



REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE  
MINISTERE DE L'ENSEIGNEMENT SUPERIEURE ET DE LA RECHERCHE  
SCIENTIFIQUE

**UNIVERSITE IBN KHALDOUN - TIARET**

# MEMOIRE

Présenté à :

FACULTÉ MATHÉMATIQUES ET INFORMATIQUE  
DÉPARTEMENT D'INFORMATIQUE

Pour l'obtention du diplôme de :

**MASTER**

Spécialité : Réseaux et Télécommunications

Par :

**AGRED Naima**  
**RABIA Kheira**

Sur le thème

---

## **Simulation de la surveillance des réseaux de capteurs sans fil (RCSF) sous Omnet++**

---

Soutenu publiquement le 07/07/2019 à Tiaret devant le jury composé de :

Mr MEBAREK Bendaoud	MCA	Université Ibn Khaldoun	Président
Mr MOSTEFAOUI Kadda	MAA	Université Ibn Khaldoun	Encadreur
Mr ALEM Abdelkader	MAA	Université Ibn Khaldoun	Examineur

## **Résumé :**

La durée de vie dans un réseau sans fil, en particulier un réseau de capteurs sans fil dépend fortement du facteur connectivité entre ses nœuds. Plusieurs facteurs peuvent être à l'origine d'une rupture de connectivité tels que : le manque d'énergie au niveau d'un nœud important, infection d'un nœud vital par un code malveillant, une défaillance logique ou physique d'un nœud primaire etc. Cette rupture peut conduire dans certains cas à une reconfiguration du réseau en engendrant ainsi un over Head préjudiciable ou dans d'autres cas à un échec de la mission assignée au réseau. Dans ce mémoire, nous avons modélisé et simulé la surveillance des réseaux de capteurs sans fil (RCSF) sous Omnet++ en se basant sur la reconfiguration des nœuds avec le principe de redondance pour garder une couverture de la zone et assurer une longue durée de vie du réseau en cas de panne ou d'épuisement d'énergie d'un nœud actif du RCSF.

**Mots clés :** RCSF, capteur, réseau ad hoc, pannes, tolérance au panne, redondance, approche distribuée, surveillance, simulation, omnet++, castalia.

## **Abstract:**

The lifespan in a wireless network, especially a wireless sensor network, strongly depends on the connectivity factor between its nodes. Several factors can cause connectivity disruption such as: lack of power at a large node, infection of a vital node with malicious code, logical or physical failure of a node primary etc. This rupture can lead in some cases to a reconfiguration of the network thus generating a detrimental overhead or in other cases to a failure of the mission assigned to the network. In this thesis, we modeled and simulated the wireless sensor network monitoring (WSN) under Omnet ++ based on node reconfiguration with the redundancy principle to maintain coverage of the area and ensure a long life of the network. Network in case of power failure or depletion of an active node of the CWNF.

**Key words:** WSN, sensor, ad hoc network, failures, fault tolerance, redundancy, distributed approach, monitoring, simulation, omnet ++, castalia.

## Remerciement

*En premier lieu nous remercions Dieu le tout puissant pour toute la volonté et le courage qu'il nous a donné pour l'achèvement de ce travail.*

*Nous remercions notre encadreur, Mr MOSTEFAOUI Kadda, maître-assistant à l'université IBN KHALDOUN de TIARET pour son entière disponibilité, son aide inestimable, ses conseils et remarques constructives tout au long de temps de travail.*

*Merci également aux membres du jury qui ont accepté d'évaluer notre travail, Mr MEBAREK BENDAOUD comme un président et Mr ALEM Abdelkader comme un examinateur, et aussi nous remercions le professeur MEGHAZI Hadj Madani pour son aide judicieuse, sa disponibilité, ses orientations et ses précieux conseils. L'aboutissement de ce travail doit beaucoup à sa contribution.*

*Un grand merci à tous nos collègues en Master 2 et tous les enseignants du département Informatique, qui nous ont offert un environnement de travail extrêmement agréable. Un grand merci à notre famille, pour son soutien qui nous a poussé à chercher au fond de nous la volonté de faire toujours beaucoup plus, à nos amis et tous ceux qui ont contribué de près ou de loin à l'aboutissement de ce travail.*

## *Dédicaces*

*Avec tout respect et amour je dédie ce travail*

*A mes chers parents : MEDJADJE LATIFA ma mère et AGRED*

*ABD\_ELRRAHMENE mon père pour tous les efforts consentis pour m'assurer  
une bonne éducation.*

*A mes frères : ZOLA, SAMIA, ABDOU, ASSIA pour tout leur soutien moral et  
leur amour et affection.*

*A mon oncle MOHAMMED et ma tante FATMA.*

*A mes cousins HAYAT, AMINA, HAFIDA, NASSIRA, KARIMA,  
ABD\_ELBASSET et KHALILE.*

*A mon marie.*

*A toutes mes amis.*

*Aussi bien à tous ceux qui m'ont aidé.*

*AGRED Naima*

## *Dédicaces*

*Merci au dieu*

*Je dédie ce travail à tous ceux qui me sont chers au cœur*

*A mes chers parents, mon père M'hamed et ma mère Ouda.*

*A mes chères sœurs et mes chers frères Bilal et Khalil.*

*A toute la famille de près et de loin.*

*A ma collègue et amie Naima qui a été une binomiale très compétente lors de la réalisation du présent mémoire.*

*A mes chers amis de la promotion, de l'université et partout dans le monde pour tout leur soutien moral.*

*A toute personne qui m'a aidé un jour à réussir jusqu'ici, en espérant être toujours à la hauteur de leurs attentes et de leurs espérances.*

*RABIA Kheira*

## Table de matières

Résumé	
Remerciement	
Dédicaces	
Table des matières	
Introduction générale	
List des figures	
List des tableaux	
<b>Chapitre 01 : Etat de L'art sur les Réseaux de Capteur Sans Fil</b>	
1.1 Introduction.....	13
1.2 Le Capteur .....	13
1.3 Qu'est-ce qu'un capteur .....	13
1.4 Architecture physique d'un capteur.....	13
1.5 Exemples de capteur sans fil.....	14
1.6 Caractéristiques d'un capteur.....	14
1.7 Les réseaux Ad Hoc.....	15
1.3.1 Les Réseaux de capteurs sans fil (RCSF) .....	15
1.3.2 Histoire de Capteurs Sans Fil des Réseaux.....	15
1.3.3 Architecture d'un Réseau de capteur sans fil.....	16
1.3.4 Les types d'architectures des RCSF.....	17
1.3.5 Différentes topologies dans les RCSF.....	18
1.3.6 La pile protocolaire.....	20
1.3.7 Les différents types de capteurs.....	22
1.3.8 Les types de communications .....	22
1.3.9 La collection d'information.....	23
1.3.10 Les modèles des réseaux de capteurs sans fil(RCSF) .....	23
1.3.11 Les domaines d'application.....	24
1.3.12 Les principales caractéristiques des réseaux sans fil.....	27
1.3.13 Contraintes de conception des RCSF.....	28
1.3.14 Notion d'énergie.....	29
1.3.15 Systèmes d'exploitation pour les RCSF.....	29
1.3.16 Principaux défis des réseaux de capteurs sans fil.....	30
1.4 Conclusion.....	30
<b>Chapitre 02 : Principaux concepts de surveillance des réseaux de capteurs sans fils</b>	
2.1 Introduction.....	32
2.2 Définition d'un système de surveillance.....	32
2.3 Type de systèmes de surveillance selon le type de nœuds.....	32
2.3.1 Systèmes de surveillance à base de capteurs scalaires.....	32
2.3.2 Systèmes de surveillance à base de capteurs multimédia.....	32
2.3.3 Systèmes de surveillance à base de capteurs hybrides.....	32
2.4 Les panne dans les réseaux de capteurs.....	32
2.4.1 Définition d'une panne.....	32
2.4.2 Classification des pannes.....	33
2.4.3 Causes de pannes.....	34
2.5 La tolérance aux pannes.....	34
2.5.1 Classification des solutions de tolérance aux pannes.....	35
2.5.1.1 Classification selon la phase de traitement .....	35
2.5.1.2 Classification architecturale.....	36
2.5.1.3 Classification selon le niveau d'implémentation.....	37
2.6 Aspect de surveillance d'un RCSF.....	37

2.6.1 La longévité du réseau.....	37
2.6.2 Connectivité.....	37
2.6.3 La couverture .....	37
2.6.4 Méthodologie de déploiement.....	38
2.7 Le concept de redondance .....	38
2.8 Les Travaux Antérieurs .....	39
2.9 Conclusion.....	40
<b>Chapitre 03 : Modélisation et simulation</b>	
3.1 Introduction.....	42
3.2 Généralité sur la simulation.....	42
3.3 Types de simulation.....	42
3.3.1 Systèmes de simulation discret.....	42
3.3.2 Systèmes de simulation continue.....	42
3.4 Les simulateurs de réseau existant.....	42
3.5 Le Simulateur OMNET++.....	43
3.5.1 Choix du simulateur OMNET++ .....	43
3.5.2 Présentation d'OMNET++ .....	43
3.5.3 Installation du simulateur OMNET++ .....	43
3.5.4 Les principaux fichiers d'OMNET++ .....	44
3.6 Environnement de Castalia .....	44
3.6.1 Installation de castalia .....	45
3.6.2 Les commandes de castalia.....	45
3.7 Environnement de Simulation .....	45
3.7.1 Environnement matériel .....	45
3.7.2 Environnement logiciel .....	45
3.8 Réalisation de la simulation .....	45
3.8.1 Objectif .....	46
3.8.2 Organisation du réseau .....	46
3.8.3 Fichiers résultats : (Résultats de la simulation) .....	49
3.9 Conclusion .....	52
Conclusion générale .....	53
Références .....	54

## List des figures

Figure 1.1–Nœud capteur sans fil.....	13
Figure 1.2–Architecture d’un capteur sans fil.....	13
Figure 1.3–Exemples de capteur sans fil.....	14
Figure 1.4–Réseau de capteurs sans fil.....	17
Figure 1.5–Architecture Plat des RCSF.....	17
Figure 1.6–Architecture hiérarchique des RCSF.....	17
Figure 1.7–Architecture Basée Localisation des RCSF.....	18
Figure 1.8–Topologie en étoile.....	19
Figure 1.9–Topologie maillée.....	19
Figure 1.10–Topologie cluster-tree.....	20
Figure 1.11–La pile protocolaire.....	21
Figure 1.12–Capteurs dans les applications militaires.....	24
Figure 1.13–Capteurs pour l’environnement, sécurité routière.....	25
Figure 1.14–Capteurs dans les infrastructures.....	25
Figure 1.15–Capteurs dans le domaine médical.....	25
Figure 1.16–Capteurs dans la domotique.....	26
Figure 1.17–Capteurs dans l’agriculture.....	27
Figure 2.1–De la faute à la défaillance.....	33
Figure 2.2–Classification des pannes.....	33
Figure 2.3–Procédures de tolérance aux pannes.....	35
Figure 2.4–Interférences Radio.....	39
Figure 3.1–Les nœuds et leurs connections en Castalia.....	44
Figure 3.2–La vie d’un nœud sous Castalia.....	45
Figure 3.3–Déploiement de réseau.....	47
Figure 3.4–Sans défaillance (Général).....	47
Figure 3.5–Avec défaillance (scénario1).....	48
Figure 3.6–a technique de la redondance. (scénario2).....	49
Figure 3.7–La liste des fichiers d’entrée disponibles et des configurations.....	49
Figure 3.8–exemple de contenu du fichier trace.....	50
Figure 3.9–exemple de contenu du fichier résumé.....	50
Figure 3.10–L’énergie moyenne consommée par chaque nœud.....	51
Figure 3.11–L’énergie totale par chaque nœud consommée dans le réseau.....	51
Figure 3.12–Le nombre moyen de paquets transmits par chaque nœud.....	51
Figure 3.13–Le nombre Total des paquets transmits à travers le réseau.....	51

## Liste des tableaux

Tableau 1.1– comparaison entre les réseaux de capteurs et les réseaux Ad-Hoc.....	15
Tableau 1.2 – Générations des RCSF.....	16
Tableau 1.3 – Les taches de la pile protocolaire.....	22
Tableau 3.1 – tableau illustre Les positions des nœuds.....	46
Tableau 3.2 – tableau illustre Les voisinages des nœuds.....	46

## Introduction générale

Depuis quelques années, le marché des réseaux et des applications sans fil s'est considérablement développé. De ce constat, une nouvelle branche s'est créée pour offrir des solutions économiquement intéressantes pour la surveillance à distance et le traitement des données dans les environnements complexes et distribués : les réseaux de capteurs sans fil (RCSFs).

Un RCSF - ou en anglais Wireless Sensor Network (WSN) - est considéré comme un type spécial de réseaux Ad hoc constitué d'un grand nombre de capteurs capables de collecter des données et les transmettre grâce à un routage multi-saut à un nœud considéré comme un "point de collecte", appelé nœud puits (ou Sink). Chaque nœud est alimenté par une batterie individuelle dont la consommation, due à la communication et le traitement des informations, doit être optimisée. La densité importante des nœuds capteurs implique la présence des nœuds redondants.

La surveillance est un dispositif utilisé pour analyser l'état du système. Elle consiste à détecter et classer les défaillances tout en observant l'état du système en temps réel, puis diagnostiquer pour pouvoir localiser les éléments défaillants et déterminer les causes. En général, La conception d'une supervision performante repose sur la combinaison des techniques de surveillance et de diagnostic.

De ce fait dans les RCSF, Certains nœuds capteurs peuvent être bloqués ou tomber en panne à cause d'un manque d'énergie, d'un dégât matériel ou d'une interférence environnementale. La panne d'un nœud capteur ne doit pas affecter le fonctionnement global de son réseau. C'est le problème de fiabilité ou de tolérance aux pannes.

La redondance se rapporte à la qualité ou à l'état d'être en surnombre, par rapport à la normale ou à la logique. Ce qui peut avoir la connotation négative de superflu, mais aussi un sens positif quand cette redondance est voulue afin de prévenir un dysfonctionnement. La redondance est conçue comme le déploiement d'un éventail de versions différentes d'une même structure (redondance structurelle) ou d'une même fonction (redondance fonctionnelle).

La simulation des réseaux de capteurs consiste principalement en la reproduction du comportement et du fonctionnement des nœuds capteurs dans un environnement informatique; pour des raisons tel que : La répétition d'expérience, L'adressage des systèmes complexes, Le gain de temps et la variation des paramètres de simulation alors que la simulation réelle s'avère coûteuse, voire impossible dans quelque cas.

L'objectif de notre travail est de créer dans l'environnement de simulation OMNeT++, un modèle de surveillance du RCSF. Ce modèle est basé sur de redondance des nœuds et qui permet de réveiller les nœuds endormis en cas de panne ou d'épuisement d'énergie d'un nœud actif, donc le but est d'optimiser la consommation énergétique et de prolonger la durée de vie du réseau.

Suite à ça notre Document est organisé en 3 Chapitres :

Le premier chapitre est une introduction pour les réseaux de capteurs sans fils. Il présente leurs domaines d'application, leurs architectures et leurs caractéristiques spécifiques qui font de ces systèmes un domaine en plein essor.

Le second chapitre commence par un rappel sur les principaux concepts de surveillance des réseaux de capteurs sans fils. Ensuite, on perlera sur la tolérance aux pannes et la redondance,

et aussi nous présenterons certains travaux développés dans le cadre de la supervision des RCSF et qui emploient l'approche distribuée.

La dernière partie de ce mémoire débute par une étude comparative des simulateurs de réseaux de capteurs sans fils afin d'en adopter le meilleur répondant à nos besoins. Ensuite, elle aborde la simulation de notre modèle de surveillance du réseau sous omnetpp.

Finalement, on donnera une conclusion générale.

# Chapitre 01

Etat de L'art sur les Réseaux de Capteur Sans Fil

## 1.1 Introduction

Les avancées technologiques récentes confortent la présence de l'informatique au cœur du monde réel.

Durant cette dernière décennie, les réseaux de capteurs sans fil (RCSF) ont été utilisés dans de nombreuses applications telles que la surveillance des forêts, la gestion des catastrophes, l'exploration spatiale, l'automatisation industrielle, et la surveillance des champs de bataille. Dans ce chapitre nous présenterons une description générale sur les capteurs et les Réseaux de capteurs sans fil.

Nous commencerons par présenter un nœud capteur, l'architecture et les caractéristiques et aussi une généralisation sur réseaux de capteurs sans fil (les caractéristiques, les domaines d'application et les topologies des RCSF).

## 1.2 Le Capteur

### 1.2.1 Qu'est-ce qu'un capteur

Un nœud capteur est un outil qui transforme une grandeur physique observée (température, pression,...) en une grandeur utilisable. [1]

Cet outil est composé d'un processeur, une mémoire, un émetteur-récepteur radio avec une antenne, un ensemble de capteurs et une batterie ou autre forme intégrée de récupération d'énergie.



Figure 1.1– Nœud capteur sans fil.

### 1.2.2 Architecture physique d'un capteur

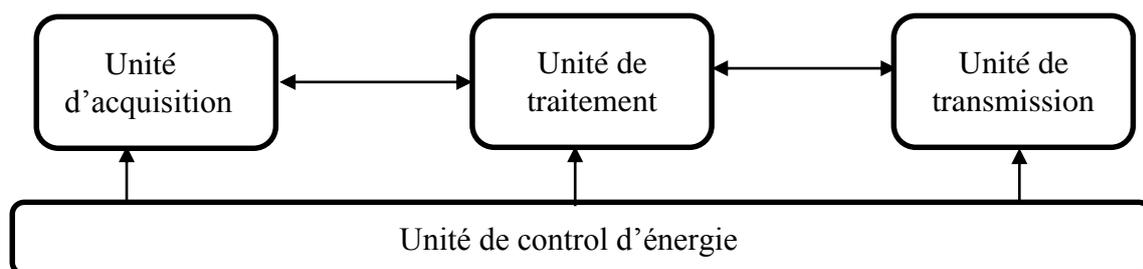


Figure 1.2– Architecture d'un capteur sans fil.

Un capteur est composé de :

- **L'unité d'acquisition** (captage): Elle est composée de deux sous unités, un capteur qui va obtenir l'information de l'environnement local et d'un convertisseur Analogique/Numérique qui va convertir l'information relevée et la transmettre à l'unité de traitement.
- **L'unité de traitement** : l'unité de traitement est composée de deux interfaces, une interface pour l'unité d'acquisition et une interface pour l'unité de transmission. Cette unité

est également composée d'un processeur et d'un système d'exploitation spécifique. Elle acquiert les informations en provenance de l'unité d'acquisition et les envoie à l'unité de transmission.

- **L'unité de transmission (communication)** : l'unité de transmission est responsable de toutes les émissions et réceptions de données via un support de communication radio.
- **L'unité de contrôle d'énergie**: C'est la batterie qui, n'est généralement ni rechargeable ni remplaçable. La capacité d'énergie limitée au niveau des capteurs représente la contrainte principale lors de conception de protocoles pour les réseaux de capteurs. Les unités d'énergie peuvent être supportées par des photopiles qui permettent de convertir l'énergie lumineuse en courant électrique. [2]

### 1.2.3 Exemples de capteur sans fil

Il existe plusieurs modèles commercialisés dans le marché. Parmi les plus célèbres, Le MicaZ de la marque Crossbow. Le Mica2, l'Imote ou le TelosB.

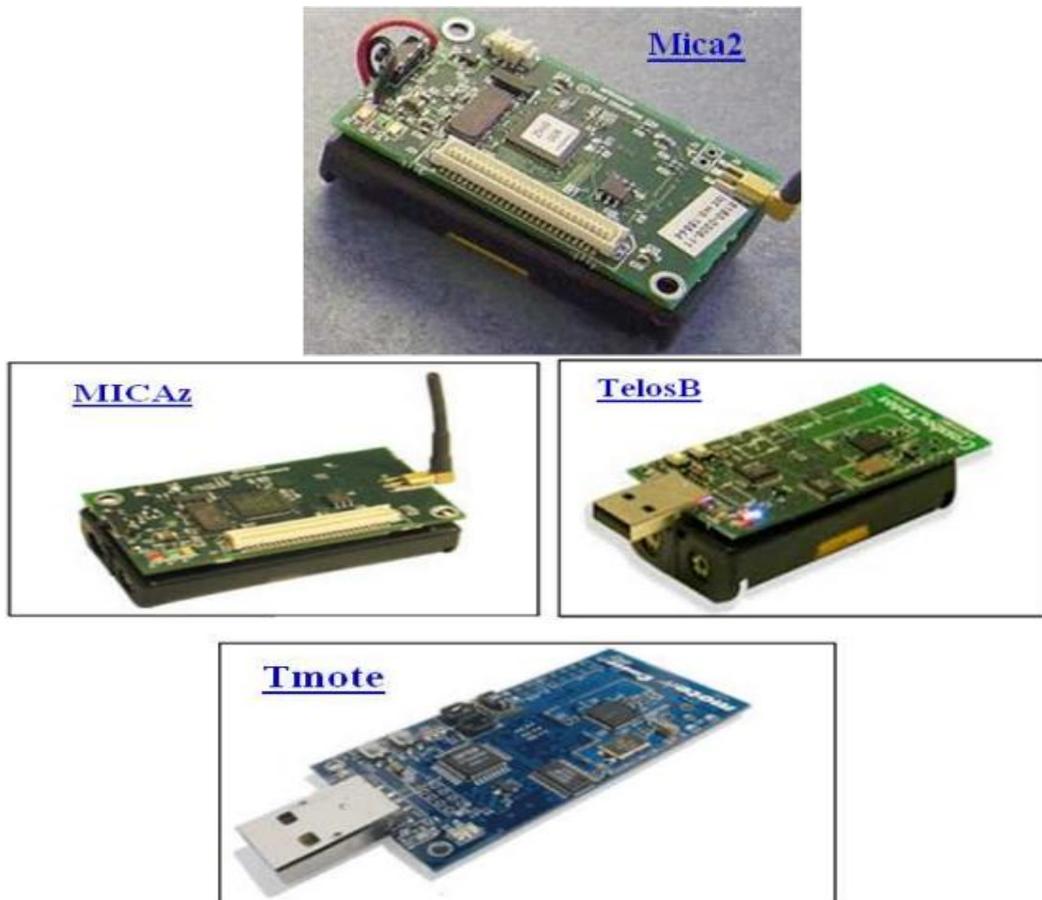


Figure 1.3– Exemples de capteur sans fil. [3]

### 1.2.4 Caractéristiques d'un capteur

- Tolérance aux pannes.
- Traitement des données en temps réel.
- Auto organisation.
- Efficace au niveau matériel consommation, coût, capacité de calcul, etc.
- Déploiement aléatoire ou planifié.

### 1.3 Les réseaux Ad Hoc

Un réseau ad hoc est un réseau formé dynamiquement par un ensemble arbitraire de nœuds indépendants, les nœuds sont supposés se comportés comme des routeurs et des clients à la fois, aussi, chaque nœud est libre de se déplacer et de s'organiser aléatoirement. Ainsi, la topologie du réseau peut changer rapidement et de manière imprévisible. [2]

#### 1.3.1 Les Réseaux de capteurs sans fil (RCSF)

Les réseaux de capteurs sans fil (Wireless Sensor Networks) sont considérés comme un type spécial des réseaux Ad hoc.

Ils sont composés de plusieurs nœuds capteurs déployés dans une zone géographique appelée zone de captage, chaque nœud est capable de surveiller son environnement et de réagir en cas de besoin en envoyant l'information collectée à un ou plusieurs points de collecte, à l'aide d'une connexion sans fil. [2]

Les communications entre les nœuds sont directes ou indirectes, d'où des topologies de réseau.

Le tableau suivant montre une comparaison entre les réseaux de capteurs et les réseaux Ad-Hoc :

Réseaux de capteurs sans fil	Réseaux Ad-hoc
Objectif bien ciblé	Objectif général en communication
Nœuds en collaboration	Chaque nœud a son propre objectif
Flot de données « many to one »	Flot «any to any»
Très grand nombre de nœuds	Nombre limité de nœuds avec notion d'ID
Energie comme facteur déterminant	Débit majeur
Communication broadcast	Communication point à point

**Tableau 1.1-comparaison entre les réseaux de capteurs et les réseaux Ad-Hoc.**

#### 1.3.2 Histoire des Réseaux de Capteurs Sans Fil

L'apparition des réseaux de capteurs ne date pas d'aujourd'hui. Leur première apparition fut à la fin des années 1970 et au début des années 1980. Jusqu'aux années 1990, il fallait un câblage encombrant et coûteux pour acheminer les données d'un capteur vers un point de collecte central. Les progrès du domaine des technologies sans fil (Wireless) ont permis aux réseaux de capteur d'avoir une autre dimension et un champ d'application vaste et varié. Les premiers pas étaient l'œuvre de recherches menées par l'armée américaine dans le but de concevoir des entités autonomes, servant de support technique pour les soldats, capables de surveiller les zones de combat. Effectivement, la croissance spectaculaire des RCSF a commencé avec l'implication de l'agence américaine pour les projets de recherches avancées de défense (DARPA ; Défense Advanced Research Projects Agency) dans le financement des projets de recherches et de développement sur les RCSF notamment le programme SensIT

(Sensor Information Technology) de 1999 à 2002. A partir des années 2000, les universités et instituts, entre autre Stanford, Berkeley et Massachusetts Institute of Technology, ont démocratisé leur développement dans tous les domaines pour constituer aujourd'hui l'une des technologies les plus influentes de notre vie quotidienne. On parle de trois générations de RCSF :

Génération Caractéristiques	1 Génération 1981– 1990	2 Génération 2000- 2003	3 Génération Après 2010
Energie (autonomie)	Grosses batterie (1 Heure à 1 jour)	Batteries AA (1 jour à une semaine)	Solaire (1 mois à plusieurs années)
Taille	Grande boîte a Chaussures	Boîte de cartes	Particule de poussière
Poids	Kilogrammes	Grammes	Négligeable
Technologie	En étoile (un seul saut)	Client/Server, Peer to Peer	En Multi- sauts(peer to peer)
Déploiement	Un seul capteur	Mise en place à la main	Embarqué

**Tableau 1.2 – Générations des RCSF.**

### 1.3.3 Architecture d'un Réseau de capteur sans fil

Un réseau de capteurs est constitué essentiellement de : plusieurs nœuds capteurs, un ou plusieurs nœuds puits et un centre de traitement des données.

- **Nœuds:** sont des capteurs, chargés de relever et de router les informations de la zone couverte vers le point de collecte.

L'énergie d'un capteur est généralement limitée puisqu'ils sont alimentés par des piles

- **Puits (Sink):** est une station de base ou un point de collecte des données capturées. Il doit être toujours actif puisque l'arrivée des informations est aléatoire. C'est pourquoi son énergie doit être illimitée.

On peut trouver deux sinks ou plus sur un même réseau de capteurs.

- **Centre de traitement des données:** c'est le centre vers lequel les données collectées par le sink sont envoyées. Ce centre a le rôle de regrouper les données issues des nœuds et les traiter de façon à en extraire de l'information utile exploitable.

Le centre de traitement peut être éloigné du sink, alors les données doivent être transférées à travers un autre réseau, c'est pourquoi on introduit une passerelle entre le sink et le réseau de transfert pour adapter le type de données au type du canal. [3]

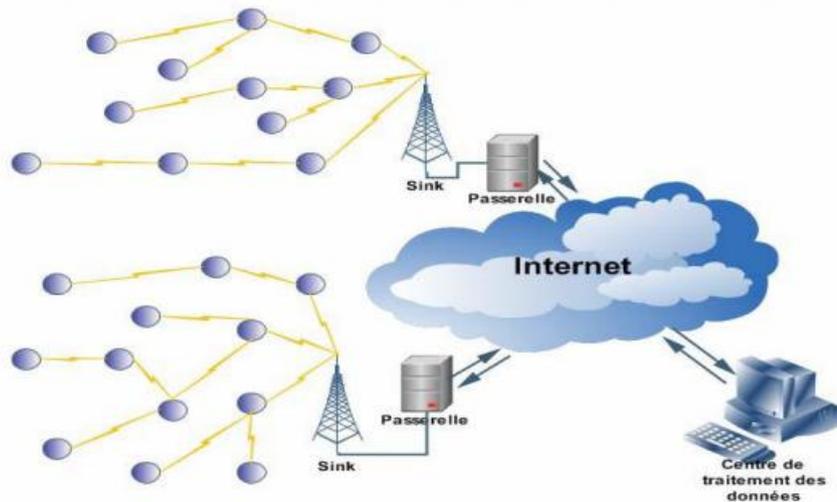


Figure 1.4– Réseau de capteurs sans fil. [3]

### 1.3.4 Les types d'architectures des RCSF

#### 1. Les réseaux de capteur sans fil plat :

Est un réseau homogène, où tous les nœuds ont le même rôle, disposent des mêmes capacités dans la communication et captage d'informations... ils peuvent communiquer entre eux sans passer par un nœud particulier ou une passerelle Seul un nœud sink qui' il joue le rôle de passerelle chargée à transmettre les informations collectées à l'utilisateur. [2]

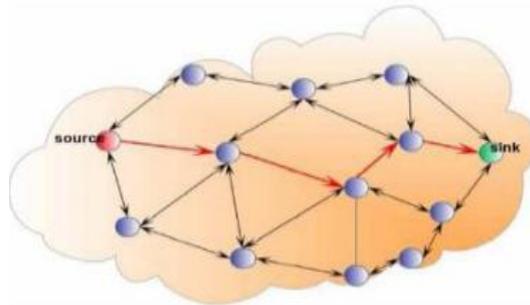


Figure 1.5– Architecture Plat des RCSF.

#### 2. Les réseaux de capteurs sans fil hiérarchique :

C'est un réseau hétérogène, pour augmenter la scalabilité du système, en divisant les nœuds en plusieurs niveaux de responsabilité. Un nœud central sink c'est le niveau supérieur de la hiérarchie. Plusieurs nœuds (capteurs) qui sont le deuxième niveau hiérarchie. Plusieurs autres nœuds de niveau plus bas dans la hiérarchie (troisième niveau). [2][3]

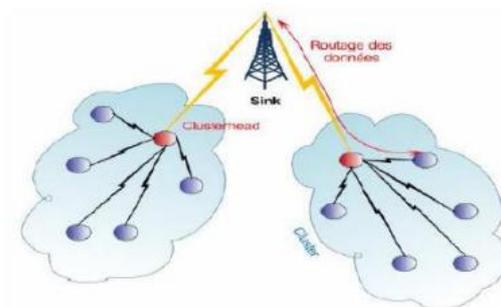


Figure 1.6– Architecture hiérarchique des RCSF.

### 3. Les réseaux de capteurs sans fil basée Localisation

Les protocoles à topologie basée localisation supposent que :

Le réseau est partitionné en plusieurs zones de localisation, chaque zone a son identifiant. Chaque nœud a un identifiant EUI (End-system Unique Identifier) et enregistre dynamiquement l'identifiant de la zone à laquelle il appartient temporairement. L'information temporaire de localisation appelée LDA (Location Dependent Address) qui est un triplé de coordonnées géographiques (longitude, latitude, altitude) obtenues, par exemple, au moyen d'un GPS avec une précision dépendant du type de l'application. Une telle topologie exige l'implémentation d'un algorithme de gestion de localisation qui permet aux nœuds de déterminer les endroits approximatifs des autres nœuds. Ce type de topologie est mieux adapté aux réseaux avec une forte mobilité Avant d'envoyer ses données à un nœud destination, le nœud source utilise un mécanisme pour déterminer la localisation de la destination puis inclus l'identifiant de zone de localisation et du nœud destination dans l'entête du paquet à envoyer.

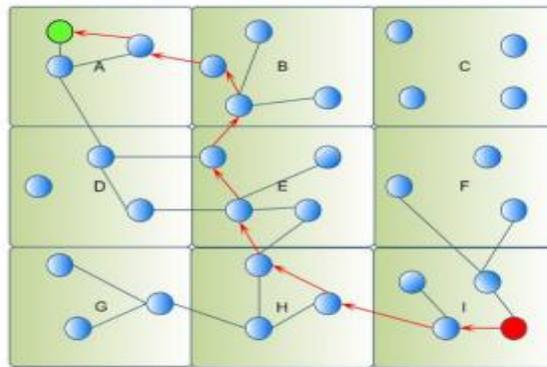


Figure 1.7– Architecture Basée Localisation des RCSF.

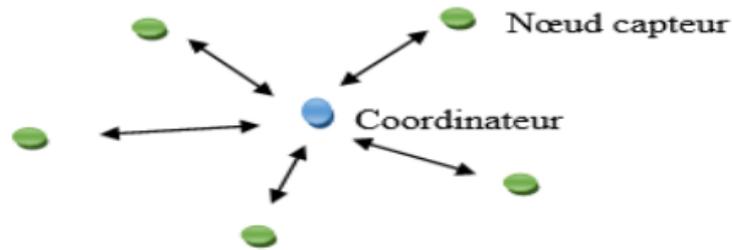
#### 1.3.5 Différentes topologies dans les RCSF

En général il existe trois types de topologies : la topologie en étoile, la topologie maillée et la topologie cluster-tree.

##### ➤ Topologie en étoile :

Dans cette topologie on a un nœud central appelé coordinateur ou sinket des nœuds capteurs sans fil. La station de base peut envoyer ou recevoir un message à un certain nombre de nœuds. Ces nœuds peuvent seulement recevoir ou envoyer un message directement vers le coordinateur, il ne leur est pas permis de s'échanger des messages.

Le nœud central a la responsabilité de contrôler et de coordonner les nœuds capteurs qui communiquent exclusivement avec lui. L'avantage de cette topologie est sa simplicité, sa capacité à minimiser la consommation d'énergie des nœuds et la minimisation de latence de la communication entre les nœuds et la station de base. L'inconvénient est que la station de base n'est pas robuste puisque tout le réseau est géré par un seul nœud. [1][2]

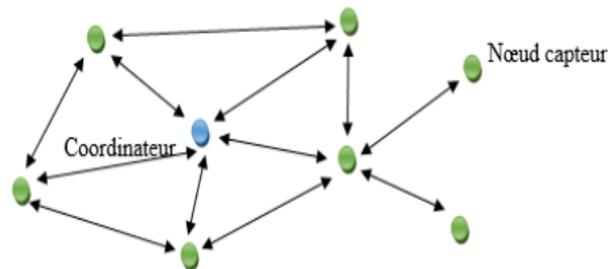


**Figure 1.8-Topologie en étoile.**

➤ **Topologie maillée :**

Dans cette topologie on a un ensemble de noeuds reliés les uns aux autres et agissant comme des routeurs. Chacun des noeuds transmet ses données à son voisin (les nœuds peuvent communiquer entre eux sans contrôle centralisé ni point d'accès) après avoir établi le meilleur chemin pour véhiculer les données à travers le réseau jusqu'au puits (coordinateur ou sink), L'avantage de cette topologie est de maintenir le fonctionnement du réseau en cas de panne d'un nœud, la robustesse et flexibilité par rapport à la variation du nombre de noeuds. Quand il n'y a pas de transfert de données, les noeuds se mettent en mode veille pour économiser de l'énergie.

L'inconvénient : Une consommation d'énergie plus importante est induite par la communication multi-sauts. Une latence est créée par le passage des messages des nœuds par plusieurs autres avant d'arriver à la station de base. [1]



**Figure 1.9-Topologie maillée.**

➤ **Topologie arbre et cluster-tree :**

Cette topologie est partitionnée en groupes appelés "clusters". Un cluster est constitué d'un noeud particulier appelé "cluster-head" ou "tête de cluster" et d'autres noeuds qui communiquent qu'avec leur "tête de cluster", ensuite ce dernier est chargé de faire suivre les messages reçus vers le puits du réseau. Cette topologie est alors hiérarchisée selon le rôle des éléments du réseau (coordinateur, têtes de cluster, noeud). Il peut y avoir un changement de hiérarchie entre les nœuds et les têtes de cluster en fonction de l'énergie disponible ou la position géographique de chaque nœud. [1]

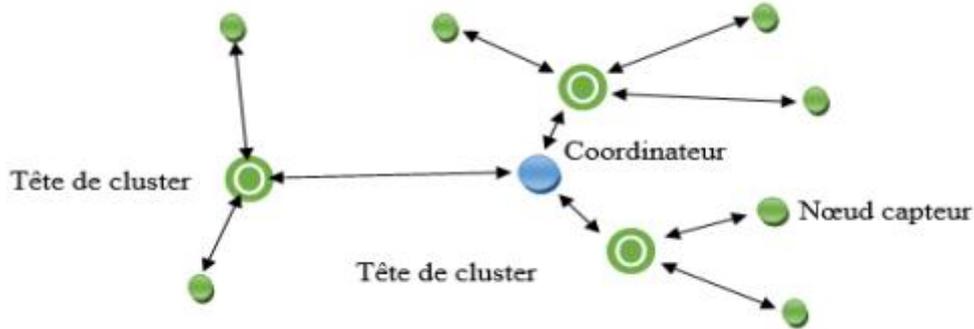


Figure 1.10–Topologie cluster-tree.

### 1.3.6 La pile protocolaire

La pile protocolaire utilisée par les réseaux de capteurs, Elle est combinée routage et gestion d'énergie et intègre les données avec les protocoles réseau, elle est composée de cinq couches : une couche application, une couche transport, une couche réseau, une couche liaison de données et une couche physique. Cette pile est également caractérisée par trois niveaux qui intègrent dans les protocoles des différentes couches la prise en compte de l'énergie consommée, de la mobilité des nœuds, et de la gestion de la distribution des tâches sur les différents nœuds du réseau.

- **La couche physique** : Spécifications des caractéristiques matérielles, des fréquences porteuses, la détection du signal, la modulation/ démodulation et le cryptage/décryptage des informations, etc...
- **La couche liaison de données** : Spécifie comment les données sont expédiées entre deux nœuds/routeurs dans une distance d'un saut. Elle est responsable de :
  - Multiplexer le flux de données.
  - Détecter et verrouiller les trames de données.
  - Contrôler l'accès au support de transmission (Media Access Control).
  - Contrôler les erreurs.
  - Et d'assurer une connexion fiable (point-à-point où point-à- multipoints) selon la topologie du réseau de capteurs.

Parmi les protocoles de liaison de données, nous citons : TRAMA, SMACS et EAR.

- **La couche réseau** : Gère les échanges (et éventuellement les connexions) au travers du RCSF. Le but principal est de trouver une route et une transmission fiable des données, Sachant que le positionnement des nœuds étant aléatoire et dense, les protocoles de routage traditionnels deviennent inadéquats. Pour cela, la communication multi-sauts est la mieux adaptée. Cette couche prend en charge la découverte du voisinage, l'allocation des ressources et le routage.

Parmi les protocoles de réseau, nous citons: TEEN, APTEEN et LEACH.

- **La couche transport** : Cette couche est chargée du transport des données, elle est essentiellement présente pour constituer une interface entre la couche application et la couche réseau. Ses principaux objectifs sont :
  - Multiplexer et démultiplexeur les messages entre les applications et la couche réseau.

- Contrôler les données à haut niveau.
- Réguler la quantité des données injectées dans le réseau. Le rôle de cette couche intervient essentiellement lorsqu'on va accéder à partir de notre RCSF vers un autre RCSF ou vers Internet.

Parmi les protocoles de transport, nous citons : UDP et UDP-Like.

➤ **La couche application** : Cette couche assure l'interface avec les applications.

Il s'agit donc du niveau le plus proche des utilisateurs, géré directement par les logiciels.

Elle est constituée l'ensemble des applications implémentées sur un réseau de capteurs.

Ces applications devraient fournir des mécanismes permettant à l'utilisateur d'interagir avec le réseau de capteurs à travers différentes interfaces, et éventuellement, par l'intermédiaire d'un réseau étendu (par exemple : Internet). Cette couche est responsable par exemple sur la collecte, le codage, l'agrégation et la compression des données collectées.

Parmi les protocoles d'application, nous citons : SMP et TADAP.

En plus les 3 niveaux (plans) :

- **Le niveau de gestion d'énergie** : Chargé de contrôler la manière dont un nœud utilise son énergie.
- **Le niveau de gestion des tâches** : Assure l'équilibrage de la distribution des tâches sur les différents nœuds pour accomplir un travail coopératif.
- **Le niveau de gestion de la mobilité** : Détecte et enregistre tous les mouvements des nœuds capteurs. [2][5]

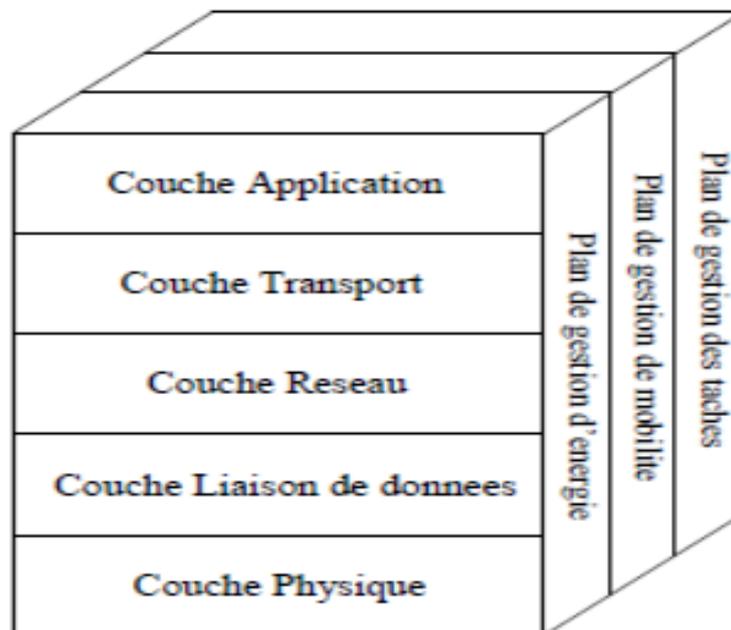


Figure 1.11– La pile protocolaire.

- **Les taches de la pile protocolaire :**

Couche	Taches
Physique	Codage, filtrage, contrôle de puissance, sélection de canaux, émission et réception des données sur le support physique, modulation.
Liaison de données	Méthode d'accès au canal, retransmission des données, validation des trames, mécanismes de cryptage, contrôle de puissance.
Réseau	Routage, découverte des voisins, sécurité du transfert de données, gestion des tables de routage, allocation de ressources.
Transport	Contrôle de flux, retransmission des données.
Application	Codage, compression, fonction d'agrégation, collecte.

**Tableau 1.3 – Les taches de la pile protocolaire.**

### 1.3.7 Les différents types de capteurs :

Un réseau de capteur sans fil(RCSF)contient différents types de nœuds, pour optimiser certains paramètres par exemple la durée de vie du réseau ou le délai de livraison des données, certains travaux se sont focalisés sur l'architecture (plat, hiérarchique etbasé localisation) des RCSF. Ces architectures définissent souvent les rôles joués par les nœuds dans un RCSF. Nous distinguons principalement 3 rôles à savoir : [7]

- **Nœud source (NS)**son rôle est de détecter les phénomènes physiques ou physiologiques qui se produisent dans son environnement afin de les transmettre, directement ou via multiples sauts, à un utilisateur final. C'est en fait un nœud capteur.
- **Nœud relais (NR)** pour d'agréger et de retransmettre les mesures provenant des NS afin que celles-ci parviennent à un utilisateur final. Dans une architecture à plat, les NS sont considérés comme NR. Dans une architecture à 2 niveaux, un nœud passerelle joue le rôle de NR pour un ou plusieurs nœuds sources. Dans ce type de configuration réseau, la capacité de transmission du NR est supposée généralement plus grande que celle du NS.
- **Nœud collecteur (NC)** ont pour jouer le rôle de collecter les mesures provenant des nœuds sources et éventuellement de les agréger. Généralement, un "Cluster-Head " ou chef de cluster est utilisé comme NC dans une architecture hiérarchique où les NS sont partitionnés en plusieurs groupes.

### 1.3.8 Les types de communications :

Les types de communications utilisés dans les RCSF sont listés ci-dessous :

- **Unicast :**

Le type Unicast est utilisé pour échanger des informations entre les nœuds (deux par deux) sur le réseau.

➤ **Broadcast :**

Ce type est utilisé pour transmettre les messages de notification à tous les nœuds du réseau, ces messages peuvent encapsuler des informations concernant les requêtes des nœuds, l'ordonnancement de traitement des requêtes des nœuds, des mises à jour des programmes des nœuds, ou des commandes de contrôle de tout le système. Il est utilisé généralement par le nœud puits « Sink ».

➤ **Convergecast :**

Ce type de communication est appelé « Convergecast », où plusieurs nœuds communiquent avec un nœud spécifique qui peut être un gestionnaire de grappe, un centre de fusion de données ou une station de base. Les nœuds ont besoin de transmettre la détection au centre d'informations pour la prise de décision.

➤ **Local gossip :**

Dans certaines situations et suivant la nature de l'application, les nœuds communiquent localement entre eux l'événement détecté. Ce type de communication est appelé " local gossip ".

➤ **Multicast :**

Il permet une communication entre un nœud et un groupe de nœuds. Ce type de communication est utilisé dans les protocoles qui incluent le principe du clustering où, le nœud chef de groupe « CH » s'intéresse à communiquer avec un groupe de nœuds. [9]

### 1.3.9 La collection d'information :

Il y a deux méthodes pour collecter les informations d'un réseau de capteurs :

➤ **A la demande :**

Lorsque l'on souhaite avoir l'état de la zone de couverture à un moment, le puits émet des broadcasts vers toute la zone pour que les capteurs remontent leur dernier relevé vers le puits.

➤ **Suite à un événement :**

Un événement se produit en un point de la zone de couverture (changement brusque de température, mouvement...), les capteurs situés à proximité remontent alors les informations relevées et les acheminent jusqu'au puits.

### 1.3.10 Les modèles des réseaux de capteurs sans fil (RCSF) :

La transmission de données dans les réseaux de capteurs peut se faire suivant plusieurs modèles dont on distingue trois essentiels : [3]

➤ **Le modèle driven-event :**

Au lieu d'avoir un nœud émetteur et un autre récepteur de l'information, on trouve un nœud récepteur (le nœud de contrôle « sink ») et un groupe de nœuds capteurs, se trouvant proche de l'événement, qui sont tous des émetteurs de la même information. L'avantage pour ce modèle, repose essentiellement sur la détection de l'événement et la rapidité des prises des réactions nécessaires pour assurer l'aspect temps réel des applications.

➤ **Le modèle query-driven :**

Ce modèle est semblable au modèle driven event sauf que la collecte des informations sur l'état de l'environnement est initiée par des interrogations envoyées par le « sink », on peut

utiliser ce modèle pour contrôler et reconfigurer les nœuds. Par exemple, le « sink » peut envoyer des commandes au lieu des interrogations pour modifier le programme d'un nœud capteur, son taux de trafic et son rôle. Seul le nœud capteur jouant le rôle de « sink » est autorisé d'émettre des demandes d'interrogations ou des commandes et ce pour assurer l'ordre et la hiérarchie de réseau de capteur.

➤ **Le modèle continu :**

Dans ce modèle, les nœuds capteurs envoient les informations d'une manière continue au nœud « sink » suivant un volume de trafic prédéterminé.

### 1.3.11 Les domaines d'application

Les RCSFs sont capables de contrôler un grand nombre de conditions physiques telles que la température, l'humidité, la lumière, la pression, le mouvement, la composition du sol et le niveau de bruit. Grâce à leur fiabilité, auto-organisation et flexibilité.

Les RCSF peuvent avoir beaucoup d'applications, Parmi elles, nous citons :

➤ **Domaine militaire :**

La surveillance de territoire ou de frontière à des fins militaires a été la première application des réseaux de capteurs. Dans le même objectif militaire, des nœuds capteurs peuvent être dispersés sur un terrain adverse afin de l'analyser avant d'y envoyer des troupes (détection d'agents chimique, biologiques, ou radioactifs). Les RCSFs peuvent être montés sur des véhicules sans pilote robotique, des chars, des avions de combat, sous-marins et des missiles pour les acheminer autour des obstacles, les guider à la position exacte et les amener à coordonner entre eux pour remplir des attaques ou des défenses plus efficaces.[8]



**Figure 1.12-Capteurs dans les applications militaires.**

➤ **Domaine environnemental :**

Les réseaux de capteurs peuvent être utilisés pour surveiller les changements environnementaux accidentels ou naturels. Cela comprend la surveillance des écosystèmes, la détection d'inondation, l'air et des eaux usées, la composition du sol, la détection des feux de forêt et l'exploration des réserves minérales. Les auteurs décrivent la mise en œuvre d'un réseau de capteurs sur le volcan « reventator » dans partie occidental de l'Amazonie en Equateur. Les données recueillies au cours des déplacements sont transmises à un satellite tandis que des nœuds sont toujours à la surface. La durée de vie espérée des nœuds est d'environ 45 ans. Ils peuvent être aussi utilisés pour surveiller des animaux.



**Figure 1.13-Capteurs pour l'environnement, sécurité routière.**

➤ **Domaine de la sécurité :**

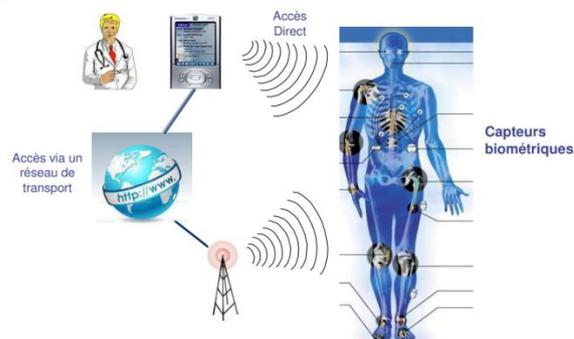
Un réseau de capteurs acoustiques sans fil peut être utilisé pour détecter et suivre des cibles souhaitées dans une zone de sécurité. Les RCSFs peuvent être déployés dans des bâtiments, des zones résidentielles, des aéroports, des gares, etc. pour identifier les intrus et faire des rapports à un centre de commandement immédiatement afin que les actions de suivi peuvent être lancées sans délai. [6]



**Figure 1.14-Capteurs dans les infrastructures.**

➤ **Domaine médical :**

Les capteurs peuvent être implantés dans ou sur le corps pour surveiller un patient et son état de santé. Les RCSFs peuvent aussi faciliter le diagnostic de quelques maladies en effectuant des mesures physiologiques telles que la tension artérielle et battements du cœur.



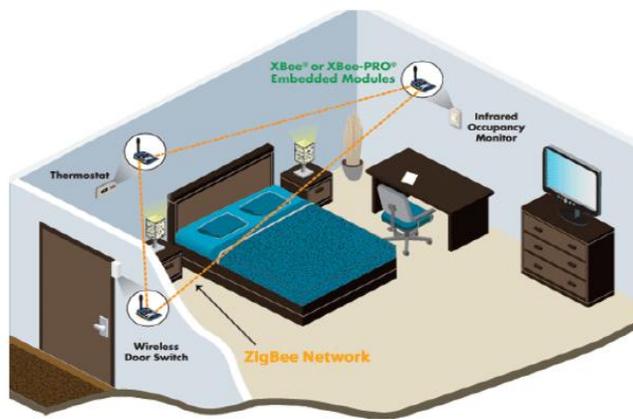
**Figure 1.15 Capteurs dans le domaine médical.**

➤ **Domaine commercial :**

Il est possible d'intégrer des capteurs dans les processus de stockage et de livraison. Pour les entreprises manufacturières, les réseaux de capteurs permettront de suivre le procédé de production à partir des matières premières jusqu'au produit final. Grâce à ces réseaux, les entreprises pourraient offrir une meilleure qualité de service.

➤ **Domaine de la domotique :**

Les réseaux de capteurs peuvent être utilisés pour détecter les intrusions, capter la température, et toute information utile pour assurer confort et sécurité dans une maison. Les smart homes, c'est-à-dire ces maisons intelligentes (des maisons avec des capteurs embarqués) vont faciliter les activités domestiques quotidiennes telles que l'automatisation de l'activation / l'extinction de la lumière qui existe déjà dans certains garages ou encore la mise en marche automatique de la télévision, la climatisation ou le chauffage lorsqu'il y a présence dans le salon.



**Figure 1.16-Capteurs dans la domotique.**

➤ **Domaine génie civil :**

Les capteurs peuvent être utilisés pour contrôler les vibrations susceptibles d'endommager la structure d'un bâtiment, d'un pont ou d'un barrage. Ils peuvent être aussi utilisés pour surveiller l'état de grandes structures comme des immeubles administratifs.

➤ **Domaine sportif :**

Les réseaux de capteurs sont utilisés dans le domaine sportif, pour savoir les systèmes de surveillance, les systèmes de calcul de trajectoires (comme dans le tennis), systèmes de détection d'erreurs d'arbitrage (comme dans le football indiquent si la balle a franchi la ligne de but).

➤ **Domaine agriculture :**

Les réseaux de capteurs mesurent l'humidité du sol ou d'autres paramètres sur la culture afin d'irriguer ou traiter avec précision les seuls endroits nécessaires.



Figure 1.17-Capteurs dans l'agriculture.

➤ **Domaine l'industrie :**

Gestion des stocks. Les capteurs peuvent remplacer les puces RFID pour contrôler les stocks de produits, contrôler les flux, surveiller des équipements en urbanisme, les capteurs peuvent détecter les déplacements, compter les voitures, ...

**1.3.12 Les principales caractéristiques des réseaux sans fil :**

Un réseau de capteurs présente les caractéristiques suivantes : [3]

- **Absence d'infrastructure** : les réseaux Ad-hoc en général, et les réseaux de capteurs en particulier se distinguent des autres réseaux par la propriété d'absence d'infrastructure préexistante et de tout genre d'administration centralisée.
- **Taille importante** : un réseau de capteurs peut contenir des milliers de nœuds.
- **Interférences** : les liens radio ne sont pas isolés, deux transmissions simultanées sur une même fréquence, ou utilisant des fréquences proches, peuvent interférer.
- **Topologie dynamique** : les capteurs peuvent être attachés à des objets mobiles qui se déplacent d'une façon libre et arbitraire rendant ainsi la topologie du réseau fréquemment changeante.
- **Sécurité physique limitée** : les réseaux de capteurs sans fil sont plus touchés par le paramètre de sécurité que les réseaux filaires classiques. Cela se justifie par les contraintes et limitations physiques qui font que le contrôle des données transférées doit être minimisé.
- **Bande passante limitée** : une des caractéristiques primordiales des réseaux basés sur la communication sans fil est l'utilisation d'un médium de communication partagé. Ce partage fait que la bande passante réservée à un nœud est limitée
- **Contrainte d'énergie, de stockage et de calcul** : la caractéristique la plus critique dans les réseaux de capteurs est la modestie de ses ressources énergétiques car chaque capteur du réseau possède de faibles ressources en termes d'énergie (batterie). Afin de prolonger la durée de vie du réseau, une minimisation des dépenses énergétiques est exigée chez chaque nœud. Ainsi, la capacité de stockage et la puissance de calcul sont limitées dans un capteur.
- **Les capteurs sont limités** : en énergie (Piles AAA), capacités de calcul (4MHz), et en mémoire.
- **Les réseaux de capteurs sont non fiables** : en tout moment, les capteurs peuvent être défaillants.

### 1.3.13 Contraintes de conception des RCSF :

Les principaux facteurs et contraintes influençant l'architecture des réseaux de capteurs peuvent être résumés comme suit : [3]

➤ **La tolérance de fautes :**

Parfois, certains nœuds peuvent entraîner une défaillance de leur fonction en raison d'un manque de force « problème physique ou interférence » Ces problèmes n'affectent pas le reste du réseau et sont connus sous le nom de principe de tolérance d'erreur.

➤ **L'échelle :**

L'occurrence de nombreux transferts entre streptocoques fait référence au grand nombre de nœuds pouvant atteindre

Pour un million, cela nécessite que le puits "sink" soit doté de beaucoup de mémoire pour stocker les informations reçues

➤ **Les coûts de production :**

Les réseaux de capteurs sont souvent constitués d'un grand nombre de nœuds. Le prix de l'appel est essentiel pour concurrencer le réseau de surveillance traditionnel. L'inflammation ne coûte souvent pas plus de 1 \$. Par exemple, Bluetooth, déjà connu sous le nom de système à faible coût, coûte environ 10 dollars.

➤ **L'environnement :**

Les capteurs sont déployés collectivement dans des zones à surveiller, telles que des champs de bataille situés en dehors des lignes ennemies, dans de grosses machines, au fond de l'océan, dans des champs contaminés biologiquement ou chimiquement. Ils doivent donc avoir la capacité et la capacité de fonctionner sans contrôle dans des zones géographiques éloignées.

➤ **La topologie de réseau :**

Un grand nombre de contrats de maintenance doivent être déployés. Cette maintenance comporte trois étapes: déploiement, post-déploiement (les capteurs peuvent se déplacer, ne plus fonctionner, ...), redistribution de nœuds supplémentaires.

➤ **Les contraintes matérielles:**

La principale contrainte physique est la taille du capteur. Les autres limitations sont la consommation d'énergie et doivent être moins pour que le réseau maintienne sa survie le plus longtemps possible, en s'adaptant à différents environnements (température élevée, eau ...), caractérisé par l'indépendance et la résistance souvent déployées dans des environnements hostiles.

➤ **Les médias de transmission :**

Dans un réseau de capteurs, les nœuds sont connectés par une structure sans fil. Pour pouvoir opérer sur ces réseaux dans le monde entier, les moyens de transmission doivent être unifiés. Le Bluetooth infrarouge et le ZigBee sont utilisés le plus souvent par infrarouge (sans licence, anti-ingérence, peu coûteux).

➤ **La consommation d'énergie :**

La taille du capteur est limitée à l'énergie (<1.2V). Dans de nombreux cas, il est souvent impossible de remplacer la batterie. Cela signifie que la vie du capteur dépend fortement de la vie de la batterie. Dans chaque réseau de capteurs (plusieurs bonds), la tâche de chaque nœud est de collecter des données et d'envoyer / d'envoyer des valeurs. Certains nœuds nécessitent une modification de la topologie du réseau et de la redirection de paquets. Tous ces processus consomment beaucoup d'énergie et la recherche porte aujourd'hui sur des solutions et des moyens de réduire cette consommation.

### 1.3.14 Notion d'énergie

Il existe plusieurs manières de provoquer un dysfonctionnement total dans un réseau de capteurs sans fil : attaque externe des nœuds, perte de connexions avec la station de base, dommages physiques, interférences, etc. Tout cela est classé en tant que causes externes. Il peut se détruire en réduisant sa capacité énergétique. En fait, l'âge du réseau dépend principalement de l'âge de la tenue de son capteur. Dans le même temps, la durée de vie d'un nœud de capteur dépend fortement de la durée de vie de sa source d'alimentation. Actuellement, la plupart des études se concentrent sur la recherche de solutions alternatives pour alimenter les capteurs en sources d'énergie telles que la conversion de l'énergie photovoltaïque en énergie électrique et l'intégration d'autres sources telles que les énergies renouvelables (modules solaires), etc. Un capteur de travail est déployé pendant des années, le capteur utilise son énergie dans les opérations suivantes: [3]

#### **Energie de capture :**

C'est l'énergie consommée pour accomplir les tâches suivantes : traitement et conversion analogique / numérique des signaux, activation de la sonde de capture... etc.

#### **Energie de traitement :**

C'est l'énergie dissipée dans l'exécution des opérations relatives aux traitements effectués sur les données reçues des autres unités.

#### **Energie de communication :**

Absorbe la plus grande partie de l'énergie totale consommée par un capteur. Son utilité est d'envoyer et de recevoir des informations. Il est déterminé par deux facteurs principaux: la quantité de données échangées et la force du signal, qui dépend de la distance entre les entités qui se connectent.

Actuellement, la plupart des études se penchent sur la manière d'améliorer la consommation d'énergie est devenu le sujet de nombreuses contributions à la recherche.

### 1.3.15 Systèmes d'exploitation pour les RCSF

Le système d'exploitation est joué le rôle de l'intermédiaire entre l'utilisateur et les périphériques matériels, Parmi les systèmes d'exploitation développés pour les réseaux de capteurs sans fil, nous citons :

- **TinyOS:** est un système d'exploitation intégré, modulaire, destiné aux réseaux de capteurs miniatures. Cette plate-forme logicielle ouverte et une série d'outils développés par l'Université de Berkeley est enrichie par une multitude d'utilisateurs. En effet, TinyOS est le plus répandu des OS pour les réseaux de capteurs sans-fil. Il est utilisé dans les plus grands projets de recherches sur le sujet (plus de 10.000 téléchargements de la nouvelle version).
- **LiteOS:** est un système d'exploitation léger en temps réel. C'est un système d'exploitation open source pour les terminaux intelligents IoT. Il supporte les normes BSD 3, ARM, X86, RISC-V et les microcontrôleurs de différentes architectures. Il prend en charge plus de 50 cartes de développement.
- **MantisOS:** Le système MANTIS Multimodal pour les capteurs sans fil in-situ Networks fournit un nouveau système d'exploitation intégré multithread pour les réseaux de capteurs sans fil.
- **Contiki:** est un système d'exploitation léger et flexible pour capteurs miniatures en réseau. Ces dernières années, le monde scientifique a porté une attention importante aux réseaux de capteurs sans fil. La miniaturisation des capteurs et leur prix

relativement faible, permettent d'imaginer des applications très variées dans les domaines scientifiques, militaires, industriels et domotiques.

#### **1.3.16 Principaux défis des réseaux de capteurs sans fil :**

- La consommation d'énergie.
- La densité (passage à l'échelle).
- La dynamique du réseau.
- La qualité de service.

### **1.4 Conclusion**

Dans ce chapitre nous avons parcouru les réseaux de capteurs sans fil, nous avons donné une vue générale en décrivant leurs architectures, leurs domaines d'applications, leurs modèles et les topologies, puis nous avons parlé sur les types de communications, ensuite nous avons décrit les caractéristiques et les défis et enfin nous avons résumé les Contraintes de conception. Le RCSF est un domaine de recherche très répondu, et devient de plus en plus vaste et de nouvelles tendances de recherches sont apparues surtout de ce qui concerne l'optimisation d'énergie et l'application de ces réseaux dans domaines spécialisés.

# Chapitre 02

**Principaux concepts de surveillance des réseaux de capteurs sans fils**

## 2.1 Introduction

La surveillance des réseaux de capteurs sans fil consiste à détecter les défauts et les pannes ensuite générer des solutions en temps réel.

Certains capteurs peuvent être bloqués ou tomber en panne à cause d'un manque d'énergie, d'un dégât matériel ou d'une interférence environnementale. La panne d'un capteur ne doit pas affecter le fonctionnement global de son réseau. C'est le problème de fiabilité ou de tolérance aux pannes. La tolérance aux pannes a pour objectif de maintenir les fonctionnalités du réseau sans interruption due à une panne de certains capteurs.

On présentera dans ce chapitre les différentes catégories de pannes qui peuvent survenir durant le cycle de vie d'un réseau de capteurs sans fil, leurs causes ainsi que les approches et solutions proposées pour la tolérance aux pannes, et quelques aspects de surveillance d'un RCSF.

## 2.2 Définition d'un système de surveillance

La surveillance est un dispositif utilisé pour analyser et contrôler l'état du système, ou même réagir sur un cas particulier, Elle consiste à détecter les anomalies, les pannes et classer les défaillances, puis diagnostiquer pour pouvoir localiser les éléments défaillants et déterminer les causes, ensuite générer des solutions en temps réel pour garder la longévité du réseau. [10]

## 2.3 Type de systèmes de surveillance selon le type de nœuds

Les systèmes de surveillance sont classifiés selon le type des nœuds en trois classes à savoir les systèmes de surveillance à base de capteurs scalaires, à base de capteurs multimédia ou une hybridation des deux types.

### 2.3.1 Systèmes de surveillance à base de capteurs scalaires

Les systèmes de surveillance à base de capteurs scalaires sont des systèmes qui prennent comme moyen de détection des capteurs de mouvements, de température ou d'autres grandeurs physiques,

Un nombre qui mesure une température, une masse, ou une hauteur est un scalaire.

On trouve ce type de systèmes de surveillance dans plusieurs domaines par exemple dans la détection de l'influence de l'activité sismique sur un pont ou placement des capteurs de distance pour mesurer les fissures et alerter en cas de risques majeurs. [10]

### 2.3.2 Systèmes de surveillance à base de capteurs multimédia

L'un des systèmes de surveillance multimédia le plus connu est la vidéosurveillance

### 2.3.3 Systèmes de surveillance à base de capteurs hybrides

C'est une hybridation entre les capteurs à grandeurs scalaires et des capteurs multimédia qui peuvent communiquer entre eux, c'est les nouveaux systèmes récents qui utilisent cette approche, car elle garantit à la fois la détection rapide fournie par les capteurs à grandeurs scalaires puisque les données transférées sont simples (comme la position d'un intrus détecté par un capteur de mouvement) et aussi une détection de plus de détails comme la vidéo par des capteurs multimédias. [10]

## 2.4 Les pannes dans les réseaux de capteurs

### 2.4.1 Définition d'une panne

Une panne du système se produit lorsque son comportement devient inconsistant et ne fournit pas le résultat voulu. La panne est une conséquence d'une ou plusieurs erreurs. Une

erreur représente un état invalide du système dû à une faute (défaut). La faute est donc la première cause de l'erreur, cette dernière provoque la faille du système. [6]

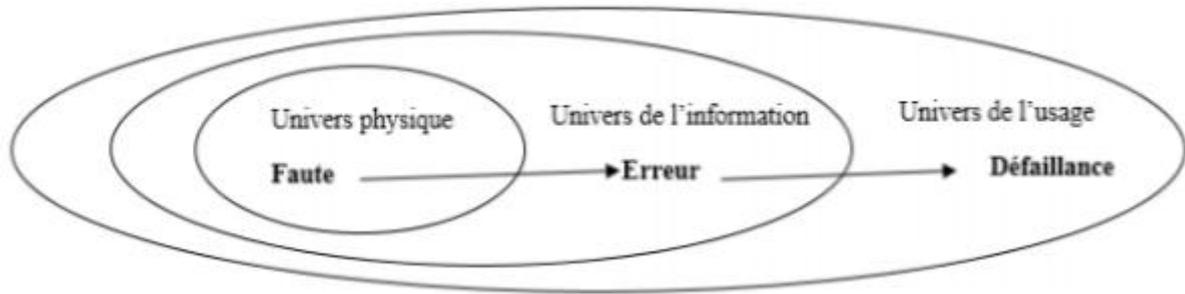


Figure 2.1- De la faute à la défaillance.

2.4.2 Classification des pannes

Les pannes dans un réseau de capteurs sans fil, sont classées selon trois critères : durée, cause et comportement. La classification classique est faite selon la nature résultante de la panne ce qui donne deux types de pannes :

- Pannes catastrophiques : elles sont inacceptables. Par exemple, les grandeurs pathologiques d'un patient.
- Pannes non catastrophiques : elles sont acceptables. La collecte de la température dans un environnement.

Le schéma suivant montre une classification générale selon la durée, la cause et le comportement d'une panne :

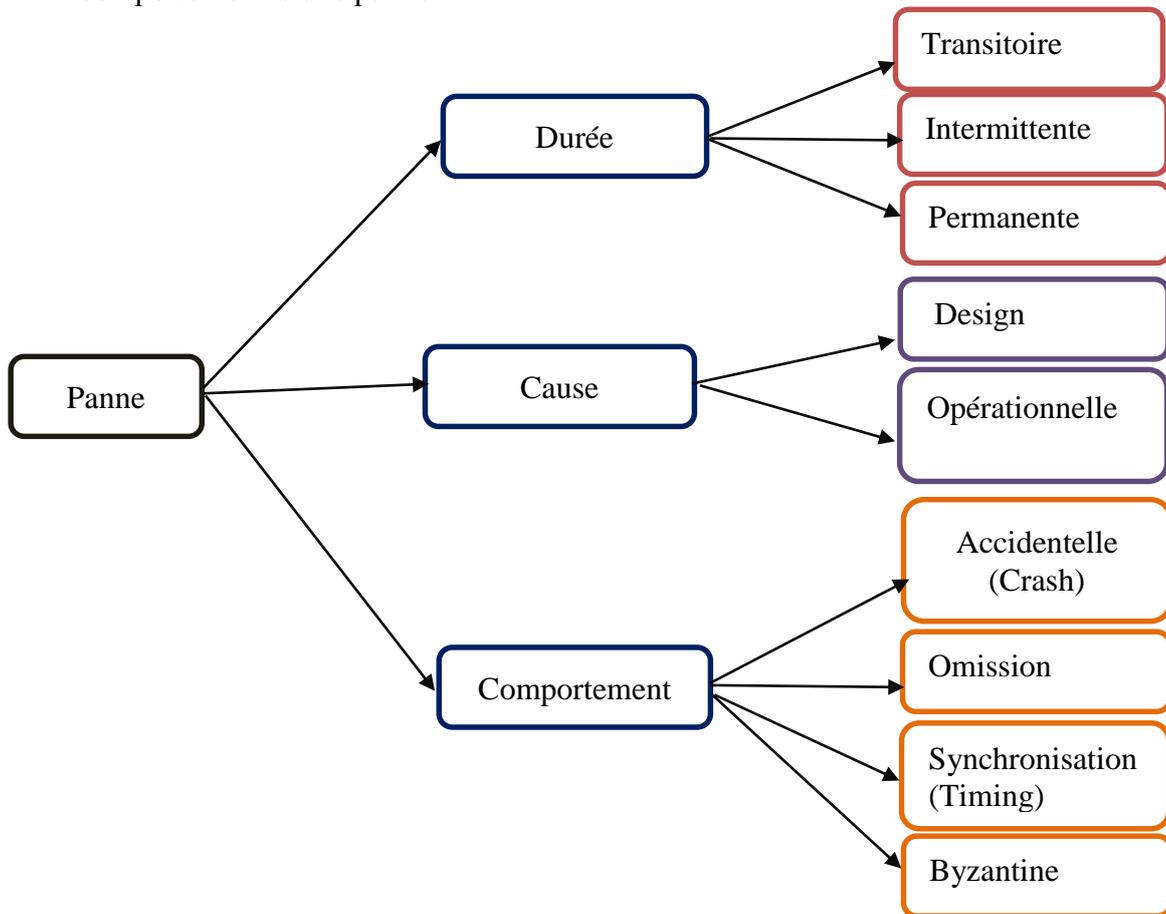


Figure 2.2- Classification des pannes.

➤ **Pannes selon durée**

- **Transitoire** : Conséquence d'un impact environnemental temporaire, elle peut éventuellement disparaître sans aucune intervention. (elle se produit de manière isolée)
- **Intermittente** : Variante de la panne transitoire, elle se produit occasionnellement et de façon imprévisible. Elle est généralement due à l'instabilité de certaines caractéristiques matérielles ou à l'exécution du programme dans un espace particulier de l'environnement. (elle se produit aléatoirement plusieurs fois.)
- **Permanente** : continue et stable dans le temps, la panne permanente persiste tant qu'il n'y a pas d'intervention externe pour l'éliminer. (elle persiste dès qu'elle apparaît jusqu'à la réparation.)  
Un changement physique dans un composant provoque une panne matérielle permanente.

➤ **Pannes selon la cause**

- **Panne de design** :

Due à une mauvaise structuration du réseau ou du composant en particulier. En pratique, ce genre de panne ne devrait pas exister grâce aux tests et simulations avant la réalisation finale du réseau.

- **Panne opérationnelle** :

Qui se produit durant le fonctionnement du système. Elle est généralement due aux causes physiques.

➤ **Pannes selon le comportement résultant**

- **Panne accidentelle (Crash)** :

Le composant soit, il s'arrête complètement de fonctionner ou bien continue mais sans retourner à un état stable (valide).

- **Panne d'omission** :

Le composant n'est plus capable d'améliorer son service (échec total).

- **Panne de synchronisation (Timing)** :

Le composant effectue son traitement mais fournit le résultat en retard.

- **Panne Byzantine**:

Cette panne est de nature arbitraire ; le comportement du composant est donc imprévisible. Dû à des attaques très malicieuses, ce type de pannes est considéré le plus difficile à gérer.

### 2.4.3 Causes de pannes

Les capteurs peuvent subir des pannes et des défaillances dues à de différentes causes et phénomènes qui peuvent être internes ou externes selon leurs origines.

Ces pannes que peut subir un réseau de capteurs peuvent être dues à :

- L'épuisement d'énergie des capteurs.
- Perte de connexion sans fil due à l'épuisement de la batterie d'un capteur.
- Destruction physique (accidentelle ou volontaire).
- Interférences environnementales.

## 2.5 La tolérance aux pannes

Le but de la tolérance aux pannes est d'éviter la faille totale du système malgré la présence de fautes dans un sous ensemble de ses composants élémentaires. La tolérance de panne est d'autant meilleure que le nombre de composants en panne est grand (avec la garantie du bon fonctionnement du système).

La conception d'une procédure pour la tolérance aux pannes dépend de l'architecture et des fonctionnalités du système. Cependant, certaines étapes générales sont exécutées dans la plupart des systèmes



**Figure 2.3- Procédures de tolérance aux pannes.**

### ➤ **Détection d'erreur**

C'est la première phase dans chaque schéma de tolérance aux pannes, dans laquelle on reconnaît qu'un événement inattendu s'est produit. Les techniques de détection de pannes sont généralement classifiées en deux catégories : en ligne et autonome (offline). La détection offline est souvent réalisée à l'aide de programmes de diagnostic qui s'exécutent quand le système est inactif. La détection en ligne vise l'identification de pannes en temps réel et est effectuée simultanément avec l'activité du système.

### ➤ **Détection de la panne**

Cette phase établit des limites des effets de la panne sur une zone particulière afin d'empêcher la contamination des autres régions. En cas de détection d'intrusion, par exemple, l'isolation des composants compromis minimise le risque d'attaque des composants encore fonctionnels.

### ➤ **Recouvrement d'erreur**

C'est la phase dans laquelle on effectue des opérations d'élimination des effets de pannes. Les deux techniques les plus utilisées sont « masquage de panne » et « répétition »

- Masquage de panne : utilise l'information redondante correcte pour éliminer l'impact de l'information erronée.
- Répétition : après que la panne soit détectée, on effectue un nouvel essai pour exécuter une partie du programme, dans l'espoir que la panne soit transitoire.

### ➤ **Traitement de panne**

Dans cette phase, la réparation du composant en panne isolé est effectuée. La procédure de réparation dépend du type de la panne. Les pannes permanentes exigent une substitution du composant avec un autre composant fonctionnel. Le système doit contenir un ensemble d'éléments redondants (ou en état standby) qui servent à remplacer les nœuds en panne. [11]

## **2.5.1 Classification des solutions de tolérance aux pannes**

Dans les RCSFs Les solutions et les approches de tolérance aux pannes peuvent être vues de plusieurs angles différents. De ce fait, un ensemble de critères est défini pour les classifier.

### **2.5.1.1 Classification selon la phase de traitement**

Dans cette classification, l'ensemble des algorithmes sont divisés en deux catégories selon la phase de traitement. Si le traitement est effectué avant la panne, on parle donc d'algorithmes préventifs sinon les algorithmes sont dits curatifs.

- **Algorithme préventif** : implémente des techniques tolérantes aux pannes qui tentent de retarder ou éviter tout type d'erreur afin de garder le réseau fonctionnel le plus longtemps possible. La conservation d'énergie, à titre d'exemple, permet de consommer moins d'énergie

et évite donc une extinction prématurée de la batterie ce qui offre une tolérance aux pannes des nœuds.

- **Algorithme curatif** : utilise une approche optimiste, où le mécanisme de tolérance aux pannes implémenté n'est exécuté qu'après la détection de pannes. Pour cela, plusieurs algorithmes de recouvrement après l'occurrence des pannes sont proposés dans la littérature, par exemple : le recouvrement du chemin de routage, l'élection d'un nouvel puits, etc.

### 2.5.1.2 Classification architecturale

Cette classification traite les différents types de gestion des composants, soit au niveau du capteur individuellement ou bien sur tout le réseau. Nous distinguons trois catégories principales :

#### ➤ Gestion de la batterie :

Cette catégorie est considérée comme une approche préventive, où les protocoles définissent une distribution uniforme pour la dissipation d'énergie entre les différents nœuds capteurs, afin de mieux gérer la consommation d'énergie et augmenter ainsi la durée de vie de tout le réseau. En outre, le mécanisme de mise en veille est une technique de gestion de batterie.

En effet, les protocoles déterminent des délais de mise en veille des nœuds capteurs inactifs pour une meilleure conservation d'énergie.

#### ➤ Gestion de flux :

Cette catégorie regroupe les techniques qui définissent des protocoles de gestion de transfert des données (routage, sélection de canal de transmission, etc.). Nous pouvons trouver des approches préventives ou curatives sur les différentes couches (réseau, liaison de données, etc.) telles que :

- **Routage multipath** : utilise un algorithme préventif pour déterminer plusieurs chemins depuis chaque capteur vers le nœud collecteur.

Ceci garantit la présence de plus d'un chemin fiable pour la transmission et offre une reprise rapide du transfert en cas de panne sur le chemin principal et choisissant un des chemins qui restent.

- **Recouvrement de routes** : après la détection de panne, une technique curative permet de créer un nouveau chemin qui soit le plus fiable pour retransmettre les données.
- **Allocation de canal** : cette solution est implémentée au niveau de la couche MAC. Elle permet d'effectuer une allocation du canal de transmission d'une manière à diminuer les interférences entre les nœuds voisins et éviter les collisions durant le transfert.
- **Mobilité** : certains protocoles proposent comme solution tolérante aux pannes la sélection d'un ensemble de nœuds mobiles chargés de se déplacer entre les capteurs et collecter les données captées. Ceci réduira l'énergie consommée au niveau de chaque capteur en éliminant sa tâche de transmission. Un nœud mobile est généralement doté d'une batterie plus importante que celle d'un nœud capteur.

#### ➤ Gestion des données :

Les protocoles classés dans cette catégorie offrent une meilleure gestion de données et de leur traitement. Deux principales sous-catégories sont déterminées :

- **Agrégation** : cette approche est considérée comme une approche préventive. Elle permet d'effectuer un traitement supplémentaire sur les données brutes captées depuis l'environnement. Un nœud agrégateur combine les données provenant de plusieurs nœuds en une information significative. Ce qui réduit considérablement la quantité de données transmises en consommant moins d'énergie pour leur dissémination.

Ceci permet donc d'augmenter la durée de vie du réseau.

En outre, elle aussi d'amortir l'erreur si le résultat de l'agrégation est une moyenne.

- **Clustering** : une des importantes approches pour traiter la structure d'un réseau de capteurs est le clustering. Il permet la formation d'un backbone virtuel qui améliore l'utilisation des ressources rares telles que la bande passante et l'énergie. Par ailleurs, le clustering aide à réaliser du multiplexage entre différents clusters. En outre, il améliore les performances des algorithmes de routage. Plusieurs protocoles utilisent cette approche préventive et parfois elle est considérée comme une approche curative.

### 2.5.1.3 Classification selon le niveau d'implémentation

Cette classification permet de répartir les protocoles sur les différentes couches de l'architecture des réseaux de capteurs. Ainsi, les algorithmes de routage sont au niveau réseau, les techniques de sélection de canal sur la couche MAC, etc. [11]

## 2.6 Aspect de surveillance d'un RCSF

### 2.6.1 La longévité du réseau

Prolonger la durée de vie du réseau a été l'objectif d'optimisation pour la plupart des protocoles de communication publiés pour les réseaux de capteurs. Les positions des nœuds significatives impactent sur la surveillance des Réseaux de Capteurs Sans Fil, la durée de vie du réseau. Par exemple, les variations de densité des nœuds dans toute la région peuvent mener à la charge de trafic déséquilibré et provoquer des goulets d'étranglement.

Un Réseau de capteur sans fil devient inexploitable quand la connectivité entre ses nœuds est perdue, on dit que le réseau est mort.

### 2.6.2 Connectivité

De nombreuses applications nécessitent l'ajout de contraintes de connectivité aux modèles de couverture. Le réseau doit assurer une très bonne communication entre ses différents nœuds. Chaque nœud capteur doit pouvoir communiquer au minimum avec un autre nœud et cela en vue d'avoir les informations nécessaires sur les phénomènes à surveiller. Il est parfois utile d'envisager des formes plus puissantes de connectivité, comme la  $k$ -connectivité. Un réseau est dit  $k$ -connecté, s'il y a au moins  $k$  chemins disjoints entre deux nœuds quelconques, un tel réseau reste connecté, même si  $k-1$  nœuds tombent en panne. Un réseau  $k$ -connecté ( $k > 1$ ) présente une meilleure tolérance aux pannes qu'un réseau 1-connecté. La  $k$ -connectivité permet de prolonger la durée de vie du réseau en cas de défaillance de nœuds. La connectivité est affectée par les changements de topologie dus à la mobilité, la défaillance de nœuds, les attaques, etc. Ce qui a pour conséquences : la perte de liens, l'isolement des nœuds, le partitionnement du réseau, etc. [12]

### 2.6.3 La couverture

La couverture d'un réseau de capteurs représente la qualité de la surveillance que le réseau peut fournir, par exemple, à quel point une zone d'intérêt est surveillée par des capteurs, et avec quelle efficacité un réseau de capteurs peut assurer son rôle principal. Il est important de comprendre comment la couverture d'un réseau de capteurs dépend de divers paramètres de réseau afin de mieux concevoir et utiliser les réseaux de capteurs pour différents scénarios d'application.

Plusieurs travaux ont déjà traité ce problème de plusieurs manières. Cependant, il existe deux catégories de couverture :

- **Couverture des cibles pré-déterminées** : Consiste à trouver un sous-ensemble de nœuds connexes et qui assure la surveillance d'un ensemble de cibles dont la position est connue à priori.
- **Couverture d'une zone** : consiste à trouver un sous-ensemble de nœuds connexes et qui assure la surveillance de toute la zone de déploiement. Soit un réseau de  $n$  nœuds déployés pour surveiller une région d'intérêt, un nœud est caractérisé par son rayon de couverture  $r$

et son rayon de transmission. La couverture dans un réseau de capteur sans fil dépend de la densité des nœuds.

#### 2.6.4 Méthodologie de déploiement :

Les capteurs sont au préalable déployés sur une zone à surveiller. Pour satisfaire de nouvelles contraintes ou pour pallier des pannes, un déploiement de nœuds supplémentaires, dit itératif, peut être requis. Différents modes de déploiement sont envisageables et dépendent essentiellement de l'application de surveillance. Une fois déployés, nous supposons que les capteurs sont statiques.

##### ➤ **Déploiement déterministe :**

Lorsque l'environnement est accessible ou connu, il est possible de placer précisément les nœuds sur la zone. Dans le problème que nous étudions, il est alors possible de programmer leurs activités au préalable. C'est ainsi, par exemple, que sont mis en place les capteurs chargés de réguler la climatisation d'un immeuble ou de surveiller les constantes médicales de malades à distance. On parle alors de déploiement déterministe.

##### ➤ **Déploiement aléatoires :**

L'utilisation des capteurs dans des zones inaccessibles ou sensibles rend impossible un déploiement déterministe, au cours duquel chaque objet serait placé à une position prédéterminée. Les nœuds peuvent alors être déployés à l'aide de moyens divers. Il a souvent été question d'un déploiement aléatoire des capteurs, effectué comme un jeté de graines. Il semble pourtant difficile, vu la fragilité des capteurs existants, d'envisager un déploiement par avion par exemple.

Néanmoins, un déploiement aléatoire peut être obtenu à partir d'une distribution de capteurs à des individus. Dans ce travail, nous supposons des réseaux déployés aléatoirement. Une fois disséminés, il est couramment admis que les capteurs sont statiques. [13]

## 2.7 La redondance

### 2.7.1 Le concept de redondance

La redondance d'une façon générale est utilisée pour prévenir un dysfonctionnement dans un système. Comme dans le cas des RCSF, la redondance est établie pour étendre la durée de vie du réseau, surtout que cette dernière se rapporte au surnombre des nœuds.

Plusieurs informations différentes sur une même variable du système sont fournies par cette redondance de connaissances.

A ce moment, on aura la possibilité de vérifier la cohérence de l'information obtenue par des tests de cohérence. Cette redondance d'informations se divise en deux catégories : la redondance physique et la redondance analytique.

##### ➤ **La redondance physique :**

Le principe de cette redondance est de disposer de plusieurs capteurs afin d'obtenir plusieurs informations sur une même variable. Pour obtenir deux mesures d'une même température, on double tout simplement le capteur de température.

La redondance physique montre un désavantage majeur qui est le coût. En doublant le nombre de capteur ça revient à doubler le prix d'achat de ce dernier. De plus, les contraintes ergonomiques liées à l'installation de ces capteurs peuvent limiter leurs utilisations.

Traditionnellement, la sûreté de fonctionnement du système dynamique est assurée en utilisant la redondance physique.

Les modes qui peuvent être appliqués aux différents niveaux d'un système sont :

**1. La redondance active:** Elle est adaptée pour les systèmes critiques à cause de son principe qui tolère la faute d'un ou de plusieurs composants.

**2. La redondance passive :** Basée sur la redondance en utilisant une seule copie, appelée "copie primaire", alors que les autres copies sont utilisées seulement en cas de défaillance.

**3. La redondance hybride :** Comme son nom l'indique c'est une combinaison entre les deux redondances citées précédemment. Le choix de la stratégie de redondance se fait en fonction des contraintes et des besoins applicatifs. Dans le cas des réseaux de capteurs sans fil, la réplication active est préférable pour tolérer aux défaillances fréquentes des nœuds capteurs.

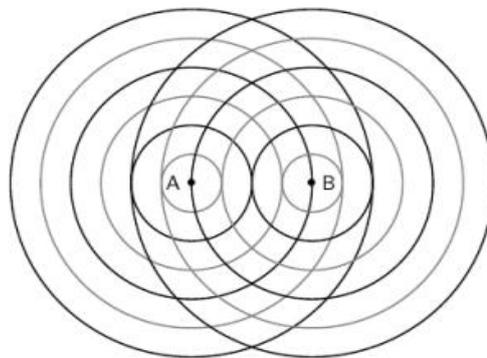
➤ **La redondance analytique :**

C'est la redondance à base de modèles. Par la notion de modèle on entend une reproduction formelle réalisant les mêmes performances que le système étudié. L'approche utilisant la redondance analytique se décompose généralement en deux phases distinctes :

- La première concerne la génération de résidus caractéristiques de la panne.
- La seconde étape concerne la prise de décision qui a trait à la détection et éventuellement à la localisation d'un élément défaillant. Elle met en œuvre des techniques de détection de ruptures et de tests d'hypothèses. Ce principe est souvent difficile à exploiter comme dans le cas des RCSF à cause du changement fréquent de la topologie du réseau liée à l'épuisement de la batterie ou même la mobilité des capteurs. Pour ces raisons, la redondance matérielle est utilisée surtout que le coût des capteurs sans fils devient de plus en plus réduit.

**2.7.2 Effets indésirables de la redondance :**

**Interférences Radio et collision de Paquets :** En radio, une interférence est la superposition de deux ou plusieurs ondes. Il est fréquent, pour les fréquences supérieures à quelques centaines de kilohertz, qu'une antenne de réception reçoive simultanément l'onde directe en provenance de l'émetteur et une (ou plusieurs) onde réfléchi(e) par un obstacle. Les deux signaux vont se superposer et, en fonction de la différence de phase entre eux, voir leurs amplitudes s'additionner ou se soustraire. Ce genre d'interférence est responsable du fading, terme anglo-saxon désignant une variation plus ou moins rapide de l'amplitude du signal reçu. Mais le phénomène ne se limite pas aux seules ondes radio. [14]



**Figure 2.4 Interférences Radio.**

**Envoi de la même information plusieurs fois :** La source diffuse son message à tous ses voisins. Chaque mobile retransmet le paquet reçu. La diffusion continue après réception par le destinataire. Tous les mobiles risquent de recevoir le message. Ce phénomène génère une consommation inutile de la bande passante, d'énergie, réceptions simultanées, et réceptions multiples en boucles infinies. La redondance physique présente des inconvénients en matière de coût de l'installation et en plus elle est encombrante. Alors que la redondance analytique, elle est synonyme aux problèmes de la complexité. [14]

**2.8 Les Travaux Antérieurs :**

Dans cette section, nous présentons certains travaux développés dans le cadre de la supervision des RCSF et qui emploient l'approche distribuée.

Dans [15], Barrenetxea a conçu un algorithme décentralisé simple et efficace pour transmettre les données d'une source unique à une destination unique. Dans ce but, il a reformulé le problème du routage en celui d'imposer les contraintes appropriées à des chemins aléatoires sur des graphes. Ensuite, il a dérivé des algorithmes distribués pour trouver les paramètres locaux des marches aléatoires qui induiront une certaine propriété de distribution de la charge dans le réseau. La caractéristique principale de cette formulation est que les nœuds seront capables de transmettre des messages entre une source et une destination par tous les chemins possibles, sans une reconnaissance du chemin au préalable, et sans devoir tenir à jour une information sur l'état des chemins à chaque nœud. L'auteur considère aussi les aspects pratiques de l'utilisation des techniques de codage par descriptions multiples dans des réseaux de capteurs à grande échelle, où le nombre de chemins disponibles entre une source et une destination est important.

L'auteur de [16] propose un algorithme distribué pour le contrôle de la topologie et l'auto-organisation basée sur une connaissance locale pour la construction d'un backbone virtuel sur les réseaux ad-hoc et les réseaux de capteurs sans fil. Il considère que les nœuds peuvent se trouver en un des états suivants: isolé, actif, dominant ou dominé. De cette manière, il arrive à construire des clusters dans le réseau. Chaque cluster est dominé par un cluster head (ou dominant).

Colette Johnen et al. [17] propose l'approche appelée DMAC (Distributed and Mobility-Adaptive Clustering algorithm). C'est un algorithme d'auto-organisation qui ressemble à celui proposé dans [16] pour la partition des nœuds en dominants et dominés, mais en utilisant d'autres métriques comme le poids des nœuds, c'est-à-dire le nombre de voisins. Les auteurs formulent le problème avec des règles et des prédicats logiques et quelques propriétés de la théorie des graphes. Un travail récent proposé dans consiste à utiliser une variante SNMP (Simple Network Management Protocol) réduit pour les RCSF.

## 2.9 Conclusion :

Dans ce chapitre, nous avons présenté des principaux concepts et quelque aspect de surveillance des réseaux de capteurs sans fils, en générale la surveillance des RCSFs consiste à détecter les défauts et les pannes ensuite générer des solutions en temps réel, puis nous avons parlé sur les pannes dans les réseaux de capteurs sans fils, et la tolérance aux pannes. En fin nous avons vu que la redondance des nœuds est gérée par l'échange des messages, et la redondance d'informations en générale se divise en deux catégories : la redondance physique et la redondance analytique. Et aussi nous avons vu les Travaux Antérieurs. Dans le chapitre suivant, nous présenterons les outils de simulations existants et une définition générale sur le simulateur omnetpp.

# Chapitre 03

**Simulation et évaluation des performances**

### 3.1-Introduction :

Après avoir vu les RCSFs, leurs structures et domaines d'application, les principaux concepts de surveillance des réseaux de capteurs sans fils.

Nous allons voir dans ce chapitre que La simulation joue un rôle très important pour étudier le comportement dynamique, les protocoles de communication, la gestion du RCSF. Pour la réalisation de notre projet de simulation, on va utiliser l'environnement OMNET++. Parmi les simulateurs populaires dédiés aux RCSFs. Enfin, nous décrirons comment installer omnetpp en Ubuntu 16.04s.

### 3.2-Généralité sur la simulation :

#### Qu'est-ce qu'un simulateur, quelle est son utilité?

La simulation est définie comme « un outil utilisé par le chercheur, l'ingénieur, le militaire, etc. pour étudier les résultats d'une action sur un élément sans réaliser l'expérience sur l'élément réel ». Les simulations sont réalisées dans un environnement simulé et non pas réel. Ils peuvent aussi bien simulés un réseau de type LAN ou bien WLAN (suivant les simulateurs). Certains simulateurs sont plus complets que d'autres dans les résultats d'une simulation mais tous permettent d'étudier le comportement d'un réseau ayant une topologie et des caractéristiques précises.

Les simulateurs permettent ainsi d'anticiper sur la topologie d'un réseau. Lorsque les résultats d'une simulation ne sont pas satisfaisants, il est facile de modifier la topologie pour corriger les problèmes avancés par la simulation précédente.

La simulation est aussi intéressante pour créer la topologie d'un réseau avant de la mettre en place réellement. Et cela est possible car les simulateurs intègrent un grands nombre d'outils permettant de réaliser des simulations assez réalistes. On peut aussi se servir d'un simulateur pour tester un nouveau protocole (la facilité de l'intégration dépend du simulateur utilisé) avant de l'utiliser réellement (un protocole de réseau de capteurs sans fil).

### 3.3-Types de simulation :

En fonction du type d'évènements dans la simulation, nous distinguons deux types de systèmes de simulation : les systèmes discrets et les systèmes continue.

#### 3.3.1-Systèmes de simulation discret :

Sont des systèmes pour lesquels les variables concernées par la simulation ne changent d'état qu'en un nombre fini de points sur l'axe du temps.

#### 3.3.2-Systèmes de simulation continue :

Ce sont des systèmes pour lesquels les variables peuvent changer d'état à n'importe quel instant pendant la simulation.

### 3.4-Les simulateurs de réseau existant :

Il existe plusieurs simulateurs de réseau tel que : NS2, OMNET++, OPNET, GLOMOSIM, JSIM...etc. Parmi ceux, on va citer [2] :

#### NS2 :

NS est un simulateur à évènements discrets très répandu dans le domaine de la recherche pour tout ce qui se réfère aux réseaux. L'utilisation de l'appellation "NS2" précise la version du simulateur NS. Il constitue un support important pour la simulation de protocoles TCP, protocoles de routage, protocoles de multicast.

#### GloMoSim :

GloMoSim (Global Mobile information system Simulator) a été conçu selon une architecture orientée « couche » similaire que celle des sept couches OSI pour les réseaux.

#### OMNET++ :

OMNeT++ IDE (Integrated Development Environment) est basé sur la plateforme Eclipse. C'est un environnement open source qui fournit des outils pour la création et la configuration des modèles de réseaux (les fichiers NED et INI) et des outils pour l'exécution d'un lot de

programmes ainsi que pour l'analyse des résultats de simulation. OMNeT++ semble être le meilleur parmi les solutions open source et freeware.

### 3.5-Le Simulateur OMNET++

#### 3.5.1-Choix du simulateur OMNET++

Le déploiement d'un réseau de capteurs exige une étape de simulation avant son installation sur site. La simulation permet de tester à moindre coût les performances d'une solution.

OMNeT++ est un environnement de simulation à événements discrets basé sur le langage C++, une application open source et sous licence GNU. Il est totalement programmable, paramétrable et modulaire ainsi grâce à son architecture flexible et générique, il a été utilisé avec succès dans divers domaines, notamment:

- La modélisation de réseaux de file d'attente,
- La modélisation de protocoles de communication,
- La validation des architectures hardware,
- L'évaluation de performances pour des systèmes software complexes.

OMNET++ sera notre environnement de simulation, grâce à son architecture modulaire, il s'agira d'étendre le simulateur en implémentant un nouveau modèle spécifique aux réseaux de capteurs.

#### 3.5.2-Présentation d'OMNET++ :

OMNet++ est un simulateur à événements discrets à base de composants implémenté en C++. OMNet++ est utilisé pour simuler les réseaux avec ou sans fil, il est plus générique que NS-2. Grâce à son architecture modulaire, OMNET++ est très largement répandu dans divers domaines d'applications tel que :

- La modélisation des protocoles de communications,
- La modélisation des réseaux filaires et sans fils,
- La modélisation des systèmes répartis,
- Les architectures HardWare,
- En général, il peut être utilisé pour n'importe quel système à évènements discrets pouvant être modélisé selon des entités communiquant par envoi de messages.

#### 3.5.3-Installation du simulateur OMNET++ :

Installation en ligne de commande avant de commencer l'installation, actualisez la base de données des packages disponibles. Tapez le terminal:

```
$ sudo apt-get update
```

Pour installer les packages requis, tapez le terminal:

```
$ sudo apt-get install build-essential gcc g++ bison flex perl tcl-dev tk-dev libxml2-dev zlib1g-dev default-jre doxygen graphviz libwebkitgtk-1.0-0
```

Pour utiliser Qtenv avec le support de visualisation 3D , installez les packages de développement pour OpenSceneGraph(3.2 ou version ultérieure) et les packages osgEarth(2.7 ou version ultérieure).

```
$ sudo apt-get install qt4-qmake libqt4-dev libqt4-opengl-dev openscenegraph libopenscenegraph-dev openscenegraph-plugin-osgearth osgearth osgearth-data libosgearth-dev
```

Pour activer le support de simulation parallèle optionnel, vous devez installer les packages MPI :

```
$ sudo apt-get install openmpi-bin libopenmpi-dev
```

Télécharger Omnetpp : <https://omnetpp.org/download/old/>

Déballage : \$ tar xvfz omnetpp-4.6-src.tgz

Variables d'environnement : \$ cd omnetpp-4.6

Editez le shell bashrc :`$ nano ./bashrcadd:`

Export `PATH=$PATH:/home/userdirectory/omnetpp-4.6/bin` export

`OMNET_DIR=/home/userdirectory/omnetpp-4.6` puis quittez et exécutez : `$ source ~/.bashrc`

Editez le fichier configure.user : `$ nano configure.user` set value `WITH_TKENV=no` and `WITH_QTENV=no`

Type terminal : `./configure`

Lorsque `./configure` est terminé, compilez OMNETpp.

Tapez le terminall : `$ make`

### 3.5.4-Les principaux fichiers d'OMNET++ :

Les différents fichiers sont :

**Fichier (.Ned)** : (Network descriptor file). Utilise le langage NED de description de réseau. Il peut être utilisé en 2 modes : Mode Graphique ou Mode Texte qui permettent de décrire les paramètres et les ports du module. Les erreurs commises sont indiquées en temps réel par un point rouge situé à la gauche du code.

**Fichier (.ini)**: Est lié étroitement avec le fichier NED. Permet à l'utilisateur d'initialiser les paramètres des différents modules ainsi la topologie du réseau.

**Fichier (.msg)** : (used to define packet types) utilisé pour définir les types de paquets.

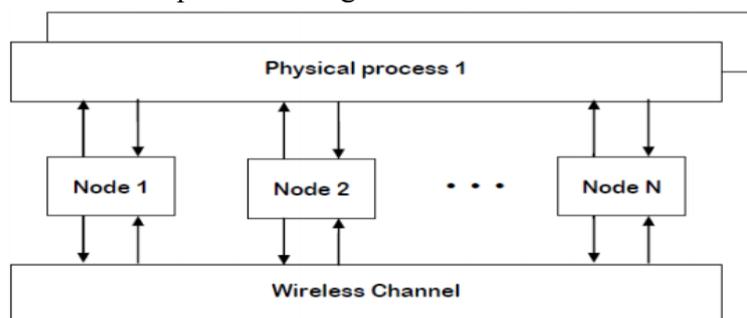
Les modules communiquent en échangeant des messages. Ces derniers peuvent être déclarés dans un fichier dont l'extension est (.msg) où l'on peut ajouter des champs de données.

**Fichier (.cc)** : (c++ file) fichier c++.

**Fichier (.h)** : (header file) En tête de fichier

### 3.6-Environnement de Castalia :

Le simulateur Castalia représente une extension des fonctionnalités d'Omnetpp notamment dans les phases de la transmission de paquets et la gestion de l'énergie [18]. Dans Omnetpp le paquet est transmis via le canal de communication en utilisant les ports des modules. Or que dans Castalia le canal de communication représente le canal sans fil "Wireless Channel" comme indiqué dans la figure



**Figure 3.1 Les nœuds et leurs connections en Castalia.**

Dans Castalia, chaque nœud est formé d'un ensemble de modules. Un paquet généré par un nœud est transmis directement vers le module de communication comme indiqué dans la figure

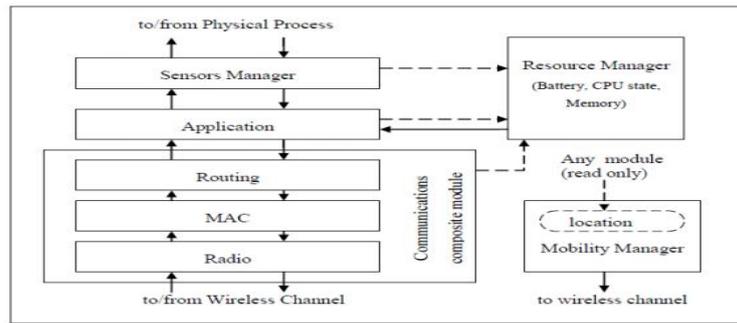


Figure 3.2 La vie d'un nœud sous Castalia. [19]

### 3.6.1-Installation de castalia :

Les étapes sont les suivantes:

1. Décompression de l'archive Castalia déjà télécharger à partir du site (<http://castalia.npc.nicta.com.au/download.php>) avec la commande :  
`xzf castalia.tgz ;`
2. Importation du projet dans OMNeT++ ide : Après avoir lancer Omnet++, l'importation du projet se fait avec un clic droit sur "General->Exiting project into workspace" et en choisir la racine du projet.
3. Exécution d'un exemple : Une fois l'importation est en complet, un clic droit sur le fichier omnetpp.ini de n'importe quel exemple lance la simulation et génère un fichier \*.txt qui se définit comme le fichier trace de la simulation.

### 3.6.2-Les commandes de castalia :

Faut se situer dans le répertoire voulu :

1. `../bin/Castalia` Affiche la liste des fichiers et des configurations.
2. `../bin/Castalia -c General` Commencer l'exécution de la configuration. Ce qui donne un fichier trace.
3. `CastaliaResults` Affiche les résultats obtenus lors de l'exécution.
4. `CastaliaPlot` Affiche le graphe des simulations.
5. `../bin/Castalia -h` affiche l'aide.

### 3.7-Environnement de Simulation :

Dans cette partie, nous allons présenter notre implémentation et les résultats de simulation que nous avons réalisée

#### 3.7.1-Environnement matériel :

La simulation a été réalisée sur un ordinateur HP dont la configuration est comme suit :

Processeur Intel Core I3 CPU 2.3 GHz

Mémoire 4 GB

Disque dur 500 GB

#### 3.7.2-Environnement logiciel :

Notre simulation a été réalisée dans l'environnement logiciel suivant :

- Système d'exploitation : Linux Ubuntu 16.04.
- Le simulateur OMNet++ 4.6 ;
- Le Simulateur Castalia 3.2.

### 3.8-Réalisation de la simulation :

Nous allons faire notre simulation via l'invite de commande sous linux. On utilise la commande **Castalia** pour afficher la liste des fichiers et des configurations

La combinaison de ses options permet de définir plusieurs scénarios de simulation dont chaque scénario utilise une configuration spécifique

Dans notre simulation nous allons utiliser la configuration de trois scénarios et en comparant entre eux.

**3.8.1 Objectif**

L’objectif de notre travail est de créer dans l’environnement de simulation OMNeT++, un modèle de surveillance du RCSF. Ce modèle est basé sur la redondance des nœuds et qui permet de réveiller les nœuds endormis en cas de panne ou d’épuisement d’énergie d’un nœud actif, donc le but est d’optimiser la consommation énergétique et de prolonger la durée de vie du réseau.

**3.8.2 Organisation du Réseau**

Nous avons créé un projet et on commence par l’ajout de fichier omnetpp.ini qui contient des paramètres spécifiques à Castalia. Ensuite, On définit trois sections (General, scénario1, scénario2) et dans chaque scénario on a définir l’état, les positions des nœuds.

Nous avons simulé un réseau composé de 29 nœuds, déployés aléatoirement. Voilà Les positions de ces nœuds et les voisins de quelques nœuds essentiels et le déploiement de notre réseau:

- **Les positions des nœuds**

Nœud	0	1	2	3	4	5	6	7	8
Position	(25,25)	(50, 75)	(25, 150)	(240, 50)	(150, 240)	(75, 200)	(100, 25)	(240, 150)	(125, 100)

Nœud	9	10	11	12	13	14	15	16
Position	(75, 100)	(100, 100)	(175, 50)	(175, 150)	(235, 200)	(100, 150)	(50, 125)	(125, 175)

Nœud	17	18	19	20	21	22	23
Position	(200, 175)	(225, 125)	(25, 230)	(175, 75)	(25, 107)	(205, 30)	(201, 81)

Nœud	24	25	26	27	28
Position	(140, 208)	(245, 230)	(110, 220)	(205, 105)	(147, 50)

**Tableau 3.1 – tableau illustre Les positions des nœuds.**

- **Les voisinages**

Le tableau illustre les voisins de quelques nœuds essentiels tel que :

Nœuds 10, 15, 25

Nœuds	10	15	25
Voisinage	14, 8 ,9 ,21	9 ,14 ,5 ,2 ,21 ,1	16 ,4 ,26 ,12

**Tableau 3.2 – tableau illustre Les voisinages des nœuds.**

- Déploiement de réseau :

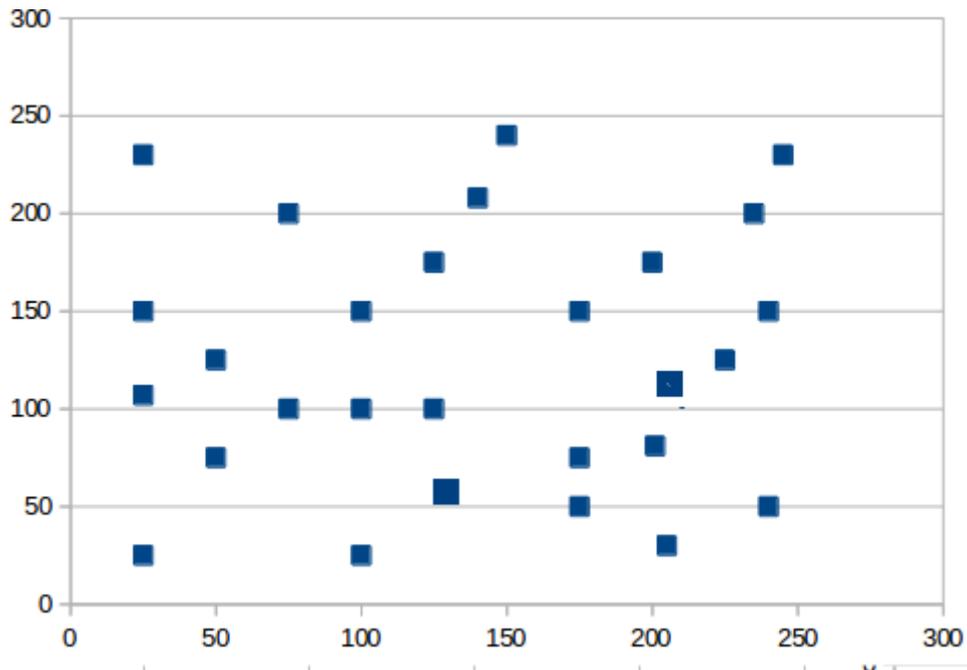


Figure 3.3- Déploiement de réseau.

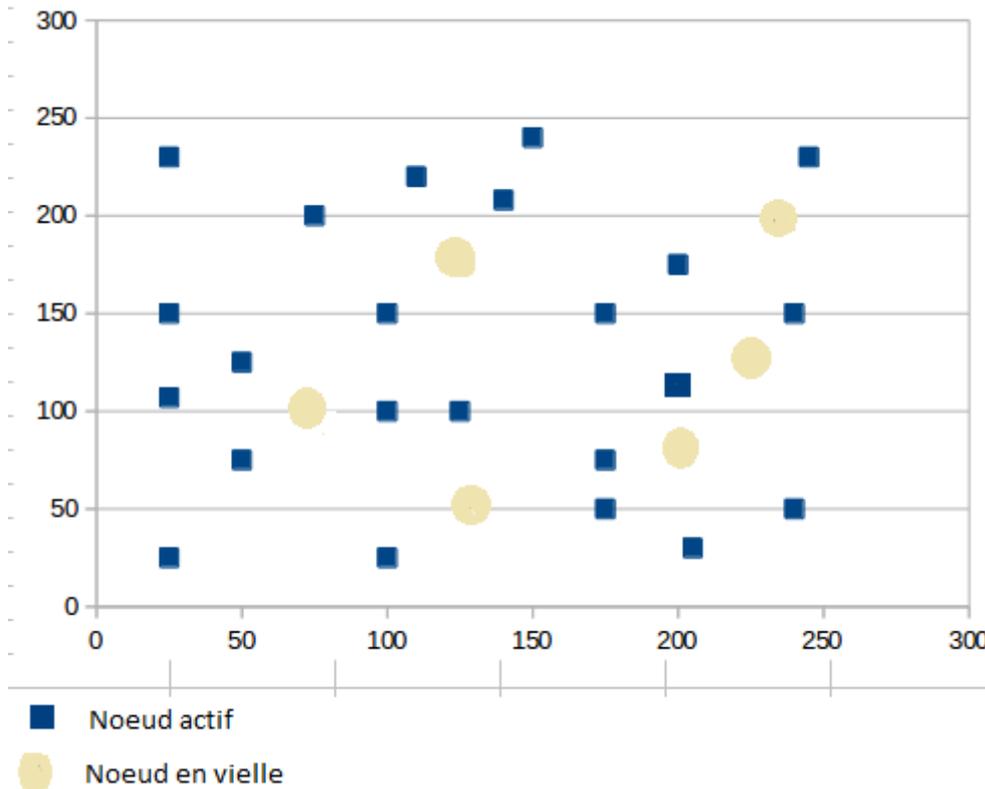


Figure 3.4- Déploiement de réseau Sans défaillance (Général).

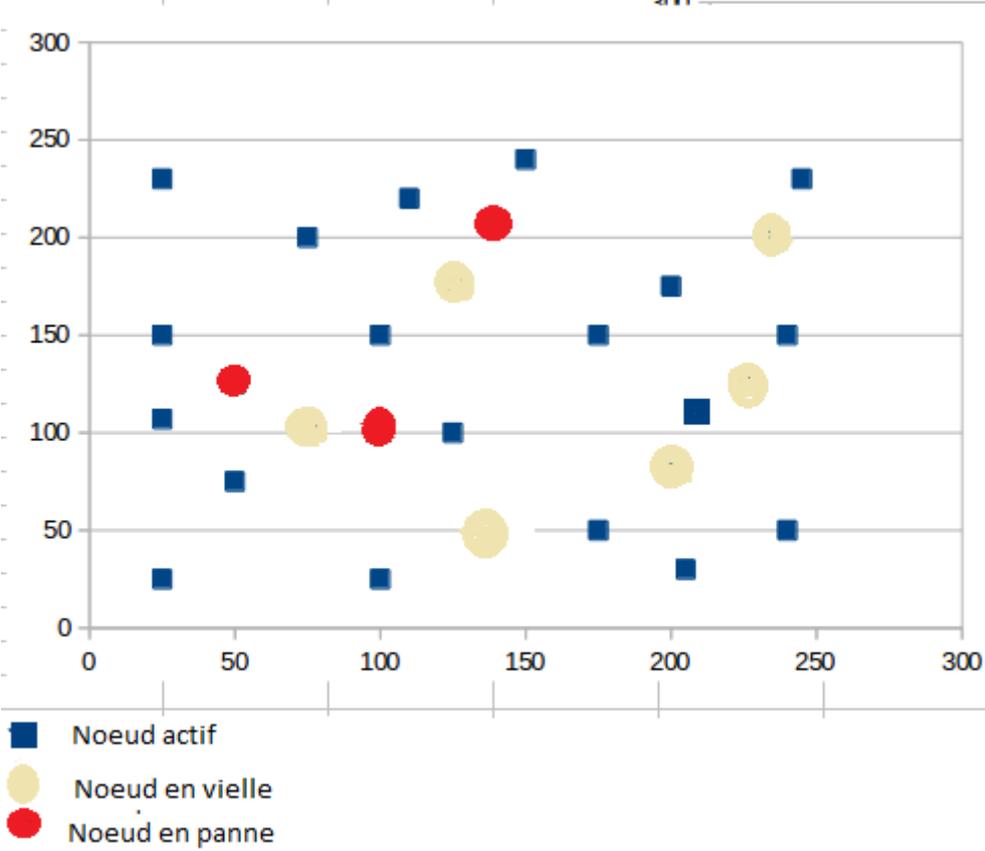


Figure 3.5- Déploiement de réseau Avec défaillance (sénario1).

Avec la technique de la redondance

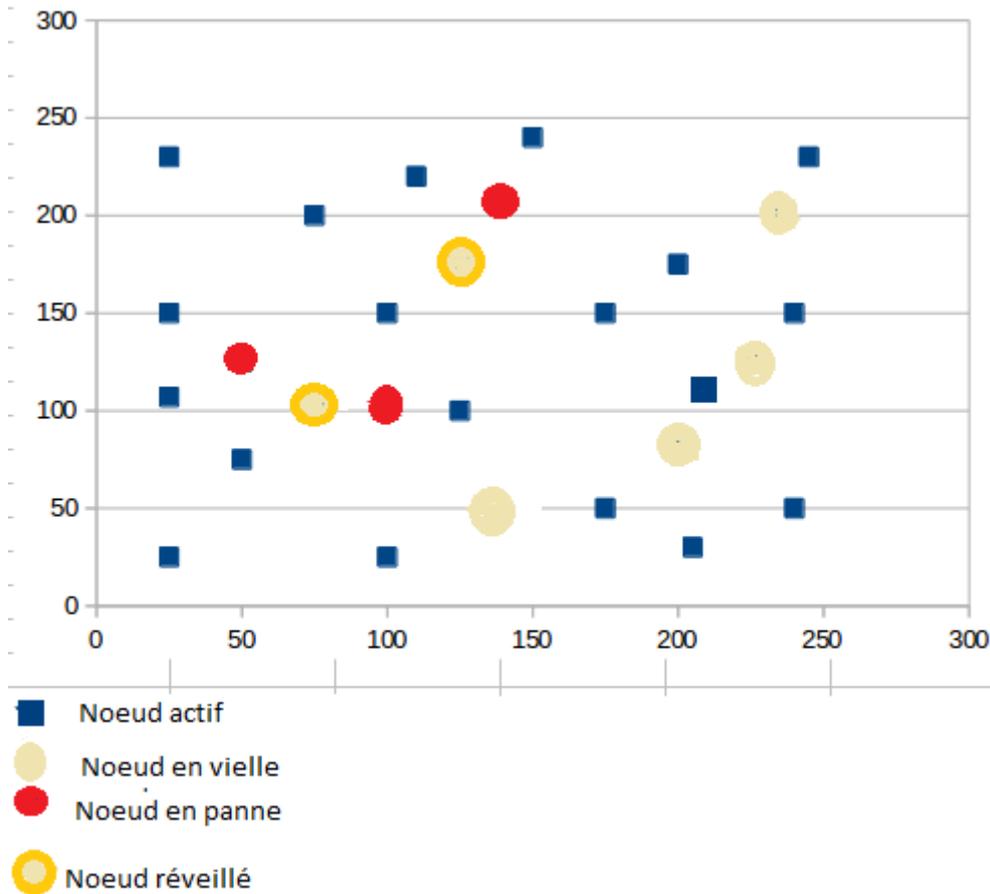


Figure 3.6- la technique de la redondance. (sénario2)

### 3.8.3 Fichiers résultats : (Résultats de la simulation)

- La liste des fichiers d'entrée disponibles et des configurations :

```
hp@hp-HP-Notebook:~/Téléchargements/omnetpp-4.6/samples/Castalia-3.2/Simulations/senario2$ Castalia
List of available input files and configurations:
* omnetpp.ini
  General
  InterferenceTest1
  InterferenceTest2
  senario1
  senario2
  CSinterruptTest
  varyInterferenceModel
```

Figure 3.7- La liste des fichiers d'entrée disponibles et des configurations

- **Fichier Trace :**

Dans ce il est enregistré en détail le scénario de la simulation c.à.d. le processus parcouru par chaque nœud durant la simulation.

```

700.110483213555SN.node[0].Application Received packet #3499 from node 10
700.310483419429SN.node[0].Application Received packet #3500 from node 10
700.510483625303SN.node[0].Application Received packet #3501 from node 10
700.710483831177SN.node[0].Application Received packet #3502 from node 10
700.910484037051SN.node[0].Application Received packet #3503 from node 10
718.110501742215SN.node[0].Application Received packet #3509 from node 10
718.310501948089SN.node[0].Application Received packet #3590 from node 10
718.510502153963SN.node[0].Application Received packet #3591 from node 10
718.710502359835SN.node[0].Application Received packet #3592 from node 10
718.910502565711SN.node[0].Application Received packet #3593 from node 10
719.110502771585SN.node[0].Application Received packet #3594 from node 10
719.310502977469SN.node[0].Application Received packet #3595 from node 10
721.873402491772SN.node[0].Application Received packet #3608 from node 1
722.473406580091SN.node[0].Application Received packet #3611 from node 1
729.673455639919SN.node[0].Application Received packet #3647 from node 1
729.873457092692SN.node[0].Application Received packet #3648 from node 1
732.710516771017SN.node[0].Application Received packet #3662 from node 10
732.910516976891SN.node[0].Application Received packet #3663 from node 10
733.110517182765SN.node[0].Application Received packet #3664 from node 10
733.310517388639SN.node[0].Application Received packet #3665 from node 10
733.510517594513SN.node[0].Application Received packet #3666 from node 10
733.710517800387SN.node[0].Application Received packet #3667 from node 10
742.059655279967SN.node[0].Application Received packet #3709 from node 17
751.110535711425SN.node[0].Application Received packet #3754 from node 10
751.510536123173SN.node[0].Application Received packet #3756 from node 10
751.710536329047SN.node[0].Application Received packet #3757 from node 10
751.910536534921SN.node[0].Application Received packet #3758 from node 10
752.110536740795SN.node[0].Application Received packet #3759 from node 10
752.310536946669SN.node[0].Application Received packet #3760 from node 10
754.873627349317SN.node[0].Application Received packet #3773 from node 1
755.07362871209 SN.node[0].Application Received packet #3774 from node 1
755.27362904099SN.node[0].Application Received packet #3777 from node 1
762.473679134691SN.node[0].Application Received packet #3811 from node 1
762.673680497464SN.node[0].Application Received packet #3812 from node 1
765.710550740227SN.node[0].Application Received packet #3829 from node 10
765.910550946101SN.node[0].Application Received packet #3828 from node 10
766.110551151975SN.node[0].Application Received packet #3829 from node 10
766.310551357849SN.node[0].Application Received packet #3830 from node 10
766.510551563723SN.node[0].Application Received packet #3831 from node 10
766.710551769597SN.node[0].Application Received packet #3832 from node 10
784.110569680635SN.node[0].Application Received packet #3919 from node 10
784.310569885095SN.node[0].Application Received packet #3920 from node 10

```

Figure 3.8- exemple de contenu du fichier trace

- **Fichier Résumé :**

Dans ce dernier ils sont enregistrés les valeurs résultat de la simulation (le nombre de paquets transmis, reçus et perdus, la consommation d'énergie...etc.), c'est en exploitant ce fichier qu'on peut produire des présentations graphiques des résultats, d'ailleurs CASTALIA permet d'effectuer ceci en utilisant le mode ligne de commande et en exploitant le Framework GNUPlot destiné pour l'élaboration des présentations graphique des résultats de CASTALIA

```

Castalia| what:[senario1,InterferenceTest1,InterferenceTest2]varyInterferenceModel (1)
Castalia| label:2019-07-01 21:51
Castalia| label:senario1,InterfModel=0
Castalia| module:SN.node[0].ResourceManager
Castalia| simple output name:Consumed Energy
Castalia| 0.79813
Castalia| module:SN.node[0].Communication.Radio
Castalia| simple output name:RX pkt breakdown
Castalia| 289 Failed with NO interference
Castalia| 270 Failed, below sensitivity
Castalia| 358 Received with NO interference
Castalia| module:SN.node[0].Application
Castalia| index:1 simple output name:Packets received per node
Castalia| 60
Castalia| index:2 simple output name:Packets received per node
Castalia| 71
Castalia| index:3 simple output name:Packets received per node
Castalia| 58
Castalia| index:4 simple output name:Packets received per node
Castalia| 35
Castalia| index:5 simple output name:Packets received per node
Castalia| 71
Castalia| index:6 simple output name:Packets received per node
Castalia| 28
Castalia| index:8 simple output name:Packets received per node
Castalia| 35
Castalia| histogram name:Application level latency, in ms
Castalia| histogram_min:0 histogram_max:200
Castalia| histogram_values 0 0 0 358 0 0 0 0 0 0
Castalia| module:SN.node[1].ResourceManager
Castalia| simple output name:Consumed Energy
Castalia| 6.28785
Castalia| module:SN.node[1].Communication.Radio
Castalia| simple output name:TXed pkts
Castalia| 499 TX pkts
Castalia| module:SN.node[2].ResourceManager
Castalia| simple output name:Consumed Energy|

```

Figure3.9- exemple de contenu du fichier résumé

**Remarque :**

Vue que GNUPlot produit des présentations graphiques décalés dans la plupart des cas ce que peut influencer la lecture et l'interprétation des résultats obtenues nous favoriser d'exporter ces résultats vers Excel afin de produire des présentations graphiques plus claires et plus interprétables.

• La Consommation d'énergie :

```
hp@hp-HP-Notebook:~/Téléchargements/omnetpp-4.6/samples/Castalia-3.2/Simulations
/zsenario$ CastaliaResults -i 190702-225755.txt -s energy

ResourceManager:Consumed Energy
+-----+-----+
| General | senario1 | senario2 |
+-----+-----+
| 56.784  | 49.403  | 54.324  |
+-----+-----+
```

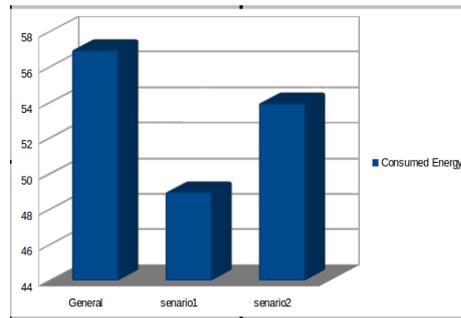


Figure3.10- L'énergie moyenne consommée par chaque nœud.

```
hp@hp-HP-Notebook:~/Téléchargements/omnetpp-4.6/samples/Castalia-3.2/Simulations
/zsenario$ CastaliaResults -i 190702-225755.txt -s energy --sum

ResourceManager:Consumed Energy
+-----+-----+
| General | senario1 | senario2 |
+-----+-----+
| 1306.032 | 1136.28  | 1249.449 |
+-----+-----+
```

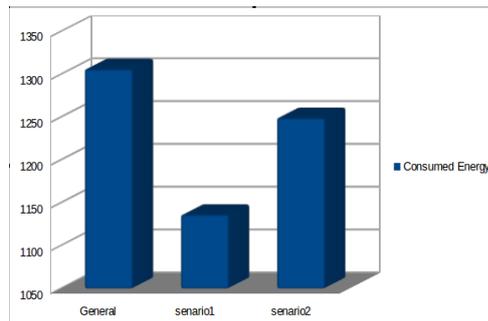


Figure3.11- L'énergie totale consommée dans le réseau.

• Nombre de paquet transmet :

```
hp@hp-HP-Notebook:~/Téléchargements/omnetpp-4.6/samples/Castalia-3.2/Simulations
/zsenario$ CastaliaResults -i 190702-225755.txt -s TXed

Communication.Radio:TXed pkts - TX pkts
+-----+-----+
| General | senario1 | senario2 |
+-----+-----+
| 4499.318 | 3885.727 | 4294.773 |
+-----+-----+
```

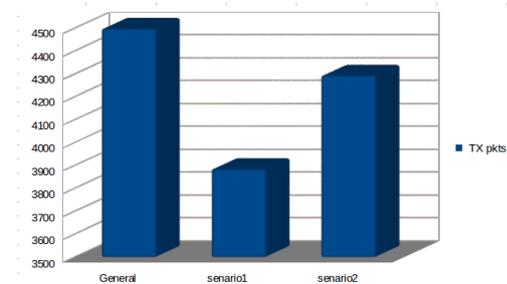


Figure3.12- Le nombre moyen de paquets transmets par chaque nœud.

```
hp@hp-HP-Notebook:~/Téléchargements/omnetpp-4.6/samples/Castalia-3.2/Simulations
/zsenario$ CastaliaResults -i 190702-225755.txt -s TXed --sum

Communication.Radio:TXed pkts - TX pkts
+-----+-----+
| General | senario1 | senario2 |
+-----+-----+
| 98985   | 85486   | 94485   |
+-----+-----+
```

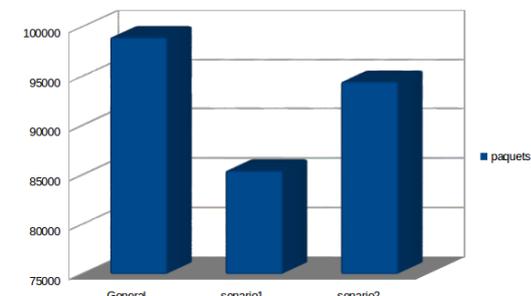


Figure3.13- Le nombre Total des paquets transmets à travers le réseau.

**3.9 Conclusion :**

Dans ce chapitre, nous avons présenté la simulation de la surveillance de réseau de capteurs sans fils avec la redondance des nœuds pour éviter les trous de couverture et assurer une surveillance complète de la zone cible, en cas de défaillance de nœuds ou l'épuisement d'énergie d'un nœud actif.

Les scénarios de simulation avec l'utilisation du simulateur OMNeT++ montrent que notre modèle assure une couverture quasi-complète avec une meilleure optimisation d'énergie.

## Conclusion générale

Durant cette dernière décennie, Les réseaux de capteurs sont devenus un sujet de discussion de nombreuses communautés scientifiques. L'objectif principal des réseaux de capteurs est la collecte, le traitement et la transmission d'un ensemble de grandeurs physiques de l'environnement dans lequel ils se trouvent. Ces opérations sont gourmandes en termes de consommation d'énergie, qui constitue un facteur crucial pour la durée de vie des capteurs et par conséquent du réseau. Le domaine des réseaux de capteurs soulève d'importantes problématiques de recherche en termes de consommation d'énergie, de communication, de surveillance, etc.

Dans ce travail, nous avons proposé un modèle de surveillance des réseaux de capteurs sans fil (RCSF) par redondances des nœuds afin d'améliorer la consommation énergétique au sein du réseau et pouvoir allonger la durée de vie des capteurs.

Afin de valider ce modèle nous avons effectué plusieurs scénarios de simulations et analysé les résultats obtenus, nous avons constaté que plus la surcharge des données par nœud augmente plus l'énergie augmente. En outre, nous avons constaté aussi que plus le nombre des nœuds dans le réseau est élevé plus la consommation d'énergie est élevée, par conséquent ce lourd surcoût de communication aura un impact négatif sur la durée de vie du réseau.

Comme perspective de notre travail nous espérons de développer d'un nouveau protocole de communication appartenant à une des différentes couches de la pile protocolaire des réseaux de capteurs sans fil, garantissant la couverture et de réduire la redondance par la mise en veille périodique des capteurs

## Références :

- [1] O, R. BOUAMARA, " Simulation d'un Réseau de capteur sans fil sous contiki "La localisation d'un mobile" ", Mémoire de Master Professionnel en Informatique, 2015/2016.
- [2] L. NIAR " Analyse Graphique pour la surveillance dans un réseau de capteurs sans fils (RCSF) Simulateur : OMNET++", Mémoire Magister : Informatique, 2012.
- [3] S. Belkheyr, " La Géo-localisation dans les Réseaux de Capteurs sans Fil", Ingénieur d'Etat en Informatique, 2010/2011.
- [4] Y. BENABBASSI, " Application de la redondance pour la surveillance par réseau de capteurs sans fil : Cas du réseau de capteurs image sans fil ", DOCTORAT EN SCIENCES Spécialité : Informatique, 2014.
- [5] une collection de travaux d'états de l'art réalisés par des étudiants doctorants et de master
- [6] L. A. R Amari, "Conception et réalisation d'un simulateur de routage avec qos pour les réseaux wban," master thesis.
- [7] C.-T. Kone, Conception de l'architecture d'un réseau de capteurs sans fil de grande dimension. PhD thesis, Université Henri Poincaré-Nancy I, 2011.
- [8] B. Abdelmalek, C. Youghorta, " Conception et Implémentation d'un Simulateur pour les Réseaux de Capteurs Sans Fil", Mémoire de Master Professionnel Option : Génie Logiciel, 2015/2016.
- [9] KHIATI Mustapha " Impact de la technique duty-cycle sur la diffusion globale dans les réseaux de capteurs sans fil (RCSF) "
- [10] Mohamed BENAZZOUZ "Surveillance de tout point d'une zone d'intérêt à l'aide d'un réseau de capteur multimédia sans fil " magistère IRM 2013
- [11] BENAHMED DAHO Amel « Détection préventive de pannes guidée par les données dans les réseaux de capteurs sans fil » Master en Informatique 2013-2014.
- [12] La couverture dans les réseaux de capteurs sans fil Tiaret.
- [13] Développement d'une bibliothèque de capteurs.
- [14] Kazem Sohraby, Daniel Minoli, Taieb Znati, 'Wireless sensor networks: Technology, Protocols, and Applications', Wiley Interscience Publication, 2007.
- [15] [G. Barrenetxea, "Distributed routing algorithms for sensor networks" - Ecole polytechnique fédérale de Lausanne EPFL – 2006.],
- [16] F. Valois, "Auto-organisation dans les réseaux ad hoc (et les réseaux de capteurs)" – INSA de Lyon – Décembre 2006.
- [17] C. Johnen, L.H. Nguyen, "Self-Stabilizing weight-based Clustering Algorithm for Ad hoc sensor Networks" - LRI– Université Paris Sud – Avril 2006.
- [18] Bouabdellah KECHAR, 'Problématique de la consommation d'énergie dans les réseaux de capteurs sans fil, thèse de doctorat, Université d'Oran, 2010.
- [19] Youssef BENABBASSI, « Application de la redondance pour la surveillance par réseau de capteurs sans fil: Cas du réseau de capteurs image sans fil », DIPLÔME DE DOCTORAT EN SCIENCES Spécialité : Informatique, 2014.