



REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
Ministère de l'Enseignement Supérieure et de La Recherche
Scientifique
Université Mouloud MAMMARI de Tizi-Ouzou
Faculté de Génie Electrique et Informatique
Département Informatique



Mémoire de fin d'étude en vue de l'obtention du diplôme de Master II en Informatique
Option : Système Informatique

Thème

Application de détection de présence par
réseau de capteurs sans fil

Simulation dans l'environnement TinyOS

Proposé et dirigé par : M^r M.DEMRI

Présenté par :
M^{lle} TOUAT Fatma
M^{lle} TOUAT Nawal

Présenté devant le jurys :

2014/2015



Remerciements

*Nous tenons à exprimer notre profondes
gratitudes à notre promoteur, Monsieur M.DEMRI
de nous avoir encadrés et guidés tout au long de notre
projet, pour ses conseils judicieux et minutieusement
prodigués. Aussi nous tenons à lui reconnaître le
temps précieux qu'il nous a consacré.*

*Que les membres de jury trouvent ici nous
remerciements les plus vifs pour accepté d'honorer par
leur jugement notre travail.*

*Nos sincères sentiments vont à tous ceux qui,
de près ou de loin, ont contribué à la réalisation de ce
projet.*

DEDICACES

Je dédie ce modeste travail tout d'abord à mes très chers parents qui m'ont toujours soutenu et qui m'ont fourni tous les moyens nécessaires tout au long de mes études.

Je le dédie également :

A mes deux adorables frères Rabah et Rayan

A mes deux adorables sœurs Souad et Lydia

A mes grands parents

A mes tantes surtout a ma tante Zahia et sa fille

Cylia ainsi que mes oncles et tout leurs enfants

A mon très cher mari Takfarinas

A ma future belle-famille

A ma binôme et cousine Fatma et toute sa famille

A mes copines Nabila, Katia, Farah et Nora ainsi

qu'à toute leurs familles

Et enfin à tous ceux qui me sont chers

Un grand merci de ma part.

Je vous aime

Nawal

DEDICACES

Je remercie Dieu de m'avoir donné le courage pour accomplir ce modeste travail que je dédie :

A mes très chers parents qui m'ont toujours soutenu et qui m'ont fourni tous les moyens nécessaires tout au long de mes études

A ma sœur adorable Dyhia

A mes frères Mohand said et Hakim

A mes grands parents

A mon très cher ami Yacine et toute sa famille

A ma sœur et binôme Nawal et toute sa famille

A mes copines Nabila, Farah, Katia et Nora ainsi qu'à toutes leurs familles

Et à tous ceux que j'aime et qui m'aiment

Fatma

Résumé

Le projet est une application de détections de présence par les réseaux de capteurs sans fil, la simulation s'effectuera sous l'environnement TinyOS.

Son objectif est d'informer le contrôleur sur toute présence autour de l'usine quelle soit animale, humaine ou véhicule. Dissuadant ainsi toutes tentatives de vols sabotage et destruction

Le projet est organisé selon le plan suivant :

Une introduction suivi par une présentation des réseaux de capteurs sans fil, on enchainera par présenter quelques applications de détection de présence, en suite on passe au déroulement de notre application et on termine par une conclusion.

Les capteurs sont des minis composants capables de détecter différentes informations environnementales telle que la température, les vibrations ainsi que la présence. Ces capteurs sont peu coûteux et multifonctionnels.

L'ensemble de ces capteurs forme ce qu'on appelle les réseaux de capteurs sans fil qui résultent d'une fusion de deux pôles de l'informatique moderne les systèmes embarqués et les communications sans fil. Tous les nœuds fonctionnent en collaboration pour accomplir une tâche commune, ils sont auto-organisés et autonomes (chaque capteur possède une batterie). Le déploiement s'effectue d'une manière aléatoire et dense souvent dans des environnements hostiles inaccessible par l'être humain (foret, profondeur de la mer..., etc.) il s'effectue de plusieurs manières soit déploiement aléatoire où les nœuds sont éjectés par un avion, soit les nœuds sont placés un à un par un agent (robot).

Les capteurs sont utilisés dans plusieurs domaines tel que le domaine militaire, médicale, transport, agriculture ainsi que la surveillance et dans ce domaine on site plusieurs techniques de détection tel que les détecteur par cameras thermiques, par laser et les détecteurs par infrarouge.

L'application se déroule sous l'environnement TinyOS qui nous offre une simulation du comportement du réseau ainsi que la consommation énergétique de chaque nœud, grâce aux différents outils qu'il nous offre tel que TinyViz, TOSSIM et PowerTOSSIM .

Mots clés : Capteurs, RCSF, LEATCH, Infrarouge, TinyOS, NesC, TinyViz, TOSSIM, PowerTOSSIM, Surveillance,

Table des matières

Introduction générale	1
Chapitre I : Généralités sur les réseaux de capteurs sans fil	
Introduction	3
I. Histoire des réseaux de capteurs	3
II. Présentation des capteurs.....	4
II. 1. Définition d'un capteur.....	4
II. 2. Caractéristiques d'un capteur	4
II. 3. Les différents types de capteurs.....	5
II. 4. Architecture d'un capteur sans fil.....	6
III. Présentation des réseaux de capteurs sans fil	8
III. 1. Définition d'un réseau de capteur sans fil.....	8
III. 2. La différence entre RCSF et les réseaux Ad' hoc	8
III. 3. Architecture d'un réseau de capteur sans fil	9
III.4. Les principales caractéristiques et limites des RCSF.....	9
III. 4.1. Les caractéristiques	9
III. 4.1. Les limites	11
III.5. Domaine d'application	12
III.6. Architecture de communication dans les réseaux de capteurs sans fil.....	15
III.7. Topologies des réseaux de capteurs	15
III.8. Types des réseaux de capteurs sans fil	16

III.9. Les systèmes d'exploitation pour les réseaux de capteurs	17
III.10. Technologies utilisés	19
Conclusion.....	20
Chapitre II : Les applications de détection de présence	
Introduction	21
I. Historique	21
II. Principe de la détection de présence	21
III. Les différentes techniques de la détection de présence	22
III.1. Détecteur par taux de CO ₂	22
III.2. Détecteur par des cameras thermiques	22
III.3. Détecteur par le laser.....	23
III.4. Détecteur par infrarouge.....	24
III.5. Technologie à ultrasons (US)	25
III.6. Technologie Dual (Infrarouge et Ultrason)	25
III.7. Détecteur par photoélectrique	26
IV. Présentation de notre application	27
IV.1. Zone à surveillé.....	27
IV.2. Les outils utilisés.....	27
IV.2.1. Le détecteur infrarouge basé sur le volume	27
IV.2.2. Le protocole de routage utilisé	28
Conclusion.....	35

Chapitre III : Simulation

Introduction	36
I. Environnement de simulation	36
I.1. Le système d'exploitation TinyOS	36
I.2. Le langage de programmation NesC.....	37
II. Les outils de simulation	38
II.1. TOSSIM.....	38
II.2. PowerTOSSIM	38
II.3. TinyViz.....	48
III. Simulation	39
III.1. Simulation graphique	40
III.1. Simulation d'énergie	43
Conclusion.....	47
Conclusion générale	48

Introduction générale

Depuis leur création, les réseaux de communication sans fil ont connu un succès croissant au sein des communautés scientifiques et industrielles. Grâce à ses divers avantages, cette technologie a pu s'instaurer comme acteur incontournable dans les architectures réseaux actuelles. Au cours de son évolution, le paradigme sans fil a vu naître diverses architectures dérivées, telles que : les réseaux cellulaires, les réseaux locaux sans fils et autres. Durant cette dernière décennie, une architecture nouvelle a vu le jour : les réseaux de capteurs sans fil. Ce type de réseaux résulte d'une fusion de deux pôles de l'informatique moderne : les systèmes embarqués et les communications sans fil. Un réseau de capteurs sans fil (RCSF), ou "Wireless Sensor Network" (WSN), est composé d'un ensemble d'unités de traitements embarquées, appelées "nœuds", communiquant via des liens sans fil. Le but général d'un WSN est la collecte d'un ensemble de paramètres de l'environnement entourant les nœuds, telles que la température, la pression de l'atmosphère ou la présence, ..., etc. afin de les acheminer vers des points de traitement « Sink ». Les RCSF sont souvent considérés comme étant les successeurs des réseaux ad hoc. En effet, les RCSF partagent avec les MANET (Mobile Ad hoc NETWORKS) plusieurs propriétés en commun, telles que l'absence d'infrastructure et les communications sans fil. Mais l'une des différences clés entre les deux architectures est le domaine d'application. Contrairement aux réseaux MANET, qui n'ont pas pu connaître un vrai succès, les RCSF ont su attirer un nombre croissant d'industriels, vu leur réalisme et leur apport concret. En effet, le besoin d'un suivi continu d'un environnement donné est assez courant dans diverses activités de la société.

Les processus industriels, les applications militaires de tracking, le monitoring d'habitat, la surveillance ainsi que l'agriculture de précision ne sont que quelques exemples d'une panoplie vaste et variée d'applications possibles du suivi continu offert par les RCSF. Grâce à ce potentiel riche en applications, les RCSF ont su se démarquer de leur origine MANET et attirer de grandes firmes à travers le monde, telles que IBM, Sun, Intel et Philips. Malheureusement, les RCSF ne sont pas parfaits ! A cause de leur faible coût et leur déploiement dans des zones parfois hostiles, les nœuds sont assez fragiles et vulnérables à diverses formes de défaillances : cassure, faible énergie, ... etc. Ces problèmes rendent les RCSF des systèmes à fragilité innée, qui doit être considérée comme une propriété normale du réseau.

Notre projet de fin d'étude consiste à simuler une surveillance des alentours d'une zone industriel permettant ainsi au contrôleur d'être informé en temps réel sur toutes les présences

déecté dans la zone à surveiller et définir sa nature (animal, humain ou véhicule) dissuadant ainsi toute tentative de vols, sabotage ou destructions et autre.

En optant pour l'utilisation du système d'exploitation adéquat TinyOS qui est conçu spécialement pour ce type de réseau et qui est écrit dans un langage spécifique optimisé pour les capteurs « le langage NesC ». Afin d'aborder tous les aspects ayant trait au fonctionnement de ces réseaux, notre mémoire est organisé comme suit :

Chapitre I : « Généralités sur les réseaux de capteurs sans fil »

Chapitre II : « Les applications de détection de présence »

Chapitre III : « Simulation de l'application »

Et enfin une conclusion générale.

Introduction

Depuis quelques décennies, le besoin d'observer et de contrôler les environnements hostiles est devenu essentiel dans différents domaines : militaires, scientifiques, médical, ..., etc.

Ceci nécessite de nouvelles technologies qui prennent en considération la sensibilité de l'environnement et qui fournit des informations pertinentes sur le milieu balayé. Ainsi, des progrès récents dans les communications sans fil et le domaine de l'électronique ont permis le développement de mini composants peu coûteux et multifonctionnels. Ces derniers sont capables de détecter différents types d'informations, comme la température, l'humidité, la lumière, les vibrations sismiques, et la présence ou la nature d'organismes biologiques. De plus, ils se doivent d'être autonomes, d'une taille miniature et peuvent être déployées d'une manière dense et aléatoire dans le champ surveillé. Ces dispositifs sont généralement appelés des "nœuds de senseurs" (ou simplement "senseurs") et peuvent être repartis sur le sol, dans les véhicules, à l'intérieur des constructions et même sur le corps humain. Comme ils peuvent être connectés sans être reliés physiquement, ils constituent un réseau de sensors sans fil (WSN : Wireless Sensor Network), capable de rassembler, traiter et relayer les informations entre les nœuds et de les envoyer au centre de contrôle.

I. Histoire des réseaux de capteurs

Dans les années 1990, dans le monde de la recherche, est apparue une idée qui paraissait plutôt un rêve pour cette époque : imaginer un système nerveux central pour la Terre, capable de surveiller en temps réel les événements, ayant comme principaux bénéfices de pouvoir empêcher les accidents et d'économiser l'énergie. (Cette poussière intelligente a mis longtemps à apparaître) dit le professeur Pister, de l'Université de Californie à Berkeley. (J'ai inventé l'expression il y a 14 ans. La poussière vraiment futée a mis du temps, mais elle est finalement arrivée). [AAC 2014]

Aujourd'hui les réseaux de capteurs sont devenus des systèmes pouvant atteindre un très grand nombre de nœuds, avec une zone de couverture déterminée et déployés d'une manière plus ou moins dense dans un environnement hétérogène dont on mesure ainsi son état global. Les derniers progrès en terme de miniaturisation, ainsi que le remplacement du câblage classique par des technologies de communication radio, ont généré de nouvelles

catégories d'applications qui visent de nombreux domaines : l'aéronautique, l'automobile, le médical, l'environnement, ..., etc. De plus, les progrès des communications sans fil permettent aujourd'hui de répondre aux exigences peu envisageables auparavant. [SAH 2013]

II. Présentation des capteurs

II.1. Définition d'un capteur [CHR Web]

Un capteur est un organe de prélèvement d'informations qui élabore à partir d'une grandeur physique environnementale (Information entrante : température, lumière, pression, etc.) une autre grandeur physique de nature différente (Information sortante : très souvent électrique, Optique). Cette grandeur, est représentative de la grandeur prélevée, la donnée résultante est communiquer à un centre de contrôle via la station de base(Sink).

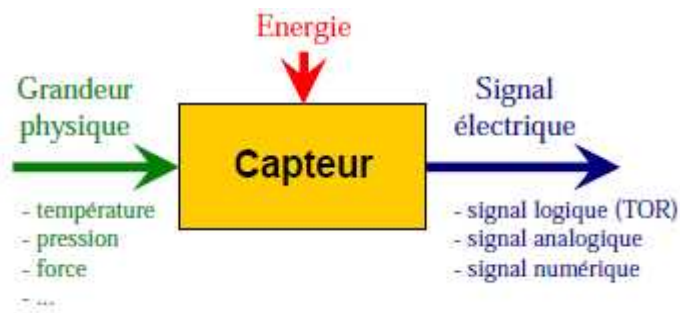


Figure 1 : Capteur. [CHR Web]

II.2. Caractéristiques d'un capteur [CHR Web]

- **Précision** : Aptitude du capteur à donner une mesure proche de la valeur réel.
- **Rapidité (temps de réponse)** : Habilité de la sortie d'un capteur à suivre les variations temporelles de la mesurande. (Temps de réaction du capteur)
- **Finesse** : Capacité à peu perturber la mesurande.
- **Etendue de mesure** : Valeurs extrêmes pouvant être mesurée par le capteur.
- **Résolution** : Plus petite variation de grandeur mesurable par le capteur.
- **Sensibilité** : Variation du signal de sortie par rapport à la variation du signal d'entrée.
- **Étalonnage** Permet d'ajuster et de déterminer sous forme algébrique ou graphique, la relation entre la grandeur de sortie et la mesurande.

II.3. Les différents types de capteurs [RAO 2010]

- **Les capteurs de lumière**

Fonctionne avec une photorésistance (résistance qui change de valeur en fonction de la luminosité). La sortie du capteur sera donc à l'état 0 ou à l'état 1 en fonction de la luminosité ambiante



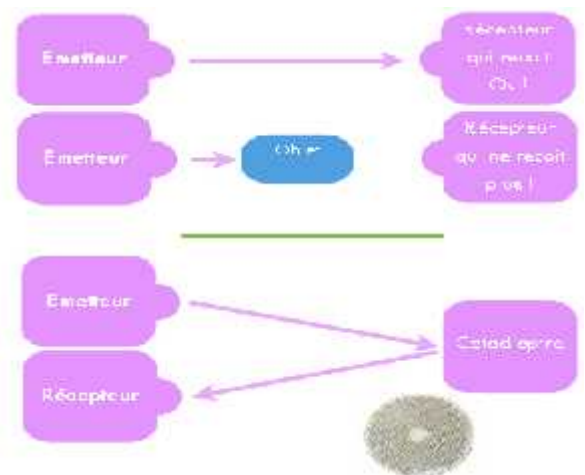
- **Capteur de passage**

Capteur infra rouge. Il se compose d'un émetteur de lumière associé à un récepteur. La détection d'un objet se fait par coupure du faisceau lumineux. On utilise la lumière infra rouge car elle est invisible à l'œil.



Il existe 2 cas d'utilisation :

- ✓ Emetteur et récepteur en face.
- ✓ Emetteur et récepteur côte à côte (idéal pour le passage des câbles), dans ce cas-là, on utilise un catadioptré qui renvoie le faisceau lumineux.



- **Le capteur de présence**

Utilise l'ultrason.



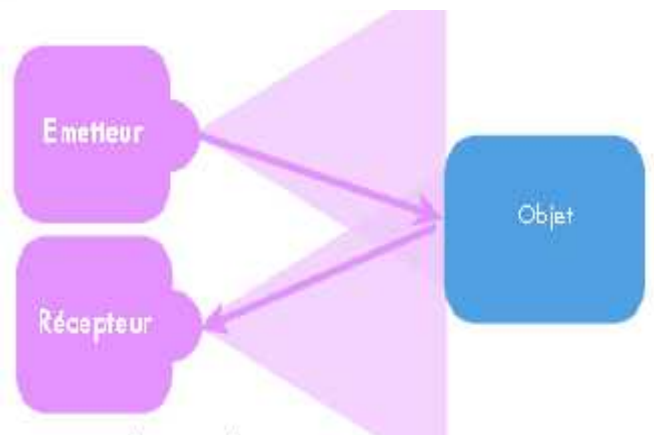
Un émetteur ultrason envoie un son inaudible à l'humain. Lorsqu'un objet est suffisamment près, le son «rebondit» sur l'objet, le récepteur ultrason reçoit alors le signal sonore se déclenche.

La distance de détection est en fonction du réglage du capteur.

On peut même savoir à quelle distance se trouve l'objet, en calculant le temps qu'a mis le son pour faire l'aller/retour (émetteur, récepteur).

Exemple : Le son a mis 2s pour faire l'aller/retour, il va à une vitesse de 10m/s. Il a donc mis 1s pour faire l'aller (émetteur objet).

L'objet se trouve donc à 10m.



Autre exemple



Capteur de température capteur de niveaux Détecteur de fumée capteur de son

II.4. Architecture d'un capteur sans fil [AAC 2014]

Un « nœud capteur » est composé de quatre unités de base : l'unité de capture, l'unité de traitement, l'unité de communication, et l'unité de contrôle d'énergie. Selon le domaine d'application, il peut aussi contenir des modules supplémentaires tels qu'un système de localisation (GPS), ou bien un système générateur d'énergie (cellule solaire).

Quelques micro-capteurs, plus volumineux, sont dotés d'un système mobilisateur chargé de les déplacer en cas de nécessité.



Figure2: Architecture d'un capteur. [AAC 2014]

- **L'unité de capture**

Elle permet la liaison entre le capteur et l'environnement, elle se compose du capteur lui-même et d'un convertisseur analogique/numérique (ADC). En effet, le signal analogique produit par le capteur suite à un phénomène observé va être transformé par l'ADC en un signal numérique. Qui va être transmis à l'unité de traitement.

- **L'unité de traitement**

Elle comprend un processeur généralement associé à une petite unité de stockage. Elle fonctionne à l'aide d'un système d'exploitation spécialement conçu pour les micros capteurs (TinyOS par exemple). Elle exécute les protocoles de communications qui permettent de faire « *collaborer* » le nœud avec les autres nœuds du réseau. Elle peut aussi analyser les données captées. Cette unité acquiert les informations en provenance de l'unité capture et les envoie à l'unité de communication.

- **L'unité de communication**

Elle effectue toutes les émissions et réceptions des données sur un médium « sans fil » Elle peut être de type optique (comme dans les nœuds Smart Dust), ou de type radio- fréquence.

Les communications de type optique sont robustes vis-à-vis des interférences électriques. Néanmoins, ne pouvant pas établir de liaisons à travers des obstacles, elles présentent l'inconvénient d'exiger une ligne de vue permanente entre les entités communicantes.

Les unités de communication de type radio-fréquence comprennent des circuits de modulation, démodulation, filtrage et multiplexage, ceci implique une augmentation de la complexité et du coût de production du micro-capteur.

Concevoir des unités de communication de type radio-fréquence avec une faible consommation d'énergie est un défi car pour qu'un nœud ait une portée de communication suffisamment grande, il est nécessaire d'utiliser un signal assez puissant et donc une énergie consommée importante. L'alternative consistant à utiliser de longues antennes n'est pas possible à cause de la taille réduite des micro-capteurs.

- **L'unité de contrôle d'énergie**

Un capteur est muni d'une ressource énergétique (généralement une batterie). Étant donné sa petite taille, cette ressource énergétique est limitée et généralement non-remplaçable. Ceci fait souvent de l'énergie la ressource la plus précieuse d'un réseau de capteurs, vu qu'elle influe directement sur la durée de vie des capteurs et donc du réseau entier.

L'unité de contrôle d'énergie constitue donc une partie essentielle du système. Elle doit répartir l'énergie disponible aux autres modules, de manière optimale (par exemple en réduisant les dépenses inutiles et en mettant en veille les composants inactifs). Cette

unité peut aussi gérer des systèmes de rechargement d'énergie à partir de l'environnement via des cellules photovoltaïque par exemple.

III. Présentation des réseaux de capteurs sans fil

III.1. Définition d'un réseau de capteurs sans fil ou (WSN) [LAB 2006]

Un réseau de capteurs sans fils (RCSF) dit «Wireless Sensor Network : WSN» est un type spécial de réseau Ad-hoc comportant un grand nombre de nœuds (capteurs). Ces nœuds sont conçus pour contrôler en coopération et d'une manière autonome les conditions physiques et environnementales dans différentes parties de l'espace où ils se trouvent implantés.

La position de ces nœuds n'est pas obligatoirement prédéterminée. Ils peuvent être aléatoirement dispersés dans une zone géographique, appelée « champ de captage », « zone de couverture » ou « zone d'intérêt ».

III.2. La différence entre les RCSF et les réseaux Ad'hoc [FLE 2004]

Comme il a été expliqué précédemment les réseaux de capteurs sans fil sont un type spécial de réseaux Ad' hoc. Chaque un a ses propres caractéristiques :

- **Senseurs**

1. Objectif ciblé
2. Nœuds collaborent pour remplir un seul objectif
3. Flux de données (many to one)
4. Très grand nombre de nœuds n'ayant pas tous un ID
5. facteur déterminant Energie
6. Utilisation du broadcast

- **Ad'hoc**

1. Générique/communication
2. Chaque nœud à son propre objectif
3. Flux de données (any to any)
4. Notion d'identificateur
5. facteur déterminant : le Débit
6. Communication point à point

III.3. Architecture d'un réseau de capteur sans fil [CRE Web]

Un RCSF est composé d'un ensemble de nœuds capteurs. Ces nœuds capteurs sont organisés en champs « sensor fields ». Chacun de ces nœuds a la capacité de collecter des données et de les transférer à la station de base (dit "sink" en anglais ou puits) par l'intermédiaire d'une architecture multi-sauts via une communication sans fil. Le puits transmet ensuite ces données par Internet ou par satellite à l'ordinateur central «Gestionnaire de tâches» pour analyser ces données et prendre des décisions.

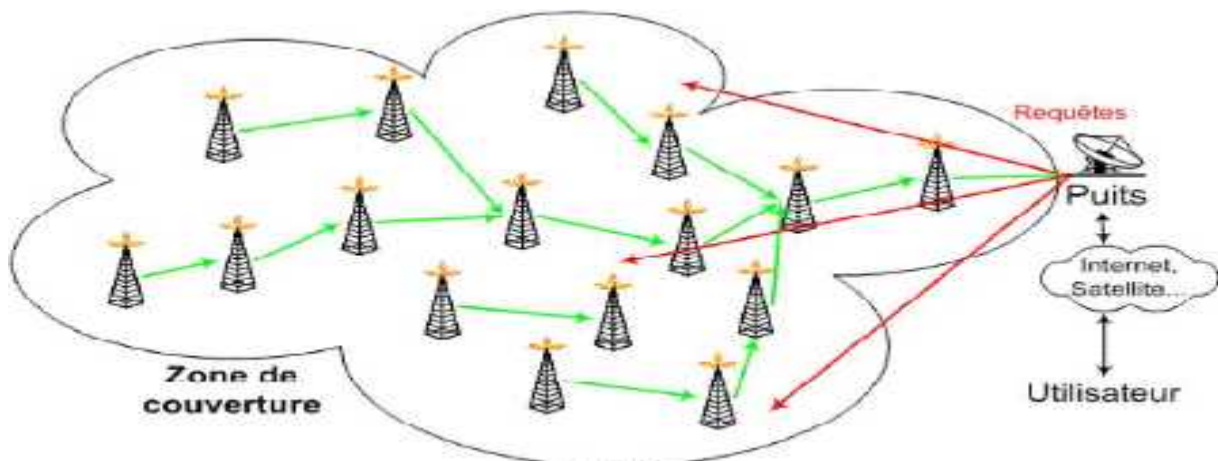


Figure 3 : Architecture d'un réseau de capteurs sans fil. [CRE Web]

III.4. Les principales caractéristiques et limites des RCSF

III.4.1 Les caractéristiques

- **La consommation réduite d'énergie** : Les nœuds capteurs utilisent des batteries de taille minuscule comme ressources énergétique, ce qui limite leur durée de vie. La spécificité des applications des RCSF (militaires, sismiques et autres) fait que la recharge ou le remplacement de ces batteries est une tâche difficile ou presque impossible, ce qui nous mène à déduire que la durée de vie d'un nœud dépend de la durée de vie de sa batterie. Ainsi, la méthode de gestion de la consommation d'énergie constitue une contrainte majeure dans ce type de réseau [AKY 2002].
- **L'auto-configuration des nœuds capteurs** : Dans un RCSF, les nœuds sont déployés soit d'une manière aléatoire (missile, avion...), soit placés nœud par nœud par un humain ou un robot, et ceci à l'intérieur ou autour du phénomène observé (champ de guerre, surface volcanique, patient malade...) [AKY 2002]. Ainsi, un nœud capteur doit avoir des capacités d'une part, pour s'auto-configurer dans le réseau, et d'autre

part pour collaborer avec les autres nœuds dans le but de reconfigurer dynamiquement le réseau en cas de changement de topologie du réseau [CHO 2003]. Dans un RCSF, chaque nœud X possède une unité émettrice/réceptrice qui lui permet de communiquer avec les nœuds qui lui sont proches; En échangeant des informations avec ces derniers, le nœud X pourra alors découvrir ses nœuds voisins et ainsi connaître la méthode de routage qu'il va adopter selon les besoins de l'application.

L'auto-configuration apparaît comme une caractéristique nécessaire dans le cas des RCSF étant donné que d'une part, leur déploiement s'effectue d'une manière aléatoire dans la majorité des applications, et d'autre part le nombre des nœuds capteurs est très grand [CHA 2009].

- **La scalabilité :** Contrairement aux réseaux sans fil traditionnels (personnel, local ou étendu), un RCSF peut contenir un très grand nombre de nœuds capteurs (des centaines, des milliers...).

Un réseau de capteur est scalable parce qu'il a la faculté d'accepter un très grand nombre de nœuds qui collaborent ensemble afin d'atteindre un objectif commun [AKY 2002].

- **La tolérance aux pannes :** Dans le cas de dysfonctionnement d'un nœud (manque d'énergie, interférences avec l'environnement d'observation...) ou aussi en cas d'ajout de nouveaux nœuds capteurs dans le réseau, le réseau doit continuer à fonctionner normalement sans interruption. Ceci explique le fait qu'un RCSF n'adopte pas de topologie fixe mais plutôt dynamique [AKY 2002].
- **Une densité importante des nœuds :** Les RCSF sont caractérisés par leur forte densité [CHA 2009]. Cette densité peut atteindre, selon le type d'application, 20 nœuds/m³ [AKY 2002].
- **La capacité de communication :** Elle peut prendre deux aspects : Le multi saut ou à un seul saut. Parce que le multi saut est moins énergivore, il reste le type de communication le plus sollicité par les applications de RCSF qui requièrent une faible consommation d'énergie [CHA2009].
- **Une architecture « data-centric » :** Du fait que le remplacement ou la recharge des batteries des nœuds capteurs est une tâche quasi impossible alors il est d'usage normal qu'on trouve des nœuds capteurs redondants (effectuant la même tâche dans la même région).

L'importance d'un nœud particulier est, par conséquent, réduite par rapport à l'importance attribuée aux données observées par les nœuds.

Ce type d'architecture diffère des architectures « node-centric » adoptées par les réseaux traditionnels où les nœuds possèdent une place importante (Exemple : un utilisateur qui veut connecter son laptop au serveur web X) [CHO 2003].

- **Une collaboration entre les nœuds** : Les contraintes strictes de consommation d'énergie mènent les nœuds capteurs à détecter et traiter les données d'une manière coopérative afin d'éviter le traitement redondant d'une même donnée observée, source de la perte d'énergie [AKY 2002].
- **La bande passante (ou capacité du canal)** : c'est une caractéristique beaucoup plus importante dans les réseaux cellulaires (GSM) et les réseaux locaux sans fils (WLAN), que dans les RCSF ; le débit étant en effet un objectif secondaire pour les RCSF [CHO 2003].

III.4.2. Les Limites [BOU 2010]

Au niveau de la Communication

- La bande passante :
 - ✓ Limitée et partagée par tous les nœuds du réseau WSN
 - ✓ Portée radio limitée à quelque dizaines de mètres et faible débit
 - ✓ Collision entre les centaines voire les milliers de nœuds
- Perte des données à cause de la transmission radio
 - ✓ Le renvoi des paquets est limité à cause du manque en énergie

Au niveau du matériel

- Puissance de calcul limitée fonctionnant dans la majorité des cas avec des registres 8 ou 16 bits.
- Mémoire limitée (2 à 250 Ko de RAM et 1 à 32 Mo de mémoire flash)
- Energie limitée
 - ✓ Un facteur critique (influt directement sur la durée de vie du réseau)
 - ✓ Toute solution implémentée (sécurité, routage,...) doit minimiser sa consommation énergétique

III.5. Domaine d'application [CHA 2008]

- **Applications militaires** : On peut penser à un réseau de capteurs déployé sur un endroit stratégique ou difficile d'accès, afin de surveiller toutes les activités des forces ennemies, ou d'analyser le terrain avant d'y envoyer des troupes (détection d'agents chimiques, biologiques ou de radiations).



Figure 4 : Application Militaires. [CHA 2008]

- **Applications domestiques** : En plaçant, sur le plafond ou dans le mur, des capteurs, on peut économiser l'énergie en gérant l'éclairage ou le chauffage en fonction de la localisation des personnes. Comme ils peuvent détecter la fumée.



Figure 5 : Application domestique. [CHA 2008]

- **Applications environnementales** : Les réseaux de capteurs sont beaucoup appliqués dans ce domaine pour détecter des incendies, surveiller des catastrophes naturelles, détecter des pollutions et suivre des écosystèmes.



Figure 6: Application environnementale. [ISA 2007]

- **Applications agricoles** : Dans les champs agricoles, les capteurs peuvent être semés avec les graines. Ainsi, les zones sèches seront facilement identifiées et l'irrigation sera donc plus efficace.



Figure 7 : Applications agricoles. [ISA 2007]

- **Applications médicales** : Les réseaux de capteurs ont aussi des développements dans le domaine du diagnostic médical. Par exemple, des micro-caméras sont capables, sans avoir recours à la chirurgie, de transmettre des images de l'intérieur d'un corps humain avec une autonomie de 24 heures. Et ils peuvent aussi détecter des comportements anormaux (chute d'un lit, choc, cri, etc.) chez les personnes dépendantes (handicapées, bébé ou âgées).

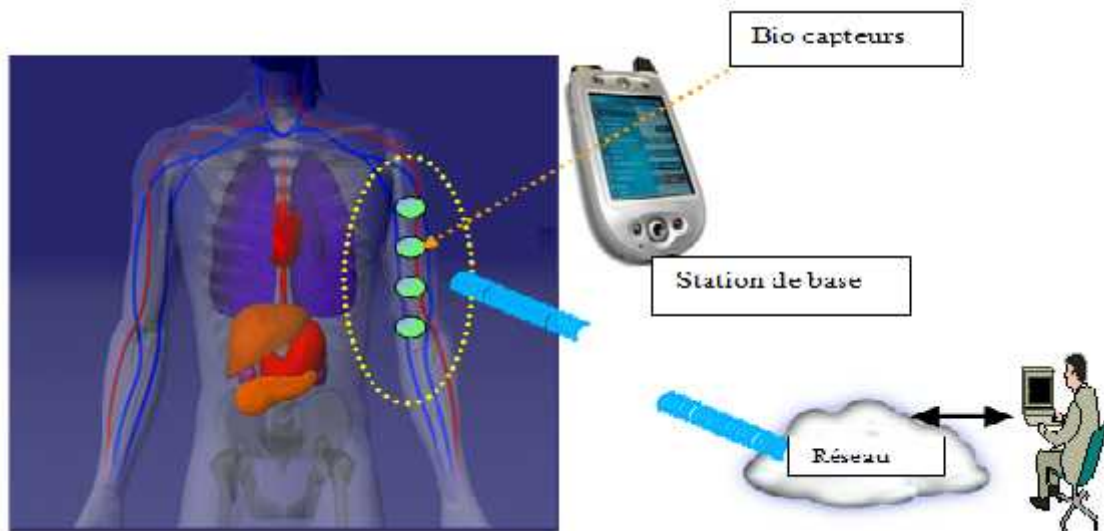


Figure 8 : Application médicale. [DJA 2008]

- **Applications transporté** : Il est possible d'intégrer des nœuds capteurs au 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.



Figure 9 : Application transporté. [DJA 2008]

III.6. Architecture de communication dans les réseaux de capteurs [KRA 2012]

Le modèle de communication comprend cinq couches qui ont les mêmes fonctions que celles du modèle OSI ainsi que trois couches pour la gestion d'énergie, la gestion de la mobilité et la gestion des tâches.

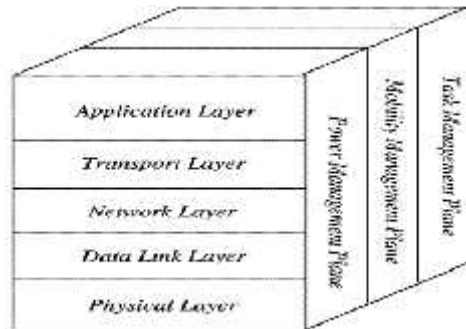


Figure 10 : Architecture de communication dans les réseaux de capteurs. [BOU 2007]

➤ Rôles des couches :

- **Couche physique** : Matériels pour envoyer et recevoir les données.
- **Couche liaison de données** : Gestion des liaisons entre les nœuds et les stations de base, contrôle d'erreurs.
- **Couche réseau** : Routage et transmission des données.
- **Couche transport** : Transport des données, contrôle de flux.
- **Couche application** : Interface pour les applications au haut niveau.
- **Plan de gestion d'énergie** : Contrôle l'utilisation d'énergie.
- **Plan de gestion de mobilité** : Gestion des mouvements des nœuds.
- **Plan de gestion de tâche** : Balance les tâches entre les nœuds afin d'économiser de l'énergie.

III.7. Topologies des réseaux de capteurs [BEN 2012]

• Topologie en étoile

Dans cette topologie une station de base envoie ou reçoit un message via un certain nombre de nœuds. Ces nœuds peuvent seulement envoyer ou recevoir un message de l'unique station de base, il ne leur est pas permis de s'échanger des messages.

Avantage : simplicité et faible consommation d'énergie des nœuds, moindre latence de communication entre les nœuds et la station de base.

Inconvénient : la station de base est vulnérable, car tout le réseau est géré par un seul nœud.

- **Topologie « en toile » ou « en grille »**

Dans ce cas (dit « communication multi-sauts »), tout nœud peut échanger avec n'importe quel autre nœud du réseau (s'il est à portée de transmission). Un nœud voulant transmettre un message à un autre nœud hors de sa portée de transmission, peut utiliser un nœud intermédiaire pour envoyer son message au nœud destinataire.

Avantage: Possibilité de passer à l'échelle du réseau, avec redondance et tolérance aux fautes.

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

- **Topologie hybride**

Une topologie hybride entre celle en étoile et en grille fournit des communications réseau robustes et diverses, en assurant la minimisation de la consommation d'énergie dans les réseaux de capteurs. Dans ce type de topologie, les nœuds capteur autonome en énergie ne routent pas les messages, mais il y a d'autres nœuds qui ont la possibilité de faire le routage des messages. En général, ces nœuds disposent d'une source d'énergie externe.

III.8. Types de réseaux de capteurs sans fil [BEN 2012]

- **Réseaux de poursuite**

Ces réseaux sont généralement développés par l'armée, ils peuvent servir à surveiller toutes les activités d'une zone stratégique ou d'accès difficile, ainsi on pourra détecter des agents chimiques, biologiques ou des radiations avant l'envoi des troupes. On peut aussi penser à des capteurs embarqués sur les soldats pour faciliter leur guidage et le contrôle de leur position depuis la base.

- **Réseaux de collection des données d'environnement**

Les nœuds de ce type de réseau peuvent avoir plusieurs fonctionnalités et différents types de capteurs. Ce type de réseau nécessite généralement un flux de données faible, une durée de vie importante, il sert à la collecte périodique des données environnementales puis leur transmission vers la station de base.

- **Réseaux de surveillance et sécurité**

La différence entre ce réseau et le réseau de collection d'environnement est que les nœuds ne transmettent pas l'ensemble des données collectées mais seulement les rapports concernant une violation de la sécurité. Ce sont en général des nœuds fixes qui contrôlent d'une façon continue la détection d'une anomalie dans le fonctionnement d'un système. Ainsi les altérations dans la structure d'un bâtiment, suite à un séisme, pourraient être détectées par des capteurs intégrés dans les murs ou dans le béton, sans alimentation électrique ou autres connexions filaires.

III.9. Les systèmes d'exploitation pour les réseaux de capteurs [ELM]

Les systèmes d'exploitation pour capteurs en réseau sont des systèmes d'exploitation embarqués au sein des capteurs.

Les systèmes d'exploitation pour capteurs en réseau sont spécifiquement conçus pour optimiser l'usage des ressources matérielles limitées dont ils disposent : peu de mémoire RAM, une vitesse de traitement processeur faible et peu d'énergie électrique. De nombreux systèmes d'exploitation spécialisés existent, parmi lesquels : Contiki, ERIKA Enterprise, Nano-RK, TinyOS, MantisOS, RETOS, Senses, Cormos, LiteOS, NanoQplus.

- **TinyOS**

TinyOS est un système d'exploitation open-source, intégré, modulaire destiné aux réseaux de capteurs sans fil. Cette plate-forme logicielle ouverte et une série d'outils développés par l'université de Berkeley est enrichie par une multitude d'utilisateurs. Il respecte une architecture basée sur une association de composants, réduisant la taille du code nécessaire à sa mise en place. Cela s'inscrit dans le respect des contraintes de mémoires qu'observent les réseaux de capteurs.

La bibliothèque de composant de TinyOS est particulièrement complète puisqu'on y retrouve des protocoles réseaux, des pilotes de capteurs et des outils d'acquisition de données.

L'ensemble de ces composants peut être adapté à une application précise.

En s'appuyant sur un fonctionnement événementiel, TinyOS propose à l'utilisateur une gestion très précise qui permet de mieux s'adapter à la nature aléatoire de la communication sans fil entre interfaces physiques.

- **Mantis**

D'après le site officiel, Mantis OS est un système d'exploitation multithread développé en C, langage choisi pour son efficacité et sa portabilité. Mantis dispose d'un environnement de développement Linux et Windows. Il peut être déployé sur de nombreuses plateformes, tels que MICA2, MICAz, et TELOS. Son empreinte mémoire est faible : 500 octets en RAM et 14 ko en flash. Mantis, apparu en 2005, a été conçu par l'université du Colorado avec comme objectif d'offrir un système d'exploitation multi threading. C'est un système modulaire dont le noyau supporte également les entrées/sorties synchrones, et un ensemble de primitives de concurrence. L'économie d'énergie est réalisée dans Mantis par une fonction de veille (sleep function) qui désactive le capteur lorsque tous les threads actifs sont terminés. Mantis est un système dynamique, les modifications applicatives peuvent être réalisées en fonctionnement. Mantis apporte une compatibilité avec le modèle événementiel TinyOS à travers TinyMOS (MOS est la contraction de MantisOS), dont son noyau est équipé. Comme la plupart des systèmes d'exploitation pour capteurs, Mantis dispose d'une pile réseau, regroupée dans la couche COMM. La communication est basée sur le principe de messages pondérés Active Message (AM). La couche COMM est dotée du protocole MAC, qui rend le réseau compatible avec de nombreux périphériques, de la gestion de mémoire tampon des paquets et de fonctions de synchronisation.

Concernant le débogage à distance, Mantis met en œuvre NodeMD, un système de déploiement et de notifications de défauts.

- **Contiki**

Contiki est un système configurable modulaire pour réseaux de capteurs. Contiki est organisé en modules, basé sur une architecture plate. Un cœur non-reconfigurable permet de télécharger les modules applications ou sous-systèmes, qui constituent alors l'unité de reconfiguration dans Contiki. Un modèle d'exécution événementiel permet une implémentation efficace d'un état stable pour les modules. Contiki est un OS conçu pour prendre le moins de place possible, avec une faible empreinte mémoire. Pour cela, le code est écrit en C. Ainsi, le système complet est supposé pouvoir tourner sans problèmes avec 2 ko de RAM et 40 ko de ROM. Contiki possède une gestion de la programmation parallèle sous forme de « proto-threads », qui sont des processus légers développés spécialement pour l'occasion de même que μ IPV6 permet d'être compatible IPV6.

Dans Contiki il existe une bibliothèque spécifique MEMB () contenue dans le noyau pour gérer la mémoire par block et réorganiser l'espace mémoire libre, cette gestion dynamique, permet d'optimiser la consommation énergétique. Contiki fournit une implémentation RPL, appelée Contiki RPL, qui utilise IPV6 utilisant le protocole de routage RPL qui augmente la durée de vie des batteries des capteurs et permet de fonctionner sur des liens sans fil de faible puissance et des liens avec perte de puissance en ligne.

III.10. Technologie utilisés

Les médias de transmission : Dans un réseau de capteurs, les nœuds sont reliés par une architecture sans-fil. Pour permettre des opérations sur ces réseaux (circulation des données) au niveau mondial, les médias de transmission doivent être normés.

Les médias les plus utilisés sont : le Bluetooth, les communications radio ZigBee et Dash7.

- **ZIGBEE/ IEEE 802.15.4 [ZIG]**

ZigBee est une norme de transmission de données sans fil permettant la communication de machine à machine. Zigbee offre des débits de données moindres, mais sa très faible consommation d'énergie et ses coûts de production très bas en font d'elle la candidate idéale pour la domotique ou les matériels de type capteur



Figure 11 : Zigbee/IEEE 802.15.4.

- **Bluetooth / IEEE 802.15.4 [JIM 2014]** 

Bluetooth est un standard de communication permettant l'échange bidirectionnel de données à très courte distance et utilisant des ondes radio. Son objet est de simplifier les connexions entre les appareils électroniques en supprimant des liaisons filaires. Malheureusement, un grand défaut de cette technologie est sa trop grande consommation d'énergie.

- **Dash 7 / ISO/IEC 18000-7 [WIK Web]**

Dash7 est une nouvelle technologie de réseaux de capteurs sans fil en utilisant la norme ISO/IEC 18000-7. Sa consommation électrique est très faible, la durée de vie de batterie peut attendre plusieurs années. Sa distance de communication est 2km. Elle fournit une faible latence pour le suivi des objets en mouvement, un protocole petite pile, des supports de capteurs et de sécurité et un débit de transmission allant jusqu'à 200kbits/s.

Conclusion

Dans ce chapitre, nous avons présenté les réseaux de capteurs sans fil, en évoquant leurs architectures, leurs composants, les technologies et topologies utilisés, leurs caractéristiques et limites, ainsi que les applications qui sont initialement réservées pour le domaine militaires, aujourd'hui, les WSNs ont réussi à conquérir d'autre domaine civils plus larges et plus pratiques changeant le quotidien des être humains.

Dans le chapitre suivant, nous allons présenter les différentes applications de détection de présence.

Introduction

La fonction essentielle des applications de détection de présence consiste à renseigner le contrôleur sur la présence ou non d'un objet ainsi que son type dans le domaine à surveillé. Pour lui permettre de réagir rapidement et efficacement.

I. Historique

En 1992/1993, l'entreprise HTS High Technology Systems SA développe le premier détecteur de présence pour la commande de l'éclairage et de la ventilation dans le but de réduire la consommation d'énergie dans les bâtiments. Au cours d'études prolongées sur le terrain, elle parvient à prouver que l'utilisation des détecteurs de présence permet de réaliser des économies atteignant 40% dans le secteur de l'éclairage et jusqu'à 70% dans le secteur de la ventilation. Compte tenu de l'énorme potentiel avéré des produits dédiés à l'économie d'énergie.

En 2007 HTS fusionne avec Theben Suisse pour former la société Theben HTS AG qui fournit sur le plan mondial des produits et des solutions au service de l'efficacité énergétique, du confort et de la sécurité dans la gestion technique des bâtiments. Et depuis, le monde a vu naître plusieurs entreprises qui se spécialisent dans la fabrication des détecteurs de présence [THE Web]

II. Principe de la détection de présence

Un détecteur de présence est un dispositif autrefois mécanique, et aujourd'hui de plus en plus caractérisé par l'absence de liaison mécanique entre le dispositif mécanique et l'objet cible (personne, animal ou objet animé tel qu'un véhicule, ..., etc.).

Son rôle est de détecter des présences anormales dans un environnement. Cet appareil peut avoir plusieurs fonctions. Tout d'abord, il permet de sécuriser un lieu en repérant toute présence anormale, et ainsi d'éviter des vols ou des intrusions.

Différentes techniques peuvent être utilisées, par exemple, le détecteur de présence peut être sensible à la chaleur, au son, à la luminosité, et au volume, ..., etc. [THE Web]

III. Les différentes techniques de détection de présence

Il existe un grand nombre de techniques pour la détection d'objet dans une image ou une vidéo numérique, répondant aux différentes contraintes des applications. Certaines peuvent être combinées entre elles (taux de CO₂, infrarouge et photoélectrique, ..., etc.). Dans le cas où l'on maîtrise le système d'acquisition de l'image ou de la vidéo, on peut choisir un système de stéréovision, un système d'imagerie à grande gamme dynamique, ou travailler dans le domaine infrarouge avec une caméra thermique. Les techniques mises en œuvre lorsque l'on recherche une personne, un animal ou un véhicule dans une image existante, sans aucune information a priori sur le contenu, sont généralement différentes.

III.1. Détecteur par le taux de CO₂

Le taux de CO₂ est utilisé beaucoup plus pour améliorer la qualité de l'aire ambiante à l'intérieur des maisons, bureaux, les classes, ..., etc. Mais peut être utilisé pour la détection de présence dans les chambres fortes. En effet, la présence des personnes augmente le taux de CO₂ dans la chambre suite à la respiration de ou des l'intrus, ce qui engendre la réaction du système.



Dans cette catégorie on mentionne : capteur de CO₂ IAQ-100 [DET Web]

III.2. Détecteur pas des caméras thermiques

La caméra thermique est l'une des techniques les plus utilisées pour la détection de présence. Leur principe est de calculer un certain nombre de caractéristiques à partir des pixels d'images, elle prend en considération les formes des mouvements, les couleurs, et surtout, basé sur la chaleur. Elle reproduit la chaleur emmagasinée d'un corps. La couleur produite par la caméra est une fausse couleur. Cette dernière est obtenue en associant une couleur à l'intensité reçue (captée) afin de faciliter la lecture des images.

Et dans ce domaine on cite l'entreprise SORHEA (camera SORH10 IP). [SOR Web]



Figure 12: Image par camera thermique [SWI Web]

Cette figure représente l'image infrarouge de la partie supérieure du corps humain. L'échelle de température et de couleur montre la relation entre la couleur et la température de surface correspondante.

III.3. Détecteur par le laser

Le laser permet une détection à longue portée. Il existe plusieurs procédés de détection par laser tel que :

- Généralement, il est constitué de deux miroirs dont l'un est réfléchissant. C'est-à-dire une partie de la lumière sort de la cavité et l'autre partie est réinjectée vers l'intérieur de la cavité laser. Ainsi, s'il y a un objet qui passe à travers, le signal se coupe (le rayonnement est coupé), le capteur détecte alors la présence d'un objet.

On cite dans ce domaine l'entreprise SafeZoneTM détecteur par laser. [ABC Web]

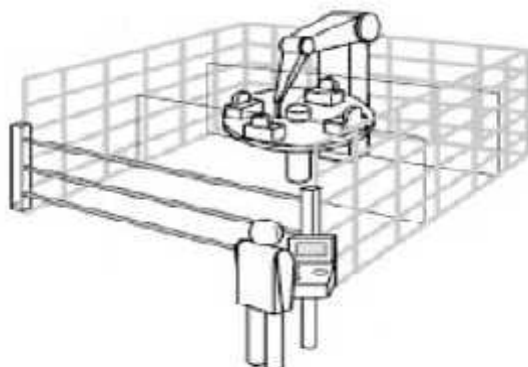


Figure 13: détecteur par laser [ABC Web]

- Le détecteur par laser fonctionne sur le principe de la réflexion diffuse et de la vitesse de la lumière réfléchi. En effet, des rayons infrarouges sont diffusés. La réflexion de la lumière diffusée est reçue à la suite d'une intrusion dans le champ de détection préconfiguré et est traitée afin de déterminer la position et la distance de l'objet par rapport au détecteur.

A l'instant où la lumière infrarouge est émise, un chronomètre électronique démarre. La lumière réfléchi est ensuite détectée par le détecteur et analysée afin de déterminer la distance de l'objet par rapport au détecteur. Dans cette catégorie on spécifie les Adaptateur détecteur de présence « Komfort » [GIR Web]

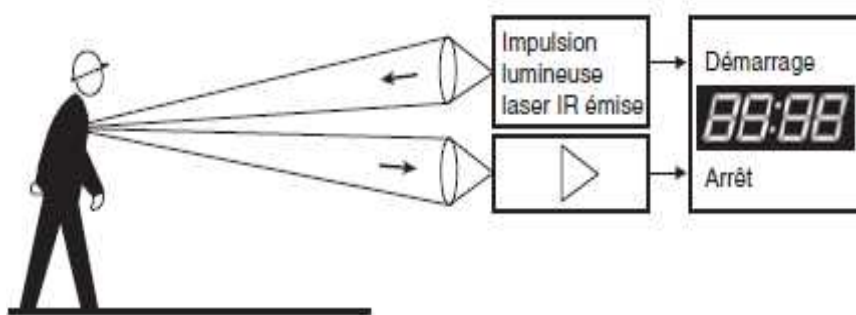


Figure 14: Schéma pour détecteur laser [ABC Web]

III.4. Détecteur par infrarouge

La technologie infrarouge permet de détecter la présence. Le détecteur infrarouge diffuse des rayons infrarouges permanents dans la zone de surveillance et dès qu'un objet rentre dans cette zone, un nombre de rayons sera coupé et ce nombre sera traité donnant ainsi le volume de l'objet détecté.

Parmi les marques les plus réputés dans ce domaine on a Gamme Busch-Guard « Gamme Busch-Guard Présence tech KNX ». [INF Web]



Figure 15 : détection de présence par infrarouge [NIK Web]

III.5. Technologie à ultrasons (US)

La détection par ultrasons envoie des ondes d'ultrasons à travers une zone de détection et mesure la vitesse à laquelle ces ondes reviennent. Les mouvements des personnes présentes, aussi infimes soient-ils, modifient la fréquence à laquelle ces ondes reviennent à leur point d'origine. Toute présence est ainsi détectée, même à une grande distance. Dans ce domaine, Génération Robots propose « Lego Mindstorms NXT ». [ULT Web]



Figure 16: détection par ultrason [NIK Web]

III.6. Technologie Dual (Infrarouge et ultrason)

Les capteurs Dual Tech associent les avantages de l'infrarouge et de la technologie à ultrason. Ils garantissent ainsi une détection optimale de l'activité humaine dans un lieu. Les capteurs Dual Tech sont synonymes de fiabilité maximale et de grand potentiel d'économies

d'énergie. Détecteur de présence Dual Tech 360°, 230 V, 1 canal, PIR + US, maître. [DUA Web]



Figure 17: Détection par infrarouge/ultrason [NIK Web]

III.7. Détecteur photoélectrique

Un détecteur photoélectrique réalise la détection d'une cible (objet ou personne) au moyen d'un faisceau lumineux. Ses deux constituants de base sont un émetteur et un récepteur de lumière.

La détection est effective quand la cible pénètre dans le faisceau lumineux et modifie suffisamment la quantité de lumière reçue par le récepteur pour provoquer un changement d'état de la sortie.

Elle s'effectue selon deux procédés :

- blocage du faisceau par la cible,
- renvoi du faisceau sur le récepteur par la cible.

Donc ce domaine on distingue le label « Schneider Electric capteur photoélectrique ». [SCH 2003]

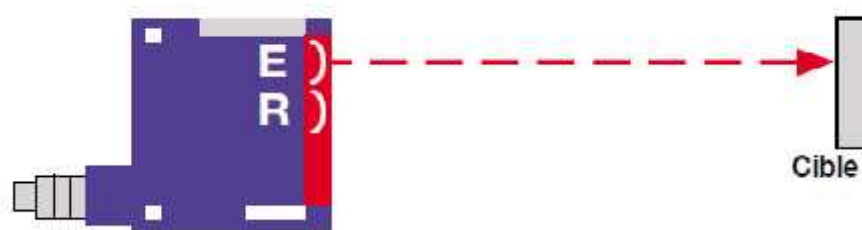


Figure 18: Faisceau bloqué par la cible. [SCH 2003]

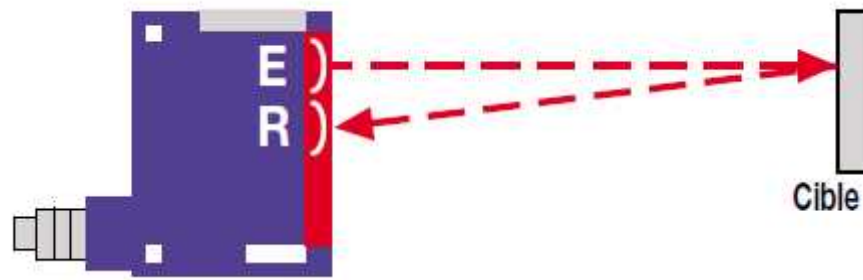


Figure 19: Faisceau renvoyé par la cible. [SCH 2003]

IV. Présentation de notre application

Notre application consiste à simuler la détection de présence. On a opté pour l'utilisation de capteur à infrarouge basé sur le volume, qui consiste à informer le contrôleur sur toutes les présences soient animales, humaines ou des véhicules.

IV.1 Zone à surveillé

Notre application consiste à détecter toute présence au tour d'une usine de manière à signaler la présence d'animaux, d'humains ou des véhicules en temps réel.

IV.2. Les outils utilisés

Dans cette partie, nous allons présenter les différents outils utilisés pour réaliser notre application de simulation.

IV.2.1. Le détecteur infrarouge basé sur le volume

Les détecteurs infrarouges sont des capteurs dotés de l'infrarouge, qui sont des rayons diffusés autour du capteur. L'introduction d'un objet dans la zone couverte engendre l'interruption d'un certain nombre de rayons, et suite à ce nombre, le nœud capteur définit le volume de l'objet capté et spécifie suite à ce volume la nature de l'objet détecté : animal, humain ou véhicule selon les valeurs suivantes :

Objet	Volume
Animal	[100 ; 400]
Humain	[401 ; 800]
Véhicule	>801

IV.2.1. Le protocole de routage utilisé

Le protocole de routage utilisé dans notre application est LEACH (Low Energy Adaptive Clustering Hierarchical). Son architecture de communication consiste, de façon similaire aux réseaux cellulaires, à former des cellules basées sur l'amplitude du signal, et utiliser les têtes de cellules comme routeurs vers le nœud puits. Ces cellules sont appelées groupes (clusters), quant aux têtes : chefs de groupes (cluster-heads CH). Les chefs de groupes sont choisis de façon aléatoire selon un algorithme spécifique d'élection basé sur une fonction de probabilité qui prend en compte différents critères comme l'énergie disponible des nœuds. Comme la figure 20 l'indique, les nœuds sont chargés de collecter des données, les envoyer à leurs CH qui les agrègent et transmettent à leur tour, les résultats d'agrégation au nœud puits selon une communication unicast (à un seul saut).

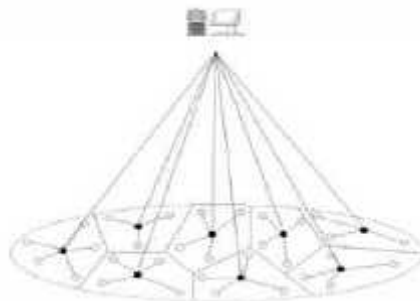


Figure 20: Architecture de communication de protocole LEACH [SAM 2006].

Les CH ont pour mission d'assurer les fonctions les plus coûteuses en énergie, à savoir la communication avec le nœud puits qui est supposé éloigné, ainsi que tous les traitements de données (agrégation, fusion et transmission de données) afin de réduire la quantité des données transmises. Ce dispositif permet d'économiser l'énergie puisque les transmissions sont uniquement assurées par les CH plutôt que par tous les nœuds du réseau. Par conséquent, LEACH réalise une réduction significative de la dissipation d'énergie [SAM 2006].

➤ Algorithme de protocole LEACH

L'algorithme se déroule en « rounds » qui ont approximativement le même intervalle de temps déterminé au préalable. Chaque round est constitué d'une phase d'initialisation et d'une phase de transmission.

• Phase d'initialisation

Comme l'indique la figure 21, la phase d'initialisation est composée de 3 sous phases: Annonce, Organisation des groupes et enfin Ordonnancement, et qui seront détaillées ci-dessous.

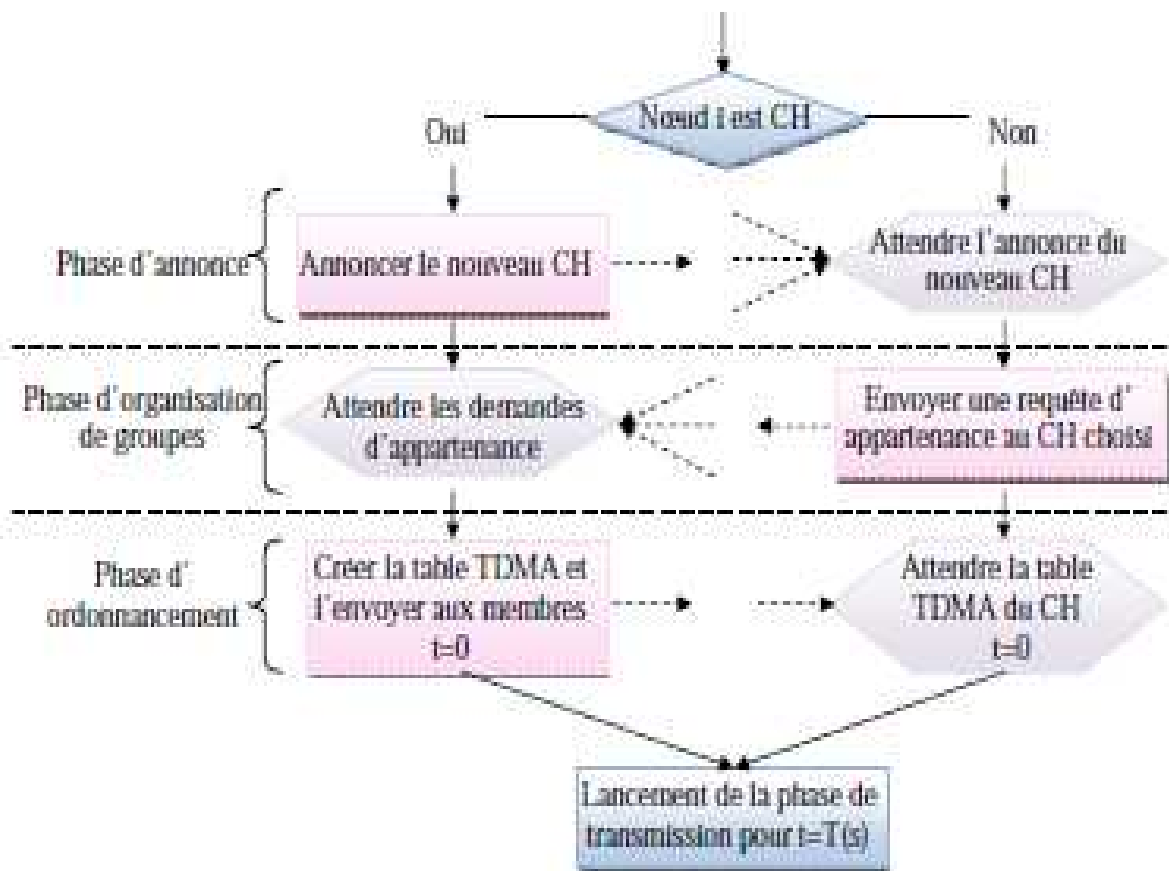


Figure 21: Opérations de l'étape d'initialisation de LEACH [SAM 2006].

✓ Phase d'annonce

Avant de lancer cette phase, on désire avoir un certain nombre de CH. Ce nombre, noté K , est fixe et il est inchangé durant tous les rounds. On estime que le pourcentage optimal du nombre de CH désirés devrait être de 5% à 15% du nombre total de nœuds [DJA 2008]. Si ce pourcentage n'est pas respecté, cela mènera à une

grande dissipation d'énergie dans le réseau. En effet, si le nombre de CH est très élevé, on aura un nombre important de nœuds (CH) qui se consacrent aux tâches très coûteuses en ressources énergétiques. Ainsi, on aura une dissipation d'énergie considérable dans le réseau. De plus, si le nombre de CH est très petit, ces derniers vont gérer des groupes de grandes tailles. Ainsi, ces CH s'épuiseront rapidement à cause du travail important qui leur est demandé.

Cette phase commence par l'annonce du nouveau round par le nœud puits, et, par la prise de décision locale d'un nœud pour devenir CH avec une certaine probabilité $P_i(t)$ au début du round $r+1$ qui commence à l'instant t . Chaque nœud i génère un nombre aléatoire entre 0 et 1. Si ce nombre est inférieur à $P_i(t)$, le nœud deviendra CH durant le round $r+1$. $P_i(t)$ est calculé en fonction de K et de round r : [WEN 2000]

$$\text{Nombre(CH)} = \sum_{i=1}^N P_i(t) = K$$

Où N est le nombre total de nœuds dans le réseau. Si on a N nœuds et K CH, alors, il faudra N/K rounds durant lesquels un nœud doit être élu seulement une seule fois autant que CH avant que le round soit réinitialisé à 0. Donc la probabilité de devenir CH pour chaque nœud i est :

$$P_i(t) = \frac{\text{Le nombre de CH désirés}}{\text{Le nombre de noeud qui n ontpas encore été élus CH durant les r rounds précédents}}$$

$$P_i(t) = \begin{cases} \frac{K}{N - K * (r \bmod \frac{N}{K})} & : C_i(t)=1 \quad \dots (1) \\ 1 & : C_i(t)=0 \end{cases}$$

Où $C_i(t)$ égal à 0 si le nœud i a déjà été CH durant l'un des $(r \bmod N/K)$ rounds précédents, et, il est égal à 1 dans le cas contraire. Donc, seuls les nœuds qui n'ont pas encore été CH, ont vraisemblablement une énergie résiduelle suffisante que les autres et ils pourront être choisis.

Le terme $\sum_{i=1}^N C_i(t)$ représente le nombre total des nœuds éligibles d'être CH à l'instant t . Il est égal à : $\sum_{i=1}^N C_i(t) = N - K * (r \bmod N/K) \quad \dots (2)$

Utilisant l'équation (1) et (2), le nombre de CH par round est :

$$\text{Nombre (CH)} = \sum_{i=1}^N C_i(t) * P_i(t) = (N - K * (r \bmod N/K)) * \left(\frac{K}{N - K * (r \bmod \frac{N}{K})} \right) = K$$

La probabilité $P_i(t)$ est basée sur la supposition que tous les nœuds sont initialement homogènes et commencent avec la même quantité résiduelle d'énergie et meurent approximativement en même temps. Cependant, ceci pourrait être le cas juste après le déploiement, mais il n'est pas réellement valable après un certain temps. Alors, si l'énergie des nœuds diffère, il sera plus pratique que la probabilité $P_i(t)$ soit en rapport avec l'énergie restante au niveau de chaque nœud. Cette probabilité sera donc égale à :

$$P_i(t) = \frac{E_i(t)}{E_{total}(t)} K \quad \dots\dots(3)$$

Où $E_i(t)$ est l'énergie résiduelle relative à chaque nœud i . Utilisant cette probabilité, le nœud avec une plus grande ressource d'énergie a une plus grande chance de devenir CH. Ainsi, le nombre de nœuds souhaités pour être CH dans chaque round est:

$$\begin{aligned} \text{Nombre (CH)} &= \sum_{i=1}^N C_i(t) * P_i(t) = \left(\frac{E_1(t)}{E_{total}(t)} + \frac{E_2(t)}{E_{total}(t)} \right. \\ &+ \dots + \left. \frac{E_n(t)}{E_{total}(t)} \right) K = K \end{aligned}$$

Les équations (2) et (3) seront égales si les nœuds commencent avec la même énergie. De plus, en utilisant l'équation (3), les nœuds requièrent des informations sur toute l'énergie disponible dans le réseau.

✓ **Phase d'organisation de groupes**

Après qu'un nœud soit élu CH, il doit informer les autres nœuds non-CH de son nouveau rang dans le round courant. Pour cela, un message d'avertissement ADV contenant l'identificateur du CH est diffusé à tous les nœuds non-CH en utilisant le protocole MAC CSMA pour éviter les collisions entre les CH. La diffusion permet de s'assurer que tous les nœuds non-CH ont reçu le message. Par ailleurs, elle permet de garantir que les nœuds appartiennent au CH qui requière le minimum d'énergie pour la communication. La décision est basée donc sur l'amplitude du signal reçu; le CH

ayant le signal le plus fort (i.e. le plus proche) sera choisi. En cas d'égalité des signaux, les nœuds non-CH choisissent aléatoirement leur CH [WEN 2000]. Chaque membre informe son CH de sa décision. Une fois que le CH ait reçu la demande, il lui envoie un message d'acquittement Join-REQ.

✓ Phase d'ordonnement

Après la formation des groupes, chaque CH agit comme un centre de commande local pour coordonner les transmissions des données au sein de son groupe. Il crée un ordonnanceur (schedule) TDMA et assigne à chaque nœud membre un slot de temps durant lequel il peut transmettre ses données. L'ensemble des slots assignés aux nœuds d'un groupe est appelé frame. La durée de chaque frame diffère selon le nombre de membres du groupe.

Par ailleurs, afin de minimiser les interférences entre les transmissions dans des groupes adjacents, chaque CH choisit aléatoirement un code dans une liste de codes de propagation CDMA. Il le transmet par la suite à ses membres afin de l'utiliser pour leurs transmissions. [SAC 2004]

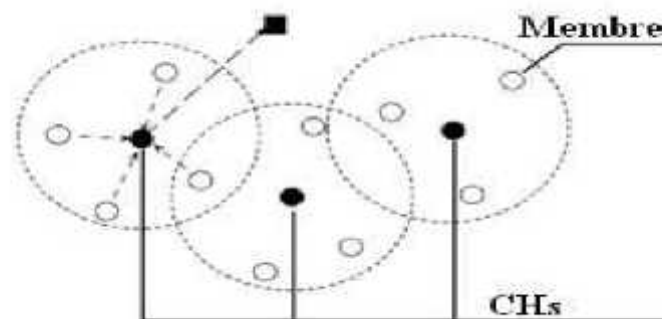


Figure 22: Interférence lors d'une communication dans LEACH [SAC 2004]

Cette phase est plus longue que la phase précédente, et permet la collecte de données captées. En utilisant l'ordonnanceur TDMA, les membres émettent leurs données captées pendant leurs propres slots. Cela leur permet d'éteindre leurs interfaces de communication en dehors de leurs slots afin d'économiser leur énergie. Ces données sont ensuite agrégées par les CH qui les fusionnent et les compressant, et, envoient le résultat final au nœud puits.

Après un certain temps prédéterminé, le réseau va passer à un nouveau round. Ce processus est répété jusqu'à ce que tous les nœuds du réseau seront élus CH, une seule fois, tout au long des rounds précédents. Dans ce cas, le round est réinitialisé à 0 [YAS 2007].

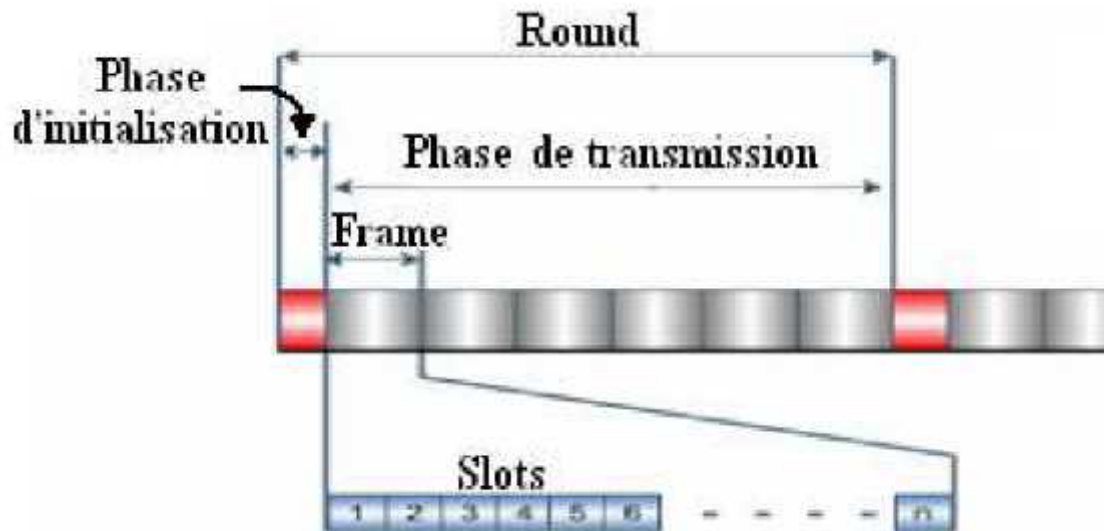


Figure 23: Répartition du temps et différentes phases pour chaque round [YAS 2007]

➤ Avantages et inconvénients de LEACH

Le protocole LEACH engendre beaucoup d'avantages en ce qu'il offre comme bonne manipulation de ressources du réseau en respectant plusieurs contraintes telle que la consommation d'énergie.

Bien que LEACH économise la consommation d'énergie par un facteur de 8 [LNA 2004] comparé à la transmission directe, grâce à l'agrégation de données et la réutilisation de largeur de bande, un nombre d'inconvénients restent plus ou moins apparents.

Dans ce qui suit, on cite quelques avantages et inconvénients du protocole LEACH.

• Avantages

- ✓ **Protocole auto-organisateur basé sur le groupement adaptatif:** LEACH est complètement distribué, autrement dit, les nœuds prennent leurs décisions de façon autonome et agissent de manière locale et n'ont pas besoin d'une information globale ni d'un système de localisation pour opérer de façon efficace. De plus, la collection de données est faite périodiquement (l'utilisateur n'a pas besoin de toutes les données immédiatement). Pour exploiter cette caractéristique, ce protocole introduit un groupement adaptatif, c'est-à-dire, il réorganise les

groupes après un intervalle de temps aléatoire, en utilisant des contraintes énergétiques afin d'avoir une dissipation d'énergie uniforme à travers tout le réseau. [SRA 2003]

- ✓ **Rotation des rôles de chefs de groupes:** La rotation des rôles de chefs de groupes s'avère un facteur important pour l'organisation des nœuds. Ce rôle est épuisant en termes de consommation d'énergie car les CH sont actifs tout au long de leur élection. Puisque le nœud puits est généralement loin du champ de surveillance, les CH diffusent une quantité plus importante d'énergie pour lui transmettre leurs données. Donc, si les CH sont choisis d'une manière fixe, leur énergie s'épuisera rapidement ce qui induit à leur défaillance. Par conséquent, tous les autres nœuds seront sans CH et donc inutiles. C'est pourquoi, les algorithmes de groupement (clustering) étudiés jusqu'ici adoptent la rotation du rôle de chefs de groupes. [YAS 2007]
- ✓ **Faible énergie pour l'accès au média:** Le mécanisme de groupes permet aux nœuds d'effectuer des communications sur des petites distances avec leurs CH afin d'optimiser l'utilisation du média de communication en la faisant gérer localement par un CH pour minimiser les interférences et les collisions.
- ✓ **Compression locale (agrégation) :** Les CH compressent les données arrivant de leurs membres, et envoient un paquet d'agrégation au nœud puits afin de réduire la quantité d'informations qui doit lui être transmise. Cela permet de réduire la complexité des algorithmes de routage, de simplifier la gestion du réseau, d'optimiser les dépenses d'énergie et enfin de rendre le réseau plus évolutif (scalable).
- **Inconvénients**
 - ✓ On pourra ne pas avoir des CH durant un round si les nombres aléatoires générés par tous les nœuds du réseau sont supérieurs à la probabilité $P_i(t)$.
 - ✓ Les nœuds les plus éloignés du CH meurent rapidement par rapport aux plus proches.
 - ✓ L'utilisation d'une communication à un seul saut au lieu d'une communication multi-sauts diminue l'énergie des nœuds.
 - ✓ La rotation des CH permet de ne pas épuiser les batteries. Cependant, cette méthode n'est pas efficace pour de grandes structures de réseaux à cause de la

surcharge d'annonces engendrées par le changement des CH, et qui réduit le gain d'énergie initial.

- ✓ Il n'est pas évident que les CH soient uniformément distribués. Donc, il est possible que les CH puissent être concentrés dans une partie du réseau. Par conséquent, certains nœuds n'auront pas des CH dans leurs voisinages.
- ✓ Le protocole LEACH n'est pas sécurisé. Aucun mécanisme de sécurité n'est intégré dans ce protocole. Ainsi, il est très vulnérable même aux simples attaques. Donc, un attaquant peut facilement monopoliser le réseau et induit à son dysfonctionnement. [BER 2009]

Conclusion

Dans ce chapitre nous avons présenté les applications de détection de présence. Nous avons commencé par expliquer leurs principes, les différentes techniques de détection de présence et nous avons présenté notre application et différents outils utilisés.

Dans le chapitre suivant, nous allons présenter la simulation de notre application

Introduction

Les avancées technologiques dans les réseaux de capteurs ont permis de développer les détecteurs de présences de plus en plus performant et intelligent.

Avant concrétisation de notre projet, le déploiement d'un réseau de capteurs nécessite une phase de simulation, afin d'assurer le bon fonctionnement de tous les dispositifs. En effet, pour les grands réseaux, le nombre de capteurs peut atteindre plusieurs milliers et donc un coût financier relativement important. Il faut donc réduire au maximum les erreurs de conception possibles en procédant à une phase de validation.

Dans le cadre du projet, nous avons mis en place une plateforme d'expérimentation qui a pour but de tester, de valider et de simuler le fonctionnement d'un réseau de capteurs. Sa principale fonction est de vérifier le comportement des capteurs développés avant même de les avoir déployé en situation réelle. Et on va présenter les résultats obtenus.

I. Environnement de simulation

Dans cette section, nous présentons les outils utilisés pour la simulation de la présence en commençant tout d'abord par présenter le système d'exploitation utilisé qui est TinyOS, conçu pour les dispositifs à ressources limitées en particulier les RCSF. Nous décrivons ensuite le langage de programmation adapté NesC avec lequel nous avons implémenté le code l'application. Et on fini cette section par la présentation d'un simulateur des RCSF TOSSIM.

I.1. Le système d'exploitation TinyOS [HIL 2000]



TinyOS est un système d'exploitation open-source conçu pour des réseaux de capteurs sans fil.

Il respecte une architecture basée sur une association de composants, réduisant la taille du code nécessaire à sa mise en place. Cela s'inscrit dans le respect des contraintes de mémoires qu'observent les réseaux de capteurs.

Pour autant, la bibliothèque de **composant** de TinyOS est particulièrement complète puisqu'on y retrouve des protocoles réseaux, des pilotes de capteurs et des outils d'acquisition de données. L'ensemble de ces composants peut être utilisé tel quel, il peut aussi être adapté à une application précise.

En s'appuyant sur un fonctionnement événementiel, TinyOS propose à l'utilisateur une gestion très précise de la consommation du capteur et permet de mieux s'adapter à la nature aléatoire de la communication sans fil entre interfaces physiques.

I.2. Le Langage de programmation NesC

NesC est un langage de programmation orienté composants syntaxiquement proche du langage C. Il est conçu pour la réalisation des systèmes embarqués distribués, en particulier, les RCSF.

Grâce à NesC il est possible de créer une application par un assemblage de composant nécessaire à l'application. Chaque composant correspond à un élément matériel (LEDs, timer, ADC ...) et peut être réutilisé dans différentes applications [MAT].

Un composant est constitué alors de trois parties essentielles : interfaces, modules et configurations [DAV 2003].

- **Les interfaces** permettent de spécifier des fonctions : des commandes ou des événements. Ces fonctions sont alors implémentées par le fournisseur ou l'utilisateur de l'interface, afin de distinguer les fonctions concernant les commandes de celles concernant les événements, les fonctions sont précédées de `command` ou `event`.
- **Les modules** sont les éléments de base de la programmation. Ils permettent d'implémenter les composants et sont stockés dans un fichier.
- **les configurations** permettent de décrire l'architecture. Une configuration est donc constituée de modules et/ou d'interfaces ainsi que de la description des liaisons entre ces composants.

II. Les outils de simulation

TinyOs offre différents outils de Simulation tel que : TOSSIM, PowerTOSSIM et TinyViz

II.1. TOSSIM

TOSSIM permet de simuler le comportement d'un capteur au sein d'un réseau de capteurs, il est un simulateur discret basé sur la programmation par événement, de même qu'il est conçu et désigné pour simuler les réseaux de capteurs qui utilisent la plateforme TinyOS. Le principal but de TOSSIM est de créer une simulation très proche de ce qui se passe dans ces réseaux dans le monde réel.

TOSSIM simule le comportement des applications de TinyOS à un niveau très bas. Le réseau est simulé au niveau des bits et chaque interruption dans le système est capturée. TOSSIM fournit deux modèles de radios pour la communication : Le modèle par défaut « simple » où les paquets sont transmis dans le réseau sans aucune erreur et ils sont reçus par chaque nœud. Avec ce modèle il est ainsi possible que deux nœuds différents peuvent envoyer un paquet en même temps avec la conséquence que ces deux paquets seront alors détruits à cause du chevauchement des signaux. Le deuxième modèle est le modèle « lossy », dans ce modèle les nœuds sont placés dans un graphe direct formé d'un couple (a, b) ce qui signifie qu'un paquet envoyé par le nœud a peut être reçu par le nœud b [WAS 2006].

TOSSIM est équipé aussi d'un simulateur graphique TinyViz. Cette application est équipée par plusieurs API plugins qui permettent d'ajouter plusieurs fonctions à notre simulateur comme par exemple contrôler les entrées de notre radio ou bien suivre la dépense d'énergie en utilisant un autre simulateur qui s'appelle PowerTOSSIM [ELB 2011].

II.2. PowerTOSSIM

L'outil PowerTOSSIM permet de faire des simulations de la même manière que TOSSIM sauf que celui-ci prend en considération la consommation d'énergie, ainsi le nœud qui ne possède plus d'énergie s'arrête de fonctionner, ce qui nous permet d'exécuter la simulation jusqu'à la mort du réseau [MEH 2007].

II.3. TinyViz

TinyViz est une application graphique qui donne un aperçu de notre réseau de capteurs à tout instant, ainsi que des divers messages qu'ils émettent. Il permet de déterminer un délai entre chaque itération des capteurs afin de permettre une analyse pas à pas du bon

déroulement des actions, il possède aussi des options afin de pouvoir simuler la consommation d'énergie [FAR 2008].

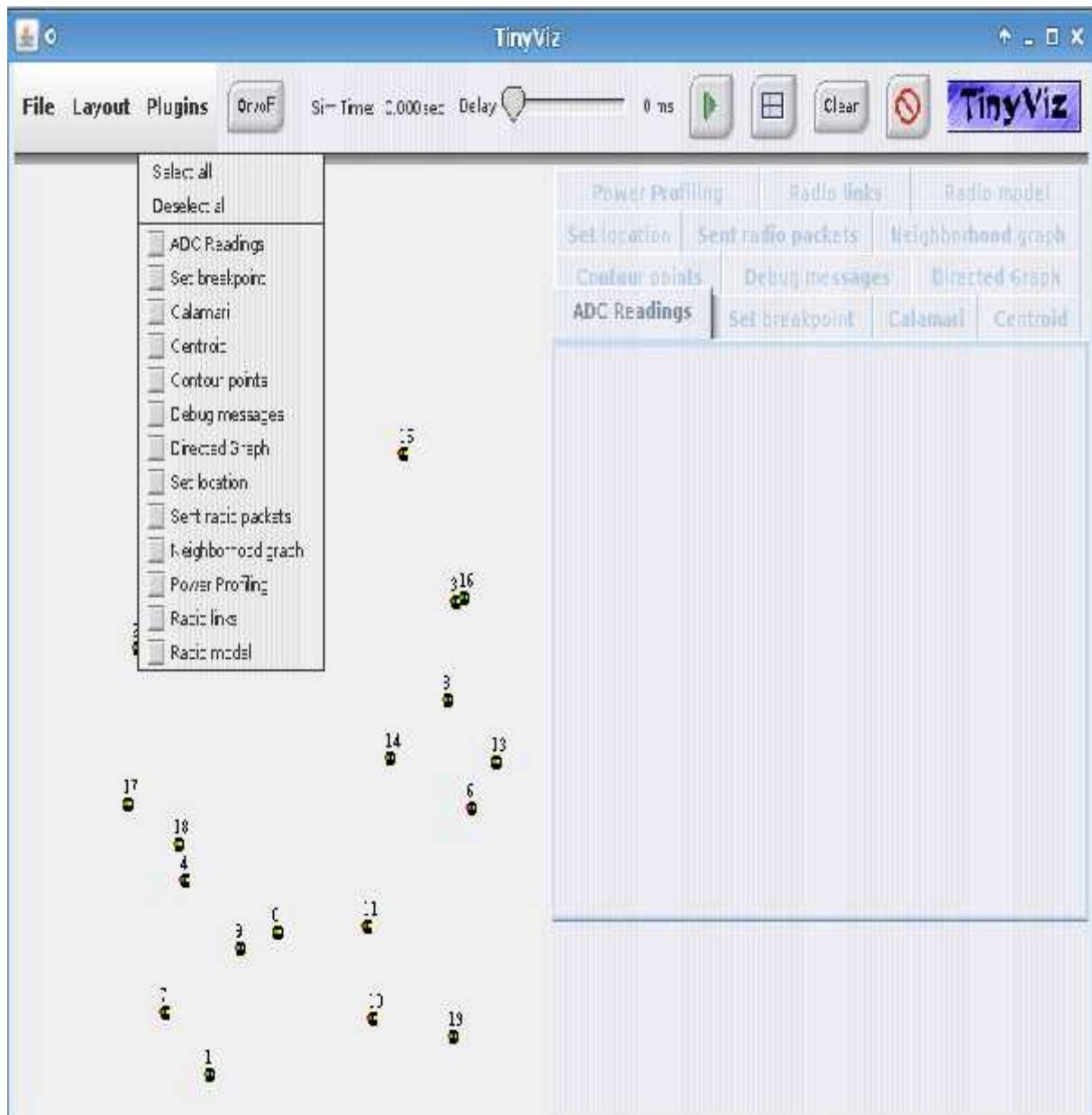
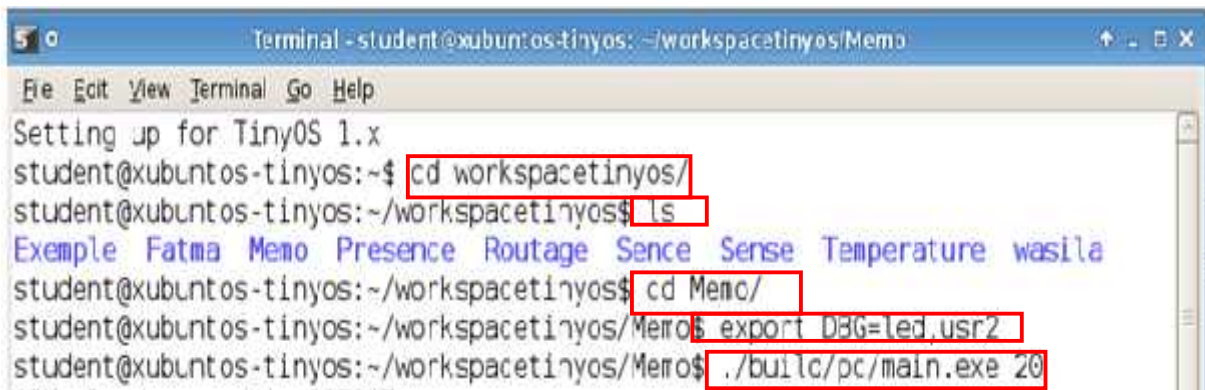


Figure 24: Environnement de simulation TinyViz.

III. Simulation

Après compilation du projet, on ouvre le terminal pour l'exécuter. Avant d'exécuter le projet, il faut y accéder par les commandes suivantes :

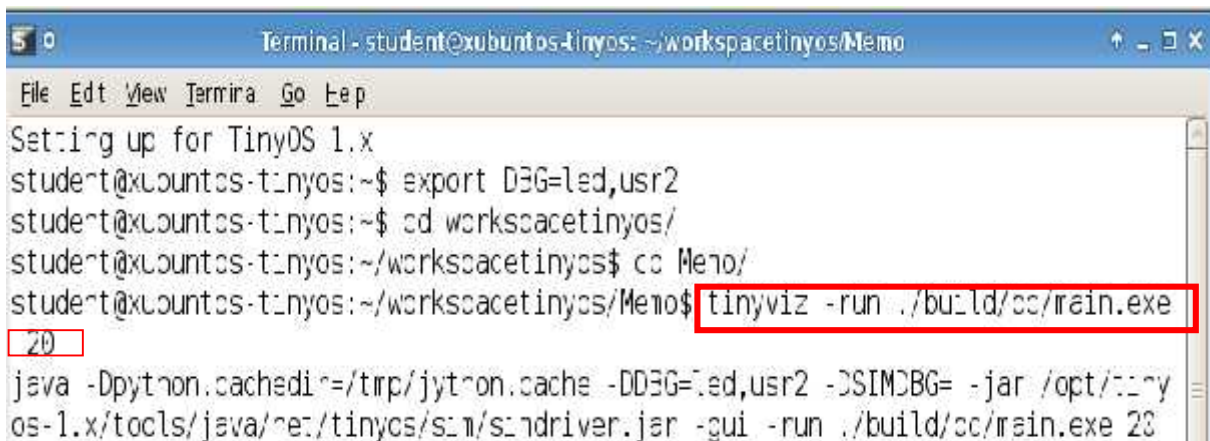


```
Terminal - student@xubuntu-os-tinyos: ~/workspacetinyos/Memo
File Edit View Terminal Go Help
Setting up for TinyOS 1.x
student@xubuntu-os-tinyos:~$ cd workspacetinyos/
student@xubuntu-os-tinyos:~/workspacetinyos$ ls
Exemple Fatma Memo Presence Routage Sence Sense Temperature wasila
student@xubuntu-os-tinyos:~/workspacetinyos$ cd Memo/
student@xubuntu-os-tinyos:~/workspacetinyos/Memo$ export DBG=led,usr2
student@xubuntu-os-tinyos:~/workspacetinyos/Memo$ ./build/pc/main.exe 20
```

Figure 25 : Commandes d'exécution

III.1. Simulation graphique

Le simulateur graphique est TinyViz et la commande d'exécution est



```
Terminal - student@xubuntu-os-tinyos: ~/workspacetinyos/Memo
File Edit View Terminal Go Help
Setting up for TinyOS 1.x
student@xubuntu-os-tinyos:~$ export DBG=led,usr2
student@xubuntu-os-tinyos:~$ cd workspacetinyos/
student@xubuntu-os-tinyos:~/workspacetinyos$ cd Memo/
student@xubuntu-os-tinyos:~/workspacetinyos/Memo$ tinyviz -run ./build/pc/main.exe
20
java -Dpython.cachedir=/tmp/jython.cache -DDBG=led,usr2 -DSIMDBG= -jar /opt/tiny
os-1.x/tools/java/net/tinyos/sim/simdriver.jar -gui -run ./build/pc/main.exe 23
```

Figure 26 : commande d'exécution TinyViz

- Réception des données par le cluster head et l'envoi à la station de base

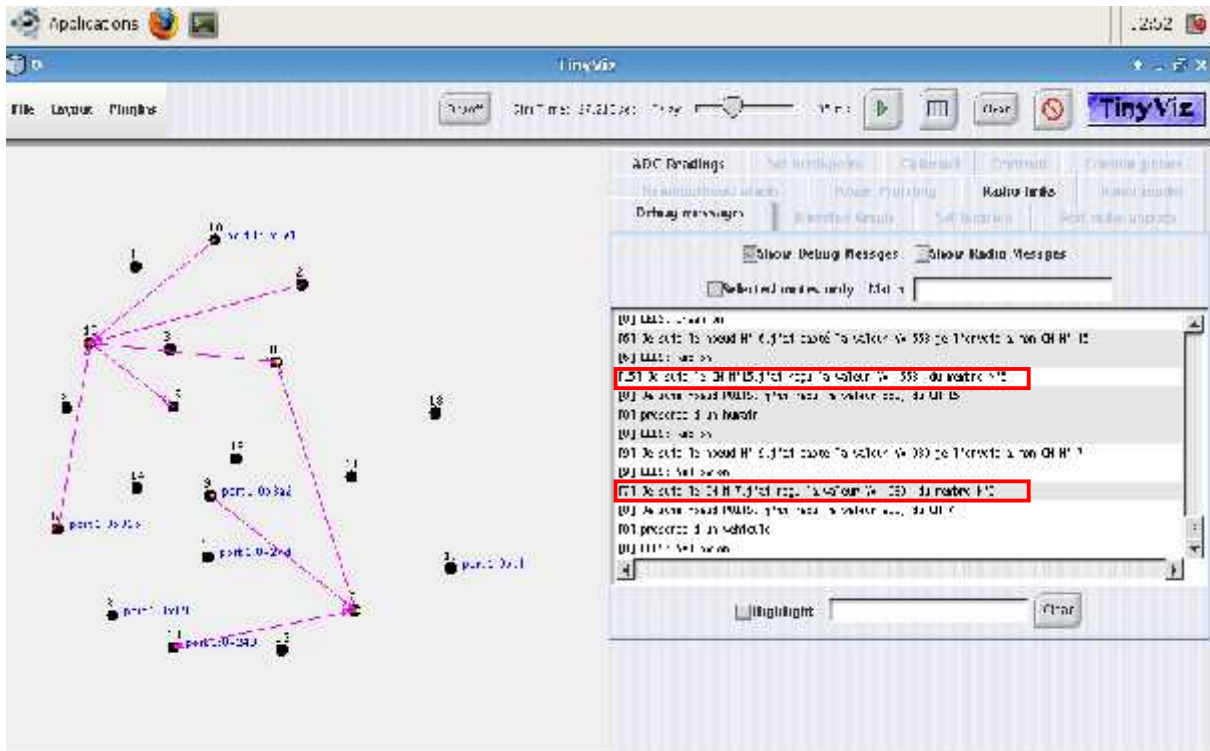


Figure 29: Réception des données par le cluster head et l'envoi à la station de base

- Réception des données par la station de base et détermination de la nature de la présence

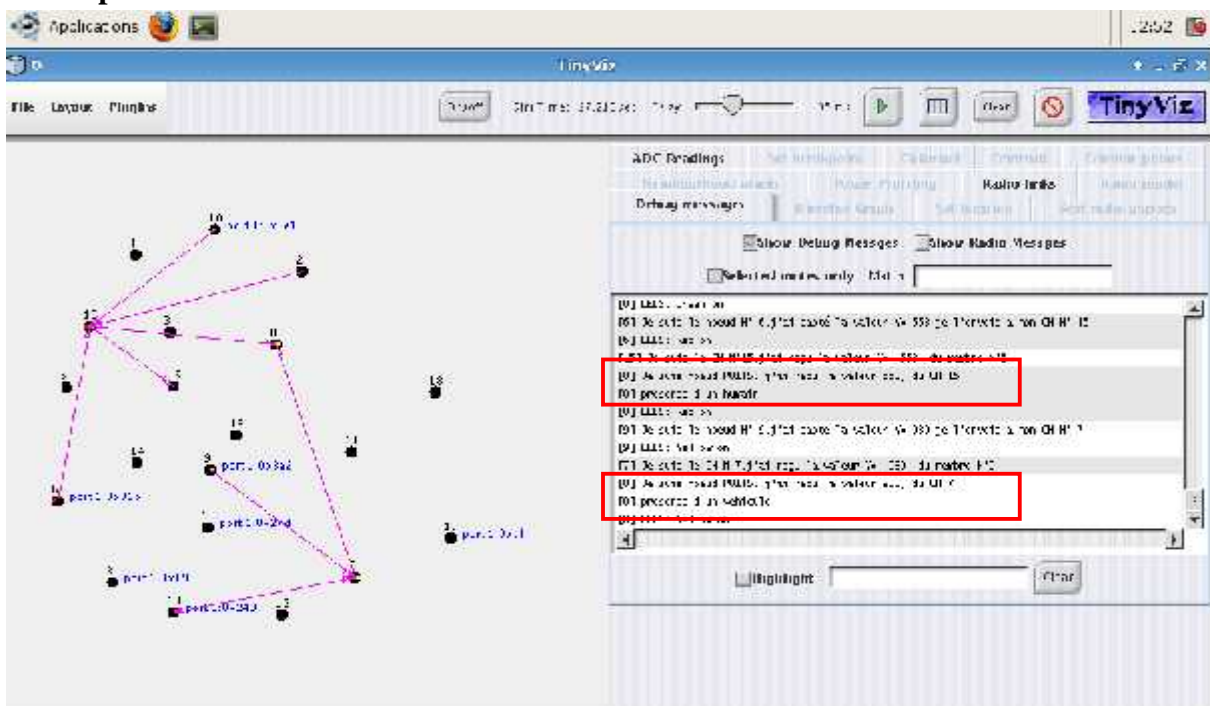
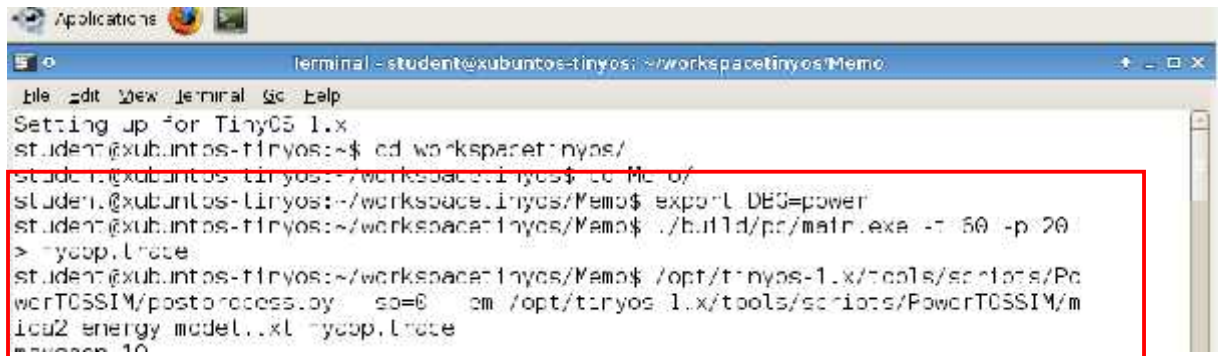


Figure 30: Réception des données par la station de base et détermination de la nature de la présence

III.2. Simulation d'énergie

a) Avec PowerTOSSIM

Les commandes de simulation d'énergie sont

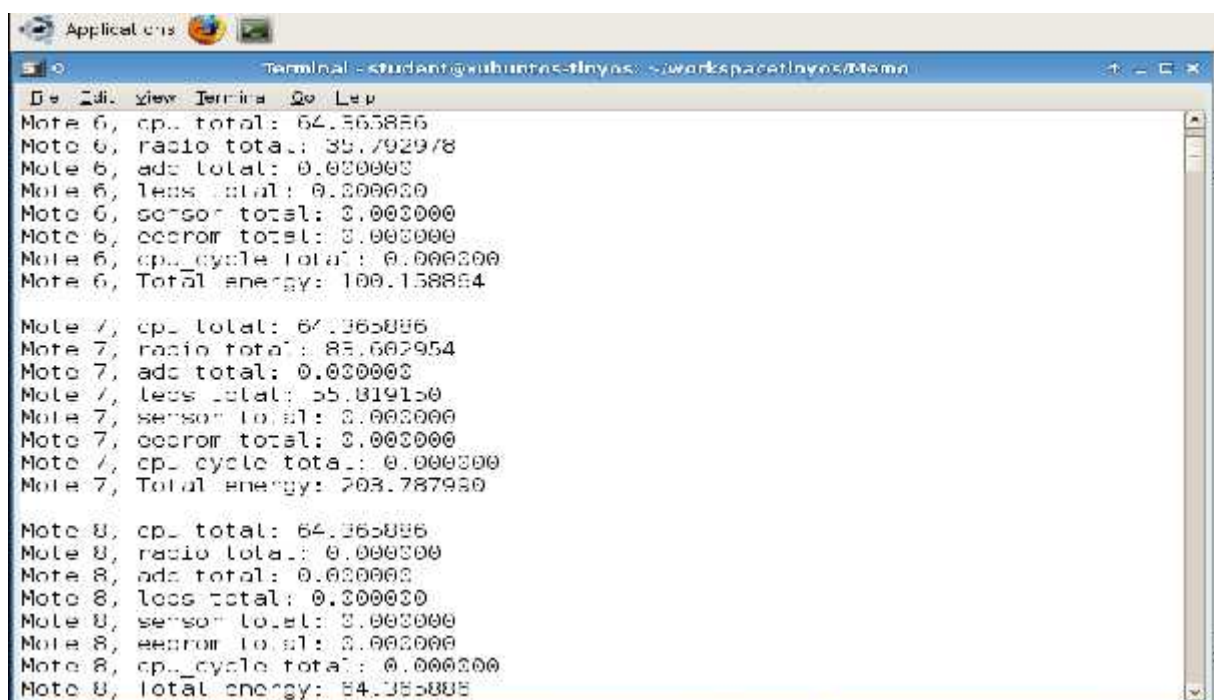


```

terminal - student@ubuntuos-tinyos: ~/workspace/tinyos/Memo
File Edit View Terminal Go Help
Setting up for TinyOS 1.x
student@ubuntuos-tinyos:~$ cd workspace/tinyos/
student@ubuntuos-tinyos:~/workspace/tinyos$ cd Memo/
student@ubuntuos-tinyos:~/workspace/tinyos/Memo$ export DEFS=power
student@ubuntuos-tinyos:~/workspace/tinyos/Memo$ ./build/pr/main.exe -t 60 -p 20
> nycpp.trace
student@ubuntuos-tinyos:~/workspace/tinyos/Memo$ /opt/tinyos-1.x/tools/scripts/PowerTOSSIM/postprocess.py so=0 em /opt/tinyos-1.x/tools/scripts/PowerTOSSIM/mica2_energy_model.xml nycpp.trace
success 10
  
```

Figure 31 : Commandes d'exécution PowerTOSSIM

La simulation de la consommation énergétique de cette application avec PowerTOSSIM pour un réseau de 20 nœuds pendant 60 secondes est illustrée à la **figure 30**. Cette simulation indique la consommation de chaque périphérique au niveau de chaque nœud du réseau en mJ. A titre d'exemple, le nœud 7 a consommé en totalité 203.787990mJ, dont 64.365886mj consommés par l'unité centrale de traitement, 83.602954mj consommés par l'unité de transmission radio et 55.819150mj consommés par les leds .



```

Terminal - student@ubuntuos-tinyos: ~/workspace/tinyos/Memo
File Edit View Terminal Go Help
Node 6, cpu total: 64.365886
Node 6, radio total: 83.602954
Node 6, adc total: 0.000000
Node 6, leds total: 55.819150
Node 6, sensor total: 3.000000
Node 6, eeprom total: 3.000000
Node 6, cpu_cycle total: 0.000000
Node 6, Total energy: 109.138884

Node 7, cpu total: 64.365886
Node 7, radio total: 83.602954
Node 7, adc total: 0.000000
Node 7, leds total: 55.819150
Node 7, sensor total: 3.000000
Node 7, eeprom total: 3.000000
Node 7, cpu_cycle total: 0.000000
Node 7, Total energy: 203.787990

Node 8, cpu total: 64.365886
Node 8, radio total: 0.000000
Node 8, adc total: 0.000000
Node 8, leds total: 0.000000
Node 8, sensor total: 3.000000
Node 8, eeprom total: 3.000000
Node 8, cpu_cycle total: 0.000000
Node 8, Total energy: 64.365886
  
```

Figure 32 : Simulation de la consommation de l'application avec PowerTOSSIM

Le graphe suivant représente la consommation moyenne en énergie des nœuds. Elle varie d'un nœud à un autre selon la fonction du nœud (Sink, Cluster Head, Nœud ordinaire).

Ces résultats sont obtenus après 10 simulations successives de l'application pendant un temps de 60 secondes pour chaque simulation.

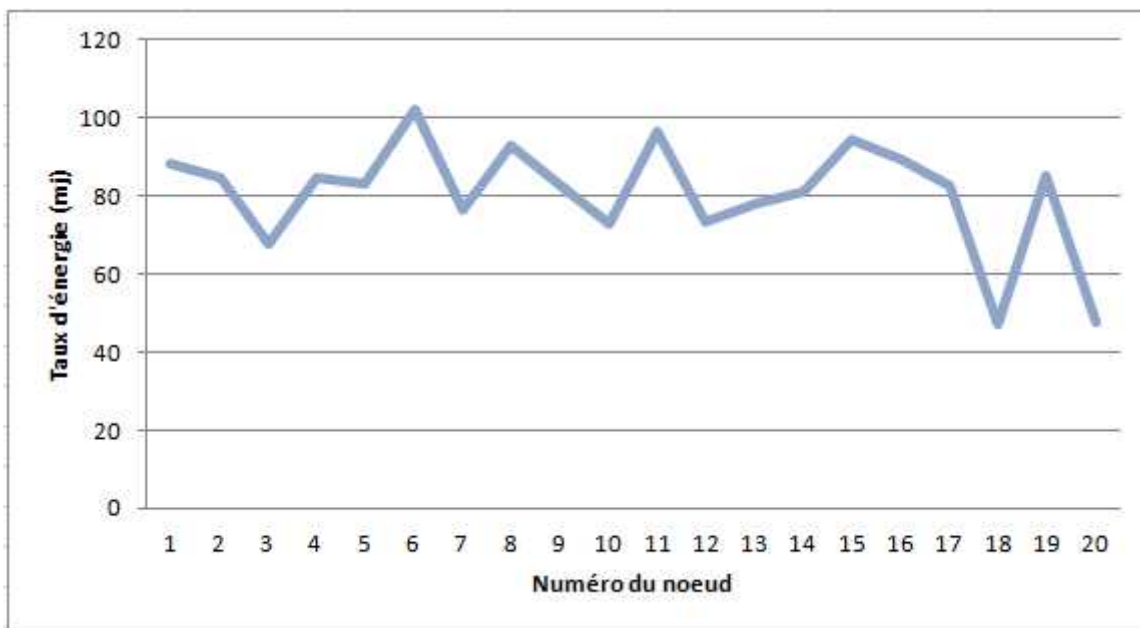


Figure 33 : représentation de la consommation moyenne en énergie des nœuds

b) Avec TinyViz

Ouvrir un terminal et exécuter les commandes suivantes

```
Terminal - student@xubuntu-tinyos: /workspacetinyos/Memo
File Edit View Terminal Go Help
Setting up for TinyOS 1.x
student@xubuntu-tinyos:~$ cd workspacetinyos/
student@xubuntu-tinyos:~/workspacetinyos$ cd Memo/
student@xubuntu-tinyos:~/workspacetinyos/Memo$ export DBG=power
student@xubuntu-tinyos:~/workspacetinyos/Memo$ ./build/pc/main.exe -gui -p 20
SIM: Random seed is 739755
```

Et ouvrir un autre terminal et exécuter la commande suivante

```
Terminal - student@xubuntu-tinyos: ~
File Edit View Terminal Go Help
Setting up for TinyOS 1.x
student@xubuntu-tinyos:~$ tinyviz
java -Dpython.cachedir=/tmp/jython.cache -DDBG= -DSIMDBG= -jar /opt/tinyos-1.x/t
```

Figure 34 : commandes de simulation de la consommation d'énergie avec TinyViz

Le plugin **Power Profiling** de TinyViz permet de voir l'état de la consommation de l'énergie de l'application en cours d'exécution.

Dans notre cas, les résultats sont représentés dans la figure suivante

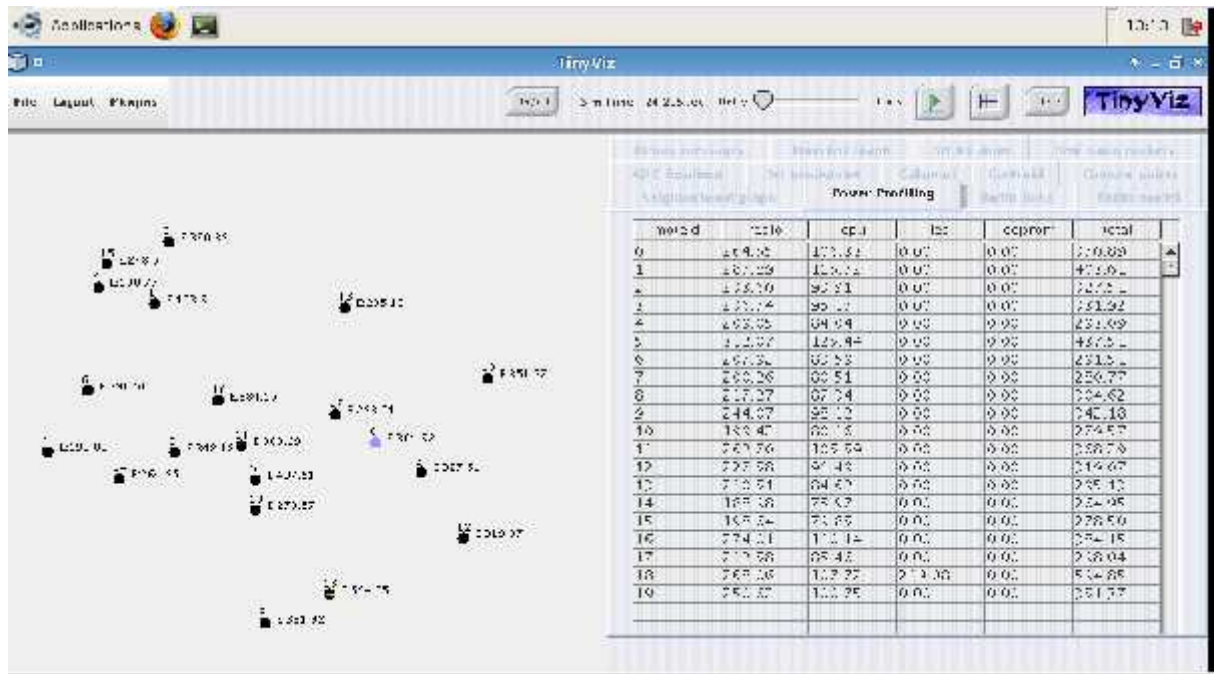


Figure 35: Représentation de la consommation d'énergie avec TinyViz

Conclusion

Dans ce chapitre, nous avons présenté l'environnement de simulation ainsi que le déroulement de notre application appuyé par quelques figures.

Conclusion générale

Les réseaux de capteurs sont une nouvelle technologie qui a surgit après les grands progrès technologique concernant le développement des capteurs intelligents, des processeurs puissants et des protocoles de communications sans fil. Ils ont été classés parmi les 21 technologies les plus importantes du 21^{ème} siècle. En effet, la recherche dans le domaine des capteurs est en train de vivre une révolution importante, ouvrant des perspectives d'impacts significatifs dans de nombreux domaines tel que le domaine de la surveillance comme c'est le cas de notre travail, pour que ces réseaux puissent mener à bien leurs missions ils doivent assurer un certain niveau de contrôle qui diffèrent selon l'application déployée.

Au cours de notre mémoire, nous nous sommes intéressées à simulé un détecteur de présence, Il a fallut pour cela prendre en main beaucoup de nouvelles technologies, comme TinyOS et le NesC.

Nous avons débuté par une étude approfondie sur les RCSF, nous avons donné un aperçu sur les RCSF et certaines de leurs applications. En suite, on a exposé quelques applications de détection de présence suivi de la présentation de protocole utilisé dans l'application et ainsi que le système d'exploitation utilisé et le langage de programmation NesC et les outils de simulation (TOSSIM, PowerTOSSIM et Tinyviz).

D'un point de vue personnel, ce projet nous a apporté des bénéfices personnels, il a permis de découvrir un nouveau domaine, une nouvelle manière de programmer et de concevoir une application, avec des contraintes techniques et matérielles très importantes, la découverte de TinyOS et le NesC. Ce projet été particulièrement intéressant par le fait que les réseaux de capteurs sans fil sont vraiment en plein expansion de nos jours, mais encore trop peu connu des personnes extérieures à ce domaine, la recherche sur les réseaux de capteurs est actuellement en pleine essor. Il a été pour nous un grand plaisir de s'intégrer à la recherche dans ce domaine. Travailler sur une activité possédant des enjeux aussi enthousiasmants nous a motivés pour mener à terme ce projet.

On a atteint le but principal de notre application qui est la surveillance des alentours d'une usine pour détecter la présence d'un animal, humain ou véhicule.

Les résultats qu'on a obtenu ne sont pas une fin en soi, au contraire ils nous mènent à poser des questions sur d'autres perspectives (faire une expérimentation réelle afin de valider nos

résultats, utilisation d'un grand nombre de capteurs), d'affiner cette recherche et utiliser cette application pour tirer partie de toutes ses ressources et arriver à de meilleurs résultats.

Cependant on n'a donné qu'un mince aperçu sur les possibilités énormes des réseaux de capteurs, il est donc nécessaire de saisir chaque opportunité et travailler davantage pour pouvoir utiliser toutes ses fonctions.

Bibliographies

[AAC 2014] Activity-aware clustering algorithm for wireless sensor networks
ieeexplore.ieee.org, juillet 2014.

[ABC Web] www.ab.com/catalogs

[AKY 2002] I. AKYILDIZ, W. SU, Y. SANKARASUBRAMANIAM, E.CAYIRCI, « A Survey on Sensor Networks » IEEE Communication Magazine, 102-114,2002.

[BEN 2012] BENAMEUR Amina, « Mise en place d'un réseau de capteur sans fil pour l'irrigation intelligente, université Abou Bakr Belkaid Tlemcen 2012.

[BER 2009]BERRACHEDI AMEL, DIARBAKIRLI AMINA, « Sécurisation du protocole de routage hiérarchique LEACH dans les réseaux de capteurs sans fil », 2009.

[BOU 2007] Bouabdellah Kechar, « Problématique de la consommation de l'énergie dans les réseaux de capteurs sans fil », Séminaire LIUPPA, Université de Pau et des Pays de l'Adour, 14 Octobre 2007.

[BOU 2010] BOUCIF Amar Bensaber, « Introduction à la sécurité des réseaux de capteurs sans fil », Université Bejaia, 2010.

[CHA 2008] Yacine CHALLAL, « Les réseaux de capteurs sans fil », Cours université Clermont, 2009.

[CHA 2009] Gérard CHALHOUB, « Réseaux de capteurs sans fil » Cours, université Clermont, 2009.

[CHO 2003] Chee-Yee CHONG, Srikanta P.KUMAR,« Sensor Networks, Evolution, Opportunities, and Challenges », proceeding of the IEEE, VOL 91, NO 8 Août 2003.

[CHR Web] CHRISTIAN BISSIERES <http://cbissprof.free.fr>

[CRE Web] <http://creativecommons.org/licenses/by-nc/2.0/fr/>.

[DAV 2003] David Gay, Philip Levis, David Culler, Eric Brewer, « NesC1.1 Language Reference Manual», May 2003.

[DET Web] <http://www.th-industrie.com/detecteur-gaz-co-co2-cov-/190-qualite-d-air-co-co2.html>

[DJA 2008] Djallel Eddine Boubiche, « Protocole de routage pour les réseaux de capteurs sans fil », Mémoire de magistère, Université de l'Hadj Lakhdar, Batna, Algérie, 2008.

[DUA Web]: <http://www.niko.eu/frbe/products/350-20065#sthash.5sUhLcNM.dpuf>

[ELB 2011] Elboughdiri Abdessalem, « Conception et modélisation d'un réseau de capteur sans fil », mémoire de master, option : Technique de communication radiofréquence, juin 2011.

[ELM] El Mehdi Damou, « Simulation d'un réseau de capteur avec TinyOS », France.

[FAR 2008] Fares Abdelfatah, « Développement d'une bibliothèque de capteurs », projet de master en informatique, 25/04/2008.

[FLE 2004] E. Fleury « réseaux de capteurs », 16 décembre 2004.

[GIR Web] www.gira.de

[HIL 2000] J. Hill, R. Szewczyk, A. Woo, S. Hollar, D. E. Culler, and K. S. J. Pister. System Architecture Directions for Networked Sensors. In Architectural Support for Programming Languages and Operating Systems, pages 93–104, 2000. TinyOS is available at <http://webs.cs.berkeley.edu>.

[INF Web] <http://www.normelec.ch/fr/produits/apercu-des-produits/detecteur-de-mouvement/detecteur-de-presence/gamme-busch-guard-presence-tech-basicline-mini/>

[ISA 2007] Isabelle Guerin Lassous, « Réseaux Ad Hoc, réseaux de capteurs », Cours M2 Recherche RTS, RTS5, Page(s) : 05-10, Université de Lyon, 26 Septembre 2007.

[JIM 2014] Jim Kardach, « Tech History: How Bluetooth got its name », EETimes, 3 April 2008, page consultée le 18 octobre 2014.

[KRA 2012] YACINE KRAMIA, « Mise en œuvre d'un réseau de senseurs sans fil pour la détection d'un incendie », Mémoire de doctorat, Université de Laval Quebec , 2012.

[LAB 2006] Houda LABIOD, « Réseaux mobiles ad hoc et réseaux de capteurs sans fil », Paris 2006.

[LYE 2004] Lyes Khelladi, Nadjib Badache « Les réseaux de capteurs: état de l'art », Rapport de recherche, Algérie, Février 2004.

[MAT] Matt Welsh, Robert von Behren, « The NesC Language: A Holistic to Networked Embedded Systems », EECS Department University of California, Berkeley, CA 94720.

[MEH 2007] Mehdi Damou, « simulation d'un réseau de capteur avec TinyOS », projet de fin d'étude, 2007.

[NIK Web] www.niko.fr

[RAO 2010] Raoul Herzog, « Cours capteur », Septembre 2010.

[SAC 2004] Sachin Mujumdar, « Prioritized Geographical Routing In Sensor Networks », Thèse, Université Vanderbilt, Mai 2004.

[SAH 2013] SAHRAOUI Belkheyr, « Etude d'un protocole de routage basé sur les colonies de Fourmis dans les réseaux de capteurs sans fil », Mémoire de master, Université ABOU BAKR BELKAID, Tlemcen, 2013.

[SAM 2006] Samra Boulfekhar, «Approches de minimisation d'énergie dans les réseaux de capteurs», Mémoire de Magistère, Université Abderahmane Mira de Bejaïa, 2006.

[SCH 2003] Le magazine Schneider Electric de l'enseignement technologique et professionnel « l'essentiel de la détection », novembre 2003.

[SOR Web] <http://www.sorhea.fr/technologie-camera-thermique.html>

[SRA 2003] Srajan Raghuwanshi, «An Energy Efficient Cross Layer Design Scheme for Wireless Sensor Networks», Master's Thesis, Virginia Polytechnic Institute and State University, 29 Août 2003.

[SWI Web] www.swisslux.ch

[ULT Web] <http://www.generationrobots.com/fr/content/65-les-capteurs-a-ultrasons-pour-les-robots>

[WAS 2006] Wassim Znaidi «modélisation formelle de réseaux de capteurs à partir de TinyOS » rapport de projet de fin d'étude option signaux et systèmes, 10 juin 2010.

[WEN 2000] Wendi Beth Heinzelman, «Application-Specific Protocol Architectures for Wireless Network », IEEE Transactions on Wireless Communications, Massachusetts Institute of Technology, June 2000.

[WIK Web] www.wikipedia.org.

[YAS 2007] Yasser Romdhane, « Evaluation des performances des protocoles S-MAC et Directed Diffusion dans les réseaux de capteurs », Projet de fin d'études, Ecole Supérieure des Communications de Tunis (Sup'Com), 2006 / 2007.

[ZIG] «ZigBee Wireless Sensor Networks and Their applications», Meng-Shiuan Pan and Yu Chee Tseng, Department of Computer Science, National Chiao Tung University, Hsin-Chu, 30010, Taiwan, E-mail: mspan@csie.nctu.edu.tw and yctseng@cs.nctu.edu.tw.