption Algorithm,” *Fast Software Encryption, B. Preneel (Ed.)*, Springer, 1995, pp. 86–96.

**[35]** D. Eastlake III et P. Jones, “US Secure Hash Algorithm 1(SHA1),” RFC 3174 (Informational), Sept. 2001.

**[36]** R. L. Rivest, “The MD5 Message-Digest Algorithm,” RFC 1321, Avril 1992.

**[37]** L. Eschenauer et V. D. Gligor, “A Key-Management Scheme for Distributed Sensor Networks,” *Proc. 9thACM Conf. Comp. and Commun. Sec*, pp. 41–47, 2002.

**[38]** W. Du , J. Deng, Y.S. Han, et P.K. Varshney, “A pair-wisekey pre-distribution scheme for wireless sensor networks,” *In Proceedings of the 10th ACM Conference on Computer and Communications Security*, pp. 42-51, New York, NY, USA, ACM Press, 2003.

**[39]** R. Blom, “An Optimal Class of Symmetric Key Generation Systems,” *Proc. EUROCRYPT ’84 Wksp. Advances in Cryptology: Theory and App. of Cryptographic Techniques*, pp. 335–38, 1985.

**[40]** S. Zhu, S. Setia, et S. Jajodia, “LEAP: Efficient Security Mechanisms for Large-Scale Distributed Sensor Networks,” *Proc. 10th ACM Conf. Comp. and Commun. Sec*, pp. 62–72, 2003.

**[41]** A. Perrig, R. Szewczyk, J.D. Tygar, V. Wen, et D.E. Culler, “SPINS: Security Protocols for Sensor Networks,” *Wireless Network*, Vol. 8, pp. 521–34, 2002.

**[42]** M. F. Younis, K. Ghumman, et M. Eltoweissy, “Location-Aware Combinatorial Key Management Scheme for Clustered Sensor Networks,” *IEEE Trans. Parallel and Distrib.Sys.*, Vol. 17, pp. 865–82, 2006.

**[43]** B. Panja, S. K. Madria, et B. Bhargava, “Energy and Communication Efficient Group Key Management Protocol for Hierarchical Sensor Networks,” *SUTC ’06:Proc. IEEE Int’l. Conf. Sensor Networks, Ubiquitous, and Trustworthy Comp*, pp. 384–93, 2006.

**[44]** M. Luk, A. Perrig, et B. Whillock, "Seven cardinal properties of sensor network broadcast authentication," *Proceedings of the fourth ACM workshop on Security of ad hoc and sensor networks*, pp. 147-156, 2006.

**[45]** C. Karlof, N. Sastry, et D. Wagner, "TinySec: a link layer security architecture for wireless sensor networks," *Proceedings of the 2nd international conference on embedded networked sensor systems*, pp. 162-175, 2004.

**[46]** M. Luk, G. Mezzour, A. Perrig, et V. Gilgor, "MiniSec: a secure sensor network communication architecture," *Proceedings of the 6th international conference on Information processing in sensor networks*, pp. 479-488, Avril 25 -27, 2007.

**[47]** P. Rogaway, M. Bellare, J. Black, et T. Krovetz, "OCB: a block-cipher mode of operation for efficient authenticated encryption," *Proceedings of the 8th ACM conference on Computer and Communications Security*, 2001.

**[48]** A. Pirzada et C. McDonald, “Establishing trust in pure ad hoc networks,” *In Proceedings of the 27th Australian Conference on Computer Science*, Dunedin, New Zealand, pp. 47-54, 2004.

**[49]** Z. Yan, P. Zhang, et T. Virtanen, “Trust evaluation based security solution in ad hoc networks,” *In Proceedings of the7th Nordic Workshop on Secure IT Systems*, 2003.

**[50]** K. Ren, T. Li, Z. Wan, F. Bao, R.H. Deng, et K. Kim, “Highly reliable trust establishment scheme in ad hoc networks”, *Computer Networks: The International Journal of Computer and telecommunications Networking*, Vol 45, pp.687-699, Aout 2004.

**[51]** S. Tanachaiwiwat, P. Dave, R. Bhindwale, et A. Helmy, “Location-centric isolation of misbehavior and trust routing in energy-constrained sensor networks,” In *Proceedings of IEEE International Conference on Performance, Computing, and Communications*, pp. 463-469, Avril 2004.

**[52]** Z. Liang et W. Shi, “PET: A Personalized Trust model with reputation and risk evaluation for P2P resource sharing,” In *Proceedings of the HICSS-38*, Hilton Waikoloa Village Big Island, Hawaii, Janvier 2005.

**[53]** Z. Liang et W. Shi, “Analysis of ratings on trust inference in the open environment,” Technical report MIST-TR-2005-002, Department of computer Science, Wayne State University, Feb 2005.

**[54]** R. Bace, “Intrusion Detection,” Mac Millan Technical Publishing, 2000.

**[55]** K. Ilgun, R. A. Kemmerer, et P. A. Porras, “State transition analysis: A rule-based intrusion detection approach," *Software Engineering*, Vol. 21, no. 3, pp. 181-199, 1995.

**[56]** U. Lindqvist et P. A. Porras, “Detecting computer and network misuse through the production-based expert system toolset (p-BEST)," *in IEEE Symposium on Security and Privacy*, pp. 146-161, 1999.

**[57]** H. S. Javitz et A. Valdes, “The NIDES statistical component: Description and justification," Annual report, Computer Science Laboratory, SRI International, Menlo Park, CA, March 1994.

**[58]** M. Stahlberg, “Radio Jamming attacks against two popular mobile networks,” *In Helsinki University of Tech. Seminar on Network Security*, 2000.

**[59]** 3 Com IEEE802.11b Wireless LANs Technical Paper.13p., referred 10.10.2000.

**[60]** C.C. Li, H.Q. Pei, L. P. Ning, Qingquan, “An Effective Scheme for Defending Denial-of-Sleep Attack in Wireless Sensor Networks,” *Fifth International Conference on Information Assurance and Security*, pp. 446-449, 2009.

**[61]** D.R. Raymond, R.C. Marchany, M.I. Brownfield, et S.F. Midkiff, “Effects of Denial-of-Sleep Attacks on Wireless Sensor Network MAC Protocols,” *IEEE transactions on vehicular technology*, Vol. 58, No. 1, pp. 367-380, Jan 2009.

**[62]** F. Rainer, et H. Hans-Joachim, “Fighting Insomnia a Secure Wake-Up Scheme for Wireless Sensor Networks”. Third International Conference on Emerging Security Information, Systems and Technologies, pp.191-196, 2009.

**[63]** A. Gabrielli, L.V. Mancini, S. Setia, et S. Jajodia, “Securing Topology Maintenance Protocols for Sensor Networks: Attacks and Countermeasures,” *IEEE Transactions on Dependable and Secure Computing*, pp. 450–465, 2011.

**[64]** J. Deng, R. Han, et S. Mishra, “INSENS: Intrusion-tolerant routing in wireless sensor networks,” Technical Report CUCS-939-02, Department of Computer Science, University of Colorado at Boulder, Nov 2002.

**[65]** W. Wang et B. Bhargava,, “Visualization of wormholes insensor networks,” *In Proceedings of the 2004 ACM Workshop on Wireless Security*, pp. 51-60, New York, NY,USA, ACM Press, 2004.

**[66]** J.-H.Yun, I.-H.Kim, J.-H.Lim, et S.W. Seo. “WODEM: Wormhole Attack Defense Mechanism in Wireless Sensor Networks,” *In Ubiquitous Convergence Technology (ICUCT2006)*, pp. 200–209. LNCS 4412, 2007.

**[67]** Y. Hu, A. Perrig, et D. Johnson, “Packet leashes: A defense against wormhole attacks in wireless ad hoc networks,” 2001.

**[68]** Q. Zhang, P. Wang, D. S. Reeves, et P. Ning. “Defending against Sybil attacks in sensor networks,” *In 25th IEEE International Conference on Distributed Computing Systems Workshops (ICDCS 2005 Workshops)*, pp. 185–191, 2005.

**[69]** J. Yin et S. K. Madria. “Sybil attack detection in a hierarchical sensor network,” *In Third International Conference on Security and Privacy in Communications Networks and the Workshops (Secure Comm 2007)*, pp. 494–503, 2007.

**[70]** H. Deng, W. Li, et D.P. Agrawal, “Routing Security in Wireless Ad Hoc Networks,” *IEEE Communications Magazine*, Vol. 40, no. 10, Oct 2002.

**[71]** I. Krontiris, T. Dimitriou et F.C. Freiling, “Towards intrusion detection in wireless sensor networks,” *Proceeding of the 13th European Wireless Conference, (EW’ 07), CiteSeer*, 2007.

**[72]** W. Lou et Y. Kwon. H-SPREAD, “A hybrid multipath scheme for secure and reliable data collection in wireless sensor networks,” *IEEE Transactions on Vehicular Technology*, Vol. 55, no.4, pp. 1320–1330, 2006.

**[73]** Y. Jian, “A hierarchical secure routing protocol against black hole attacks in sensor networks,” *IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing*, Vol.1, 2006.

**[74]** M. Satyajayant, B. Kabi, et X. Guoliang, “BAMBi: Blackhole Attacks Mitigation with Multiple Base Stations in Wireless Sensor Networks”, *IEEE ICC proceedings*, 2011.

**[75]** S. Marti, T. J. Giuli, K. Lai, et M. Baker, “Mitigating routing misbehavior in mobile ad hoc networks,” *in MobiCom ’06: Proceedings of the 6th annual international conference on Mobile computing and networking*. ACMPress, pp. 255–265, 2000.

**[76]** Y. Zhang, W. Lee, et Y.-A.Huang, “Intrusion detection techniques for mobile wireless networks,” *Wirel.Netw*, Vol. 9, no. 5, pp. 545–556, 2003.

**[77]** P. Michiardi et R. Molva, “Core: a collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks,” *in Proceedings of the IFIP TC6/TC11 Sixth Joint Working Conference on Communications and Multimedia Security. Deventer*, The Netherlands: Kluwer, *pp. 107–121*, 2002.

**[78]** F. Kargl, S. Schlott, et M. Weber, “Sensors for detection of misbehaving nodes in MANETs” *Praxis der Informationsverarbeitung und Kommunikation*, 28(1), 38-44. 2005.

**[79]** T. Aura, P. Nikander, et J. Leiwo, “DOS-resistant authentication with client puzzles,” *In Revised papers from the 8th International Workshop on Security Protocols*, pp.170-177, Springer-Verlag, 2001.

**[80]** H. Choi, S. Zhu, et T. F. La Porta. “SET: Detecting node clones in sensor networks”. *In Third International Conference on Security and Privacy in Communications Networks and the Workshops (Secure Comm 2007)*, pp. 341–350, 2007.

**[81]** R. Brooks, P. Y. Govindaraju, M. Pirretti, N. Vijaykrishnan, et M. T. Kandemir, “On the Detection of Clones in Sensor Networks Using Random Key Pre distribution,” *IEEE*

*Transactions on Systems, Man, and Cybernetics*, Part C: Applications and Reviews, Vol. 37 no. 6, pp.1246–1258, 2007.

**[82]** C. Bekara et M. Laurent-Maknavicius. "A New Protocol for Securing Wireless Sensor Networks against Nodes Replication Attacks," *In Third IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMOB 2007)*, pp. 59–59, 2007.

**[83]** F. Hartung, et M. Kutter, "Watermarking Techniques," *Proc. IEEE*, Vol. 87, pp. 1079-1107, 1999.

**[84]** F. Chuhong, D. Kundur, et R.H. Kwong, "Analysis and Design of Secure Watermark-Based Authentication Systems," *IEEE Trans. Inf. Forensics Secur*, Vol. 1, pp. 43-55, 2006.

**[85]** M. Chen, Y. He, et R.L. Lagendijk, "Error Detection by Fragile Watermarking," *In Proceedings of the 22nd of the Picture Coding Symposium*, Seoul, Korea, pp. 287-290, Avril 2001.

**[86]** P.F. Luis, C. Pedro, R.T.P Juan, et P.F Fernando, "Watermarking Security: A Survey," LNCS *Trans. Data Hiding Multimedia Secur*, Vol. 1, pp. 41-72, 2006.

**[87]** H. Guo, Y. Li, et S. Jajodia, "Chaining Watermarks for Detecting Malicious Modifications to Streaming Data," *Inf. Sci*, Vol. 177, pp.281-298, 2007.

**[88]** I. Kamel, et H. Guma, "Simplified Watermarking Scheme for Sensor Networks," *Int. J. Internet Protoc. Technol. Indersci*, Vol. 5, pp.101-111, 2010.

**[89]** R. Sutharshan, L. Christopher, et P. Marimuthu, "Anomaly detection in wireless sensor networks," *IEEE Wireless Communications*, Vol. 15, no. 4, pp.34–40, 2008.

**[90]** D. Subhadrabandhu, S. Sarkar et F. Anjum, "Rida: Robust intrusion detection in ad hoc networks," *in 4th International IFIP-TC6 Networking Conference on Networking Technologies, Services, and, Protocols*, Vol. 3462, pp. 1069-1082, 2005.

**[91]** C. Ko, P. Brutch, J. Rowe, G. Tsafnat, et K. N. Levitt, "System health and intrusion monitoring using a hierarchy of constraints," in *RAID '00: Proceedings of the 4th International Symposium on Recent Advances in Intrusion Detection*, pp. 190-204, 2001.

**[92]** C. Ko, M. Ruschitzka, et K. Levitt, "Execution monitoring of security-critical programs in distributed systems: a specification-based approach," in *SP '97: Proceedings of the 1997 IEEE Symposium on Security and Privacy*, pp. 175-187, 1997.

**[93]** A. P. da Silva, M. Martins, B. Rocha, A. Loureiro, L. Ruiz, et H. C. Wong, "Decentralized intrusion detection in wireless sensor networks," *in Proceedings of the 1st ACM international workshop on Quality of service & security in wireless and mobile networks (Q2SWinet '05)*. ACM Press, pp. 16–23, Oct 2005.

**[94]** I. Onat et A. Miri, "An intrusion detection system for wireless sensor networks," *Proceeding of the IEEE International Conference on Wireless and Mobile Computing, Networking and Communications*, vol. 3, Montreal, Canada, pp. 253–259, Aout 2005.

**[95]** C. E. Loo, M. Y. Ng, C. Leckie, et M. Palaniswami, "Intrusion detection for routing attacks in sensor networks," *International Journal of Distributed Sensor Networks*, 2005.

**[96]** V. Bhuse et A. Gupta, "Anomaly intrusion detection in wireless sensor networks," *Journal of High Speed Networks*, Vol. 15, no. 1, pp. 33–51, 2006.

**[97]** F. Anjum, D. Subhadrabandhu, S. Sarkar, et R. Shetty. “On optimal placement of intrusion detection modules in sensor networks,” *In BROADNETS '04: Proceedings of the First International Conference on Broadband Networks*, pp. 690-699. 2004.

**[98]** R. Roman, J. Zhou, et J. Lopez. “Applying intrusion detection systems to wireless sensor networks,” *In Proceedings of IEEE Consumer Communications and Networking Conference (CCNC '06)*, pp. 640-644. Las Vegas, USA, Jan 2006.

**[99]** C.C. Su, K.M. Chang, Y.H. Kue, et M.F. Horng, “The new intrusion prevention and detection approaches for clustering-based sensor networks,” *in Proceedings of 2005 IEEE Wireless Communications and Networking Conference (WCNC’05)*, Vol. 4, New Orleans, L.A., Mar, pp. 1927-1932, 2005.

**[100]** C.H.N. Edith, L. Jiangchuan, et R.L. Michael, “On the intruder detection for sinkhole attack in wireless sensor networks,” *In Proceedings of the IEEE International Conference on Communications*, pp. 3383–3389, 2006.

**[101]** E.L. Chong, N. Mun Yong, L. Christopher, et P. Marimuthu, “Intrusion detection for routing attacks in sensor networks,” *International Journal of Distributed Sensor Networks*, Vol. 2 pp. 313–332, 2006.

**[102]** S. Wei-Tsung, C. Ko-Ming, et K. Yau-Hwang, “ehip: An energy-efficient hybrid intrusion prohibition system for cluster-based wireless sensor networks”. *Computer Networks*, Vol. 51, pp.1151–1168, 2007.

**[103]** I. Krontiris, T. Dimitriou, T. Giannetsos, et M. Mpasoukos, “Intrusion detection of sinkhole attacks in wireless sensor networks,” *In Algorithmic Aspects of Wireless Sensor Networks*, Vol. 4837, pp. 150–161, Springer Berlin / Heidelberg, 2008.

**[104]** F. Liu, X. Cheng, et D. Chen. “Insider attacker detection in wireless sensor networks,” *In INFOCOM 2007.26th IEEE International Conference on Computer Communications.IEEE*, pp. 1937–1945, 2007.

**[105]** Z. Yu et J.P. Jeffrey Tsai. “A framework of machine learning based intrusion detection for wireless sensor networks,” *In IEEE International Conference on Sensor Networks, Ubiquitous, et Trustworthy Computing*, pp. 272–279, 2008.

**[106]** W.W. Cohen, et Y. Singer, “A simple, fast, and effective rule learner,” *In Proceedings of the sixteenth national conference on Artificial intelligence and the eleventh Innovative applications of artificial intelligence conference*, pp. 335–342, 1999.

**[107]** I. Krontiris, Z. Benenson, T. Giannetsos, F.C. Freiling, et T. Dimitriou, “Cooperative intrusion detection in wireless sensor networks,” *In Proceedings of the 6th European Conference on Wireless Sensor Networks*, pp. 263–278, 2009.

**[108]** T. Dimitriou, et A. Giannetsos, “Wormholes no more? localized wormhole detection and prevention in wireless networks,” *In Distributed Computing in Sensor Systems*, pp. 334–347, Springer Berlin/Heidelberg, 2010.

**[109]** L. Coppolino, et L. Romano, “Open issues in ids design for wireless biomedical sensor networks,” *In Intelligent Interactive Multimedia Systems and Services*, Vol. 6, pp. 231–240. Springer Berlin Heidelberg, 2010.

**[110]** T.H. Hai, E.N. Huh, et M. Jo. “A lightweight intrusion detection framework for wireless sensor networks,” *Wirel. Commun. Mob. Comput*, Vol. 10, no.4, pp.559–572, 2010.

**[111]** K. Akkaya et M. Younis,"A survey on routing protocols for wireless sensor networks," *Ad Hoc Networks*, Vol.3, no. 3, pp. 325–349, 2005.

**[112]** A.A. Ahmed, H. Shi, et Y. Shang, "A Survey on Network Protocols for Wireless Sensor Networks," *Proc. IEEE Int Conf. Information Technology: Research et Education (ITRE '03)*, 2003.

**[113]** J. Kulik, W.R. Heinzelman, H. Balakrishnan, "Negotiation–Based Protocols for Disseminating Information in Wireless Sensor Networks," *In: Wireless Networks*, Vol. 8, pp. 169-185, 2002.

**[114]** C. Intanagonwiwat, R. Govindan et D. Estrin, "Directed Diffusion: a scalable et robust communication paradigm for sensor networks," *ACM Press*, 2000.

**[115]** J. N. Al-Karaki, et A. E. Kamal, , "Routing Techniques in Wireless Sensor Networks: A Survey," *IEEE Wireless Communications*, Vol. 11, pp. 6-28, 2004.

**[116]** Y. Xu, J. Heidemann, et D. Estrin, "Geography-informed energy conservation for ad hoc routing," *7th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom 01)*, Rome, Italy, July 2001.

**[117]** S. Bandyopadhyay, E. Coyle, "An Energy Efficient Hierarchical Clustering Algorithm for Wireless Sensor Networks," *Proceedings of IEEE INFOCOM*, Vol. 3, pp. 1713-1723, 2003.

**[118]** S. Lindsey, et S. Raghavendra, "Data Gathering Algorithms in Sensor Networks Using Energy Metrics," *IEEE Transactions on parallel and distributed systems*, Vol.13, no.9, 2002.

**[119]** S. Lindsey, C. Raghavendra, "PEGASIS: Power-Efficient Gathering in Sensor Information Systems," *IEEE Aerospace Conference Proceedings*, Vol. 3, pp. 1125-1130, 2002.

**[120]** K. Du, J. Wu et D. Zhou, "Chain-based protocols for data broadcasting and gathering in sensor networks," *International Parallel and Distributed Processing Symposium*, Avril 2003.

**[121]** M. Younis, M. Youssef et K. Arisha, "Energy-aware Routing in Cluster-Based Sensor Networks", *the Proceedings of the 10th IEEE/ACM International Symposium on Modelling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS2002)*, Oct 2002.

**[122]** G. Lee, J. Kong, et O. Byeon, "Cluster based Energy Aware Routing Protocol for Sensor Networks," *From Proceeding (527) Networks and Communication Systems*, 2006.

**[123]** Handy, M., Haase, M., Timmermann, "Low Energy Adaptive Clustering Hierarchy with Deterministic Cluster-Head Selection," *4th IEEE International Conference on Mobile and Wireless Communications Networks*, Stockholm, 2002

**[124]** S.Lee, J.Yoo, T.Choong "Distance-Based Energy Efficient Clustering for Wireless Sensor Networks," *29th Annual IEEE International Conference on Local Computer Networks (LCN'04)*, pp. 567-568, 2004.

**[125]** W.R. Heinzelman, A. Chandrakasan, et H.Balakrishnan, "An Application-Specific Protocol Architecture for Wireless Micro sensor Networks," *IEEE Transactions on the wireless communications*, Vol. 1, no. 4, pp. 660-670, Oct 2002.

**[126]** W.R. Heinzelman, A. Chandrakasan, H. Balakrishnan "Energy-efficient communication protocol for wireless micro sensor networks," *IEEE Hawaii International Conference on System Sciences*, 2000.

**[127]** P. Agarwal et C. Procopiuc, "Exact and Approximation Algorithms for Clustering," *In Proceedings of the Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 658-667, Jan 1999.

**[128]** J. Neander, E. Hansen, M. Nolin, et M. Bjorkman, "Asymmetric Multihop Communication in Large Sensor Networks," *Wireless Pervasive Computing, 1st International Symposium on.* pp. 16-18, Jan 2006.

**[129]** H. Sajid, et W.M. Abdul, "Hierarchical Cluster-based Routing in Wireless Sensor Networks," *in Proceeding of 5th Intl. Conf. on Information Processing in Sensor Network (IPSN06)*, USA, 19-21, Avril 2006.

**[130]** P.T.V. Bhuvaneswari, V. Vaidehi, et S. Shanmugavel, "SPEAR: sensor protocol for energy aware Routing in wireless sensor network," *in Proceeding of IEEE Third International Conference on Wireless Communication & Sensor Networks (WCSN-2007)*, pp. 133-137, Dec 2007.

**[131]** F. Xiangning, et S. Yulin, "Improvement on LEACH Protocol of Wireless Sensor Network," *Sensor Technologies and Applications, SensorComm, International Conference on*, pp. 14-20, Oct 2007.

**[132]** M. Lehsaini, H. Guyennet, et M. Feham, "An efficient cluster-based self-organisation algorithm for wireless sensor networks," *International Journal of Sensor Networks (IJSNET)*, Vol. 7, pp. 85-94, 26 Feb 2010.

**[133]** X. Jia, J. Ning, et L. Xizhong, "Improvement of LEACH protocol for WSN. In : Fuzzy Systems and Knowledge Discovery (FSKD)," *9th International Conference on. IEEE*, pp. 2174-2177, 2012.

**[134]** Z. Fuzhe, X. You, et L. Ru, "Improved Leach Communication Protocol for WSN," *In Control Engineering and Communication Technology (ICCECT), International Conference on IEEE*, pp. 700-702, 2012.

**[135]** F. HUI, X. WANG, et S. Xin, "A STATIC-LEACH WSNs for Hazardous Materials Monitoring," *Advanced Materials Research*, Vol. 463, pp. 261-265, 2012.

**[136]** B. MANZOOR, N. JAVAID, et O. REHMAN, "Q-LEACH: A New Routing Protocol for WSNs," *Procedia Computer Science, Elsevier*, pp.1–6, 2013.

**[137]** F. Chunyao, J. Zhifang, et W. Wei, "An Energy Balanced Algorithm of LEACH Protocol in WSN," *IJCSI International Journal of Computer Science Issues*, Vol. 10, Issue 1, no. 1, pp.354-359, 2013.

**[138]** S. Lindsey, C. S. Raghavendra et K. Sivalingam, "Data Gathering in Sensor Networks using the Energy*Delay Metric," in the Proceedings of the IPDPS Workshop on Issues in Wireless Networks and Mobile Computing, San Francisco, CA, Avril 2001.

**[139]** J. Sung-Min, H. Young-Ju, et C. Tai-Myoung, "The concentric clustering scheme for efficient energy consumption in the PEGASIS," *Proceedings of 9th IEEE Advanced Communication Technology Conference*, pp. 260-265, Feb 2007.

**[140]** L. Jung-Eun, et K. Keecheon, "Diamond-Shaped Routing Method for Reliable Data Transmission in Wireless Sensor Networks," *IEEE International Symposium on Parallel and Distributed Processing with Applications*, 2008.

**[141]** Ye, Wei, "Medium access control with coordinated adaptive sleeping for wireless sensor networks," *IEEE/ACM Transactions on Networking*, Vol. 12, no. 3, pp. 783–791, Juin 2004.

**[142]** T. Van Dam et K. Langendoen, "An adaptive energy-efficient MAC protocol for wireless sensor networks," *in ACM Sensys*, Nov 2003.

**[143]** G. Lu, B. Krishnamachari, et C. Raghavendra, "An adaptive energy-efficient and low-latency MAC for data gathering in sensor networks," *in Ad Hoc and Sensor Networks*, Avril 2004.

**[144]** J. Polastre, J. Hill, et D. Culler, "Versatile low power media access for wireless sensor networks," *in ACM Sensys*, Nov 2004.

**[145]** A. El-Hoiydil et J.-D. Decotigniel, "WiseMAC : An ultra low power MAC protocol for multi-hop wireless sensor networks," *in Algorithmic Aspects of Wireless Sensor Networks*, ser. LNCS, Vol. 3121. Springer Berlin / Heidelberg, pp. 18–31, 2004.

**[146]** M. Buettner, G. Yee, E. Anderson, et R. Han, "X-MAC : A short preamble Mac protocol for duty-cycled wireless sensor networks," *in ACM Sensys*, Nov 2006.

**[147]** I. Rhee, A. Warrier, M. Aia, et M. J., "ZMAC : a hybrid MAC for wireless sensor networks," *in ACM Sensys*, Nov 2005.

**[148]** M. Brownfield, K. Mehrjoo, A. Fayez, et N. Davis, "Wireless sensor network energy-adaptive macprotocol," *in IEEE Consumer Communications and Networking Conference*, Jan 2005.

**[149]** T. Nieberg, S. Dulman, P. Havinga, L. Van Hoese, et J. Wu, "Collaborative algorithms for communication in wireless sensor networks," *in Ambient Intelligence : Impact on Embedded Systems*, Kluwer Academic Publishers, Nov 2003.

**[150]** P. Havinga et L. Van Hoese, "A lightweight medium access protocol LMAC for wireless sensor networks: Reducing preamble transmissions and transceiver state switches," *in International Conference on Networked Sensing Systems (INSS)*, Juin 2004.

**[151]** M. Nosovic et T. Todd. "Low Power Rendezvous and RFID Wakeup for Embedded Wireless Networks," *In Annual IEEE Computer Communications Workshop*, 2000.

**[152]** L. C. Z. C. Guo et J. M. Rabaey, "Low Power Distributed MAC for Ad Hoc Sensor Radio Networks," *In IEEE Globe-Com*, Nov. 2001.

**[153]** E. Shih, P. Bahl, et M. J. Sinclair, "Wake on Wireless: An Event Driven Energy Saving Strategy for Battery Operated Devices," *In ACM Mobicom'2002*.

**[154]** C. Schurgers, V. Tsiatsis, S. Ganeriwal, et M. Srivastava. "Topology Management for Sensor Networks: Exploiting Latency and Density," *In MobiHoc'02*, 2002.

**[155]** T. Melodia , C. M. Vuran , D. Pompili, "The State of the Art in Cross-Layer Design for Wireless Sensor Networks," *IEEE proceedings of eurongi workshops on wireless and mobility, Springer*, 2005.

**[156]** V. Srivastava et M. Motani, "Cross-Layer Design: A Survey and the Road Ahead," *IEEE Communications Magazine*, Vol. 43(12), pp.112-119, 2005.

**[157]** S. Shakkottai, T. S. Rappaport, and P. C. Karlsson, "Cross-Layer Design for Wireless Networks," *IEEE Commun. Mag.*, Vol. 41, no. 10, pp. 74–80, Oct. 2003.

**[158]** V. T. Raisinghani and S. Iyer. "Cross layer design optimizations in wireless protocol stacks," *Computer Communications*, Vol.27 (8), pp.720-725, Mai 2004.

**[159]** R. Winter, J. Schiller, N. Nikaein, and C. Bonnet, "Crosstalk: Cross-layer decision support based on global knowledge," *IEEE Communications Magazine*, Vol. 44, pp. 2-8, Jan 2006.

**[160]** V. T. Raisinghani and S. Lyer., "Eclair : An efficient cross layer architecture for wireless protocol stacks," 5th World Wireless Congress, San Francisco, USA, Mai 2004.

**[161]** R. Sasanka, J. Srinivasan, and W. Yuan, "The illinois grace project : Global resource adaptation through cooperation," *Proceedings of the Workshop on Self-Healing, Adaptive, and self-MANaged Systems (SHAMAN)*, pp. 144-155, Juin 2002.

**[162]** K. Seada, M. Zuniga, A. Helmy, B. Krishnamachari, "Energy-efficient forwarding strategies for geographic routing in lossy wireless sensor networks," *in ACM Sensys '04*, Nov 2004.

**[163]** J. Yuan, Z. Li, W. Yu, B. Li, "A Cross-Layer optimization framework for multicast in multi-hop wireless networks wireless internet," *in Proc. WICON '05*, pp. 47-54, July 2005.

**[164]** M. Kubisch, H. Karl, A. Wolisz, L.C. Zhong et J. Rabaey, "Distributed Algorithms for Transmission Power Control in Wireless Sensor Networks," *IEEE WCNS*, 2003.

**[165]** Y. Fang, et B. McDonald, "Dynamic code word routing (DCR): a cross-layer approach for performance enhancement of general multi-hop wireless routing," 2004.

**[166]** M. L. Sichitiu, "Cross-layer scheduling for power efficiency in wireless sensor networks", *in: INFOCOM 2004,Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies*, pp. 7-11, 2004.

**[167]** L. van Hoesel, T. Nieberg, J. Wu, et P. J. M. Havinga, "Prolonging the lifetime of wireless sensor networks by cross-layer interaction," *In proceeding of the IEEE Wireless Communications*, Vol. 11, no. 6, pp. 78-86, Dec 2004.

**[168]** B. DeCleene, V. Firoiu, M. Dorsch, et S. Zabele, "Cross-layer protocols for energy-efficient wireless sensor networking," *in Proceedings of IEEE Military Communications Conference (MILCOM '05)*, Vol. 3, Atlatnic City, NJ USA, pp. 1477–1484, Oct 2005.

**[169]** C. Suh, Y. Ko, et D. Son, "An Energy Efficient Cross-Layer MAC Protocol for Wireless Sensor Networks," *in: proceeding of the APWeb 2006*, LNCS 3842, pp. 410–419, 2006.

**[170]** C.F. Chou, "A Cross-Layer Design of Energy-Efficient Wireless Sensor Networks", *in: proceeding of the 2005 Systems Communications (IEEE ICW'05)*, 2005.

**[171]** D. Ferrara, et. al., "MACRO: An Integrated MAC/Routing Protocol for Geographical Forwarding in Wireless Sensor Networks," *in Proc. IEEE Infocom '05*, vol. 3, pp. 1770 - 1781, Mars 2005.

**[172]** I. F. Akyildiz., M. C. Vuran et O. B. Akan, "A Cross layer protocol for wireless sensor networks", *in: proceeding of Conference on Information Sciences and Systems (CISS'06)*, Princeton, NJ, 2006.

**[173]** S. Liu, Y. Bai, M. Sha, Q. Deng, et D. Qian, "CLEEP: A Novel Cross-Layer Energy-Efficient Protocol for Wireless Sensor Networks," *in: proceeding of the Wireless Communications, Networking and Mobile Computing, IEEE WiCOM*, 2008.

**[174]** N. Chilamkurti, S. Zeadally, A. Vasilakos, et V. Sharma1, "Cross-Layer Support for Energy Efficient Routing in Wireless Sensor Networks," *Journal of Sensors*, Hindawi Publishing Corporation, Vol. 2009, pp. 9, 2009.

**[175]** P. Kumar, M. Günes¸, Q. Mushtaq, et J. Schiller, "A real-time and energy-efficient MAC protocol for wireless sensor networks," *International Journal of Ultra Wideband Communications and Systems (IJUWBCS)*, pp.28-30, Avril 2009.

**[176]** F. Tong , R. Xie, L. Shu et Y.C. Kim, "A Cross-Layer Duty Cycle MAC Protocol Supporting a Pipeline Feature for Wireless Sensor Networks," *MDPI Sensors Journal*, pp. 5183-5201, 2011.

**[177]** J.K MURTHY, S. KUMAR, et A. SRINIVAS, "Energy efficient scheduling in cross layer optimized clustered wireless sensor networks," *Int Journal of Computer Science and Communication*, Vol. 3, no 1, pp. 149-153, 2012.

**[178]** **L.** Lazos R. Poovendran, "Cross-layer design for energy-efficient secure multicast communications in ad hoc networks," *Communications. IEEE IC*, Vol. 6(20-24), pp. 3633-3639, 2004.

**[179]** L. Eschenauer et V.D. Gligor "A Key-Management Schemes for Sensor Networks," *The 9th ACM CCCS*, pp. 41-47, 2002.

**[180]** L. Lazos, J. Salido, R. Poovendran, "VP3: Using vertex path and power proximity for energy efficient key distribution," *VTC2004-Fall. IEEE*, Vol. 2, pp.1228-1232, 2004.

**[181]** K. Jones, A. Wadaa, S. Oladu, et L. Wilson, "Towards a New Paradigm for Securing Wireless Sensor Networks," *In Proceedings of the 2003 workshop on NSP*, pp. 115-121, 2003.

**[182]** R. MURALEEDHARAN, et L.A. OSADCIW, "Security: Cross Layer Protocol in Wireless Sensor Network,' *In : INFOCOM 2006. 25th IEEE International Conference on Computer Communications*, Proceedings IEEE, pp. 1-2, 2006.

**[183]** R. MURALEEDHARAN, W. GAO, et L.A. OSADCIW, "Meta-heuristic cross-layer protocol for UWB emergency responder sensor network," *In : Swarm Intelligence Symposium, SIS 2008. IEEE*, pp. 1-6, 2008.

**[184]** G. THAMILARASU, et R. SRIDHAR, "XLSEC-A Distributed Cross-layer Framework for Security in Wireless Sensor Networks," *In : Consumer Communications and Networking Conference, CCNC 2009*, 6th IEEE, pp. 1-2, 2009.

**[185]** K. Sharma, et M. K. Ghose, "Complete Security Framework for Wireless Sensor Networks," *(IJCSIS) International Journal of Computer Science and Information Security*,Vol. 3, No. 1, 2009.

**[186]** C.I. Hsun, H. Chou-Ting, et K. Yau-Hwang, "An adaptive cross-layer design approach for network security management,' *In : Advanced Communication Technology (ICACT), 13th International Conference on*, IEEE, pp. 1085-1089, 2011.

**[187]** H.A. Rahhal, I.A. Ali, et S.I. Shaheen, "A novel Trust-Based Cross-Layer Model for Wireless Sensor Networks," *28th National Radio Science Conference NRSC*, pp. 1-10, 2011.

**[188]** S. Puri, et S.P Tripathi, "Dynamic High Level Cross Layer Security Mechanisms for Wireless Sensor Networks," *International Journal of Information Technology and Computer Science (IJITCS)*, Vol.4, no.6, pp.45-56, 2012.

**[189]** A. Bilami, et D. Boubiche, "A hybrid Energy Aware Routing Algorithm for Wireless Sensor Networks," *in: proceeding of the IEEE Symposium on Computers and Communications (ISCC'08)*, Marrakech-Morocco, pp. 975-980, 6 July 2008.

**[190]** D.E. Boubiche, et A. Bilami, "HEEP (Hybrid Energy Efficiency Protocol) based on chain clustering," *Int. J. Sensor Networks*, Inderscience Publishers, Vol.10, No. 1/2, pp.25–35, 2011.

**[191]** D.E. Boubiche, et A. Bilami, "Un Protocole de Communication Cross Layer pour l'économie d'énergie dans les RCSFs, " *JEESI 12*, 16 Avril 2012.

**[192]** Data Sheet–TR1000 916.50 MHz Transceiver 6.5*10mmPackage, http://www.rfm.com/ products/data/tr100 0.pdf.

**[193]** Data Sheet–CC1000 Single Chip Very Low Power RF Transceiver," http://www. chipcon.com/files/CC1000_ Data_Sheet_2_3.pdf

**[194]** Data Sheet–2.4 GHz IEEE 802.15.4/Zigbee-Ready RF Tranceiver, http: //www. chipcon.com/files/CC2420_Data_Sheet_1_4.pdf.

**[195]** EM2420 – 2.4 GHz IEEE 802.15.4/Zigbee RF Tranceiver, http://www.ember.com/pdf/ EM2420 datasheet .pdf

**[196]** B. Berthomieu, et M. Menasche, "An Enumerative Approach for Analyzing Time Petri Nets," *IFIP Congress Series*, North Holland, Vol. 9, pp. 41-46, 1983.

**[197]** IEEE Std. 802.11-, Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, ISO/IEC 8802-11:1999(E), IEEE Std. 802.11. Part 11, 1999.

**[198]** D.E. Boubiche, et A. Bilami, "A Cross-Layer Energy-efficient Transmission Power Control Mechanism for Wireless Sensor Networks," *2nd International Symposium on Modelling and Implementation of Complex Systems*, pp. 97-101, Constantine, Algeria, May 20-21, 2012.

**[199]** D.E. Boubiche, et A. Bilami, "Cross Layer Intrusion Detection System for Wireless Sensor Network," *International Journal of Network Security & Its Applications (IJNSA)*, Vol.4, No.2, pp.35-52, Mars 2012.

**[200]** D.E. Boubiche, A. Bilami, et S. Athmani, "A Cross Layer Energy Efficient Security Mechanism for Denial of Sleep Attacks on Wireless Sensor Network," *CCIS Part II*, Vol. 294, Spinger LNCS. Avril 2012.

**[201]** D.E. Boubiche, et A. Bilami, "A Defense Strategy against Energy Exhausting Attacks in Wireless Sensor Networks," *Journal of Emerging Technologies in Web Intelligence*, Vol. 5(1), pp. 18-27. 2013.

**[202]** Information Sciences Institute, "The Network Simulator ns-2" http://www.isi.edu/nanam/ns/, University of Southern California.