



REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université Hadj Lakhder - Batna

Faculté des Sciences de l'Ingénieur

Département d'Informatique



Mémoire de Magistère

Thème :

Traitement et Transfert d'images Par Réseau de Capteurs sans Fil

Préparé par : **Adel CHOUHA**

Proposé et dirigé par : **Dr. Azeddine BILAMI**

Pour l'obtention du **Magistère en Informatique**

Option : Ingénierie des **Systèmes d'Informatiques**

Soutenue publiquement le : 16/ 03/2011 devant le jury composé de :

Pr. Mohammed BENMOHAMMED	Professeur	Président	Université de Constantine
Dr. Azeddine BILAMI	M.C.	Rapporteur	Université de Batna
Dr. Abdelmadjid ZIDANI	M.C.	Examineur	Université de Batna
Dr. Allaoua CHAOUI	M.C.	Examineur	Université de Constantine

Année Universitaire : 2010 - 2011

Traitement et Transfert d'images Par Réseau de Capteurs sans Fil

Remerciements

Je tiens à remercier dieu le tout puissant qui m'a donné le courage et la patience qui a éclairé mon chemin pour achever ce travail.

Je remercie très vivement mon directeur de thèse Dr. Bilami Azeddine pour m'avoir suivi et soutenu dans ce parcours, pour m'avoir conseillé et encouragé et pour m'avoir fait partager son expérience et ses connaissances.

Je remercie chaleureusement les membres du jury, Dr. Benmohammed Mohamed en qualité de président du jury, Dr. Zidani Abdelmadjid et Dr. Chaoui Allaoua en qualité d'examineurs.

Enfin, je remercie mes parents et toute ma famille, mes amis pour leur soutien et leurs encouragements.

Sommaire

REMERCIEMENTS	2
SOMMAIRE	3
LISTE DES FIGURES	8
RESUME	9
INTRODUCTION GENERALE	11
CHAPITRE 1 : GENERALITES SUR LES RESEAUX DE CAPTEURS SANS FIL	13
1. INTRODUCTION	14
2. DEFINITION	14
3. ARCHITECTURE D'UN NŒUD CAPTEUR	15
3.1. L'UNITE D'ACQUISITION DES DONNEES	15
3.2. L'UNITE DE TRAITEMENT DES DONNEES	16
3.3. L'UNITE DE TRANSMISSION DE DONNEES.....	16
3.4. L'UNITE D'ENERGIE	16
4. ARCHITECTURE D'UN RESEAU DE CAPTEURS SANS FIL	17
5. PILE PROTOCOLAIRE	18
6. CONTRAINTES DE CONCEPTION DES RCSF	19
6.1. DUREE DE VIE DU RESEAU	20
6.2. RESSOURCES LIMITEES	20
6.3. BANDE PASSANTE LIMITEE	20
6.4. FACTEUR D'ECHELLE	20
6.5. TOPOLOGIE DYNAMIQUE	20
6.6. AGREGATION DE DONNEE	21
7. DOMAINES D'APPLICATIONS DES RESEAUX DE CAPTEURS	21
7.1. APPLICATIONS MILITAIRES.....	21
7.2. APPLICATIONS A LA SECURITE	22
7.3. APPLICATIONS ENVIRONNEMENTALES	22
7.4. APPLICATIONS MEDICALES	23
7.5. LA DOMOTIQUE.....	23
7.6. APPLICATIONS COMMERCIALES	23
8. CLASSIFICATION DES PROTOCOLES DE ROUTAGE DES RCSFS	24

8.1.1. LE PROTOCOLE FLOODING	22
8.1.2. LE PROTOCOLE DE « DIRECTED DIFFUSION »	23
8.1.3. SPIN (SENSOR PROTOCOLS FOR INFORMATION VIA NEGOTIATION)	24
8.1.4. ROUTAGE PAR RUMEUR.....	25
8.2. PROTOCOLES DE ROUTAGE HIERARCHIQUE.....	26
8.2.1. LEACH (LOW ENERGY ADAPTIVE CLUSTERING HIERARCHY)	27
8.2.2. PEGASIS (POWER-EFFICIENT GATHERING IN SENSOR INFORMATION SYSTEMS).....	28
8.3. PROTOCOLE BASE SUR LA LOCALISATION.....	29
8.3.1. GEAR.....	30
9. CONSOMMATION D'ENERGIE DANS LES RCSF	32
9.1. MODELE DE CONSOMMATION D'ENERGIE	32
10. MECANISMES DE CONSERVATION DE L'ENERGIE DANS LES RCSF	33
10.1. AU NIVEAU DES DIFFERENTES COUCHES.....	33
10.2. CONSERVATION DE L'ENERGIE SOUS CONTRAINTE DE COUVERTURE	36
10.3. CONSERVATION DE L'ENERGIE PAR LA FORMATION DE GRAPPES (CLUSTERING).....	36
10.4. PAR L'AJUSTEMENT OPTIMISE DES PUISSANCES DE TRANSMISSION	37
10.5. PAR LA PLANIFICATION OPTIMISEE DES ETATS DES CAPTEURS	38
11. CONCLUSION	39
CHAPITRE 2 : LES RESEAUX DE CAPTEURS SANS FIL DE VISION	40
1. INTRODUCTION	41
2. APPLICATIONS	41
2.1 APPLICATIONS MILITAIRES.....	42
2.2 VIGILANCE ENVIRONNEMENTALE	42
2.3 SURETE ET SECURITE DE ZONES SENSIBLES.....	43
2.4 SUIVI DU TRAFIC ROUTIER	43
2.5 APPLICATIONS A LA ROBOTIQUE	43
3. CLASSIFICATION DES RCSF DE VISION.....	44
3.1 RESEAUX DE CAPTEURS D'IMAGES FIXES	44
3.2 RESEAUX DE CAPTEURS DE VIDEO.....	44
4. CONTRAINTES DES RESEAUX DE CAPTEURS DE VISION.....	44
4.1 CAPTURE DU SIGNAL.....	45

4.2 BESOINS DE MEMOIRE	45
4.3 TRAITEMENT DU SIGNAL.....	45
4.4 TRANSMISSION DE DONNEES	46
5. CAPTEUR D'IMAGE.....	46
5.1 CAMERAS BASEES SUR DES COMPOSANTS COMMERCIAUX	47
5.2 CAMERAS CONÇUES POUR LES RESEaux DE CAPTEURS SANS FIL.....	49
6. CONCLUSION	51
CHAPITRE 3 : COMPRESSION D'IMAGES DANS LES RESEaux DE CAPTEURS SANS FIL	52
1. INTRODUCTION	53
2. CARACTERISTIQUES D'UNE IMAGE.....	53
2.1. GENERALITES.....	53
3. CODAGES ET COMPRESSIONS D'UNE IMAGE.....	57
3.1 LE FORMAT RLE	57
3.2 LA COMPRESSION RL.....	57
3.3 LE CODAGE DE HUFFMAN	58
3.4 LA COMPRESSION LZW	59
3.5 LE MODE DE COMPRESSION JPEG.....	59
3.6 LES METHODES DE COMPRESSION RECENTES.....	61
3.6.1 LA COMPRESSION FRACTALE	61
3.6.2 LA COMPRESSION PAR ONDELETTES.....	62
4. TRAITEMENT D'IMAGES DANS LES RESEaux DE CAPTEURS SANS FIL	63
4.1. CLASSIFICATION DES ALGORITHMES DE COMPRESSION D'IMAGE	64
4.1.1 COMPRESSION LOCALE	64
4.1.2 COMPRESSION DISTRIBUEE.....	66
5. CONCLUSION	70
CHAPITRE 4 : APPROCHE PROPOSEE POUR LA COMPRESSION D'IMAGES.....	71
1. INTRODUCTION	72
2. TRANSFORMEE EN ONDELETTES D'UNE IMAGE	72
3. SOLUTION PROPOSEE.....	74
4. PRESENTATION DU SIMULATEUR NS2.....	76
5. MODELISATION DE LA TRANSFORMEE EN ONDELETTES DYADIQUE	77

6. SIMULATION ET PERFORMANCE	77
6.1. ENVIRONNEMENT DE SIMULATION	77
6.2. RESULTATS DE SIMULATION.....	80
7. CONCLUSION	81
CONCLUSION GENERALE.....	84
REFERENCES	85

Liste des Figures

FIGURE 1 : LES COMPOSANTS D'UN CAPTEUR	15
FIGURE 2 : CAPTEURS FABRIQUES PAR UC BERKELEY	17
FIGURE 3 : ARCHITECTURE D'UN RESEAU DE CAPTEUR SANS FIL	18
FIGURE 4 : PILE PROTOCOLAIRE	19
FIGURE 5 : UN SERVICE MILITAIRE UTILISANT LES RCSF	22
FIGURE 6 : LE PROBLEME D'IMPLOSION <i>ET</i> D'OVERLAP	25
FIGURE 7 : LES PHASES DE PROTOCOLE DIRECTED DIFFUSION	26
FIGURE 8 : FONCTIONNEMENT DE PROTOCOLE SPIN	27
FIGURE 9 : TOPOLOGIE BASEE SUR LA LOCALISATION	31
FIGURE 10 : MODELE DE CONSOMMATION D'ENERGIE	34
FIGURE 11 : CYCLOPS ATTACHE A UN MOTE MICA2	48
FIGURE 12 : ALOHA ATTACHE A UN MOTE MICA2	50
FIGURE 13 : REPRESENTATION D'UNE SINUSOÏDE ET D'UNE ONDELETTE	62
FIGURE 14 : APPROCHE TRADITIONNELLE DE DIVISION DES DONNEES	67
FIGURE 15 : TRANSFORMATION EN ONDELETTE	67
FIGURE 16 : METHODE 1 COMPRESSION D'IMAGE DISTRIBUEE BASEE SUR LA TRANSFORMATION EN ONDELETTE	68
FIGURE 17 : METHODE 2 COMPRESSION D'IMAGE DISTRIBUEE BASEE SUR LA TRANSFORMATION EN ONDELETTE	69
FIGURE 18 : LA TO DYADIQUE APPLIQUEE UNE FOIS (A) ET DEUX FOIS (B)	73
FIGURE 19 : TRANSFORMATION EN ONDELETTE CENTRALISEE	74
FIGURE 20 : TRANSFORMATION EN ONDELETTE DISTRIBUEE	75
FIGURE 21 : TRANSMISSION D'IMAGE SANS TO	79
FIGURE 22 : TRANSMISSION D'IMAGE AVEC UNE TO	79
FIGURE 23 : TRANSMISSION D'IMAGE AVEC UNE TO DISTRIBUEE DEUX FOIS	79
FIGURE 24 : TRANSMISSION D'IMAGE AVEC UNE TO CENTRALISEE DEUX FOIS	80
FIGURE 25 : L'ENERGIE CONSOMMEE DANS LES DIFFERENTS SCENARIOS	81

Résumé

Les réseaux de capteurs sans fil sont aujourd'hui utilisés dans plusieurs domaines. L'image à une importance capitale dans plusieurs applications se basant sur les réseaux de capteurs sans fil, mais sa capture, son traitement et sa transmission posent des défis car ces opérations consomment beaucoup d'énergie à cause de la grande quantité de données que renferme la représentation d'une image.

Pour économiser l'énergie, on remédie dans ce cas à la compression pour réduire la quantité de données à transmettre. La contrainte de la limitation des ressources capteur, comme la capacité de traitement et de stockage rend impossible l'exécution des algorithmes de compression d'image standard. Nous proposons dans ce mémoire une solution qui consiste à distribuer la compression entre les nœuds pour minimiser l'énergie consommée par le nœud source et augmenter la durée de vie du réseau.

Abstract

Today, wireless sensor networks are used in several fields. The image has an importance in several applications of wireless sensor networks (WSN) but its transmission is gourmand in energy because of great quantity of data which contains an image. To save the power consumption it is necessary to reduce the quantity of data to transmit and this is done by compression.

The constraint of the limitation of the resources sensor, like the storage and processing capacity makes impossible the execution of the algorithms of compression standard of image. We proposed a solution which consists of distributed compression between the nodes to minimize the power consumption by the node source and to increase the lifetime of network.

Introduction Générale

Les récentes avancées dans les domaines des technologies de communication sans-fil et microélectroniques ont permis le développement à faible coût de minuscules capteurs (de quelque millimètre cube de volume) consommant peu d'énergie (solution low-cost et low-power). De ce fait, un nouveau domaine de recherche a été créé pour offrir des solutions économiquement intéressantes qui est le domaine des réseaux de capteurs sans fil ayant pour rôles de collecter des données d'un environnement et de les diffuser au sein du réseau. Ce type de réseaux de capteurs pourrait avoir de très diverses applications.

Un réseau de capteurs est composé d'un nombre souvent très important de nœuds, ces nœuds appelés en anglais **nodes**, sont alimentés par des piles et qui sont, soit posés à un endroit précis, soit dispersés aléatoirement (souvent déployés par voie aérienne à l'aide d'avions ou hélicoptères). Afin de résister aux déploiements, ces capteurs doivent être très solides et de plus, ils doivent aussi pouvoir survivre dans les conditions les plus extrêmes dictées par leur environnement d'utilisation (feu ou eau par exemple). En plus des contraintes environnementales, une contrainte très importante est l'économie de batterie. En effet, un réseau de capteurs ne peut survivre si la perte de nœuds est trop importante car ceci engendre des pertes de communication dues à une trop grande distance entre les capteurs. Donc il est très important que les batteries durent le plus longtemps possible étant donné que dans la plupart des applications ils sont placés aléatoirement (impossible de retourner changer les batteries). Les réseaux de capteurs peuvent être programmés à un grand nombre de fins, telles que le contrôle d'intrusions, le calcul de températures, le calcul de changements climatiques, la surveillance des déplacements d'animaux (avec récepteurs GPS), surveillance de malades,....

L'image a une importance dans plusieurs applications utilisées par des réseaux de capteur sans fil, y compris la surveillance et reconnaissance d'objets, localisation et pistage (tracking) d'objets. Cependant, la représentation des données visuelles exige une grande quantité d'information et ceci demande une énergie élevée pour les transmettre. Donc il faut compresser l'image avant la transmission pour réduire l'énergie consommée. Beaucoup de travaux qui sont publiés concentrent sur la transmission d'image avec un algorithme de compression centralisé. Mais la compression d'une image demande des transformations et des traitements au niveau de nœud source qui sont coûteux à cause de grande quantité d'information qui doit être traitée donc le coût de traitement est pris en considération

contrairement au traitement de la valeur d'une température que sa coût est très très petit qui n'est pas pris en considération.

Ce mémoire est structuré en quatre chapitres :

Dans le premier chapitre, nous avons présenté les différents concepts liés aux réseaux de capteurs sans fil et les différents mécanismes utilisés pour la conservation d'énergie.

Le deuxième chapitre est consacré à un type particulier de RCSF qui est le réseau de capteur sans fil de vision, en mettant la lumière sur ses caractéristiques et ses spécificités, puis nous présentons quelques capteurs d'image.

La compression d'image dans les réseaux de capteurs sans fil, ainsi que les travaux développés pour minimiser l'énergie lors de la transmission d'une image, sont illustrés dans le troisième chapitre.

Dans le quatrième chapitre, nous proposons une compression d'image adaptée au réseau de capteurs sans fil basée sur la transformation en ondelette discrète (TO), nous présentons d'abord le principe de fonctionnement de la TO ainsi la solution proposée et enfin simulation et résultats.

Chapitre 1 :

Généralité sur les réseaux de capteur sans fil

1. Introduction :

Un réseau de capteurs sans fil (plus connus sous le nom de **Wireless Sensor Network (WSN)** en anglais) est composé d'un ensemble de terminaux ou ce qu'on appelle des nœuds capteurs qui peuvent communiquer via des liaisons radio, sans infrastructure fixe préalable. Le réseau devra fonctionner de façon autonome, sans intervention humaine.

Les nœuds sont généralement matériellement petits, construits à partir des composants pas chers. Ce type de réseau est composé de centaines ou de milliers d'éléments (capteurs), a pour but la collecte de données de l'environnement, leur traitement et leur transmission vers le monde extérieur.

Les capteurs sont utilisés dans plusieurs domaines d'application: la surveillance environnementale, les opérations militaires et de sécurité, la médecine ... Les réseaux de capteurs doivent remonter une information détectée au niveau d'un capteur vers un nœud de collecte (un puits).

En effet, la transmission d'information entre les capteurs ne peut être assurée directement entre tout couple de capteurs, vu leur batterie limitée (qui entraîne une limite des puissances de transmissions). L'information est donc transmise de proche en proche. Les trois fonctions principales (mesure, traitement et communication de l'information) du capteur engendrent une consommation d'énergie. De ce fait, le capteur a une durée de vie limitée.

2. Définition :

Un réseau de capteurs sans fil (RCSF ou WSN : Wireless Sensor Network) est composé d'un grand nombre de nœuds distribués sur une zone donnée afin de mesurer une grandeur physique ou surveiller un événement et de réagir en cas de besoin en envoyant l'information collectée à un ou plusieurs points de collecte, à l'aide d'une connexion sans fil. Dans un tel réseau, chaque nœud est un dispositif électronique qui possède une capacité de calcul, de stockage, de communication et d'énergie [1].

3. Architecture d'un nœud capteur :

Un nœud capteur est composé de quatre composants de base [2] comme représentée dans la figure1: *une unité d'acquisition, une unité de traitement, une unité de communication*

et une source d'énergie. Ils peuvent également avoir d'autres composants dépendants de l'application tels qu'un système de localisation, un générateur d'énergie et un mobilisateur

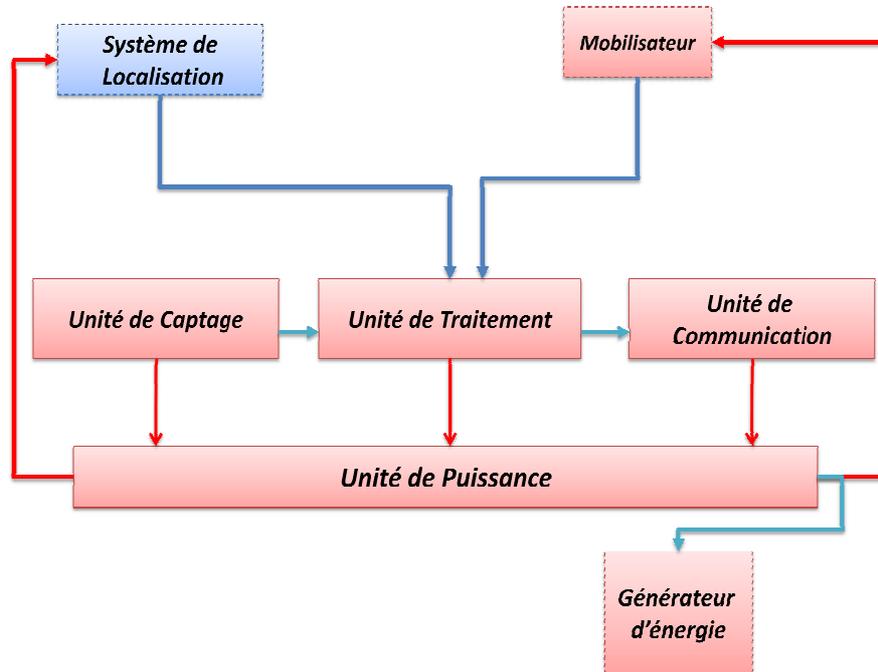


Figure1 : les composants d'un capteur

3.1. L'unité d'acquisition des données:

Elle se compose de deux sous unités, unité de captage et un convertisseur analogique – numérique (CAN). Le capteur permet de mesurer des informations environnementales : température, humidité, pression, accélération, sons, image, vidéo etc., puis produit des signaux analogiques qui sont convertis par un convertisseur analogique – numérique pour pouvoir être traitées par l'unité de traitement.

3.2. L'unité de traitement des données :

L'unité de traitement comprend un processeur avec une petite unité de stockage, une RAM pour les données et une ROM pour les programmes et souvent une mémoire flash. Cette unité fonctionne à l'aide d'un système d'exploitation spécialement conçu pour les micro-capteurs (TinyOS par exemple). Elle est chargée de gérer des procédures qui permettent à un nœud capteur de collaborer avec les autres nœuds du réseau. Elle peut aussi analyser les données captées pour alléger la tâche du nœud puits.

3.3. L'unité de transmission de données :

Cette unité est responsable d'effectuer toutes les émissions et réceptions des données sur un medium sans fil. Les composants utilisés pour réaliser la transmission sont des composants classiques, les unités de transmission de type radio-fréquence (RF) sont préférables pour les RCSF parce que les paquets transportés sont de petites tailles avec un bas débit. Ainsi on retrouve les mêmes problèmes que dans tous les réseaux sans fil : la quantité d'énergie nécessaire à la transmission augmente avec la distance. Pour les réseaux sans fil classiques (LAN, GSM) la consommation d'énergie est de l'ordre de plusieurs centaines de milliwatts alors que pour les réseaux de capteurs, le système de transmission possède une portée de quelques dizaines de mètres. Pour augmenter ces distances tout en préservant l'énergie, le réseau utilise un routage multi sauts.

3.4. L'unité d'énergie :

Les capteurs sont de petits composants alimentés avec une batterie ou avec des piles. Pour qu'un réseau de capteurs reste autonome pendant une durée de quelques mois à quelques années sans intervention humaine, la consommation d'énergie devient le problème fondamental. Celle-ci n'est pas un grand problème pour les réseaux sans fil traditionnel, car on peut toujours recharger les batteries des dispositifs sans fil comme les téléphones portables ou les ordinateurs portables. Mais, dans un RCSF, il est difficile (parfois impossible dans certaine applications) de changer la batterie. Cette unité peut aussi contenir des systèmes de rechargement d'énergie à partir de l'environnement observé telles que les cellules solaires, afin d'étendre la durée de vie totale du réseau.

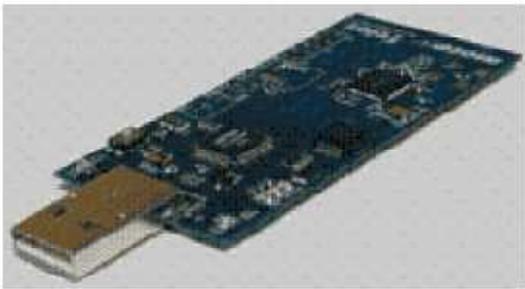
La figure2 montre des capteurs fabriqués par UC Berkeley.



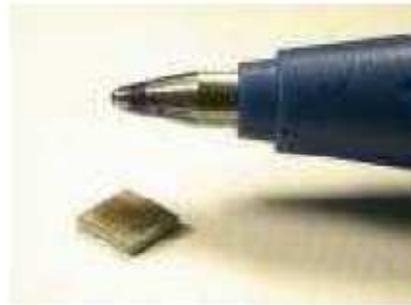
(a) WeC



(b) Mica family



(c) Telos



(d) Spec prototype

Figure 2 : Capteurs fabriqués par UC Berkeley

4. Architecture d'un réseau de capteurs sans fil :

Un réseau de capteurs sans fil est composé d'un grand nombre de nœuds. Chaque capteur est doté d'un module d'acquisition qui lui permet de mesurer des informations environnementales : température, humidité, pression, accélération, sons, image, vidéo etc.

Les données collectées par ces nœuds capteurs sont routées vers une ou plusieurs stations de base ou nœud puis (**sink** en anglais). Ce dernier est un point de collecte de données capturées. Il peut communiquer les données collectées à l'utilisateur final à travers un réseau de communication, éventuellement l'Internet ou un satellite. L'utilisateur peut à son tour utiliser la station de base comme passerelle, afin de transmettre ses requêtes au réseau (Voir la figure 3).

En général, un RCSF est composé de quatre éléments montrés par la figure : les nœuds capteurs, une station de base, phénomène à mesurer et l'utilisateur. [3], [10]

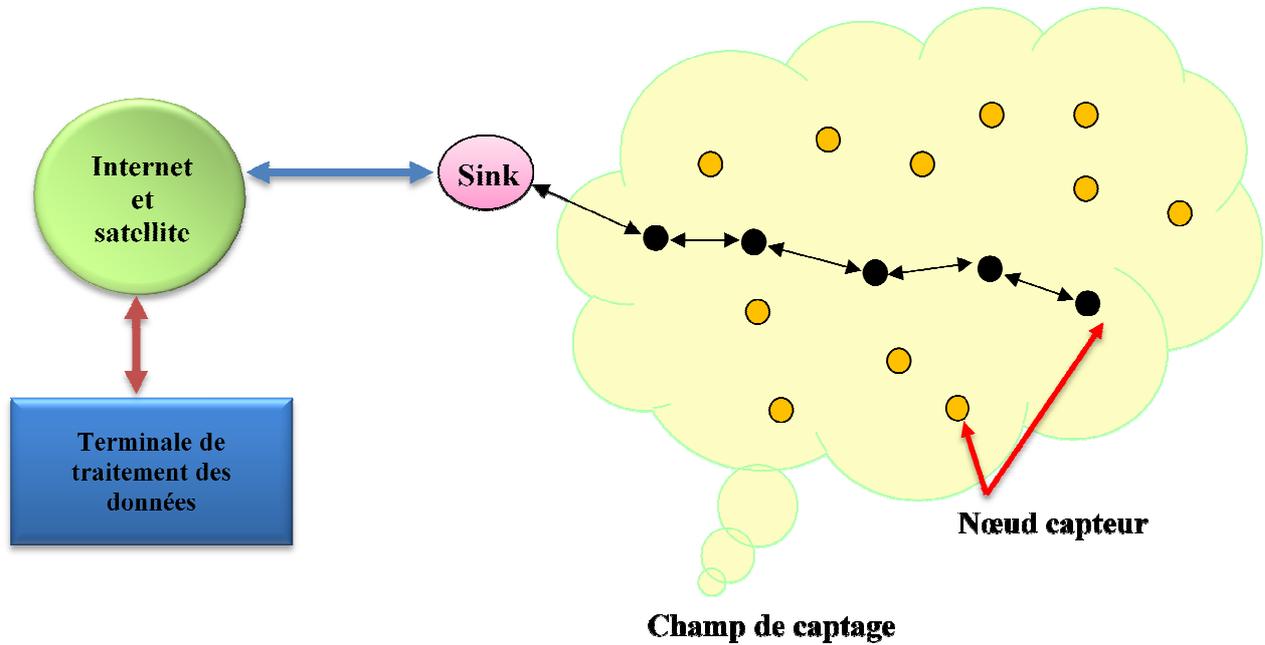


Figure 3 : Architecture d'un Réseau de Capteur Sans Fil

5. Pile protocolaire :

La pile protocolaire [1], [10] utilisée par la station de base ainsi que tous les autres capteurs du réseau est illustrée par la figure 4. La pile protocolaire comprend la couche application, la couche transport, la couche réseau, la couche liaison de données, la couche physique, le plan de gestion de l'énergie, le plan de gestion de la mobilité et le plan de gestion des tâches.

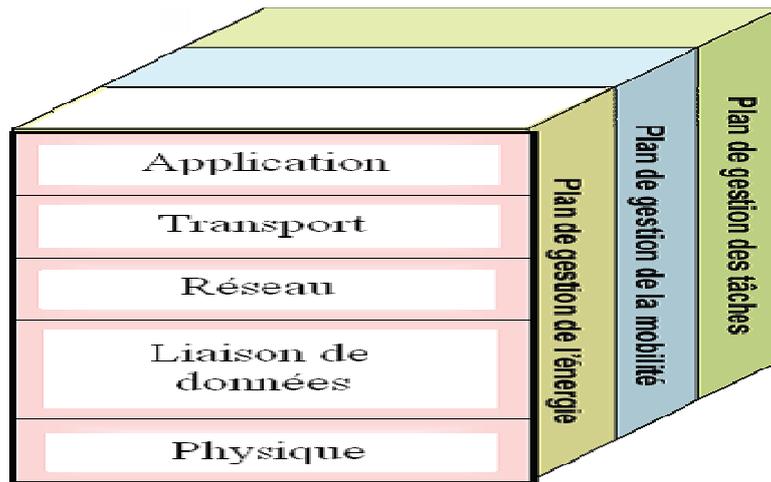


Figure 4 : Pile protocolaire.

Suivant la fonctionnalité des capteurs, différentes applications peuvent être utilisées et bâties sur la couche application. La couche transport aide à gérer le flux de données si le réseau de capteurs l'exige. Elle permet de diviser les données issues de la couche application en segments pour les délivrer, ainsi elle réordonne et rassemble les segments venus de la couche réseau avant de les envoyer à la couche application. La couche réseau prend soin de router les données fournies par la couche transport. Le protocole MAC (Media Access Control) de la couche liaison assure la gestion de l'accès au support physique. La couche physique assure la transmission et la réception des données au niveau bit.

En outre, les plans de gestion de l'énergie, de la mobilité et des tâches surveillent la puissance, le mouvement et la distribution des tâches, respectivement, entre les nœuds capteurs. Ces plans de gestion sont nécessaires, de sorte que les nœuds capteurs puissent fonctionner ensemble d'une manière efficace pour préserver l'énergie, router des données dans un réseau de capteurs mobile et partager les ressources entre les nœuds capteurs. Du point de vue global, il est plus efficace d'utiliser des nœuds capteurs pouvant collaborer entre eux. La durée de vie du réseau peut être ainsi prolongée.

6. Contraintes de conception des RCSF :

La conception et la réalisation des réseaux de capteurs sans fil sont influencées par plusieurs paramètres. Ces facteurs servent comme directives pour le développement des algorithmes et protocoles utilisés dans les RCSF.

6.1. Durée de vie du réseau

C'est l'intervalle de temps qui sépare l'instant de déploiement du réseau de l'instant où l'énergie du premier nœud s'épuise. Selon l'application, la durée de vie exigée pour un réseau peut varier entre quelques heures et plusieurs années.

6.2. Ressources limitées

En plus de l'énergie, les nœuds capteurs ont aussi une capacité de traitement et de mémoire limitée. En effet, les industriels veulent mettre en œuvre des capteurs simples, petits et peu coûteux.

6.3. Bande passante limitée

Afin de minimiser l'énergie consommée lors de transfert de données entre les nœuds, les capteurs opèrent à bas débit. Typiquement, le débit utilisé est de quelques dizaines de Kb/s. Un débit de transmission réduit n'est pas handicapant pour un réseau de capteurs où les fréquences de transmission ne sont pas importantes.

6.4. Facteur d'échelle

Le nombre de nœuds déployés pour une application peut atteindre des milliers. Dans ce cas, le réseau doit fonctionner avec des densités de capteurs très grandes. Un nombre aussi important de nœuds engendre beaucoup de transmissions inter nodales et nécessite que la station de base soit équipée de mémoire suffisante pour stocker les informations reçues [1], [10], [4].

6.5. Topologie dynamique

La topologie des réseaux de capteurs peut changer au cours du temps pour les raisons suivantes [1], [4]:

- Les nœuds capteurs peuvent être déployés dans des environnements hostiles (champ de bataille par exemple), la défaillance d'un nœud capteur est, donc très probable.
- Un nœud capteur peut devenir non opérationnel à cause de l'expiration de son énergie.
- Dans certaines applications, les nœuds capteurs et les stations de base sont mobiles.

6.6. Agrégation de donnée

Dans les réseaux de capteurs, les données produites par les nœuds capteurs voisins sont très corrélées spatialement et temporellement. Ceci peut engendrer la réception par la station de base d'informations redondantes. Réduire la quantité d'informations redondantes transmises par les capteurs permet de réduire la consommation d'énergie dans le réseau et ainsi d'améliorer sa durée de vie. L'une des techniques utilisée pour réduire la transmission d'informations redondantes est l'agrégation des données [5]. Avec cette technique, les nœuds intermédiaires agrègent l'information reçue de plusieurs sources. Cette technique est connue aussi sous le nom de fusion de données.

7. Domaines d'applications des réseaux de capteurs

La miniaturisation, l'adaptabilité, le faible coût et l'avancement dans les communications sans fil permettent aux réseaux de capteurs d'envahir plusieurs domaines d'applications. Ils permettent aussi d'étendre le domaine des applications existantes. Parmi ces domaines où ces réseaux se révèlent très utiles et peuvent offrir de meilleures contributions, on peut noter le militaire, la santé, l'environnemental, et les maisons intelligentes....

7.1. Applications militaires

Les premières applications potentielles des réseaux de capteurs ont concerné le domaine militaire. L'idée était de déployer un réseau de capteurs invisibles sur des champs de bataille ou des zones ennemies pour surveiller le mouvement des troupes. Historiquement, Le projet DSN (Distributed Sensor Network) [6] au DARPA (Defense Advanced Research Projects Agency) était l'un des premiers projets dans les années 80 ayant utilisés les réseaux de capteurs pour rassembler des données distribuées. Les applications militaires sont les premières et certainement les plus représentatives des applications trouvées actuellement dans le domaine des réseaux de capteurs sans fil. Les chercheurs du laboratoire national Lawrence Livermore ont mis en place le réseau WATS (Wide Area Tracking System) [7]. 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. Ces chercheurs ont mis en place ensuite un autre réseau appelé JBREWS (Joint Biological

Remote Early Warning System) [8] pour avertir les troupes dans le champ de bataille des attaques biologiques possibles.



Figure 5 : Un service militaire utilisant les RCSF

7.2. Applications à la sécurité

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 mouvement peut constituer un système d'alarme qui servira à détecter les intrusions dans une zone de surveillance.

7.3. Applications environnementales

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 aux changements climatiques par exemple le processus d'irrigation lors de la détection de zones sèches dans un champ agricole.

7.4. Applications médicales

Dans le domaine de la médecine, les réseaux de capteurs peuvent être utilisés pour assurer une surveillance permanente des organes vitaux de l'être humain grâce à des micro-capteurs qui pourront être avalés ou implantés sous la peau (surveillance de la glycémie, détection de cancers, ..). Ils peuvent aussi faciliter le diagnostic de quelques maladies en effectuant des mesures physiologiques telles que : la tension artérielle, battements du cœur, ... à l'aide des capteurs ayant chacun une tâche bien particulière. Les données physiologiques collectées par les capteurs peuvent être stockées pendant une longue durée pour le suivi d'un patient [9]. D'autre part, ces réseaux peuvent détecter des comportements anormaux (chute d'un lit, choc, cri, ...) chez les personnes dépendantes (handicapées ou âgées).

7.5. La domestique

Avec le développement technologique, les capteurs peuvent être embarqués dans des appareils, tels que les aspirateurs, les fours à micro-ondes, les réfrigérateurs,... . 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 se met en état d'arrêt quand la chambre est vide, la climatisation et le chauffage s'ajustent selon les points multiples de mesure, le déclenchement d'une alarme par le capteur anti-intrusion quand un intrus veut accéder à la maison.

7.6 Applications commerciales

Des nœuds capteurs pourraient améliorer le processus de stockage et de livraison. Le réseau ainsi formé, pourra être utilisé pour connaître la position, l'état et la direction d'un paquet ou d'une cargaison. Un client attendant un paquet peut alors avoir un avis de livraison en temps réel et connaître la position du paquet. Des entreprises manufacturières, via des

réseaux de capteurs pourraient suivre le procédé de production à partir des matières premières jusqu'au produit final livré. Grâce aux réseaux de capteurs, les entreprises pourraient offrir une meilleure qualité de service tout en réduisant leurs coûts. Les produits en fin de vie pourraient être mieux démontés et recyclés ou réutilisés si les micro-capteurs en garantissent le bon état. Dans les immeubles, le système de climatisation peut être conçu en intégrant plusieurs micro-capteurs dans les tuiles du plancher et les meubles. Ainsi, La climatisation pourra être déclenchée seulement aux endroits où il y a des personnes présentes et seulement si c'est nécessaire [1].

8. Classification des protocoles de routage des RCSFs :

Les protocoles de routage pour les RCSF sont divisés selon la structure de réseau en trois catégories, routage basé sur une structure plat (**flat-based routing**), routage hiérarchique (**hierarchical-based routing**) et routage géographique. Dans le routage basé sur une structure plat, tous les nœuds ont le même rôle. Dans le routage hiérarchique, les nœuds jouent des différents rôles. Dans le routage géographique la position de nœud est nécessaire pour router les données dans un réseau. Les protocoles de routage peuvent être classés selon les critères de nœud capteur (vitesse de processeur, capacité de mémoire, énergie limitée) en routage multi chemin (**multipath**), basé sur les requêtes (**query-based**), basé sur la négociation (**negotiation-based**), basé sur la QoS (**QoS-based**) ou basé sur la cohérence (**coherent-based**). On peut aussi classer les protocoles de routages en trois catégories, nommés, proactifs, réactifs et hybrides. Dans les protocoles proactifs les chemins sont calculés à l'avance, alors que dans les protocoles réactifs les chemins sont calculés à la demande. Les protocoles hybrides emploient une combinaison de ces deux idées [11]. Dans le reste de cette section, nous présentons un aperçu des principaux protocoles de routage dans les réseaux de capteurs.

8.1. *Protocoles De Routage Plat (Flat Routing):*

Dans les réseaux plats, chaque nœud joue généralement le même rôle. En raison de grand nombre de ces nœuds, il n'est pas possible d'attribuer un identifiant global à chaque nœud. Cette considération a conduit à un routage centré, où la station de base envoie des requêtes à certaines régions et attend les réponses par les capteurs situés dans les régions sélectionnées. Puisque les données sont demandées par le biais des requêtes, les attributs basés sur les noms sont nécessaires pour préciser les propriétés des données. Des travaux

dans le routage centré, par exemple, SPIN et diffusion directe [12] ont montré l'économie d'énergie à travers les données de négociation et l'élimination des données redondant. Ces deux protocoles ont motivé la conception de nombreux autres protocoles qui suivent un concept similaire.

8.1.1. Le protocole flooding

Dans le protocole Flooding [13] (appelé aussi l'inondation), chaque nœud reçoit les messages (sous forme d'un paquet de donnée), ensuite il le diffuse dans le réseau. Ainsi ce protocole consiste à transmettre tous les nouveaux paquets reçus et qui ne lui sont pas destiné. Ce protocole n'a nul besoin ni de maintenir une table de routage, ni de découvrir son voisinage et maintenir une topologie bien précise. Par contre, ce protocole présente deux inconvénients majeurs qui sont le problème de duplication des paquets (le problème d'implosion) et le problème d'overlap. En effet, le problème d'implosion est illustré par la figure 6.a, les deux nœuds B et C reçoivent le même paquet du nœud A, ensuite ces mêmes nœuds vont diffuser le même paquet au nœud D, d'où ce derniers reçoit deux copies de même paquet. Ainsi on ne peut plus distinguer entre les paquets récents et les paquets vieux.

D'autre part, le problème d'overlap est illustré par la figure 6.b, il se produit lorsque deux nœuds observent la même région puis ils diffusent la même information vers d'autres nœuds. La faiblesse de ce protocole est qu'il est « aveugle » en termes de consommation d'énergie. En effet ce protocole autorise une forte circulation de données et une grande consommation en terme énergie, ce qui engendre une diminution importante de sa durée de vie.

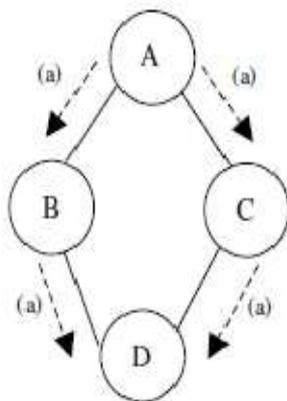


Figure 6.a : le problème d'implosion

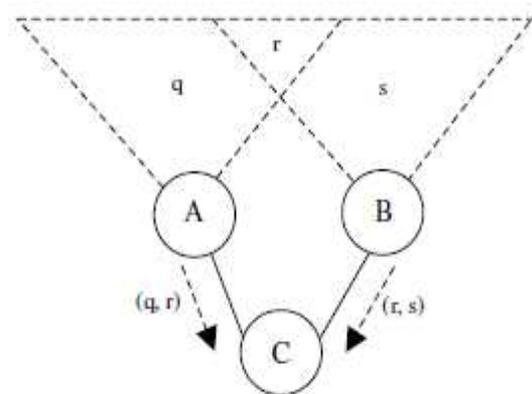


Figure 6.b : le problème d'overlap

8.1.2. Le protocole de « Directed Diffusion »

Dans [14], a été proposé un protocole de routage pour les réseaux de capteurs, appelé Directed Diffusion. Ce protocole est composé principalement de deux phases.

La première consiste à la phase de diffusion des messages d'intérêt. En effet, le Sink demande un service en envoyant les intérêts à tout le réseau (voir figure7 (a)). L'intérêt représente une tâche à accomplir par le réseau et il peut être destiné pour un ou plusieurs nœuds (par exemple dans la figure7 les trois nœuds à gauche).

La deuxième phase présente la réaction d'un nœud suite à la réception d'un intérêt. D'abord, le nœud vérifie s'il est concerné par ce message, ensuite si oui il enregistre l'identité du nœud émetteur de l'intérêt en question (figure7 (b)), afin de construire le gradient des routes menant vers le Sink. Si le nœud n'est pas destiné par l'intérêt, il continue à le propager à tous ces voisins. Une fois le message arrivé au destinataire, la route vers la station de base est alors établie et ainsi le nœud cible choisit cette route pour envoyer les informations demandées (figure7 (c)). La tâche demandée par le message d'intérêt peut être générer périodique, par exemple du genre «envoi moi la température chaque heure ».

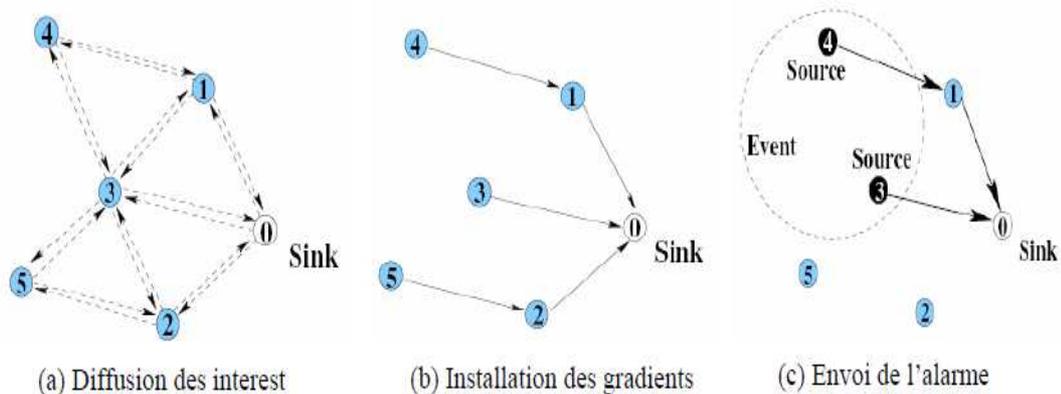


Figure 7 : Les phases de protocole Directed Diffusion

Le protocole « Directed Diffusion » permet de diminuer le nombre de messages circulant dans le réseau comparé au protocole flooding [14]. Ce type de protocole se diffère du flooding vu qu'il utilise la notion de routage par saut (multihop path routing).

8.1.3. SPIN (Sensor Protocols for Information via Negotiation)

Dans [15], [16] a été proposé une famille de protocoles appelés protocoles de capteur d'informations via négociation (SPIN). L'idée derrière le SPIN est de nommer les données en utilisant des descripteurs de haut niveau ou des métas données. Avant transmission, les métas données sont échangés entre les capteurs par un mécanisme de publicité de données. Chaque nœud recevant de nouvelles données, l'annonce à ses voisins et les voisins intéressés récupèrent les données en envoyant une requête. Cela permet à un utilisateur d'interroger n'importe quel nœud et d'obtenir immédiatement l'information demandée. Le fonctionnement du protocole SPIN permet de réduire la charge du réseau par rapport aux méthodes de diffusion traditionnelles telles que l'inondation ou l'algorithme de Gossiping.

Le protocole SPIN utilise essentiellement trois types de paquets ADV/REQ/DATA. Un nœud voulant émettre une donnée commence par envoyer un paquet ADV. Ce paquet ADV consiste d'une méta-donnée sur les données à émettre. Les méta-données peuvent décrire plusieurs aspects comme le type des données et la localisation de son origine. Les nœuds qui reçoivent ce paquet vérifient si les données les intéressent. Si oui, ils répondent par un paquet REQ. Le nœud qui a initié la communication envoie alors un paquet DATA pour chaque réponse REQ reçue (voir la Figure 8). Un nœud peut parfaitement ne pas répondre aux messages ADV, par exemple dans le but d'économiser son énergie. Ensuite chaque nœud qui fait office de relais peut très bien agréger ses propres données aux données qui sont déjà contenues dans le paquet [11].

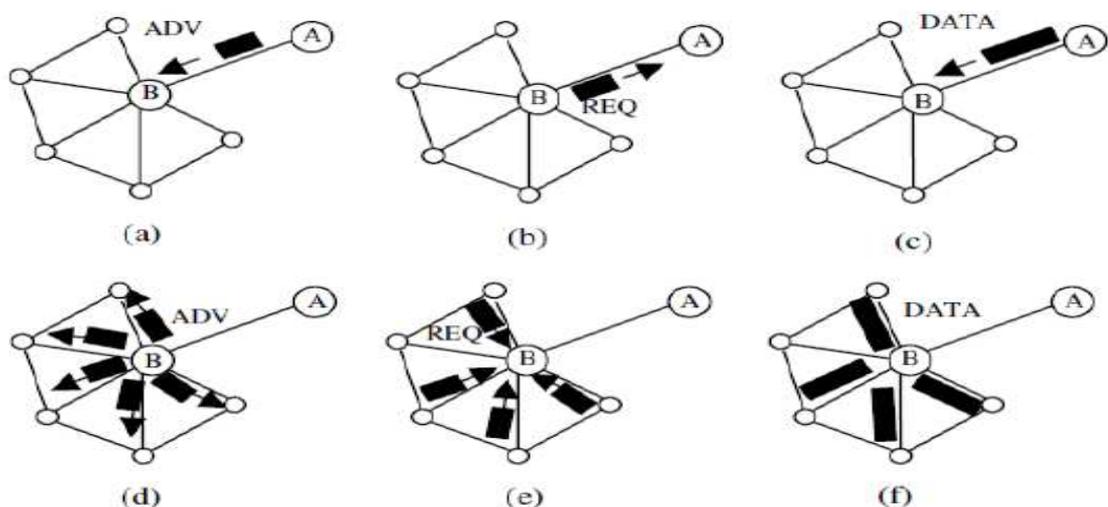


Figure 8 : Fonctionnement de protocole SPIN

8.1.4. Routage par rumeur:

L'idée est de transmettre les requêtes aux nœuds qui ont observé un évènement particulier. Quand un nœud détecte un évènement, il l'ajoute à sa table locale et génère un agent. L'agent parcourt le réseau afin de propager des informations sur les évènements locaux aux nœuds distants. Quand un nœud génère une requête pour un évènement, les nœuds qui connaissent l'itinéraire, peuvent répondre en référant la table d'évènement. [11]

8.2. Protocoles De Routage Hiérarchique

Lorsque la taille du réseau devient de plus en plus importante, sa gestion devient plus difficile. Les protocoles de routage à plat fonctionnent bien quand le réseau ne comprend pas un grand nombre de nœuds. Une des structures les plus connues est la hiérarchie. La technique de hiérarchisation sert à partitionner le réseau en sous ensembles afin de faciliter la gestion du réseau surtout le routage. Dans ce type de protocoles, des nœuds spéciaux peuvent avoir des rôles supplémentaires. Nous distinguons deux types de groupes de nœuds : la zone et le cluster.

Un cluster est défini par un ensemble de nœuds et possède un nœud nommé nœud-chef ou Cluster Head (CH). Le rôle du CH est de faire le relais entre les nœuds du cluster et la station de base directement ou via d'autres CHs. Le CH possède généralement des ressources énergétiques supérieures aux autres nœuds du réseau. Cette technique est appelée clusterisation. Une zone est définie par un ensemble de nœuds mais ne possède pas un nœud-chef (ou CH). Ainsi, un cluster est une sous-classe d'une zone [11]. Dans la suite nous citons quelques uns.

8.2.1. LEACH (Low Energy Adaptive Clustering Hierarchy)

Dans [17], ont proposé un algorithme de clustering distribué appelé LEACH pour le routage dans les réseaux de capteurs homogènes. LEACH choisit aléatoirement les nœuds cluster-heads et attribue ce rôle aux différents nœuds selon la politique de gestion Round-Robin (c'est-à-dire tourniquet) pour garantir une dissipation équitable d'énergie entre les nœuds. Dans le but de réduire la quantité d'informations transmises à la station de base, les cluster-heads agrègent les données capturées par les nœuds membres qui appartiennent à leur propre cluster, et envoient un paquet agrégé à la station de base. LEACH est exécuté en deux phases : la phase « set-up » et la phase « steady-state ». Dans la première phase, les clusters

heads sont sélectionnés et les clusters sont formés, et dans la seconde phase, le transfert de données vers la station de base aura lieu.

Cependant, bien que LEACH puisse augmenter la durée de vie du réseau, il présente certaines limitations. LEACH suppose que tous les nœuds puissent transmettre des données avec une grande puissance pour atteindre la station de base et que chaque nœud a une puissance de calcul lui permettant de supporter différentes couches MAC. Par conséquent, LEACH ne convient pas aux réseaux déployés dans de vastes régions. En outre, LEACH choisit aléatoirement la liste des clusters heads et il ne pose aucune contrainte sur leur distribution ainsi que sur leur niveau d'énergie. Ainsi, les clusters heads peuvent se concentrer dans un même endroit et par conséquent, il pourrait exister des nœuds isolés (sans cluster head) pouvant se déclarer. D'autre part, dans LEACH, l'agrégation des données est centralisée et est exécutée périodiquement. Or, dans certains cas, la transmission périodique des données pourrait ne pas être nécessaire, ce qui épuise rapidement l'énergie limitée des capteurs.

Une variante de LEACH appelée LEACH-C [18] a été conçue pour améliorer les performances de LEACH. Cette variante utilise une architecture centralisée pour choisir les clusters heads tout en impliquant la station de base et l'information de localisation des capteurs. Cependant, elle augmente considérablement le surcoût du réseau puisque tous les capteurs devront envoyer leurs informations de localisation à la station de base en même temps pendant chaque phase d'élection de cluster heads.

Dans [19], les auteurs ont proposé un protocole hiérarchique basé sur LEACH. Les clusters heads formés dans LEACH sont groupés et organisés en une hiérarchie. Ils ont montré que la consommation énergétique diminue lorsque le nombre de niveaux de l'hiérarchie augmente.

8.2.2. 8.2.2.PEGASIS (*Power-Efficient Gathering in Sensor Information Systems*)

Dans [20], ont proposé une version améliorée de LEACH appelée PEGASIS. L'idée principale de PEGASIS est de former une chaîne entre les nœuds de sorte que chaque nœud communique à un voisin proche. Les données collectées sont transmises d'un nœud à un autre qui les agrège jusqu'à ce qu'elles arrivent à un nœud particulier qui les transmet à la station de base. Les nœuds qui transmettent les données à la station de base, sont choisis tour à tour selon une politique round-robin dans le but de réduire l'énergie moyenne dépensée par un

nœud durant une période (round). Contrairement à LEACH, PEGASIS évite la formation des clusters et procure à un seul nœud dans la chaîne l'envoi de données à la station de base. D'ailleurs, PEGASIS suppose que les nœuds sont capables de modifier leur puissance de transmission.

Les résultats de simulation ont montré que PEGASIS peut prolonger de deux à trois fois la durée de vie d'un réseau de capteurs relativement à LEACH en fonction du critère choisi pour évaluer la durée de vie d'un réseau. Un tel gain de performance est réalisé par l'élimination du surcoût causé par le processus de formation de clusters dans LEACH, et par la réduction du nombre de transmissions et de réceptions en agrégeant de données. Bien que le surcoût du clustering soit évité, PEGASIS exige toujours un ajustement dynamique de la topologie puisqu'un nœud devrait connaître le niveau d'énergie de ses voisins avant de relayer ses données. Cependant, un tel ajustement de la topologie pourrait causer un surcoût important. En outre, PEGASIS suppose que tout nœud communique directement avec la station de base qui gère la topologie d'une manière centralisée. Or, cette supposition est loin de la réalité car les capteurs communiquent généralement en mode multi-sauts pour atteindre la station de base. D'autre part, PEGASIS suppose que tous les nœuds maintiennent une table contenant les localisations de tous les autres nœuds dans le réseau. En résumé, PEGASIS est adapté seulement aux capteurs sans fil dont les nœuds sont immobiles. Son évaluation dans des environnements mobiles pourrait dégrader considérablement ses performances [11].

8.3. Protocole basé sur la localisation

Les protocoles à topologie basée localisation suppose que (voir la figure 9) :

- Le réseau est partitionné en plusieurs zones de localisation.
- Chaque zone a son identifiant.
- Chaque nœud a un identifiant EUI (End-system Unique Identifier) et enregistre dynamiquement l'identifiant de la zone à laquelle il appartient temporairement.

L'information temporaire de localisation appelée LDA (Location Dependent Address) qui est un triplet de coordonnées géographiques (longitude, latitude, altitude) obtenues, par exemple, au moyen d'un GPS avec une précision dépendant du type de l'application. Une telle topologie exige l'implémentation d'un algorithme de gestion de localisation qui permet aux

nœuds de déterminer les endroits approximatifs des autres nœuds. Ce type de topologie est mieux adapté aux réseaux avec une forte mobilité.

Avant d'envoyer ses données à un nœud destination, le nœud source utilise un mécanisme pour déterminer la localisation de la destination puis inclus l'identifiant de zone de localisation et du nœud destination dans l'entête du paquet à envoyer [11].

À titre d'exemple des protocoles utilisant une topologie basée localisation nous pouvons citer GEAR (Geographic and Energy Aware Routing) et LAR (Location-Aided Routing protocol).

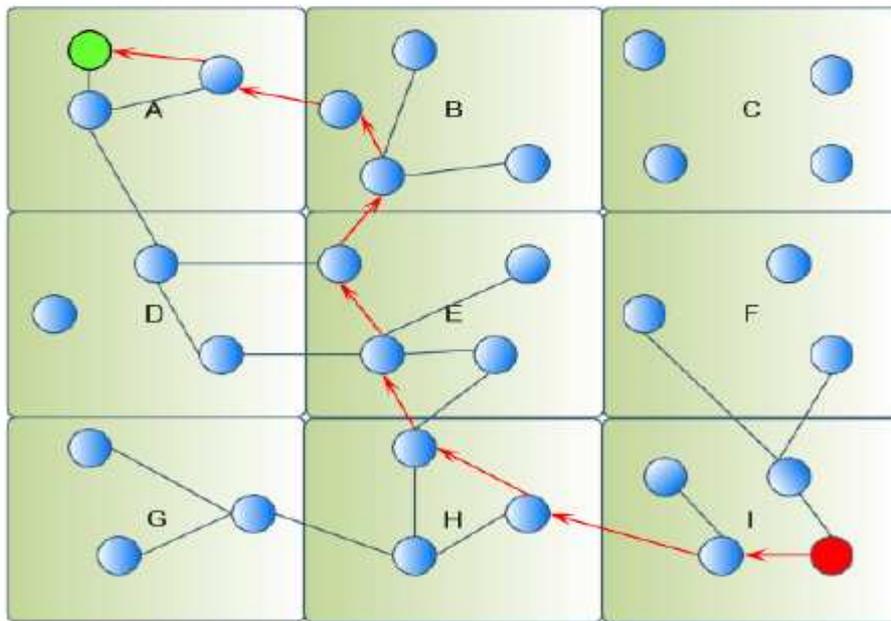


Figure 9 : Topologie basée sur la localisation

8.3.1. GEAR

GEAR (Geographic and Energy Aware Routing) [11], [21] suggère l'utilisation d'information géographique tout en disséminant les demandes des données vers les régions appropriées plutôt que d'envoyer les intérêts à tout le réseau puisque ces demandes incluent souvent des attributs géographiques concernant une certaine région. GEAR se comporte comme un protocole de routage sur demande. Dans le cas de ce type de protocoles, la

découverte de la route est basée sur la demande. Ainsi il n'y a aucun maintien de table de routage dans le nœud. La découverte de la route est lancée seulement quand un nœud veut transmettre des données, aussi aucune mise à jour périodique ou le maintien des tables de routage n'est effectué.

Chaque nœud garde un coût estimé et un coût instruit pour atteindre une région cible à travers ses voisins. Le coût estimé est une combinaison entre l'énergie et la distance résiduelles jusqu'à la destination. Le coût instruit est une amélioration du coût estimé qui prend compte d'autres paramètres de routage comme les trous dans le réseau. Un trou se produit quand un nœud n'a aucun voisin plus proche de la région cible que lui-même, alors le trou doit être contourné. S'il n'y a aucun trou, le coût estimé est égal au coût instruit. Il y a deux phases dans l'algorithme de livraison des paquets.

La première phase c'est l'expédition des intérêts vers la région de cible: après avoir reçu un paquet, le nœud en question vérifie si ses voisins appartiennent à la région cible, si il existe un, il est choisis comme le prochain saut. Si non, il vérifie ses voisins si l'un d'eux est plus proche de la région cible que lui même. Si il y en a plus, le voisin le plus proche de la région cible est choisi comme prochain saut. S'ils sont tous plus loin que le nœud lui-même, ceci signifie qu'il y a un trou. Dans ce cas, un des voisins est sélectionné aléatoirement pour expédier le paquet en se basant sur coût instruit dans le but de contourner le trou. Ce choix peut être mis à jour selon la convergence du coût instruit pendant la livraison des paquets. La deuxième phase c'est l'expédition des paquets dans la région : Si le paquet a atteint la région, il peut être diffusé dans cette région par le Flooding géographique récursif ou le Flooding restreint. Le Flooding restreint est bon lorsqu'on n'a pas une grande densité de capteurs dans la région. Dans les réseaux à haute densité, le Flooding géographique récursif est plus efficace en termes d'énergie que le Flooding restreint.

Chaque nœud connaît sa propre localisation, son niveau d'énergie, la localisation de chacun de ses voisins et leurs niveaux d'énergie par l'échange des messages d'information (échange des messages Hello). Un chemin existe entre deux nœuds si chacun d'eux a un niveau d'énergie suffisante pour échanger les paquets entre eux (chaque nœud est dans la marge de transmission de l'autre), dans le cas contraire c'est un trou.

10. Consommation d'énergie dans les RCSF

L'énergie consommée par un nœud capteur est due essentiellement aux opérations suivantes : la capture, le traitement et la communication de données [1].

Energie de capture

L'énergie de capture est dissipée pour accomplir les tâches suivantes : échantillonnage, traitement de signal, conversion analogique/numérique et activation de la sonde du capteur. En général, l'énergie de capture représente un faible pourcentage de l'énergie totale consommée par un nœud.

Energie de traitement

L'énergie de traitement se divise en deux parties : 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.

Energie de communication

L'énergie de communication se divise en deux parties : l'énergie de réception et l'énergie de l'émission. 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.

10.1. Modèle de consommation d'énergie

Dans [17] a été proposé un modèle radio de consommation d'énergie (voir figure 10). Ainsi, les énergies nécessaires pour émettre $E_{Tx}(s, d)$ et recevoir $E_{Rx}(s)$ des messages sont données par:

- Pour émettre un message de s bits vers un récepteur loin de d mètres, l'émetteur consomme:

$$E_{Tx}(s, d) = E_{Tx\ elec}(s) + E_{Tx\ amp}(s, d).$$

$$E_{Tx}(s, d) = (E_{elec} * s) + (E_{amp} * s * d^2).$$

- Pour recevoir un message de s bits, le récepteur consomme :

$$E_{Rx}(s) = E_{Rx\ elec}(s)$$

$$E_{Rx}(s) = E_{elec} * s$$

E_{elec} et E_{amp} représentent respectivement l'énergie de transmission électronique et d'amplification.

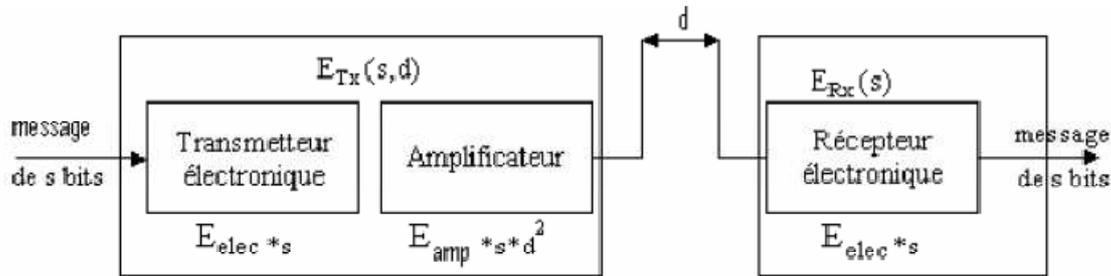


Figure 10: Modèle de consommation d'énergie.

11. Mécanismes de conservation de l'énergie dans les RCSF

11.1. Au niveau des différentes couches

L'énergie consommée par les réseaux de capteurs est répartie entre le traitement, la capture et la communication. En ce qui a trait à la communication, beaucoup de protocoles et mécanismes ont été proposés dans la littérature pour optimiser la consommation de l'énergie, surtout au niveau des couches directement impliquées dans la communication (les couches physique, liaison, réseau et transport). Dans ce qui suit, nous allons décrire quelques mécanismes de conservation de l'énergie au niveau de ces couches.

❖ Couche physique

Comme démontré dans [22], parmi toutes les opérations effectuées par un capteur (capture, traitement et transmission de données), la communication (émission et réception des données) est l'opération qui consomme le plus d'énergie.

Au niveau de la couche physique, la compression, ou codage de source [23], est une technique qui permet de réduire le taux moyen de données transmises par des capteurs dont les lectures sont corrélées selon une fonction de corrélation connue.

D'autres techniques au niveau de la couche physique consistent en une planification dynamique du voltage qui alimente le processeur, en fonction des besoins de calcul de ce dernier. Cette famille de techniques s'appelle Dynamic Voltage Scheduling (DVS) [24]. Il s'agit de prévoir le comportement du processeur et d'adapter sa fréquence et son courant en entrée en fonction des tâches qu'il a accomplir. Pour cela, des algorithmes complexes appelés "planificateurs de voltage" (Voltage Schedulers (VS)) sont requis pour évaluer la charge future du processeur et adapter son courant d'alimentation et sa vitesse. Ces techniques demeurent cependant très complexes à implémenter.

❖ Couche liaison

La conservation de l'énergie des capteurs au niveau de la couche liaison passe principalement par deux techniques : la limitation du temps d'accès au canal, moyennant des mises en veille des capteurs, et une gestion plus efficace des transmissions afin de limiter les collisions. Ces deux techniques sont en fait liées car une limitation de l'accès au canal tend à baisser le taux de contention sur le canal et donc baisser la probabilité de collision et de retransmission, ce qui va sauvegarder l'énergie des nœuds émetteurs. S-MAC [25] est un protocole adaptatif d'accès au canal qui introduit une modification sur la couche MAC afin de l'adapter aux contraintes énergétiques des RCSF. S-MAC favorise l'accès au canal aux nœuds ayant beaucoup de trafic à transmettre. S-MAC permet aussi aux nœuds de passer en mode "En veille" (Sleep) périodiquement lorsqu'il n'y pas de trafic à acheminer pour éviter l'écoute excessive en l'absence de trafic. Dans [26] a été proposé un mécanisme de conservation de l'énergie qui consiste à réduire les transmissions en évitant les collisions. Pour ce faire, la taille des trames est modifiée et continuellement adaptée aux conditions du canal. Quand le canal présente un bruit élevé, la taille de la trame est diminuée pour réduire le risque de collision et donc de retransmission.

❖ Couche réseau

Certains travaux s'intéressent à minimiser et à équilibrer la consommation d'énergie des nœuds au niveau de la couche réseau. Ainsi, un bon nombre de protocoles de routage qui optimisent l'énergie dissipée [11] ont été proposés. Par exemple, dans la catégorie des protocoles de routage orientés données, SPIN [15] intègre la conservation de l'énergie grâce à la négociation. En effet, avant d'envoyer toutes les données, qui pourraient être volumineuses, un nœud diffuse d'abord dans tout le réseau une méta-donnée décrivant les données à

transmettre. Seuls les nœuds qui manifestent leur intérêt pour cette méta-donnée recevront les données en entier, ce qui sauvegarde considérablement la bande passante et l'énergie des nœuds.

Dans le protocole présenté dans [14], et qui est un protocole basé sur la diffusion orientée (Directed Diffusion), la conservation de l'énergie est réalisée grâce à l'agrégation de données. Dans [27] ont proposé de répartir la charge de routage sur un ensemble de chemins sous optimaux pour ne pas épuiser l'énergie des mêmes nœuds faisant partie du chemin optimal et ainsi prolonger la durée de vie du réseau.

Dans les protocoles de routage orientés position, ils nécessitent la connaissance de l'emplacement géographique des nœuds pour optimiser le choix de la route. Par exemple, le protocole Geographical Adaptive Fidelity (GAF) [28] qui conserve l'énergie du réseau en éteignant les capteurs qui ne sont pas nécessaires pour le fonctionnement du réseau.

❖ Couche transport

Au niveau de la couche transport, quelques protocoles ont été proposés pour assurer une communication fiable de bout en bout dans les RCSF.

Event-to-Sink Reliable Transport protocol (ESRT) [29] est un protocole qui assure un transport fiable des données des capteurs vers la station de base. Pump Slowly, Fetch quickly (PSFQ) [30] est un protocole de transport basé sur une garantie locale d'arrivée des données, c'est-à-dire que chaque nœud intermédiaire doit conserver une copie du paquet transmis en cache jusqu'à s'assurer que ce dernier est bien arrivé au nœud suivant sur le chemin.

11.2. Conservation de l'énergie sous contrainte de couverture

L'optimisation de l'énergie dissipée sous contrainte de couverture dans les RCSF se base généralement sur l'alternance des états des capteurs entre un état d'activité et un autre de mise en veille. Par exemple, dans [31] font une analyse géométrique de la relation entre la couverture de la zone surveillée et la connectivité du réseau. Ils proposent Coverage Configuration Protocol (CCP), un protocole qui permet de configurer dynamiquement le réseau pour garantir différents degrés de couverture en fonction des requis de l'application. Dans CCP, chaque capteur choisit son état (Actif ou En Veille) en fonction du degré de couverture des points d'intersection de son cercle de couverture avec celui de ses voisins. Il

permet ainsi de garantir une K-couverture, c'est à dire la couverture de chaque point de surface par au moins K capteurs.

L'algorithme Scalable COverage Maintenance (SCOM) [32], est un protocole localisé de maintien de couverture, comme celui proposé dans [31], et dans lequel les capteurs utilisent une politique de redondance pour décider s'ils doivent s'allumer ou se mettre en veille. SCOM propose d'implémenter, pour chaque capteur, un compteur d'attente (back-off timer) proportionnel à son énergie résiduelle. Ce mécanisme de compteur permet aux capteurs ayant une énergie résiduelle faible de décider de leur état avant ceux ayant une énergie résiduelle plus élevée, ce qui leur permet, de se mettre en veille avant les capteurs à énergie résiduelle plus élevée, favorisant ainsi la prolongation de la durée de vie du réseau.

11.3. Conservation de l'énergie par la formation de grappes (clustering)

La formation de grappes est une technique qui consiste à subdiviser le réseau en sous-ensembles de nœuds, appelés grappes. Les capteurs sont alors groupés en sous-ensembles ayant chacun un capteur désigné comme TG (Tête de Grappe) et des nœuds membres connectés à la TG. Les nœuds membres transmettent leurs données à la TG qui, à son tour, les achemine jusqu'au centre de traitement, soit directement [17], soit en multi-sauts via des passerelles ou d'autres TG voisines [33]. Les TG peuvent effectuer certaines opérations comme le filtrage et l'agrégation des données collectées, ce qui permet de limiter la quantité de données transmises au station de base et ainsi alléger la bande passante et sauvegarder l'énergie des nœuds collecteurs. Dans certains cas, la formation de grappes offre des économies d'énergie importantes [11], [34] puisque seules les TG sont impliquées dans les tâches de collecte, routage et agrégation des données.

Plusieurs protocoles de formation de grappes ont été proposés dans la littérature et ont généralement le prolongement de la durée de vie du réseau comme objectif principal. Pour cela, la formation de grappes est typiquement basée sur l'énergie résiduelle des nœuds. Par exemple, Low-Energy Adaptive Clustering Hierarchy (LEACH) [17] est l'un des premiers protocoles de formation de grappes proposés pour les RCSF, il est réparti, proactif et dynamique. Dans LEACH, la formation des grappes se base sur la puissance du signal reçu pour évaluer le degré d'un nœud et sa proximité de ses voisins. Chaque nœud prend sa décision de devenir ou non TG en se basant sur le nombre de fois qu'il a joué ce rôle dans le passé. Les TG sont les seuls nœuds qui peuvent router les données vers la station de base, ce qui sauvegarde l'énergie des autres capteurs.

Power-Efficient GATHERING in Sensor Information Systems (PEGASIS) [20] et sa variante Hierarchical-PEGASIS sont deux améliorations de LEACH; plutôt que de former plusieurs grappes, PEGASIS forme une chaîne de capteurs voisins telle que chacun de ces capteurs envoie et reçoit à partir de ses voisins directs et un seul de ces capteurs est sélectionné pour envoyer directement les données à la station de base.

11.4. Par l'ajustement optimisé des puissances de transmission

Dans un RCSF où les capteurs sont équipés de plusieurs puissances de transmission ou d'une puissance ajustable, l'énergie consommée dépend directement de l'énergie de transmission. Certains travaux se sont penchés sur l'optimisation de la puissance de transmission des capteurs. Par exemple, dans [35] ont proposé un protocole réparti qui permet d'ajuster dynamiquement la puissance de transmission de chaque capteur pour contrôler la taille des grappes, indépendamment de la distribution des nœuds. L'idée de base du protocole proposé est que chaque capteur ajuste sa puissance de transmission en fonction du nombre de ses voisins.

11.5. Par la planification optimisée des états des capteurs

Beaucoup de travaux se sont intéressés à la planification optimale des états des capteurs et à l'ordonnancement optimal des activités de ces derniers, dans le but de minimiser l'énergie consommée par le réseau. Il s'agit, d'une façon générale, d'alterner entre un (ou des) état(s) où le capteur est activé, et donc consomme un certain niveau d'énergie, et un autre où il est éteint et donc consomme une quantité d'énergie négligeable. Beaucoup de travaux proposent des mécanismes qui mettent en veille des capteurs, autant que possible, pour préserver leur énergie. Par exemple, dans [36] ont proposé un algorithme d'ordonnancement des transmissions des capteurs dans une architecture hiérarchisée avec un seul niveau de TG, où les capteurs disposent de séquences de données de différentes longueurs à transmettre durant une période de temps T . L'algorithme proposé trouve, pour chaque capteur, un ensemble optimal d'intervalles temporels (time slots) de transmission, qui minimise l'énergie dissipée par tout le réseau durant une période T . Cependant, aucune planification traitant la contrainte de couverture n'est considérée.

D'autres travaux se sont intéressés à la planification spatiale des états des capteurs, basée sur une activation sélective d'un sous-ensemble de capteurs, de manière à maximiser la durée de vie du réseau. Par exemple, [37] proposent un algorithme réparti pour trouver un sous-ensemble optimal de couverture ou, si ce n'est pas possible, un ensemble de couverture

offrant le minimum de points géographiques non couverts. Dans cet algorithme, si la portée entière d'un capteur est couverte par un sous-ensemble de ses voisins directs, alors ce capteur se met en veille.

12. Conclusion

Les réseaux de capteurs sans fil présentent un intérêt considérable et une nouvelle étape dans l'évolution des technologies de l'information et de la communication. Cette nouvelle technologie suscite un intérêt croissant vu la diversité de ces applications : santé, environnement et industrie.

Dans ce premier chapitre, nous avons présenté les réseaux de capteurs sans fil, leurs architectures de communication, la pile protocolaire des capteurs, leurs diverses applications et les différents protocoles de routage utilisés dans ce type de réseau.

Cependant, nous avons remarqué que plusieurs facteurs et contraintes compliquent la gestion de ce type de réseaux. En effet, les réseaux de capteurs se caractérisent par une capacité énergétique limitée rendant l'optimisation de la consommation d'énergie dans des réseaux une tâche critique pour prolonger la durée de vie du réseau. Nous avons présenté quelques techniques de conservation d'énergie.

Dans le chapitre suivant, nous aborderons sur un type particulier des RCSFs qui sont les réseaux de capteurs sans fils de vision.

Chapitre 2 :

Les Réseaux de Capteur Sans Fil

de Vision

1. Introduction :

Le développement des micros caméras et microphones a observé une forte évolution au cours de la dernière décennie, avec les évolutions des téléphones mobiles. Ces dispositifs deviennent de plus en plus petits et fournissent de plus en plus de performances en termes de rapidité et de qualité du signal. Jusqu'`a il y a quelques années, l'usage d'un appareil photo impliquait la connexion d'un périphérique attaché au téléphone mobile. Aujourd'hui, nous trouvons ces micros caméras embarquées dans pratiquement tous les téléphones cellulaires et les assistants numériques personnels, sans augmentation significative du coût de l'équipement, de son poids et de sa forme.

Les réseaux sans fil n'ont pas été en dehors de ce progrès et aujourd'hui, nous pouvons voir des micro-caméras CMOS, sous la forme de cartes de capteurs compatibles avec des nœuds sans fil, Cela a permis la naissance d'un nouveau type d'applications utilisant des réseaux de capteurs sans fil multimédia.

La conservation de l'énergie est un problème fondamental dans les réseaux de capteurs sans fils, en effet les nœuds du réseau ont une durée de fonctionnement limitée à la capacité de leur batterie. Pour les applications engageant des capteurs d'image, ce problème de l'énergie est très largement accentué par la dimension de l'information transportée, la taille des images en comparaison des valeurs scalaires classiques (une mesure de température, par exemple) étant supérieur de plusieurs ordres de grandeur.

Une solution évidente pour diminuer la quantité de données envoyée, et donc l'énergie consommée dans le réseau, est de compresser l'image à la source. La contrainte de la limitation des ressources nœuds, comme la capacité de traitement et de stockage de données, rend impossible ou inefficace en pratique l'exécution des algorithmes de compression standard.

2. Applications

Parmi les nombreuses applications potentielles des réseaux de capteurs multimédia, celles utilisant des capteurs d'image sont appréciables pour tout ce qui concerne la reconnaissance, la localisation et le dénombrement d'objets par la vision. Certaines applications ont besoin d'identifier exactement le ou les objets qui traversent le champ du réseau de capteurs. Ce le cas par exemple dans surveillances environnementales comme

d'étude du comportement des oiseaux, ou il faut repérer qui entre et sort du nid, ou combien d'œufs il y a dans le nid. Ce repérage n'est possible qu'à travers la prise d'images.

D'autres applications n'ont pas besoin directement d'images, mais la prise d'image peut servir à compléter et enrichir les mesures initiales. La surveillance des feux de forêt en constitue un exemple. Ce type d'application collecte des mesures de température pour détecter les départs de feux, mais la prise d'image va aider à avoir une idée plus précise de la situation. Dans les cas mentionnés, les mesures de données scalaires vont nous aider à obtenir une certaine idée de ce qu'il se passe sur le terrain, mais la visualisation directe d'images permettra une classification plus efficace du phénomène étudié.

Il existe de nombreuses possibilités d'application des réseaux de capteurs d'image dans des scénarios réels. Voici des exemples

2.1 Applications militaires

Les réseaux de capteurs de vision peuvent être très utiles pour l'espionnage militaire et la surveillance des champs de bataille. Un réseau de capteurs de vision pourrait être utilisé pour la reconnaissance et la classification des cibles, par exemple. Le déploiement d'un réseau de capteurs sur les champs de bataille peut être réalisé manuellement ou aléatoirement. Dans le premier cas, des troupes de reconnaissance peuvent marcher dans le champ de bataille pour positionner et cacher stratégiquement des capteurs d'image. Comme les capteurs sont, par définition, des dispositifs très petits, ils seront à priori faciles à dissimuler. L'orientation des caméras doit être effectuée rigoureusement si on veut couvrir visuellement toute la zone à observer. Cela peut être un problème lorsque le réseau de capteurs est déployé aléatoirement, par exemple en larguant les capteurs depuis un avion. Comme les caméras ont un angle de vue limité, le risque est grand que des nœuds tombent dans une mauvaise position (pointant vers le sol, le ciel, etc.).

2.2 Vigilance environnementale

Les réseaux de capteurs de vision sont aussi utiles pour la vigilance environnementale. Comme les nœuds consomment très peu d'énergie, ils peuvent être déployés dans des endroits stratégiques pour de longues périodes de temps (de l'ordre de plusieurs mois), afin d'obtenir des images d'intérêt scientifique sur de larges zones géographiques. Grâce à cela, les observateurs de la nature peuvent étudier le comportement et les habitudes des diverses espèces animales, en obtenant des scènes de lieux, qui peuvent être très éloignées, sans avoir

besoin de se déplacer physiquement. Des systèmes pour les tâches de gardes de parcs, comme la détection des incendies de forêt, pourraient être développés.

2.3 Sûreté et sécurité de zones sensibles

Les réseaux de capteurs de vision peuvent être appliqués pour la sécurité des zones privées et publiques. Néanmoins, le véritable intérêt des réseaux de capteurs de vision sans fil n'est pas dans la surveillance des établissements fermés (par exemple, des industries, des bureaux, des magasins commerciaux, des résidences, etc.). Pour ce type de demande, une longue liste de produits est disponible sur le marché. Des webcams rotatives sans fil, micro-caméras et autres dispositifs existent déjà pour la vidéo-surveillance. Ils s'appuient sur des méthodes de compression et des protocoles de communication normalisés, et des technologies de transmission à haut débit filaires ou sans fil.

Des systèmes de surveillance sans fil limités en énergie pourraient être mis en place pour protéger des parcs, des zones sauvages, et d'autres zones liées à la protection des ressources naturelles. D'autres applications pourraient être trouvées dans la surveillance des lieux privés ouverts, comme dans les industries forestières ou agricoles, par exemples.

2.4 Suivi du trafic routier

Des réseaux de capteurs de caméras peuvent être déployés pour le suivi et le contrôle de la circulation routière. Des algorithmes d'analyse d'images peuvent être utilisés pour faire le dénombrement des véhicules ou des personnes pour estimer le niveau de trafic en fonction des heures de la journée.

2.5 Applications à la robotique

Dans [38] un réseau de caméras sans fil est employé pour des robots miniatures dans des applications de recherche et de sauvetage dans les zones urbaines. Dans cette expérience, une série de petits robots de capacité limitée se déplace dans une zone sinistrée. Un robot équipé d'une caméra joue le rôle de source, en enregistrant des séquences d'images et en les transmettant vers le puits à travers de multiples autres nœuds (robots) qui se déplacent dans la région.

3. Classification des RCSF de vision :

En fonction des exigences imposées à l'application, et bien évidemment en fonction du type de technologie disponible, les réseaux de vision peuvent être de deux types :

3.1 Réseaux de capteurs d'images fixes :

Des capteurs d'images numériques peuvent prendre des photos qui peuvent être mémorisées en format matriciel ou vectoriel. Ce type de capteur est facile à réaliser et peut être adapté facilement à des dispositifs avec des ressources limitées, tels que les nœuds de capteurs sans fil.

3.2 Réseaux de capteurs de vidéo :

Des capteurs d'images numériques peuvent aussi envisager de prendre des séquences d'images et de transmettre le flux vidéo vers le puits. Cette application exige des nœuds avec des capacités de calcul, de mémoire et de communication que pour les images fixes. Les séquences d'image doivent être compressées fortement pour satisfaire à la contrainte de bande passante des liaisons sans fil. Ces applications consomment nécessairement une quantité d'énergie bien supérieure à celles utilisant des images fixes.

En raison de la difficulté que comporte la transmission d'un flux vidéo, la plupart des prototypes de capteurs d'images sont dédiés aux images statiques.

4. Contraintes des réseaux de capteurs de vision

Les travaux sur des images sont plus complexes que sur des signaux numériques ou analogiques qui sont plus simples. Ces différences sont dues évidemment à la complexité du signal capturé. En effet, tandis que pour le codage d'un signal simple tel que le niveau de température ou la pression, un ou deux octets sont suffisants, le codage d'une image numérique conduit à l'emploi de plusieurs centaines ou milliers d'octets. Cette différence de grandeur a des conséquences sur différents facteurs : capture du signal, besoins en mémoire, traitement du signal et transmission de données.

4.1 Capture du signal :

La complexité du matériel est multipliée par rapport aux captures de phénomènes simples. En effet, un capteur de caméra CMOS est normalement composé de nombreux

capteurs photo-sensibles qui capturent les différentes intensités pour chaque pixel. Tandis que pour la capture d'un signal de lumière un seul photo-capteur est suffisant, pour capturer une image nous avons besoin de beaucoup plus (normalement un par pixel). Cette évidence entraîne avec elle un coût supplémentaire en énergie et en temps de capture.

4.2 Besoins de mémoire :

Comme nous l'avons dit, tandis que pour le codage d'un signal simple sollicite quelques bits d'information (de 1 à 8 octets, en fonction de la précision du capteur), le codage d'une image numérique conduit à l'emploi de plusieurs centaines ou milliers d'octets. En particulier, la quantité de mémoire nécessaire dépend principalement de deux facteurs clés : La résolution de l'image et le format. En effet, une image de 128×128 pixels utilisera en principe 4 fois plus de mémoire qu'une image de 64×64 . Maintenant, en fonction du format, une image peut être en noir et blanc, en niveaux de gris ou en couleur [39]. En principe, le format définit le nombre de bits nécessaires pour coder un pixel (une intensité capturée par l'un des photo-capteurs). Une image en niveaux de gris est normalement codée sur 8 bits par pixel (désigné 8bpp). Pour coder un pixel en couleur, nous pouvons le faire en utilisant, soit un codage RGB (Red, Green, Bleu), soit un codage YCrCb. Ceci implique normalement l'utilisation d'un octet par plan de couleur (24bpp).

4.3 Traitement du signal :

Dans les applications traditionnelles de vision, il est commun de vouloir faire des traitements sur les images à la source, afin d'extraire une information (par exemple : l'emplacement d'un objet), ou de compresser l'image afin de diminuer la quantité de données nécessaires pour la représenter. Alors que ces traitements sont aisés à mettre en œuvre dans des dispositifs informatiques dotés de beaucoup de ressources, comme les ordinateurs portables et les assistants numériques personnels, compte tenu des capacités limitées des matériels utilisés dans les réseaux de capteurs, le traitement d'image à la source devient très difficile. Les temps de calcul sont considérablement augmentés et l'énergie investie est parfois plus importante que celle économisée.

4.4 Transmission de données :

Comme l'unité de communication est l'un des composants les plus gourmands en énergie, les protocoles de communication ont un rôle important à jouer pour faire des

économies d'énergie. Dans les applications traditionnelles (par exemple : la température ou le mouvement), on peut envisager d'enregistrer plusieurs mesures et les embarquer dans un seul paquet pour augmenter le rendement de la communication. Une des techniques les plus utilisées est la fusion (agrégation) de données. Cela est possible parce que les mesures des différents capteurs sont généralement codées sur peu de bits et nous pouvons créer de paquets combinant des informations provenant de plusieurs sources. Dans le cas des images, la fusion de données n'est plus possible puisque les images sont transmises sur plusieurs centaines ou même milliers de paquets. Toutefois, les images naturelles ont des corrélations spatiales assez marquées et par conséquent la transmission d'images offre une certaine tolérance aux pertes de paquets. En effet, on peut reconstruire une version approximative de l'image originale même si une partie des informations est perdue dans le réseau. Ces corrélations spatiales sont exploitées dans les algorithmes de compression.

5. Capteur d'image

Les capteurs d'image de faible consommation d'énergie ont fait l'objet de grands progrès au cours des dernières années. En réponse à la forte demande du marché, nous pouvons trouver des capteurs d'image de plus en plus petits et de résolutions de plus en plus grandes, destinés principalement à être intégrés dans les téléphones portables, les ordinateurs portables, et les PDAs. Cependant, ces dispositifs sont dotés de ressources importantes en termes de mémoire et vitesse de calcul. Les développements ont été centrés sur l'offre de meilleures qualités d'image et de taux de compression plus élevés puisque la demande des utilisateurs porte surtout sur ces aspects. L'autonomie en énergie est aussi importante, mais c'est de l'ordre de la journée.

Dans les réseaux de capteurs sans fil de vision, comme dans les réseaux de capteurs en général, le problème de la consommation d'énergie est d'un autre ordre de grandeur, les nœuds devant avoir une autonomie de l'ordre du mois, voir de l'année. Dans beaucoup d'applications, la résolution des images n'a pas besoin d'être très élevée. Pour compter des œufs dans un nid par exemple, une image de 64×64 pixels sur 16 ou 32 niveaux de gris est suffisante. Selon le travail effectué dans [40], les nœuds de capteurs d'image doivent avoir une capacité de calcul et de mémoire très importante, répondre à des contraintes de temps réel, et avoir une unité de communication haut débit, tout en consommant peu d'énergie. Les

dispositifs disponibles actuellement ne sont pas encore capables d'atteindre ces niveaux d'exigence.

Plusieurs travaux fondent leurs prototypes de capture d'image sur des composants commerciaux de faible consommation d'énergie. Par exemple dans [41] ont utilisé un capteur d'image ADCM 1670 1 de Agilent pour fournir des capacités de vision dans une application de pistage d'objets.

Le travail [42], de l'Université de Stanford, ont utilisé le même capteur d'image dans leur architecture intégrée de capteurs sans fil qui comprend un module de capteur d'image, de traitement de données et de communication tout en un. Le « mote » est constitué d'un microprocesseur ARM7 32 bits d'Atmel, avec 64ko de mémoire RAM et 256Ko de mémoire flash, un module radio Chipcon CC2420 qui est basé sur le standard 802.15.4, et plusieurs interfaces.

Une autre architecture de capteur d'image sans fil est proposée dans [40]. Celle-ci comprend un module radio Chipcon CC1000, un processeur S3C44BOX de chez Samsung, une caméra VGA, une SDRAM comme mémoire principale pour le processeur, une SRAM pour le module de traitement d'images et de la mémoire flash.

5.1 Caméras basées sur des composants commerciaux

Plusieurs nœuds et cartes de capture d'images pour des applications de réseaux de capteurs sans fil ont été développés en utilisant des composants électroniques commerciaux (dit en anglais composants COTS par Commercial Off-The-Shelf), par exemple, des caméras CMOS, des microcontrôleurs, des mémoires, .etc.

- ***MeshEye :***

C'est un mote-caméra intelligent conçu pour la surveillance distribuée. La mote MeshEye intègre un microcontrôleur Atmel AT91SAM7S, une mémoire flash MMC/SD, deux capteurs ADNS-3060 originellement utilisés pour des souris optiques (le mote permet jusqu'au huit de ces capteurs), une caméra CMOS ADCM-2700 et un module radio CC2420 de Texas Instruments qui respecte la norme 802.15.4 [43].

- *Cyclops* :

Les caméras Cyclops [44] 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 <http://cvs.cens.ucla.edu/>) de l'UCLA. Elle permet la capture et le traitement d'images de faible résolution avec une faible consommation d'énergie. Quatre versions de Cyclops ont été développées : Cyclops1, Cyclops2A, Cyclops2B et Cyclops2C (seule cette dernière version est traitée, parce qu'elle intègre des améliorations significatives par rapport aux versions plus anciennes). 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 motes 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 que lui permet d'être attachée à un mote Mica2 ou Micaz de Crossbow.



Figure 11 : Cyclops attaché à un mote Mica2

Le microcode de la caméra Cyclops a été écrit avec le langage de programmation NesC/TinyOS. La dernière version de Cyclops supporte différentes dimensions d'image. La résolution maximale pour le capteur ADCM est CIF (352×288), mais le microcode programmé par défaut limite la taille maximale à 128×128 pixels, due probablement à des restrictions du matériel. Elle peut générer des images de trois formats différents : monochrome codé 8bpp, couleur RGB (24 bits) et YCbCr couleur (16 bits). La carte Cyclops peut aussi avoir différents états, qui définissent l'énergie consommée par la plateforme. Par exemple, quand on capture une image, Cyclops consomme 42mW, 0.7mW quand on est en mode endormi et moins de 1μW en état OFF. Afin d'économiser de l'énergie, Cyclops est une bonne initiative dans la route vers un capteur d'image peu gourmand en énergie. Cependant,

ces consommations d'énergie pourraient être trop importantes pour des applications de réseaux de capteurs de longue durée. En outre, Cyclops présente des contraintes surtout dues à la faible vitesse du processeur qui induit des temps de traitement assez longs.

- **CMUcam3 :**

Une autre carte caméra est la CMUcam3 [45], la troisième version des systèmes CMUcam (toute l'information à propos du projet est disponible sur <http://www.cmucam.org/>).

Essentiellement, la CMUcam3 est composée d'une caméra CMOS Omni vision OV6620 ou une OV7620 comme module de capture d'image, en permettant la capture d'images aux formats couleur RGB et YCbCr, 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 1Mo. Cet dispositif peut être connecté à une mote Telos de Berkeley. Afin d'obtenir des économies d'énergie, la caméra peut travailler selon trois modes d'opérations : active, idle et power down. Cependant, la consommation d'énergie de la CMUcam est bien plus grande que celle de la caméra Cyclops. Le bénéfice réel de la CMUcam est dans la rapidité de traitement.

5.2 Caméras conçues pour les réseaux de capteurs sans fil

ALOHA [46] est un capteur prototype CMOS conçu spécifiquement pour des applications de réseaux de capteurs. ALOHA intègre une représentation de l'information basée sur des événements. L'idée de base est simple: 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.

Jusqu'à présent, deux versions du capteur d'images ALOHA ont été développés :

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

Certaines évaluations pratiques ont été rapportées pour cette plate-forme. Dans [47], la latence et la consommation de puissance ont été évalués pour un nœud simple composé d'un

ALOHAim1 et d'un mote Mica2. Les auteurs montrent de la dissipation de puissance d'environ 111mW et 60.4mW pour le mote Mica2, pendant les phases de transmission et collecte d'images, respectivement. Avec ceci, les durées théoriques initialement annoncées par l'utilisation du capteur ALOHA sont baissées de plusieurs mois à quelques jours, dus à la consommation des nœuds sans fil. L'impact du nombre d'événements est aussi commenté quand un deuxième nœud, composé d'une mote Mica2dot et une ampoule à faible consommation est activée pendant la phase de capture afin d'incrémenter la qualité des images résultantes.

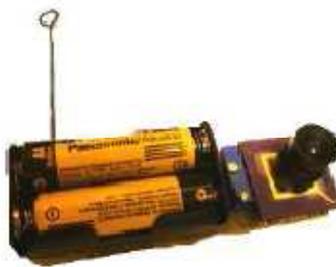


Figure 12 : ALOHA attaché à un mote Mica2

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 (en principe) 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.

6. Conclusion

Plus les contraintes imposées par les réseaux de capteur sans fils, les RCSFs de vision introduit d'autres problèmes à cause de la grande quantité de donnée que renferme la représentation d'une image.

Dans ce chapitre nous avons présenté l'importance et l'utilité d'image dans plusieurs applications se basant sur les RCSF, des différents capteurs d'images existantes et qui sont utilisés dans des travaux de recherche et les facteurs influençant sur les RCSF de vision.

Dans le prochain chapitre on va parler sur les différentes méthodes de compression d'images traditionnelles existant, puis des exemples de travaux concernant la compression d'image dans les RCSFs.

Chapitre 3 :

Compression d'image dans les réseaux de capteur sans fil

1. Introduction :

La compression de données est une activité ancienne, l'utilisation d'abréviations est une preuve. Les langues elles-mêmes utilisent des mots de longueurs variées, les plus fréquents étant les plus courts, afin de réduire la taille des phrases. La compression des images numériques, C'est elle qui a permis la diffusion des images sur Internet ou encore la propagation des appareils photos numériques. Elle constitue également la base de la compression vidéo.

L'image devient vite lourde de données, donc excessivement longue à transmettre et à traiter. Voilà pourquoi depuis quelques années, les centres de recherche en informatique dépendent de nombreuses heures sur des algorithmes de compression. Afin de limiter la taille, ou le poids d'une image, nous devons la compresser, c'est-à-dire éliminer les informations inintéressantes ou redondantes. Il existe de nos jours plus d'une vingtaine de formats de compression, spécifiquement dans la compression d'image (.gif, .jpeg, .bmp...).

2. Caractéristiques d'une image [48]

Pour comprendre comment fonctionne la compression d'image, nous devons tout d'abord savoir ce qu'est une image, quelles sont les différentes représentations informatiques, par quels moyen peut-on réduire la taille des fichiers, comment représenter les couleurs bref autant de choses qu'il est nécessaire de détailler dans une première partie.

2.1. Généralités

a. Notion de pixel

En informatique, et ce pour des raisons de gain de place, une image est composée d'un ensemble de points, appelés pixel (abréviation de PICTURE Element). Ces pixels sont regroupés en lignes et en colonnes afin de former un espace à deux dimensions. Chaque point sera représenté par ses coordonnées (X, Y), avec X l'axe orienté de gauche à droite, et Y l'axe orienté de haut en bas.

b. Représentation de la couleur

En plus de ses coordonnées planaires, un pixel se caractérise par sa pondération, appelée aussi profondeur de codage, qui représente sa couleur ou son intensité. Cette valeur peut être codée sur un nombre n différent de bits (ou octet) selon les méthodes de codage de la couleur utilisées. Les standards les plus répandus sont $n=1$ bit (noir ou blanc), $n=4$ (16 couleurs ou 16 niveaux de gris), $n=8$ bits (256 couleurs ou 256 niveaux de gris) ...

On appelle la palette de couleur, l'ensemble des couleurs que peut contenir une image. Il est fréquent de voir des images qui n'utilisent jamais certaines couleurs, il devient dès lors intéressant de limiter la palette de couleur en ne sélectionnant que la ou les couleurs utilisées réellement par l'image. Plusieurs types d'images existant les plus connues sont :

- ***Les images binaires (noir ou blanc)***

Images la plus simples, un pixel peut prendre uniquement les valeurs noir ou blanc. C'est typiquement le type d'image que l'on utilise pour scanner du texte quand celui-ci est composé d'une seule couleur.

- ***Les images en niveaux de gris***

En général, les images en niveaux de gris renferment 256 teintes de gris. Par convention la valeur zéro représente le noir (intensité lumineuse nulle) et la valeur 255 le blanc (intensité lumineuse maximale). Le nombre 256 est lié à la quantification de l'image. En effet chaque entier représentant un niveau de gris est codé sur 8 bits. Il est donc compris entre 0 et $2^8 - 1 = 255$. C'est la quantification la plus courante. On peut coder une image en niveaux de gris sur 16 bits ($0 \leq n \leq 2^{16} - 1$) ou sur 2 bits : dans ce dernier cas le « niveau de gris » vaut 0 ou 1 : il s'agit alors d'une image binaire (Noir et Blanc).

- ***Les images couleurs***

L'espace couleur est basé sur la synthèse additive des couleurs, c'est à dire que le mélange de trois composantes (par exemple (R, V, B)) donne une couleur. Un pixel est codé par trois valeurs numériques. La signification de ces valeurs dépend du type de codage choisi. Le plus utilisé pour la manipulation des images numériques est l'espace couleur Rouge, Vert, Bleu (R, V, B) (RGB en anglais). La restitution des couleurs sur écran utilise cette représentation (synthèse additive).

c. Taille et définition d'une image

Pour connaître la définition (en octets) d'une image, il est nécessaire de compter le nombre de pixels que contient l'image, cela revient à calculer le nombre de cases du tableau, soit la hauteur de celui-ci que multiplie sa largeur. La taille (ou poids) de l'image est alors le nombre de pixels que multiplie la taille de chacun de ces éléments.

Prenons l'exemple d'une image 1024 x 768, dont la couleur est codée sur 24 bits (1 octet pour les nuances de rouge, 1 pour le bleu et 1 octet pour le vert, codage True color ou RGB)

- ❖ Nombre de pixels :

$$1024 \times 768 = 786432 \text{ pixels}$$

- ❖ Poids de l'image :

$$786432 \times 3 = 2359296 \text{ octets}$$

- ❖ soit une image de $2359296 / 1024 = 2304$ Ko, ou $2304 / 1024 = 2,25$ Mo, ce qui est assez conséquent, surtout lorsqu'on veut transmettre l'image...

d. Types de compression

L'enjeu de la recherche sur la compression d'image est de trouver un moyen de diminuer la taille d'une image, tout en essayant de limiter la dégradation due à la compression.

Il existe deux grandes familles d'algorithmes de compression : compression sans perte (non-destructrice) et compression avec perte (destructrice).

- ***Compression sans perte***

Elle restitue une image totalement identique à l'original, il n'y a donc aucune perte d'information. Cette famille d'algorithmes est essentielle dans nos ordinateurs, la totalité des programmes d'archivage sont bâtis sur cette technique de compression sans perte d'information. Le principe courant de ces programmes est de réduire les séquences d'information redondantes.

- *Compression avec perte*

La méthode destructrice délivre après compression une image différente, elle contient beaucoup moins d'information que l'original, certains détails auront été éliminés lors du codage. Cette modification est plus ou moins visible, selon le degré de compression. L'attrait de cette famille est qu'elle peut obtenir un rendement formidablement grand. Cependant, on ne peut utiliser ce genre de compression que pour des sons, vidéos et images, mais pas sur des fichiers ou sur du texte puisque ceux-ci ne doivent subir aucune dégradation.

On appelle ratio le rapport entre la taille d'une image non compressée et la taille de la même image, mais compressée. Le taux de compression étant l'inverse du ratio, un ratio de $n : 1$ est équivalent à un taux de $1/n$. Ainsi, avec une image de 2 Mégaoctets non compressée, et avec un algorithme de ratio 4 : 1 (ou par équivalence, avec un taux de compression de $1/4$), nous devrions obtenir une image compressée de taille 0,5 Mo ($2 \times 1/4 = 0,5$).

- e. Différentes formes d'images*

Nous avons vu auparavant qu'une image était constituée d'un ensemble de points, nommés pixels, de position et de couleur différente. C'est ce qu'on appelle une image du type « bitmap », ou en français « tableau de données binaires ». Cependant, il existe une autre forme d'image, appelée image **vectorielle**.

Les images vectorielles sont des représentations géométriques (cercle, rectangle, droite, etc...). Cette propriété spécifique aux images vectorielles, nous permet de coder une image entière en formules mathématiques (exemple : un cercle est défini par son centre et son rayon, le rectangle lui peut être déterminé par ses deux points extrêmes...). C'est le processeur qui sera ensuite chargé de "traduire" ces formes en informations interprétables par la carte graphique.

Le grand avantage des images vectorielles est le suivant : puisque ces images sont constituées d'entités mathématiques, il est possible de leur appliquer des transformations géométriques. Lorsqu'on applique un grossissement sur une image du type bitmap, on s'aperçoit que l'image n'est plus nette, et selon le degré de grossissement, on peut observer les pixels. Avec les images vectorielles, la précision ne dépend pas du facteur d'agrandissement.

Les images vectorielles, appelées aussi « Cliparts » permettent donc de définir une image avec très peu d'information, d'où le gain de place. En comparaison, une image vectorielle de 1m sur 1m a le même poids qu'une image du type Bitmap de taille 0,1mm sur 0,1mm. Cependant, ce format ne convient pas pour des images trop complexes, comme des photographies par exemple.

3. Codages et compressions d'une image

3.1 Le format RLE

Le principe de compression de RLE est assez simple à mettre en œuvre. Il repose sur le fait que dans une image, il existe de nombreuses répétitions d'un même pixel, ou d'une même séquence de pixel, tous juxtaposés. Ainsi, au lieu de coder chaque pixel d'une image, le RLE propose de coder d'une part le nombre de répétitions, et d'autre part la séquence ou l'octet à répéter. Lorsqu'il s'agit d'une répétition de séquence, on ajoute entre le nombre de répétitions et la séquence, un octet représentant le nombre d'octets de la séquence. Si la taille de celle-ci est impaire, alors on ajoute un octet nul (00) à la fin.

Ce système de compression est assez bon dans certain cas (images monochromes ou 256 couleurs), mais il s'avère très coûteux pour des images dont la couleur est codée sur 16 ou 24 bits... Il est à noter qu'il existe plusieurs variantes de ce codage, certains types encodent l'image en décelant des séquences redondantes selon les lignes, les colonnes, ou même en zigzag.

3.2 La compression RL

La compression RL est une variante de la compression RLE. Au lieu de coder la répétition d'une même séquence sur une ligne, comme le fait le codage vu précédemment, la compression RL utilise ce qu'on désigne par codage par zone. Le procédé est simple : le programme tente de déterminer dans toute l'image des rectangles dont les pixels sont identiques. Ainsi, il suffira de coder le nombre de répétitions du rectangle à réaliser lors de la décompression.

La difficulté majeure de ce mode de compression est de trouver des rectangles identiques dans l'image. Les algorithmes actuels sont récursifs, c'est-à-dire qu'ils partent d'une taille de rectangle $n \times m$, avec n et m maximum, puis ils font tendre n et m vers 1 (1

pixel). Le problème est que cette manière de procéder est excessivement coûteuse en temps, lors de la compression, mais aussi lors de la décompression puisqu'il faut redessiner (presque) complètement l'image.

3.3 Le codage de Huffman

En 1952, Huffman créa une nouvelle méthode de compression appelée compression à arbre. Le principe est simple, une image est formée par de nombreux caractères différents, mais certains reviennent plus souvent que d'autres. Aussi, l'algorithme de Huffman établit un arbre contenant les signes, les symboles les plus fréquents, ainsi que leur fréquence d'apparition. Puis, à chaque caractère est assigné un code. Le signe le plus souvent utilisé, placé à la base de l'arbre, reçoit le code le plus court. Le caractère le moins fréquent aura donc le plus long code binaire. Par conséquent, la suite finale des mots codés sur des longueurs variables (petite pour les symboles courants et longs pour les mots peu fréquents), sera mathématiquement plus petite qu'avec un codage sans compression.

Lors de la compression, l'algorithme écrira le tableau des équivalences Code-Séquence, le fichier contiendra donc non plus les séquences binaires de l'image, mais les différents codes attribués par le programme de compression. Pour décompresser, il suffira de remplacer tous les codes par les séquences équivalentes (présentent dans l'entête du fichier).

Cette méthode de codage donne de bon taux de compression, principalement sur des images monochromes, pour les fax par exemple, mais il est aussi utilisé dans la compression JPEG, et ceux, pour tous les types d'image.

3.4 La compression LZW

Le système de compression d'image le plus utilisé à travers le monde reste la compression LZW (acronyme de ses inventeurs Lempel-Ziv-Welch). Cet algorithme utilise, comme la compression de Huffman vue précédemment, un tableau, un dictionnaire pour réaliser une compression du type non-destructrice. Contrairement au codage précédent, la compression LZW n'encode pas dans le fichier le dictionnaire, celui-ci sera reconstruit lors de la décompression. Le LZW est un dérivé du codage LZ. En 1978 il a été créé le compresseur LZ78 spécialisé dans la compression d'images et de fichiers binaires. Puis en 1984, Terry Welch modifia le format LZ78 pour pouvoir l'utiliser dans les contrôleurs de disques durs, le format LZW est né.

Ce type de compression possède de nombreux avantages. Le principal étant la reconstruction du dictionnaire pendant la décompression, ce qui permet de ne pas le coder dans le fichier. Cette méthode de compression peut obtenir des ratios supérieurs à 2 : 1, et sans aucune altération de l'image.

3.5 Le mode de compression JPEG

Le problème majeur de tous les codages précédents était que ces méthodes de compression ne délivraient de bon taux de compression que pour des images ciblées, c'est-à-dire que pour des images vectorielles ou pour des images monochromes bref, pas pour des photographies, ou toute autre image où les détails sont nombreux. Ce pourquoi, en 1982, une réunion entre un groupe d'experts en photographie et l'ITU-T abouti sur la création du JPEG (comité conjoint d'experts de la photographie).

A la différence du codage LZW, la compression JPEG est une compression avec pertes, ou destructrice, et qui par conséquent peut obtenir des ratios de 20 : 1 à 25 : 1, sans perte notable de qualité.

1. Principe de compression

Le principe de compression du JPEG est constitué de plusieurs étapes :

- ❖ la préparation
- ❖ la transformation
- ❖ la quantification
- ❖ l'encodage

Prenons l'exemple suivant : une image 640x480 RGB 24 bits par pixel.

➡ La première étape consiste à transformer le codage de la couleur RGB, de l'image bitmap de départ, en un codage YIQ, avec les combinaisons linéaires:

$$Y = 0.299 R + 0.587 G + 0.114 B$$

$$I = 0.596 R - 0.275 G - 0.321 B$$

$$Q = 0.212 R - 0.523 G + 0.311 B$$

Mais pour gagner un peu de place, on arrondit généralement les coefficients de la luminance Y et de la chrominance (I et Q), pour obtenir :

$$Y = 0.3 R + 0.59 G + 0.11 B$$

$$I = 0.6 R - 0.28 G - 0.32 B$$

$$Q = 0.21 R - 0.52 G + 0.31 B$$

Les résultats sont ensuite regroupés en matrice 640x480. Nous obtenons donc trois matrices de taille 640x480 représentant la luminance et la chrominance (2 matrices). Une des astuces du code JPEG est d'éliminer les variations de la chrominance entre deux pixels. En effet, l'œil de l'être humain n'est que peu sensible à ces minuscules variations. Aussi, l'algorithme rétrécit la taille des matrices de I et Q, en effectuant la moyenne des deux composantes de la chrominance, dans des carrés de 2x2 pixels. Nous obtenons alors des matrices de chrominance de 320x240.

➡ La deuxième action est ce qu'on appelle la transformation. L'algorithme de compression découpe premièrement les matrices en blocs de 8x8 pixels, et leur applique ensuite la fonction DCT (Discrete Cosinus Transform). Cette fonction DCT est une transformation en série (Fourier), qui délivre une représentation non plus spatiale, mais dans le domaine fréquentiel. En effet, la ligne et la colonne de la matrice représentent les axes X et Y dans le domaine spatial, et la valeur d'une case particulière représente la valeur de Z. Nous raisonnons donc en 3 dimensions. La transformation de la matrice donne une nouvelle matrice contenant cette fois-ci, les différentes puissances spectrales pour chaque fréquence. L'élément (0,0) représente la valeur moyenne du signal.

➡ La quantification réside dans le fait que l'algorithme attribue à chaque fréquence, à chaque cellule de la matrice 8x8 pixels, un coefficient de perte. L'algorithme annulera ou diminuera les hautes fréquences qui, représentent les plus petits détails de l'image. L'atténuation de l'amplitude des différentes fréquences est déterminée par le ratio demandé par l'utilisateur.

➡ Enfin, la matrice obtenue est linéarisée en zigzag, selon le codage RLE, et est compressée avec la méthode de Huffman.

La compression JPEG permet néanmoins d'obtenir de très bon taux de compression, surtout pour des images complexes. Il est à noter qu'il existe une variante du codage JPEG sans aucune perte, qui est principalement utilisé dans l'imagerie médicale, et qui possède un ratio de 2 : 1, au maximum.

3.6 Les méthodes de compression récentes

3.6.1 La compression fractale

Dans les années 1950, un mathématicien nommé benoît Mandelbrot, découvrit ce qu'on appelle aujourd'hui, les fractales. La représentation graphique des solutions délivrées par un algorithme spécifique, donne une image constituée de répétitions infinies d'une même forme, d'un même motif.

Puisqu'une image fractale est une répétition d'un motif plus ou moins rétréci, plus ou moins transformé (translation, rotation, ...), Barnsley eût l'idée de rechercher dans n'importe quelle image, des formes similaires. Ainsi, au lieu de coder toute l'image, la compression fractale n'encode plus que le ou les motifs, ainsi que les transformations à réaliser.

En 1988, on annonçait des taux de compression allant jusqu'à 1/10 000. Mais ce taux de compression imaginaire a été réalisé en 100 heures de travail, avec un ordinateur biprocesseur (donc deux processeurs), et toute une équipe de chercheurs. Aujourd'hui, et malgré l'avancée considérable des microprocesseurs, la compression fractale peut compter sur des taux allant de 1/4 à 1/100, pour des temps raisonnables.

3.6.2 La compression par ondelettes

Il y a quelques années encore, les chercheurs travaillaient à l'élaboration d'un nouveau format, le JPEG 2000, utilisant une nouvelle et prometteuse méthode de compression, appelée compression par ondelettes.

La théorie des ondelettes a été inventée par le mathématicien hongrois Alfred Haar dans les années 1910. Une ondelette est une transformation de fonction, comme Laplace ou Fourier, qui balance principalement dans un intervalle restreint.

Nous pouvons voir sur la figure 13 que, contrairement à la fonction Cosinus (en vert), l'ondelette possède une amplitude variable, et est quasiment nulle en dehors de l'intervalle

[-4,4]. Cette variation très locale de la fonction permet néanmoins de savoir précisément ce qui se passe en n'importe quel endroit du signal original (non transformé).

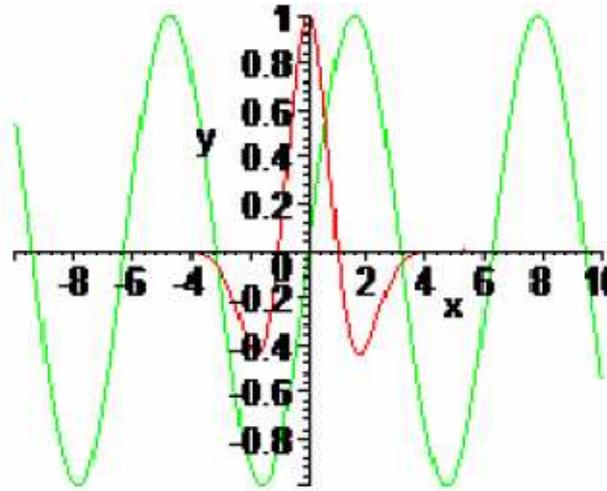


Figure 13 : Représentation d'une sinusoïde et d'une ondelette

Contrairement au format JPEG, qui décompose une image en blocs de 8x8 pixels, la compression JPEG 2000 transforme chaque ligne horizontale en un signal, qui sera ensuite transformé en somme d'ondelettes. En effet, la variation de couleur et d'intensité de chaque pixel d'une ligne peut être assimilée à la variation de deux signaux. Chaque signal sera ensuite directement transformé en une série d'ondelettes, répétées en différents endroits, et à différentes échelles, pour que la somme décrive le plus exactement le signal original. L'algorithme éliminera les variations les plus infimes pour compresser encore d'avantage l'image.

Une variante de ce format a aussi vu le jour, le MJ2 (Motion JPEG 2000). Celui-ci utilise aussi la compression par ondelettes, mais pour des images en mouvement c'est à dire pour la vidéo numérique.

4. Traitement d'images dans les réseaux de capteurs sans fil :

L'unité de communication est l'un des composants électroniques qui consomme beaucoup d'énergie dans un nœud capteur sans fil [1] le moyen d'économiser de l'énergie est de réduire la quantité de données à transmettre, une des solutions est la compression.

La compression a un coût d'énergie qui peut être très élevé. Des travaux [49] ont montré en effet que des algorithmes bien connus comme JPEG ou JPEG2000 a un coût d'énergie bien supérieur au gain qu'ils amènent sur l'unité de communication. Autrement dit, le capteur d'image consomme plus d'énergie en envoyant des images compressées que des images non compressées, Le problème est de disposer d'un algorithme de compression de données qui soit peu gourmand en énergie. Une longue liste d'algorithmes de compression est aujourd'hui disponible [48], Dont plusieurs concernant La compression d'images.

Nous pouvons voir aujourd'hui une grande évolution dans le domaine de la compression. Cependant, la limitation de ressources des nœuds de capteurs, comme la capacité de stockage et la vitesse des processeurs, rend inapplicables la plupart des algorithmes de compression existants utilisés dans l'informatique traditionnelle [50].

Pour que la compression de données soit rentable, il faut que le traitement des données et la transmission des données compressées coûte moins d'énergie qu'un scénario en absence de compression. En effet, quelques travaux [49], [51] ont démontré que la complexité de certains algorithmes de compression conduisent à des consommations d'énergie plus importantes que la transmission simple d'une image sans aucun traitement. Un travail qui considère le compromis entre consommation énergétique par traitement et par communication a été présenté dans [52]. Un autre [49] qu'ils évaluent plusieurs algorithmes de compression traditionnels en analysant le compromis entre énergie consommée et traitement de données.

Plusieurs propositions sont basées sur la transformation en ondelettes. Ceci est principalement dû au fait que les ondelettes permettent la décomposition des images en plusieurs niveaux de résolution. Des exemples d'application d'ondelettes sur réseaux de capteurs de vision sont donnés dans [51], [53], [54], [55], [56].

4.1. Classification des algorithmes de compression d'image :

La compression d'image dans les réseaux de capteur sans fil est classifiée en deux types : compression locale et compression distribuée.

4.1.1 Compression locale :

Dans ce type la compression est fait au niveau de nœud source c'est-à-dire sans distribution de la charge de traitement des données avec d'autres nœuds. Il existe plusieurs algorithmes de compression qui peuvent fournir un taux de compression plus élevé mais ne

sont pas applicable dans les réseaux de capteur sans fils en raison de limitation des ressources de nœud capteur. Le travail [49] qui présente une plate forme pour évaluer les performances de plusieurs algorithmes de compression traditionnel (JPEG2000, SPITH, DCT, SS, JPEG), les résultats montre que le coût d'énergie de calcul est supérieur au coût d'énergie de transmission de l'image sans compression pour JPEG2000, SPITH, DCT, JPEG. Le coût d'énergie de traitement est le coût d'implantation d'algorithme de compression dépendent aussi des caractéristiques de matériel. La plate forme de [49] est basée sur un microcontrôleur qui est plus rapide alors plus gourmand en énergie que d'autres qui sont moins rapide.

Des travaux sur la compression locale pour les réseaux de capteur d'image sans fil sont proposés en [51], [53], [57], [58].

Le problème de la compression locale est que le nœud source consomme beaucoup d'énergie (la compression + la transmission) par rapport aux nœuds intermédiaires qui font seulement la transmission, donc la durée de vie de nœud source va diminuer implique la diminution de la durée de vie de réseau.

Quelques systèmes de compression locaux proposés pour les réseaux de capteurs de vision sont présentés ci-dessous.

❖ *Schéma basé sur le codage SPIHT (Set Partitioning In Hierarchical Tree)*

L'une des premières propositions pour la compression d'images dans les réseaux de capteurs a été introduite dans [53]. Ils ont proposé un schéma basé sur le codage SPIHT, des blocs de données sont générés par relation parent-enfant de coefficients d'ondelettes. L'algorithme fonctionne comme suit : premièrement, l'image capturée est décomposée en multiple résolutions en appliquant une transformée en ondelettes discrète. Puis les coefficients d'ondelettes sont regroupés en fonction de leur relation parent-enfant. Chacun de ces groupes est codé indépendamment par l'algorithme SPIHT.

Les expériences ont été réalisées avec un processeur Intel StrongARM SA 1110 et un émetteur radio LMX3162. Les résultats montrent des économies d'énergie en termes de traitement de données.

❖ *Compression locale par JPEG*

La technique de compression d'image la plus répandue de nos jours est le standard JPEG. Il est basé sur un découpage de l'image en blocs de 8×8 pixels et sur la transformée en cosinus discrète (DCT). Les blocs sont ensuite quantifiés et codés avec RLE et Huffman.

De toutes les étapes de l'algorithme JPEG, c'est la DCT qui coûte le plus en calculs. Il faut optimiser cette opération, pour la rendre plus rapide (on parle donc de DCTs rapides) et donc applicable sur des composants électroniques plus limités que les ordinateurs de nos jours, comme les appareils photo-numériques, les PDAs, et, bien sûr, les nœuds de capteurs d'image sans fil. Un exemple récent est l'approche utilisée dans [57], qui ont adopté l'algorithme JPEG en calculant la DCT dans un mode à virgule non flottante. Il calculait le nombre de bits minimums pour représenter les parties entières et décimales des valeurs réelles.

Par ailleurs, pour diminuer la quantité de calculs à faire sur chaque bloc de coefficients DCT, dans [58] on propose l'application d'un algorithme connu comme Triangular JPEG (TJPEG). A lieu de traiter un bloc de coefficients DCT entier de $k \times k$ coefficients ($k = 8$ pour le cas de JPEG traditionnel), ils vont sélectionner une région réduite et représentative du bloc. La méthode coûte donc moins cher en énergie.

❖ *Compression locale par JPEG2000*

JPEG2000 est un algorithme de compression d'images basé sur un processus très complexe qui inclut une transformée en ondelettes dyadique, une allocation de bits, une quantification et un codage entropique. Dans [51] ont introduit une technique de faible consommation énergétique qui incorpore le standard JPEG2000 pour la compression d'images depuis un nœud caméra sans fil. Il est supposé qu'un nœud source a une connaissance du nombre de nœuds intermédiaires entre lui et le puits, et chaque nœud est capable de connaître son état de batterie. L'algorithme travaille comme suit :

Le nœud de capteur compresse l'image capturée en appliquant la compression par ondelettes, puis, il calcul un taux de compression et une dissipation d'énergie. Une nouvelle compression est donc réalisée pour calculer des nouveaux taux de compression et dissipation d'énergie. Ces valeurs sont comparées avec les anciennes. Ces étapes sont répétées jusqu'à un certain nombre d'itérations.

On peut voir que cette méthode pousse à compresser l'image plusieurs fois et on ne peut pas voir comment la source va économiser l'énergie. Cette méthode vise à optimiser l'énergie dans les nœuds de transit mais elle n'est pas bonne pour la source.

4.1.2 Compression distribuée :

La nature distribuée des réseaux de capteurs a permis d'avoir des approches qui considèrent la distribution du traitement de données entre plusieurs capteurs. Deux types d'algorithmes de compression distribués, Le premier est basé sur la corrélation existante entre deux ou plus images principalement quand elles contiennent des scènes voisines, et le deuxième est basé sur la distribution du processus de compression d'une image à travers plusieurs nœuds.

❖ Compression distribuée d'images corrélées :

Plusieurs approches proposées pour le traitement collaboratif distribué d'images sur réseaux de capteurs sans fil sont basées sur le théorème de **Slepian** et **Wolf** [59] qui consiste à coder les images corrélées indépendamment mais décodé ensemble. Des travaux avec ce type de compression distribué ont été proposés en [60], [61].

❖ Compression basée sur la distribution du processus de compression :

Le travail proposé par HUAMING WU et ALHUSSEIN A. ABOUZEID [55] montre deux stratégies de compression distribué d'image qui sont proposées par ce travail.

- ***Méthode 1 : Méthode de transformée en ondelettes parallèle (Division par ligne/colonne)***

Dans cette méthode, nous considérons le schéma de partitionnement des données proposé on [62], mais dans notre cas appliqués à un réseau sans fil ici, l'énergie consommée pour l'échange de données est considérée.

Figure 14 illustre le schéma de partitionnement des données pour un seul niveau de décomposition on ondelette. Au début, les données sont divisées on n blocs R_1, R_2, \dots, R_n où chaque bloc se compose d'une ou plusieurs rangées. Puis, chaque nœud exécute une transformation en ondelette discrète sur R_i . Une fois que la transformation en ondelette est fait sur tous les rangées, un nœud rassemble les résultats intermédiaires Q_1, Q_2, \dots, Q_n et

divise les résultats en m blocs I_1, I_2, \dots, I_m . Puis chaque nœud applique une transformation en ondelette discrète sur I_i .

En conclusion, un nœud collecte les résultats J_1, J_2, \dots, J_m de la transformation en ondelette discrète appliqué deux fois. Bien qu'il n'y ait aucune perte de qualité comparée à la transformation en ondelette centralisé, il faut noter que cette approche de division exige une communication très élevée en raison de grand nombre d'échange de données.

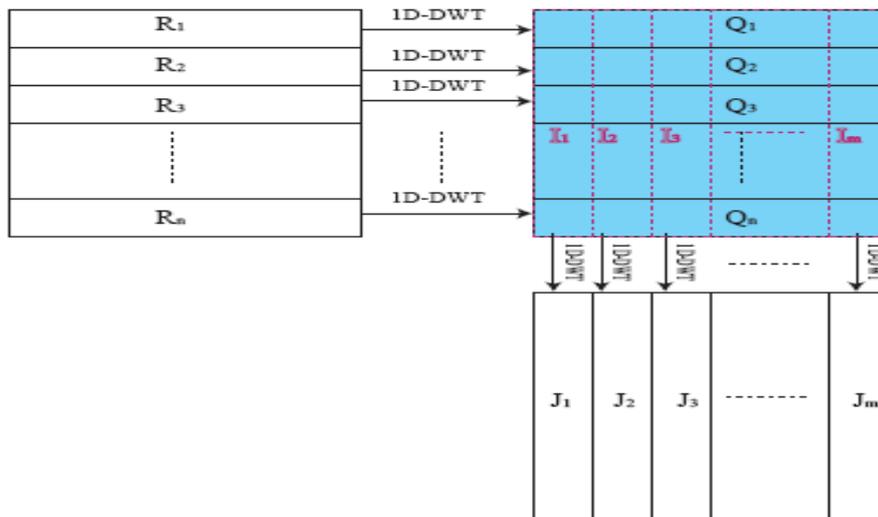


Figure 14 : Approche traditionnelle de division des données

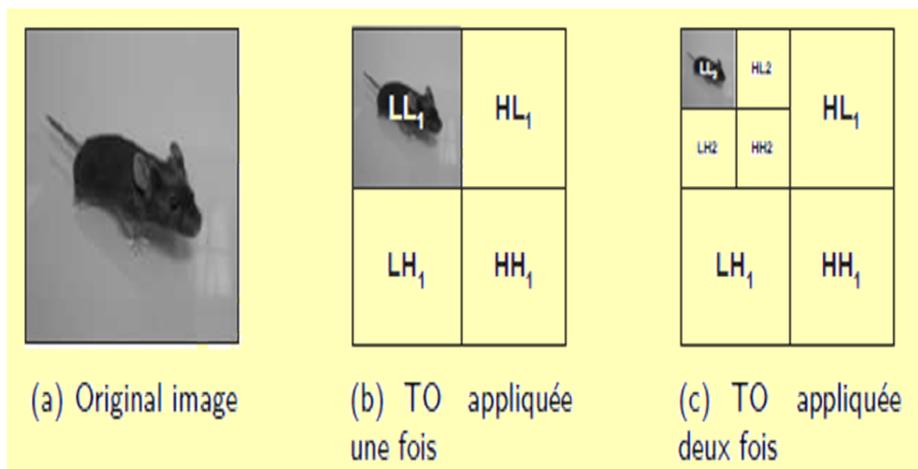


Figure 15 : Transformation en ondelette

Un exemple de la compression distribuée d'image employant quatre nœuds à chaque niveau est montré dans la figure 16.

Après la réception d'un message d'un nœud source S , le nœud $c1$ sélectionne un ensemble de nœuds $P1$ qui participeront à la transformation distribué en ondelette puis informe S . S divise les données d'image par des rangées et les transmet à $P1$ ($p11, p12, p13$ et $p14$). Ces nœuds appliquent l'algorithme de transformation en ondelette sur leurs données reçues puis envoient les résultats intermédiaires à $c1$. $C1$ combine les résultats et les divise par des colonnes puis les transmet encore à $P1$. Ces nœuds traitent ces données et envoient les résultats (niveau 1 dans la figure 16) au prochain nœud $c2$. Après la réception des résultats, $c2$ choisit une partie des résultats (correspondant à LL figure 15 (b)) et le divise en pièces pour $P2$ ($p21, p22, p23$ et $p24$). $C2$ code et envoi les autres parties des données reçues (correspondant aux HL et au HH dans la figure 15 (b)) directement au prochain nœud $c3$. Les nœuds dans $P2$ ($p21, p22, p23$ et $p24$) envoient directement leurs résultats traités (correspondant au niveau 2 de la décomposition en ondelette comme il est montré dans la figure 15(c)) à $c3$ après avoir appliqué la transformation en ondelette deux fois.

Ce processus peut continuer sur $c3$ et ses nœuds suivants jusqu'à ce que l'image compressée arrive au nœud destination.

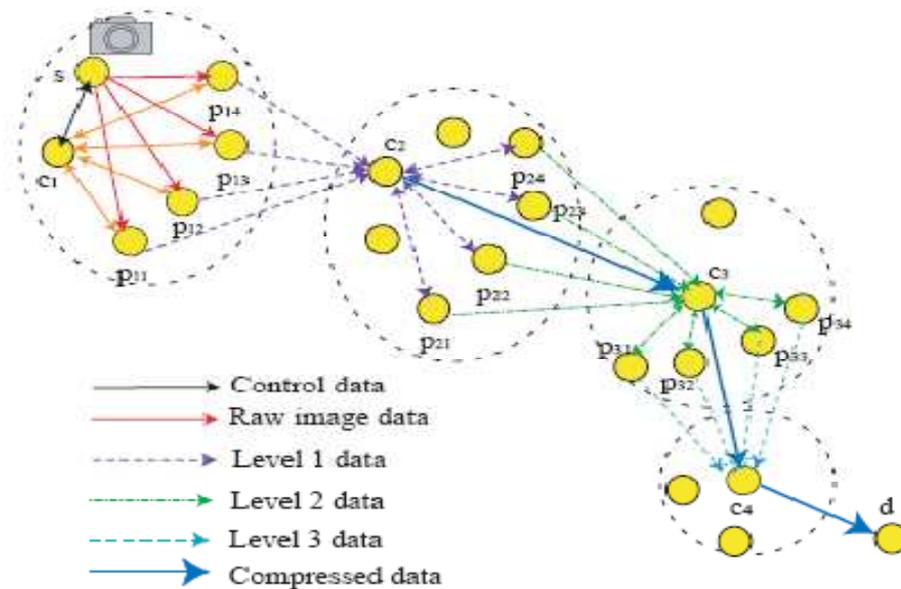


Figure 16 : Méthode 1 compression d'image distribuée basée sur la transformation en ondelette

- **Méthode 2 : Partitionnement (de carrelage)**

Le concept du partitionnement d'image est appliqué généralement dans un environnement de basse mémoire. Dans ce cas, l'image est divisée en blocs, et chaque bloc est

envoyé à un nœud pour faire la transformation en ondelette discrète indépendamment. Une fois qu'un nœud termine sa transformation en ondelette, il envoie le résultat à un nœud central. Le traitement des blocs indépendamment mène à une perte d'information. La perte de qualité et d'information est petite quand le nombre de bloc est petit.

La figure 17 montre un exemple de compression distribuée d'image employé quatre blocs et trois niveaux de décomposition en ondelette semblable à la méthode 1. Après la réception d'un message de nœud source S , le nœud $c1$ choisit un ensemble de nœuds $P1$ qui participera à la transformation distribuée en ondelette puis informe S . S divise les données d'image en blocs et les transmet à $P1$. Chaque nœud applique l'algorithme de transformation en ondelette discrète sur leurs données reçues puis envoie les résultats (données de niveau 1 dans la figure 17) au prochain nœud $c2$. Pour chaque bloc, $c2$ choisit une partie des résultats (correspondant à LL dans la figure 15(b)) et l'envoie à un de son ensemble de nœuds de traitement ($p21, p22, p23$ et $p24$). Ainsi, $p21, p22, p23$ et $p24$ obtiennent le LL correspondant aux différents composants des blocs. $C2$ combinera également les autres parties des données reçues (correspondant aux LH, HL et HH dans la figure 15(b)) de toutes les blocs puis les codent et les envoient directement au prochain nœud $c3$. De cette façon, l'image finale compressée atteint le nœud destinataire.

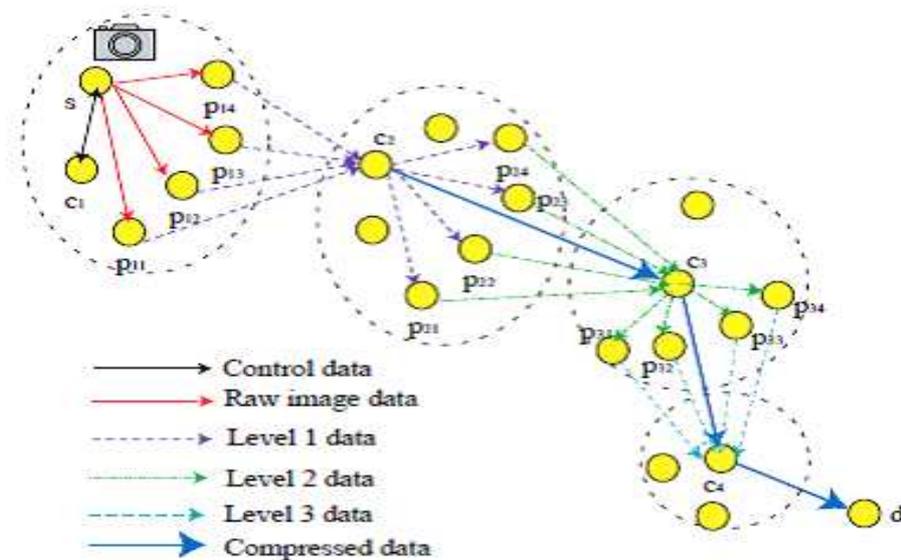


Figure 17 : Méthode 2 compression d'image distribuée basée sur la transformation en ondelette

5. Conclusion

Les besoins d'applications pour les réseaux de capteurs d'images deviennent de plus en plus nombreux. La demande aujourd'hui concerne notamment les applications environnementales. Avec la contrainte de la limitation des ressources des nœuds, des mécanismes de traitement et de transmission d'images efficaces en énergie restent à développer. Dans ce chapitre, nous avons classé les travaux qui traitent essentiellement le traitement des images sur les réseaux de capteurs sans fil.

Dans le domaine du traitement d'images, on remarque des efforts dans plusieurs travaux pour adapter les algorithmes de compression d'images de l'informatique traditionnelle, comme JPEG ou JPEG2000, aux contraintes particulières des réseaux de capteurs sans fil. Cependant, même si les algorithmes classiques présentent des bons performances, des expériences sur des capteurs démontrent que ces algorithmes sont beaucoup trop coûteux en énergie, plus coûteux que la transmission d'une image sans compression. Plusieurs adaptations ont été proposées pour rendre ces algorithmes moins gourmands en énergie.

Chapitre 4 :

Approche de compression d'image proposée

1. Introduction

La transmission d'image dans un réseau de capteur sans fil consomme beaucoup d'énergie à cause de grande quantité d'information qu'elle contient, donc il faut une stratégie pour minimiser l'énergie consommée, l'un de ces méthodes est la compression d'image.

Plusieurs algorithmes de compression d'image existant (JPEG, JPEG2000,...), mais qui ne sont pas adaptés aux capteurs sans fil à cause de limitation des ressources, en particulier la vitesse de calcul et la vitesse de transmission. Des performances élevées en termes de vitesse de traitement et de transmission impliquent une consommation d'énergie élevée. De manière générale, il est souhaitable que la durée de vie de la batterie de nœuds soit la plus grande possible, donc les différentes unités qui composent un nœud sont généralement très limitées en termes de ressources et de performance pour que leur consommation d'énergie soit extrêmement faible.

Dans la prochaine section on présente la transformation en ondelette discrète que notre idée est basée sur elle.

2. Transformée en ondelettes d'une image

La transformée en ondelettes discrète (TO) [63] est une opération qui décompose un signal (une série d'échantillons numériques) en deux parties par projection sur un filtre passe-bas L et un filtre passe-haut H. La partie résultant du filtrage passe-bas représente une approximation du signal d'origine à la nouvelle résolution (ou échelle), celle résultant du filtrage passe-haut représentant les détails perdus entre les deux résolutions.

Puisqu'une image est typiquement un signal en deux dimensions, une TO dyadique est réalisée, en appliquant d'abord les filtres L et H sur les échantillons ligne par ligne, puis en réappiquant les mêmes filtres sur les échantillons résultants, mais colonne par colonne cette fois-ci. Au final, l'image est divisée en 4 parties, les sous-images LL, LH, HL et HH comme présenté sur la figure 18(a). La sous-image LL fournit une version à l'échelle $\frac{1}{2}$ de l'image d'origine, LH, HL et HH représentant les détails perdus respectivement dans les directions horizontale, verticale et diagonale. La TO peut être répétée sur LL pour obtenir plusieurs niveaux de résolution. La figure 18(b) présente une image décomposée en trois niveaux de résolution.

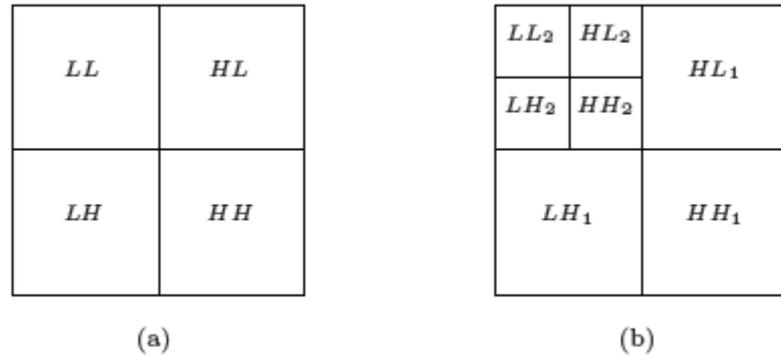


Figure 18 : La TO dyadique appliquée une fois (a) et deux fois (b).

La transformation de l'image en une représentation multi résolution fournit à la source les moyens de préparer des paquets de données de différentes priorités. En fait, l'image sera divisée en p résolutions si la TO est appliquée $(p - 1)$ fois, ce qui donne au moins p niveaux de priorité. La plus petite résolution (la résolution 0), qui est représentée par la sous-image LL_p , est la plus importante. Cette sous-image doit être transmise jusqu'au puits de manière fiable pour que l'utilisateur puisse reconstruire une version de l'image d'origine avec un niveau de qualité minimum garanti. Quelques informations supplémentaires doivent obligatoirement être connues du décodeur, il s'agit des informations relatives au format de l'image proprement dit (taille de l'image, nombre de plans, nombre de bits par pixels, nombre de niveaux de résolution, etc. . . .). Elles sont regroupées dans ce qu'on appelle les informations d'entête de l'image. Si la source est programmée pour transmettre des images toujours au même format, le récepteur connaîtra implicitement ce format et il n'aura pas de problèmes pour décoder l'image. Sinon, l'entête de l'image devra aussi être transmis de manière fiable jusqu'au puits. Les données de la résolution 0 de l'image, et si besoin celles de l'entête de l'image, seront donc placées dans des paquets ayant la priorité la plus élevée (la priorité 0). Les données associées aux autres niveaux de résolution ont une importance qui décroît de la résolution 1 à $(p - 1)$.

Pour appliquer la TO, nous avons adopté les filtres 5-3 de Le Gall [63], qui ont des coefficients rationnels. Le filtre passe-bas est donné par :

$$f_l(z) = -1/8 \cdot (z^2 + z^{-2}) + 1/4 \cdot (z + z^{-1}) + 3/4$$

Et le filtre passe-haut par :

$$f_H(z) = -1/2 \cdot (z + z^{-1}) + 1$$

Cette ondelette a été adaptée pour opérer spécialement dans l'espace des valeurs entières. Les valeurs obtenues en sortie des filtres sont des arrondis à l'entier le plus proche. La quantité de données de l'image reste donc la même après transformation.

Les filtres 5/3 sont particulièrement appréciables dans les applications contraintes par l'énergie car leur implantation fait intervenir des instructions simples, des additions et des décalages de valeurs entières, et non des multiplications et des divisions. Ils sont donc moins gourmands en énergie que des filtres à coefficients non rationnels, c'est la raison pour laquelle ils ont été choisis.

3. Solution proposée

Notre idée consiste à distribuer la compression, et en particulier distribué la transformation en ondelette discrète pour minimiser l'énergie consommée par le nœud source, c'est-à-dire :

Tant que le processus de la transformation en ondelette est récursif, on a supposé que tous les nœuds capteurs sont chargés de faire la TO, puis on a distribué la répétition de la transformation entre les nœuds capteurs. Autrement dit, dans la méthode centralisé le nœud source capte une image puis il applique la transformation en ondelette discrète n fois (nombre de niveaux de résolution) dépendant de l'application montré par la figure 19 donc il va consommer une énergie de traitement pour la TO égale à :

$$\text{Énergie de TO d'image originale} + \dots + \text{énergie de TO d'image originale} / 4^{n-1}$$

Par contre, notre méthode consiste à distribuer la charge du traitement de la TO qui se faite au niveau de nœud source avec plusieurs nœuds capteurs c à d la première transformation discrète en ondelette se fait au niveau de nœud source, la deuxième transformation est faite dans un autre nœud et ainsi de suite (voir la figure 20).

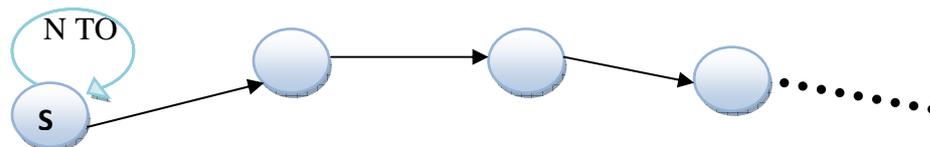


Figure 19 : Transformation en ondelette centralisée

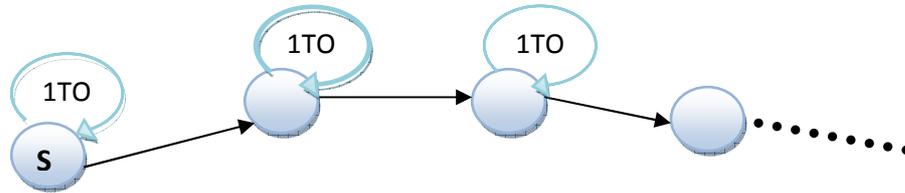


Figure 20 : Transformation en ondelette distribuée

Par exemple, si on veut transmettre une image avec trois niveaux de résolutions, dans la méthode centralisée la transformation est faite au niveau de nœud source trois fois.

Alors que dans notre solution la charge de transformation est distribuée avec les autres nœuds capteurs, c à d la première transformation discrète en ondelette se fait au niveau de nœud source, la deuxième transformation est faite dans le nœud suivant et la troisième transformation dans un autre nœud.

Voici un algorithme qui montre notre idée :

Tous les nœuds ont la même capacité en calcul et de transmission

- 1) Le nœud destination **D** envoie un message au nœud source **S** pour capter une image en précisera le niveau de décomposition (niveau de résolution) **x**.
- 2) Le nœud source capte l'image après recevoir de message.

$L=0$.

- 3) Tant que $((L < x) \text{ et } (\text{nœud} \neq D))$ faire
 - Transformer l'image; $l=l+1$.
 - Transmettre les données après transformation au nœud suivant en suivant un algorithme de routage.

Fin tant que.

- 4) Transmettre les données compressées en suivant un algorithme de routage à **D**.

4. Présentation du simulateur NS2

Le simulateur réseau NS (Network Simulator) [64] est un simulateur à événements discrets orienté objet, basée sur le simulateur réseau REAL [65]. Au départ, la version 1.0 de NS a été développée au Laboratoire National de Lawrence Berkeley (LBNL) par le groupe de recherche réseau. Son développement fait maintenant partie du projet VINT (Virtual InterNetwork Testbed) sous lequel la version 2.0 est sortie.

Le simulateur NS est un simulateur à événements discrets qui permet d'exécuter tout type de scénarios sur des topologies définies par l'utilisateur. Il permet la description et la simulation de réseaux IP. Le réseau est représenté (modélisé) par ses sources de trafic (applications), ses protocoles (UDP, TCP), ses routeurs (avec leurs files d'attente) et les liens qui les relient. Le réseau est ensuite simulé, ce qui produit des traces et des statistiques. Des outils périphériques permettent l'animation du réseau (NAM : Network Animator) ou la conversion vers d'autres outils (comme par exemple xgraph pour dessiner des courbes).

4.1. Composition d'un modèle :

Les modèles de simulations de NS sont réalisés à l'aide du langage C++ pour ce qui concerne le fonctionnement interne des composants du réseau et en tcl pour ce qui concerne la description du réseau à simuler. Enfin plus précisément, le langage est Otcl qui est la version objet de tcl.

Un modèle de réseau en ns est constitué :

- De nœuds de réseau : endroits où est généré le trafic, ou nœuds de routage.
- De liens de communication entre les réseaux.
- D'agents de communication, représentant les protocoles de niveau transport (TCP, UDP) ; ces agents sont attachés aux nœuds et connectés l'un à l'autre, ce qui représente un échange de données (connexion TCP, flux UDP).
- D'applications qui génèrent le trafic de données selon certaines lois, et se servent des agents de transport.

De plus, la description d'une expérience de simulation contient les instructions de trace des objets que l'on veut observer, et les mesures que l'on veut collecter.

5. Modélisation de la transformée en ondelettes dyadique

La transformation de l'image à la source en une représentation multi-résolution est une opération qui coûte de l'énergie. Il faut donc en tenir compte. Un modèle de consommation d'énergie pour la transformée en ondelettes dyadique est défini dans [63]. Ce modèle a été établi en décomposant le processus global en instructions élémentaires, et en déterminant combien de fois ces instructions étaient exécutées lorsque le filtre 5-3 de Le Gall est appliqué (c'est justement celui que nous utilisons). En fait, pour chaque pixel de l'image d'origine :

- L'application du filtre passe-bas nécessite 8 décalages et 8 additions.
- Le filtre passe-haut exige 2 décalages et 4 additions.
- De plus, chaque pixel doit être lu deux fois en mémoire et écrit deux fois.

En considérant que l'image d'origine a une dimension de $M \times N$ pixels, et que l'image est décomposée en p niveaux de résolution, et donc que la transformée en ondelettes dyadique est exécutée itérativement $p - 1$ fois, alors le coût d'énergie est approximativement donné par:

$$E_{DWT}(M, N, p) = M.N.(10.\varepsilon_{shift} + 12.\varepsilon_{add} + 2.\varepsilon_{rmem} + 2.\varepsilon_{wmem}). \sum_{i=1}^{p-1} \frac{1}{4^{(i-1)}}$$

Où ε_{shift} , ε_{add} , ε_{rmem} et ε_{wmem} représentent le coût d'énergie des quatre instructions élémentaires sur octet, respectivement le décalage, l'addition, la lecture et l'écriture.

6. Simulation et Performance

Dans cette partie nous avons simulé quatre scénarios, puis on a comparé l'énergie consommée dans chaque nœud capteur pour les différents scénarios pour évaluer notre solution.

6.1. Environnement de simulation

Notre modèle d'expérimentation est établi sur 5 nœuds, dispersés sur une surface carrée de 800m x 800m avec les coordonnées suivantes qui sont fixés pour tous les scénarios, les valeurs de localisation des nœuds capteurs sont données par la table1 :

Localisation	Valeur
nœud 0	(10,20)
nœud 1	(160,20)
nœud 2	(310,20)
nœud 3	(460,20)
nœud 4	(610,20)

Table 1 : paramètres de localisation des nœuds capteurs

Nous assumons que tous les nœuds ont une position fixe durant toute la période de simulation. Notre simulation est fait sur une image avec une résolution 128x128 pixels, chaque pixel est représenté par 8 bits, donc on va transmettre 16394 octets avec 10 octets de l'entête d'image.

Nous avons utilisé NS2 [65] avec les caractéristiques de mica2 pour notre simulation, prenant en considération l'énergie consommée pour la transformation en ondelette. Nous avons simulé 4 scénarios en calculant l'énergie consommée de chaque nœud avec les paramètres donnée par la table 2 :

Paramètres	Valeur
Le nombre des nœuds	5
Energie initiale	1J
La taille des paquets	230 bytes
Modèle de propagation radio	TwoRayGround

Table 2 : paramètres de simulation

Voila les différents scénarios qu'on a simulés :

- **Premier scénario** : on a transmet l'image sans appliquer la transformation en ondelette. Voir la figure suivante :

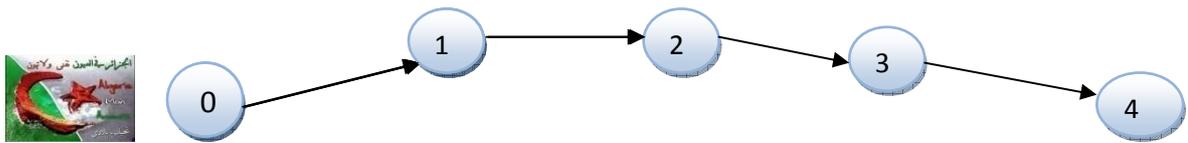


Figure 21 : transmission d'image sans TO

- **Deuxième scénario** : on a fait la transformation discrète en ondelette une fois dans le nœud 0 puis la transmission de ces données après transformation comme il est montré par la figure suivante :

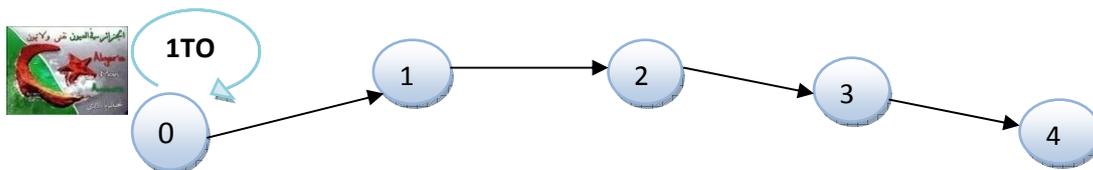


Figure 22 : transmission d'image avec une TO

- **Troisième scénario** : on a fait la transformation discrète en ondelette une fois dans le nœud 0 puis le résultat de la transformation est transmis au nœud 1 qui va appliquer une transformation discrète en ondelette sur les données reçues de nœud 0, puis le résultat de la transformation est transmis jusqu'à le nœud récepteur (ce scénario est notre idée proposée ou nous avons distribué la transformation discrète en ondelette entre le nœud 0 et le nœud 1 c'est-à-dire la première transformation dans le nœud 0 et la deuxième transformation dans le nœud 1).

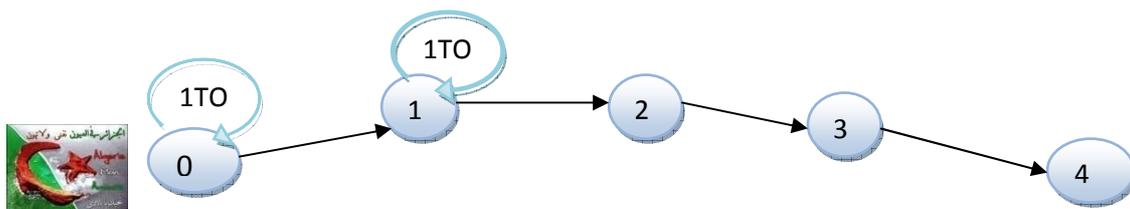


Figure 23 : transmission d'image avec une TO distribué deux fois

- **Quatrième scénario** : on a fait la transformation discrète en ondelette deux fois dans le nœud0 (transformation centralisé).

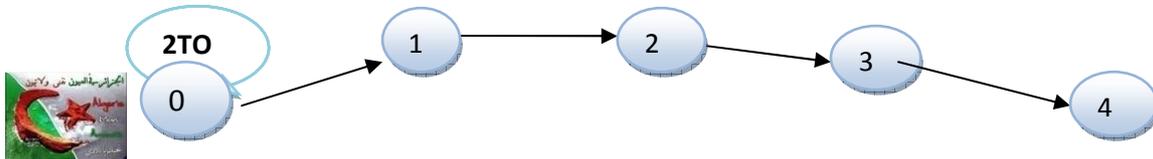


Figure 24 : transmission d'image avec une TO centralisée deux fois

6.2. Résultats de simulation :

Après la simulation des 4 scénarios, nous avons calculé l'énergie consommée de chaque nœud capteur pour la transmission et la réception des données à partir des fichiers de trace que produit NS2, puis on a calculé l'énergie consommée par la transformation en ondelette pour chaque nœud qui il a appliqué.

	Noeud0	Noeud1	Noeud2	Noeud3	Noeud4
Scénario 1	34029	37525	37554	37553	28995
Scénario 2	10890	10556	10527	10482	8035
Scénario 3	6265	5436	4822	4827	3076
Scénario 4	10459	9584	9496	9451	6957

Table 3 : Energie consommée de chaque nœud

La figure 6.3 montre l'énergie consommée de chaque nœud dans les différents scénarios.

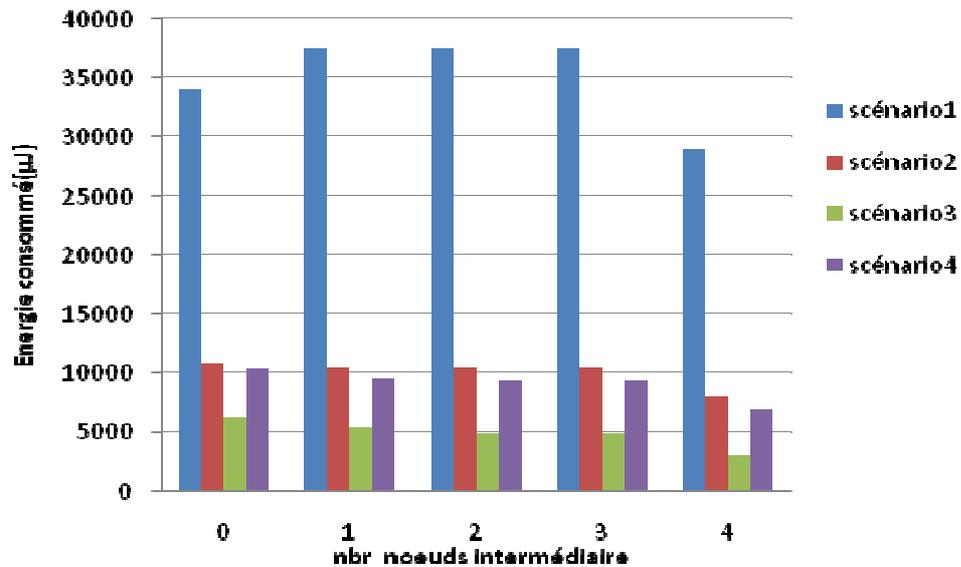


Figure 25 : l'énergie consommée dans les différents scénarios

Les résultats de notre simulation montre que :

La transformation en ondelette est une méthode de compression adaptable aux réseaux de capteurs sans fil en comparant l'énergie consommée par les nœuds dans le scénario1 (transmission d'image sans compression) et les autres scénarios qui utilisent la transformation en ondelette.

La distribution de la transformation en ondelette (scénario3) à montrer des bons résultats par rapport à la méthode centralisée (scénario4).

7. CONCLUSION

Dans cette partie nous avons proposé une méthode de compression d'image distribuée pour les RCSF basée sur la transformation discrète en ondelette.

Notre simulation a montré que la transformation d'ondelette discrète est une méthode de compression adaptée aux RCSF. L'énergie consommée par les nœuds capteurs utilisant la méthode centralisée est plus importante que dans le cas de notre méthode.

La compression d'image dans les RCSF est encore au début, la création des algorithmes de compression d'image adaptée aux contraintes de limitation de ressources du nœud capteur est très difficile.

Conclusion Générale

Dans ce mémoire, nous avons présenté un type particulier de réseau ad hoc qui est le réseau de capteur sans fil avec ses caractéristiques particulières et ses ressources limitées, puis on a parlé sur les différents mécanismes de conservation d'énergie dans les RCSFs. Ensuite on a concentré sur les applications de RCSF utilisant l'imagerie et les difficultés engendré par cette dernière à cause de la grande quantité de données que renferme la représentation d'une image. Nous avons étudié un ensemble de travaux qui traitent essentiellement le traitement des images sur les réseaux de capteurs sans fil.

Dans le domaine du traitement d'images, on remarque des efforts dans plusieurs travaux pour adapter les algorithmes de compression d'images aux contraintes particulières des réseaux de capteurs sans fil. Cependant, même si les algorithmes classiques présentent des bons performances, des expériences sur des capteurs démontrent que ces algorithmes sont beaucoup trop coûteux en énergie, plus coûteux que la transmission d'une image sans compression. Plusieurs adaptations ont été proposées pour rendre ces algorithmes moins gourmands en énergie.

Notre solution consiste à distribuer la compression, et en particulier distribué la transformation en ondelette discrète pour minimiser l'énergie consommée par le nœud source, on a supposé que tous les nœuds capteurs sont chargés de faire la TO, puis on a distribué la répétition de la transformation entre les nœuds capteurs.

La compression d'image dans les RCSF est un sujet de recherche très difficile à cause des contraintes de limitation de ressources du nœud capteur. On peut bien sûr construire des capteurs avec de grandes capacités de calcul mais qui consomment malheureusement et certainement beaucoup d'énergie. La bonne solution dans ce cas, consiste à trouver des algorithmes de compression réalisant le compromis entre l'économie d'énergie et les hautes performances.

Conférence nationale :

A.Chouha et A.Bilami, *traitement et transfert d'images par réseau de capteurs sans fil*. ICIEM'10 (International Conference on Industrial Engineering & Manufacturing) May 2010, Université de Batna.

Références

- [1]. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, A survey on sensor networks. IEEE Communications Magazine, vol 40, pp. 102-114, August, 2002.
- [2]. CAYIRCI, E. (2004). "Wireless sensor networks". In : D. Katsaros et al. (éd), Wireless information highways (pp. 273-301). Hershey : Idea group inc.
- [3]. S. Tilak, N. Abu-Ghazaleh, and W. Heinzelman,"A taxonomy of Wireless Micro-sensor Network Models," ACM SIGMOBILE, Mobile Computing and Communications Review, vol.6, issue: 2, pp. 28-36, April, 2002. [Available from the World Wide Web (WWW): <http://www.cs.binghamton.edu/nael/research/papers/taxonomy.pdf>].
- [4]. Eiko Yoneki, Jean Bacon, "A survey of Wireless Sensor Network technologies: research trends and middleware's role," Technical Report, no: 646, UCAM ;CL;TR;646, [Available from the World Wide Web <http://www.cl.cam.ac.uk/TechReports/UCAM;CL;TR;646.pdf>].
- [5]. H. Cam, S. Ozdemir, P. Nair, and D. Muthuavinashippan, "ESPDA: Energy-Efficient and Secure Pattern Based Data Aggregation for Wireless Sensor Networks," in press, IEEE Sensor, Toronto, Canada, 2003.
- [6]. C.Y. Chong and S.P. Kumar. Sensor Networks: Evolution, Opportunities, and Challenges. In Proceedings of the IEEE, vol.91, no.8, pp. 1247-1256, 2003.
- [7]. T.B. Gosnell, J.M. Hall, C.L. Ham, D.A. Knapp, Z.M. Koenig, S.J. Luke, B.A. Pohl, A. Schach von Wittenau, and J.K. Wolford. "Gamma-Ray Identification of Nuclear Weapon Materials". Technical Report DE97053424. Lawrence Livermore National Lab., Livermore, CA (USA). February 1997.
- [8]. M. J. Brown. "Users Guide Developed for the JBREWS Project". Technical Report LA-UR- 99-4676. Los Alamos National Laboratory of California University. 1999.
- [9]. P. Johnson and D.C Andrews. "Remote continuous monitoring in the home". Journal of Telemedicine and Telecare, vol.2, no.2, pp.107-113, June 1996.
- [10]. Yazeed Al-Obaisat, Robin Braun "On Wireless Sensor Networks: Architectures, Protocols, Applications, and Management" Institute of Information and Communication Technologies University of Technology, Sydney, Australia.

- [11]. Jamal N. Al-Karaki Ahmed E. Kamal "Routing Techniques in Wireless Sensor Networks: A Survey", Dept. of Electrical and Computer Engineering Iowa State University, Ames, Iowa 50011.
- [12]. F. Ye, A. Chen, S. Liu, L. Zhang, "A scalable solution to minimum cost forwarding in large sensor networks", Proceedings of the tenth International Conference on Computer Communications and Networks (ICCCN), pp. 304-309, 2001.
- [13]. S. Hedetniemi and A. Liestman, "A survey of gossiping and broadcasting in communication networks," Networks, Vol. 18, No. 4, pp. 319-349, 1988.
- [14]. C. Intanagonwiwat, R. Govindan, and D. Estrin. Directed diffusion: a scalable and robust communication paradigm for sensor networks. In MOBICOM, pages 56–67, 2000.
- [15]. W. Heinzelman, J. Kulik, and H. Balakrishnan, "Adaptive protocols for information dissemination in wireless sensor networks," in the Proceedings of the 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'99), Seattle, WA, August 1999.
- [16]. J. Kulik, W. R. Heinzelman, and H. Balakrishnan, "Negotiation-based protocols for disseminating information in wireless sensor networks," Wireless Networks, Volume: 8, pp. 169-185, 2002.
- [17]. W.R. Heinzelman, A. Chandrakasan, H. Balakrishnan. "Energy-efficient Communication Protocol for Wireless Microsensor Networks". Proceedings of the IEEE Hawaii International Conference on System Sciences. 2000, Vol. 2, p. 10.
- [18]. W.R. Heinzelman, A.P. Chandrakasan , and H. Balakrishnan. "An application specific protocol architecture for wireless microsensor networks". IEEE Transactions Wireless Communications. October 2002, Vol. 1, 4, pp. 660-670.
- [19]. Seema Bandyopadhyay, Coyle E.J. "An energy efficient hierarchical clustering algorithm for wireless sensor networks". INFOCOM. 2003, pp. 1713-1723.
- [20]. Lindsey, S. Raghavendra, C.S. PEGASIS: Power-efficient gathering in sensor information systems. IEEE Aerospace Conference Proceedings. 2002, Vol. 3, pp. 3-1130.
- [21]. Y. Yu, D. Estrin, and R. Govindan, "Geographical and Energy-Aware Routing: A Recursive Data Dissemination Protocol for Wireless Sensor Networks", UCLA Computer Science Department Technical Report, UCLA-CSD TR-01-0023, May 2001.
- [22]. POTTIE, G. J. et KAISER, W. J. (2000). "Wireless integrated network sensors".ACM Communications, 43, 51-58.
- [23]. ZIXIANG, X., LIVERIS, A. et CHENG, S. (2004). "Distributed source coding for sensor networks". IEEE Signal Processing Magazine, 21, 80-94.
- [24]. PANTAZIS, N. et VERGADOS, D. (2007). "A survey on power control issues in wireless sensor networks". IEEE Communications Surveys & Tutorials, 9, 86-107.

- [25]. YE, W., HEIDEMANN, J. et ESTRIN, D. (2004). "Medium access control with coordinated adaptive sleeping for wireless sensor networks". *IEEE/ACM Transactions on Networking*, 12, 493-506.
- [26]. CI, S., SHARIF, H. et NULI, K. (2005). "Study of an adaptive frame size predictor to enhance energy conservation in wireless sensor networks". *IEEE Journal on Selected Areas in Communications*, 23, 283-292.
- [27]. SHAH, R. et RABAEY, J. (2002). "Energy aware routing for low energy ad hoc sensor networks". *Proc. IEEE Wireless Communications and Networking Conference (WCNC)*. vol. 1, 350-355.
- [28]. XU, Y., HEIDEMANN, J. et ESTRIN, D. (2001). "Geography-informed energy conservation for ad hoc routing". *Proc. 7th annual international conference on Mobile computing and networking (MobiCom)*. 70-84.
- [29]. AKAN, O. B. et AKYILDIZ, I. F. (2005). "Event-to-sink reliable transport in wireless sensor networks". *IEEE/ACM Transactions on Networking*, 13, 1003-1016.
- [30]. WAN, C.-Y., CAMPBELL, A. T. et KRISHNAMURTHY, L. (2002). "PSFQ: a reliable transport protocol for wireless sensor networks". *Proc. 1st ACM international workshop on Wireless sensor networks and applications (WSNA)*. ACM Press, 1-11.
- [31]. XING, G., WANG, X., ZHANG, Y., LU, C., PLESS, R. et GILL, C. (2005). "Integrated coverage and connectivity configuration for energy conservation in sensor networks". *ACM Transactions on Sensor Networks*, 1, 36-72.
- [32]. LU, J., WANG, J. et SUDA, T. (2006). "Scalable coverage maintenance for dense wireless sensor networks". *Proc. 3rd Annual IEEE Communications Society on Sensor and Ad Hoc Communications and Networks (SECON)*. vol. 2, 651-660.
- [33]. MHATRE, V. et ROSENBERG, C. (2004). "Homogeneous vs heterogeneous clustered sensor networks: a comparative study". *Proc. IEEE International Conference on Communications*. vol. 6, 3646-3651.
- [34]. YOUNIS, M., MUNSHI, P., GUPTA, G. et ELSHARKAWY, S. (2006). "On efficient clustering of wireless sensor networks". *Proc. IEEE Workshop on Dependability and Security in Sensor Networks and Systems (DSSNS)*. 10-91.
- [35]. LIU, J.-S. et LIN, C.-H. R. (2005). "Energy-efficiency clustering protocol in wireless sensor networks". *Elsevier Ad Hoc Networks*, 3, 371-388.
- [36]. YAO, Y. et GIANNAKIS, G. (2005). "Energy-efficient scheduling for wireless sensor networks". *IEEE Transactions on Communications*, 53, 1333-1342.
- [37]. TIAN, D. et GEORGANAS, N. (2002). "A coverage-preserving node scheduling scheme for large wireless sensor networks". *Proc. 1st ACM international workshop on Wireless sensor networks and applications (WSNA)*. ACM, New York, NY, USA, 32-41.

- [38]. Bae, Jaewook et Richard M. Voyles (2006). Wireless video sensor networks over bluetooth for a team of urban search and rescue robots. In : Proceedings of the 2006 International Conference on Wireless Networks. Las Vegas, NV.
- [39]. Schettini, R, C. Fernandez-Maloigne et S. Susstrunck (2003). Color image processing. Pattern Recognition Letter.
- [40]. Cao, Zhi-Yan, Zheng-Zhou Ji et Ming-Zeng Hu (2005). An image sensor node for wireless sensor networks. In: International Conference on Information Technology: Coding and Computing (ITCC'05). Vol. 2. pp. 740–745.
- [41]. McCormick, Chris, Pierre-Yves Laligand, Huang Lee et Hamid Aghajan (2006). Distributed agent control with self-localizing wireless image sensor networks. In : Proceedings of Cognitive Systems and Interactive Sensors (COGIS 2006).
- [42]. Downes, Ian, Leili Baghaei Rad et Hamid Aghajan (2006). Development of a mote for wireless image sensor networks. In: Proceedings of Cognitive Systems and Interactive Sensors (COGIS 2006).
- [43]. Hengstler, Stephan, Daniel Prashanth, Sufen Fong et Hamid Aghajan (2007). MeshEye : A hybridresolution smart camera mote for applications in distributed intelligent surveillance. In : Proceedings of the 6th international conference on Information processing in sensor networks (IPSN-SPOTS). ACM Press. Cambridge, Massachusetts, USA. pp. 360–369.
- [44]. Rahimi, Mohammad, Deborah Estrin, Rick Baer, Henry Uyeno et Jay Warrior (2004). Cyclops: Image sensing and interpretation in wireless networks. In : Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems. ACM Press.
- [45]. Rowe, Anthony, Adam Goode, Dhiraj Goel et Illah Nourbakhsh (2007). CMUcam3: An open programmable embedded vision sensor. Technical Report RI-TR-07-13. Carnegie Mellon Robotics Institute. Pittsburgh, Pennsylvania 15213.
- [46]. Culurciello, Eugenio et Andreas G. Andreou (2004). ALOHA CMOS imager. In : IEEE International Symposium on Circuits and Systems (ISCAS). Vancouver, Canada. pp. IV–956–9.
- [47]. Teixeira, Thiago, Andreas G. Andreou et Eugenio Culurciello (2005). Event-based imaging with active illumination in sensor networks. In : IEEE International Symposium on Circuits and Systems (ISCAS 2005). Vol. 1. pp. 644–647.
- [48]. Mr. Brault, Mme. Dougherty (2002). “Les formats de compression d'image“. Institut Universitaire de Technologie de Tours Département Génie Électrique et Informatique Industrielle.
- [49]. Ferrigno, L., S. Marano, V. Paciello et A. Pietrosanto (2005). Balancing computational and transmission power consumption in wireless image sensor networks. In: IEEE International Conference on Virtual Environments, Human-Computer Interfaces, and Measures Systems (VECIMS 2005). Giardini Naxos, Italy.

- [50]. Kimura, Naoto et Shahram Latifi (2005). A survey on data compression in wireless sensor networks. In :International Conference on Information Technology : Coding and Computing (ITCC 2005). Vol. 2,pp. 8–13.
- [51]. Wu, Huaming et Alhussein A. Abouzeid (2004b). Power aware image transmission in energy constrained wireless networks. In : Proceedings of the 9th IEEE Symposium on Computers and Communications (ISCC'2004). Alexandria, Egypt.
- [52]. Maniezzo, D., K. Yao et G. Mazzini (2002). Energetic trade-off between computing and communication resource in multimedia surveillance sensor network. In : 4th IEEE International Workshop on Mobile and Wireless Communications Network (MWCN 2002). Stockholm, Sweden.
- [53]. Wu,Min et ChangWen Chen (2003).Multiple bitstream image transmission over wireless sensor networks. In : Proceedings of IEEE Sensors. Vol. 2. pp. 727–731.
- [54]. Yu, Wei, Zafer Sahinoglu et Anthony Vetro (2004). Energy efficient JPEG 2000 image transmission over wireless sensor networks. In : IEEE Global Telecommunications Conference (GLOBECOM '04). Vol. 5. pp. 2738–2743.
- [55]. Wu, Huaming et Alhussein A. Abouzeid (2004a). Energy efficient distributed JPEG2000 image compression in multihop wireless networks. In: 4th Workshop on Applications and Services in Wireless Networks (ASWN 2004). pp. 152–160.
- [56]. Wu, Huaming et Alhussein A. Abouzeid (2006). Error resilient image transport in wireless sensor networks. *Computer Networks*, 50(15), 2873–2887.
- [57]. Lee, Dong-U, Hyungjin Kim, Steven Tu, Mohammad Rahimi, Deborah Estrin et John D. Villasenor (2007a). Energy-optimized image communication on resource-constrained sensor platforms. In: IPSN'07. Cambridge, Massachusetts, USA.
- [58]. Mammeri, Abdelhamid, Ahmed Khoumsi, Djemel Ziou et Brahim Hadjou (2008). Modeling and adapting JPEG to the energy requirements of VSN. In : Proceedings of 17th International Conference on Computer Communications and Networks ICCCN'08.US Virgin Islands.
- [59]. Slepian, David et Jack K. Wolf (1973). Noiseless coding of correlated information sources. *IEEE Transactions on Information Theory*, IT-19(4), 471–480.
- [60]. Wagner, Raymond, Robert Nowak et Richard Baraniuk (2003). Distributed image compression for sensor networks using correspondence analysis and super-resolution. In:Proceedings of 2003 International Conference on Image Processing (ICIP). Vol. 1. pp.597–600.
- [61]. Gehrig, Nicolas et Pier Luigi Dragotti (2004). Distributed compression in camera sensor networks. In: Proceedings of IEEE International Workshop on Multimedia Signal Processing (MMSP). Siena, Italy.

- [62]. F. Marino, V. Piuri, and E. J. Swartzlander, "A parallel implementation of the 2-D discrete wavelet transform without interprocessor communications," IEEE Transactions on Signal Processing, vol. 47, no. 11, pp. 3179 – 3184, November 1999.
- [63]. Vincent Lecuire, Cristian Duran-Fandez, Nicolas Krommenacker."Energy-Efficient Transmission of Wavelet-Based Images in Wireless Sensor Networks" Centre de Recherche en Automatique de Nancy (CRAN - UMR 7039), EURASIP Journal on Image and Video Processing 2007.
- [64]. Network Simulator: <http://www.isi.edu/ns2/> .
- [65]. Simulateur réseau REAL : <http://www.cs.cornell.edu/skeshav/real/overview.html> .