



REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE  
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique  
**Université Mohamed Khider – BISKRA**  
Faculté des Sciences Exactes, des Sciences de la Nature et de la Vie  
**Département d'informatique**

N° d'ordre : RTIC 2 /M2/2020

## Mémoire

Présenté pour obtenir le diplôme de master académique en

# Informatique

Parcours : Réseaux et TIC

## **QUALITE DE SERVICE DANS** **LES RÉSEAUX DE CAPTEURS** **SANS FILS (RCSF)**

Par :

**Badjouia Nassima**

Soutenu le Septembre 2020 .

Foudil Cherif

M C A

Rapporteur

# **Remerciements**

Je tiens premièrement à prosterner remerciant Allah le tout puissant de m'avoir donné le courage et la patience pour terminer ce travail.

Je tiens ensuite à exprimer ma gratitude et mes vifs remerciements au **Pr. Foudil Cherif** pour m'avoir honoré par son encadrement, pour m'avoir bien suivi durant la réalisation de ce travail, pour ses conseils et ses remarques constructives qui m'ont réellement aidé à l'élaboration et l'amélioration de ce mémoire, et surtout pour sa patience.

Je remercie les honorables membres de jury d'avoir accepté d'être membre de mon jury de thèse, d'évaluer mon travail et pour nous avoir honorés de leurs présences.

Un grand merci va également au **Dr. BenAmeur Sabrina** pour ses relectures, et pour ses remarques et ses conseils qui m'ont vraiment permis d'améliorer ce travail.

Je remercie également **Dr. Boukhrouf Djemaa** pour ses encouragements et ses conseils.

Je tenais également à remercier tous mes enseignants de la faculté d'informatique à l'université Mohammed kheider Biskra et plus précisément de la branche RTIC 2020 .

Mes sincères remerciements s'adressent à tous mes collègues du département d'informatique pour leurs encouragements.

Une pensée à l'égard de mes proches qui ont suivi de près ou de loin cette longue aventure .

Aux personnes qui m'aiment, aux personnes que j'aime .

# **TABLE DES MATIERS**

<i>Titre</i>	<i>Page</i>
Résumé .....	11
Introduction générale .....	12
<b>CHAPITRE 1</b>	<b>13</b>
<b>1.1 Introduction</b> .....	<b>14</b>
<b>1.2 Historique d'évolution des réseaux de capteurs sans fil (RCSF)</b> .....	<b>14</b>
<b>1.3 Les réseaux sans fil</b> .....	<b>15</b>
<b>1.3.1 Le réseau avec infrastructure (cellulaire)</b> .....	<b>15</b>
<b>1.3.2 Le réseau sans infrastructure (Adhoc)</b> .....	<b>16</b>
<b>1.4 Les réseaux de capteurs sans fil</b> .....	<b>16</b>
<b>1.4.1 Définition d'un capteur (sensor)</b> .....	<b>17</b>
<b>1.4.2 Architecture d'un nœud capteur</b> .....	<b>17</b>
<b>1.4.2.1 Architecture logicielle</b> .....	<b>19</b>
<b>1.4.3 Caractérisation des réseaux de capteurs</b> .....	<b>20</b>
<b>1.5 Domaines d'applications des RCSFs</b> .....	<b>21</b>
<b>1.5.1 Applications militaires</b> .....	<b>21</b>
<b>1.5.2 Applications environnementales et agricoles</b> .....	<b>22</b>
<b>1.5.3 Applications sanitaires</b> .....	<b>23</b>
<b>1.5.4 Applications commerciales</b> .....	<b>23</b>
<b>1.5.5 Applications de surveillance</b> .....	<b>24</b>
<b>1.5.6 Applications industrielles</b> .....	<b>24</b>
<b>1.5.7 Applications domestiques</b> .....	<b>24</b>
<b>1.6 Types de réseaux de capteurs sans fil</b> .....	<b>25</b>
<b>1.6.1 RCSF terrestre</b> .....	<b>25</b>
<b>1.6.2 RCSF souterrain</b> .....	<b>25</b>
<b>1.6.3 RCSF aquatique</b> .....	<b>26</b>
<b>1.6.5 RCSF mobile</b> .....	<b>28</b>
<b>1.7 Réseau de capteur (RdC) vidéo sans-fils</b> .....	<b>28</b>
<b>1.8 Architecture des Réseau de capteur (RdC) vidéo sans-fils</b> .....	<b>29</b>
<b>1.9 Dispositifs de capture vidéo</b> .....	<b>29</b>

---

1.9.1	Caméras basées sur des composants commerciaux.....	30
1.9.2	Caméras conçues pour les réseaux de capteurs.....	30
1.10	Caractéristiques et contraintes d'un RCVSF.....	33
1.11	Conclusion.....	35
<b>CHAPITRE 2</b>	.....	36
2.1	Introduction .....	37
2.2	Définition de la QoS .....	37
2.3	But de la QoS .....	38
2.4	Les paramètres de qualité de service.....	38
2.4.1	La bande passante (ou débit binaire).....	38
2.4.2	Le délai (ou latence) .....	39
2.4.3	La perte de données .....	39
2.5	Les modèles de QoS .....	40
2.5.1	IntServ (Integrated Service) .....	40
2.5.1.1	Fonctionnement du modèle IntServ.....	41
2.5.2	Modèle Diffserv .....	42
2.5.2.1	Les composants du modèle DiffServ .....	43
2.6	Les classes de services .....	44
2.7	Qualité de service dans les réseaux de capteurs .....	44
2.8	Garantie de la qualité de service dans les RCSF .....	45
2.8.1	Solutions MAC .....	46
2.8.2	Routage .....	46
2.9	Qualité de service pour les RCMSF .....	47
2.9.1	La couche application .....	47
2.9.2	La couche réseau .....	48
2.9.2.1	Protocoles basés sur IntServ .....	49
2.9.2.2	Protocoles basés sur DiffServ .....	50
2.9.2.3	D'autres approches .....	51
2.10	Consommation d'énergie dans les réseaux de capteurs sans fil .....	54
2.10.1	Types Consommation d'énergie dans les RCSF .....	54
2.10.1.1	Energie de capture.....	54
2.10.1.2	Energie de traitement.....	54
2.10.1.3	Energie de communication .....	55

<b>2.10.2</b> Sources de consommation d'énergie .....	55
<b>2.10.2.1</b> La longueur des chemins .....	55
<b>2.10.2.2</b> La qualité des liens .....	55
<b>2.10.2.3</b> Le mode de communication .....	56
<b>2.10.2.4</b> Le routage de paquets inutiles .....	56
<b>2.10.2.5</b> Le choix d'un chemin .....	56
<b>2.11</b> La consommation d'énergie .....	56
<b>2.12</b> Les protocoles de routage dans les RCSF .....	56
<b>2.12.1</b> Introduction.....	56
<b>2.12.2</b> Définition de routage .....	56
<b>2.12.3</b> Routage dans les réseaux de capteurs sans fil .....	57
<b>2.12.4</b> Les Protocoles de routage .....	57
<b>2.12.4.1</b> Protocoles Proactifs .....	57
<b>2.12.4.1.1</b> DSDV .....	58
(Destination Sequenced Distance-Vector Routing)	
<b>2.12.4.2</b> Protocoles Réactifs .....	58
<b>2.12.4.2.1</b> AODV .....	59
(ad hoc On-Demande Distance-Vector)	
<b>2.12.4.2.2</b> DSR (Dynamic Source Routing).....	61
<b>2.13</b> Conclusion.....	61
<b>CHAPITRE 3</b> .....	64
<b>3.1.</b> Introduction.....	65
<b>3.2</b> Préparation de l'environnement .....	66
<b>3.3</b> Outils de simulation .....	66
<b>3.3.1</b> Le simulateur NS-2 .....	66
<b>3.3.2</b> Composants de NS-2.....	67
<b>3.3.3</b> L'architecture NS-2 .....	67
<b>3.3.4</b> Choix du simulateur NS-2 .....	67
<b>3.3.5</b> Arborescence des fichiers .....	68
<b>3.4</b> Processus de simulation .....	69
<b>3.5</b> Paramètres de simulation .....	73
<b>3.6</b> Exemple de Scénarios de simulation .....	73

<b>3.7</b> Scénarios de simulation .....	75
<b>3.7.1.</b> Les Scénarios .....	76
<b>3.7.2</b> Interprétation des résultats .....	76
<b>3.8</b> Conclusion .....	80
<b>3.9</b> Conclusion générale.....	81
<b>3.10</b> Bibliographie.....	82

# Liste des figures

1.1	Evolution des capteurs	14
1.2	Le modèle de réseaux mobiles avec infrastructure	16
1.3	Un réseau mobile Ad Hoc	16
1.4	Quelques capteurs existants sur le marché	17
1.5	Architecture matérielle typique d'un capteur sans fil	18
1.6	Tracé du chemin d'un véhicule militaire	22
1.7	Les RCSFs pour la surveillance des lieux hostiles	23
1.8	Les RCSFs pour les applications sanitaires	23
1.9	Les RCSFs pour les applications commerciales	24
1.10	<i>RCSF</i> terrestre	25
1.11	<i>RCSF</i> souterrain	25
1.12	<i>RCSF</i> aquatique	26
1.13	Architecture de référence d'un <i>RCMSF</i>	27
1.14	Exemple d'un <i>RCVSF</i>	29
1.15	Architecture physique d'un noeud capteur	30
1.16	Prototype de l'architecture <i>MeshEye</i>	31
1.17	<i>Cyclops</i> connecté à un capteur <i>Mica2</i>	31
1.18	<i>CMUcam3</i>	32
1.19	<i>ALOHA</i> connecté à un capteur <i>Mica2</i>	33
2.1	Architecture du modèle IntServ	41
2.2	Opérations de configuration de réservation de ressources de base du protocole de signalisation RSVP	42
2.3	Tâches réalisées dans un modèle d'architecture DiffServ	44
2.4	Exemple de routes formés par RGD avec PathNum = 2	52
2.5	Schéma de routage du protocole présenté dans (Wu et Abouzeid, 2006)	54
2.7	Découverte des chemins dans AODV	59
3.1	Composants de NS-2	67
3.2	Arborescence des fichiers de la distribution NS	68
3.3	Exemple de TCL	70
3.4	Exemple de Nam.	72

2.6	L a répartition de la consommation d'énergie d'un nœud de MicaZ	55
2.7	Découverte des chemins dans AODV	60
2.8	Coupure de route et envoi du RER	60
2.9	Mécanisme de découverte de route dans DSR	61

## Liste des graphes

3.1	Le débit	74
3.2	Taux d'abandon des paquets	75
3.3	Délai moyen de bout en bout de paquets	75
3.4	Le débit - Protocole AODV	77
3.5	Le débit -Protocole DSDV	77
3.6	Le débit -Protocole DSR	77
3.7	Le perte-Protocole AODV	78
3.8	Le perte-Protocole DSDV	78
3.9	: Le perte- Protocole DSR	78
3.10	Le délai- Protocole AODV	79
3.11	Le délai- Protocole DSDV	79
3.12	Le délai-Protocole DSDV	79

# **Liste des tableaux**

Tableau 3.1 : Composants de NS-2 .....	67
Tableau 3.2 : les différents paramètres utilisés.....	76

## *Liste des abréviations*

<b>RCSF</b>	Réseau de Capteurs Sans Fil
<b>RCMSF</b>	Réseau de Capteurs multimédia Sans Fil
<b>SB</b>	Station de Base
<b>RCVSF</b>	Réseau de Capteurs Vidéo Sans Fil
<b>RCMSF</b>	Réseau de Capteurs Multimédias Sans Fil
<b>CMOS</b>	ComplimentayMetalOxideSilicon
<b>UM</b>	unités mobiles
<b>MANET</b>	Mobile Ad hoc NETwork
<b>IETF</b>	Internet Engineering Task Force
<b>DSN</b>	Distributed Sensor Network
<b>DARPA</b>	Defense Advanced Research Projects Agency
<b>WATS</b>	Wide Area Tracking System
<b>JBREWS</b>	Joint Biological Remote Early WarningSystem
<b>QoS (QoS)</b>	Quality of service ( qualité de service )
<b>CLS</b>	Controlled Load Service
<b>GS</b>	Guaranteed Service
<b>RSVP</b>	ReSource Réserveation Protocol
<b>RESV</b>	RESerVation
<b>MAC</b>	Medium Access Control
<b>ISO</b>	International Organization for Standardization
<b>SAR</b>	Single-Aliquot Regenerative-dose
<b>SPEED</b>	Spatiotemporal Communication Protocol for Sensor Networks
<b>RGD</b>	Routage Géographique Rirectionnel
<b>FEC</b>	Forward Error Correction
<b>DSR</b>	Dynamic Source Routing
<b>RREP</b>	Route Reply
<b>RREQ</b>	Route Request
<b>RERR</b>	Route Error Message

<b>AODV</b>	Ad Hoc On-Demand Distance Vector
<b>DSDV</b>	Destination-Sequenced Distance-Vector
<b>NS-2</b>	Network Simulator version 2
<b>NAM</b>	Network Animator
<b>TCL</b>	Tool Command Language
<b>TCP</b>	Transfer control Protocol
<b>UDP</b>	User Datagram Protocol

## Résumé

Les réseaux de capteurs sans fil présentent une nouvelle étape dans l'évolution des technologies de la télécommunication. Contrairement aux réseaux traditionnels qui se préoccupent de garantir une bonne qualité de service, les réseaux de capteurs doivent, en plus, prendre en compte la conservation d'énergie.

Dans la littérature plusieurs protocoles de routage ont été développés, dont ils se distinguent selon la manière de création et de maintenance de routes lors de l'acheminement des données. Dans ce travail, nous nous sommes intéressés à l'analyse des performances des protocoles de routage, et à la qualité de service des réseaux de capteur sans fil pour les différents protocoles de routage. Nous avons donc utilisé l'outil de simulation de réseau NS2 (Network Simulator 2), pour faire une comparaison entre trois protocoles de routage, deux de la même classe (AODV, DSR) et un d'une autre classe (DSDV),

La simulation nous a montré que chaque protocole a sa propre nature et donne des performances variables. Nous avons constaté que le protocole DSR offre de meilleurs résultats et il répond le mieux aux besoins des réseaux de capteur sans fil.

## **Introduction générale :**

Ces dernières années, le développement de la technologie de transmission sans fil a offert de nouvelles perspectives dans le domaine des télécommunications. Les réseaux sans fil constituent un nouveau type de réseaux basés sur cette technologie. Cette dernière occupe une large gamme d'équipements mobiles à savoir : les ordinateurs portables, les téléphones mobiles, les assistants numériques personnels, etc. à l'heure actuelle la recherche dans les réseaux sans fil est plus en plus active. Dans un tel contexte, il n'est pas surprenant de voir apparaître des solutions de communication sans fil de plus en plus performantes et évoluées. Si encore les nœuds communicants sont éloignés, plusieurs nœuds intermédiaires participent à l'acheminement des données vers leurs destinations. Dans les réseaux Ad Hoc, chaque nœud joue le rôle d'un routeur et/ou d'un hôte, alors il doit transmettre les paquets pour les autres nœuds de réseau, d'où la nécessité d'un protocole de routage.

Plusieurs protocoles de routage ont été développés, suivant la manière de création et de maintenance de routes lors de l'acheminement des données. Ces protocoles de routage peuvent être classés en trois catégories : les protocoles réactifs, protocoles actifs et hybrides. Dans le cadre de notre mémoire de fin d'étude, les différentes classes des protocoles de routage proposées sont étudiées pour les réseaux sans fils. Ce travail repose principalement pour évaluer le routage dans les réseaux sans fil, Pour cela nous allons faire une comparaison entre trois protocoles de routage, deux de la même classe (AODV, DSR) et un d'une autre classe (DSDV), chaque nœud est susceptible d'être mis à contribution pour participer au routage et pour retransmettre les paquets d'un nœud qui n'est pas en mesure d'atteindre sa destination, tout nœud joue ainsi le rôle de station et de routeur.

Le problème donc qui se pose dans le contexte des réseaux sans fils est l'adaptation de la méthode d'acheminement utilisée avec le grand nombre d'unités existant dans la mobilité des unités, avec l'assurance de moins de paquets perdus et un débit élevé dans un temps d'acheminement des données le plus court.

C'est dans ce contexte que s'inscrit ce sujet de master, en exploitant le meilleur protocole de routage, les performances des réseaux sans fils seront optimisées en termes de qualité de service.

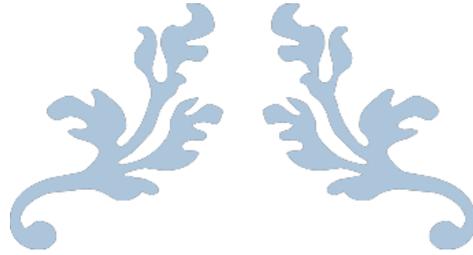
L'objectif de ce projet est d'analyser en détail les performances et la qualité de service des réseaux de capteur sans fil pour les différents protocoles de routage. Pour analyser les performances de tels réseaux, nous avons utilisé l'outil de simulation de réseau NS2 (Network Simulator 2).

Notre mémoire s'articule autour de trois chapitres :

Dans le premier chapitre, nous donnerons une présentation générale des réseaux de capteur sans fil. Dans le deuxième chapitre, nous présenterons les performances en matière de qualité de service(QoS) et les protocoles de routages.

Dans le troisième chapitre nous rendrons compte des aspects liés purement à la simulation et à la programmation puis nous présenterons les résultats de notre simulation.

Enfin, nous finalisons notre mémoire avec une conclusion générale.



**CHAPITRE 1**  
**RÉSEAUX DE**  
**CAPTEURS**  
**SANS FILS (RCSF)**



## 1.1 Introduction :

Les réseaux de capteurs sans fil (*RCSF*) représentent une révolution technologique qui change radicalement la façon de concevoir les systèmes de surveillance de très grandes échelles. En effet, les progrès croissants de la microélectronique et des communications par ondes radio permettent de fabriquer des nœuds capteurs de plus en plus petits, embarquant une unité de traitement de données, un module de transmission sans fil et une batterie. Ces capteurs déployés sur le terrain vont s'organiser en réseau de manière spontanée et collaborer entre eux pour capter des données et les acheminer à l'aide d'un schéma de routage multi-sauts jusqu'à un point de collecte, appelé Station de Base (*SB* ou *Sink*).

Parmi toutes les applications potentielles des *RCSF*, celles utilisant des capteurs vidéo sont appréciables pour tout ce qui concerne la reconnaissance, la localisation et le dénombrement d'objets par la vision. L'émergence des réseaux de capteurs vidéo sans fil (*RCVSF*) suscite le développement de nouveaux protocoles pour le transfert des données multimédias et de mécanismes de gestion de l'énergie qui soient adaptés aux exigences sévères requises.

Dans ce chapitre, nous présentons les principes de conception des *RCVSF* ainsi que leurs différentes caractéristiques et domaines de recherches. [02]

## 1.2 Historique d'évolution des réseaux de capteurs sans fil (RCSF) :

Au fil des années, avec la disponibilité du matériel et dispositifs de prix abordable, des capteurs de plus en plus sophistiqués ont été proposés, permettant de capturer des données scalaires ou encore multimédias dans le but de satisfaire une grande panoplie d'applications sur les RCSF.

Dans ce qui suit, nous présentons une évolution au cours du temps des capteurs les plus utilisés ces dernières années.

Gamme de capteur

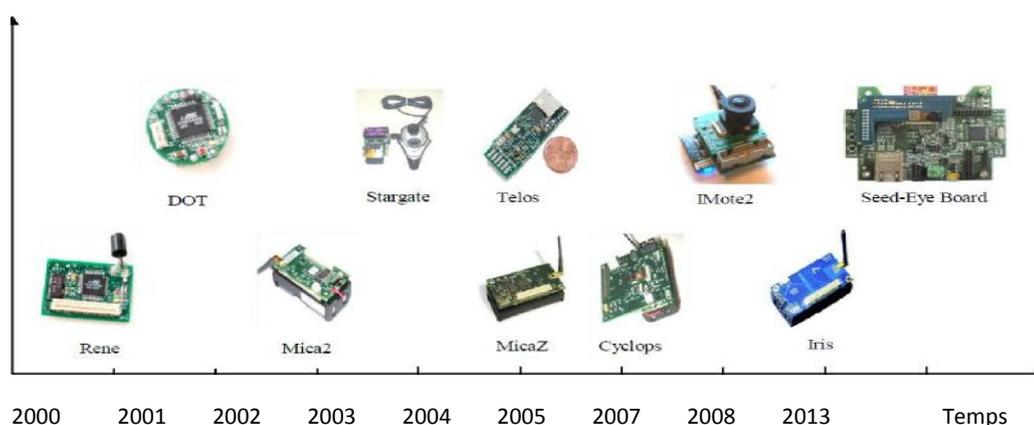


Figure 1.1: Evolution des capteurs [02]

Comme l'illustre la figure I.1, plusieurs capteurs ont été proposés par différentes universités comme celle de Berkeley qui a présenté plusieurs travaux sur les *RCSF*. Les plus utilisés sont les capteurs fabriqués par l'entreprise *Xbow* (aussi appelé *Crossbow*) qui est née au sein de la célèbre université californienne. Cette entreprise a revendu en 2010 son activité réseau de capteurs à la société sino-américaine *MEMSIC*. Ces capteurs conçus pour collecter des données scalaires telles que la température ou l'humidité se basent sur le composant *Chipcon CC2420* qui est devenu le standard au niveau des modules de transmission utilisant le protocole de communication *IEEE 802.15.4*.

Avec les exigences des applications critiques de surveillance, une nouvelle catégorie de capteurs dotés de capacités multimédias a vu le jour, Les capteurs conçus initialement pour collecter des données scalaires peuvent contribuer à la mise en œuvre des capteurs multimédias. Pour la transmission sans fils, par exemple, l'émetteur/récepteur des capteurs *Mica2* ou *MicaZ* peut être utilisé via l'interface du capteur d'images *Cyclops* afin de transmettre les images capturées.

Plus récemment, le capteur *Seed-Eye* a vu le jour, développé par la société *Evidence Embedding Technology*, il a été conçu pour l'implémentation des *RCMSF* à faible coût, il est constitué de plusieurs interfaces de communication telles que *Ethernet*, *IEEE 802.15.4/Zigbee* et *USB*. Il contient de plus une caméra *CMOS* pour la capture d'images.[02]

### **1.3 Les réseaux sans fil :**

Les réseaux sans fil se divisent en deux catégories, le réseau avec infrastructure et les réseaux sans infrastructure.

#### **1.3.1 Le réseau avec infrastructure (cellulaire)**

Un réseau mobile avec infrastructure est composé des sites mobiles. Il permet à ses utilisateurs d'accéder à l'information indépendamment de leurs positions géographiques. Dans cette classe, on distingue deux ensembles d'entités distinctes : les sites fixes d'un réseau de communication filaire classique (wired network) et les sites mobiles (wireless network). Certains sites fixes, appelés stations de support mobile (Mobile Support Station) ou stations de base (SB), sont munis d'une interface de communication sans fil pour la communication directe avec les sites ou les unités mobiles (UM) localisés dans une zone géographique limitée, appelée cellule (voir figure (1-2)).

Chaque station de base correspond une cellule à partir de laquelle des unités mobiles peuvent émettre et recevoir des messages. Alors que les sites fixes sont interconnectés entre eux à travers un réseau de communication filaire, qui se caractérise généralement par sa fiabilité et son débit élevé, par contre, les liaisons sans fil ont une bande passante limitée qui réduit sévèrement le volume des informations échangées.

Dans ce modèle, une unité mobile ne peut être, à un instant donné, directement connectée qu'à une seule station de base. Elle peut communiquer avec les autres sites à travers la station laquelle elle est directement rattachée. L'autonomie réduite de sa source d'énergie, lui occasionne de fréquentes déconnexions du réseau, sa reconnexion peut alors se faire dans un nouveau environnement voire dans une nouvelle localisation. [01]

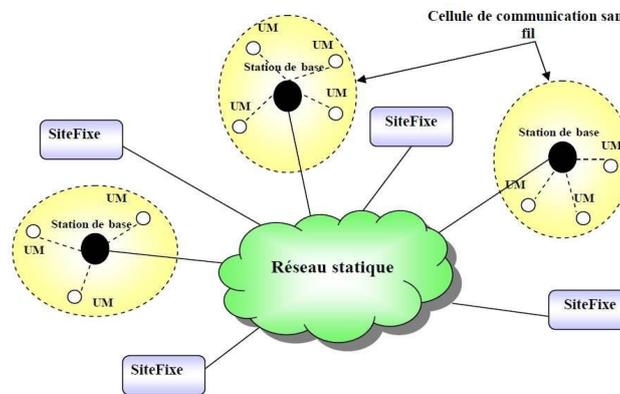


Figure 1.2 – Le modèle de réseaux mobiles avec infrastructure[01]

### 1.3.2 Le réseau sans infrastructure (Adhoc)

Le modèle de réseau sans infrastructure préexistante ne comporte pas l'entité site fixe, tous les sites du réseau sont mobiles et se communiquent d'une manière directe en utilisant leurs interfaces de communication sans fil. L'absence d'infrastructure ou de réseau filaire des stations de base oblige les unités mobiles à se comporter comme des routeurs qui participent à la découverte et la maintenance des chemins pour les autres hôtes du réseau.

Le groupe MANET (Mobile Ad hoc NETwork) de l'IETF (Internet Engineering Task Force). Un réseau Ad Hoc comprend des plates-formes mobiles appelées nœuds qui sont libres de se déplacer sans contrainte.

Un réseau ad hoc est donc un système autonome de nœuds mobiles. Ce système peut fonctionner d'une manière isolée ou s'interfacer à des réseaux fixes à travers des passerelles. [01]

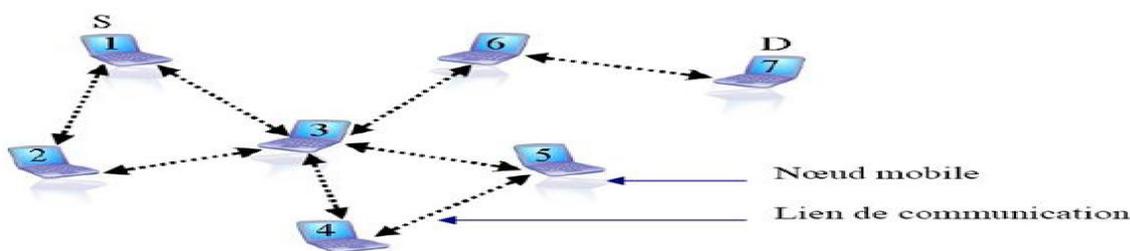


Figure 1.3-Un

réseau mobile Ad Hoc [01]

### 1.4 Les réseaux de capteurs sans fil

Les réseaux de capteurs sans fil sont considérés comme un type spécial des réseaux Ad hoc où l'infrastructure fixe de communication et l'administration centralisée sont absentes et les nœuds jouent, à la fois, le rôle des hôtes et des routeurs. Ce type de réseaux consiste en un ensemble de micro-capteurs éparpillés aléatoirement à travers une zone géographique qui définit le terrain d'intérêt pour le phénomène capté. Les micro-capteurs déployés sont capables de surveiller, d'une manière continue, une grande variété de conditions ambiantes telles que la température, l'humidité, et de détecter également l'occurrence des événements tel que les séismes. Malgré leur capacité limitée de captage et de traitement de données, qui n'est qu'une conséquence de leur taille miniaturisée (de l'ordre de  $1\text{cm}^3$ ), les composants de communication sans fil intégrés à ces capteurs leur permettent de collaborer et de coordonner entre eux afin d'accomplir des tâches de captage complexes. [01]

### **1.4.1 Définition d'un capteur (sensor)**

Un capteur est un petit appareil autonome capable d'effectuer de simples mesures sur son environnement immédiat, telles que la température, les vibrations et la pression. Cet appareil est augmenté de capacités de calcul et de communication ainsi que de batteries lui conférant une autonomie. [01]

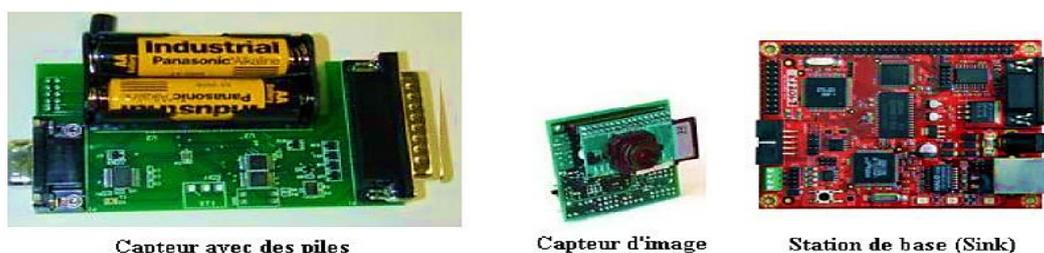


Figure 1.4 – Quelques capteurs existants sur le marché. [01]

### **1.4.2 Architecture d'un nœud capteur**

L'évolution technologique nous permet d'accepter et de conforter la présence de l'informatique et de l'électronique au cœur du monde réel. Nous voyons de nos jours de plus en plus d'objets électroniques et d'objets connectés équipés de différents modules permettant de traiter les informations mais également de les retransmettre. La technologie des réseaux de capteurs entre dans cette catégorie. En effet, il s'agit d'un ensemble d'objets, ou capteurs, ayant des ressources particulièrement limitées, mais qui néanmoins ont la possibilité d'acquérir des données de l'environnement dans lequel ils sont déployés, de les traiter et aussi de les transmettre. Les capteurs sont des objets de taille réduite avec des ressources très limitées, qui sont autonomes et sont capables de traiter les informations de l'environnement qui les entoure et de les transmettre à d'autres dispositifs grâce aux ondes radios sur une distance limitée. Les RCSF se composent d'un grand nombre de capteurs, pour former un réseau sans infrastructure établie. Un capteur a pour objectif d'analyser l'environnement dans lequel il est déployé, et transmet les données récoltées aux différents nœuds capteurs se trouvant dans sa zone de couverture. Ainsi de suite, chaque nœud relaye

l'information à un point de collecte où toutes les données sont recueillies. Chaque capteur relayant l'information sur sa propre zone de couverture, l'environnement où sont déployés les capteurs se trouve entièrement couvert.

Un nœud capteur contient quatre unités de base : l'unité de capture, l'unité de traitement, l'unité de transmission, et l'unité de contrôle d'énergie. Il peut contenir également, suivant son domaine d'application, des modules supplémentaires tels qu'un système de localisation (GPS), ou bien un système générateur d'énergie (cellule solaire). On peut même trouver des micro-capteurs, un peu plus volumineux, dotés d'un système mobilisateur chargé de déplacer le micro-capteur en cas de nécessité. La figure 1.5 présente les différents composants qui constituent un capteur sans fil. Pour être plus précis chaque groupe de composants possède son propre rôle. [03]

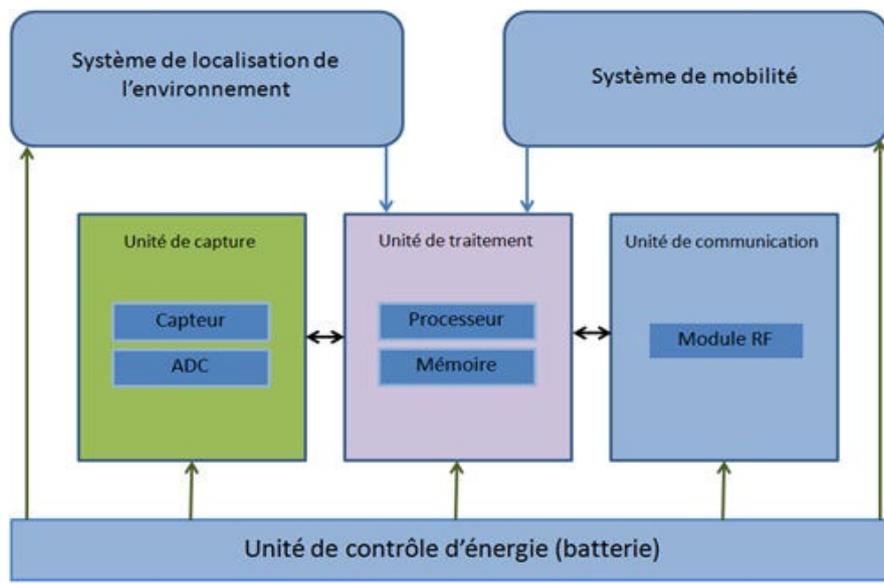


FIGURE 1.5 – Architecture matérielle typique d'un capteur sans fil[03]

#### - **Unité de capture (Sensing Unit) :**

On retrouve donc des équipements de différents types de détecteur et d'autre entrée. Le capteur est généralement composé de deux sous-unités : le récepteur (reconnaissant l'analyse) et le transducteur (convertissant le signal du récepteur en signal électrique). Le capteur est responsable de fournir des signaux analogiques, basés sur le phénomène observé, au convertisseur Analogique/Numérique. Ce dernier transforme ces signaux en un signal numérique compréhensible par l'unité de traitement. [03]

- **Unité de communication (Transceiver Unit) :** Les équipements étudiés sont donc généralement équipés d'une radio ainsi que d'une antenne. Cette unité est responsable d'effectuer toutes les émissions et réceptions des données sur un médium sans fil. Elle peut-être de type optique (comme dans les nœuds Smart Dust), ou de type radiofréquence. Les communications de type optique sont robustes vis-à-vis des interférences électriques. Néanmoins, elles présentent l'inconvénient d'exiger une ligne de vue permanente entre les entités communicantes. Par conséquent, elles ne peuvent pas établir de liaisons à travers des obstacles.[03]

**- Unité de traitement (Processing Unit) :**

L'unité de traitement est généralement composée d'une mote, d'un processeur, d'une mémoire RAM et Flash. On appelle généralement mote la carte physique utilisant le système d'exploitation pour fonctionner. Celle-ci a pour cœur le bloc constitué du processeur et des mémoires RAM et flash. Cet ensemble est la base du calcul binaire et du stockage, temporaire pour les données et définitif pour le système d'exploitation. Cette unité est chargée d'exécuter les protocoles de communications qui permettent de faire collaborer le nœud avec les autres entités du réseau. Elle peut aussi analyser les données captées pour alléger la tâche du nœud puits. [03]

**- Unité de contrôle d'énergie (Power Unit) :**

Un micro-capteur est muni d'une ressource énergétique pour alimenter tous ses composants qui se présentent généralement sous forme d'une batterie (par exemple pile AA normale d'environ 2.2-2.5 Ah, fonctionnant à 1.5 V). Cependant, en conséquence de sa taille réduite, la ressource énergétique dont il dispose est limitée et il est généralement irremplaçable. Cette unité peut aussi gérer des systèmes de rechargement d'énergie à partir de l'environnement observé telles que les cellules photovoltaïques, afin d'étendre la durée de vie totale du réseau. Cependant quelques différences existent suivant les fabricants. [03]

**- Système de localisation (Location Finding System) :** Il fournit des informations sur la localisation requise par les techniques de routage. [01]

**- Unité de mobilité (Mobiliser) :** Il est appelé si le nœud capteur doit être déplacé pour accomplir la requête à traiter. [01]

### **1.4.2.1 Architecture logicielle**

Un capteur est doté d'un système d'exploitation open source appelé TinyOS qui est une série d'outils développés par l'université de Berkeley et enrichi par une multitude d'utilisateurs.

TinyOS est en grande partie s'appuie sur un langage NesC est conçu pour les réseaux de capteurs sans fil. Il respecte une architecture basée sur une association de composants.

La bibliothèque de composants de Tiny OS est particulièrement complète puisqu'elle inclut des protocoles réseaux, des pilotes de capteurs et des outils d'acquisition de données. En s'appuyant sur un fonctionnement évènementiel, TinyOS propose à l'utilisateur une gestion très précise de la consommation du capteur et permet de mieux s'adapter à la nature aléatoire de la communication sans fil entre les interfaces physiques. [01]

### 1.4.3 Caractérisation des réseaux de capteurs

La réalisation d'un réseau de capteur utilise les techniques des réseaux Ad hoc. Cependant, les protocoles et les algorithmes proposés dans ce dernier ne conviennent pas aux réseaux de capteurs sans fil. Voici donc quelques caractéristiques spécifiques à ces réseaux :

#### **- Topologie dynamique des réseaux de capteurs sans fil :**

La mobilité continue des nœuds crée un changement dynamique de topologie. Par exemple, un nœud peut rejoindre un réseau, changer de position ou quitter le réseau (suppression de capteurs à cause de défaillances ou autres choses, etc). Ce déplacement a naturellement un impact sur la morphologie du réseau.

#### **- Absence d'infrastructure :**

Les réseaux de capteurs en général, se distinguent des autres réseaux mobiles par la propriété d'absence d'infrastructure préexistante et de tout genre d'administration centralisée à l'exception des réseaux ad hoc. Les hôtes mobiles sont responsables d'établir et de maintenir la connectivité du réseau d'une manière continue.

#### **- Un grand nombre de capteurs :**

Des réseaux de 10000 nœuds peuvent être envisagés.

#### **- Sécurité physique limitée :**

Les réseaux de capteurs sans fil mobiles 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.

#### **- Auto déploiement :**

Les nœuds capteurs peuvent être déployés dans l'environnement sans intervention humaine et demeureront sans surveillance pendant longtemps après déploiement.

#### **- Auto configuration :**

Consiste avant tout à permettre la communication des nœuds, et donc l'affectation d'adresses aux interfaces réseaux, la diffusion des préfixes des sous réseaux et des passerelles d'interconnexion. Il s'agit ensuite de diffuser les informations comme le DNS aux nœuds du réseau.

#### **- Auto-organisation :**

Cherche à structurer la topologie du réseau en tirant partie des propriétés des nœuds tels que l'énergie résiduelle, la densité, etc. Par exemple, est-il préférable de proposer une architecture à plat du réseau ou au contraire est-il nécessaire de proposer des mécanismes regroupant les nœuds suivant des critères comme les services, les interfaces radio, les capacités, l'énergie disponible, etc. L'auto-organisation doit également répondre à la prise en compte efficace de la dynamique du réseau.

#### **- Autogestion :**

A pour objectif de proposer une supervision autonome : une fois un réseau spontané auto-configuré et

auto-organisé, il est nécessaire de fournir des algorithmes et des protocoles permettant sa surveillance, sa maintenance réactive.

#### **- Contrainte d'énergie :**

Dans plusieurs applications, les nœuds de capteurs sont placés dans des surfaces distantes, le service du nœud peut ne pas être possible, dans ce cas la durée de vie du nœud peut être déterminée par la vie de la batterie, ce qui exige la minimisation des dépenses énergétiques.

### **1.5 Domaines d'applications des RCSFs**

La miniaturisation des capteurs, le coût de plus en plus faible, la large gamme des types de capteurs disponibles ainsi que le support de communication sans fil utilisé, permettent aux réseaux de capteurs de se développer dans plusieurs domaines d'application, Ils permettent aussi d'étendre les applications existantes.

Les réseaux de capteurs peuvent se révéler très utiles dans de nombreuses applications lorsqu'il s'agit de collecter et de traiter des informations provenant de l'environnement. Parmi les domaines où ces réseaux peuvent offrir les meilleures contributions, nous citons les domaines : militaire, surveillance, environnemental, médical, domestique, commercial, etc.[03]

#### **1.5.1 Applications militaires**

Le faible coût et le déploiement rapide sont des caractéristiques qui ont rendu les réseaux de capteurs efficaces pour les applications militaires. Plusieurs projets ont été lancés pour aider les unités militaires dans un champ de bataille et protéger les villes contre des attaques, telles que les menaces terroristes. Le projet DSN (Distributed Sensor Network) au DARPA (Defense Advanced Research Projects Agency) était l'un des premiers projets dans les années 80 ayant utilisé les réseaux de capteurs pour rassembler des données distribuées. Les chercheurs du laboratoire national (Lawrence Livermore) ont mis en place le réseau WATS (Wide Area Tracking System), Ce réseau est composé de détecteurs des rayons gamma et des neutrons pour détecter et dépister les dispositifs nucléaires. Il est capable d'effectuer la surveillance constante d'une zone d'intérêt. Il utilise des techniques d'agrégation de données pour les rapporter à un centre intelligent. Ces chercheurs ont mis en place ensuite un autre réseau appelé JBREWS (Joint Biological Remote Early Warning System) pour avertir les troupes dans le champ de bataille des attaques biologiques possibles. Un réseau de capteurs peut être déployé dans un endroit stratégique ou hostile, afin de surveiller les mouvements des forces ennemies, ou analyser le terrain avant d'y envoyer des troupes (détection des armes chimiques, biologiques, radiations). L'armée américaine a réalisé des tests dans le désert de Californie. [03]



Figure 1.6 – Tracé du chemin d'un véhicule militaire. [03]

### **1.5.2 Applications environnementales et agricoles**

Le contrôle des paramètres environnementaux par les réseaux de capteurs peut donner naissance à plusieurs applications. Par exemple, le déploiement des thermo-capteurs dans une forêt peut aider à détecter un éventuel début de feu et par suite faciliter la lutte contre les feux de forêt avant leur propagation. Le déploiement des capteurs chimiques dans les milieux urbains peut aider à détecter la pollution et analyser la qualité d'air. De même leur déploiement dans les sites industriels empêche les risques industriels tels que la fuite de produits toxiques (gaz, produits chimiques, éléments radioactifs, pétrole, etc.). Dans le domaine de l'agriculture, les capteurs peuvent être utilisés pour réagir convenablement aux changements climatiques par exemple le processus d'irrigation lors de la détection de zones sèches dans un champ agricole. Cette expérimentation a été réalisée par Intel Laboratoire de recherche et Agriculture et Agri-FoodCanada sur une vigne à British Columbia. L'avantage principal de l'utilisation des RCSF dans l'agriculture est l'absence de câblage ce qui réduit considérablement le coût d'installation. Un autre avantage est la flexibilité de déploiement et la facilité d'entretien. En effet, les capteurs sont autonomes et nécessitent très peu d'intervention humaine sur les champs notamment dans le cas où les protocoles de communication sont tolérants aux panne et supportent la mobilité des nœuds. Plusieurs projets ont utilisé la technologie des RCSF dans l'agriculture.

Nous pouvons également prendre l'exemple du projet AGROSENS, lancé par le laboratoire (Heudiasyc ,qui signifie HEUristique et DIAgnostic des Systèmes Complexes), est une unité mixte de recherche entre l'Université de Technologie de Compiègne et le CNRS, qui a pour objectif l'étude et la conception d'une architecture de réseaux de capteurs sans fil (RCSF) pour le contrôle de l'environnement et le développement de pratiques d'agriculture de précision. [03]



Figure 1.7 – Les RCSFs pour la surveillance des lieux hostiles. [03]

### **1.5.3 Applications sanitaires**

Les RCSFs sont récemment très répandus dans le secteur de la santé. Dans certains hôpitaux, ils sont installés pour surveiller des données physiologiques des patients et pour surveiller aussi comportement des patients et des médecins à l'intérieur d'un hôpital.



Figure 1.8 – Les RCSFs pour les applications sanitaires.

Dans le projet : les capteurs intelligents et les microsystèmes intégrés (SSIM), la puce prothèse rétine qui est constituée de 100 micros capteurs est construite et implémentée dans l'œil humain. Cette application a permis aux patients non-voyants ou à vision limitée de voir ou d'améliorer leur niveau de vision.

Quelques autres applications ont été envisagées à savoir : la surveillance de niveau du glucose ,le contrôle des organes, la détection du cancer, etc. L'idée d'intégrer des capteurs biomédicaux sans fil dans le corps humain est prometteuse, bien qu'ils existent beaucoup de défis et de contraintes à prendre au sérieux : le système doit être ultra prudent, sans danger sur la santé du patient et stable et nécessite peu de maintenance.

Avec beaucoup de recherches et de progressions dans ce domaine, un coût de traitement médical peut être réduit et une qualité de vie meilleure est promise.[01]

### **1.5.4 Applications commerciales :**

Il est possible d'intégrer des capteurs au processus de stockage et de livraison dans le domaine commercial

Le réseau ainsi formé pourra être utilisé pour connaître la position, l'état et la direction d'un paquet. Il devient alors possible pour un client qui attend la réception d'un paquet, d'avoir un avis de livraison en temps réel et de connaître la localisation actuelle du paquet. Pour les entreprises manufacturières, les réseaux de capteurs permettront de suivre le procédé de production à partir des matières premières jusqu'au produit final livré. [03]



Figure 1.9 – Les RCSFs pour les applications commerciales. [03]

### **1.5.5 Applications de surveillance**

L'application des réseaux de capteurs dans le domaine de la sécurité peut diminuer considérablement les dépenses financières consacrées à la sécurisation des lieux et des êtres humains. Ainsi, l'intégration des capteurs dans de grandes structures telles que les ponts ou les bâtiments aidera à détecter les fissures et les altérations dans la structure suite à un séisme ou au vieillissement de la structure. Le déploiement d'un réseau de capteurs de détection de mouvement peut constituer un système d'alarme qui servira à détecter les intrusions dans une zone de surveillance. [03]

### **1.5.6 Applications industrielles**

Les machines de production peuvent être configurées et pilotées en interconnexion grâce à des systèmes d'acquisition et réseaux de capteurs sans fil, autonomes et communicants. Grâce à ce système, la connaissance précise et en continu de l'état de santé des machines est la garantie d'un maintien en conditions d'exploitation optimale (condition monitoring), d'interventions de maintenance réalisées juste au bon moment, et d'économies conséquentes. [03]

### **1.5.7 Applications domestiques**

Avec le développement technologique, les capteurs peuvent être embarqués dans des appareils, tels que les aspirateurs, les fours à micro-ondes, les réfrigérateurs, les magnétoscopes, etc. Ces capteurs embarqués peuvent interagir entre eux et avec un réseau externe via Internet pour permettre à un utilisateur de contrôler les appareils domestiques localement ou à distance. Le déploiement des capteurs de

mouvement et de température dans les futures maisons dites intelligentes permet d'automatiser plusieurs opérations domestiques telles que :

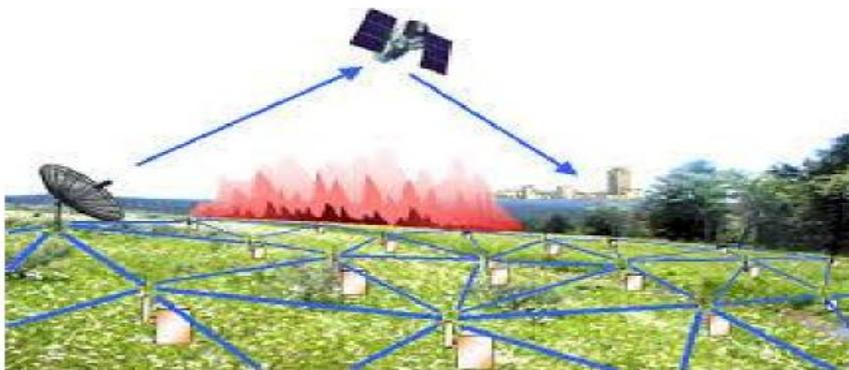
La lumière s'éteint et la musique s'arrête quand la chambre est vide, la climatisation et le chauffage s'ajustent selon les points multiples de mesure, l'alarme est déclenchée par le capteur anti-intrusion quand un étranger veut pénétrer dans la maison. [03]

## **1.6 Types de réseaux de capteurs sans fil**

Actuellement, plusieurs *RCSF* sont déployés sur terrain, sous terre et sous l'eau. Ils font face à différents challenges et contraintes dépendant de leur environnement. Nous pouvons distinguer les cinq types de *RCSF* suivants. [02]

### **1.6.1 RCSF terrestre**

Un *RCSF* terrestre consiste en un grand nombre (centaines à milliers) de nœuds à faible coût qui sont déployés sur terrain dans une zone donnée, de manière ad hoc (déployé à partir d'un hélicoptère par exemple). Dans les *RCSF* terrestres, les nœuds capteurs doivent être capables de transmettre des données à une *SB* dans un environnement dense. Puisque l'énergie de la batterie est limitée et la plupart du temps inaccessible, les nœuds capteurs terrestres peuvent être équipés d'une source d'énergie secondaire telle que les cellules solaires. Les applications communes des *RCSF* terrestres sont la capture et le monitoring environnemental (voir figure 1.10), le monitoring industriel et les explorations de surfaces.



**Figure 1.10:** *RCSF* terrestre

### **1.6.2 RCSF souterrain**

Un *RCSF* souterrain comporte un ensemble de nœuds capteurs qui sont déployés sous terre, dans des caves ou des mines pour contrôler les conditions souterraines. Afin de transmettre des données à partir des nœuds capteurs souterrains jusqu'à la *SB*, des nœuds *Sinks* additionnels seront situés à la surface (voir figure 1.11). Ces nœuds capteurs coûtent plus cher que les capteurs terrestres car ils nécessitent des équipements appropriés pour assurer une communication fiable à travers la terre, les rochers et l'eau. La

communication sans fil est un challenge dans de tels environnements à cause de la forte atténuation et perte du signal. De plus, il est difficile de recharger ou de remplacer les batteries des capteurs sous terre, ce qui nécessite l'utilisation de protocoles de communication à économie d'énergie pour prolonger la durée de vie du réseau. Les capteurs souterrains sont utilisés dans plusieurs applications telles que le monitoring agricole, le monitoring souterrain du sol et de l'eau et la surveillance des frontières militaires.

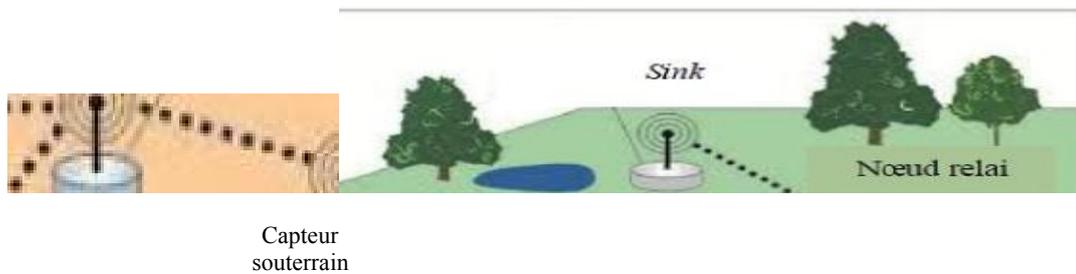


Figure 1.11: RCSF souterrain

### 1.6.3 RCSF aquatique

Un *RCSF* aquatique se compose de capteurs déployés sous l'eau, par exemple, dans un environnement océanique. Comme ces nœuds coûtent très cher, donc très peu de nœuds capteurs seront déployés et des véhicules aquatiques autonomes seront utilisés pour explorer ou récupérer les données capturées. Les communications sans fil aquatiques utilisent des ondes acoustiques (voir figure 1.12), qui présentent différentes contraintes telles qu'une bande passante limitée, un délai de propagation plus long, une haute latence et des problèmes d'affaiblissement du signal. Ces nœuds doivent être capables de s'auto-configurer et de s'adapter aux conditions extrêmes de l'environnement océanique. Les applications des *RCSF* aquatiques incluent le contrôle de la pollution, la surveillance et l'exploration sous-marine, la prévention et le monitoring de désastres tels que les séismes.

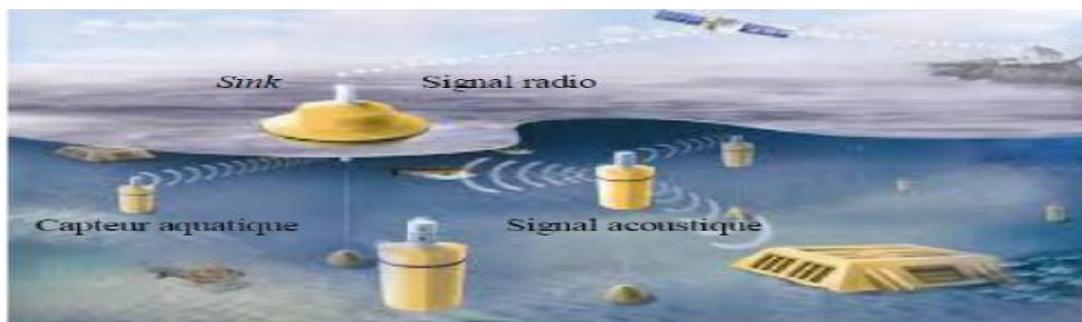


Figure 1.12: RCSF aquatique

### 1.6.4 RCSF multimédia

Il s'agit d'un ensemble de nœuds capteurs à faible coût équipés de caméras ou de microphones, déployés de manière prédéterminée ou aléatoire pour garantir une couverture. Les dispositifs de capteurs multimédias sont capables de stocker, traiter et de récupérer des données multimédias telles que la vidéo, l'audio et l'image. Ils doivent faire face à différents challenges tels qu'une forte demande de bande

passante, une grande consommation d'énergie, une garantie de *QoS* et des techniques de traitement et de compression des données. Il est nécessaire alors de développer des techniques de transmission qui supportent une bande passante assez élevée et une faible consommation d'énergie pour la délivrance fiable de contenu multimédia tel qu'un streaming vidéo. Les *RCMSF* améliorent les applications de *RCSF* existantes comme le suivi et la surveillance.

Une architecture de référence d'un *RCMSF* proposée est présentée sur la figure 1.13 où trois types de *RCSF* avec différentes caractéristiques sont illustrés. Les capteurs multimédias peuvent être déployés dans un réseau à un seul niveau, comme indiqué sur les figures I.13(a) et I.13(b), ou d'une manière hiérarchique, comme le montre la figure I.13(c). Les propriétés des deux types de niveaux sont détaillées sur la figure 1.13. Les hubs de traitement multimédia sont utilisés dans cette architecture car ils disposent de ressources importantes en calcul, et sont appropriés pour l'agrégation des flux multimédia à partir des nœuds capteurs. Ils permettent la réduction de la dimension et le volume des données transmises au *Sink* et les dispositifs de stockage. Enfin, une architecture plus complexe d'un *RCMSF* peut être implémentée en fonction des besoins de l'application.

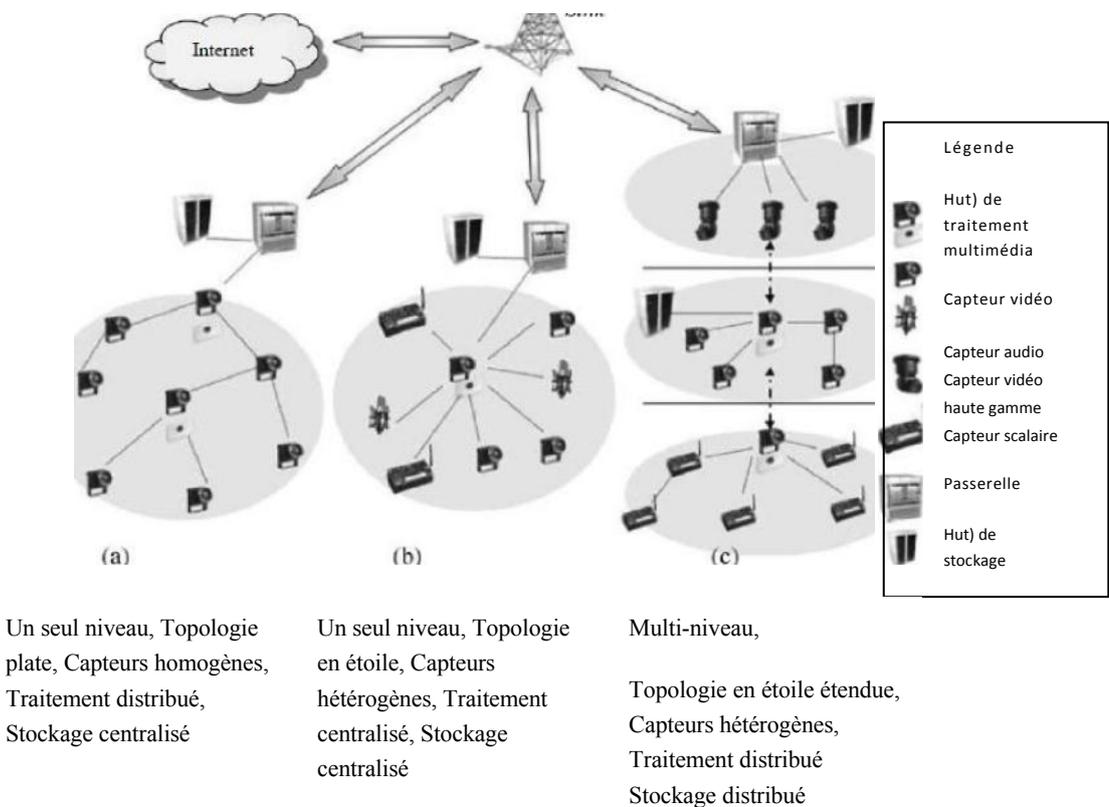


Figure 1.13: Architecture de référence d'un *RCMSF*

### **1.6.5 RCSF mobile**

Un *RCSF* mobile se compose de nœuds capteurs mobiles (voir figure 1.6), qui peuvent se déplacer et interagir avec un environnement physique. Les nœuds mobiles peuvent se repositionner et s'auto-organiser dans le réseau de plus de leur capacité de capturer, traiter et de communiquer. Un algorithme de routage dynamique doit être utilisé au lieu du routage fixe approprié aux *RCSF* statiques. Les *RCSF* mobiles doivent faire face à plusieurs challenges tels que le déploiement, la gestion de la mobilité, la localisation avec mobilité, le contrôle et la navigation des nœuds mobiles, le maintien de la couverture et de la connectivité et la minimisation de la consommation d'énergie dans la locomotion. Les applications des *RCSF* mobiles concernent le monitoring environnemental, la surveillance militaire, le suivi de cibles et l'assistance aux opérations de secours. Un plus grand degré de couverture et de connectivité peut être assuré avec les nœuds capteurs mobiles comparé aux nœuds statiques.

### **1.7 Réseau de capteur (RdC) vidéo sans-fils :**

Un réseau de capteur (RdC) vidéo sans-fils consiste en un ensemble de nœuds autonomes dotés d'une petite caméra embarquée. Ce type de réseau permet de déployer un grand nombre d'applications.

Les applications sur RdC qui sont orientées surveillance ont des besoins spécifiques du fait de leur criticité. Par exemple, la qualité des images capturées et transmises doit être en adéquation avec les besoins et objectifs de l'application. De telles infrastructures de surveillance perdent très rapidement leurs intérêts si les scènes importantes ne sont pas correctement capturées.

Un aspect important en surveillance est la couverture et son maintien. Ce problème a été largement étudié dans les systèmes multi-robots et les RdCs de type scalaire en considérant une capacité de capture omnidirectionnelle. Ainsi 2 nœuds sont considérés comme redondants s'ils sont proches l'un de l'autre. Dans les RdCs vidéo, les caméras possèdent un champ de vision (CdV) et éventuellement des capacités de zoom. Deux nœuds peuvent ainsi être redondants même s'ils sont relativement éloignés l'un de l'autre. Parfois plusieurs vues sont souhaitables pour résoudre les ambiguïtés, parfois les nœuds éloignés peuvent fournir des informations mieux exploitables en fonction des conditions météo par exemple.

Un autre aspect très important est celui de la gestion de l'énergie. La rareté de cette ressource va avoir un impact très fort sur la couverture puisqu'il n'est pas réaliste que tous les nœuds puissent être actifs en même temps. Lorsque le déploiement est aléatoire, il peut y avoir une grande redondance entre les nœuds et une approche couramment utilisée est de définir un sous-ensemble de nœuds qui seront actifs pendant que d'autres seront inactifs. Le résultat est un ordonnancement de l'activité des nœuds qui maintient la couverture et la connectivité de la zone à surveiller. De plus, il est souhaitable de pouvoir définir plusieurs niveaux d'activité qui peuvent correspondre au nombre d'images qui seront capturées par unité de temps. Par exemple, certaines applications se focalisent sur la surveillance de la zone frontière plutôt que de la zone intérieure. Dans ce cas, les nœuds se situant à la frontière doivent capturer les images à un rythme

plus soutenu que ceux se situant à l'intérieur, qui, à la rigueur, peuvent même se mettre en veille. Disposer de plusieurs niveaux d'activité est aussi nécessaire car les applications de surveillance comme la détection d'intrusion doivent pouvoir être opérationnelles sur le long terme puisque personne ne sait quand une intrusion se produira. [04]

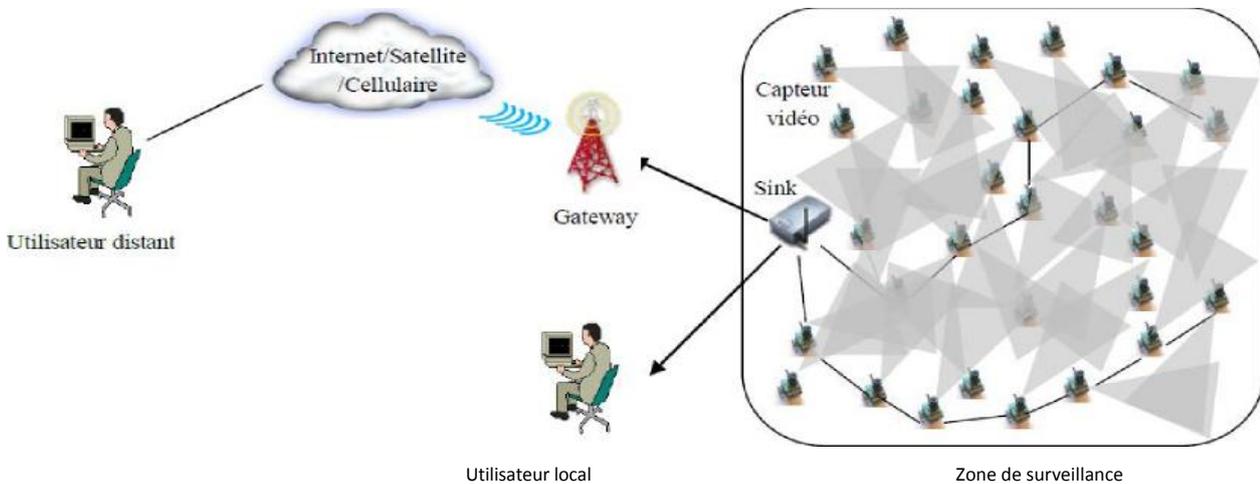


Figure 1.14: Exemple d'un RCVSF [04]

### 1.8 Architecture des Réseau de capteur (RdC) vidéo sans-fils

En plus de sa fonction initiale qui est de collecter des données, un capteur est doté d'une unité de traitement embarquée à faible coût avec des capacités de calcul, de mémoire et de communication. La capacité d'un capteur est néanmoins limitée, ceci à cause en partie de la miniaturisation de ses composants. Un capteur se compose en général de quatre unités fondamentales (illustrées sur la figure 1.15) : [04]

- **Unité de capture** : contient le ou les capteurs embarqués sur le nœud afin d'assurer la collecte des données de l'environnement. Les signaux provenant du capteur sont convertis à l'aide d'un convertisseur analogique/numérique (*CAN*) en signaux compréhensibles par l'unité de traitement.
- **Unité de traitement** : se charge d'exploiter, de traiter et de conserver les données capturées à l'aide d'un processeur associé à un microcontrôleur. Il s'agit du module principal du capteur du moment qu'il contrôle le bon fonctionnement des autres unités.
- **Unité de communication** : elle permet au capteur de communiquer en sein du réseau à l'aide d'un couple émetteur/récepteur qu'on appelle aussi *transceiver*. Elle utilise essentiellement des technologies sans fil telles que le 802.11 et le 802.15.4.
- **Unité d'alimentation** : il s'agit de la source d'énergie qui alimente toutes les unités du capteur. Elle correspond généralement à une batterie ou une pile dont les ressources limitées en font l'une des contraintes majeures des RCVSF.

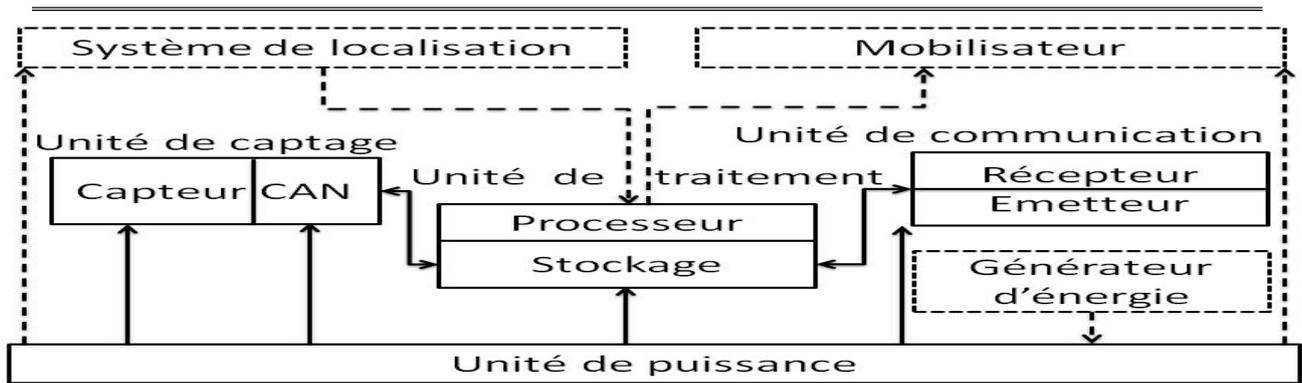


Figure 1.15 – Architecture physique d'un nœud capteur [04]

### 1.9 Dispositifs de capture vidéo :

Plusieurs prototypes de capteurs d'images ont été proposés dans la littérature. Nous pouvons en distinguer deux catégories : d'un côté, ceux qui se sont basés sur des composants commerciaux à faible consommation énergétique. Par exemple, les auteurs dans ont utilisé un capteur d'image *ADCM1670 de Agilent* pour fournir des capacités de vision dans une application de suivi de cibles. D'un autre côté, on trouve les développeurs de dispositifs de caméras conçus spécialement pour les *RCSF*. La plupart se basent sur le principe de modularité des composants technologiques proposés par les fabricants commerciaux de capteurs.

Quelques exemples de prototypes proposés dans les deux catégories sont présentés dans les deux sous-sections suivantes. [04]

#### 1.9.1 Caméras basées sur des composants commerciaux

Voici une liste non exhaustive parmi les dispositifs de capture d'images qui ont été développés en utilisant des composants électroniques commerciaux.

- *MeshEye*

Conçue pour la surveillance distribuée, l'architecture *MeshEye* (voir figure 1.16) intègre un microcontrôleur *Atmel AT91SAM7S*, une mémoire flash *MMC/SD*, deux capteurs *ADNS-3060* à basse résolution comme ceux qui ont été utilisés à l'origine pour des souris optiques, une caméra *CMOS ADCM-2700* et un module radio *CC2420* de *Texas Instruments* qui respecte la norme *802.15.4*.



Figure 1.16: Prototype de l'architecture *MeshEye*

- ***Cyclops***

Les caméras *Cyclops* ont ouvert un grand espace de recherche dans le domaine des réseaux de capteurs d'image. La caméra *Cyclops* a été développée par les laboratoires *Agilent* et le *CENS* (*Center for Embedded Network Sensing*) de l'*UCLA*. Elle permet la capture et le traitement d'images de faible résolution avec une consommation d'énergie relativement faible. Plusieurs versions de *Cyclops* ont été développées : *Cyclops1*, *Cyclops2A*, *Cyclops2B* et *Cyclops2C* (cette dernière version intègre des améliorations significatives par rapport aux versions antérieures). *Cyclops* est composé d'un module *CMOS* de capture d'images de moyenne qualité *ADCM-1700* (*Agilent*), un microcontrôleur *ATmega128L* de *Atmel* avec *128Ko* de mémoire flash pour le stockage du code d'application et *4Ko* de mémoire *RAM* (la même qui est utilisée dans les capteurs *Mica* de chez *Crossbow*), une mémoire *SRAM TC55VCM208A* de *Toshiba* avec *64Ko* et une mémoire flash *AT29BV040A* d'*Atmel* pour le stockage de données. *Cyclops* a aussi un connecteur de *51* pins qui lui permet de se connecter à un capteur *Mica2* ou *Micaz* de *Crossbow* (voir figure I.17). L'inconvénient de *Cyclops* est le temps de traitement qui est relativement long à cause de la faible vitesse du processeur.



Figure 1.17: *Cyclops* connecté à un capteur *Mica2*

- ***CMUcam3***

Il s'agit de la troisième version des systèmes *CMUcam*. Dans une carte caméra *CMUcam3* (voir figure I.18), nous pouvons trouver une caméra *CMOS* Omni vision *OV6620* ou une *OV7620* comme module de capture d'image, un microcontrôleur *NXP LPC2106* (*ARM7TDMI* de *60MHz*) avec *64Ko* de mémoire

RAM et 128Ko de mémoire flash, et un frame buffer de vidéo FIFO Averlogic AL4V8M440 de /Mo. Ce dispositif peut être connecté à un capteur *Telos* de *Berkeley*. Afin d'économiser l'énergie, la caméra peut fonctionner selon trois modes d'opérations : actif, en veille et inactif.

Par rapport à la caméra *Cyclops*, la *CMUcam* consomme beaucoup plus d'énergie, cependant, elle offre un temps de traitement plus rapide.



Figure 1.18: *CMUcam3*

### 1.9.2 Caméras conçues pour les réseaux de capteurs

Concernant les caméras conçues spécifiquement pour les réseaux de capteurs, nous pouvons trouver dans la littérature les propositions suivantes :

*ALOHA*, illustré sur la figure 1.19 est un capteur prototype *CMOS* conçu spécialement pour des applications de réseaux de capteurs. *ALOHA* intègre une représentation de l'information basée sur des événements. L'idée de base est la suivante : des événements sont exécutés quand des pixels individuels atteignent un seuil de tension déterminé. Le pixel exécute une requête au circuit récepteur, en manipulant son adresse sur le bus par activation d'une cellule *ROM* sur l'intersection des lignes et colonne.

Pour le moment, deux versions du capteur d'images *ALOHA* ont été développées:

- ***ALOHAim1*** avec une grille de 32x 32 pixels et un détecteur de contention analogique pour les collisions,
- ***ALOHAim2*** avec une grille de 64x 64 pixels organisés comme 4 quadrants de 32 x 32 pixels indépendants et un détecteur de contention numérique.

*ALOHA* intègre une nouvelle philosophie dans le domaine du développement des réseaux de capteurs, selon laquelle l'information est captée et transmise seulement quand il y a besoin, et tout le traitement des données est exécuté au niveau du capteur matériel lui-même, ce qui permet des économies d'énergie très importantes en laissant de côté la nécessité d'intégrer des composants supplémentaires comme c'est le cas pour les architectures commerciales.



Figure 1.19: ALOHA connecté à un capteur Mica2

## 1.10 Caractéristiques et contraintes d'un RCVSF

Avec des données vidéo, la  $QoS$  devient désormais une exigence fondamentale pour la transmission dans un environnement contraint en ressources. En effet, il existe plusieurs caractéristiques et contraintes dans les RCVSF dont nous citons les plus importantes : [04]

- **Efficacité énergétique** : les capteurs se caractérisent par une limitation de ressources en termes de calcul, de stockage et d'autonomie d'énergie. Cependant, l'énergie demeure la contrainte majeure des RCVSF car les capteurs sont alimentés par des batteries à capacité limitée, la plupart du temps irremplaçables due à la difficulté d'accès de l'environnement de déploiement. Contrairement aux données scalaires, la transmission de flux vidéo provoque un trafic intense et un traitement plus coûteux. Ce phénomène conduit à une consommation énergétique importante pouvant occasionner le dysfonctionnement prématuré du réseau. Par conséquent, l'utilisation de nouveaux algorithmes contribuant à la prolongation de la durée de vie du réseau est indispensable.
- **Auto-organisation** : Dans le cas où les RCVSF sont déployés aléatoirement, où aucune intervention humaine n'est envisageable ; il est nécessaire de maintenir une structure qui répond de manière efficace aux besoins de l'application. Ceci peut être réalisé grâce à l'auto-configuration des capteurs vidéo, sans contrôle centralisé, uniquement avec des interactions locales entre les capteurs.
- **Bande passante** : le streaming de flux vidéo nécessite une bande passante très élevée par rapport à la capacité actuelle des capteurs. En effet, alors qu'un streaming vidéo peut nécessiter des débits supérieurs à l'ordre de  $500\text{ kbps}$ , le débit maximal d'un nœud capteur est de  $250\text{ kbps}$  avec le standard  $802.15.4$ . Des études ont montré que le débit réel utile est de l'ordre de  $70\text{ kbps}$ . Par conséquent des solutions matérielles et logicielles sont nécessaires pour une fourniture suffisante en bande passante afin de supporter la transmission de flux vidéo.
- **Applications multimédias temps réel**. Les données émises doivent être correctement reçues au *Sink* dans les délais exigés par l'application ; sinon elles deviendront obsolètes. Les applications de streaming vidéo temps réel requièrent des garanties strictes de délai bout à bout, de bande passante et de gigue .
- **Topologie dynamique** : La topologie du réseau est en constante évolution à cause du changement

de l'état d'activité des capteurs (actif, en veille et inactif) ou bien suite à la défaillance ou l'épuisement d'énergie des capteurs. Par conséquent, il faudrait que ces derniers soient capables de communiquer et de collaborer entre eux en utilisant des protocoles d'auto-organisation afin de maintenir la topologie du réseau.

- **Scalabilité** : Les applications de surveillance dans des environnements hostiles peuvent nécessiter un déploiement très dense des capteurs vidéo. Lorsque le nombre de capteurs augmente, ceci ne doit en aucun cas influencer sur les performances du réseau. Les différents algorithmes et protocoles proposés pour les *RCVSF* doivent prendre en considération le facteur de passage à l'échelle.
- **Tolérance aux pannes** : La défaillance de certains capteurs vidéo peut être la conséquence de l'épuisement de leur batterie ou encore suite à un endommagement physique du capteur dans l'environnement de surveillance (après un largage à partir d'un avion par exemple). Le *RCVSF* doit pouvoir remédier aux défaillances d'une partie de ses capteurs et continuer à fonctionner même avec des performances réduites. La tolérance aux pannes est la capacité de soutenir les fonctionnalités d'un réseau de capteurs sans causer d'interruption lorsqu'un capteur cesse de fonctionner. La tolérance aux pannes dépend aussi de l'environnement de déploiement du *RCVSF*, dans le cas où le réseau est déployé dans un habitat, la tolérance aux pannes exigée peut être basse car le degré d'endommagement des capteurs est réduit. Cependant, si le *RCVSF* est déployé dans un environnement hostile de surveillance, la tolérance aux pannes doit être plus importante à cause de la criticité de l'application.
- **Sécurité** : Concernant les applications de surveillance qui nécessitent un niveau de sécurité plus élevé telles que les applications militaires, les données collectées par les capteurs vidéo doivent être sécurisées. Des mécanismes d'authentification, de confidentialité, et d'intégrité peuvent être utilisés dans ce type d'applications. Cependant, les algorithmes de cryptographie conçus pour les réseaux de capteurs doivent tenir compte des ressources restreintes des capteurs vidéo.

**Couverture** : La couverture peut être considérée comme l'une des métriques de la *QoS* d'un réseau de capteurs. Une mauvaise répartition des capteurs vidéo peut rendre certains nœuds inutilisables s'ils ne sont plus en mesure de communiquer avec les autres capteurs pour atteindre à la fin la *SB*. La stratégie la plus adoptée dans la problématique de couverture est l'exploitation de la redondance issue du déploiement aléatoire des capteurs vidéo et de procéder à leur mise en veille en alternance.

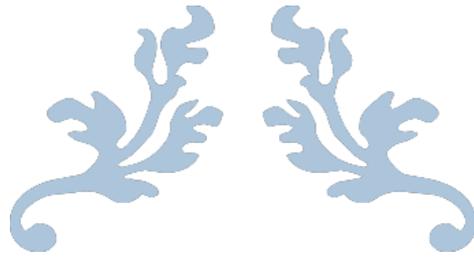
**Qualité des images capturées** : Dans un contexte d'applications critiques, la capacité à exploiter les informations visuelles d'une image à la réception est capital. Pour ce faire, la vitesse de capture associée à chaque capteur doit être satisfaisante. De plus, afin de réduire la quantité de données visuelles collectées nous devons faire appel à la technique de compression d'images. Cette dernière peut s'effectuer de plusieurs manières : avec ou sans perte de données, centralisée ou distribuée. La

compression avec perte de données demeure la plus pertinente pour des transmissions à bas débit et pour la réduction des coûts en termes de mémoire et de transmission. Malgré que la compression participe à l'amélioration des performances du réseau, un taux de compression maximal peut dégrader fortement la qualité de l'image restituée.

- **Routage multi-chemins :** Le transfert de données vidéo doit respecter certaines métriques de *QoS*, par exemple, les applications de surveillance de données critiques sont sensibles au temps de réponse (délai) et exigent une certaine fiabilité dans la livraison des données (taux d'erreur) . Afin de pallier ces problèmes, de nouvelles formes de routage sont utilisées, comme le routage multi-chemins pouvant garantir une communication de qualité entre les différents nœuds dans le réseau. Un protocole de routage multi-chemins permet de construire et de maintenir plusieurs chemins entre la source et la destination. Son objectif est d'augmenter la robustesse et la disponibilité des chemins.

### **1.11 Conclusion**

Les réseaux de capteurs sans fil constituent une technologie récente et innovante qui suscite un intérêt considérable étant donné son utilité dans différents domaines. Cependant, il reste encore de nombreux problèmes à résoudre dans ce domaine afin de pouvoir les utiliser dans des conditions optimales. L'un des problèmes qu'on peut rencontrer dans ce genre de réseau est la consommation d'énergie au niveau des capteurs lors de la transmission de leurs données à la station de base qui peut être assez éloignée de ces derniers. De ce fait plusieurs travaux se sont intéressés, en premier lieu, à la problématique de l'économie d'énergie au niveau des capteurs et l'amélioration de la durée de vie du réseau .



## **CHAPITRE 2**

**La qualité de service**

**&**

**les réseaux de capteurs**

**sans fils (QoSRCSF)**



## **2.1. Introduction :**

Les réseaux de capteurs sans fil présentent un intérêt considérable et une nouvelle étape dans l'évolution des technologies de l'information et de la communication. Cette nouvelle technologie suscite un intérêt croissant étant donnée la diversité de ces applications : santé, environnement, industrie et même dans le domaine sportif. Contrairement aux réseaux traditionnels qui se préoccupent de garantir une bonne qualité de service, les réseaux de capteurs doivent, en plus, prendre en compte la conservation d'énergie.

Dans ce chapitre nous allons détailler les concepts liés aux réseaux de capteurs sans fil à savoir la qualité de service et la conservation d'énergie dans ces réseaux.

## **2.2 Définition de la QoS:**

Plusieurs définitions ont été proposées pour le terme de la qualité de service (QoS) ou qualité of service (QoS) dont les plus importantes sont :

- La **Qualité de Service** (QoS) est la capacité à véhiculer dans de bonnes conditions un type de trafic donné, en termes de disponibilité, débit, délais de transmission, gigue, taux de perte de paquets...
- La **Qualité de Service** est une notion subjective. Selon le type d'un service envisagé, elle pourra résider dans le débit (Un débit permet de mesurer le flux d'une quantité relative à une unité de temps au travers d'une surface quelconque.), le délai (pour les applications interactives ou la téléphonie), la disponibilité (accès à un service partagé) ou encore le taux de pertes de paquets (pertes sans influence de la voix ou de la vidéo (La vidéo regroupe l'ensemble des techniques, technologie, permettant l'enregistrement ainsi que la restitution d'images animées...)). [08]
- La **Qualité de Service** regroupe un ensemble de technologies mises en œuvre pour assurer des débits suffisants et constants sur les réseaux, y compris Internet. [09] Appliquée aux réseaux à commutation de paquets (réseaux basés sur l'utilisation de routeurs) la QoS désigne l'aptitude à pouvoir garantir un niveau acceptable de perte des paquets, défini contractuellement, pour un usage donné (voix sur IP, vidéoconférence, etc.) [12].

## **2.3 But de la QoS :**

Le but de la QoS est donc d'optimiser les ressources du réseau (Un réseau informatique est un ensemble d'équipements reliés entre eux pour échanger des informations. Par analogie avec un filet (un réseau est un « petit rets », c'est-à-dire un petit filet), on appelle nœud (node) l'extrémité d'une connexion, qui peut être une intersection de plusieurs connexions (un ordinateur, un routeur, un concentrateur, un commutateur) et de garantir de bonnes performances aux applications critiques. La Qualité de Service sur les réseaux permet d'offrir aux utilisateurs des débits et des temps (Le temps est un concept développé pour représenter la variation du monde : l'univers n'est jamais figé, les éléments qui le composent bougent, se transforment et évoluent pour l'observateur qui est l'homme. Si on considère l'univers...) de réponse différenciés par application suivant les protocoles mis en œuvre au niveau de la couche réseau.

Elle permet ainsi aux fournisseurs de services (départements réseaux des entreprises, opérateurs...) de s'engager formellement auprès de leurs clients sur les caractéristiques de transport (Le transport, du latin trans, au-delà, et portale, porter, est le fait de porter quelque chose .[08]

Selon le type d'un service envisagé, la qualité pourra résider :[11]

- Le débit (téléchargement ou diffusion vidéo).
- Le délai (pour les applications ou la téléphonie).
- La disponibilité (accès à un service partagé).
- Le taux de pertes de paquets.

## **2.4 Les paramètres de qualité de service:**

### **2.4.1 La bande passante (ou débit binaire):**

C'est la quantité de données pouvant être transmise durant un laps de temps. Le plus souvent, elle est mesurée en nombre de bits/seconde (bits par seconde,- bps) ou un de ses ordres de grandeur: kilobits par seconde ou Kbps, mégabits par seconde ou Mbps, gigabits par seconde ou Gbps, etc... La bande passante est un paramètre clé de la vitesse de transmission des données: plus la bande passante est élevée, plus la transmission d'un

fichier sera rapide. Ce paramètre est également crucial pour les applications temps réel. Par exemple, la vidéo en direct nécessite une bande passante plancher sous laquelle on ne doit pas descendre pour garantir une transmission en temps réel des images. La contrainte de bande passante est donc une valeur seuil de bande passante disponible sous laquelle il ne faut pas descendre [13].

### **2.4.2 Le délai (ou latence) :**

C'est le temps qui sépare l'émission d'un paquet de données de l'arrivée à sa destination. Il est mesuré en secondes ou un de ses ordres de grandeur (de quelques millisecondes pour les transmissions nécessitant une très grande réactivité à plusieurs minutes pour les applications tolérantes aux délais). Une contrainte de délai définit un temps maximum à ne pas dépasser pour tous les paquets. Les applications temps réel et interactives sont très sensibles au délai. Par exemple un jeu vidéo en ligne nécessite un délai très court (non perceptible par les joueurs) pour garantir la fluidité de l'interaction entre les joueurs et avec le jeu [13].

### **2.4.3 La perte de données :**

Le taux de perte est la proportion des paquets qui ne parviennent pas à leur destination. Ces pertes dépendent de :

***La fiabilité du support des liaisons*** : un support peu fiable entraîne une corruption fréquente des paquets. Si un paquet s'avère corrompu à l'issue de sa vérification grâce aux bits de contrôle (checksum), il sera détruit. On peut toutefois estimer qu'actuellement la fiabilité des liens dans le réseau Internet est relativement élevée, et que par conséquent, les pertes de paquets dues à des défauts de support sont très faibles.

***L'occurrence de surcharges locales (congestions) dans le réseau*** : les paquets devant être placés dans une file d'attente pleine sont détruits. Lorsque le protocole de transport garantit l'arrivée de l'information transmise, les paquets perdus doivent être renvoyés. Les pertes ont donc une influence directe sur l'augmentation du délai et sur la gigue. [15]

On distingue donc les services qui peuvent adapter leur fonctionnement vis-à-vis de la congestion du réseau des services qui ont besoin de garanties sur un ou plusieurs critères de QoS pour fonctionner. Premièrement les services "élastiques". qui s'agit typiquement des services IP classique comme le transfert de fichier etc. Certes, la satisfaction de l'utilisateur de ces services est meilleure lorsque la bande passante est la plus élevée possible, lorsque les délais sont faibles

et qu'il n'y a pas de pertes. Mais, ces services n'ont pas d'exigences fortes en terme de QoS, ils pourront toujours s'adapter à une dégradation du réseau. À l'inverse, les services "multimédias" ont besoin d'une qualité de service minimale pour fonctionner, ils ne peuvent pas (ou peu) s'adapter à l'état du réseau. Si le débit fourni est inférieur au seuil du débit, leur qualité se dégrade fortement et le service devient inutilisable. Ces services sont aussi sensibles aux autres critères de QoS. Les services "interactifs" sont particulièrement exigeants en termes de délais : il faut que le service réagisse au plus vite à l'action de l'utilisateur [14].

## 2.5 Les modèles de QoS :

### 2.5.1 IntServ (Integrated Service) :

IntServ est une architecture de QoS qui permet la réservation de ressources sur le réseau pour un flux spécifique. Ce dernier correspond à une séquence de paquets possédant les mêmes adresses IP sources et destinations et les mêmes besoins de QoS. Pour décrire les besoins d'un flux en QoS, IntServ se base sur la description du trafic TSpec qui inclut plusieurs informations : p (débit crête, octets/seconde), b (taille maximale du burst, octets), r (débit moyen, octets/seconde), m (unité minimum contrôlée, octets), M (taille maximum d'un paquet, octets). L'architecture IntServ définit deux types de services principaux, en plus du service Best-Effort :

- ***Le service de charge contrôlée (CLS: Controlled Load Service)*** : Ce service fournit approximativement la même QoS pour un flux indépendamment de l'état du réseau : état normal ou état de surcharge. La différence entre ce service et le service Best-effort est décelable uniquement quand le réseau est surchargé.

- ***Le service garanti (GS : Guaranteed Service)*** : Ce service assure une bande passante stricte, un délai de transmission de bout-en-bout borné et une perte de paquet nulle pour un flux [14]. L'idée de base du modèle IntServ est de fournir une QoS individualisée à chaque connexion en utilisant un mécanisme de contrôle d'admission et de réservation de ressources via le protocole RSVP. Avec ce protocole, chaque routeur est interrogé sur ses ressources disponibles. S'il est possible de trouver un chemin entre les deux extrémités de connexion sur lequel tous les routeurs disposent de suffisamment de ressources pour assurer la QoS de la connexion, alors celle-ci est établie, sinon elle est refusée [31].

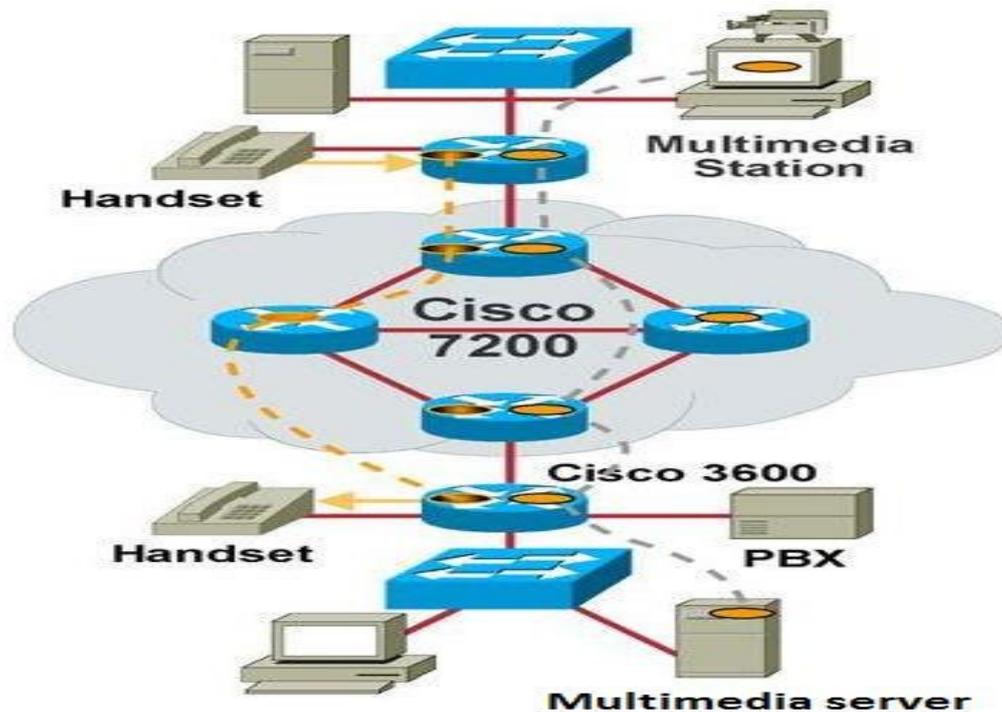
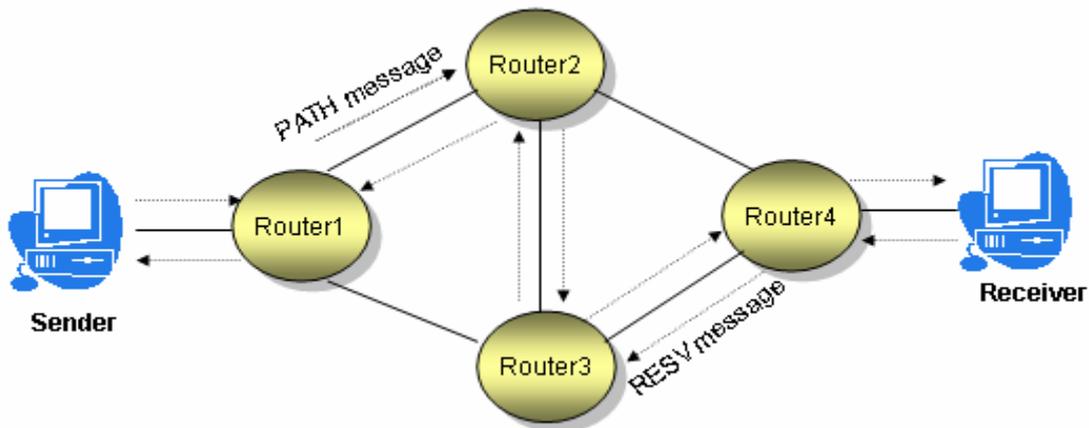


Figure 2.1 Architecture du modèle IntServ [31]

### 2.5.1.1 Fonctionnement du modèle IntServ

L'idée fondamentale de l'architecture IntServ est de réserver des ressources telles que bande passante et tampons. Pour recevoir une assurance de performance du réseau, l'application doit configurer la réservation de ressources le long de son chemin avant de pouvoir commencer à transmettre des paquets. Il est basé sur l'utilisation du protocole RSVP (Resource Reservation Protocol) qui a été inventé comme protocole de signalisation pour les applications de réserve Ressources. L'application lance une session à la demande avec le réseau à l'aide de RSVP. [28]

Cette session identifie les exigences de service de l'application, y compris la bande passante et délai. RSVP est un protocole de réservation orienté récepteur pour la configuration réservation de ressources sur Internet, c'est-à-dire que les destinataires sont responsables de quelles ressources seront réservées. Les accusés de réception des destinataires voyagent des destinataires vers l'expéditeur et construisent progressivement un chemin [29].



**Figure 2.2 Opérations de configuration de réservation de ressources de base du protocole de signalisation RSVP [29]**

L'expéditeur envoie un message PATH au destinataire spécifiant les caractéristiques du trafic. Chaque routeur intermédiaire le long du chemin transmet le message PATH au saut suivant déterminé par le protocole de routage. À la réception d'un message PATH, le destinataire répond par un message RESV pour demander des ressources pour le flux. Le routeur intermédiaire le long du chemin peut rejeter ou accepter la demande du message RESV. Si la demande est rejetée, le routeur enverra un message d'erreur au destinataire et le processus de signalisation se terminera. Si la demande est acceptée, liaison bande passante et tampon espace est alloué pour le flux et les informations d'état de flux associées sont installées dans le routeur [28].

Malgré sa robustesse, l'architecture IntServ n'a pas rencontré le succès escompté pour assurer la QoS au niveau IP. Le premier inconvénient de cette architecture est sa sociabilité. En effet, la nécessité de maintenir l'état des ressources réseau pour chaque flux limite sérieusement le déploiement à grande échelle d'IntServ. Les autres reproches concernent sa complexité de mise en œuvre et l'overhead introduit sur le réseau en utilisant un protocole de réservation de ressources [28].

### **2.5.2 Modèle Diffserv :**

Ce modèle définit une architecture de réseau qui permet de différencier les services des données, il a été mis en place afin de pallier aux difficultés de déploiement du modèle définissant l'architecture IntServ. Le mécanisme qui en découle classe et contrôle le trafic, le trafic est

séparé en classe de trafic et ces derniers peuvent être identifiés grâce à une valeur codée dans l'en-tête IP. Ces tâches sont réalisées en périphérie du réseau et les routeurs du cœur de ce réseau traitent les paquets en fonction de la classe qui les définit. Une combinaison de ce modèle avec le protocole MPLS fournit un niveau de QoS optimal. [32].

### **2.5.2.1 Les composants du modèle DiffServ :**

Dans un domaine DiffServ, il faut distinguer les routeurs d'accès, ou routeurs de bordure (edgerouters), et les routeurs du cœur de réseau (corerouters).

**Les routeurs de bordure** sont les portes d'entrées obligatoires pour un flot pénétrant dans le domaine DiffServ. Ils effectuent des traitements (figure 2.3) sur les paquets entrant qui ont pour effet :

- de déterminer leur classe de service (classifier) grâce à l'en-tête du paquet (champ Type de service (ToS), adresse source et/ou destination, protocole, port) sur la base d'un contrat avec l'opérateur, et affecter l'information d'élimination possible ou non, selon les mesures qui sont faites (meter) sur les débits moyens et crête, et en accord avec le contrat passé avec l'opérateur. En cas de congestion dans le cœur de réseau, un paquet éliminable (out profile) sera traité dans une classe moins prioritaire, voire détruit. [31].

- de réguler leur débit selon leur classe, souvent par un token bucket (shaper),
- de les détruire ou de les déclasser (dropper, policer) en cas de congestion (s'ils sont marqués out profile). L'opération de déclassement n'est possible que dans les routeurs de bordure [31].

**Les routeurs de cœur de réseau :** Les routeurs du cœur de réseau réalisent des opérations simples de bufferization et de routage des paquets en se basant uniquement sur le marquage effectué par les routeurs situés en bordure de domaine DiffServ. La différenciation de service se fait au niveau des deux mécanismes cruciaux du modèle DiffServ: le Planification (C'est l'un des composants les plus critiques dans l'architecture d'un réseau à QoS.) et le buffer management (Le mécanisme de gestion de tampon consiste à éliminer des paquets en cas de congestion du tampon d'un port de sortie d'un routeur.). Chaque sortie du routeur possède un nombre fixe de files logiques où le routeur dépose les paquets arrivant selon leur classe de service. Les files sont servies en accord avec l'algorithme d'ordonnancement [31].

Dans un modèle d'architecture DiffServ, les tâches de QoS, qui sont le contrôle et la classification du trafic sont réalisées en périphérie du réseau sur les routeurs Edge [30]. Une fois

ces tâches réalisées, les routeurs "Core" n'ont plus qu'à traiter ces flux en fonction de leur classification. Voici une figure représentant ces mécanismes:[32]

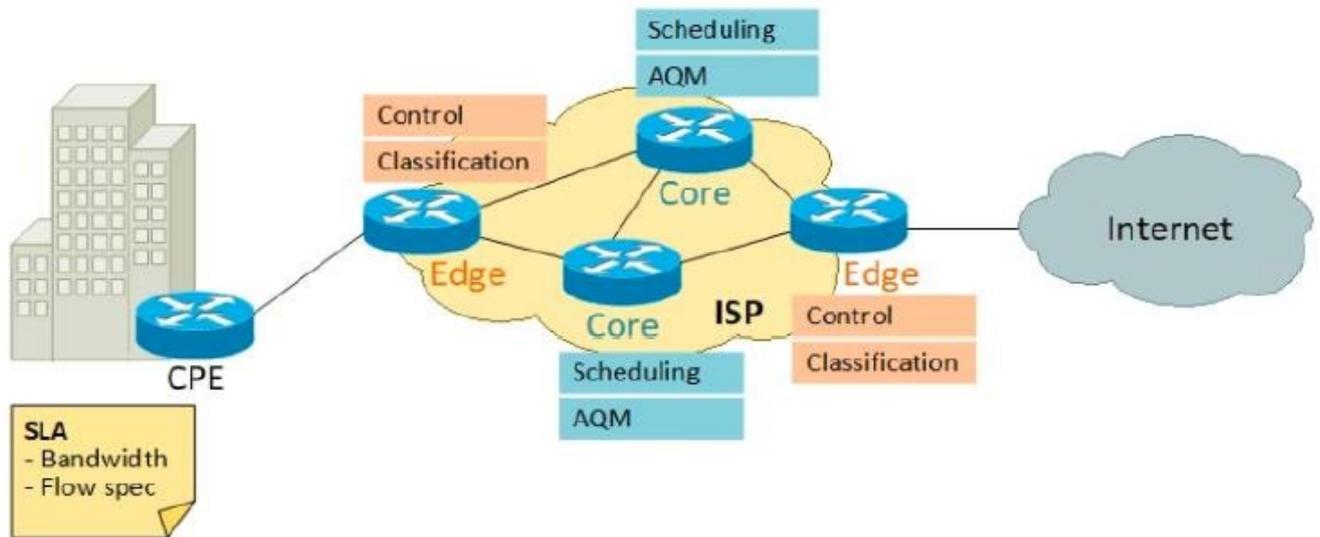


Figure 2.3 Tâches réalisées dans un modèle d'architecture DiffServ [32]

## 2.6 Les classes de services

Une classification des principales applications est présentée comme suit :

- **Voix** : Regroupe toutes les applications du type conversationnel (Voix, Visio Conférence, ...) ayant pour contrainte forte des objectifs sur le délai et la gigue. Elles sont également sensibles au taux de perte bien qu'il ne soit pas possible de retransmettre les données et requièrent des débits assez faibles [27].
- **Vidéo** : Regroupe toutes les applications multimédia diffusées ou non (Vidéo à la Demande – VoD, la télévision sur IP – IP TV, ...) ayant pour contrainte forte le taux de perte et le débit et dans une moindre mesure le **délai** et la gigue [27].
- **Donnée** : Regroupe toutes les applications de transfert de données ayant pour seule contrainte un taux de perte nul et qui s'accommodent d'un délai et d'une gigue quelconque. Un débit garanti caractérise cette classe sans toutefois en faire une contrainte stricte [27].

## 2.7 Qualité de service dans les réseaux de capteurs :

Depuis l'émergence des réseaux de capteurs, les principaux travaux de recherche se sont concentrés sur la gestion des données issues des capteurs, la gestion de l'énergie et le

routage, ce qui est important pour la bonne exploitation de ces réseaux [23]. La qualité de service (QoS) n'a pas fait partie des axes d'intérêt des chercheurs dès l'apparition de ces derniers mais le devient de nos jours.

Selon les applications utilisées, les réseaux de capteurs doivent assurer un niveau de qualité de service minimum afin de répondre aux besoins des utilisateurs finaux. Par exemple, dans un système de détection d'incendie, la transmission de l'information doit être faite en temps réel dans le but de mener les actions nécessaires pour contrôler la situation. Le délai de transmission devient moins critique dans le cas d'un système de climatisation qui a la tâche de maintenir une certaine température dans un local.

La notion qualité de service est un ensemble de mécanismes permettant de garantir les bonnes caractéristiques du réseau pour les utilisateurs. Le réseau est composé de plusieurs éléments notamment les équipements actifs et le support de transmission chargé de véhiculer les données. Ces éléments sont appelés ressources du réseau et sont généralement limitées et finies, il faudra donc optimiser l'utilisation de ces ressources pour satisfaire les besoins des utilisateurs et des applications.

La qualité de service a été bien étudiée dans le cadre des réseaux filaires. Dans ce type de réseaux, ce sont généralement les routeurs qui assurent la QoS en utilisant un mécanisme de priorisation des flux de données. Selon la priorité du paquet reçu, le routeur peut soit l'acheminer soit l'ignorer, cela dépendra des ressources disponibles au moment de la réception.

Dans les réseaux de capteurs, le contexte est complètement différent, les ressources de ces derniers sont limitées notamment au niveau de la bande passante, de la capacité de calcul, etc. De plus, si l'on considère que les nœuds peuvent être mobiles, ceci rend plus difficile la transmission des données et par conséquent la qualité de service devient difficile à garantir.

## **2.8 Garantie de la qualité de service dans les RCSF :**

La garantie de la qualité de service peut se faire selon plusieurs approches et les mécanismes utilisés peuvent exister dans tous les niveaux du modèle OSI.

### **2.8.1. Solutions MAC :**

Les protocoles MAC « Medium Access Control » sont responsables de la gestion de l'accès au canal sans fil partagé. Cette gestion se fait selon plusieurs méthodes en fonction des besoins des applications, à savoir ; les méthodes avec contention, les méthodes sans contention et les méthodes d'accès hybrides.[34]

### **2.8.2. Routage :**

Même si la couche MAC arrive à garantir la qualité de service localement (un saut), la qualité de service de bout en bout ne peut pas être assurée dans un environnement multi-sauts s'il n'y a pas de garantie au niveau de la couche réseau. Les protocoles de routage participent à la garantie de la qualité de service par le choix du chemin optimal en vue de transmettre les paquets tout en respectant les besoins de l'application à travers plusieurs techniques de routage. L'objectif du protocole de routage dans ce cas est de trouver le chemin le plus court, mais qui offre une garantie des besoins requis par l'application cible. Les protocoles de routage existant offrent aussi une garantie de QoS soit probabiliste ou déterministe. On note qu'il existe des protocoles MAC conçus pour la communication multi saut, notamment la topologie en arbre de clusters (cluster tree), comme les protocoles DMAC et DB-MAC . Ce type de protocoles MAC peut garantir la QoS de bout en bout à ce niveau du modèle OSI (Open System Interconnexion) de l'ISO (International Organisation for Standardisation). [34]

### **2.8.3. Traitement des données :**

L'agrégation des données permet d'améliorer l'efficacité en énergie, de réduire la charge du trafic global, notamment dans le « Backbone » du réseau et les points les plus proches de la destination, et par conséquent, d'améliorer la qualité de service.

Il existe plusieurs techniques utilisées pour l'agrégation des données comme les approches basées sur les clusters, les approches basées sur l'arbre d'agrégation, l'agrégation basée sur le traitement distribué des données à l'intérieur du réseau « In-Network » et l'agrégation centralisée qui se fait dans un serveur à l'extérieur du réseau « out-of-network ». [34]

### **2.8.4. Cross-layer :**

Dans les modèles de couches (OSI et TCP/IP par exemple), le rôle de chaque couche est d'offrir des services à la couche supérieure adjacente. L'intérêt de ce modèle de communication est de garder une transparence pour éviter à ce que les couches supérieures connaissent le détail des autres couches inférieures. Toutes les communications doivent se faire à travers des interfaces prédéfinies par le protocole

utilisé dans chaque couche. Ceci a donné comme résultats, une facilité de standardisation des protocoles de communication, une aisance dans le développement de nouveaux protocoles de toutes les couches, en plus de l'interopérabilité entre différents réseaux et équipements. Malgré tous ces avantages, ce modèle n'est pas très flexible, car les communications s'effectuent de manière stricte et prédéfinie. Cela peut mener dans certains cas à une utilisation moins efficace des ressources du réseau, et par conséquent, à une dégradation de la qualité de service, notamment dans les environnements dynamiques. Ceci a motivé des chercheurs à adopter le modèle de cross-layer. Le concept du cross-layer est centré sur l'idée d'optimiser le contrôle et l'échange des informations entre deux ou plusieurs couches de la pile protocolaire utilisée à travers un agent ou un plan de gestion partagé par les différentes couches. L'objectif est d'améliorer les performances en exploitant cette interaction entre les protocoles des différents niveaux du modèle OSI. [34]

## **2.9 Qualité de service pour les RCMSF :**

La gestion de la QoS pour les RCMSF reste encore un défi pour les chercheurs. Selon Almalkawi et al. (2010), il y a plusieurs problèmes à résoudre pour ce type de réseaux qui ne permettent leur vaste utilisation. Les problèmes que l'on doit tenir compte se trouvent dans les couches d'application, de transport, et de réseau (routage). [33]

### **2.9.1 La couche application :**

Pour les auteurs de (Akyildiz et al., 2007; Gurses et Akan, 2005), le contrôle d'admission offert par la couche application est un des problèmes à résoudre pour les RCMSF. Le contrôle d'admission, c'est-à-dire, le fait d'éviter aux applications d'établir des flots de données lorsque les ressources nécessitées du réseau ne sont pas disponibles, doit être basé sur les exigences de qualité de service de l'application sous-jacente. Selon Akyildiz et al. (2007), les RCMSF fourniront des services différenciés pour les différents types de paquets qui y circuleront. En particulier, ils devront fournir le service différencié entre les paquets de temps-réel et les paquets qui tolèrent des délais, ou entre les applications qui admettent les pertes de données et celles qui ne les admettent pas. En plus, il y a des applications qui demandent un flot continu de données multimédia par une période prolongée de temps (multimédia streaming), tandis qu'il y a d'autres applications qui peuvent demander des observations obtenues dans une période de temps plus courte (snap shot multimedia content). Perillo et Heinzelman (2003) présentent un algorithme de contrôle d'admission de la couche d'applications dont l'objectif est de maximiser le temps de vie du réseau soumis à des demandes de l'application telles que la bande passante et la fiabilité. Dans (Boulis et Srivastava, 2004), une méthode de contrôle d'admission pour des applications est proposée. Les auteurs proposent

des manières pour mesurer en temps réel la consommation d'énergie qu'une application utilise dans un nœud capteur. A partir de ces mesures, les auteurs présentent une politique de contrôle d'admission optimale qui tient compte de l'énergie ajoutée aux nœuds individuels par la nouvelle application. Bien que ces approches considèrent des éléments de qualité de services dans la couche d'applications, elles ne considèrent pas les conditions multiples de qualité de service (comme le délai, la fiabilité ou la consommation de l'énergie) appliquées simultanément, tel qu'il est exigé par les RCMSF. Par conséquent, il y a clairement une nécessité à établir des mécanismes et des critères pour gérer l'admission des flots multimédia conformément aux conditions de qualité de service souhaitées par la couche d'application. [33]

### **2.9.2 La couche réseau :**

La couche réseau d'un RCMSF est la responsable du transport de données entre un nœud et le SINK, et constitue un élément important pour fournir de la qualité de service grâce aux raisons suivantes :

- a) Cette couche est responsable de l'obtention de routes efficaces qui tiennent compte de la consommation de l'énergie, qui sont stables et qui satisferont des paramètres de qualité de service demandés par l'application.
- b) Cette couche sert d'intermédiaire entre la couche MAC et la couche application pour l'échange de paramètres de performance entre elles.

En raison des exigences intensives de ressources qui ont les applications multimédia et de la basse disponibilité de telles ressources dans un RCMSF, le travail du protocole de routage est très compliqué. Aussi, tel que nous l'avons mentionné au-dessus, la couche réseau sert comme un intermédiaire entre l'application et la couche MAC de chaque nœud du réseau. En plus, la couche réseau a la connaissance des diverses caractéristiques des routes trouvées entre chaque nœud du réseau et le SINK. La couche MAC ne connaît que les caractéristiques de point à point entre les différents liens du réseau. Finalement, la couche application n'a pas d'information des conditions du réseau et n'a que de l'information de l'application. Cela est la raison pour laquelle, pour répondre à des exigences de QoS de la couche application, il faut que chacune des trois couches collaborent parmi elles. La couche réseau est l'unique que peut faire correspondre les paramètres de QoS que la couche application demande aux paramètres de performance de la couche MAC. De manière semblable, grâce à la rétroaction de la couche MAC à la couche application, cette dernière pourra réaliser des ajustements à ses propres paramètres. La qualité de service est une exigence typique pour les protocoles de la couche réseau qui transportent de l'information multimédia. Cependant, le support de la QoS dans les RCMSF est une tâche difficile. La raison est simple

: les restrictions d'énergie, la puissance de calcul limitée et la basse capacité de la mémoire des nœuds capteurs. Malgré cela, il y a aussi plusieurs recherches dans la littérature de protocoles pour la couche réseau qui fournissent de la QoS et qui permettent le transport d'information multimédia. Ces protocoles, on va les diviser en trois catégories : basés sur IntServ, basés sur DiffServ et routage à multiples routes (Bhuyan et al., 2010). [33]

### **2.9.2.1 Protocoles basés sur IntServ :**

Ces protocoles utilisent des réservations des ressources par flot de données. Dans (He et al., 2003), les auteurs ont proposé un protocole routage dénommé SPEED qui supporte les communications en temps réel dans un réseau de capteurs et fournit des garanties de temps réel de façon soft. Le protocole SPEED utilise un schéma d'acheminement géographique non-déterministe et sans sauvegarder l'état (Stateless Non-déterministe Geographic Forwarding) comme le mécanisme primaire de routage. L'avantage du routage géographique est qu'il n'existe pas la nécessité d'établir de routes entre l'origine et le destinataire des messages. Pour SPEED, les nœuds du réseau doivent supporter une vitesse maximale de transmission pour chaque paquet admis. Le terme "vitesse de transmission" est défini par les auteurs comme le taux auquel le paquet avance le long de la ligne qui va de la source à la destination. De cette définition de vitesse on peut déduire que le délai de bout-en-bout dans le réseau est proportionnel à la distance entre la source et la destination. Si une application veut une vitesse plus grande que la vitesse maximale de transmission, elle ne serait pas admise au réseau. L'algorithme de routage calcule le délai de transmission pour un paquet en utilisant la distance de entre le nœud actuel et la destination du paquet, et la vitesse maximale de transmission. Ce schéma est similaire en esprit au modèle d'IntServ car la connexion n'est admise que dans le cas que le réseau puisse garantir de la supporter. Dans les circonstances où quelques liens de la route deviennent congestionnés et ils ne sont plus capables de supporter la vitesse maximale de transmission, le protocole a des mécanismes pour détourner le trafic vers d'autres routes. SPEED utilise la technique nommée back-pressure re-routing pour surmonter la dégradation dans la transmission des paquets due à la congestion du réseau. Cette technique évite que les paquets traversent des liens congestionnés, et de cette manière, la vitesse de transmission de paquets est maintenue. Un des inconvénients de SPEED est que le protocole n'a pas de schéma pour établir les priorités des paquets. De plus, chaque nœud ne peut que transmettre à une vitesse inférieure ou égale à la vitesse maximale pour laquelle le protocole a été configuré. Néanmoins, si le paquet a besoin d'une transmission à plus haute vitesse (par l'exemple, pour se récupérer des congestions dans les nœuds précédents), cela n'est pas possible, même si le réseau peut supporter telle vitesse. [33]

### 2.9.2.2 Protocoles basés sur DiffServ :

L'approche DiffServ est très populaire dans les réseaux de capteurs sans fils étant donné son extensibilité. Dans cette approche, les paquets qui vont être transmis sont classifiés par des niveaux de priorité différents. Chacun de ces niveaux présente différentes garanties de temps, de bande passante, de gigue, etc. Chaque priorité représente une classe de trafic, et aussi, chaque paquet appartient à une de ces classes de service en fonction de ses besoins. Le protocole SAR présenté par Sohrabi et al. (2000), appartient à cette catégorie. Il utilise le schéma de priorités constantes pour chaque paquet. Pour ce protocole, chaque paquet qui appartient à un flot donné, a une valeur constante de la priorité et cette valeur reste fixe tout le temps que le paquet traverse la route vers sa destination. SAR utilise une approche multi route basée sur des tables de routage pour découvrir de différentes routes qui répondent aux exigences de QoS et de conservation de l'énergie dans le réseau de capteurs. Le nœud source sélectionne une route particulière parmi toutes les routes découvertes pour l'utiliser dans la transmission d'un flot donné. Cette sélection est faite en tenant compte des exigences de délai du flot et les intentions d'équilibrage de charge de la source. Les nœuds intermédiaires de la route choisie prennent en compte la priorité du paquet au moment de le transmettre. L'avantage de cette approche est sa capacité pour supporter des classes diverses de trafic pour les paquets. Toutefois, l'utilisation des tables dans la mémoire des capteurs pour réaliser le routage est son majeur inconvénient. En effet, cette table requiert une quantité significative de mémoire dans chaque nœud capteur et, évidemment, cette méthode n'est pas extensible aux réseaux assez grands. De même, le fait qu'un paquet ne peut jamais changer de priorité empêche que les nœuds réagissent à des changements inattendus dans le réseau.

Akkaya et Younis (2005a) proposent un protocole basé sur la QoS pour le trafic généré par un réseau de capteurs sans fils consistant en des capteurs d'images. Ce protocole utilise aussi un schéma de priorités constantes où tous les paquets qui vont être transmis en temps réel ont la même priorité. Le protocole travaille avec le concept de « le coût d'un lien ». Tel coût est défini à partir de l'énergie résiduelle de chaque nœud, l'énergie consommée pendant la transmission, le taux d'erreurs et d'autres paramètres de la communication. Tout le trafic dans le système est divisé en deux classes : best-effort et temps réel. Dans chaque nœud, une file d'attente est utilisée pour stocker les paquets de chaque classe. Le protocole trouve plusieurs routes de la source à la destination en utilisant une version étendue de l'algorithme de Dijkstra. Ensuite, la source sélectionne une route qui satisfait les exigences du délai de bout-en-bout du paquet et après envoi le susdit paquet au prochain nœud de la route. Chaque nœud intermédiaire classe le paquet reçu dans les catégories de temps-réel ou de best-effort. L'algorithme de répartition associé au protocole ne permet jamais le blocage des paquets best-effort. Le mérite de cet algorithme réside dans le fait qu'il garantit la transmission des paquets best-effort tout en maximisant le débit du trafic en temps réel. Le

principal inconvénient de cette approche réside dans le manque de support pour multiples priorités du trafic en temps-réel. Dans une application multimédia, des paquets différents pourraient avoir des exigences de QoS différentes, et pour ces motifs, cette approche ne satisfait pas ce besoin. D'autre part, l'algorithme pour calculer des routes multiples a besoin d'une connaissance complète de la topologie du réseau dans chaque nœud, et pour cette raison, on peut armer que cette approche n'est pas évolutive (scalable).

Felemban et al. (2006) présentent un mécanisme pour la transmission de paquets nommé Multi-path Multi-speed Routing Protocol (MMSPEED). Ce protocole utilise des catégories différentes pour les paquets et ces catégories peuvent changer dans chaque nœud du réseau. MMSPEED fournit un schéma de garantie de la QoS en deux domaines notamment : la gestion du temps et la fiabilité. Par rapport à la fiabilité, elle est obtenue grâce à un routage multi-chemin avec un nombre de chemins dépendant du degré de fiabilité requis pour un paquet. Quant au temps, la transmission des paquets dans un délai spécifique est obtenu grâce à la capacité du réseau d'offrir différentes vitesses pour les paquets (un mécanisme paraît à celui du protocole SPEED (He et al., 2003)). Ce schéma emploie aussi un routage géo-localisé, avec des procédures de compensation dynamique de position puisque les nœuds intermédiaires font des décisions basés seulement sur leur information locale. En plus, les nœuds intermédiaires ont la capacité d'augmenter la vitesse de transmission d'un paquet dans le cas où le paquet ne puisse pas atteindre la destination avec sa vitesse actuelle. MMSPEED utilise toujours IEEE 802.11e comme sa couche MAC, et le protocole profite des mécanismes d'affectation de priorités qui est offert par 802.11. De cette façon, chaque vitesse correspond à une classe de priorité de la couche MAC. Bien que MMSPEED résout plusieurs aspects de la QoS pour le trafic multimédia dans un réseau de capteurs, il y en a d'autres, tels que l'agrégation des données ou la considération de l'énergie au moment de transmettre des messages dont le protocole ne s'occupe pas et qui sont relevant pour une architecture de communication multimédia. [33]

### **2.9.2.3 D'autres approches :**

Il existe dans la littérature, d'autres schémas de routage qui ne peuvent pas être classifiés comme basés sur l'IntServ ou sur le DiffServ. Ces protocoles sont spéciaux et résolvent des problèmes très spécifiques de la communication multimédia dans les RCSF.

Chen et al. (2007a) présentent une solution intéressante au problème de la transmission de vidéo dans un réseau de capteurs. Les auteurs partent de l'hypothèse que dans le réseau il n'y a qu'un capteur de vidéo (CV), et que tous les autres capteurs font seulement une autre activité : celle de véhiculer le vidéo du CV à la station base (le SINK). En plus, chaque capteur connaît parfaitement sa position géographique, de telle façon que le protocole proposé (nommé le "routage géographique directionnel" ou RGD) utilise les

positions des nœuds pour trouver la meilleure route vers la station base. Pour le protocole, la vidéo est d'abord codée en utilisant le standard H.26L, mais aucune raison n'est explicitée dans l'article sur les motifs ou les avantages de l'utilisation de telle codification. Le protocole a deux caractéristiques importantes qui font de lui une solution originale. La première caractéristique est l'utilisation de plusieurs routes pour envoyer le flot de vidéo. A partir d'un paramètre connu comme le PathNum (lequel indique le nombre de chemins qui seront formés pour envoyer les données), le capteur de vidéo détermine les angles dans lesquels il trouvera les nœuds qui achemineront les paquets. La figure 2.4 présente le schéma de construction de chemins du protocole RGD.

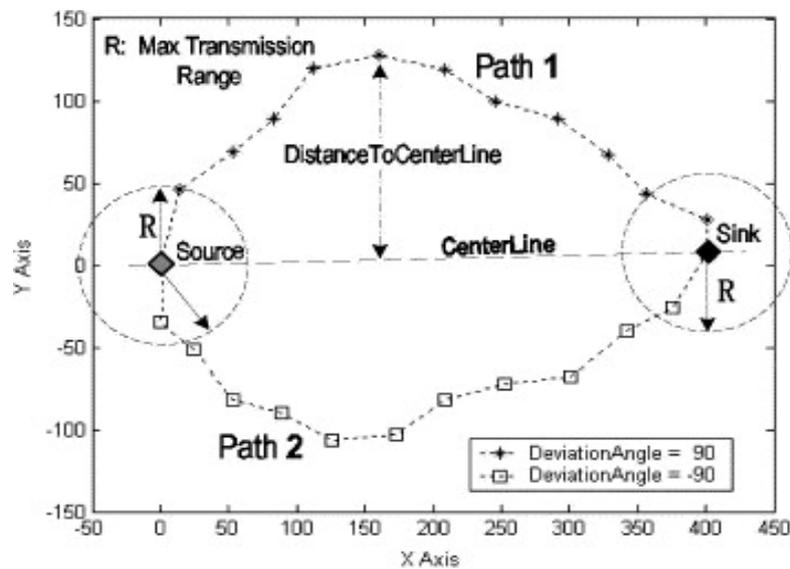


Figure 2.4 Exemple de routes formées par RGD avec PathNum = 2

Par chaque chemin, le capteur de vidéo envoie des sous-flots du flot de vidéo principal. Les sous-flots empruntent des routes différentes avec des nœuds non-partagés. Lors que le SINK reçoit les données, celui reconstruira la vidéo à partir de la fusion des paquets de chaque sous-flot. Les auteurs mentionnent que le routage multi-chemin permet d'augmenter la fiabilité de la transmission des données. Pour améliorer cette fiabilité, le protocole RGD transmet les paquets avec le code Forward Error Correction (FEC) pour permettre la correction, dans le SINK, des erreurs survenues au cours de la transmission sans besoin de retransmettre les paquets endommagés. Chaque sous-flot de  $n$  paquets envoyé par le capteur de vidéo consiste de  $k$  paquets d'une trame de la vidéo et  $n-k$  paquets redondantes qui sont générés pour protéger les  $k$  premiers paquets. Si le SINK reçoit  $k$  des  $n$  paquets, la trame de vidéo correspondante pourra être restaurée sans problèmes. En définitive, RGD offre un bon protocole pour la transmission de vidéo dans un réseau de capteurs. La combinaison de routage géographique, de multiples routes et du code FEC permet d'obtenir un protocole fiable et rapide, et ce, on peut le constater dans les résultats présentés dans l'article. Mais, les restrictions que ce protocole impose sont beaucoup : l'impossibilité d'avoir deux ou

plus sources de vidéo, la spécialisation du réseau pour la transmission de vidéo (par exemple, les capteurs ne peuvent pas capter de données scalaires comme la température ou la lumière), la manque de considération de l'énergie des nœuds intermédiaires au moment de choisir les routes, la nécessité d'un grand nombre de capteurs intermédiaires qui permettent la création de plusieurs routes disjointes (sans nœuds en commun). Ces inconvénients nous révèlent que ce protocole est à une étape préliminaire et qu'il y a de travaux à faire pour l'améliorer.

Une autre approche, bien qu'il ne soit que pour le transport d'images, est présentée par Wu et Abouzeid (2006). Le protocole présenté dans cet article utilise un routage basé sur la formation des grappes dans le réseau de capteurs (clusters), et aussi sur l'utilisation de multiples routes dans une grappe et sur le traitement d'images dans le réseau (in-network processing). Quand la source souhaite transmettre une image vers le SINK, celle l'envoie d'abord à plusieurs nœuds dans la même grappe à laquelle appartient la source. Chaque nœud achemine l'image vers la tête de la prochaine grappe du réseau. Partant du fait que la source a envoyé l'image par différents chemins, c'est sûr que la tête de grappe recevra plusieurs copies de la même image. Cela est l'avantage du protocole, puisque à partir de ces multiples images, la tête de grappe pourra choisir la meilleure ou simplement prendre de données de diverses images pour créer une bonne image de l'information reçue. Ce processus est appelé par les auteurs "in-network diversité combining" ou combinaison en diversité dans le réseau. En plus, les images sont envoyées en utilisant le schéma de protection d'erreurs FEC, que nous avons vu plus haut. En théorie, la combinaison de multiples routes avec le traitement dans le réseau et la redondance des données rend presque impossible la perte de l'information. Le routage continue à chaque tête de grappe. Celle fait le même processus que la source, c'est-à-dire, envoyer l'image à plusieurs nœuds dans la grappe, lesquels achemineront l'information vers la prochaine tête de grappe, et ainsi de suite, jusqu'à arriver au SINK. La figure 2.5 présente un schéma de ce type de routage.

Les inconvénients principaux de ce protocole résident dans la spécialisation du réseau pour ne transmettre que des images, l'utilisation très grande de la mémoire des têtes de grappe pour recevoir et stocker tous les images qui proviennent des nœuds routeurs intermédiaires, l'impossibilité de que plusieurs sources d'images transmettent au même temps et la manque de critères de QoS pour choisir les routes. Néanmoins, il y a des concepts et des idées de ce protocole dont on peut profiter pour en construire un autre qui travaille d'une façon semblable mais pour la transmission de vidéo. [33]

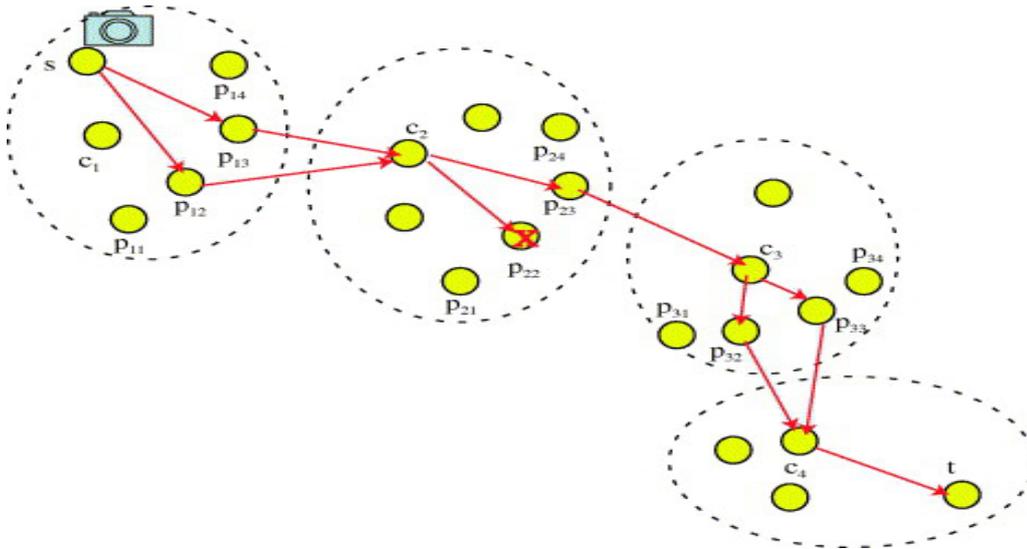


Figure 2.5 Schéma de routage du protocole

## **2.10 Consommation d'énergie dans les réseaux de capteurs sans fil :**

Les réseaux de capteurs sans fil doivent intégrer des mécanismes qui permettent aux utilisateurs de prolonger la durée de vie du réseau en entier, car chaque nœud est alimenté par une source d'énergie limitée et généralement irremplaçable. Dans un nœud capteur, l'énergie est consommée en assurant les fonctions suivantes : la capture, le calcul (traitement) et la communication. Cette dernière représente une grande portion de l'énergie totale consommée. De ce fait, la communauté de recherche est en train de développer et de raffiner plusieurs techniques de conservation d'énergie.

### **2.10.1 Types Consommation d'énergie dans les RCSF :**

La première étape dans la conception de système énergétique de capteurs consiste à analyser les caractéristiques de consommation d'énergie d'un nœud de capteur sans fil. Cette analyse systématique de l'énergie d'un nœud capteur est extrêmement importante pour identifier les problèmes dans le système énergétique pour permettre une optimisation efficace. L'énergie consommée par un capteur est principalement dû aux opérations suivantes : la détection, le traitement et la communication [23].

#### **2.10.1.1 Energie de capture**

Un capteur peut être équipé par multiples senseurs (humidité, la chaleur, les mouvements, la position, ...). De ce fait les sources de consommation d'énergie pour les opérations de détection ou de capture sont : l'échantillonnage, la conversion analogique-numérique, le traitement de signal et l'activation de la sonde de capture [23].

#### **2.10.1.2 Energie de traitement**

L'énergie de traitement est composée de deux sortes d'énergie: l'énergie de commutation et l'énergie de fuite. L'énergie de commutation est déterminée par la tension d'alimentation et la capacité totale commutée au niveau logiciel (en exécutant un logiciel). Par contre, l'énergie de fuite correspond à

l'énergie consommée lorsque l'unité de calcul n'effectue aucun traitement. En général, l'énergie de traitement est faible par rapport à celle nécessaire pour la communication. [23]

### **2.10.1.3 Energie de communication**

L'énergie de communication se décline en trois parties : l'énergie de réception, l'énergie de l'émission et l'énergie en état de veille. Cette énergie est déterminée par la quantité des données à communiquer et la distance de transmission, ainsi que par les propriétés physiques du module radio. L'émission d'un signal est caractérisée par sa puissance ; quand la puissance d'émission est élevée, le signal aura une grande portée et l'énergie consommée sera plus élevée. Notons que l'énergie de communication représente la portion la plus grande de l'énergie consommée par un nœud capteur. [23]

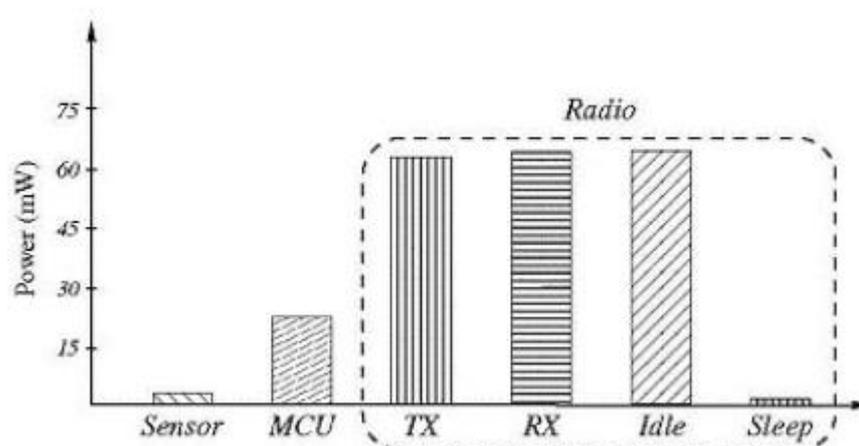


Figure 2.6 : La répartition de la consommation d'énergie d'un nœud de MicaZ

## **2.10.2 Sources de consommation d'énergie :**

Les sources de consommation d'énergie au niveau de la couche réseau sont diverses

### **2.10.2.1 La longueur des chemins :**

Les paquets de données suivent des chemins en un certain nombre de sauts. Le coût d'un saut en termes d'énergie est mesuré par la distance qui sépare les deux nœuds concernés par la communication, et le coût global du routage est la somme des énergies consommées à tous les sauts. Les chemins les plus longs sont les plus gourmands en énergie. [23]

### **2.10.2.2 La qualité des liens :**

La retransmission des données suite à l'interruption du cheminement entre la source et la destination est une opération qui engendre un coût énergétique supplémentaire. [23]

### **2.10.2.3 Le mode de communication :**

Le routage point-à-point est inefficace pour les RCSF du fait qu'il consomme beaucoup d'énergie au niveau des nœuds capteurs. Par ailleurs, les modes de communication adéquats sont en effet le mode « one-to-many » et le mode « manyto-one ». [23]

### **2.10.2.4 Le routage de paquets inutiles :**

Certains paquets de données deviennent inutiles si on ne respecte pas les échéances temporelles des transmissions, fixées soit par l'émetteur ou par le récepteur. Router des données dont l'échéance est expirée engendre une consommation d'énergie inutile. [23]

### **2.10.2.5 Le choix d'un chemin :**

La non-prise en compte de l'énergie du nœud lors du processus d'établissement de chemins peut provoquer l'épuisement des capacités énergétiques de certain nœud. [23]

## **2.11 La consommation d'énergie :**

Un capteur, à cause de sa taille, est équipé par de petite batterie ce qui le rend limité en énergie (<1.2V). Dans la plupart des cas le remplacement et le chargement de la batterie est impossible. Ce qui veut dire que la durée de vie d'un capteur dépend grandement de la durée de vie de la batterie. Dans un réseau de capteurs (multi-sauts) chaque nœuds collecte des données et envoie/transmet des valeurs. Le dysfonctionnement de quelques nœuds nécessite un changement de la topologie du réseau et un ré-routage des paquets. Toutes ces opérations sont gourmandes en énergie, c'est pour cette raison que les recherches actuelles se concentrent principalement sur les moyens de réduire cette consommation [23].

## **2.12 Les protocoles de routage dans les RCSF :**

### **2.12.1 Introduction**

Les protocoles de routage pour les réseaux de capteurs sans fil ont été largement étudiés, et différentes études ont été publiées. Ces protocoles doivent garantir l'acheminement de l'information entre tout nœud du réseau et la station de base à moindre coût en termes d'énergie. Dans les réseaux de capteurs, chaque nœud joue le rôle de source et de relais. De ce fait, la défaillance énergétique d'un capteur peut changer significativement la topologie du réseau donc il faut minimiser les dépenses énergétiques, car l'énergie est une contrainte clé dans les réseaux de capteurs.

L'objectif de ce chapitre est de présenter quelques approches et techniques sur lesquelles se basent les protocoles de routage dans les réseaux de capteurs, on effectue une comparaison entre ces protocoles. [35]

### **2.12.2 Définition de routage :**

Le routage est un processus qui permet de sélectionner des chemins dans un réseau pour transmettre des données depuis un expéditeur jusqu'à un ou plusieurs destinataires. Sa performance est importante dans les réseaux décentralisés, c'est-à-dire où l'information n'est pas distribuée par une seule source, mais échangée entre des agents indépendants. [35]

### **2.12.3 Routage dans les réseaux de capteurs sans fil :**

La propagation et la délivrance des données dans un RCSF représentent la fonctionnalité la plus importante du réseau. Elle doit prendre en considération toutes les caractéristiques des capteurs afin d'assurer les meilleures performances du système : durée de vie, fiabilité, temps de réponse ... etc. Le problème de routage consiste à déterminer un acheminement optimal des paquets à travers le réseau au sens d'un certain critère de performance. Le problème consiste à trouver l'investissement de moindre coût en capacités nominales et de réserves qui assure le routage du trafic nominal et garantit sa survie en cas de n'importe quelle panne d'arc ou de nœud. Le problème qui se pose dans le contexte des réseaux ad hoc est l'adaptation de la méthode d'acheminement utilisée avec le grand nombre d'unités existant dans un environnement caractérisé par de modestes capacités de calcul et de sauvegarde et de changements rapides de topologies. Il semble donc important que toute conception de protocole de routage doive étudier les problèmes suivants : [35]

1. La minimisation de la charge du réseau
2. Offrir un support pour pouvoir effectuer des communications multipoints fiables
3. Assurer un routage optimal
4. Offrir une bonne qualité concernant le temps de latence

### **2.12.4 Les Protocoles de routage :**

Dans le cas des réseaux ad hoc, l'architecture est caractérisée par une absence d'infrastructure fixe, les nœuds doivent donc s'organiser automatiquement et réagir rapide mentaux différents mouvements. Ainsi dans le cas des réseaux ad hoc, les nœuds sont considérés comme des routeurs, chaque nœud participe dans le fonctionnement de routage. Le protocole de routage est un programme qui sert à déterminer la route optimale pour le transfert des données entre deux nœuds. Le routage dans les réseaux mobiles ad hoc est différent de routage traditionnel utilisé dans les réseaux a infrastructure ce dernier dépend de plusieurs facteurs dont la mobilité, la topologie, la manière de la sélection de la route.

Il est à noter que Les protocoles de routage ad hoc sont divisés en trois catégories, deux principales et la troisième est issue de leur combinaison, ces catégories sont : Les protocoles réactifs, les protocoles proactifs et les protocoles hybrides. La différence entre les deux grandes catégories est que les réactifs doivent initialiser le chemin, entre la source et la destination avant d'envoyer les données par une demande de chemin. Par contre les proactifs ont dans leur table de routage tous les chemins vers tous les nœuds du réseau, il ne leur manque que d'utiliser ce chemin pour envoyer directement les données à la destination. Les protocoles hybrides sont le compromis entre les protocoles réactifs et les protocoles proactifs [35].

#### **2.12.4.1 Protocoles Proactifs :**

Les protocoles de routage proactifs dans les réseaux ad hoc, sont basés sur le même principe des protocoles de routage utilisés dans les réseaux filaires. Les deux principales méthodes sont la méthode " états des liens" et la méthode "vecteur de distance". Ces deux méthodes essaient de maintenir les meilleurs chemins existants vers toutes les destinations possibles (qui peuvent représenter l'ensemble de tous les nœuds du

réseau) au niveau de chaque nœud de réseau. Les routes sont sauvegardées mêmes si elles ne sont pas utilisées.

La sauvegarde des routes, permanente des chemins de routage, est assuré par un échange continu des messages de mise à jour des chemins, ce qui induit un contrôle excessif surtout dans le cas des réseaux de grande taille. L'avantage de ces protocoles réside dans la disponibilité des informations de routage pour router un paquet vers n'importe quelle destination. L'inconvénient majeur est le coût de maintenance des connaissances sur la topologie et de routage du a l'envoi périodique des messages. Ceci génère une consommation continue de la bande passante. [35]

#### **2.12.4.1.1 DSDV (Destination Sequenced Distance-Vector Routing)**

Dans son fonctionnement de base, le protocole DSDV partage la même philosophie avec les algorithmes à vecteur de distances. La contribution principale du protocole DSDV est l'utilisation des numéros de séquence qui permettent d'éviter la formation des boucles de routage. Quand un nœud reçoit plusieurs paquets de mise à jour au sujet d'un même nœud destination, il choisit celui avec le numéro de séquence le plus haut. Un nœud qui détecte la rupture d'un lien, génère un paquet de mise à jour dont le numéro de séquence possède une valeur infinie. En recevant ce paquet, chaque nœud retire l'entrée correspondante de sa table de routage. Pour assurer la consistance des tables de routage, les mises à jour de ces derniers s'effectuent périodiquement et immédiatement après un changement de topologie. Pour réduire la quantité du trafic que génèrent ces mises à jour. Principe de fonctionnement de ce protocole est chaque nœud dans le réseau diffusé par inondation un paquet de mise à jour des tables de routage qui inclue les destinations accessibles et le nombre de sauts exigé pour atteindre chaque destination avec le numéro de séquence lié à chaque route. Des paquets de mise à jour sont aussi diffusés immédiatement s'il y a un changement dans la topologie du réseau afin de propager les informations de routage aussi rapidement que possible. À la réception d'un paquet de mise à jour, chaque nœud le compare avec les informations existantes dans sa table de routage. Les routes les plus récentes (qui ont le plus grand numéro de séquence) avec la distance la plus courte sont gardées, les autres sont simplement ignorées [35].

#### **2.12.4.2 Protocoles Réactifs**

Les protocoles de routage réactifs (ou à la demande) ont été conçus pour minimiser la charge de contrôle des protocoles de routage proactifs en ne maintenant des informations que concernant les chemins actifs. Les protocoles de routage réactifs peuvent être classifiés en deux catégories : à routage source ou à routage saut-par-saut. Dans les protocoles à routage source, les paquets de données portent dans leurs entêtes les adresses de tous les nœuds constituant le chemin à partir de la source jusqu' à destination. De ce fait, les

nœuds intermédiaires acheminent les paquets selon les informations qui se trouvent dans l'entête de chaque paquet de données. Cela veut dire que les nœuds intermédiaires ne sont pas besoin de maintenir des informations sur les chemins actifs.

De plus, ils ne sont pas besoin de maintenir la connectivité avec leurs voisins. Dans le routage saut-par saut chaque paquet de données porte uniquement l'adresse de la destination et celle du saut prochain. De ce fait, chaque nœud intermédiaire utilise sa table de routage pour acheminer chaque paquet de donnée.

L'avantage de ce type de routage est qu'il permet une meilleure adaptabilité aux changements fréquents de topologie, car chaque nœud utilise les informations les plus fraîches dans le routage. L'efficacité de l'approche réactive commence à se diminuer quand le trafic et la mobilité deviennent plus importants. De plus, les protocoles de routage réactifs possèdent un temps de réponse plus élevé que celui des protocoles de routage proactifs ce qui peut influencer sur les performances des applications interactives [35].

#### **2.12.4.2.1 AODV (ad hoc On-Demand Distance-Vector)**

AODV est un protocole de routage réactif, base sur le principe des vecteurs de distance, capable à la fois de routage unicast et multicast Il représente essentiellement une amélioration de protocole DSDV et il repose sur le mécanisme de découverte de chemins à la demande mais il n'utilise pas le routage source, AODV repose sur le routage saut par saut. Chaque nœud dans AODV, maintient une table de routage et utilise les numéros de séquence comme DSDV pour éviter le problème des boucles de routage. On générale AODV fonctionne par l'utilisation de trois types de messages :[35]

- Les messages de demande de route RREQ : Route Request Message.
- Les messages de réponse de route RREP : Route Reply Message.
- Les messages d'erreur de route RERR : Route Error Message.

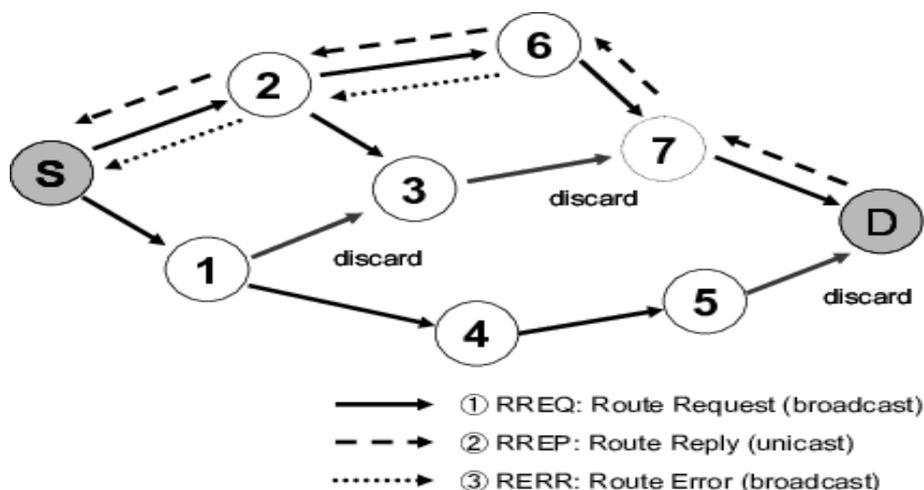


Figure 2.7 Découverte des chemins dans AODV

Et basé sur l'utilisation de deux mécanismes « Découverte de route » et « maintenance de route. »

Lorsque les nœuds veulent établir une communication et il n'existe pas de route valide entre un nœud source et destinataire, AODV diffuse dans le réseau une requête de demande de route RREQ (Route REQuest). Si un nœud quelconque dans le réseau possède une route à la destination il renvoie une réponse par la requête RREP (Route REPLY) à l'émetteur, et si ce dernier ne reçoit pas de réponse RREP pendant une Période appelé (RREP-WAIT-TIME) il inonde encore une fois la requête RREQ .

Chaque nœud dans AODV met à jour périodiquement la liste de ses voisins, un nœud transmet un message HELLO, Si un nœud ne reçoit pas d'un voisin trois messages HELLO consécutifs le lien avec le voisin est considéré invalide. Et si ce dernier est invalide (à cause de la mobilité ou la défaillance d'un nœud), les nœuds utilisant ce lien sont prévenus par un message d'erreur (RERR), ils vont alors diffuser une autre requête. La figure ci dessus illustre la coupure d'un lien entre deux nœuds et l'envoi du RERR dans AODV [35].

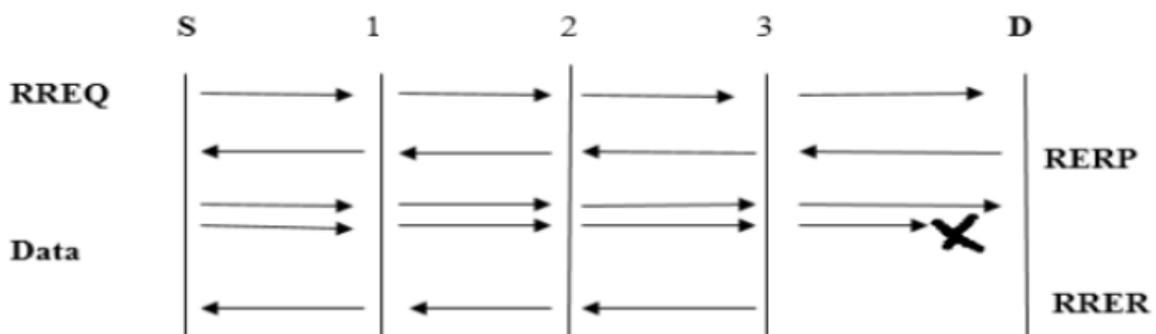


Figure 2.8 Coupure de route et envoi du RERR

#### **2.12.4.2.2 DSR (Dynamic Source Routing)**

Le protocole DSR est un protocole réactif basé sur le routage par la source, c'est-à-dire que la source des données détermine le chemin complet par lequel les données vont transiter et ce dernier sera transmis avec les données. Dans chaque paquet de données transmis, il y a un champ qui contient la séquence de nœuds à suivre pour atteindre la destination.

Si un destinataire est dans le cache du nœud source, la route connue est utilisée. Sinon, une procédure de découverte de route est déclenchée. Les paquets de découverte de route contiennent les adresses source et destination ainsi qu'un identifiant permettant aux nœuds intermédiaires de savoir s'ils ont déjà traité les paquets. Le chemin vers la destination est créé dans le paquet de recherche de route. Chaque nœud qui reçoit ce paquet ajoute à la route préexistante dans ce paquet sa propre adresse. Lorsqu'un des paquets de

recherche de route atteint sa destination ou un nœud qui possède une route valide vers la destination, ce nœud répond à la source initiale [35].

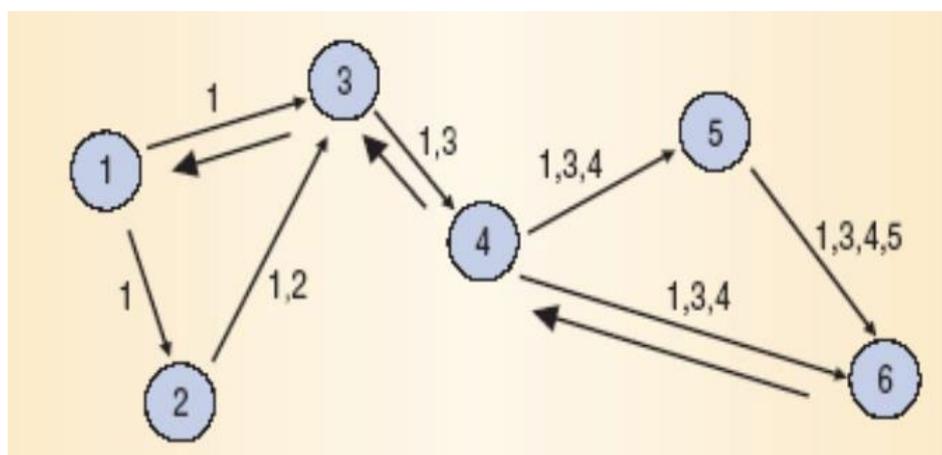


Figure 2.9 Mécanisme de découverte de route dans DSR

Ce nœud ne possède pas de route vers la source initiale, donc une nouvelle recherche de route doit être entreprise. Pour éviter un bouclage infini des recherches de route, la route construite est ajoutée à la nouvelle recherche de route, de sorte que le nœud source connaisse une route valide pour sa réponse.

A fin d'assurer la validité des chemins utilisés, DSR exécute une procédure de maintenance de routes. Quand un nœud détecte un problème fatal de transmission, à l'aide de sa couche de liaison, un message RERR est envoyé à l'émetteur original du paquet. Le message d'erreur contient l'adresse du nœud qui a détecté l'erreur et celle du nœud qui le suit dans le chemin.

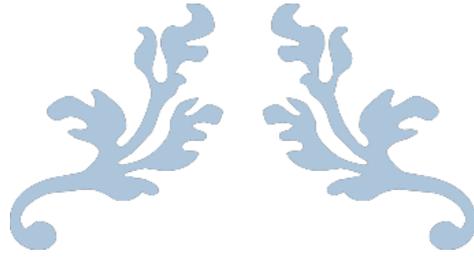
L'un des inconvénients de protocole DSR est l'ajout dans les paquets de données qui accroît le surcoût et consomme plus de bande passante et d'énergie, et aussi le délai avant de commencer la transmission des paquets provoqués par la procédure de découverte de route [35].

### **Conclusion :**

Les récents progrès technologiques ont permis l'émergence de petits appareils à faible consommation dotés à la fois de capacités de captures, de traitements et de communications sans fil. Cependant, ces nœuds souffrent du manque de ressources et surtout celles énergétiques, du fait qu'ils sont alimentés par une batterie dont le remplacement est indésirable et dans certains cas impossibles. Les réseaux constitués de ces nœuds, communément appelés réseaux de capteurs sans fil, attirent une attention considérable. La majorité de cette attention, cependant, a été concentrée sur l'efficacité énergétique de la phase de transmission. Ceci est en effet dû au fait que les données transmises sont scalaires, telles que la température et l'humidité, et ne nécessitent pas alors un traitement important.

Nous avons étudié les caractéristiques et le mécanisme des protocoles proactif et réactif, Nous avons distingué trois protocoles de routage, les protocoles de routage AODV et DSR sont des protocoles réactifs et DSDV est du type proactif. D'après l'étude théorique et la comparaison entre ces trois protocoles on peut dire que, le mécanisme de protocole DSR est le plus efficace par rapport aux deux autres protocoles. Les critères de la qualité de service permettant à nous de choisir les meilleurs protocoles de routage au niveau de performance de protocole et pour réduire la perte des paquets, le délai et augmenter le débit sur les réseaux.





# **CHAPITRE 3**

**Résultats**

**de**

**Simulation**



### **3.1. Introduction**

Pour les réseaux sans fil, la simulation permet de tester à moindre coût les nouveaux protocoles de routage ainsi que d'anticiper les problèmes qui pourront se poser dans le futur afin d'implémenter la technologie la plus adaptée aux besoins. Dans ce contexte, le simulateur NS2 est un logiciel de simulation à événements discrets de réseaux informatiques. Ce logiciel a été initialement développé dans le cadre du projet VINT (Virtual Inter Network Testbed), dont l'objectif principal visait la construction d'un simulateur multi-protocole pour faciliter l'étude de l'interaction entre les protocoles et le comportement d'un réseau à différentes échelles.

Ce projet contient des bibliothèques pour la génération de topologies réseau, des trafics ainsi que des outils de visualisation tels que l'animateur réseau NAM (network animator). Il est essentiellement élaboré avec les idées de la conception par objets, de la réutilisation du code et de modularité. Il est aujourd'hui un standard de référence en ce domaine, plusieurs laboratoires de recherche recommandent son utilisation pour tester les nouveaux protocoles. L'objectif de ce chapitre est d'étudier les caractéristiques des trois protocoles AODV, DSDV, et DSR étudiés dans nos simulations. Par conséquent, un nœud qui souhaite envoyer un paquet à un autre nœud, les protocoles réactifs recherchent l'itinéraire sur une base à la demande et établissent une connexion pour transmettre et recevoir un paquet. La découverte de route consiste généralement en une inondation de messages de requête sur l'ensemble du réseau. En revanche, dans le routage proactif, chaque nœud maintient en permanence la route entre deux nœuds.

Dans ce chapitre nous allons étudier les performances et qualité de service d'un réseau sans fil. Nous allons utiliser trois protocoles de routage qui sont : AODV, DSDV et DSR . Nous allons diviser ce chapitre en deux parties : la première partie est consacrée au critère de la qualité. Les paramètres particuliers de Qos que nous avons choisis qui sont les pertes, le débit utile et le délai de transmission. Dans la deuxième partie, nous présentons les résultats de simulation qui décrivent les performances du réseau simulé pour les trois types de routage.

Les simulateurs réseaux peuvent être divisés en plusieurs types : par protocoles, par technologie ou par méthodes de traitement, mais la catégorisation la plus utilisée est les méthodes de simulation. Il existe typiquement deux méthodes de simulation : méthode à événements discrets et méthode de simulation analytique. La première méthode produit des prédictions dans le bas niveau du réseau (paquet par paquet), pour qu'ils deviennent exacts mais la génération des résultats est lente. La deuxième méthode utilise des modèles mathématiques pour produire les résultats à une vitesse beaucoup plus rapide, mais elle peut sacrifier l'exactitude.

### **3.2 Préparation de l'environnement :**

Plusieurs environnements de simulation sont utilisés pour évaluer les performances des protocoles et des architectures proposés pour les RCSFs. Certains sont libres et parfois Open Source par contre il y en a d'autres qui sont commercialisés. Aucun de ces simulateurs n'est parfait et ne répond à tous les besoins. Chacun d'eux présente des avantages et des inconvénients. [36]

### **3.3 Outils de simulation :**

Les besoins croissants de tester les nouvelles technologies et les nouveaux protocoles avant leur déploiement a conduit à la prolifération des simulateurs. On peut les classer en deux types : les logiciels libres et gratuits tels que OMNet++, J-Sim et NS2... et les logiciels commerciaux tels que OPNET et NetRule .

Dans notre cas d'étude on a choisi le simulateur ns-2.34 pour mesurer les performances. [36]

#### **3.3.1 Le simulateur NS-2 :**

C'est un simulateur développé à Lawrence Berkeley National Laboratory (LBNL). Il est conçu principalement pour le monde de l'Internet. Il permet de simuler le comportement des protocoles standard de l'Internet tels que TCP, IP et permet d'étendre les émulateurs aux nouveaux protocoles de l'Internet (routage, transport, application) et aux nouvelles architectures de qualité. NS2 est écrit en C++ et utilise le langage OTCL (Object Tools Command Language) dérivé de TCL. A travers OTCL, l'utilisateur décrit les conditions de la simulation : la topologie du réseau, les caractéristiques des liens Physiques, les protocoles utilisés, les communications qui ont eu lieu. La simulation doit d'abord être saisie sous forme de fichier que NS va utiliser pour produire un fichier contenant les résultats.

Mais l'utilisation de l'OTCL permet aussi à l'utilisateur de créer ses propres procédures. L'outil NAM (Network Animator) associé au simulateur NS permet de visualiser des animations de la simulation (transfert des paquets d'un nœud à un autre, taille des paquets, remplissage des files d'attente. . .) [36]

### 3.3.2 Composants de NS-2 :

<b>Application</b>	Web, ftp, telnet, générateur de trafic (CBR, ...)
<b>Transport</b>	TCP, UDP, RTP, SRM
<b>Routage</b>	Statique, dynamique (vecteur distance) et routage multipoint (DVMRP, PIM)
<b>Gestion de file d'attente</b>	RED, DropTail, Tokenbucket
<b>Discipline de service</b>	CBQ, SFQ, DRR, Fairqueueing
<b>Système de transmission</b>	CSMA/CD, CSMA/CA, lien point à point

Tableau 3.1 : Composants de NS-2 [35]

### 3.3.3 L'architecture NS-2 :

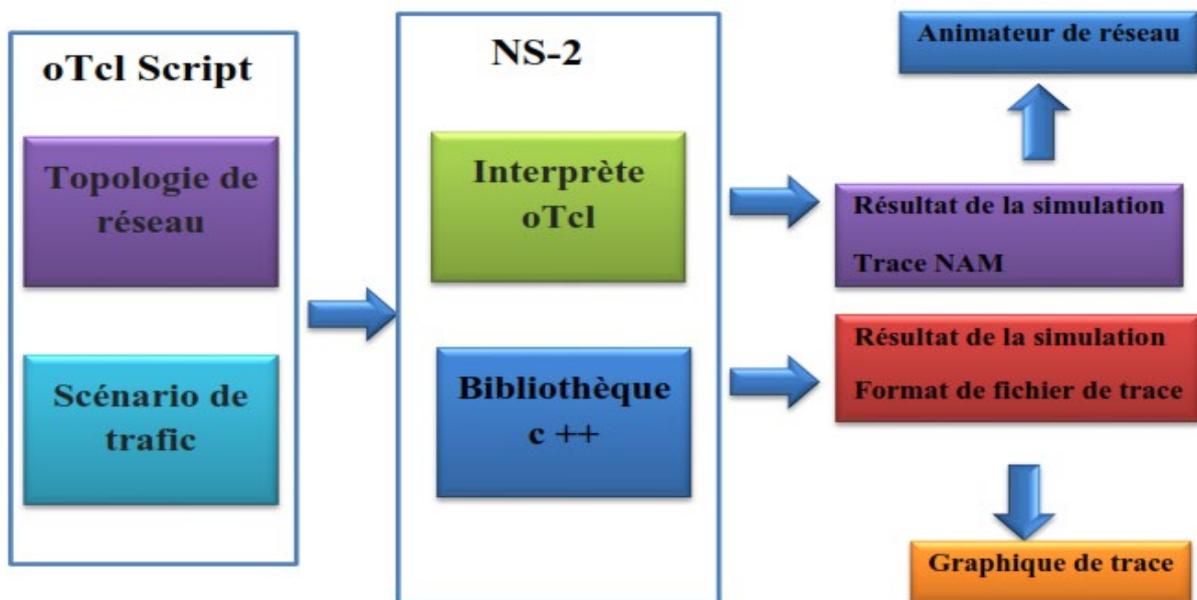


Figure 3.1 Composants de NS-2 [35]

### 3.3.4 Choix du simulateur NS-2 :

- **Avantages**

- Un logiciel de simulation multicouche.
- Un outil complètement libre pour plusieurs plateformes.
- Permet l'ajout de composants à la demande.
- Développement orienté objet.

- Support pour la simulation de la plupart des versions de TCP, des protocoles de routage et de diffusion restreinte (multicast) et pour la simulation des mécanismes de gestion et des disciplines de service des routeurs.
- Du fait de sa popularité, de nombreux protocoles sont a priori disponibles pour NS2.
- Quelques protocoles spécifiques aux réseaux de capteurs sont disponibles.
- L'analyse des résultats est en général peu aisée, le résultat de la simulation étant essentiellement composé d'un fichier retraçant l'ensemble des envois, réceptions et suppressions de paquets. Un certain nombre de scripts ont été développés (ou sont en cours de développement) pour faciliter cette analyse [36].

- **Inconvénients**

- Contient peu de paramètres de configuration dans ses modèles standards.
- NS2 a été initialement conçu pour simuler des réseaux filaires . Le support des réseaux mobiles a été rajouté ultérieurement.
- Les performances du simulateur sont assez limitées et ne semblent pas adaptées pour des simulations de réseaux importantes.
- Ne supporte pas un grand nombre de nœuds. Il est lent en exécution dès que le nombre de nœuds devient important [36]

### **3.3.5 Arborescence des fichiers :**

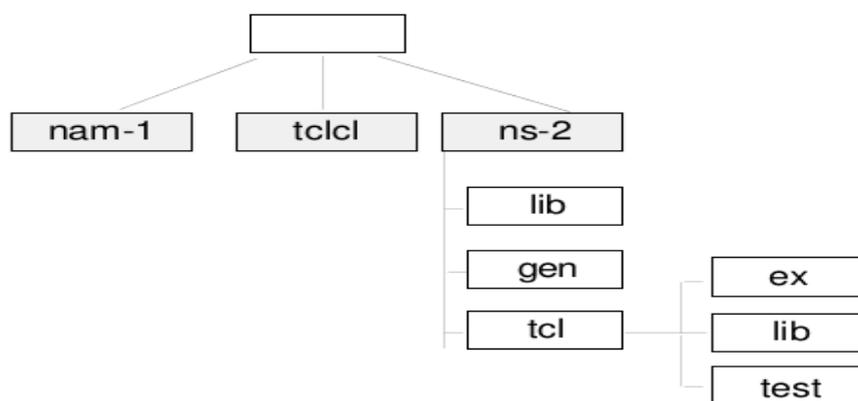


Figure 3.2 : Arborescence des fichiers de la distribution NS [36]

**La distribution de NS comprend principalement 3 répertoires :**

- nam-1, l'outil de visualisation des résultats de la simulation: l'animateur réseau.
- tclcl, sources du code assurant la liaison entre l'interpréteur et le simulateur. Citons l'un des principaux fichiers : tcl-object.tcl.
- ns-2, l'application NS. Ce répertoire contient l'ensemble des fichiers .h et .cc de NS.

**Dans le répertoire ns-2, on trouve les répertoires :**

- tcl pour tous les codes interprétés.
- bin pour les utilitaires et les exécutables pour la réalisation du binaire ns-2 ;
- lib pour la bibliothèque de libg++ ;
- gen pour les sources générées lors de la réalisation du binaire ns-2 par le Makefile.
- test\_output pour les résultats des simulations.
- tous les fichiers .h et .cc des classes C++ du simulateur.

**Le répertoire tcl contient les répertoires :**

- lib pour les méthodes OTcl des classes de l'arborescence interprétée. Dans ce répertoire, les fichiers ns\_lib.tcl et ns\_default.tcl ont un rôle particulier. Le premier est interprété automatiquement au lancement de NS. Il contient l'ensemble du code interprété de NS via la commande Tcl "source" (lecture d'un fichier) des différents fichiers OTcl. Les valeurs par défaut affectées aux objets sont déclarées dans le second. Les autres fichiers de ce répertoire sont l'API OTcl de NS. Ils ont un nom préfixé par ns\_.
- ex pour les scripts de simulation donnés à titre d'exemple.
- test, il recense l'ensemble des scripts des suites de tests pour la validation du simulateur.

On peut également prendre le contenu de ce répertoire comme des exemples. Les autres répertoires contiennent les codes interprétés des contributions. [36]

**3.4 Processus de simulation :**

Le simulateur a besoin du fichier scénario Tcl comme entrée, le scénario prévu du réseau est présenté en tant que séquence de commandes Tcl qui sont alimentés à un simulateur de réseau, et le simulateur produit le résultat d'analyse des performances réseau dans deux fichiers séparés : [36]

- Fichier trace (.tr)
- Fichier NAM (.nam)

## 1. Fichier (.tcl)

Le langage TCL est un langage de script puissant qui permet d'utiliser éventuellement une approche de programmation orienté objet. Il est facilement extensible par un certain nombre de modules.

Ce fichier est amené à un simulateur du réseau qui génère des fichiers additionnels, pour le traçage. Le script d'analyse du fichier (fichier.awk) prend le fichier de trace en entrée et écrit ses résultats dans des fichiers personnalisés, ce qui sont à nouveau utilisés pour un traitement ultérieur. [36]

```

set ns [new Simulator]
#ns-random 0

set f [open 1_out.tr w]
$ns trace-all $f
set namtrace [open 1_out.nam w]
$ns namtrace-all-wireless $namtrace $val(x) $val(y)
set f0 [open proj_out0.tr w]
set f1 [open proj_out1.tr w]
set f2 [open proj_out2.tr w]
set f3 [open proj_out3.tr w]

set topo [new Topography]
$topo load_flatgrid 800 800

create-god $val(nn)

set chan_1 [new $val(chan)]
set chan_2 [new $val(chan)]
set chan_3 [new $val(chan)]
set chan_4 [new $val(chan)]

# CONFIGURE AND CREATE NODES

$ns node-config -adhocRouting $val(rp) \
               -llType $val(ll) \
               -macType $val(mac) \
               -ifqType $val(ifq) \
               -ifqLen $val(ifqlen) \
               -antType $val(ant) \

```

Tcl ▾ Largeur des tabulations: 8 ▾ Lig 2, Col 17 INS

Figure 3.3 : Exemple de TCL.

## 2. Fichier trace (.tr)

Le fichier de trace contient des informations sur les divers événements qui ont eu lieu au cours de la simulation. Il contient tous les détails du comportement des nœuds, les transmissions et les réceptions de paquets, type de paquet, la couche responsable de la communication, les paquets perdus et les raisons de la perte, de la consommation d'énergie, etc ... [36]

L'exemple ci-dessous décrit le processus général de la manière dont une simulation de réseau est effectuée sous NS-2. Les fichiers de sortie tels que les fichiers de trace doivent être analysés pour extraire des informations utiles. L'analyse peut être effectuée à l'aide de la commande awk (sous UNIX et LINUX ou

le script perl. Les résultats ont été analysés avec un logiciel capable de raccourcir le processus d'analyse des fichiers de trace (Xgraph et Gnuplot) a également été utilisé dans cet article [24].

La signification des différents champs du fichier de trace est décrite comme suit :

**Exemple :**

```
+ 5.7550 0 1 cbr 512 ----- 0 2.0 3.0 4562 102
- 7.5469 0 1 cbr 512 ----- 1 2.0 3.0 4586 103
D 8.7739 2 3 cbr 512----- 1 2.0 3.0 4586 99
R 10.9600 2 3 cbr 512 ----- 1 2.0 3.0 4596 253
```

Chacune des lignes correspond à un évènement survenu à un paquet pendant la simulation Chaque colonne contient des informations comme suit :

- Effectuée sur le paquet. Un « + » signifie que le paquet est reçu dans une file, un « - » signifie que le paquet quitte la file, un « d » signifie que le paquet est jeté et un « r » signifie que le paquet est réceptionné par un agent.
- Instant ou l'action est effectuée.
- Noeud de départ du lien concerné.
- Noeud d'arrivée du lien concerné.
- Type de paquet.
- Taille du paquet en octets.
- Flags
- Identificateur de flux.
- Agent de départ.
- *Agent d'arrivée.*
- Numéro de séquence.
- Identificateur unique pour chaque paquet.

**3. Le fichier NAM (.nam) :**

Contient des informations sur la topologie, par exemple, les nœuds, les liens, ainsi que des traces de paquets. On peut dire que c'est un miroir du fichier de trace, à l'exception qu'il utilise une syntaxe différente c'est de travailler avec le visualiseur. [36]

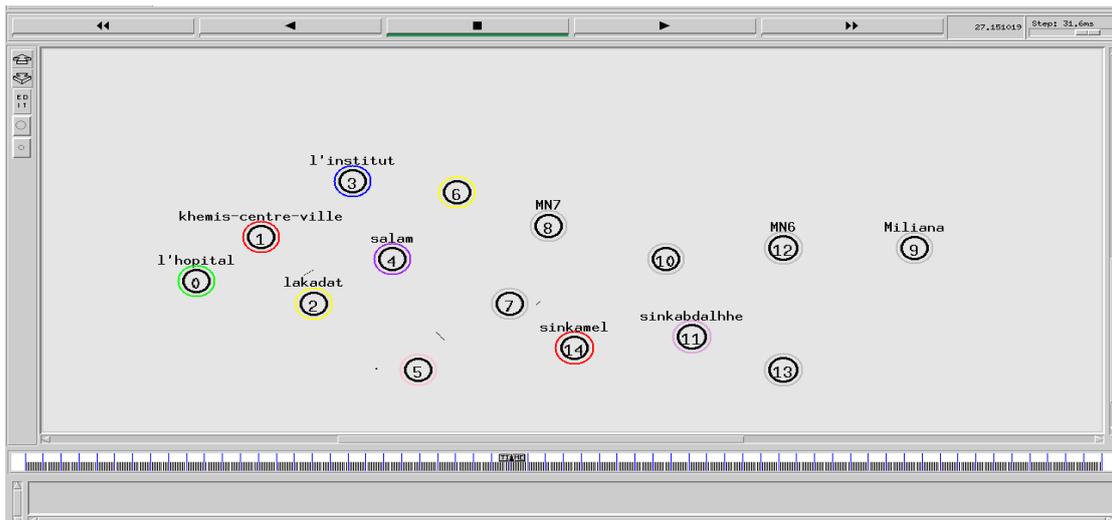


Figure 3.4 : Exemple de Nam. [36]

#### **4. Gnuplot :**

C'est un outil de type utilitaire graphique pour L'Unix, son code est protégé par copyright, mais il est distribué gratuitement. Il a été créé pour permettre aux scientifiques et aux étudiants de visualiser des fonctions mathématiques par des graphes 2D et 3D . [36]

#### **5. Fichier (.scn) :**

C'est un fichier qui représente la position des nœuds autour du Coordinateur.

Actuellement, ce fichier est généré avec l'utilitaire (scn.gen) C'est un Fichier texte simple, qui peut être facilement adapté pour placer les nœuds à des positions souhaitées. Notez que les positions des nœuds doivent respecter les limites du réseau mentionné dans le fichier source, Fichier.tcl. [35]

#### **6. Fichier.awk :**

##### **Paramètre :**

Le fichier doit être analysé avec le fichier de trace pour générer les résultats analysés.

Fonctionnement : Le fichier peut prendre en charge la génération des indicateurs de performance (les métriques) suivants : [35]

1. Débit
2. Délai minimum
3. Délai maximum
4. Délai moyen
5. Paquets de données transmis
6. Les paquets de données reçues avec succès par leurs destinations respectives
7. Le taux de livraison de paquets de données
8. Énergie moyenne utilisée

9. Pourcentage moyen de l'énergie consommée

10. Statistiques des paquets perdus

### **3.5 Paramètres de simulation :**

#### **Débit (throughput) :**

Il est une mesure de la quantité de données transmise depuis la source vers la destination dans une période unitaire de temps (seconde).

Le débit est mesuré en nombre total de bits reçus par seconde totales. Le débit d'un nœud est calculé comme suit :

**Débit d'un nœud** = Le nombre total des bits reçus \* Le temps d'exécution.

Le débit du réseau est défini par :

**Débit moyen** = Somme du débit des nœuds du réseau / Nombre de nœuds

#### **Taux de livraison de paquets (%)**

Il indique le pourcentage des paquets de données transmis qui ont été bien reçus.

Premièrement le nombre total des paquets transmis est compté, suivie par le nombre total de paquets reçus et du nombre total de paquets perdus.

Le taux de livraison est calculé comme le pourcentage des paquets reçus par les paquets transmis.

**Taux de livraison** = (Nombre de paquets reçus / Nombre de paquets transmis) \*100

#### **Le délai**

Le délai est le temps entre l'envoi d'un paquet par un émetteur et son temps de la réception par le destinataire.

**Délai** = (temps de réception du paquet) – ( le temps de transmission du paquet).

### **3.6 Exemple de Scénarios de simulation :**

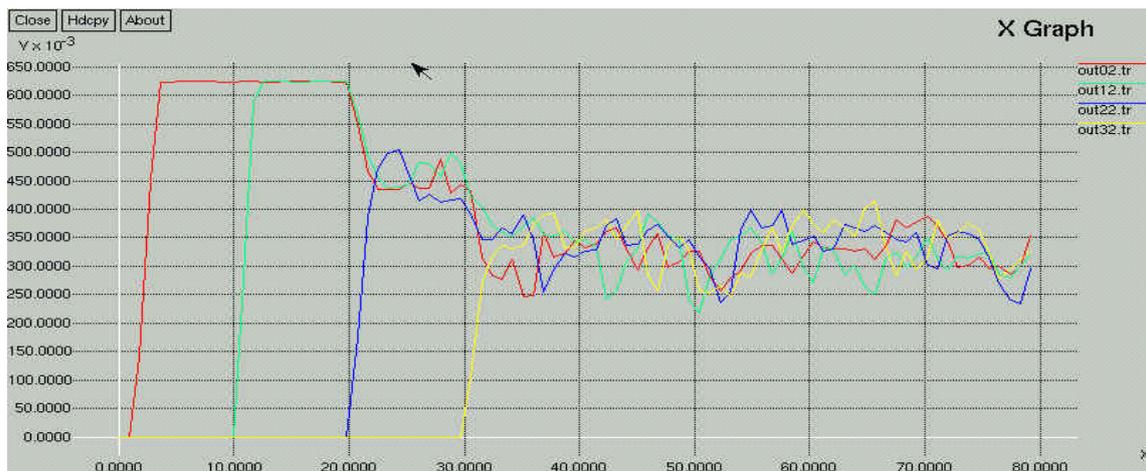
Un exemple de lecture des courbes...

Comment mesurer le débit, le taux d'abandon de paquets et le délai de bout en bout pour une application UDP sur des réseaux sans fil ?

## Le Scénario

Il se compose de 8 nœuds mobiles : 4 nœuds sources et 4 nœuds de destination. Chaque source est une source CBR sur UDP. La taille d'un paquet transmis est de 512 octets. La vitesse de transmission d'un nœud est ( $y \cdot 10^3$ ) de 600 Kbps. Nous avons supposé que les nœuds sont à portée de transmission à une distance constante de. Le temps de simulation a duré ( $x$ ) 80 secondes.

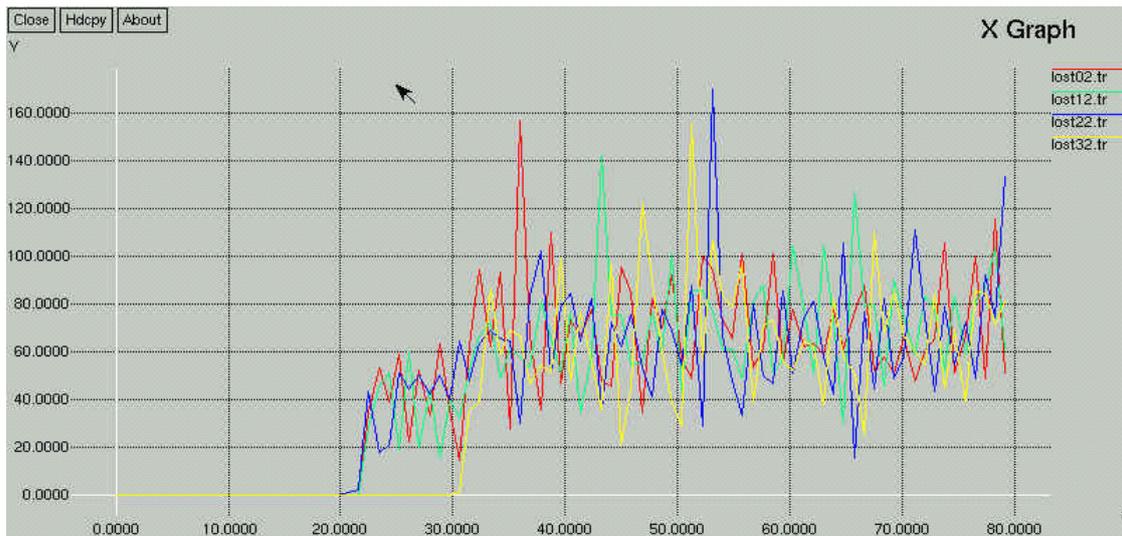
## Débit:



Graphe 3.1 Le débit

**Analyse:** Le nœud 1 commence à émettre au temps  $T = 1,4$  s tandis que le nœud 2 commence à transmettre au temps  $T = 10$  s. Pendant la période de temps  $[1,4$  s,  $10$  s], le nœud 1 est le seul nœud de transmission utilisant toute la bande passante disponible. Cela justifie les performances élevées du nœud 1 pendant l'intervalle de temps spécifié. Au temps  $T = 10$  s, le nœud 2 commence la transmission, partageant ainsi les ressources du canal avec le nœud 1. Ceci explique la forte réduction du débit binaire. De plus, le tracé du débit binaire subit des oscillations et une réduction plus importantes à mesure que le nombre de nœuds de transmission augmente. Les oscillations se traduisent par de graves troubles des performances du réseau.

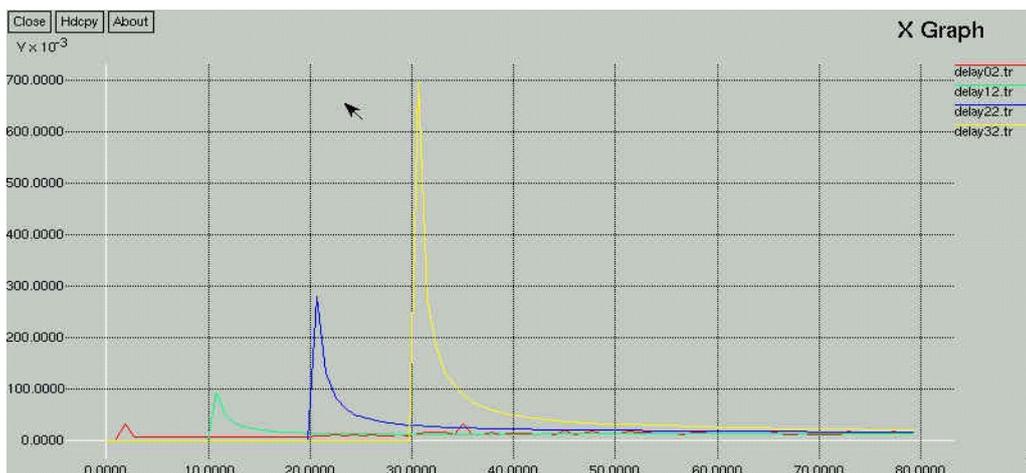
**Taux d'abandon des paquets:**



Graphe 3.2 Taux d'abandon des paquets

**Analyse:** cette figure montre un taux d'abandon de paquets élevé chaque fois que le nombre de nœuds partageant des ressources réseau augmente. On peut montrer que le taux d'abandon de paquet dans l'intervalle [1,4 sec, 10 sec] est égal à 0. Cela peut être facilement justifié car un seul nœud utilise le réseau pendant cet intervalle de temps. Cependant, ces performances de haute qualité se détériorent à mesure que davantage de nœuds commencent à partager les ressources du réseau.

**Délai moyen de bout en bout de paquets :**



Graphe 3.3 Délai moyen de bout en bout de paquets

**Analyse :** lorsque le nombre de nœuds partageant les ressources du réseau, le délai augmente considérablement et le réajustement de chaque nœud prend plus de temps.

### 3.7 Scénarios de simulation :

Pour bien analyser notre réseau, nous avons choisi de mettre en œuvre différents scénarios de simulation avec différents protocoles de routage. Dans ce cadre nous allons faire une comparaison entre les protocoles de routage suivants : AODV, DSDV et DSR. Pour chaque scénario de simulation on augmente les nombres des nœuds de 08 à 15 nœuds.

#### 3.7.1. Les Scénarios

Dans ce scénario nous avons retenu un réseau, et un type de trafic CBR qui envoie des paquets de taille 512 octets avec 1Mbps pour voir les performances des protocoles de routage (AODV, DSDV et DSR) dans le cas de la mobilité des nœuds, le tableau suivant résume les différents paramètres utilisés dans ce scénario :

Paramètres	Valeurs
nombre nœuds	8 , 15
La mobilité des nœuds	Mobile
La dimension de topographié	500*500 m
Les protocoles de routage	AODV, DSDV, DSR
DataRate	11Mb
MAC type	Mac/802 11
SIFS	10us
DIFS	50us
A slot time	20us
CWmin	31
CWmax	1023
APreambleLength	144 bit
APLCPHeaderLength	48 bits
Traffic	CBR
Taille du Paquet	512 octets
Bit rate	1Mbps
Temps de simulation	200s

Tableau 3.2 : les différents paramètres utilisés

Trois métriques sont utilisées afin d'évaluer notre simulation, que se sont les critères les plus pertinent et les plus utilisés pour évaluer les performances d'un QoS.

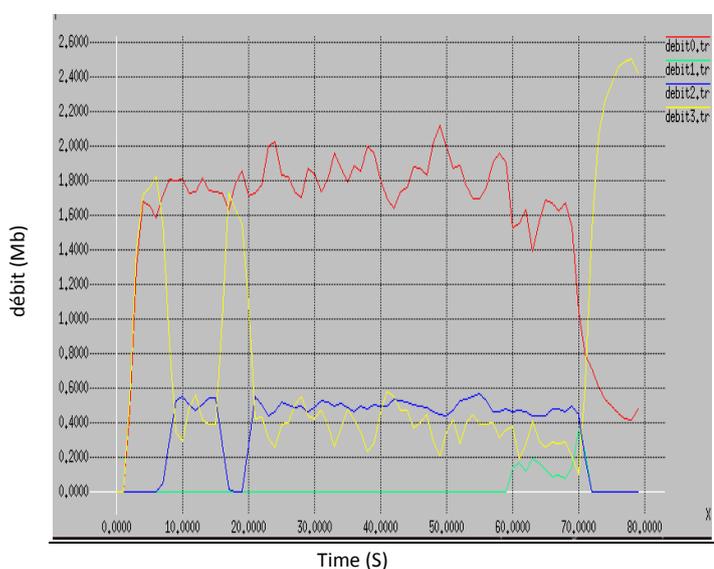
Dans une topologie mobile, la manière de trouver une route pour atteindre une destination est très importante. Dans cette situation, le nœud veut émettre des données, il cherche sa destination puisque la topologie est mobile il ne va pas la trouver facilement, donc chaque protocole de routage va utiliser son mécanisme pour essayer de trouver la destination dans un temps limité pour assurer une bonne performance de réseaux

### 3.7.2 Interprétation des résultats :

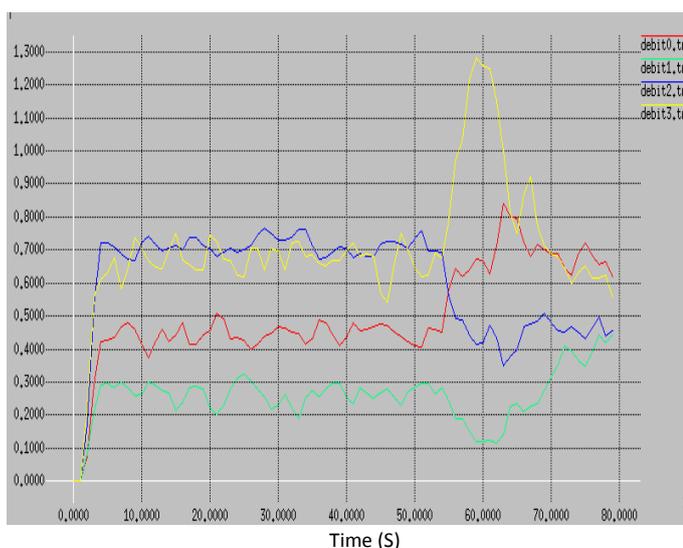
Dans cette étape, nous allons essayer d'interpréter tous les résultats des simulations effectuées dans le cadre de cette étude. Ces résultats sont présentés sous formes de courbes calculées en fonctions des trois métriques telles que **Débit d'un nœud**, **Taux de livraison**, **Délai**. Ces métriques sont évaluées en fonction des nombres de nœud, de connexion ainsi qu'à la variation de la vitesse de mobilité. Dans la suite nous allons présenter les résultats des différents scénarios successivement.

#### Le débit :

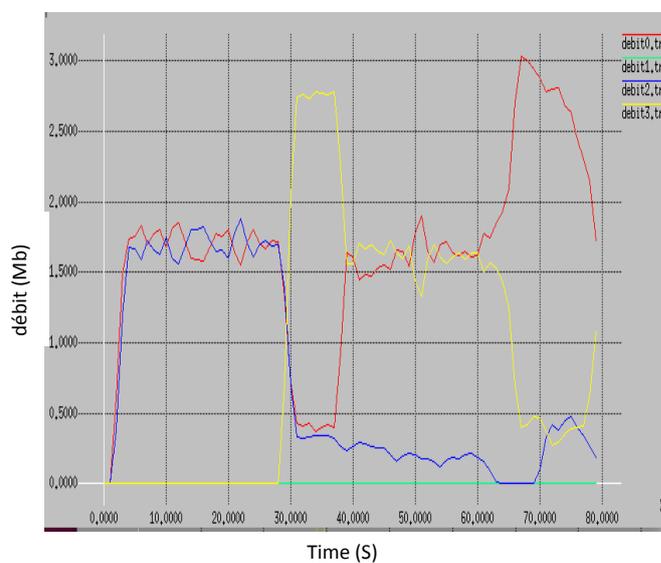
Evolution de débit en fonction du temps pour différents protocoles pour 08 nœuds mobile .



Graph 3.4 Le débit - Protocole AODV



Graph 3.6 Le débit -Protocole DSR

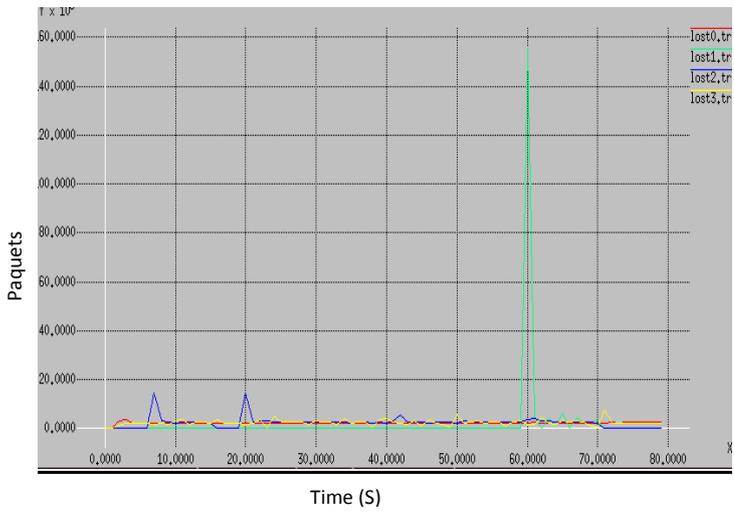


Graph 3.5 Le débit -Protocole DSDV

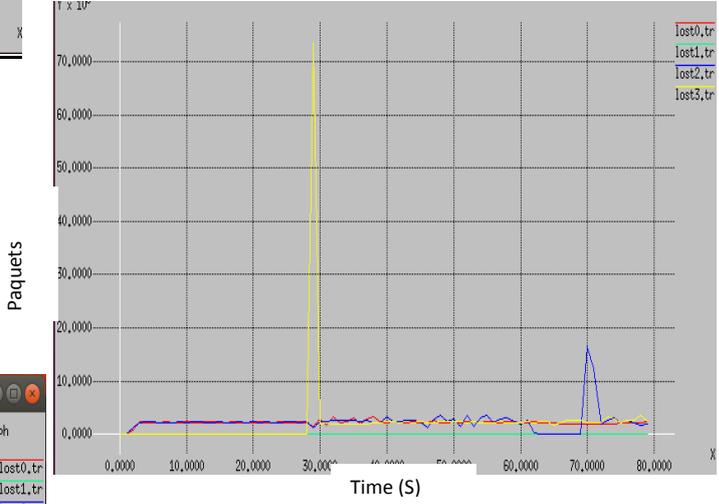
Ces figures montrent que le DSR donne un bon débit par rapport aux autres protocoles. DSR sélectionne le meilleur chemin vers la destination avec un petit nombre de sauts et a un acheminement minimal car il est de nature réactive et ne construit pas de table de routage, on peut en déduire que DSR génère moins de paquets de contrôle du fait qu'il soit réactif (qui établit une route à la demande), il se comporte de façon stable par rapport aux deux autres même avec un maximum de nombre de nœud. Nous pouvons quasiment dire qu'il est insensible à la variation des nœuds.

**Les pertes :**

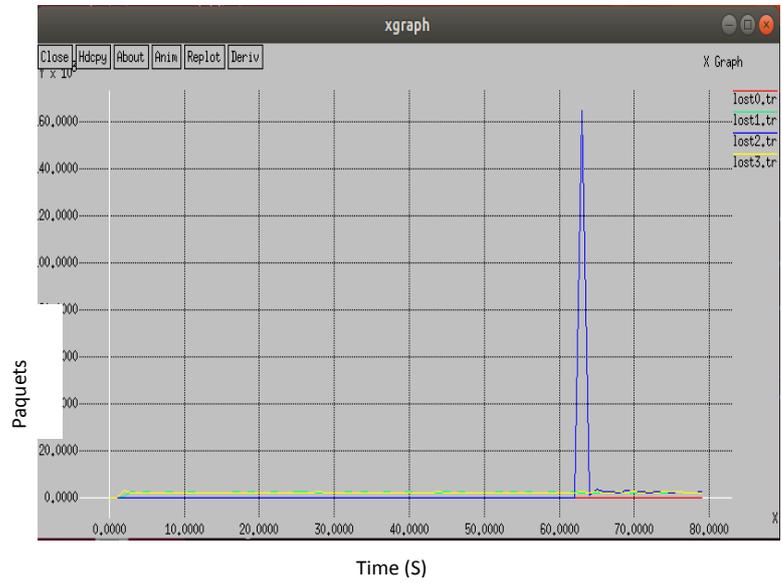
Evolution des pertes en fonction du temps .



Graph 3.7 Le perte-Protocole AODV



Graph 3.8 Le perte-Protocole DSDV

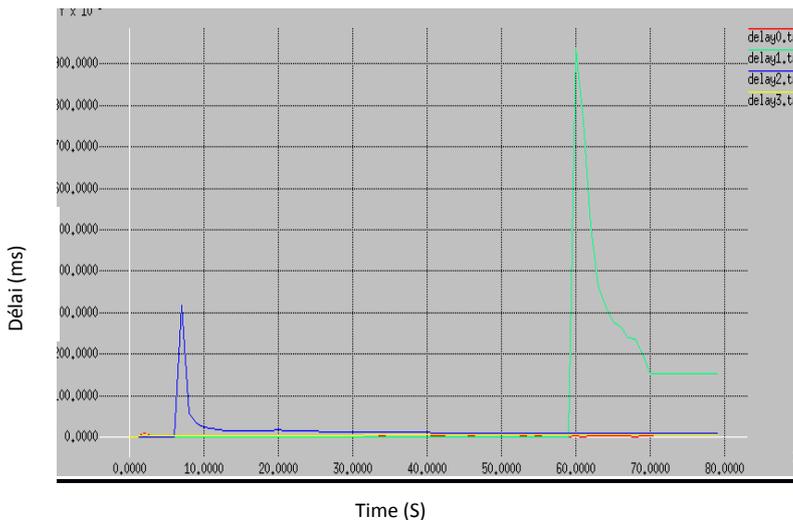


Graph 3.9 : Le perte- Protocole DSR

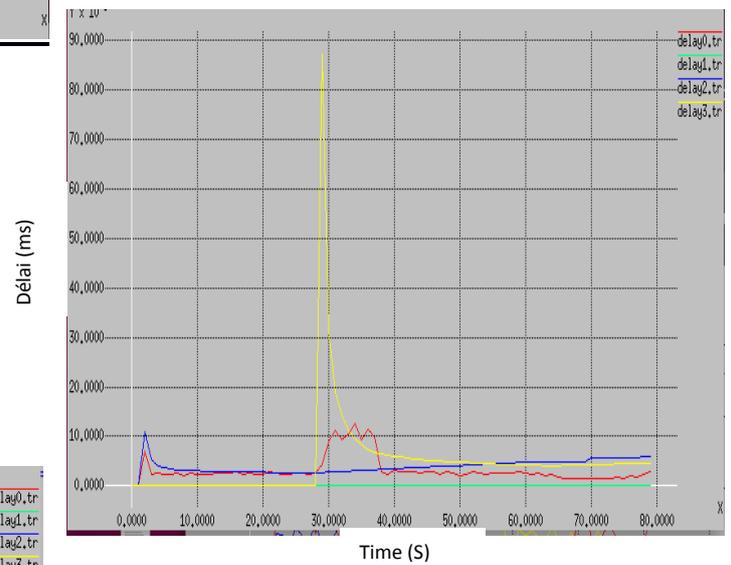
La différence est bien visible sur les deux courbes entre les trois protocoles. Sur la première figure AODV . DSDV assure moyennement la préservation des paquets. DSR a une perte presque négligeable. Ces deux derniers ont un comportement presque identique malgré l'augmentation de la mobilité. Par contre, AODV perd de plus en plus de paquets avec une forte mobilité à cause des chemins qui ne restent plus valides.

**Le délai :**

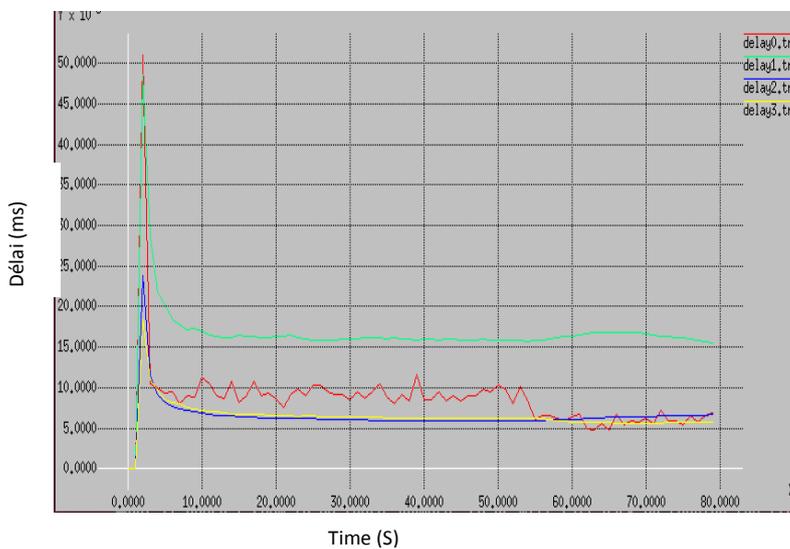
Evolution de délai en fonction du temps pour différents protocoles .



Graphe 3.10 Le délai- Protocole AODV



Graphe 3.11 Le délai- Protocole DSDV



Graphe 3.12 Le délai-Protocole DSDV

D'après les figures, il est bien clair que le délai le plus faible et celui de protocole DSR, car il choisit les plus courts chemins. L'opération de découverte de routes de protocole DSR est bien réussie, pour les deux autres protocoles DSDV et AODV prends un temps de mise à jour des tables de routage avant de découvrir la route, et après cette période tous les protocoles donnent un délai converge.

### **3.8 Conclusion :**

Dans ce chapitre, Les performances du protocole de routage dans les réseaux de capteur sans fil: DSR (Dynamic Source Routing) et AODV (topologique a vecteur de distance) et DSDV (Destination Sequenced Distance-Vector Routing) à l'aide du SIMULATEUR NS-2 (VERSION 2.35) ont été analysé. Chaque protocole a sa propre nature et donne des performances variables. Nous avons constaté que le protocole DSR offre de meilleurs résultats. Les résultats présentés dans notre étude sont conformes à la réalité vu la simulation que nous avons réalisée. De ce fait, on peut conclure que parmi les protocoles étudiés, DSR répond le mieux au besoin des réseaux de capteur sans fil.

## **Conclusion générale**

Les réseaux de capteurs sans fil sont une nouvelle technologie qui a surgi après les grands progrès technologiques concernant le développement des capteurs, des processeurs puissants et des protocoles de communication sans fil. Ce type de réseau composé de certains ou de milliers d'éléments a pour but la collecte de données de l'environnement, leurs traitements et leurs disséminations vers le monde extérieur. Les applications des réseaux de capteurs sont nombreuses. Elle comprend différents domaines : médicale, agricole, militaire, etc.

Tout au long de ce mémoire, nous avons tout d'abord présenté le problème de routage dans les réseaux sans fil tout en distinguant le protocole de routage le plus performant. Dans un premier temps nous avons présenté les réseaux sans fil en générale en donnant leurs caractéristiques. Puis nous avons exposé la définition de la qualité de service d'un réseau sans fil. Les performances et la qualité de service sont les principales caractéristiques sur lesquelles les travaux de recherche ont proposé d'améliorer les performances des réseaux sans fils avec les protocoles de routage.

Dans ce projet nous nous sommes intéressés à la simulation des protocoles AODV, DSDV et DSR avec le simulateur NS-2, et ceci dans le but d'analyser leurs performances selon le : débit, délais, taux de livraison des paquets.

Les trois protocoles ont été étudiés, simulés et comparés dans le scénario qui a donné une vision assez claire sur les contraintes. En effet, les comparaisons entre la simulation prouvent que DSR offre un meilleur délai de bout en bout et livraison du paquet. Pour le débit l'AODV est meilleur. On constate que globalement le DSR est le meilleur protocole convenable pour les réseaux de capteurs afin de garantir la rapidité et la transmission sûre.

## **Bibliographie**

- [01] Y. Youbi ; Y.Istrig , **Le problème d'exclusion mutuelle dans les réseaux mobiles ad hoc** ,  
Memoire de master, UniversiteA.Mira de Bejaia, Algerie [ juin 2010 ].
- [02] Bendimerad Nawel , **Système de surveillance d'infrastructures publiques à l'aide des réseaux de capteurs vidéo sans fil** , These de doctorat, [année 2015].
- [03] Mehdi Boullegue , **Protocoles de communication et optimisation de l'énergie dans les réseaux de capteurs sans fil** , These de doctorat , Université Bretagne Loire [année 2016].
- [04] Abdallah Makhoul, Rachid Abdallah, Cong-Duc Pham , **Surveillance vidéo sur réseaux de capteurs sans-fils : ordonnancement adaptatif avec prise en compte de la criticité** , Memoire de master , <https://hal.inria.fr/inria-00419473> , [année 2009] .
- [05] Hamel Rebhalmachi , Khalfi Asma , **QoS, routage dans les réseaux ad hoc >>** , **Mémoire de Master** , Université Akli Mohand Oulhadj – Bouira [année 2017] .
- [06] Mohamed Lamine Lamali , **Qualité de Service et calcul de chemins dans les réseaux inter-domaine et multicouches** , These de doctorat , université de Versailles Saint-Quentin, [année 2014].
- [07] Younes , Nadine , La qualité de service des services multimédia sur les réseaux AD HOC sans fil à multi-sauts , Memoire de master, École de technologie supérieure université du québec , [année 2009]
- [08] : <http://www.commentcamarche.net/contents/wireless/wlintro.php3> , consultation avril 2020 .
- [09] : <http://www.urec.cnrs.fr/IMG/pdf/cours.sf.pdf>
- [11] : <http://www.commentcamarche.net/contents/wireless/wpan.php3> [ année 2008 ]
- [12] Belabdelli Abdelheq, Oukaz Mokhtar , **Dimensionnement D'un Réseau Sans Fil Wifi** , Université aboubekrbelkaid ,Mémoire de :Master. [ année 2012 ].
- [13] Mohamed Lamine Lamali , **Qualité de service et calcul de chemins dans les réseaux inter-domaine et multicouches** , Thèse doctorat . Thèse doctorat. Université de Versailles Saint Quentin [ année 2015 ].

- [14] Antoine Mahul . **Apprentissage de la qualité de service dans les réseaux multiservices: applications au routage optimal sous contraintes** , Thèse de doctorat.I. Université Blaise pascal - clermont-ferrand II. [ année 2012 ].
- [15] Said Hoceini . **Techniques d'Apprentissage par Renforcement pour le Routage Adaptatif dans les Réseaux de Télécommunication à Trafic Irrégulier** , Thèse de doctorat , Université de PARIS XII – VAL DE MARNE [ année 2005 ].
- [16] Bouatia Wassila ,Meziane TaniFadia Selma. **étude et évaluation des performances du réseau wifi mesh (802.11s)** . Mémoire de Master , Université *aboubakrbelkaid*. [ année 2013 ].
- [17] EmnaTrigui. **Gestion multi-agents du spectre pour des terminaux mobiles à radio cognitive** , Thèse doctorat. Université de technologie Troyes [ année 2013 ].
- [18] Kyeongja Lee. **Global QoS model in the ISP networks: DiffServ aware MPLS Traffic Engineering** , Thèse doctorat . Université des sciences et technologies de lille [ année 2006 ].
- [19] Ismail Adeldjama. **Adaptations inter-couches pour la diffusion des services vidéo sans fil Cross-Layer Adaptations for wireless video streaming services** , Thèse doctorat. université bordeaux [ année 2008 ].
- [20] David Gauchard . **Simulation hybride des réseaux IP-DiffServ-MPLS multi-services sur environnement d'exécution distribuée** , Thèse doctorat. Université Toulouse III [ année 2005].
- [21] Guerrou el-Hachemi . **Optimisation des Protocoles de Routage avec la méthode de Colonie de Fourmis** . Thèse doctorat . Institut National de formation en Informatique [ année 2007 ].
- [22] Boudour Ouissem-eddine Amir. **Conception et implémentation d'un outil de simulation de la QoS dans un réseau IP** , Mémoire Master , Université mohamed Boudiaf [ année 2008].
- [23] Mekki nabil , Mohammedi kada ,**Techniques de conservation d'énergie pour les réseaux de capteur sans fil** , Mémoire Master. UNIVERSITE Dr. TAHAR MOULAY SAIDA [ juin 2018 ]
- [24] Direm Khaled & BERTAL Ayyoub **Évaluation de performance et qualité de service des réseaux sans fil WLAN** , Mémoire de Master , Université 8Mai 1945 – Guelma [ année 2019].

- [25] Emna Trigui . **Gestion multi-agents du spectre pour des terminaux mobiles à radio cognitive** , Thèse doctorat , Université de technologie Troyes [ année 2013].
- [26] Tani Fadia Selma. **Etude et evaluation des performances du réseau wifi mesh (802.11s)**. Mémoire de Master . Université abou bakr belkaid , [ année 2013].
- [27] Bessaih Aldja, Bouchakel Siham. **Routage et simulation dans les réseaux mobiles ad hoc** , Mémoire de Master , Université de bejaia , [ année 2017 ]
- [28] Kyeongja Lee. **Global QoS model in the ISP networks: DiffServ aware MPLS Traffic Engineering** , Thèse doctorat , Université des sciences et technologies de lille [ année 2006]
- [29] Boudour Ouissem-eddine Amir. **Conception et implémentation d'un outil de simulation de la QoS dans un réseau IP** , Mémoire de Master , Université mohamed Boudiaf [ année 2008 ]
- [30] Ismail Adel Djama. **Adaptations inter-couches pour la diffusion des services vidéo sans fil Cross-Layer Adaptations for wireless video streaming services** , Thèse doctorat , université bordeaux [ année 2008 ]
- [31] David Gauchard . **Simulation hybride des réseaux IP-DiffServ-MPLS multi-services sur environnement d'exécution distribuée** , Thèse doctorat , Université Toulouse III [ année 2008 ]
- [32] Guerrou el-Hachemi . **Optimisation des Protocoles de Routage avec la méthode de Colonie de Fourmis** . Mémoire de doctorat , Institut National de formation en Informatique [ année 2007 ]
- [33] Cobo Campo, Luis. **Gestion de la qualité de service et planification optimale de réseaux de capteurs multimédia sans fil** , Mémoire doctorat , Université de montréal [ année 2011 ]
- [34] Khalid El Gholami. **La gestion de la qualité de service temps-réel dans les réseaux de capteurs sans fil** , Mémoire doctorat , UNIVERSITÉ BLAISE PASCAL - CLERMONT II [ année 2015 ]
- [35] KHawla Smida , **Les performances des protocoles de routage dans les réseaux de capteurs sans fil** , Mémoire doctorat , UNIVERSITÉ BADJI MOKHTAR ANNABA [ année 2018 ]
- [36] Tahar Chaouch Amel Benmoumene Abdellah Mahfoudh. **Analyse par simulation des performances d'un réseau de capteurs sans fil (WSN)** , Mémoire de Master , Université Djilali Bounaama Khemis Miliana . [ année 2016 ] .

