

Table des matières

Liste des figures.....	6
Liste des tableaux.....	7
Introduction	8
Chapitre 1 Réseaux de capteurs.....	10
1.1 Introduction	11
1.2 Réseaux sans fil.....	11
1.2.1 Classification selon la couverture géographique.....	12
1.2.2 Classification selon l'infrastructure mise en place	14
1.3 Architecture des réseaux de capteurs	16
1.3.1 Architecture d'un capteur.....	16
1.3.2 Architecture d'un réseau de capteurs	18
1.4 Contraintes liées aux réseaux de capteurs.....	19
1.5 Communication dans les réseaux de capteurs.....	20
1.5.1 Types de communications dans les réseaux de capteurs	20
1.5.2 Pile protocolaire dans les réseaux de capteurs	21
1.6 Réseaux de capteurs vs les réseaux MANETs	25
1.7 Domaines d'application des réseaux de capteurs	27
1.7.1 Applications militaires	27
1.7.2 Habitat	27
1.7.3 Applications environnementales	27
1.7.4 Applications médicales	28
1.7.5 Applications industrielles	28
1.8 Quelques axes de travaux pour les réseaux de capteurs.....	28
1.8.1 Systèmes d'exploitation	28
1.8.2 Agrégation des données.....	29
1.8.3 Clustering.....	30
1.8.4 Localisation des nœuds.....	31
1.8.5 Synchronisation.....	32
1.8.6 Sécurité	32
1.8.7 Routage.....	32
1.9 Conclusion	33
Chapitre 2 Routage dans les réseaux de capteurs	34
2.1 Objectif.....	35
2.2 Éléments de conception d'un protocole de routage.....	35
2.3 Métriques de mesure d'efficacité des protocoles de routage.....	39
2.4 Classification des protocoles de routage dans les réseaux de capteurs	41
2.5 Routage basé sur la construction des chemins.....	42
2.5.1 Routage à état de liens.....	43
2.5.2 Routage à vecteur de distance.....	43

2.5.3	Approches réactives.....	43
2.5.4	Approches proactives.....	44
2.5.5	Approches hybrides.....	44
2.6	Routage basé sur la structure du réseau.....	45
2.6.1	Routage horizontal ou à plat.....	45
2.6.2	Routage hiérarchique.....	46
2.6.3	Routage basé sur la localisation.....	47
2.7	Routage selon l'initiateur de la communication.....	47
2.7.1	Source initiateur.....	47
2.7.2	Destination initiateur.....	47
2.8	Routage basé sur le fonctionnement du protocole.....	48
2.8.1	Routage multi-chemins.....	48
2.8.2	Routage basé sur les requêtes.....	48
2.8.3	Routage basé sur la négociation.....	49
2.8.4	Routage basé sur la qualité de service.....	49
2.9	Conclusion.....	49
Chapitre 3 Les principaux protocoles de routage pour les réseaux de capteurs.....		50
3.1	Introduction.....	51
3.2	Les protocoles de routage dédiés aux WSNs.....	51
3.2.1	DD (Directed Diffusion).....	51
3.2.2	RR (Rumor Routing).....	54
3.2.3	F&G (Flooding and Gossiping).....	55
3.2.4	MCFA (Minimum Cost Forwarding Algorithm).....	56
3.2.5	GBR (Gradient Based Routing).....	57
3.2.6	EAR (Energy Aware Routing).....	58
3.2.7	Chang.....	59
3.2.8	MECN (Minimum Energy Communication Network).....	60
3.2.9	LEACH (Low Energy Adaptive Clustering Hierarchy).....	61
3.2.10	TEEN (Threshold Sensitive Energy Efficient Sensor Network Protocol).....	62
3.2.11	Younis.....	63
3.2.12	GAF (Geographic Adaptive Fidelity).....	64
3.2.13	HEAR (Hierarchical Energy Aware Routing).....	65
3.2.14	GEAR (Geographic and Energy Aware Routing).....	65
3.2.15	SPIN (Sensor Protocols for Information via Negotiation).....	67
3.2.16	SAR (Sequential Assignment Routing).....	68
3.2.17	SPEED.....	68
3.3	Classification des principaux protocoles de routage pour WSNs.....	69
3.4	Conclusion.....	70
Chapitre 4 Elaboration du protocole.....		72
4.1	Introduction.....	73
4.2	Choix d'élaboration.....	73
4.2.1	Scalabilité.....	74
4.2.2	Efficacité énergétique.....	74
4.2.3	Simplicité.....	75

4.3	Fonctionnement du protocole	75
4.3.1	Phase d'installation.....	76
4.3.2	Phase de transmission des données	79
4.3.3	Phase d'estimation des niveaux d'énergie des voisins	82
4.3.4	Phase de maintenance	82
4.4	Algorithmes du protocole.....	83
4.4.1	Algorithme 1 (Lors de la réception d'un paquet d'installation).....	83
4.4.2	Algorithme 2 (Lors de la réception d'un paquet de données).....	84
4.4.3	Algorithme 3 (Lors de la réception d'un paquet de contrôle)	84
4.5	Conclusion	85
Chapitre 5	Tests et résultats	86
5.1	Objectifs	87
5.2	Simulateur OMNeT++	87
5.2.1	Architecture du simulateur OMNeT++.....	87
5.2.2	Langage NED.....	88
5.2.3	Messages dans OMNeT++.....	91
5.2.4	Construction d'un programme de simulation OMNeT++.....	92
5.3	Tests et résultats	92
5.3.1	Performances mesurées.....	92
5.3.2	Construction de la topologie	93
5.3.3	Paramètres de simulation	94
5.3.4	Discussion des résultats	95
5.4	Conclusion	100
	Conclusion générale et perspectives	101
	Références.....	103

Liste des figures

Fig.1.1 Réseaux en mode infrastructure [TAN 03].....	14
Fig.1.2 Réseaux sans infrastructure ou ad hoc [TAN 03].....	15
Fig.1.3 Composants d'un capteur [AKY 02]	16
Fig.1.4 Architecture d'un WSN [AKY 02].....	19
Fig.1.5 Pile protocolaire dans les WSNs [AKY 02]	22
Fig.1.6 Consommation d'énergie dans les WSNs [POM 04].....	25
Fig.1.7 Configuration des cluster-heads	30
Fig.2.1 Familles des protocoles de routage dans les WSNs [ALK 05, JIA 04].....	42
Fig.2.2 Routage horizontal [DHO 03].....	45
Fig.2.3 Routage hiérarchique [DHO 03].....	46
Fig.3.1 Phases de la diffusion dirigée [INT 00]	53
Fig.3.2 Fonctionnement de Rumor Routing [BRA 01]	55
Fig.3.3 Chemin de coût minimum dans MCFA [YE 01]	57
Fig.3.4 Les états des capteurs dans un cluster [YOU 02].....	64
Fig.3.5 Protocole SPIN [WEI 99].....	67
Fig.3.6 Protocole SPEED [HE 03]	69
Fig.4.1 Découverte de voisins.....	76
Fig.4.2 Les états possibles d'un voisin.....	79
Fig.5.1 Architecture du simulateur OMNeT++ [OM web]	88
Fig.5.2 Une topologie du réseau avec 50 noeuds.....	94
Fig.5.3 Evolution du taux de défaillance des noeuds	96
Fig.5.4 Taux de succès ou de livraison des paquets	97
Fig.5.5 Moyenne d'énergie consommée.....	98
Fig.5.6 Délai moyen	99
Fig.5.7 Fonction énergétique.....	100

Liste des tableaux

Tab. 1.1 Réseaux de capteurs vs les MANETs [ILY 05]	26
Tab. 3.1 Classification des principaux protocoles de routage pour les WSNs	70
Tab. 4.1 Format du paquet d'installation.....	77
Tab. 4.2 Format de la table de voisins	78
Tab. 4.3 Format d'un paquet de données	81
Tab. 4.4 Format d'un paquet de contrôle.....	81
Tab. 5.1 Portée du signal radio selon le nombre de noeuds.....	95

Introduction

Les récents progrès dans les communications sans fil et le domaine de l'électronique ont permis le développement de petits capteurs, moins coûteux et multifonctionnels. Un réseau de capteurs est composé d'un grand nombre de nœuds capteurs déployés dans une région géographique et qui communiquent entre eux à travers un médium sans fil. L'objectif est de collecter des informations à partir de l'environnement ou d'un objet cible afin de les faire parvenir à une station de base qui effectue des calculs dessus. Les nœuds sont restreints en termes de ressources énergétiques, c'est pourquoi ils ne peuvent émettre directement à la station de base. Les messages transmis doivent donc être routés à travers le réseau. Il est par conséquent indispensable de recourir à des protocoles de routage qui conservent l'énergie pendant la communication afin d'augmenter la durée de vie du réseau.

La gestion de la consommation d'énergie dans un réseau de capteurs est un problème complexe qui nécessite une solution non seulement individuelle, comme l'utilisation de composants à faible consommation, mais également globale, par la mise en œuvre d'algorithmes de communication et de mesure maximisant la durée de vie du réseau. Cette durée de vie peut être accrue par la prise en compte de la contrainte énergétique à chaque étape de conception et de la mise en œuvre du réseau.

La recherche de protocoles efficaces en termes de consommation d'énergie s'est intensifiée durant les dernières années. Bien que l'optimisation de la consommation ne doive pas se restreindre à un aspect du comportement du réseau mais bien à l'ensemble de sa mise en œuvre. Notre travail s'intéresse particulièrement aux opérations de communication (couche réseau). En effet, l'émission et la réception d'un paquet par une interface radio est un processus coûteux. Dans ce contexte, nous nous intéressons à la conception d'un nouveau protocole de routage avec conservation d'énergie pour les réseaux de capteurs qui a pour objectif de prolonger la durée de vie du réseau en insistant sur la manière de gérer la consommation d'énergie au niveau de chaque nœud.

Ce document est organisé comme suit :

Dans le chapitre 1, nous apportons, dans un premier temps, une description des réseaux sans fil. Dans un second temps, seront données les définitions généralement acceptées pour les réseaux de capteurs, leurs principales caractéristiques et contraintes, ainsi que leurs domaines d'application.

Nous présentons ensuite, dans le chapitre 2, une vue globale sur les manières de classer les protocoles de routage pour les réseaux de capteurs et qui ont un objectif commun à savoir augmenter leur durée de vie. Les protocoles de routage dans les réseaux de capteurs peuvent être classifiés en suivant l'une des quatre manières : selon la structure de réseau, selon les fonctions des protocoles, selon l'établissement de route, et enfin selon l'initiateur de la communication. Dans le chapitre 3, nous présentons un état de l'art sur les principaux protocoles de routage utilisés dans le cas des réseaux de capteurs.

Au chapitre 4, nous proposons une approche dans la problématique de la conservation de l'énergie dans les réseaux de capteurs, en décrivant les détails conceptuels de notre approche. Le chapitre 5 présente, dans un premier temps un scénario de simulation ainsi que les critères de performances sur lesquels nous nous sommes basés pour tester notre modèle. Dans le second temps, les résultats de notre simulation sont comparés à ceux du protocole EAR (*Energy Aware Routing*).

Enfin, notre mémoire s'achève par une conclusion générale résumant les grands points qui ont été abordés ainsi que des perspectives que nous souhaitons accomplir prochainement.

Chapitre 1

Réseaux de capteurs

1.1 Introduction

Surveiller les différents phénomènes physiques (Par exemples la température, le niveau du fluide, l'humidité, l'acidité, les vibrations, etc.) est une nécessité pour beaucoup de secteurs comprenant une structure technologique tels que l'agriculture, la santé, la logistique, le transport, l'industrie ou les applications militaires. Des réseaux de capteurs filaires ont été longtemps déployés afin d'observer ces phénomènes.

Récemment, grâce aux progrès dans les domaines de l'électronique digitale et de la technologie des communications sans fil, des capteurs multifonctionnels à basse puissance avec un coût très bas ont été développés. Ces derniers ont une taille très petite et peuvent communiquer entre eux sur de courtes distances, d'où l'apparition des réseaux de capteurs sans fil WSNs (*Wireless Sensor Networks*) utilisés essentiellement quand une infrastructure de câble est difficile à mettre en place, comme dans des endroits hostiles ou inaccessibles.

Dans ce chapitre, sont introduites des connaissances de base nécessaires à la bonne compréhension de ce document. Nous commençons d'abord par donner certaines notions générales concernant les réseaux sans fil. Ensuite des définitions généralement acceptées pour les réseaux de capteurs seront données.

1.2 Réseaux sans fil

Les réseaux sans fil (*Wireless Networks*) constituent de plus en plus une technologie émergente permettant à ses utilisateurs un accès à l'information et aux services électroniques indépendamment de leur position géographique.

Récemment, les réseaux sans fil caractérisés par leur topologie dynamique, facilité et rapidité de déploiement, ont suscité un intérêt particulier, notamment dans des situations spécifiques où les connexions ne peuvent pas être assurées comme dans certaines zones géographiques ou dans des sites classés monuments historiques et pour lesquels il est impossible d'installer du matériel. Les réseaux sans fil offrent donc une réelle alternative.

Dans les réseaux sans fil, il est possible de distinguer deux modes de classification : dans la première, la classification se base sur la zone de couverture du réseau; et dans la seconde, elle dépend de l'infrastructure mise en place.

1.2.1 Classification selon la couverture géographique

On peut répartir les réseaux sans fil en quatre catégories principales, d'après leur usage :

Les réseaux personnels sans fil (WPAN)

Les réseaux personnels sans fil ont une faible couverture géographique. Ils sont le plus souvent utilisés dans des petits espaces personnels et servent généralement à l'interconnexion de dispositifs électroniques se trouvant à proximité les uns des autres (clavier - ordinateur, souris - ordinateur, téléphone cellulaire – téléphone cellulaire ...). Ces réseaux sont à performance modérée, avec un débit allant jusqu'à 2 Mbps. La plupart des réseaux personnels sans fil utilisent les ondes électromagnétiques comme support de transmission. Par exemple, la norme *Bluetooth* définit le traitement d'un WPAN en fonctionnement, dans la bande de fréquence 2,4-GHz, avec une couverture de plus de 15 mètres et un débit allant jusqu'à 2 Mbps. D'autres réseaux personnels utilisent la lumière infrarouge comme médium. La norme *IrDA (Infrared Data Association)* définit l'utilisation de rayons infrarouges directs afin d'obtenir une couverture allant jusqu'à 1 mètre, et un débit pouvant atteindre les 4 Mbps. L'avantage de l'utilisation de l'infrarouge est l'absence d'interférences radios [GEI 04].

Les réseaux locaux sans fil (WLAN)

La deuxième catégorie des réseaux sans fil concerne les LAN sans fil. Il s'agit de systèmes dans lesquels chaque ordinateur est équipé d'un modem radio et d'une antenne grâce auxquels il peut communiquer avec les autres machines. Un réseau local sans fil peut avoir une portée d'une centaine de mètres et un débit allant jusqu'à 54 Mbps. Ainsi il convient pour la plupart des applications d'entreprises ou dans les foyers. La norme IEEE 802.11 est la plus répandue pour les réseaux locaux sans fil ;

néanmoins, il subsiste un problème d'interopérabilité limitée entre les différentes versions de la norme IEEE 802.11 (IEEE 802.11a, IEEE 802.11b ...). Afin de pallier à ce problème, les fonctions du standard IEEE 802.11 ont été reprises par une nouvelle norme : *Wireless-Fidelity (Wi-Fi)*. La norme *Wi-Fi* permet de créer des WLAN à haut débit variant de 11 Mbps (pour la norme 802.11b) jusqu'à 54 Mbps (pour la norme 802.11a/g). Une autre norme utilisée dans les réseaux locaux sans fil est l'*HyperLan 2*. Elle offre un débit théorique de 54 Mbps, et utilise la plage de fréquences entre 5150 et 5300 MHz.

Les réseaux métropolitains sans fil (WMAN)

Un WMAN est connu sous le nom de "Boucle Locale Radio" (BLR). Une BLR permet de relier un opérateur de télécommunication à ses clients grâce aux ondes radio sur une distance allant de quatre à dix kilomètres et offrant un débit qui varie entre 1 et 10 Mbps. Les WMAN sont basés sur la norme IEEE 802.16, connue sous le nom commercial de *WiMax*. Ce type de réseaux exploite les bandes de fréquences entre 2,4Ghz et 3,5Ghz [GEI 04].

Les réseaux étendus sans fil (WWAN)

Le réseau étendu sans fil est connu aussi sous le nom de réseau cellulaire mobile. Grâce à l'essor qu'a connu la téléphonie mobile, il est devenu le réseau sans fil le plus répandu. Ce type de réseaux connaît des performances assez faibles, avec un débit atteignant un maximum de 170 Kbps, et travaillant usuellement avec un débit de 56 Kbps. Les technologies les plus utilisées pour ce genre de réseaux sont GSM, GPRS et UMTS. La norme GSM (*Global System for Mobile Communication*) est un standard de téléphonie mobile utilisé essentiellement en Europe, en Asie et en Afrique. Le GPRS (*General Packet Radio Service*) quant à lui permet l'accès à de nombreux services multimédias via le WEB ou le WAP. Finalement, l'UMTS (*Universal Mobile Telecommunications System*) est une technologie de transmission de troisième

génération ; son débit peut atteindre 2 Mbps alors qu'il ne dépasse pas 9,6 Kbps pour le GSM et 115 Kbps pour le GPRS.

1.2.2 Classification selon l'infrastructure mise en place

Dans le cadre des réseaux sans fil, le standard IEEE 802.11 définit deux modes de fonctionnement différents : le mode infrastructure, où les clients sont connectés à un point d'accès et le mode sans infrastructure ou ad hoc où les clients sont connectés les uns aux autres d'une façon spontanée [BRA 06].

Le mode avec Infrastructure

En mode avec infrastructure, le réseau sans fil consiste au minimum en un point d'accès AP (*Access Point*) connecté à l'infrastructure du réseau filaire et un ensemble de stations sans fil. L'ensemble formé par le point d'accès et les stations situés dans la zone de couverture de ce point d'accès est appelé l'ensemble de services de base BSS (*Basic Service Set*). Comme le montre la figure 1.1, chaque BSS constitue une cellule et est identifié par l'adresse MAC du point d'accès. Les points d'accès peuvent être reliés entre eux (plusieurs BSS) par une liaison appelée système de distribution DS (*Distribution System*), qui est une dorsale ou *backbone* responsable du transfert des trames entre les différents BSS afin de constituer un ensemble de services étendu ESS (*Extended Service Set*). Le système de distribution (DS) peut être aussi bien un réseau filaire, qu'un câble entre deux points d'accès ou même un réseau sans fil.

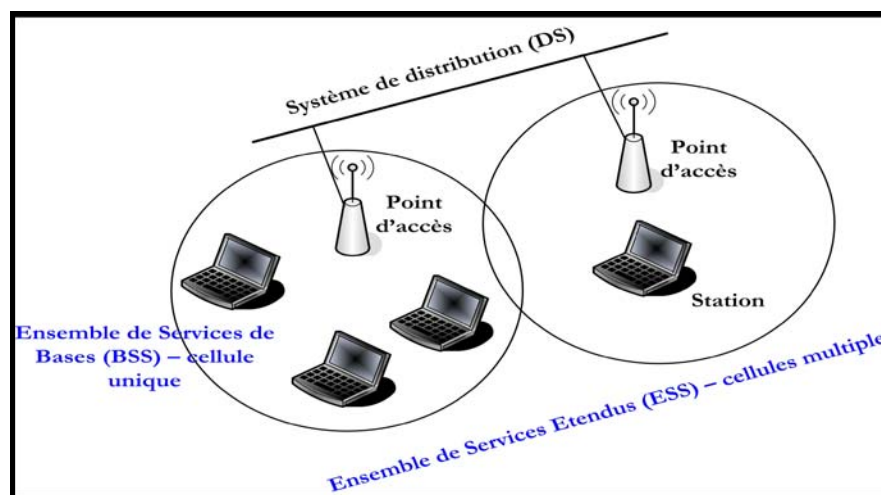


Fig. 1.1 Réseau en mode infrastructure [TAN 03].

Le mode sans infrastructure

Le mode sans infrastructure, appelé aussi réseau ad hoc ou MANET, représente simplement un ensemble de stations sans fil 802.11 qui communiquent directement entre elles sans point d'accès ni connexion à un réseau filaire. Ce mode permet de créer rapidement et simplement un réseau sans fil (dit *peer to peer* ou point à point) là où il n'existe pas d'infrastructure filaire ou encore là où une telle infrastructure n'est pas nécessaire pour les services attendus. Les différentes stations constituent l'ensemble de services de base indépendants ou IBSS (*Independent Basic Service Set*). Comme le montre la figure 1.2, un IBSS est ainsi un réseau sans fil constitué au minimum de deux stations et n'utilisant pas de points d'accès.

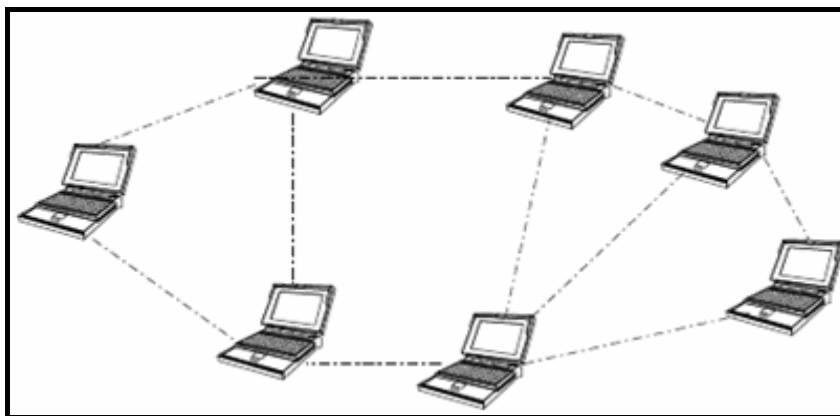


Fig. 1.2 Réseau sans infrastructure ou ad hoc [TAN 03].

La conception des protocoles de communication pour les réseaux ad hoc est principalement soumise à trois facteurs, qui peuvent être résumés comme suit :

- **Energie limitée** : les hôtes fonctionnent grâce à une batterie, dont la durée de vie est généralement limitée à quelques heures d'utilisation ; les communications doivent donc être réduites au strict minimum.
- **Autonomie de décision** : aucune autorité centrale n'est présente pour gérer les opérations des différentes hôtes; le réseau est donc entièrement décentralisé.

- **Topologie dynamique** : les hôtes sont mobiles et peuvent être connectés entre eux de manière arbitraire. Les liens radios changent régulièrement, lorsque les objets se déplacent, s'éteignent, ou lorsque des obstacles apparaissent ou disparaissent.

1.3 Architecture des réseaux de capteurs

1.3.1 Architecture d'un capteur

Un capteur est composé de quatre composants de base (figure 1.3) : une unité de capture, une unité de traitement, une unité d'émission/réception, et une unité d'énergie. Il se peut aussi qu'il existe d'autres composants additionnels dépendant de l'application, par exemples : un générateur d'énergie, un système de localisation, et un mobilisateur.

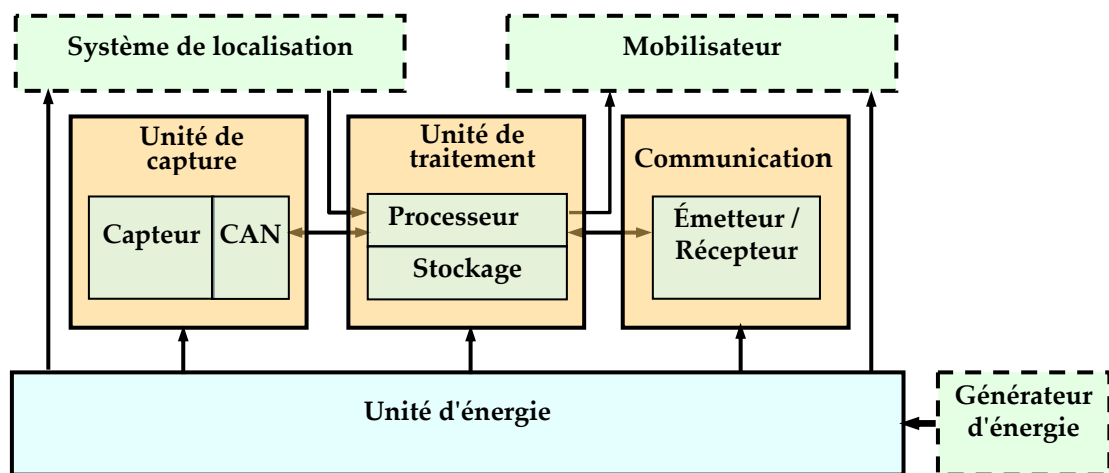


Fig. 1.3 Composants d'un capteur [AKY 02].

Unité de capture

La fonction principale de l'unité de capture est de capturer ou mesurer les données physiques à partir de l'objet ou de l'environnement cible. Le signal analogique correspondant aux événements observés par le capteur est ensuite transformé en données numériques qui peuvent être utilisées par l'unité de traitement.

Unité de traitement

L'unité de traitement joue un rôle majeur dans la collaboration entre les nœuds afin d'accomplir les tâches ¹prédéfinies. L'unité de traitement a besoin de stocker les informations pendant le traitement local et l'agrégation des données. Une mémoire flash (mémoire non volatile servant au stockage stable d'informations) est généralement utilisée vu son faible coût et capacité de stockage.

Unité d'émission \ réception

Il existe trois schémas de communication pour les réseaux de capteurs : la communication optique (Laser), l'infrarouge et la radiofréquence (*RF : Radio Frequency*). Le Laser consomme moins d'énergie que la RF et fournit une haute sécurité, mais il exige l'utilisation d'une ligne optique et il est sensible à la perturbation physique. L'infrarouge n'a pas besoin d'antennes, mais sa capacité de diffusion est limitée. La RF est la plus simple à utiliser mais elle exige l'utilisation d'antennes [KHE 04].

Plusieurs stratégies de réduction de la consommation d'énergie sont développées, comme la modulation/démodulation et le filtrage. La modulation en amplitude est plus simple par rapport à celle en fréquence, mais elle est sensible au bruit.

Unité d'énergie

La consommation d'énergie est un point très important pour les réseaux de capteurs. Les batteries utilisées sont soit rechargeables ou non. Souvent, dans les environnements sensibles, il est impossible de recharger ou changer une batterie. Donc avoir une meilleure gestion de la consommation d'énergie est primordial pour augmenter la durée de vie du réseau.

Il existe deux grandes politiques pour conserver la consommation d'énergie. Dans la première, appelé DPM (*Dynamic Power Management*), les composants inactifs sont mis en veille. Dans ce cas une analyse stochastique pour prédire les prochains événements

¹ Prédéfinies : les procédures et protocoles permettant la communication entre les nœuds.

est nécessaire. Pour la deuxième, appelée DVS (*Dynamic Voltage Scheduling*), l'énergie est fournie aux composants selon leur charge de travail.

Les nouveaux capteurs peuvent contenir des générateurs d'énergie renouvelable, par exemple: l'énergie solaire, et l'énergie mécanique (vibration, l'air etc.).

Système de localisation \ Mobilisateur

Certaines applications ont besoin de connaître l'emplacement du capteur. Pour cette raison, le capteur doit avoir un système de localisation tel qu'un GPS (*Global Positioning System*). Dans les réseaux de capteurs mobiles, les nœuds doivent se déplacer, donc un mobilisateur doit exister dans les composants du capteur [AKY 02].

1.3.2 Architecture d'un réseau de capteurs

Un WSN se compose d'un certain nombre de capteurs (des centaines voir des milliers) dispersés dans une zone géographique. Chaque capteur est capable de communiquer avec les autres sans utiliser le support filaire. Le déploiement d'un WSN exige un minimum d'installation et un faible coût d'administration. Souvent, il est intégré avec un plus grand réseau tel que l'Internet.

Après avoir transformé le signal analogique (information capturée) en un signal numérique par un nœud du réseau, les données sont transmises par relais à nœud puit dit station de base (*Sink*). Celui-ci peut être situé près de la région à observer ou un nœud plus puissant et mieux équipé du réseau de capteurs. Le nœud station de base peut éventuellement effectuer un traitement sur les données reçues avant de les envoyer à son tour via un autre réseau au nœud directeur des tâches (*Task manager node*) à partir duquel l'utilisateur accède à ces données.

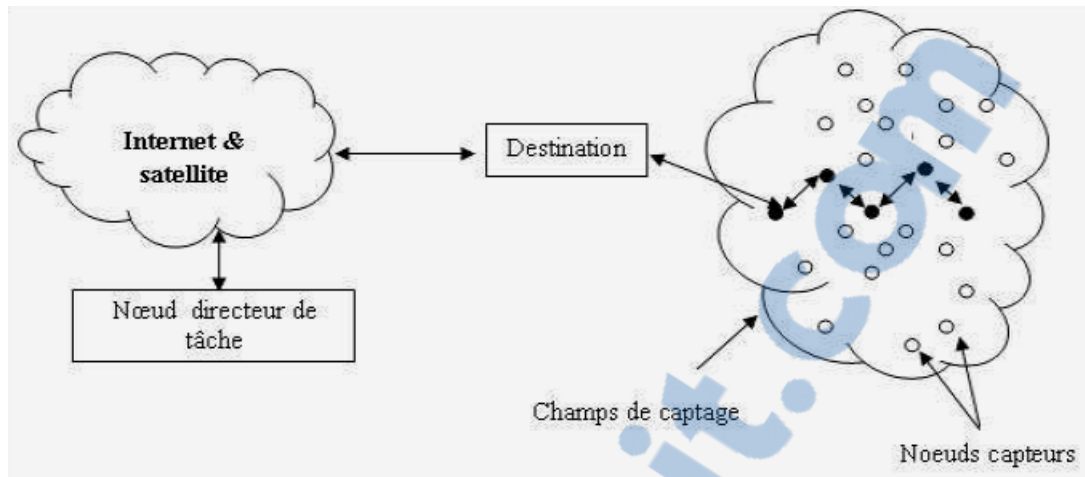


Fig. 1.4 Architecture d'un WSN [AKY 02].

1.4 Contraintes liées aux réseaux de capteurs

Un réseau de capteurs a beaucoup de contraintes selon le type de capteurs qui composent le réseau. Certaines de ces caractéristiques sont les suivantes [GUP 02, VIS 04] :

Durée de vie limitée

Les nœuds capteurs sont très limités par la contrainte d'énergie. Ils fonctionnent habituellement sans surveillance dans des régions géographiquement éloignées et par conséquent recharger ou remplacer leurs batteries devient impossible.

Ressources limitées

Habituellement, les nœuds capteurs ont une taille très réduite. Ce facteur limite la quantité de ressources qui peuvent être mises dans ces nœuds. Par conséquent la capacité de traitement et de mémoire est très limitée.

Bande passante limitée

En raison de la puissance limitée, les nœuds capteurs ne peuvent pas supporter des débits élevés.

Scalabilité

Le nombre de nœuds capteurs dans un réseau de capteurs peut être dans l'ordre des centaines ou des milliers (selon l'application).

Topologie dynamique

Comme les nœuds capteurs peuvent être déployés dans des environnements durs (par exemple un champ de bataille), la défaillance d'un nœud est très probable, indépendamment de ceci, ces nœuds de capteurs et les nœuds finaux où ils doivent envoyer l'information saisie peuvent être mobiles.

Agrégation des données

Dans les WSNs, les données produites par les nœuds capteurs sont très corrélées, ce qui implique la redondance de données, les utilisateurs sont intéressés par le phénomène qui est saisi non pas par les données générées par chaque nœud, et par conséquent, ces réseaux nous fournissent l'occasion d'agréger les données qui réduisent la largeur de la bande passante.

1.5 Communication dans les réseaux de capteurs

1.5.1 Types de communications dans les réseaux de capteurs

En général, dans les WSNs, deux types de nœuds sont identifiés logiquement : les nœuds qui principalement transmettent leurs propres données capturées (nœuds capteurs), et les nœuds qui transmettent les messages aux autres nœuds (nœuds de relais ou stations de bases). Les données capturées sont acheminées depuis les nœuds sources jusqu'aux nœuds destinataires à travers les nœuds intermédiaires, créant ainsi une topologie multi sauts. Cette organisation logique implique quatre types de communications :

1. *La communication nœud capteur à un nœud capteur* : ce type de communication directe est employé pour des opérations locales, par exemple pendant le processus de clustering² ou le processus de création de route.
2. *La communication nœud capteur à un nœud intermédiaire* : les données capturées sont transmises d'un nœud capteur à un nœud intermédiaire. Ce type de communication est souvent unicast.
3. *La communication nœud intermédiaire à un nœud capteur* : les requêtes et la signalisation des messages, sont souvent multicast, elles sont diffusées par les nœuds intermédiaires, pour atteindre un sous-ensemble des nœuds immédiatement.
4. *La communication nœud intermédiaire à un nœud intermédiaire* : la communication entre ces nœuds peut être dans la plupart du temps unicast. Noter que chaque nœud est équipé d'une radio d'émissions / réception [NIE 03].

2.5.2 Pile protocolaire dans les réseaux de capteurs

Comme dans les réseaux filaires, l'architecture de communication dans les réseaux de capteurs est divisée en couches.

Chaque couche a ses rôles et ses protocoles qui opèrent en dessus. Puisque l'objectif d'un réseau de capteurs n'est pas la communication elle même et que la consommation d'énergie est un critère très important, d'autres unités doivent exister afin de gérer la consommation d'énergie, la mobilité des nœuds et l'ordonnancement des tâches comme le montre la figure suivante [AKY 02] :

² Clustering : processus de l'organisation du réseau en cellules, chaque cellule est gérée par un nœud appelé *cluster-head*.

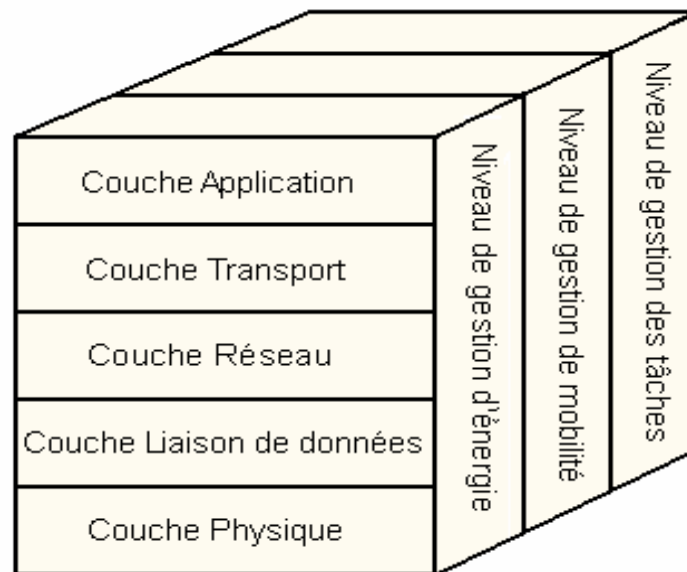


Fig. 1.5 Pile protocolaire dans les WSNs [AKY 02].

Couche physique (*Physical layer*)

La couche physique est responsable du choix de la fréquence, de la génération de la porteuse, la modulation, la détection de signal et le chiffrement de données. Deux différents médias peuvent être utilisés dans un WSN : Les infrarouges et les radiofréquences [ABO 05]. Les infrarouges sont simples et peu coûteux, ils sont capables de fournir des débits très élevés à des coûts relativement faibles et une bande passante très large. Enfin les radiofréquences, le principe est d'émettre des ondes électromagnétiques qui constituent la porteuse du signal à transmettre, le spectre radio est découpé en bandes de fréquences divisées en canaux. L'ISM (*Industrial, Scientific and Medical*) est la bande la plus suggérée pour les réseaux sans fil pour les radiofréquences, ces bandes sont gratuites mais avec des contraintes en taux d'utilisation (rapport entre le temps d'émission et le temps total) et en puissance d'émission. La génération de fréquence et la détection du signal ont plus à faire avec le matériel fondamental et la conception d'émetteur/récepteur et par conséquent sont au-delà de la portée de notre mémoire.

Couche liaison de données (*Data link layer*)

Parmi les fonctions de la couche liaison de données, on peut citer le contrôle d'accès au support de transmission MAC (*Media Access control*) et le contrôle d'erreurs.

Dans la sous-couche MAC, plusieurs techniques d'accès au support de transmission ont été proposées afin de minimiser la consommation de l'énergie. Ces techniques sont divisées en deux classes à savoir : les méthodes d'accès centralisées parmi celles-ci : FDMA (*frequency Division Multiple Access*), TDMA (*Time Division Multiple Access*) et CDMA (*Code Division Multiple Access*). Enfin les méthodes d'accès distribuées qui sont : ALOHA et CSMA (*Carrier Sence Multiple Access*) [TAN 03].

Les WSNs sont utilisés pour des applications très sensibles, donc avoir des données fiables est très important. Il existe deux catégories pour les mécanismes de contrôle des erreurs : FEC (*Forward Error Correction*) et ARQ (*Automatic Repeat reQuest*) [ILY 05]. Le contrôle des erreurs basé sur ARQ gère la perte des données par la retransmission des paquets erronés. Il est évident que cela va surcharger le réseau vu le nombre important des nœuds, donc il est déconseillé de l'utiliser pour les réseaux de capteurs à grande densité. D'autre part, les schémas FEC procèdent par le décodage des données ce qui est plus complexe et nécessite des ressources de traitements considérables. Donc la solution consiste à avoir des codes de contrôle d'erreurs simples avec moins de complexité de codage/décodage.

Couche réseau (*Network layer*)

Les capteurs sont souvent déployés avec grande densité dans la surface observée. Ainsi ils peuvent être très proches entre eux. Dans une telle situation une communication multi-sauts est la plus adéquate. De plus les nœuds doivent consommer moins d'énergie en transmettant les messages ce qui rend l'utilisation des protocoles de routage traditionnels des réseaux ad-hoc impraticable. Plusieurs technique sont utilisées dans le routage telles que : le data-centric, l'agrégation des données et le clustering [AKY 02]. Dans les prochains chapitres, une étude des différentes familles de protocoles de routage, ainsi que des exemples seront présentés.

Couche transport (*Transport layer*)

Le besoin de la couche transport est précisé dans la littérature [RAM 00]. Cette couche est particulièrement nécessaire quand le système est projeté pour être consulté par l'Internet ou d'autres réseaux externes. Cependant, jusqu'ici, il n'y a eu aucune tentative de proposer un arrangement ou de discuter les issues à la couche transport d'un réseau de capteurs.

Les objectifs principaux de cette couche sont : Multiplexer et démultiplexer les messages entre les applications et la couche réseau, réaliser un contrôle de haut niveau des données et réguler la quantité des données injectées dans le réseau.

Couche application (*Application layer*)

Bien que beaucoup de domaines d'application aient été définis et proposés pour les réseaux de capteurs, les protocoles de la couche application demeurent en grande partie un champ encore inexploré. Dans ce paragraphe, nous citons à titre d'exemples trois protocoles : SMP (*Sensor Management Protocol*), TADAP (*Task Assignment and Data Advertisement Protocol*) et SQDDP (*Sensor Query and Data Dissemination Protocol*) qui sont décrits dans [AKY 02].

Niveau de gestion de l'énergie (*Power management plane*)

La vie du nœud montre, une dépendance forte à l'égard de la vie de la batterie. La consommation de l'énergie dans un réseau de capteurs, peut être divisée selon l'ordre décroissant suivants [POM 04] : la communication (émission et réception), le traitement des données et enfin, l'acquisition ou la capture.

Cette partie gère la manière dont les nœuds utilisent leurs énergies. Par exemple le nœud doit se mettre en sommeil après la réception d'un message à partir d'un voisin. Il doit aussi informer ses voisins de son incapacité à participer dans le routage quand sa réserve d'énergie atteint un seuil particulier.

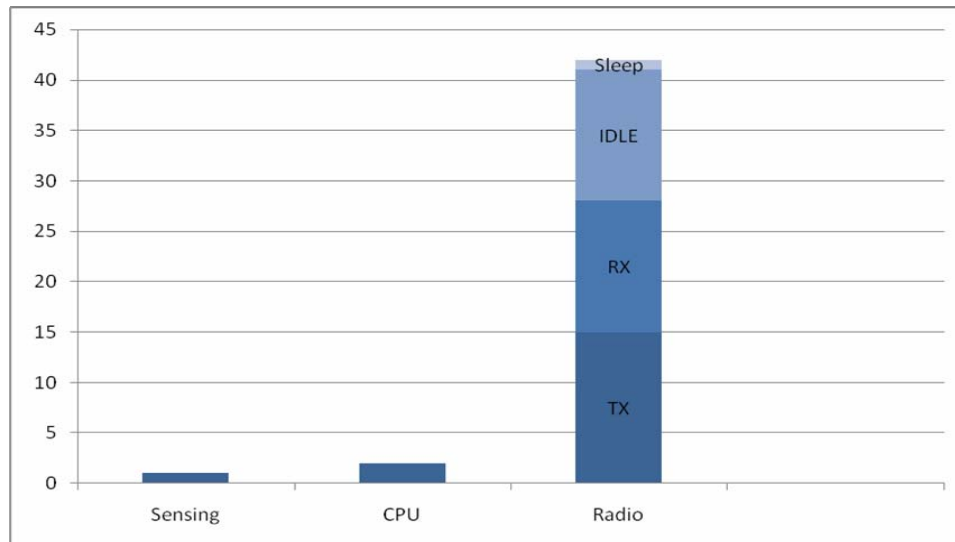


Fig. 1.6 Consommation d'énergie dans les WSNs [POM 04].

Niveau de gestion de la mobilité (*Mobility management plane*)

Puisque les nœuds peuvent être mobiles, un système de gestion de mobilité doit exister. Un tel système doit être capable d'enregistrer les mouvements du nœud afin de l'aider à se localiser. Parfois l'utilisateur souhaite changer l'emplacement du nœud, ou bien une nouvelle topologie du réseau nécessite une auto-organisation des nœuds. Dans ce cas, le système de gestion de mobilité doit être capable de commander le nœud pour réaliser les mouvements nécessaires.

Niveau de gestion des tâches (*Task management plane*)

Le niveau de gestion des tâches équilibre la distribution des tâches de captage sur les différents nœuds concernés afin d'optimiser la consommation d'énergie. En effet, la nature des capteurs ou leur niveau d'énergie implique que les nœuds peuvent avoir des rythmes différents de travail [LAA 07].

1.6 Réseaux de capteurs vs les réseaux MANETs

Bien que beaucoup de protocoles et algorithmes ont été proposés pour les réseaux MANETs, ces derniers ne sont pas adaptés pour les WSNs à cause des différences qui existent entre ces deux types de réseaux. Le tableau qui suit illustre une comparaison entre les réseaux de capteurs et les réseaux MANETs :

Réseaux de capteurs	Réseaux MANETs
Utilisation d'un médium sans fil.	Utilisation d'un médium sans fil.
Déploiement ad hoc.	Déploiement ad hoc.
Robuste aux pannes des nœuds (Auto-organisation).	Robuste aux pannes des nœuds (Auto-organisation).
Routage multi-sauts.	Routage multi-sauts.
La mobilité des nœuds est restreinte.	Grande mobilité des nœuds.
Grand nombre de nœuds (de l'ordre de mille).	Nombre de nœud moyen (de l'ordre de cents).
Des petits nœuds plus susceptibles aux pannes, avec moins capacité de traitements et de stockage.	Des nœuds ayant plus de capacité de traitements et de stockage.
<i>Data-centric</i> : souvent pas d'adresses uniques, les requêtes sont envoyés à tous les nœuds.	<i>Adress-centric</i> : une adresse unique pour chaque nœud utilisée pour réaliser la communication entre les nœuds.
Utilisation du broadcast.	Communication point à point.
Flot de données « <i>Many-to-one</i> ».	Flot de données « <i>Any-to-any</i> ».
L'énergie est un facteur déterminant.	Débit est majeur.
Nœuds collaborent pour remplir un objectif commun.	Chaque nœud a son propre objectif.
Les nœuds agrègent les données avant de les transmettre.	Généralement pas d'agrégation de données.
Les protocoles de routage doivent prendre en considération certaines contraintes : le grand nombre de nœuds, ressources limitées et la consommation d'énergie.	Plusieurs types de protocoles de routage peuvent être utilisés.

Tab. 1.1 Réseaux de capteurs vs les MANETs [ILY 05].

1.7 Domaines d'application des réseaux de capteurs

1.7.1 Applications militaires

Depuis le 11 septembre 2001, les soucis de sûreté face aux menaces du terrorisme international ont augmenté considérablement les recherches relatives aux WSNs incluant la détection et la prévention des attaques terroristes. Dans [XU 04], des capteurs de rayonnement peuvent être déployés dans des zones urbaines, par exemples aux intersections de routes, pour détecter d'éventuelles attaques terroristes utilisant des substances radioactives. Les WSNs militaire peuvent détecter des informations sur les positions et mouvements de l'ennemi etc.

1.7.2 Habitat

Les WSNs sont déployés pour le contrôle et la détection de problèmes structuraux dans les bâtiments, les immeubles. Ceci facilite la tâche de maintenance des édifices et permet d'empêcher les désastres.

1.7.3 Applications environnementales

Les WSNs peuvent être employés pour détecter et observer les changements environnementaux : les plaines, les forêts, les océans, les niveaux d'inondation l'agriculture etc. [MAR 04, BUR 04, CHA 04, SHA 04, BRI 04, SZE 04]. La caractéristique commune de ces applications est que les données capturées sont agrégées dans les serveurs et distribuées aux utilisateurs avec d'autres informations telles que l'endroit et les conditions ambiantes de leur capture. Un exemple est *GlacsWeb* [MAR 04] qui surveille des glaciers afin d'observer des changements, probablement provoqués par le réchauffement climatique. Burrell [BUR 04] emploie des sondes pour analyser les mouvements des fermiers pendant le travail dans les champs de vignes. Zhang [ZHA 04] a développé *ZebraNet* pour observer les traces des animaux sauvages à l'aide de capteurs.

1.7.4 Applications médicales

Des capteurs implantables peuvent surveiller les états des patients sans interruption à tout moment, ceci à un impacte direct sur le taux de rétablissement des patients. Pour la survie des sujets de trauma, il est particulièrement important que les médecins puissent faire un diagnostic rapide et précis pour recommander le traitement approprié. La télésurveillance peut également permettre l'essai et le début du traitement dans les sites éloignés, aussi bien qu'aider aux endroits précis des emplacements d'accidents ou de désastre [YON 05].

1.7.5 Applications industrielles

Pour les entreprises manufacturières, les réseaux de capteurs permettent de suivre le procédé de production à partir des matières premières jusqu'au produit fini. Ceci permettrait d'offrir une meilleure qualité de service. D'autres applications sont possibles, telles que le contrôle et la maintenance des équipements industriels [LAA 07].

1.8 Quelques axes de travaux pour les réseaux de capteurs

Le domaine des réseaux de capteurs est encore prometteur et nécessite plusieurs travaux pour arriver à un certain degré de stabilité. Dans cette section nous allons présenter quelques axes de recherche pour les WSNs.

1.8.1 Systèmes d'exploitation

Beaucoup de groupes de recherches, dans les milieux universitaire et industriel sont actuellement consacrés au développement des plateformes pour les capteurs. Un des premier et plus populaire des systèmes est celui développé par l'université de Berkeley TinyOS [HIL 00]. C'est un système d'exploitation *Open Source* conçu pour les capteurs, il repose sur : une architecture basée composant, un modèle de programmation basé évènement et un modèle de concurrence basé sur des évènements et des tâches. La bibliothèque de composants de TinyOS est particulièrement complète puisqu'on y retrouve des protocoles réseaux, des pilotes de capteurs et des outils d'acquisition de données. Un programme s'exécutant sur TinyOS est constitué d'une

sélection de composants systèmes et de composants développés spécifiquement pour l'application à laquelle il sera destiné (mesure de température, du taux d'humidité...).

Le NESC [GAY 03], qui est une extension du langage C peut être employé sur TinyOS, celui-ci possède des dispositifs tels que : conduite des événements, exécution parallèle et éléments d'orientation etc. Ces derniers correspondent aux composants et au traitement d'évènement de TinyOS. Greenstein et autres [GRE 04] proposent une bibliothèque de composants SNACK (*Sensor Network Application Construction Kit*) pour faciliter le développement sur TinyOS. Dai et autres [DAI 04] ont conçu un système de fichier ELF (*Efficient Log structured Flash file system*) pour le stockage en utilisant la mémoire flash.

1.8.2 Agrégation des données

L'une des caractéristiques des réseaux de capteurs est la possibilité de réduire la quantité de données circulant dans le réseau, afin de conserver de l'énergie, en fusionnant les données par des nœuds particulier du réseau. Ce processus est appelé agrégation de données. L'agrégation exige, non seulement, la transmission des données mais aussi les messages de contrôle, ce qui impose des contraintes sur l'architecture du réseau. Pendant l'agrégation nous devons aussi prendre en considération quelques problèmes : Les erreurs dans les messages, les messages perdus, la redondance des données, la synchronisation entre les nœuds ...etc. Dans ce qui suit nous présentons quelques travaux dans ce sujet.

Karl et all. [KAR 03] ont proposé une *framework* pour l'agrégation de données dans laquelle un mécanisme de contrôle des erreurs est assuré. Chaque message contient trois parties : la valeur des données, le nombre des nœuds qui ont contribuées à cette valeur, et la zone qui forme l'ensemble des nœuds qui doivent contribuer dans l'opération de l'agrégation. Ding et al. [DIN 03] ont proposé une méthode heuristique afin de construire et maintenir un arbre d'agrégation dans un réseau de capteurs sans fil. Les auteurs dans [KAR 04] ont introduit deux algorithmes, exacte et approximatif, pour l'agrégation des données. Le processus se fait en deux niveaux. Premièrement, des

agrégateurs locaux sont utilisés pour fusionner les données reçues à partir des nœuds locaux, puis des agrégateurs maîtres sont choisis de façon optimale pour réaliser le deuxième niveau d'agrégation. Motegi et al. [MOT 06] ont proposé une méthode pour l'agrégation de données qui est divisé en deux phases : la première consiste en la formation d'un graphe acyclique orienté DAG (*Directed Acyclic Graph*) qui permet aux nœuds d'avoir plusieurs parents pour la collection des données. Dans la seconde, pendant la transmission des données, le graphe est étendu en utilisant un contrôle temporel selon le nombre de saut actuel dans le graphe

1.8.3 Clustering

Le clustering consiste à découper le réseau en groupes de nœuds appelés *Cluster*. Pour chaque cluster un nœud maître appelé « cluster-head » est élu selon l'état courant du réseau afin d'accomplir des tâches spécifiques. Il existe deux configurations possibles pour les clusters : dans la première, les membres du cluster ne communiquent qu'avec le cluster-head, par contre dans la deuxième, ils peuvent utiliser d'autres nœuds comme passerelles vers le cluster-head, comme le montre la figure suivante :

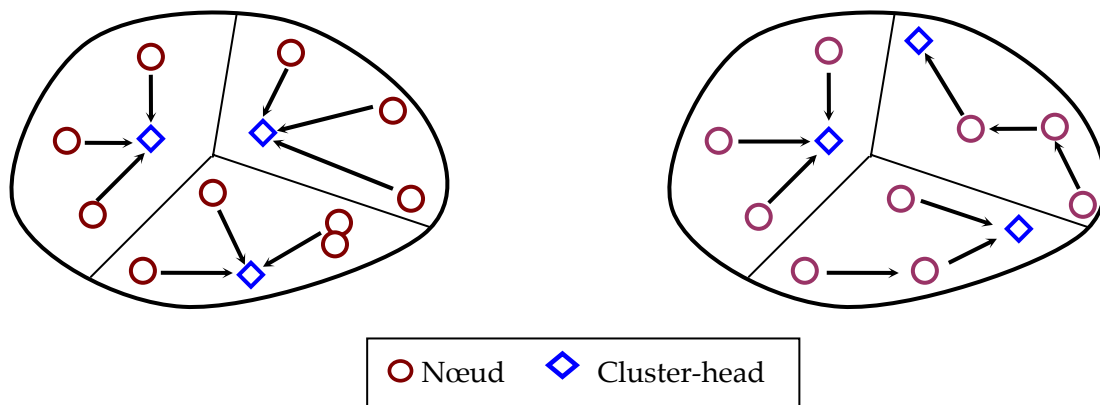


Fig. 1.7 Configurations des cluster-heads

Le clustering aide à minimiser la consommation d'énergie en réduisant la quantité des données circulées par l'agrégation. De plus, il permet de faciliter la localisation des nœuds [CHA 04]. Plusieurs techniques sont proposées pour la construction des clusters

et le choix du *cluster-head*. Ghiasi et al. [CHI 02] ont étudié les aspects théoriques du problème de clustering en considérant l'énergie comme paramètre d'optimisation. Ils ont modélisé le problème qu'ils appelaient *balanced k-clustering* comme un problème *min-cost flow* qui peut être résolu de façon optimale en utilisant les techniques existant pour ce problème. Dans [BOU 03] un schéma de clustering est illustré. Il prend en considération des informations sémantiques plus des propriétés de connectivités. La construction de l'arbre de cluster se fait en trois phases. La première est la propagation des requêtes dans laquelle chaque nœud satisfaisant une requête et des contraintes prédéfinies décide d'être un cluster-head. Dans la phase de formation des clusters, chaque cluster-head diffuse une annonce aux nœuds voisins les invitant à être membres de son cluster. La dernière phase est la diffusion des données des nœuds vers leurs cluster-head suivant un schéma d'arbre.

1.8.4 Localisation des nœuds

Quand l'information est obtenue par un certain nœud, sa position physique est celle à laquelle cette dernière est capturée. Par conséquent, la position d'un nœud dans un WSN est l'information essentielle pour analyser l'information capturée.

Diverses recherches se sont concentrées sur la localisation des nœuds. Par exemple, Moore [MOR 04] propose un algorithme qui localise avec succès des nœuds dans un WSN avec des mesures de distance bruyantes, pas à l'aide d'aucune balise ou ancre. Shang [SHA 04] et Ji [JI 04] emploient MDS (*Multi Dimensional Scaling*) pour identifier les nœuds. Ceci utilise l'information de connectivité pour savoir quels sont les nœuds impliqués dans telle ou autre communication, pour dériver les endroits des nœuds dans le WSN. Cette méthode permet aussi de tirer profit des données additionnelles : les distances estimées entre les voisins ou les positions connues pour certains nœuds d'ancre. Cheng [CHE 04] propose TPS (*Time based Positioning Scheme*), TPS se fonde sur TDoA (*Time Difference of Arrival*) des RF (*radio frequency*) des signaux mesurés localement à un capteur pour estimer les positions des nœuds.

1.8.5 Synchronisation

La gestion du temps dans les WSNs est essentielle pour l'accomplissement des tâches telles que l'estampillage des événements, le contrôle des cycles d'opérations ainsi que leur synchronisation. Le NTP (*Network Time Protocol*) [MIL 91] permet l'attribution des estampilles en temps réel en donnant les erreurs maximales, ainsi, l'ordre global et partiel des événements peut être obtenu avec une exactitude basé sur les estampilles. Le NTP exige l'échange fréquent de messages de synchronisation des horloges dans chaque nœud. Dai et autres [DAI 04] ont proposé une architecture *TSync*, où des mécanismes de synchronisation *Puch and Pop time* sont combinés. Li [LI 04] propose trois méthodes pour réaliser la synchronisation globale dans un WSN qui sont : NBA (*Node Based Approach*), HCBM (*Hierarchical Cluster Based Method*) et FLDBM (*Fully Locad Diffusion Based Method*).

1.8.6 Sécurité

La sécurité dans les WSNs est l'une des issues critique, mais un minimum de recherche a été fait comparer à d'autres aspects. Les ressources des nœuds capteurs sont limitées d'où une technologie différente de sécurité pour les WSNs est exigée. Avancha et autres [AVA 02] fournissent un bon sommaire montrant les directions de recherches en sécurité pour les WSNs. Wood [WOO 03] présente un aperçu sur beaucoup de genres d'attaques DoS (*Denial of Service*) dans les WSNs et discute des technologies de défenses. Karloff a implémenté TinySec [KAR 04] comme module de chiffrage pour la couche liaison de données, celui-ci réalise l'opération sur les ressources de TinyOS qui sont très limitées.

1.8.7 Routage

Beaucoup de protocoles de routage ont été proposés pour les WSNs. Ces derniers prennent en compte plusieurs facteurs, notamment la conservation d'énergie (pour prolonger la durée de vie du réseau de capteurs) et la topologie dynamique des WSNs.

Le chapitre qui suit est entièrement dédié à l'étude du routage dans les WSNs, ainsi des exemples de protocoles seront présentés dans le deuxième et troisième chapitre.

1.9 Conclusion

Le développement des réseaux de capteurs a été rendu possible grâce aux avancées importantes des technologies de réseaux sans fil et de l'électronique digitale. Les WSNs sont composés d'un nombre important de nœuds, contenant chacun un capteur, un processeur et une interface de communication et sont soumis à des contraintes qui proviennent des faibles ressources physiques disponibles au sein de chaque capteur et concernant aussi bien la réserve de l'énergie, la capacité de calcul ou la taille de la mémoire. Ces réseaux peuvent se révéler très utiles pour de nombreuses applications militaires ou civiles lorsqu'il s'agit de collecter et traiter des informations provenant de l'environnement.

Aujourd'hui, les WSNs constituent un champ de recherche très vaste vu leurs importances. Pour l'aspect informatique, les travaux se portent sur le développement de nouvelles techniques de sécurité, des méthodes de synchronisation, des protocoles et algorithmes de communication notamment le routage efficace en terme de consommation de l'énergie. L'objectif de ces recherches est d'optimiser les performances des WSNs tout en prolongeant leurs durées de vie.

La consommation d'énergie est un facteur à prendre en considération dans la conception des réseaux de capteurs. Puisqu'une grande partie de cette ressource critique est utilisée pendant la communication, plusieurs protocoles de communication en particulier ceux relatifs au cheminement de données ont été proposés afin de minimiser considérablement la dissipation de l'énergie. L'objectif du prochain chapitre est de présenter une étude des différentes familles des protocoles de routage dédiés aux réseaux de capteurs.

Chapitre 2

Routage dans les réseaux de capteurs

2.1 Objectif

La problématique du routage dans les réseaux sans fil ressemble à celle du routage dans les réseaux traditionnels. Il s'agit d'aiguiller les paquets vers leur destination. Pour connaître la topologie du réseau, les nœuds échangent des paquets comportant de l'information topologique. En effet, le problème de routage consiste à déterminer un acheminement optimal des paquets à travers le réseau selon un certain critère de performance.

Dans les réseaux de capteurs, la conservation d'énergie, qui a une grande influence sur la durée de vie du réseau peut être considérée plus importante que les performances du réseau. Donc les protocoles de routages doivent permettre une consommation minimale de l'énergie sans dégrader considérablement les performances du réseau.

2.2 Éléments de conception d'un protocole de routage

En dépit des nombreuses applications des WSNs, ces réseaux ont plusieurs restrictions à titre d'exemples : les contraintes d'énergie, la puissance de calcul souvent limitée et la largeur de la bande passante radio entre les nœuds qui est limitée. Un des buts principaux lors de la conception de protocoles de routage pour les WSNs est d'effectuer les communications tout en essayant de prolonger la durée de vie du réseau et d'empêcher la dégradation de connectivité en utilisant des techniques de gestion efficaces d'énergie. La conception de ces protocoles est influencée par certaines contraintes qui doivent être surmontées avant que la communication puisse être réalisée. Dans ce qui suit, nous récapitulons certains des défis à surmonter pour la conception de protocoles de routage.

Déploiement des nœuds

Le déploiement des nœuds est une considération importante. Les nœuds du réseau peuvent être installés d'une manière « prédéterminée » ou « aléatoire ». Un autre aspect de déploiement des nœuds est la mobilité éventuelle des nœuds.

Consommation d'énergie

L'énergie des capteurs ne pouvant être renouvelée, les opérations de capture, de traitement et de routage doivent être correctement évaluées afin d'éviter l'épuisement prématuré de cette énergie et de prolonger la durée de vie des nœuds et par là-même éviter le partitionnement du réseau [HEI 00]. L'émission et la réception étant les opérations les plus coûteuses en énergie, il est en général impossible pour un nœud quelconque de transmettre directement l'information vers la station de base. C'est la raison pour laquelle la communication dans les WSN est multi-sauts. Chaque nœud joue à la fois son rôle de capteur, mais également un rôle de relais dans le routage des informations vers la station de base.

L'arrêt de fonctionnement de quelques nœuds dus à l'épuisement de la batterie peut entraîner des changements topologiques du réseau.

Modèle de données rapportées

Les données capturées et transmises dans les WSNs dépendent de l'application et de la contrainte liée au temps de transmission. La capture et la transmission de données peut être classée dans l'une de catégories suivantes : dirigée par le temps : (*time-driven*), événementielle (*event-driven*), suite à des requêtes de la station de base (*query-driven*), ou hybride [YAO 02]. Le modèle continu correspond aux applications qui nécessitent des données capturées périodiquement. Dans les modèles événementiel et à base de requête, les nœuds capteurs réagissent à des événements tels que des changements des valeurs de la grandeur mesurée en raison d'événement (exemple, élévation brusque de la température) ou de requêtes produites par la station de base. Une combinaison des modèles précédents est également possible.

Hétérogénéité des nœuds

De nombreux travaux supposent que tous les nœuds sont homogènes en termes de calcul, de paramètres et de puissances de communication. Cependant, selon l'application un nœud peut avoir un rôle ou des possibilités différentes. Par exemple,

certaines applications peuvent nécessiter des capteurs de natures différentes pour surveiller la température, la pression et l'humidité de l'environnement, détectant le mouvement par l'intermédiaire de signatures acoustiques, et capturant l'image ou le cheminement visuel d'objets mobiles. Ces capteurs peuvent inclure des fonctionnalités différentes.

De même, la capture et la transmission des données peuvent être effectuées par ces capteurs avec des caractéristiques différentes. Par exemple, les protocoles hiérarchiques nécessitent des nœuds « cluster-heads » différents des nœuds capteurs classiques. Ces cluster-heads ont un rôle particulier de routage des données transmises souvent en raison de ressources plus importantes que les nœuds classiques en termes d'énergie, bande passante, ou de mémoire.

Tolérance aux fautes

Certains nœuds peuvent tomber en panne ou être bloqués par manque d'énergie, ou en raison de dommages physiques, ou d'interférence environnementale. La défaillance de certains nœuds capteurs ne doit pas affecter la tâche globale du réseau. Si plusieurs nœuds sont en panne, les protocoles de routage doivent s'adapter à la formation de nouveaux liens et router les données collectées aux stations de base. Ceci peut impliquer l'ajustement des puissances de transmission et modifier la vitesse de transmission sur les liens existants pour réduire la consommation d'énergie, ou router différemment des paquets à travers des régions du réseau où plus d'énergie est disponible. Par conséquent la redondance peut être nécessaire dans un réseau de capteurs pour le rendre moins sensible aux défaillances.

Scalabilité

Le nombre de nœuds capteurs déployés dans une région peut être de l'ordre de quelques dizaines à plusieurs milliers. Le schéma de routage doit pouvoir répondre à ces différences d'échelle entre les différents réseaux.

Dynamacité du réseau

Les réseaux de capteurs peuvent être dynamiques ou statiques. Dans les réseaux de capteurs statiques, les composants demeurent stationnaires. C'est habituellement le cas dans la plupart des réseaux de capteurs, par exemple pour surveiller la température ou l'humidité d'une salle ou bien d'une région. Dans les réseaux de capteurs dynamiques, au moins l'un des composants est mobile, exemple un mouvement dépistant le système de sécurité [AKK 03].

Média de transmission

Dans un réseau de capteurs multi-sauts les nœuds communiquant sont liés par un médium sans fil. En général, la largeur de bande requise des données de capteur sera basse, sur l'ordre de 1-100 kb/s. Les problèmes classiques liés au canal sans fil (par exemple un taux d'erreur plus élevé en raison de la présence d'interférences) peuvent affecter le fonctionnement du réseau.

Connectivité

La densité parfois élevée des nœuds dans les réseaux de capteurs exclut complètement l'isolement entre eux. Par conséquent, on s'attend à ce que des nœuds de capteurs soient fortement reliés. Ceci, n'élimine pas le risque du partitionnement du réseau dû aux échecs des nœuds capteurs.

Couverture

Dans les WSNs, chaque nœud capteur a une vue restreinte et limitée de l'environnement surveillé. Par conséquent, la couverture d'une région est un paramètre de conception important dans les WSNs.

Agrégation de données

Selon la densité des nœuds dans l'espace couvert, les données mesurées peuvent être identiques (lorsqu'elles sont relevées par des capteurs très proches les uns des autres). Pour réduire le nombre de transmissions et donc économiser de l'énergie, les

données identiques peuvent être agrégées. L'agrégation de données est la combinaison des données de différentes sources selon une certaine fonction d'agrégation, par exemple, qui supprime les valeurs identiques, transmet le minimum, le maximum ou la moyenne [KRI 02].

Qualité de service

Dans certaines applications, la durée de vie des données mesurées est limitée. Par conséquent, la ³latence dans la transmission des données vers la station de base est une autre condition pour les applications qui sont soumis à des contraintes de temps.

2.3 Métriques de mesure d'efficacité des protocoles de routage

Dans cette section, nous allons présenter les métriques pour mesurer l'efficacité des protocoles de cheminement de données dans les WSNs. Les trois premières métriques sont applicables à toutes les architectures de réseau sans fil statiques, alors que les cinq suivantes sont utilisées pour les environnements où l'énergie est importante [SCH 05].

Nombre de sauts

C'est la métrique la plus utilisée dans la gestion des réseaux. Elle représente le nombre de nœuds qu'une transmission traversera pour transférer les données à partir de la source vers la destination.

L'inconvénient principal de cette métrique est qu'elle ne mesure pas, ou ne tient pas compte de la largeur de la bande passante qui est disponible entre les nœuds. Elle ignore les longs chemins (qui ont un nombre élevé des sauts) même si certains présentent des bandes passantes plus larges que celle des plus courts chemins.

Temps de traversée d'un saut

Cette métrique mesure le temps d'aller-retour des requêtes envoyées aux nœuds voisins. Cette métrique peut être calculée en ayant un nœud qui va envoyer un paquet

³ Latence : délai d'acheminement des données

de requête avec une estampille à l'un de ses voisins chaque 500ms. Quand le voisin reçoit le paquet, il le renvoie à l'expéditeur. En comparant l'estampille avec la durée du retour, la qualité du lien peut être évaluée. Naturellement les résultats de ce test peuvent être altérés par la charge sur les deux nœuds.

Différence en temps d'arrivée de deux paquets par saut

Cette métrique est une amélioration de la métrique précédente, car elle prend en compte la charge entre deux voisins qui peut modifier les résultats. Cette métrique peut être calculée en ayant un nœud examinateur qui va envoyer à l'un de ses voisins deux requêtes toutes les deux secondes. La première requête doit être envoyée avant la deuxième. Le récepteur calcule la différence de temps entre la réception des deux paquets et fait un rapport à l'expéditeur. Ce dernier, à la réception des deux messages, peut calculer la différence des temps.

Puissance

Cette métrique est utilisée pour mesurer la quantité d'énergie dépensée pour acheminer un message de données depuis la source jusqu'à la destination. Pour employer cette métrique, on doit assigner des poids basés sur les coûts en termes d'énergie sur chaque saut possible entre les nœuds du réseau.

Temps du premier nœud à mourir

Cette métrique détermine le temps auquel le premier nœud utilise complètement son énergie. Elle n'est pas concernée par la défaillance d'un nœud dû aux raisons techniques.

Temps du dernier nœud à mourir

Elle détermine le temps où le dernier nœud du réseau a consommé toute son énergie.

2.4 Classification des protocoles de routage dans les réseaux de capteurs

Les protocoles de routage dans les WSNs peuvent être classifiés en suivant l'une des quatre manières : selon la structure de réseau, selon les fonctions des protocoles, selon l'établissement de route, et enfin selon l'initiateur de communication. La classification des protocoles de routage dans les WSNs est montrée dans la figure suivante [ALK 05, JIA 04] :

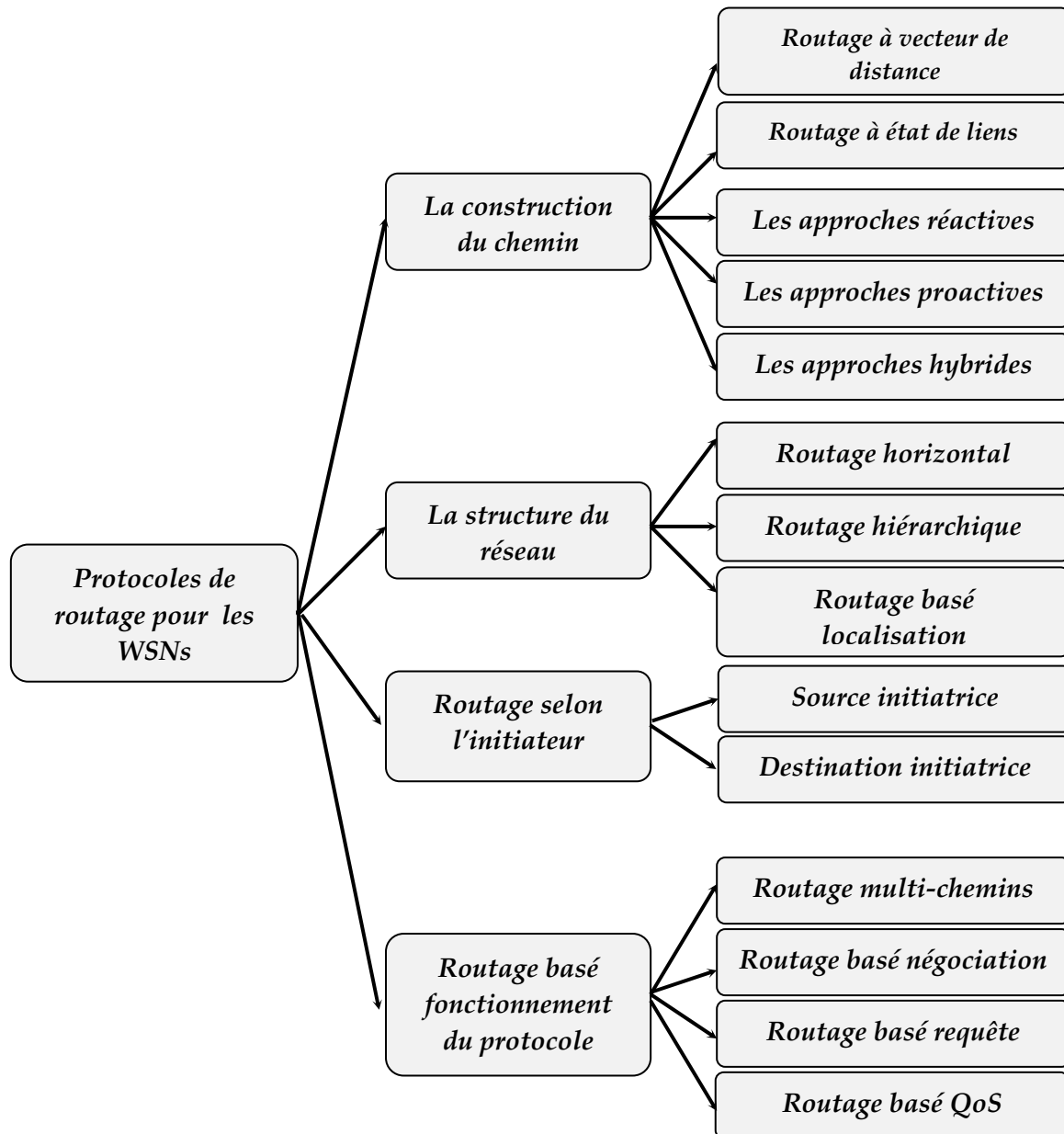


Fig. 2.1 Familles des protocoles de routage dans les WSNs [ALK 05, JIA 04].

2.5 Routage basé sur la construction des chemins

Cette famille de protocoles est spécifique au routage dans les réseaux Ad-hoc. Elle comprend deux manières de classer les protocoles. Dans la première, la classification se base sur l'information utilisée pour calculer les routes (routage à état de liens et routage à vecteur de distance). Dans la seconde, selon la manière de création et de maintenance de routes lors de l'acheminement des données, les protocoles de

routage peuvent être séparés en deux catégories : les protocoles proactifs et les protocoles réactifs.

2.5.1 Routage à état de liens

Il cherche à maintenir dans chaque nœud une carte plus ou moins complète du réseau où figurent les nœuds et les liens les reliant. A partir de cette carte, il est possible de construire les tables de routage. Un des avantages de ce type de protocole est la capacité de trouver des routes alternatives lorsqu'un lien est rompu. Il est même possible d'utiliser simultanément plusieurs routes vers une même destination, augmentant ainsi la répartition de la charge et la tolérance aux pannes dans le réseau. En contre partie, si le réseau est étendu, la quantité d'informations à stocker et diffuser peut devenir considérable.

2.5.2 Routage à vecteur de distance

Plutôt que de maintenir une carte complète du réseau (ce qui peut s'avérer extrêmement lourd), ces protocoles ne conservent que la liste des nœuds du réseau et l'identité du voisin par lequel il faut passer pour atteindre la destination par le chemin le plus court. A chaque destination possible sont donc associés un *next-hop* et une distance. Si un voisin envoie un paquet de contrôle dans lequel il indique être plus près d'une destination que le *next-hop* que l'on utilisait jusqu'alors, alors il le remplace dans la table de routage. Un des inconvénients de cette technique est qu'il est plus difficile de conserver plusieurs routes alternatives au cas où celle qui est privilégiée serait rompue (on ne dispose que du *next-hop*, et on ne sait pas si la suite de la nouvelle route est indépendante de celle qui a été rompue).

2.5.3 Approches réactives

Le principe des protocoles réactifs est de ne rien faire tant qu'une application ne demande pas explicitement d'envoyer un paquet vers un nœud distant. Cela permet d'économiser de la bande passante et de l'énergie. Lorsqu'un paquet doit être envoyé, le protocole de routage recherche un chemin jusqu'à la destination. Une fois ce chemin

trouvé, il est inscrit dans la table de routage et peut être utilisé. En général, cette recherche se fait par inondation (un paquet de recherche de route est transmis de proche en proche dans tout ou partie du réseau). L'avantage majeur de cette méthode est qu'elle ne génère du trafic de contrôle que lorsqu'il est nécessaire. Les principaux inconvénients sont que l'inondation est un mécanisme coûteux qui fait intervenir tous les nœuds du réseau en très peu de temps et qu'un délai est nécessaire pour l'établissement des routes [DHO 03].

2.5.4 Approches proactives

Le principe de base des protocoles proactifs est de maintenir à jour les tables de routage, de sorte que lorsqu'une application désire envoyer un paquet à un autre mobile, une route soit immédiatement connue. Dans le contexte des réseaux Ad-hoc les nœuds peuvent apparaître ou disparaître de manière aléatoire et la topologie même du réseau peut changer ; cela signifie qu'un échange continu d'informations est nécessaire pour que chaque nœud ait une image à jour du réseau. Les tables sont donc maintenues grâce à des paquets de contrôle, et il est possible d'y trouver directement et à tout moment un chemin vers les destinations connues en fonctions de divers critères. On peut par exemple privilégier les routes comportant peu de sauts, celles qui offrent la meilleure bande passante, ou encore celles où le délai est le plus faible. L'avantage premier de ce type de protocole est d'avoir les routes immédiatement disponibles quand les applications en ont besoin, mais cela se fait au coût d'échanges réguliers de messages (consommation de bande passante et d'énergie) qui ne sont certainement pas tous nécessaires (seules certaines routes seront utilisées par les applications en général) [DHO 03].

2.5.5 Approches hybrides

Combine les deux approches pour tirer profit de leurs avantages. Généralement, le réseau est divisé en deux zones et le principe est d'utiliser une approche proactive pour avoir des informations sur les voisins les plus proches, qui se trouvent au

maximum à deux sauts d'un nœud donné. Une approche réactive est utilisée au delà de cette zone prédéfinie afin de chercher des routes.

2.6 Routage basé sur la structure du réseau

La structure du réseau peut jouer un rôle significatif dans le fonctionnement d'un protocole de routage dans un réseau de capteurs. Dans cette section nous citons quelques protocoles qui rentrent dans cette catégorie.

2.6.1 Routage horizontal ou à plat

Il considère que tous les nœuds sont égaux comme le montre la figure 2.2. La décision d'un nœud de router des paquets vers un autre dépendra de sa position et pourra être remise en cause au cours du temps.

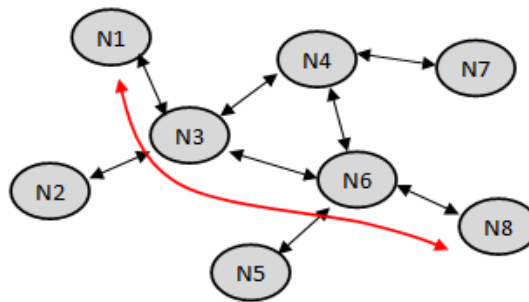


Fig.2.2 Routage horizontal [DHO 03].

Avantages du routage à plat

Le réseau à plat est scalable en raison du fait que chaque nœud participe équitablement à la tâche de cheminement et que les nœuds ont besoin seulement des informations sur leurs voisins directs. Un autre avantage est que les réseaux à plat rendent les protocoles de routage plus simples car il est possible d'établir le réseau sans beaucoup de frais et il n'y a aucun besoin d'algorithmes complexes de choix des *cluster-heads*.

Inconvénients du routage à plat

Si des nœuds sont uniformément distribués dans tout le réseau et s'il y a seulement une station de base, les nœuds autour de celle-ci épuiseront leurs sources

d'énergie plus tôt que les autres nœuds. C'est dû au fait que tout le trafic transite par les nœuds entourant la station de base.

2.6.2 Routage hiérarchique

Ce type de routage confie aux nœuds des rôles qui varient de l'un à l'autre. Certains nœuds sont élus et assurent des fonctions particulières qui conduisent à une vision en plusieurs niveaux de la topologie du réseau. Par exemple, un nœud pourra servir de passerelle pour un certain nombre de nœuds qui se seront attachés à lui. Le routage en sera simplifié, puisqu'il se fera de passerelle à passerelle, jusqu'à celle directement attachée au destinataire. Un exemple est donné sur la figure 2.3, où le nœud N3 passe par les passerelles P1, P2 et P3 pour atteindre N7.

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

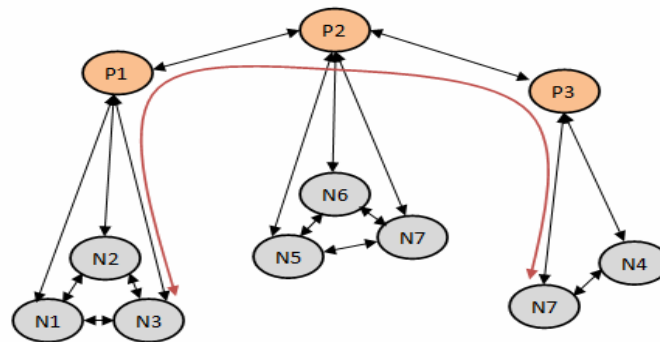


Fig.2.3 Routage hiérarchique [DHO 03]

Avantages du routage hiérarchique

L'avantage du routage hiérarchique est que les données d'un cluster entier peuvent être agrégées par le cluster-head avant d'être envoyées à la station de base.

Inconvénients du routage hiérarchique

Les nœuds élus cluster-heads consomment plus d'énergie que les autres nœuds du réseau. S'ils possèdent des caractéristiques identiques à celles des autres nœuds, et qu'il n'y a pas de rotation dans le choix des cluster-heads, alors ils risquent d'épuiser plus rapidement leur énergie et le réseau risque d'être divisé. De nombreux protocoles nécessitent que les cluster-heads aient plus de ressources et des niveaux d'énergie plus importants que les nœuds responsables uniquement de l'opération de capture. Dans le

cas où les cluster-heads ont les mêmes caractéristiques que les autres nœuds dans le réseau, alors il est nécessaire de recourir à des techniques pour élire régulièrement un nouveau cluster-head afin d'équilibrer la consommation d'énergie dans le réseau.

Enfin, les protocoles hiérarchiques ne s'adaptent pas bien au changement de densité de déploiement des nœuds (scalabilité) car à chaque fois les nœuds doivent s'organiser dans de nouveaux clusters. Par conséquent, les coûts induits par les messages lors de la formation des clusters et cluster-heads augmentent.

2.6.3 Routage basé sur la localisation

La famille des protocoles basés sur la localisation se fonde sur des informations géographiques externes (par exemple obtenues par GPS) et/ou sur la position des nœuds pour trouver des routes. Ce type d'approche peut présenter des inconvénients en milieu très urbain où la proximité géographique n'induit pas la proximité au sens radio c'est à dire la possibilité de communiquer. Néanmoins, couplées à des systèmes de localisation locaux, ces techniques peuvent être intéressantes [CHE 04].

2.7 Routage selon l'initiateur de la communication

La communication dans un WSN peut être lancée par les nœuds sources ou par les nœuds destinataires.

2.7.1 Source initiateur

Dans les protocoles lancés par la source, les nœuds envoient des données à la destination quand ils les ont capturées. Ces protocoles utilisent les données rapportées en fonction du temps (*time-driven*) ou d'évènements (*event-driven*). Ceci signifie que les données sont envoyées à certains intervalles ou quand les nœuds perçoivent certains événements.

2.7.2 Destination initiateur

Les protocoles où la communication est initiée par la destination utilisent les données rapportées à la suite d'une requête (*query-driven*). Dans ce cas, les nœuds

répondent aux requêtes envoyées par la destination ou un autre nœud. Les communications initiées par la destination créent un grand surcoût à cause de l'inondation des requêtes dans le réseau. Ceci signifie que chaque requête aura comme conséquence une diffusion (*flooding*) dans le réseau.

2.8 Routage basé sur le fonctionnement du protocole

Les protocoles de routage peuvent être classifiés selon leurs fonctionnalités en quatre catégories :

2.8.1 Routage multi-chemins

Ce sont des protocoles de routage utilisant des chemins multiples plutôt qu'un chemin simple afin d'augmenter la performance du réseau. La tolérance aux fautes (résilience) d'un protocole est mesurée par la probabilité qu'un chemin alternatif existe entre une source et une destination quand le chemin primaire est défaillant. Ceci peut être augmenté en maintenant des chemins multiples entre la source et la destination aux dépens d'une consommation d'énergie plus importante et d'une génération accrue du trafic. Ces chemins alternatifs sont maintenus par l'envoi périodique de messages. Par conséquent, la fiabilité du réseau peut être augmentée en maintenant les chemins alternatifs les plus récents.

2.8.2 Routage basé sur les requêtes

Dans ce type de routage, les nœuds destinataires propagent une requête de données de la part de la station de base à travers le réseau et le nœud qui détient ces données les envoie au nœud demandeur en suivant le chemin inverse de la requête. Tous les nœuds détiennent des tables contenant les intérêts sollicités par la station de base. Les nœuds envoient les données correspondantes aux requêtes dès qu'ils les capturent. Les protocoles *Directed Diffusion* et *Rumor Routing* sont deux exemples de ce type de routage.

2.8.3 Routage basé sur la négociation

Ces protocoles utilisent des descripteurs de données de haut niveau et des messages de négociation afin d'éliminer la transmission des données redondantes. La famille des protocoles SPIN est un bon exemple de ce type de protocoles.

2.8.4 Routage basé sur la qualité de service

Dans les protocoles de routage basés QoS, le réseau doit équilibrer entre la consommation d'énergie et la qualité de service. En particulier, le réseau doit satisfaire certaines métriques de QoS, par exemple : retard, largeur de bande passante, etc. en fournissant des données à la station de base.

2.9 Conclusion

Le problème de routage consiste à déterminer un acheminement proche de l'optimum des paquets à travers le réseau en respectant certaines contraintes (latence, ⁴gigue, tolérance aux fautes...). Dans les réseaux de capteurs, la conservation de l'énergie qui a une grande influence sur la durée de vie du réseau est considérée comme un paramètre essentiel. Donc les protocoles de routage doivent permettre une consommation minimale de l'énergie sans dégrader les performances du réseau. Néanmoins d'autres éléments doivent aussi être pris en considération lors de la conception de ces protocoles afin d'atteindre une communication efficace dans le réseau. Cette dernière est mesurée par les métriques citées précédemment.

Dans ce chapitre, nous avons donné un aperçu des manières de classier les protocoles de routage pour les WSNs. Les protocoles de routage dans les WSNs peuvent être classifiés :selon la structure du réseau, les fonctions des protocoles, l'établissement de route ou encore selon l'initiateur de la communication.

Dans le chapitre qui suit, nous présentons un état de l'art sur certains des principaux protocoles de routage dédiés aux WSNs, ainsi qu'une classification de ces derniers.

⁴ Gigue : variation des délais d'acheminement des paquets.

Chapitre 3

Les principaux protocoles de routage pour les réseaux de capteurs

3.1 Introduction

Le développement des protocoles de routage spécifiques aux WSNs a suscité de nombreux travaux dans le domaine. D'une part, parce que ces protocoles peuvent varier suivant les différentes approches de routage et la nature de l'application, et d'autre part, les solutions de routage proposées dans le cas des réseaux ad hoc ne sont pas applicables directement dans les WSNs. Bien que l'objectif principal reste le même, c'est à dire de trouver la façon la plus efficace d'acheminer des paquets de données d'une source à une destination, le routage dans les WSNs, en raison des contraintes d'énergie, peut obliger les concepteurs à faire des concessions sur certains paramètres de QoS.

Ce chapitre présente un état de l'art sur les protocoles de routage dans les WSNs. Il fournit une classification de ces protocoles suivant les différentes approches présentées dans le chapitre précédent, ainsi que leur comparaison selon un ensemble de critères qui ont trait à la consommation d'énergie et l'optimisation du trafic.

3.2 Les protocoles de routage dédiés aux WSNs

Les protocoles de routage minimisant la consommation d'énergie sont nombreux et variés. Chaque protocole est conçu de façon à fonctionner suivant des critères ou des conditions particulières à l'application. Dans cette section nous allons présenter une synthèse sur certains des principaux travaux qu'il est possible de trouver dans la littérature.

3.2.1 DD (Directed Diffusion)

La diffusion dirigée [INT 00] est le plus populaire des protocoles de routage connus dans les WSNs. Beaucoup d'autres protocoles qui ont été proposés se sont basés sur ce dernier. La diffusion dirigée est un protocole initié par la destination. Il s'applique aux réseaux ayant des structures plates et appartient aussi à la famille des protocoles réactifs. Le fonctionnement de la diffusion dirigée est centré sur les données. Des requêtes sont envoyées seulement à certains secteurs dans le réseau. Trois étapes

sont nécessaires pour ce protocole : la diffusion d'intérêt, l'installation de gradient et la livraison de données.

Durant la propagation d'intérêt, la station de base diffuse un intérêt (requête) pour certaines données vérifiant une condition bien précise (données requises). Ces données se composent des paires (attribut, valeur). A titre d'exemple, l'attribut peut être « Température » et la valeur peut prendre la valeur « 30 ». Ainsi, la requête peut être formulée comme suit : acheminer les données de tous les nœuds ayant mesuré une température supérieure à 30. Cette idée est une manière efficace d'éliminer la possibilité de recevoir des données peu désirées ou non pertinentes. L'intérêt initial indique également la fréquence à laquelle les nœuds doivent envoyer des données à la station de base, qui pourrait être toutes les dix secondes. Il inclut une estampille qui indique quand les nœuds peuvent cesser d'envoyer des données, par exemple après dix minutes. Chaque nœud ajoute l'intérêt à une liste d'intérêts qu'il maintient. Celle-ci contient une entrée pour chaque intérêt suscité. L'entrée d'un intérêt contient un identificateur de chaque voisin dont l'intérêt est suscité et le débit vers ce voisin.

Pendant la deuxième étape de la diffusion dirigée, les nœuds ayant des paires attribut-valeur commencent à envoyer des données à tous les voisins identifiés dans la liste des intérêts selon le débit indiqué (des gradients sont installés pour les intérêts, ils représentent les débits à utiliser pour l'envoi de données au sujet d'un intérêt spécifique à un voisin spécifique). La diffusion dirigée incorpore également l'agrégation de données. Les nœuds recevant des données dirigées à la station de base ajoutent ces données à un cache de données. A chaque réception d'un prochain message les nœuds vérifieront le cache pour voir si les données sont nouvelles, si ce n'est pas le cas, alors le message est négligé. Lorsque les données atteignent la station de base, un ou plusieurs chemins sont renforcés en envoyant d'autres intérêts. Un chemin peut être calculé en envoyant des données seulement au nœud duquel la réponse d'intérêt a été reçue la première fois, à chaque saut.

Enfin, pendant la dernière étape du protocole, un nœud qui a été renforcé envoie des données vers la station de base en respectant le débit spécifié. Ces données suivent le chemin qui a été établi. La figure 3.1 montre les étapes de la diffusion dirigée.

Un problème avec la diffusion dirigée est le surcoût associé au protocole. Les intérêts sont inondés, ce qui consomme de l'énergie. Les réponses initiales des nœuds sont également inondées et augmentent ainsi la consommation d'énergie. Un autre problème est la nécessité de disposer de mémoires de suffisamment grande capacité pour les nœuds afin de stocker la table des intérêts. Dans un grand WSN, le nombre de capteurs peut être de l'ordre de plusieurs milliers et il pourrait avoir beaucoup d'intérêts ce qui peut augmenter considérablement la taille de la table d'intérêts. Par conséquent, la diffusion dirigée ne peut pas être employée pour des applications où les données sont requises fréquemment et de tous les nœuds.

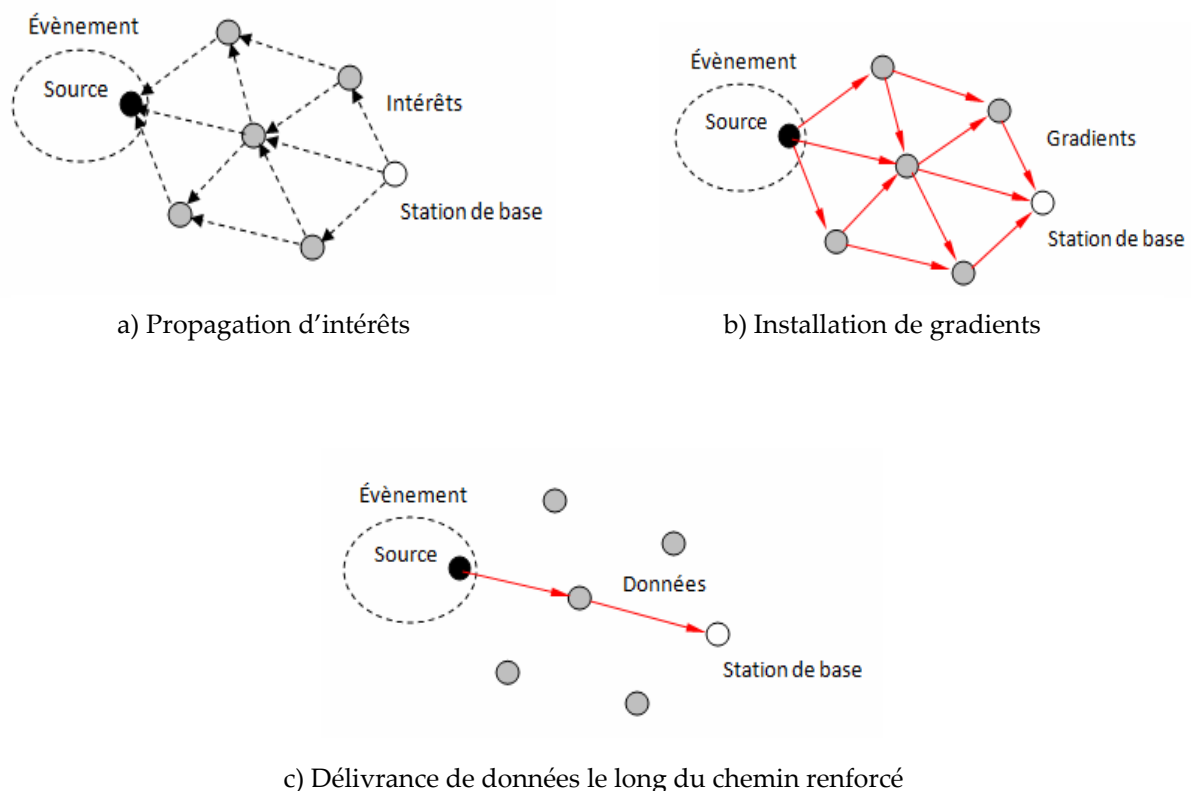


Fig. 3.1 Phases de la diffusion dirigée [INT 00].

3.2.2 RR (Rumor Routing)

Ce protocole [BRA 02] est dérivé de la diffusion dirigée. L'échange est initié par la destination. Il est utilisé dans les réseaux ayant des structures de cheminement plates. Son but est d'inonder les évènements plutôt que les intérêts. L'idée derrière cette approche est que l'intérêt peut être routé au nœud qui a observé l'évènement plutôt que d'inonder l'intérêt dans le réseau entier. Pour inonder un évènement, le cheminement de rumeur emploie des paquets qui ont une certaine durée de vie TTL (*Time To Live*), appelés les agents. Si un nœud détecte un évènement, il l'ajoute à sa table des évènements et transmet un agent avec un certain TTL, celui-ci contient aussi la table des évènements du nœud. Pendant que l'agent voyage dans le réseau, tous les nœuds le recevant mettent à jour leurs tables d'évènements. Si un agent observe un autre évènement, il met à jour sa table d'évènement et propage le nouvel évènement avec l'évènement original. Un agent maintient une liste des nœuds qu'il a visités, et choisit le prochain voisin qui n'est pas dans cette liste. Quand un agent arrive à un nouveau nœud, son TTL est décrémenté avant qu'il soit envoyé plus loin. Si le TTL atteint zéro, l'agent est éliminé et n'est plus transmis. Quand la station de base envoie un intérêt, elle l'envoie sur un chemin aléatoire jusqu'à ce qu'il trouve un nœud avec un chemin portant l'évènement exigé.

Ce protocole est prévu pour les réseaux avec peu d'évènements et beaucoup d'intérêts et ne garantit pas l'efficacité énergétique. La taille du paquet agent peut croître en raison du fait qu'il est possible que la table des évènements et la liste des nœuds visités aient des volumes importants. Un agent contient seulement un itinéraire à chaque évènement et donc si beaucoup de requêtes sont produites pour un évènement donné, les nœuds le long de ce chemin épuiseront leur source d'énergie rapidement. La manière dont les agents choisissent leurs prochains sauts influe également sur le réseau puisque des requêtes seront conduites le long du chemin choisi.

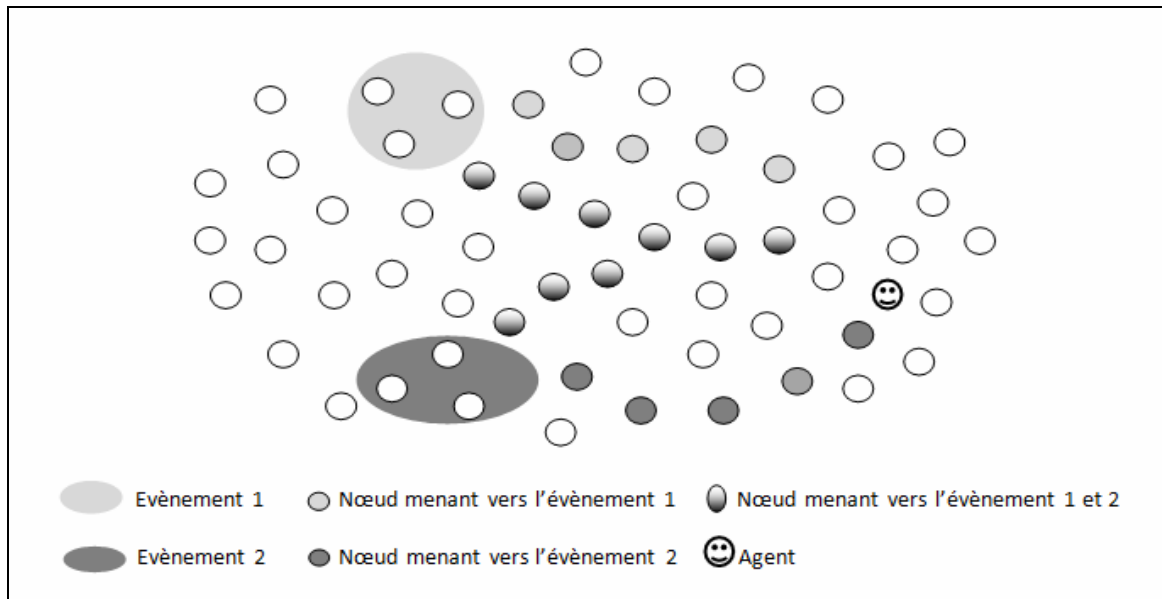


Fig. 3.2 Fonctionnement de Rumor Routing [BRA 01].

3.2.3 F&G (Flooding and Gossiping)

Flooding et *Gossiping* [HED 88] sont les deux premiers protocoles de routage appliqués pour les WSNs. Le *Flooding* (inondation) est une technique classique qui peut être employée pour le routage de paquets dans les réseaux de capteurs. Dans cette technique, chaque station retransmet les données reçues par diffusion. Ses inconvénients sont l'implosion, le chevauchement et le gaspillage de ressource. L'implosion se produit quand des messages doubles sont envoyés au même nœud. Le chevauchement quant à lui se produit lorsque deux nœuds qui se trouvent dans la même région envoient le même message à un voisin commun et le gaspillage de ressources est causé par des nœuds qui ne prennent pas en considération les contraintes d'énergie.

Gossiping évite l'implosion en choisissant d'une manière aléatoire un ensemble ou un sous ensemble de voisins pour envoyer des messages seulement à ceux-ci.

Ces deux protocoles n'ont pas été spécifiquement conçus pour les réseaux qui ont la contrainte d'énergie.

3.2.4 MCFA (Minimum Cost Forwarding Algorithm)

MCFA [YE 01] suppose que la direction du cheminement est toujours connue, c'est un protocole de routage à plat, lancé par la destination et appartient à la famille des protocoles proactifs. Les nœuds n'ont pas besoin d'avoir des identifiants uniques et au lieu des tables de cheminement, ils maintiennent la moindre estimation des coûts à la station de base. La station de base met son coût minimum à zéro tandis que tous les nœuds dans le réseau placent leurs coûts initiaux à l'infini. La station de base diffuse alors un message avec son coût. Un nœud recevant le message met à jour son coût si le coût dans le message plus le coût du lien qui le relie au nœud depuis lequel il a reçu le message est inférieur à son coût actuel (Le coût peut être le nombre de sauts, le délai ou une autre métrique). Si c'est le cas alors le message est rediffusé, sinon il sera jeté. Ceci établit un coût minimum à chaque nœud dans le réseau.

Quand un message de données doit être envoyé, celui-ci est diffusé par un nœud à tous ses voisins. Un nœud recevant ce message, vérifie s'il est sur le chemin du coût minimum à la station de base. Si c'est le cas alors le message est envoyé sur ce chemin jusqu'à ce que ce dernier atteigne la station de base.

Un des problèmes de MCFA est que les nœuds épuisent l'énergie le long de certains chemins si le coût minimum n'est pas mis à jour régulièrement. Un autre problème est que si le nombre de sauts est employé comme métrique de coût ou si des nœuds sont uniformément distribués et que la consommation d'énergie est employée comme coût, tous les nœuds se considèrent sur le chemin de coût minimum et le protocole est réduit à une simple inondation.

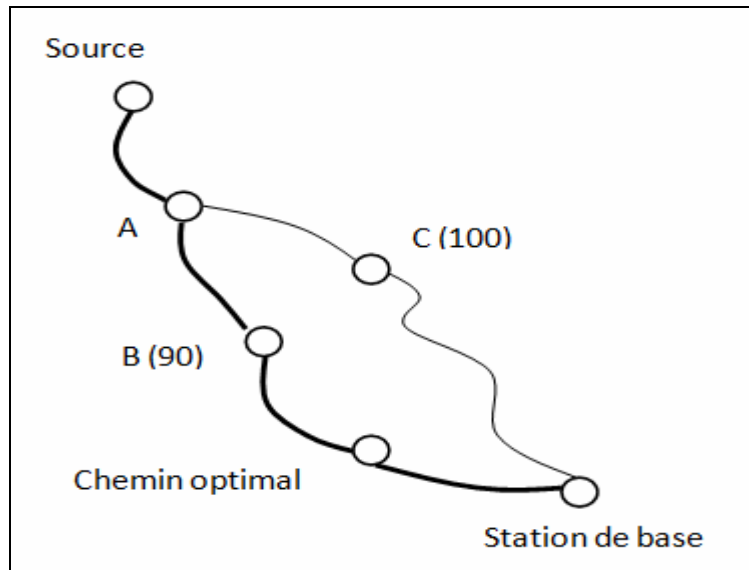


Fig. 3.3 Chemin de coût minimum dans MCFA [YE 01].

3.2.5 GBR (Gradient Based Routing)

Le routage basé sur le gradient [SCH 01] est un autre protocole basé sur la diffusion dirigée. GBR est un protocole lancé par la destination. Il s'applique dans les structures à plat et fait partie de la famille des protocoles réactifs. GBR ajoute le nombre de sauts à l'intérêt quand il est répandu par le réseau. Ceci permet à chaque nœud de calculer un paramètre appelé « hauteur du nœud », qui est le nombre minimum de sauts pour atteindre la station de base. Ce paramètre constitue le gradient. S'il y a des voisins multiples avec le même gradient, le protocole emploie alors l'une des trois méthodes pour choisir l'un d'entre eux :

- La première méthode consiste à choisir un voisin au hasard.
- La deuxième méthode augmente le gradient pour les voisins ayant des niveaux d'énergie au dessous d'un certain seuil.
- Enfin, la troisième méthode opte pour les voisins qui ont une charge en termes de trafic moins importante. L'objectif étant d'équilibrer la charge entre les nœuds.

GBR fait face aux mêmes frais d'inondation que la diffusion dirigée, puisque des intérêts sont encore inondés dans le réseau.

3.2.6 EAR (Energy Aware Routing)

Le protocole EAR [SHA 02] est semblable à la diffusion dirigée, mais il maintient des chemins multiples à chaque nœud au lieu d'un seul. C'est un protocole initié par la destination qui emploie une structure plate du réseau et appartient à la famille des protocoles proactifs. Des chemins multiples sont trouvés entre la source et les destinations. A chaque chemin est assignée une probabilité d'être choisi, selon la métrique de l'énergie. Chaque fois que des données doivent être envoyées, un des chemins est choisi selon les probabilités. La diffusion envoie des données le long de tous les chemins à intervalles réguliers, alors que dans EAR les données sont seulement envoyées le long d'un certain chemin. Ce protocole s'exécute en trois phases qui sont : l'installation, la communication des données et la phase de maintenance.

Phase d'installation

La station de base met son coût minimum à zéro puis initie la connexion par une diffusion dirigée vers les nœuds sources. Chaque nœud N_j recevant la requête de la part d'un voisin N_i calcule son coût pour atteindre la station de base à partir de ce voisin comme suit :

$$C_{N_j, N_i} = \text{Coût}(N_i) + \text{Métrique}(N_j, N_i)$$

Le nœud N_j assigne à chacun de ses voisins une probabilité inversement proportionnelle à son coût dans la table de cheminement FT_j (*Forwarding Table*) :

$$P_{N_j, N_i} = \frac{1/C_{N_j, N_i}}{\sum_{k \in FT_j} 1/C_{N_j, N_k}}$$

Ainsi, chaque nœud N_j possède un certain nombre de voisins par lesquels il peut conduire des paquets à la destination. N_j calcule alors son coût moyen pour atteindre la station de base :

$$\text{Coût}(N_j) = \sum_{i \in FT_j} P_{N_j, N_i} \times C_{N_j, N_i}$$

Phase de communication de données

Le nœud source envoie les données capturées via le voisin qui possède la plus grande probabilité dans la table de cheminement. Les nœuds intermédiaires font la même chose jusqu'à ce que les données atteignent leur destination.

Phase de maintenance des routes

La maintenance des routes est accomplie périodiquement par la station de base grâce à la diffusion dirigée.

La comparaison d'EAR avec la diffusion dirigée a démontré qu'il procure une amélioration globale de 21% d'énergie consommée et une augmentation de 44% de la durée de vie du réseau. Cependant, une telle utilisation des routes gêne la capacité de récupération d'un nœud lors de l'échec d'une route par opposition à la diffusion dirigée. En outre, l'approche exige de recueillir l'information de localisation et d'installer un mécanisme d'adressage pour les nœuds, ce qui complique la création d'itinéraires en comparaison avec la diffusion dirigée.

3.2.7 Chang

Chang [CHA 00] présente une solution au problème du routage dans les réseaux de capteur. Cette solution consiste à définir deux algorithmes basés sur le contrôle de flux des données. L'objectif principal de ces algorithmes est de maximiser la durée de vie du réseau en définissant soigneusement le coût du lien en fonction de l'énergie

restante du nœud et de l'énergie de transmission nécessaire en utilisant ce lien. Les deux algorithmes diffèrent dans leur définition du coût du lien de communication et de l'incorporation de l'énergie résiduelle des nœuds. La route de moindre coût est la route dont l'énergie résiduelle est la plus grande. Les algorithmes utilisant ces coûts de lien sont comparés à l'algorithme de transmission par minimum d'énergie MTE (*Minimum Transmission Energy*), qui ne considère que l'énergie consommée du lien.

Les résultats de simulation ont démontré que les deux algorithmes donnent une meilleure durée de vie moyenne que MTE. Ceci est dû au fait que la métrique que MTE emploie repose sur l'énergie résiduelle absolue, alors que la métrique utilisée par Chang porte sur l'énergie résiduelle relative qui reflète mieux le taux prévisible de la consommation d'énergie. L'inconvénient majeur de ces deux algorithmes réside dans le fait qu'ils sont des algorithmes hors-ligne ; ils considèrent que le taux de génération des données est connu à priori.

3.2.8 MECN (Minimum Energy Communication Network)

MECN [ROD 99] est un protocole basé localisation, il essaye de réduire la consommation de l'énergie en réduisant au minimum la puissance de transmission employée par les nœuds. C'est un protocole réactif, initié par la source et emploie une structure plate du réseau. Le protocole crée une région de relais pour les nœuds. Une région de relais pour un nœud se compose d'autres nœuds par lesquels le nœud peut transmettre des données vers la station de base d'une façon à économiser l'énergie (La région de relais est construite de façon à contenir que les voisins les plus proches). La puissance de transmission requise pour atteindre un nœud à la distance d dans n'importe quelle région d'un réseau sans fil en utilisant la communication par radio, est proportionnelle à d^2 [HEI 00]. A partir de là, le protocole choisit un chemin de la source à la station de base s'il a plus de sauts avec des distances plus courtes de transmission en supposant que les nœuds ont une puissance variable de transmission. Le protocole suppose également que chaque nœud dans le réseau est à portée de transmission de tous les autres nœuds dans le réseau, ce qui n'est pas pratique pour les WSNs. On suppose

aussi que les nœuds connaissent tous leurs positions et ont des récepteurs GPS, ce qui pousse les nœuds à consommer d'avantage d'énergie.

3.2.9 LEACH (Low Energy Adaptive Clustering Hierarchy)

LEACH [HEI 00] est l'un des protocoles de clustering fondamental, son exécution se fait en cycles, où chaque cycle se déroule en quatre étapes :

1. Dans la première étape un groupe de nœuds prédéterminés est élu pour former les cluster-heads et informent le réseau de leur disponibilités, les nœuds dans leur proximité décident dans quel cluster ils souhaitent appartenir en fonction d'une métrique présélectionnée. Pour être sélectionné comme cluster-head, un nœud choisit un nombre aléatoire compris entre 0 et 1, si ce nombre est inférieur à un certain seuil $T(n)$ alors le nœud est élu cluster-head pour le cycle courant. $T(n)$ est calculé en se basant sur une équation qui prend en compte : le pourcentage des nœuds désirant devenir cluster-heads P (La valeur de P est estimée à 5%), le cycle courant r et l'ensemble des nœuds qui n'ont pas été sélectionnés cluster-heads durant les $1/P$ cycles, cet ensemble est dénoté par G . La formule est donnée par :

$$T(n) = \frac{P}{1 - P(r \bmod (1/P))} \quad \text{si } n \in G$$

2. Dans la deuxième étape, les nœuds capteurs communiquent leurs décisions d'appartenir à un certain cluster aux cluster-heads en employant la technique CSMA MAC définie dans la section 2.5.2.
3. Durant la troisième étape et après la réception de tous les messages des nœuds désirant être membre d'un cluster, le cluster-head crée un ordonnancement TDMA (vue dans la section 2.5.2) qui sera diffusé à tous les membres du cluster, et assigne à chaque nœud un intervalle de temps.

4. Enfin dans la quatrième étape, les nœuds communiquent en respectant leurs intervalles de temps pendant un certain temps prédéterminé, ensuite le réseau va rentrer dans un nouveau cycle en revenant à la première phase.

LEACH suppose que les stations de base sont stationnaires et que le réseau est homogène (car chaque nœud peut devenir un cluster-head). Les avantages principaux de LEACH sont : l'emploi de l'agrégation de données pour réduire la quantité de l'information transmise entre les nœuds et la station de base, permet une meilleure auto-organisation des nœuds pour améliorer la scalabilité du réseau et l'emploi d'un mécanisme de choix de cluster-heads pour réduire l'épuisement d'énergie. L'inconvénient majeur de LEACH est l'hypothèse que les stations de base sont statiques qui limite son efficacité. De plus, le procédé d'élection rend ce protocole peu applicable aux applications temps-réel.

3.2.10 TEEN (Threshold Sensitive Energy Efficient Sensor Network Protocol)

TEEN [MAN 01] est un protocole orienté événements et basé sur le concept de clustering. Chaque nœud capteur possède deux valeurs seuil : HT (*Hard Threshold*) et ST (*Soft Threshold*) : la transmission débute lorsque la valeur mesurée excède HT, ou lorsqu'un changement de la valeur mesurée est supérieure ou égale à ST. TEEN suppose que tous les nœuds du réseau ont initialement le même niveau d'énergie et que les cluster-heads ont une source de puissance constante pendant qu'elles doivent attendre avec la survenue d'un événement sur les nœuds capteurs.

Les avantages de ce protocole sont : adéquat aux applications critiques en termes de temps, consomme moins d'énergie que les précédents car le nombre de messages transmis est réduit et enfin la possibilité de changer le ST selon les besoins.

Les inconvénients sont : si les seuils ne sont pas atteints les nœuds ne vont jamais se communiquer, les cluster-heads perdent la majorité du temps dans l'attente des données transmises par les autres nœuds au lieu de transmettre et enfin pas de

mécanisme de distinction entre les nœuds qui n'ont pas d'évènement entraînant une transmission de donnée de ceux qui sont tombés en panne.

3.2.11 Younis

Younis et al. [YOU 02] ont proposé un algorithme différent du routage hiérarchique. Il est basé sur une architecture à trois niveaux :

1. Les capteurs sont regroupés au sein de cellules ;
2. L'algorithme de routage utilise des cluster-heads, qui sont moins contraints en énergie que les capteurs et qui sont supposés connaître l'emplacement des capteurs ;
3. Les cluster-heads jouent le rôle de passerelles, maintiennent la connaissance de l'état des capteurs, établissent des itinéraires multi-sauts et collectent les données des capteurs.

Une couche MAC basée sur le TDMA (*Time division multiple access*), vue dans la section 2.5.2, est utilisée pour permettre aux nœuds d'envoyer leurs données à la passerelle. Cette dernière informe chaque nœud des espaces temporels (slots) dans lesquels ils doivent écouter la transmission des autres nœuds, et les autres slots disponibles pour déterminer leur propre transmission. Les auteurs supposent que le capteur est capable de fonctionner en mode actif ou en mode de veille (basse puissance). Les nœuds capteurs d'une cellule peuvent se trouver dans l'un des quatre états : capture uniquement, transmission seule, capture et transmission simultanées ou inactif. Dans l'état de capture, le nœud mesure l'environnement et produit des données à un taux constant ou à la survenue d'un évènement. Dans les autres cas, le nœud est considéré inactif et peut arrêter ses circuits de capture et de communication. La figure 3.4 montre un exemple de l'état des capteurs et des itinéraires dans une cellule typique pour une application de routage. Une fonction de coût est définie entre deux nœuds quelconques en termes de consommation d'énergie, de latence ou toute autre métrique de performance. L'utilisation de cette fonction de coût permet d'établir un lien (une route de moindre coût entre les nœuds capteurs et la passerelle).

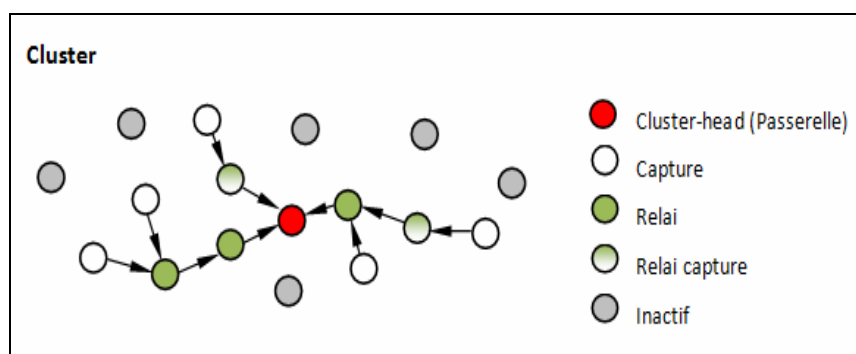


Fig.3.4 Les états des capteurs dans un cluster [YOU 02].

Les simulations ont montré que cette approche présente des résultats satisfaisants grâce à la prise en charge des métriques liées à la consommation d'énergie telle que la durée de vie globale du réseau, ainsi que d'autres métriques telles que le débit et les délais de transmission. Les résultats ont indiqué également que la combinaison de cette approche de routage avec une politique d'accès au canal basé sur TDMA augmente considérablement la durée de vie du réseau. Cependant, cette approche suppose un modèle de propagation simple, qui nécessite le déploiement d'un nombre suffisant de passerelles afin de permettre une bonne couverture des capteurs.

3.2.12 GAF (Geographic Adaptive Fidelity)

GAF [XU 01] est un algorithme de routage de localisation orienté énergie. Il a été conçu principalement pour les réseaux Ad-hoc mobiles. Il peut cependant être appliqué aux réseaux de capteurs. GAF économise l'énergie par la déconnexion des nœuds inutiles dans le réseau sans affecter l'opération de routage. Il constitue une grille virtuelle pour le domaine couvert par le réseau. Chaque nœud utilise le GPS pour indiquer sa position et l'associer à un point virtuel dans la grille. Les nœuds associés au même point de la grille sont considérés comme équivalents en termes de coût de routage des paquets. L'intérêt d'une telle équivalence est de n'utiliser qu'un seul nœud à la fois et de mettre les autres nœuds en sommeil afin d'économiser un maximum d'énergie. Ainsi GAF peut sensiblement augmenter la vie du réseau à mesure que le nombre de nœuds augmente. Les nœuds dorment aussi longtemps que leurs données

ne sont pas nécessaires. Les nœuds actifs transmettent leurs données mais effectuent également des répartitions équitables de la charge pour équilibrer la consommation d'énergie des nœuds d'un même point de la grille virtuelle.

3.2.13 HEAR (Hierarchical Energy Aware Routing)

L'objectif du protocole HEAR [HEM 05] est de minimiser la consommation d'énergie à travers l'utilisation de l'agrégation de données. Les auteurs affirment que les protocoles de cheminement qui sont centrés données ne sont pas applicables aux réseaux où les données sont nécessaires pour tous ou la plupart des nœuds dans le réseau. Les cluster-heads sont supposés être des nœuds puissants (plus de ressources en énergie et meilleurs équipements radio que les autres nœuds du réseau). Ces derniers envoient périodiquement une balise qui consiste en un paquet contenant le coût du lien entre lui et les nœuds qui se situent dans sa portée radio, les nœuds recevant cette balise évaluent l'information dans la balise et décident de la mise à jour de leurs tables de routage.

Une mise à jour se produit si l'information de lien est meilleure que l'information de lien précédemment stockée. Si la mise à jour est faite, le nœud retransmet la balise. Le paquet de balise est employé par des nœuds pour trouver le meilleur itinéraire à leur cluster-head. Le problème principal avec HEAR, comme avec beaucoup de protocoles hiérarchiques, est le fait que les nœuds cluster-heads doivent avoir un meilleur équipement radio et plus de ressources énergétiques que les autres nœuds dans le réseau.

3.2.14 GEAR (Geographic and Energy Aware Routing)

Yu et al [YU 01] ont suggéré l'utilisation d'informations géographiques afin de les disséminer vers des régions appropriées. Ce protocole utilise l'information sur le niveau d'énergie des nœuds et une heuristique de voisinage géographique pour choisir un chemin de transmission de paquets vers la région cible. L'idée consiste à limiter le nombre des intérêts à diffuser vers une certaine région plutôt que d'envoyer l'ensemble

des intérêts au réseau entier. GEAR est assimilable à la diffusion dirigée, dans le sens où il ajoute une fonction d'économie.

Dans GEAR chaque nœud conserve un coût estimé d'accès à la destination par l'intermédiaire de ses voisins. Le coût estimé est une combinaison de l'énergie résiduelle et de la distance au destinataire. Un coût d'apprentissage constitue une amélioration du coût estimé dans le sens où il intègre de façon plus affinée une métrique combinée de son énergie résiduelle et d'un chemin de routage qui tient compte des trous dans son voisinage. Un trou se produit quand un nœud n'a aucun voisin plus proche de la région cible autre que lui-même. S'il n'y a aucun trou, le coût d'apprentissage est égal au coût estimé. Le coût estimé est propagé chaque fois qu'un paquet atteint la destination de sorte que l'itinéraire pour le prochain paquet soit ajusté. L'algorithme de routage se décompose en deux phases :

La première phase consiste à expédier des paquets vers la région cible. Lors de la réception d'un paquet, le nœud vérifie ses voisins pour voir s'il y en a un qui soit plus proche de la région cible que lui-même. Dans l'affirmative, ce voisin plus proche est choisi pour le prochain saut. S'ils sont plus éloignés que le nœud lui-même, ceci signifie qu'il y a un trou. Dans ce cas, un des voisins est sélectionné pour expédier le paquet en tenant compte du coût estimé de ce voisin. A la suite de ce choix, les coûts estimés doivent être mis à jour.

La seconde phase expédie les paquets à l'intérieur de la région. Si le paquet a atteint la région, il peut être répandu dans cette région soit par une diffusion géographique récursive, soit par une inondation limitée. L'inondation restreinte est une bonne solution lorsque les capteurs ne sont pas déployés en masse. Dans les réseaux à haute densité, l'inondation géographique récursive est plus efficace en matière d'énergie que l'inondation limitée. Dans ce cas, la région est divisée en quatre régions secondaires et quatre copies du paquet sont créées. Ce processus de dédoublement de l'expédition continue jusqu'à ce que toutes les régions soient desservies.

3.2.15 SPIN (Sensor Protocols for Information via Negotiation)

SPIN [HEI 99] est un protocole présentant une idée originale dans les WSNs, il suppose que tous les nœuds dans le réseau sont potentiellement des stations de base. Pour manipuler les quantités importantes de données qui seront transférées sur le réseau, SPIN emploie des métadonnées (Descripteurs de données). Par exemple, un paquet de métadonnées pour un capteur photographique est le suivant $[x, y, \phi, t]$ avec x, y les coordonnées des nœuds, ϕ l'orientation de l'objectif et t l'intervalle de temps entre deux prises. SPIN tente de réduire le trafic du réseau en gardant que les traces des nœuds qui sont intéressés sur certains événements en utilisant des paquets *ADV/REQ* (*ADV* utilisé par un nœud pour informer ses voisins de ce qu'il offre comme données. *REQ* est utilisé pour demander à un voisin de lui transmettre certaines données qu'il possède), alors qu'en même temps c'est un protocole puissance avertie et adapte sa stratégie de routage basée sur l'énergie restante dans chaque nœud du réseau. La figure suivante illustre le déploiement de ce protocole :

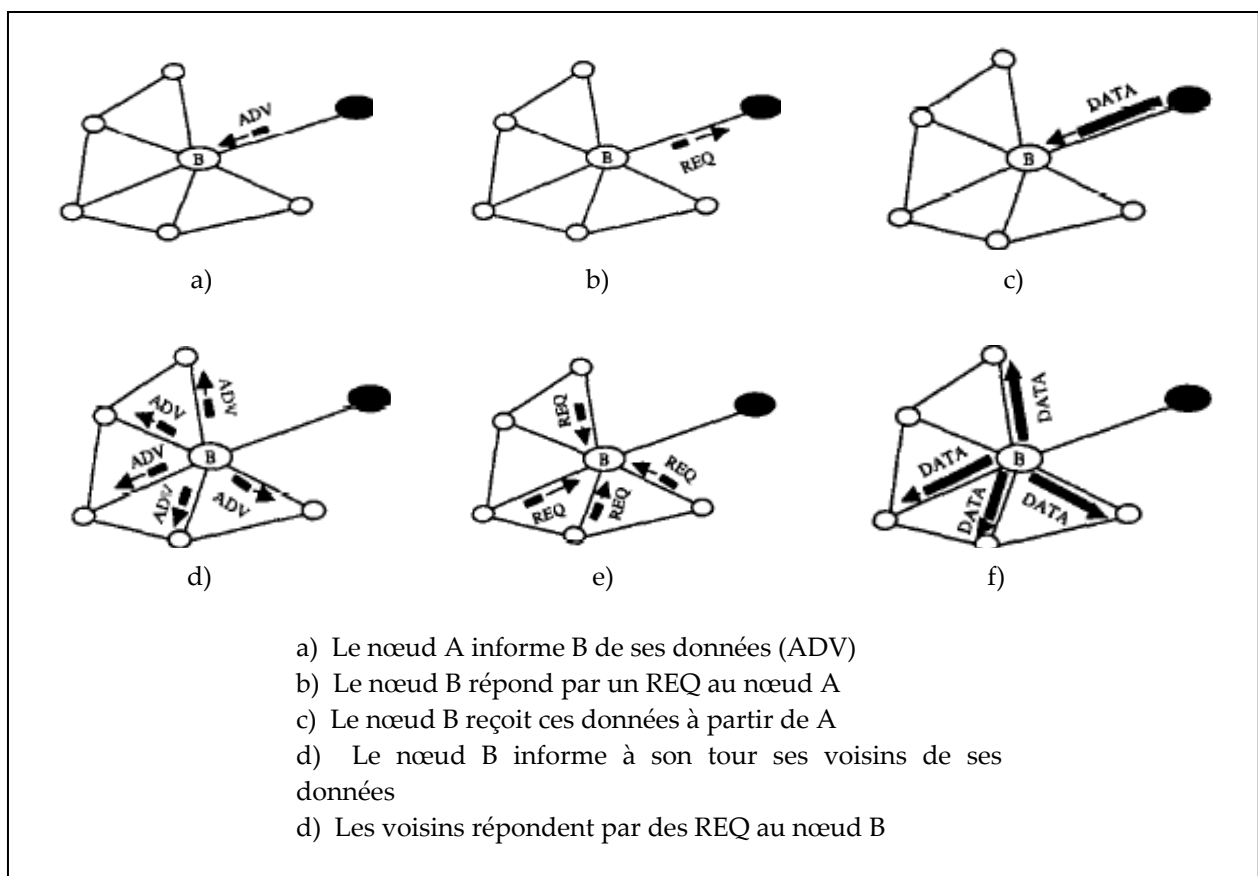


Fig. 3.5 Protocole SPIN [HEI 99].

3.2.16 SAR (Sequential Assignment Routing)

SAR [SOH 00] est un protocole qui inclut la notion de QoS dans ses décisions de routage. C'est une approche par trajets multiples pilotés par une table de routage qui cherche à assurer l'efficacité énergétique et la tolérance aux fautes. Le protocole SAR élabore un arbre de liaisons de voisin à voisin (donc sur un seul saut) de chaque source jusqu'au destinataire en intégrant une métrique de QoS et de réserve énergétique sur chaque chemin et un niveau de priorité pour chaque paquet en circulation (La priorité peut par exemple être fixée selon l'importance des données, la fraîcheur des données...). En créant des arbres, des routes multiples sont ainsi formées entre les capteurs et les destinataires.

Une de ces routes est choisie selon les critères ressources énergétiques et la QoS. En cas de rupture de route, le rétablissement est possible en proposant automatiquement une restauration à partir de la table de routage entre les nœuds ascendants et descendants de chaque route. Bien que ce mécanisme assure une tolérance aux fautes et le rétablissement facile de route, le protocole souffre d'une surcharge pour la mise à jour des tables et des états de chaque nœud capteur, particulièrement quand le nombre de nœuds est élevé.

3.2.17 SPEED

SPEED [HE 03] est un protocole basé sur la qualité de service. Il nécessite que chaque nœud maintienne des informations sur ses voisins et son emplacement géographique pour trouver les chemins. De plus, il essaye d'assurer une certaine rapidité à chaque paquet dans le réseau pour que l'application puisse estimer les délais de bout en bout pour ces paquets en divisant la distance entre la station de base par la vitesse du paquet. Il tente ainsi également d'éviter l'encombrement qui peut se produire dans le réseau.

Le module de routage dans SPEED est appelé SNGF (*Stateless Non-deterministic Geographic Forwarding*), et il travaille avec quatre modules comme le montre la figure suivante :

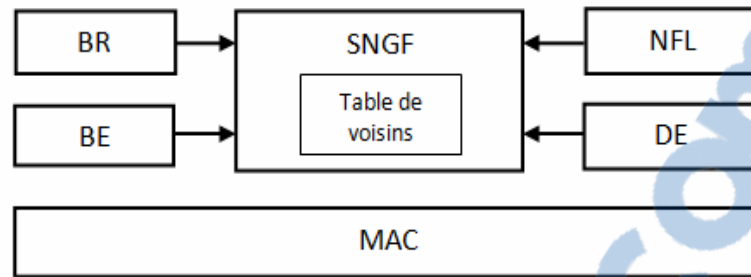


Fig. 3.6 Protocole SPEED [HE 03].

1. BE (*Beacon Exchange*) : collecte les informations sur les nœuds et leurs emplacements et diffuse une information concernant sa position dans le réseau à ses voisins.
2. BR (*Backpressure Rerouting*) : utilisé pour prévenir les vides (quand un nœud n'arrive pas à trouver le nœud du prochain saut) et éliminer l'encombrement dans le réseau en évitant les nœuds trop chargés.
3. NFL (*Neighborhood Feedback Loop*) : est responsable du calcul du taux de relais des voisins. Cette information est nécessaire pour le module BR pour accomplir sa tâche.
4. DE (*Delay Estimation*) : pour chaque nœud on calcule le temps écoulé avant de recevoir un ACK. Ce temps aide SNGF à sélectionner la vitesse nécessaire à ce nœud.

Dans l'architecture de SPEED, aucune métrique d'énergie n'est prise en considération. De plus, aucune comparaison n'est faite avec les autres protocoles qui prennent en charge la consommation d'énergie.

3.3 Classification des principaux protocoles de routage pour WSNs

Dans cette section, nous allons présenter une classification des protocoles de routage vus précédemment en nous basant sur les approches de routage étudiées dans le chapitre précédent, ainsi que sur les travaux [AKY 02] et [ALK 05]. Le tableau 3.1 illustre cette classification. :

	Etablissement du chemin	Structure du réseau	Initiateur	Multi-chemins	Basé QoS	Basé requête	Basé négociation
DD	Réactif	Plat	Destination	Oui	Non	Oui	Oui
RR	Réactif	Plat	Destination	Non	Non	Oui	Non
F&G	Réactif	Plat	Destination	Non	Non	Non	Non
MCFA	Proactif	Plat	Destination	Non	Oui	Non	Non
GBR	Réactif	Plat	Destination	Non	Non	Oui	Non
EAR	Proactif	Plat	Destination	Non	Non	Oui	Non
Chang	Réactif	Plat	Source	Oui	Oui	Non	Non
MECN	Proactif	Plat	Source	Non	Non	Non	Non
RPRW	Proactif	Plat	Source	Oui	Non	Non	Non
LEACH	Proactif	Hiérarchique	Source	Non	Non	Non	Non
TEEN	Proactif	Hiérarchique	Destination	Non	Non	Non	Non
Younis	Proactif	Hiérarchique	Destination	Non	Oui	Non	Non
HEAR	Proactif	Hiérarchique	Destination	Non	Oui	Non	Non
GEAR	Réactif	Plat	Source	Non	Non	Non	Non
SPIN	Réactif	Plat	Source	Oui	Non	Oui	Oui
SAR	Proactif	Plat	Source	Oui	Oui	Non	Oui
SPEED	Proactif	Plat	Destination	Non	Oui	Oui	Non

Tab. 3.1 Classification des principaux protocoles de routage pour les WSNs.

Notons qu'un protocole peut appartenir à plusieurs catégories de routage. A titre d'exemple le protocole EAR est proactif, fonctionne sur une structure plate et initié par la source.

3.4 Conclusion

Ce chapitre représente une brève étude de certains des principaux protocoles de routage dans le cas des réseaux de capteurs. Cette étude nous a permis de mettre en évidence leur fonctionnement, les points forts et les limites de ces derniers. L'objectif commun de la majorité de ces protocoles est de conserver le plus longtemps possible l'énergie au niveau de chacun des nœuds constituant le réseau. Une synthèse sur la

classification de ces protocoles a également été présentée. Elle se base sur les approches de classification des protocoles de routage vues dans le chapitre précédent

Bien que plusieurs de ces stratégies paraissent prometteuses, il existe toujours certains défis qui persistent et nécessitent d'être pris en considération par les protocoles de routage dans les réseaux de capteurs, d'où notre intérêt à développer un nouveau protocole de routage efficace en termes de consommation d'énergie.

Chapitre 4

Elaboration du protocole

4.1 Introduction

Dans ce chapitre nous présentons un nouveau modèle de protocole de routage pour les WSNs ECRP (*Energy Conserving Routing Protocol*). Son objectif est d'augmenter la durée de vie du WSN. Pour ce faire, notre protocole prend en considération les facteurs qui influent sur la consommation de l'énergie lors de l'acheminement des données et assure l'équilibrage de l'énergie consommée par chaque nœud constituant le WSN.

ECRP est un protocole de routage réactif adapté pour des réseaux de capteurs sans fil constitués d'un ensemble de nœuds et une ou plusieurs stations de base, chaque nœud est capable de communiquer avec un sous ensemble de nœuds qui se situent dans sa portée de son signal et qui se trouvent dans la direction de la destination.

Dans le modèle sans fil, une transmission est omnidirectionnelle : chaque message envoyé par N_i est en fait diffusé à tous les nœuds N_j se trouvant dans la portée de N_i . Nous supposons que les communications sont bidirectionnelles. Les nœuds capteurs doivent interagir avec l'environnement pour en extraire des données. Ils doivent ensuite les faire parvenir à la station de base qui procédera éventuellement aux traitements des données reçues par l'intervention de l'utilisateur. Le problème majeur réside dans le fait qu'un nœud capteur ne dispose que d'une quantité d'énergie limitée. Cette quantité d'énergie doit lui permettre d'effectuer la capture des données et d'acheminer les messages au destinataire. Pour ce faire, il nécessite l'aide des autres nœuds et doit donc lui-même aider les autres à faire parvenir leurs données à la station de base.

4.2 Choix d'élaboration

L'élaboration d'un protocole efficace en termes de consommation de l'énergie est discutée dans ce chapitre. Les choix sur lesquels on s'est basés pour élaborer notre modèle de protocole sont : la scalabilité, l'efficacité énergétique et la simplicité. Les sous sections suivantes décrivent comment chacun de ces buts est réalisé.

4.2.1 Scalabilité

Un réseau de capteur peut se composer de quelques nœuds à plusieurs milliers. Par conséquent, un protocole de cheminement dédié aux WSNs doit être scalable. Un facteur important qui influence la scalabilité dans un protocole de routage est la structure du réseau. Dans notre cas, nous avons opté pour une structure plate du réseau étant donné qu'elle est mieux adaptée que la structure hiérarchique (section 2.6.2). L'utilisation d'une structure plate de réseau implique que chaque nœud dans le réseau pourra participer également à la tâche de cheminement.

4.2.2 Efficacité énergétique

Ce but à pour objectif de prolonger la durée de vie du réseau dans l'ensemble, en prolongeant la vie de chacun des nœuds. Pour atteindre ce but, il faut faire en sorte que la consommation de l'énergie au niveau de tous les nœuds dans le réseau doivent être les mêmes. Si quelques nœuds épuisent leurs sources d'énergie plus tôt que d'autres nœuds, le réseau pourrait se partitionner. Comme discuté dans la section 1.6.1, la communication et le calcul sont les facteurs qui consomment la majeure partie de l'énergie d'un nœud. Par conséquent, pour s'assurer que les nœuds survivent aussi longtemps que possible, notre protocole met en application les stratégies de conception suivantes :

1. Le protocole est initié par la source, ceci élimine le besoin de la station de base d'inonder un intérêt pour des données à travers le réseau. Par conséquent le nombre de messages qui sont transmis par différents nœuds est réduit ainsi que le nombre de collisions.
2. Le protocole emploie le reportage de donnée entraîné par le temps et par les événements. Les nœuds génèrent puis transmettent des données périodiquement depuis la cible et lorsque un changement important du phénomène observé se fait senti.

3. Les données sont routées le long d'un seul chemin qui est établi dynamiquement. A chaque fois qu'un nœud veut transmettre, il sélectionne un voisin pour lui envoyer les données. Le voisin est choisi en se basant sur trois métriques : La première métrique est l'énergie résiduelle du voisin, les voisins qui ont des résiduels d'énergie plus grand sont privilégiés pour être choisis. La deuxième métrique est le coût du lien en termes d'énergie entre l'émetteur et le récepteur, ceci permet à l'émetteur de réduire sa consommation de l'énergie. Enfin la dernière métrique, est le nombre de sauts minimum séparant ce voisin de la station de base, cette dernière diminue le nombre d'émissions et réceptions ce qui diminue à son tour la consommation de l'énergie.

4.2.3 Simplicité

Les méthodes de calcul utilisées par notre protocole lors de l'opération du routage des paquets de données ne requièrent pas de règles, d'expressions ou de formules complexes à évaluer ou à exécuter. La simplicité de calcul permet aussi de minimiser la consommation d'énergie au niveau du traitement des données.

4.3 Fonctionnement du protocole

Le protocole fonctionne suivant quatre phases : La première phase est celle de l'installation du réseau, durant cette phase une diffusion vers les nœuds capteurs du WSN est lancée par la station de base afin de recenser toutes les routes possibles menant vers la destination (Construction des tables de routage). La deuxième phase consiste à communiquer les informations senties à la station de base. Pendant la troisième phase, les nœuds mettent à jour leurs niveaux d'énergie et estiment ceux de leurs voisins, les nœuds possédant des niveaux faibles sont écartés lors de l'acheminement des données. Enfin la dernière phase, celle-ci consiste à maintenir les routes, en exécutant périodiquement une diffusion dirigée vers les capteurs par la station de base.

4.3.1 Phase d'installation

1. La station de base « SB » initie la connexion par une diffusion d'un paquet d'installation dirigée vers les nœuds capteur à travers le réseau. Ce paquet contient un champ appelé « Nombre de sauts NS » qui est mis à zéro. Le tableau 4.1 montre le format d'un paquet d'installation.

$$NS_{SB} = 0$$

Où :

NS_{SB} est le nombre de sauts séparant la station de base à elle-même.

2. Chaque nœud intermédiaire N_i recevant le paquet d'installation vérifie si son émetteur N_j se situe entre lui et la station de base, cela est réalisé en évaluant l'expression suivante :

$$d(N_i, N_j) \leq R_i$$

$$d(N_i, SB)^2 \leq d(N_i, N_j)^2 + d(N_j, SB)^2$$

Où :

$d(N_i, N_j)$ est la distance entre les nœuds N_i et N_j ;

R_i est la portée du nœud N_i ;

$d(N_i, SB)$ est la distance entre le nœud N_i et la station de base.

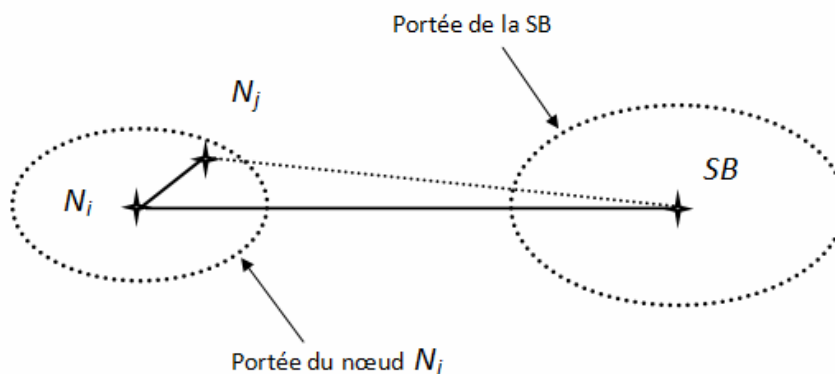


Fig. 4.1 Découverte de voisins

3. Si le nœud N_j vérifie le test alors, une entrée dans la table des voisins (Voir le tableau 4.2) du nœud N_i est ajoutée. Dans cette dernière, il met l'identifiant du nœud N_j , son nombre de saut minimum qui le sépare de la station de base, son énergie résiduelle et son taux de consommation de l'énergie. Le nœud N_i à son tour va diffuser le même message, mais après avoir remplacé les informations du nœud N_j par les siennes.
4. La formule suivante montre comment chaque nœud calcule le nombre de sauts le séparant de la station de base :

$$NS_i = \text{Min} (NS_j) + 1, \quad j=1\dots n$$

Où :

NS_i est le nombre de sauts du nœud N_i ;

NS_j est le nombre de sauts des voisins N_j ;

$\text{Min} (NS_j)$ est le minimum des nombres de sauts des voisins N_j .

Les tableaux suivants montrent le format du paquet d'installation et celui de la table de routage respectivement :

Champs	Taille (Bits)
Identifiant de l'émetteur	16
Numéro de séquence	8
Nombre de sauts	8
Energie résiduelle	16
Taux de consommation	16

Tab.4.1 Format du paquet d'installation.

- L'identifiant de l'émetteur est sur 16 bits permettant 2^{16} (65536) adresses uniques. On assume que chaque nœud dans le réseau possède un identifiant unique.

- Le numéro de séquence est sur 16 bits. Il sert à identifier une nouvelle diffusion, le numéro de séquence est incrémenté à chaque diffusion d'un nouveau paquet d'installation par la station de base. Les nœuds sauvegardent le numéro de séquence localement et retransmettent les messages seulement si leur numéro de séquence est supérieur ou égal à la valeur sauvegardée localement. Le numéro de séquence permet de ne pas envoyer aux nœuds d'anciens messages d'installations.
- Le nombre de sauts exprime le nombre minimum de nœuds à traverser depuis le nœud qui a transmis le paquet d'installation pour atteindre la station de base.
- Energie résiduelle indique le reste de l'énergie du nœud qui a transmis le paquet d'installation.
- Taux consommation est l'énergie consommée par ce nœud durant une unité de temps.

Identifiant	Nombre de sauts	Energie Résiduelle	Taux de consommation	Etat	Coût du lien

Tab. 4.2 Format de la table de voisins.

- Le premier champ indique les identifiants des nœuds voisins.
- Nombre de sauts exprime le nombre minimum de nœuds à traverser depuis ce voisin pour atteindre la station de base.
- Energie résiduelle, c'est l'estimation du niveau d'énergie résiduelle du voisin.
- Taux de consommation, c'est la quantité d'énergie consommée par le voisin pendant une unité de temps. Celui-ci nous permet d'estimer l'énergie résiduelle du voisin sans que ce dernier nous envoie des messages périodiquement pour nous informer de son niveau d'énergie. Le but est de réduire le nombre de messages circulant dans le réseau, ainsi l'énergie consommée.
- Le champ état indique si le nœud voisin est en mesure de participer à la fonction de cheminement.

4.3.2 Phase de transmission des données

Le rôle principal de cette phase est de transmettre les données collectées par les capteurs vers la station de base. Elle consiste à choisir le prochain nœud qui va recevoir ces données pour les transmettre à son tour vers d'autres. Ce processus se répète jusqu'à ce que les données atteignent leur destination.

Un nœud peut être pour lui-même, ⁵actif ou ⁶passif par contre pour les autres nœuds, il peut être concerné ou non concerné. La classification « concerné et non concerné » des voisins d'un nœud N_i dépend d'un seuil d'énergie variable α_i propre à ce nœud. Les voisins qui possèdent des énergies résiduelles supérieures (resp. inférieures) à α_i sont des voisins concernés (resp. non concernés).

$$\alpha_i = \frac{1}{2} \text{MAX}(ER_j) \quad j=1 \dots n$$

Où :

ER_j est l'énergie résiduelle du voisin N_j .

Un nœud concerné : est un nœud passif prêt à transmettre les données. Il possède une quantité d'énergie considérée par rapport aux autres, et dans le cas où il est choisi comme le prochain nœud pour transmettre les données, il sera un nœud actif.

Un nœud non concerné : est un nœud passif qui n'a pas le droit d'être choisi pour router les données à cause de son niveau d'énergie faible.

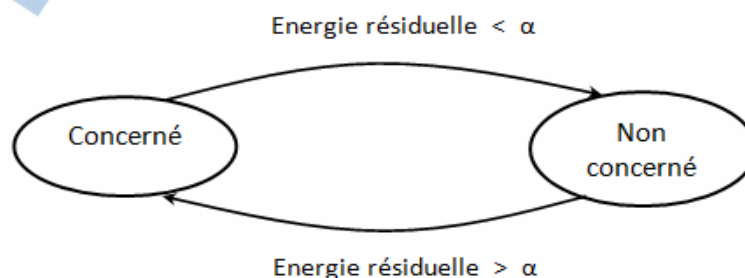


Fig. 4.2 Les états possible d'un voisin.

⁵ Actif : Le nœud en cours de communication de données

⁶ Passif : Le nœud en attente d'un évènement (émission ou réception)

Après avoir identifié les voisins N_j capables de participer à l'opération de routage par le nœud N_i , un choix se fait sur ces derniers pour élire le nœud qui va enfin servir de relais comme suit :

$$MAX ((ER_j / NS_j) \times C_{ij}) \quad j=1 \dots n$$

Où :

ER_j est l'énergie résiduelle du voisin N_j ;

NS_j est le nombre de saut minimum séparant le nœud N_j de la station de base ;

C_{ij} est le coût en termes d'énergie (Emission et Réception) sur le lien ij .

Une fois le voisin N_j ait été sélectionné, le nœud N_i insère dans le paquet de données son estimation d'énergie pour ce voisin avant de le lui transmettre.

Avant de lancer le processus d'élection du voisin N_j qui va servir de relais pour transmettre les données, le nœud N_i vérifie l'estimation de son propre niveau d'énergie faite par le nœud N_k depuis lequel il a reçu le message de données. Dans le cas où cette estimation est mauvaise (dépasse un certain seuil), un message de contrôle (Voir le tableau 4.4) est envoyé par le nœud N_i au nœud N_k afin de corriger son estimation :

$$|ER_i - ER_{i \text{ par } k}| \leq ST$$

Où :

ER_i est l'énergie résiduelle réelle du nœud N_i ;

$ER_{i \text{ par } k}$ est l'énergie résiduelle estimée du nœud N_i par le nœud N_k ;

ST est le seuil qui ne doit pas être dépassé.

Notons que dans le message de données, une information concernant l'estimation du niveau d'énergie du voisin élu est incluse.

Les tableaux suivants nous montrent le format du paquet de données et celui du paquet de contrôle respectivement :

Champs	Taille (Bits)
Identifiant émetteur	16
Identifiant destinataire	16
Identifiant générateur	16
Niveau d'énergie	16
Données	512

Tab. 4.3 Format d'un paquet de données

- L'identifiant du générateur est utilisée pour informer la station de base, quel est le nœud qui a généré le message, car l'adresse de l'émetteur change à chaque saut. Il est supposé que la station de base connaît la position de chaque nœud dans le réseau. Si la station de base ne reconnaît pas le générateur du message et sa localisation les données seront inutiles.
- Le niveau d'énergie représente l'estimation de l'énergie résiduelle du voisin élu pour servir de relai.
- Le champ de données est utilisé pour insérer l'information capturée par la source. Sa taille est fixée à 64 octets.

Champs	Taille (Bits)
Identifiant émetteur	16
Identifiant destinataire	16
Niveau d'énergie	16
Taux de consommation	16

Tab. 4.4 Format d'un paquet de contrôle.

Les champs niveau d'énergie et taux de consommation sont utilisés par le voisin (Emetteur du paquet de contrôle) pour corriger l'estimation faite par le destinataire (Emetteur du paquet de données).

4.3.3 Phase d'estimation des niveaux d'énergie des voisins

Périodiquement, chaque nœud calcule son énergie résiduelle, son nouveau taux de consommation d'énergie et estime les énergies résiduelles de ses voisins en se basant sur leurs taux de consommation d'énergie :

$$\tau_i = EC_i / T$$

$$ER_{j \text{ par } i} = ER_{j \text{ par } i} - (\tau_j \times t)$$

Où :

τ_i est le taux de consommation d'énergie du nœud N_i ;

EC_i est l'énergie consommée par le nœud N_i ;

T est le temps écoulé ;

$ER_{j \text{ par } i}$ est l'énergie résiduelle estimée du voisin N_j par le nœud N_i ;

t est la période.

L'objectif de cette phase est de réduire considérablement le nombre de messages de contrôles échangés par les voisins pour communiquer leurs états (Niveau d'énergie de chacun) et diminuer de manière significative le nombre de diffusions lancées par la station de base pour la maintenance des routes.

4.3.4 Phase de maintenance

La station de base lance périodiquement une diffusion dirigée vers les nœuds constituant le WSN pour qu'ils puissent mettre à jour les informations concernant leurs voisins : ajouter le nouveau voisin qui a joint le réseau aux tables de voisins, et enlèvent ceux qui ont échoués.

4.4 Algorithmes du protocole

4.4.1 Algorithme 1 (Lors de la réception d'un paquet d'installation)

Soit les nœuds N_i et N_j

Soit NT_i la table des voisins du nœud N_i

Lors de la réception d'un paquet d'installation par N_i depuis le nœud N_j

Si (N_j voisin N_i) Alors

Si (N_j existe dans NT_i) Alors

 Mettre à jour les informations de N_j ;

 Sinon

 Ajouter une entrée pour N_j dans NT_i ;

Fin Si

 Rediffuser le paquet d'installation avec ses propres informations ;

 Sinon

 Rejeter le paquet d'installation ;

Fin Si

Fin lors

4.4.2 Algorithme 2 (Lors de la réception d'un paquet de données)

Soit N_j le $j^{\text{ème}}$ voisin du nœud N_i
 Soit N_i un des voisins du nœud N_k
 Soit NT_i la table des voisins du nœud N_i
 Soit $VC(i)$ l'ensemble des voisins concernés du nœud N_i
 Soit $NC(i)$ le nombre de voisins concernés du nœud N_i
 ER_i est l'énergie résiduelle du nœud N_i
 $ER_{i \text{ par } k}$ est l'énergie résiduelle estimée du nœud N_i par le nœud N_k
 Soit a_i le seuil d'énergie du nœud N_i

Lors de la réception d'un paquet de données par N_i depuis N_k

Si ($|ER_i - ER_{i \text{ par } k}| \geq ST$) Alors
 Envoyer un message de contrôle à N_k ;

FinSi

(1) Si ($NV(i) \neq 0$) Alors
Pour ($j=1$ jusqu'à $NV(i)$)
 Calculer $((ER_j / NS_j) * C_{ij})$ pour chaque voisin N_j dans $VC(i)$;
Fin Pour
 Choisir N_j possédant la plus grande valeur depuis $VC(i)$;
 Insérer dans le paquet de donnée $ER_{j \text{ par } i}$;
 Envoyer le paquet vers N_j ;
 Sinon
 Déterminer un nouvel a_i ;
 Déterminer $VC(i)$ parmi les voisins existants au niveau de $NC(i)$;
 Aller à (1) ;

Fin Si

Fin lors

4.4.3 Algorithme 3 (Lors de la réception d'un paquet de contrôle)

Soit N_i un des voisins du nœud N_k
 Soit NT_k la table de voisins du nœud N_k

Lors de la réception d'un paquet de contrôle par N_k depuis N_i

Mettre à jour les informations du voisin N_i dans NT_k ;

Fin lors

4.5 Conclusion

Ce chapitre est complètement consacré à la manière dont l'élaboration du protocole est réalisée. Nous avons commencé par motiver nos choix de conception puis une illustration sur le fonctionnement de notre modèle de protocole est présentée de façon détaillée, enfin les algorithmes constituant notre protocole sont présentés.

Dans le chapitre qui suit nous allons aborder le côté simulation de notre protocole, ainsi qu'une comparaison de ce dernier avec le protocole EAR sera présenté.

Chapitre 5

Tests et résultats

5.1 Objectifs

Le but de ce chapitre est d'évaluer en premier lieu les performances de notre protocole ECRP du point de vue efficacité énergétique. Dans ce chapitre, nous allons commencer par une brève description de l'environnement de simulation choisi pour évaluer notre protocole à savoir le simulateur OMNeT++ (*Object-oriented Modular discrete event Network simulator Test-bed in C++*) [OM web]. Nous terminons ensuite par une analyse des résultats obtenus en se basant sur certains critères de performance considérés afin de mettre en évidence l'apport de notre protocole ECRP par rapport au protocole EAR.

5.2 Simulateur OMNeT++

L'OMNeT++ est un simulateur modulaire à évènement discret orienté objet dédié aux réseaux informatiques. Ce simulateur peut être utilisé pour :

- Modéliser le trafic dans les réseaux de télécommunication;
- Modéliser les protocoles de communication;
- Modéliser les phénomènes de files d'attente dans les réseaux;
- Evaluer les aspects de performances des systèmes logiciels complexes ;
- Valider les architectures matérielles, etc.

5.2.1 Architecture du simulateur OMNeT++

Le fonctionnement du simulateur OMNeT++ repose entièrement sur l'utilisation de modules qui communiquent entre eux par le biais de messages. Ces modules sont organisés hiérarchiquement. Les modules de base sont appelés les modules simples. Ceux-ci sont regroupés en modules composés. Ces derniers peuvent eux mêmes être regroupés en d'autres modules composés. Le nombre de niveau hiérarchique n'est pas limité.

Les modules simples sont codés en C++ et sont des instances du type de base module. L'architecture est construite de telle sorte que les simples modules sont à la fois les émetteurs et destinataires des messages. Les modules composés se contentent de

relayer les messages aux modules simples de façon transparente. On peut attribuer différents paramètres aux connexions reliant les modules : des délais de propagation, des débits de données, des taux d'erreur, etc.

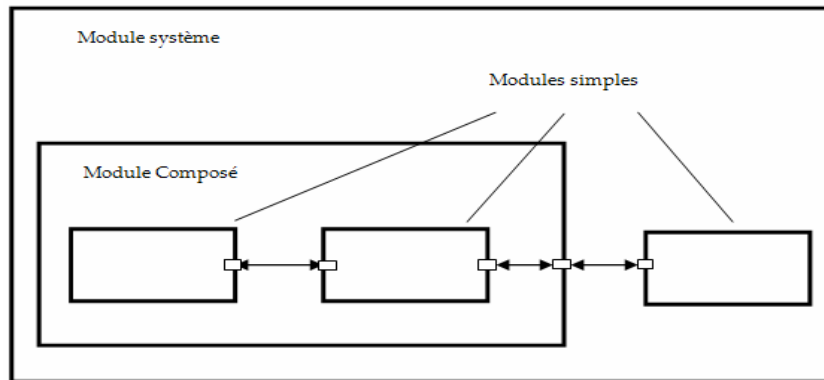


Fig. 5.1 Architecture du simulateur OMNeT++ [OM web].

Les messages sont transmis par le biais de portes (*gates*). Les portes sont les interfaces d'émission et de réception des modules. Une connexion peut relier une porte d'entrée à une porte de sortie. On ne peut créer de connexions que dans un seul niveau de hiérarchie des modules. Il est par exemple impossible de créer une connexion directe entre un module M1 et un sous-module appartenant à un autre module M2 de même niveau de hiérarchie que M1.

Un exemple de ce type d'architecture peut être deux LANs reliés par un routeur. Chaque poste ou élément de chacun des réseaux pourrait être représenté par un simple module, alors que chacun des LANs serait un module composé. Ces deux modules composés seraient reliés par un simple module qui serait un routeur.

5.2.2 Langage NED

La définition de la structure d'une simulation et de sa topologie se fait par le biais du langage NED. Une description NED est en général constituée de la déclaration de modules simples, de définitions de modules composés et d'une description du réseau. Typiquement, la déclaration d'un module simple décrit les interfaces de ce module et ses paramètres. La définition d'un module composé comprend la déclaration des

interfaces externes du module ainsi que la définition de ses sous modules et leurs interconnexions. La définition d'un réseau définit l'ensemble des modules qui le constitue. Cet ensemble à son tour est considéré comme étant le module principal (Le module qui se situe au sommet de la hiérarchie).

Un fichier NED peut contenir les composants suivant arbitrairement et dans n'importe quel ordre :

Importation de modules

Dans cette partie on peut importer des fichiers ".ned" contenant les modules qu'on veut utiliser dans notre de simulation. Elle se fait par le biais de la directive "import " :

```
import  
"nomModule.ned"
```

Définition de canaux

Elle spécifie les caractéristiques de la connexion entre les modules. Trois attributs peuvent être définis : le délai de propagation en secondes, le taux d'erreur (la probabilité qu'un bit transmis est erroné) et la bande passante en bit/second.

```
channel WirelessCh  
    delay 0.0018 // sec  
    error 1e-8  
    datarate 128000 // bit/sec  
endchannel
```

Déclaration de modules simples et complexes

Le type d'un module est identifié par son nom. Par convention, les noms des modules commencent par une lettre majuscule.

Un module simple est défini par la déclaration de ses paramètres et ses ports comme le montre la syntaxe suivante :

```

simple NetworkInterface
  parameters :
      numOfMessages :
  const,
      address : string;

  gates :
      in : fromPort,
          fromHigherLayer;
      out : toPort,
          toHigherLayer;
endsimple

```

Un module complexe est composé d'un ou de plusieurs sous-modules (simples ou complexes). La déclaration d'un module complexe est similaire à celle d'un module simple, elle contient la déclaration de paramètres et de ports, plus deux autres sections qui sont : les sous-modules et les connexions. Elle se fait par la syntaxe suivante :

```

import "NetworkInterface.ned"

module Router
  parameters :
      bufferSize : numeric,
      numOfPorts : const;
  gates :
      in : inputPort[];
      out : outputPort[];
  submodules:
      interface : NetworkInterface [numOfPorts];
  connections:
      interface [0] .toPort --> WirelessCh --> interface [1] .
fromPort;
      for i=0.. numOfPorts-1 do
          interface [i] .toPort --> inputPort[i];
      endfor;
endmodule

```

Déclaration du module système

C'est une instance du module principal qui définit tous les sous-modules de notre simulation. Il peut définir les paramètres globaux du système :


```

network wirelessLAN: WirelessLAN
  parameters :
    numUsers=10,
    httpTraffic=true,

    distanceFromHub=truncnormal(100,60);
endnetwork

```

Définition des paramètres

Les paramètres sont des variables qui appartiennent aux modules. Les valeurs peuvent être assignées aux paramètres directement dans la déclaration du module dans le fichier NED. Cependant, dans certains cas, les modules ont besoin de paramètres plus complexes qu'on ne peut pas définir dans un module simple. Dans ce cas, les paramètres sont introduits par un fichier de configuration externe. C'est pour cette raison que le fichier "omnetpp.ini" est nécessaire. Il est également possible d'utiliser un fichier XML.

5.2.3 Messages dans OMNeT++

La classe "*cMessage*" permet de modéliser des événements, des messages, des paquets, des bits ou bien des signaux traversant le réseau. Elle contient des attributs et des méthodes pour la manipulation de messages. La définition d'un message se fait par la directive suivante :

```

Message MyPacket
{
  int srcAddress;
  int destAddress;
  int hopsCount;
}

```

Si cette définition est écrite dans un fichier "*mypaquet.msg*" le compilateur générera deux fichiers : "*mypaquet_m.h*" et "*mypaquet_m.cc*" qui contiendront la définition de la classe "*MyPacket*" qui est une sous classe de la classe "*cMessage*".

5.2.4 Construction d'un programme de simulation OMNeT++

Un projet de simulation sous le simulateur OMNeT++ consiste en les étapes suivantes :

- La définition des modules intervenant dans la simulation en utilisant le langage NED dans des fichiers "*.ned" ;
- La définition des messages échangés entre les modules dans des fichiers "*.msg" ;
- L'implémentation des modules simples avec un code C++ ;
- Création du fichier « *omnetpp.ini* » contenant les paramètres de configuration du système nécessaire pour l'exécution de ce programme.
- Compilation du programme en utilisant le compilateur C++ ;
- Finalement, l'édition de lien et la génération de l'exécutable de la simulation.

5.3 Tests et résultats

5.3.1 Performances mesurées

Pour évaluer l'efficacité énergétique de notre protocole EERP, nous avons choisi de le comparer au protocole EAR (*Energy Aware Routing*). Ce choix se justifie par le fait que la comparaison d'EAR avec la diffusion dirigée a démontré que le premier permet une diminution globale de 21% de l'énergie consommée et une augmentation de 44% de la durée de vie du réseau.

La comparaison de notre protocole avec EAR portera sur les points suivants :

1. Le taux de défaillance (*TD*): c'est : le nombre de nœuds qui défont dans le temps en raison de l'épuisement de leur énergie ;
2. Le taux de succès ou taux de livraison (*TS*) : il exprime le nombre de paquets de données reçus par la station de base sur l'ensemble des paquets de données générés par les nœuds dans le réseau.

3. La moyenne d'énergie consommée par tous les nœuds constituant le réseau (E_m).

Elle est calculée comme suit :

$$E_m = \frac{\sum_{k=1}^{k=n} (E_{ik} - E_{rk})}{n}$$

Où :

E_{ik} est l'énergie initiale des nœuds ;

E_{rk} est l'énergie résiduelle des nœuds ;

n est le nombre de nœuds capteurs dans le réseau.

4. Le délai moyen (DM) : il exprime le délai moyen d'acheminement des données capturées pour atteindre la station de base. Ce critère permet d'avoir une idée sur la fraîcheur des données capturées.
5. La fonction énergétique (F) : elle mesure le taux de données envoyées avec succès par rapport au nombre de nœuds qui ont cessé de fonctionner à cause de l'épuisement de leurs ressources énergétiques, elle est calculée comme suit :

$$F = TS/TD$$

Où :

TS est le taux de succès ;

TD est le taux de défaillance.

5.3.2 Construction de la topologie

Pour tester notre protocole ECRP, nous avons réalisé une série de simulations. Chaque simulation utilise une seule station de base et un ensemble de nœuds capteurs. La taille du réseau prend les valeurs suivantes : 25, 50, 75, 100, 125, 150, 175 ou 200 nœuds, distribués sur une surface de 120×120 mètres de manière aléatoire (pas de distribution uniforme). La même topologie est utilisée pour le protocole EAR.

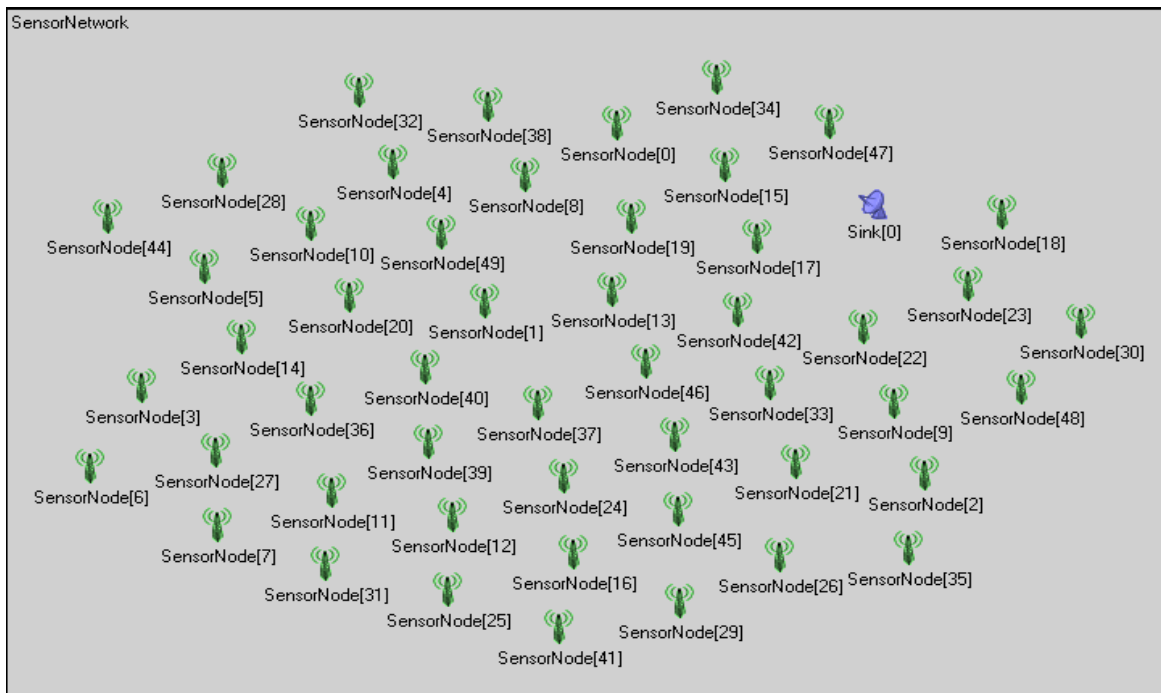


Fig. 5.2 Une topologie du réseau avec 50 nœuds.

5.3.3 Paramètres de simulation

Modèle radio

Le modèle radio proposé par Heinzelman [HE 00] a été utilisé pour calculer l'énergie consommée pendant l'émission et la réception des messages. Selon ce modèle, l'énergie consommée pendant la transmission est donnée par :

$$E_e = (E_{elec} \times pk) + (E_{amp} \times pk \times d^2)$$

Par contre l'énergie consommée pendant la réception est donnée par :

$$E_e = E_{elec} \times pk$$

Où :

E_{elec} est l'énergie consommée par le transceiver électronique;

pk est la taille du message voir les tableaux (Tab. 5.2, Tab. 5.3 et Tab. 5.4) ;

E_{amp} est l'énergie consommée par l'amplificateur ;

d est la distance qui sépare l'émetteur du récepteur.

La portée maximale du signal radio est la même pour tous les nœuds. Celle-ci varie en fonction du nombre de nœuds constituant notre réseau comme le montre le tableau suivant :

Nombre de nœuds	Portée maximale (en mètres)
25	32.5
50	30
75	27.5
100	25
125	22.5
150	20
175	17.5
200	15

Tab. 5.1 Porté du signal radio selon le nombre de nœuds.

Paramètres énergétiques et de temps

Les sources d'énergie des nœuds capteurs sont toutes initialisées à 5 joules. Les valeurs E_{elec} et E_{amp} sont respectivement : 3000 nj/bit et 30 nj/bit et le temps d'exécution de chaque simulation est de 30 secondes.

5.3.4 Discussion des résultats

Taux de défaillance

D'après la figure Fig.5.3, il est possible de constater que le protocole ECRP donne de bons résultats par rapport au protocole EAR en termes de nombre de nœuds qui restent toujours fonctionnels. Dans EAR, le pourcentage des nœuds défaillants est plus

important que celui d'ECRP. Il est possible d'expliquer cela par le fait qu'ECRP gère de manière plus économique la consommation de l'énergie en réduisant considérablement le nombre de messages qui circulent dans le réseau. Cela est rendu possible grâce au fait que la station de base dans ECRP réduit le nombre de diffusions de messages de maintenance qui ont un coût considérable.

La deuxième raison est l'introduction d'une méthode qui permet d'estimer la ressource énergétique des voisins. Cette approche permet de diminuer le nombre de messages de contrôle qui sont utilisés par les voisins pour échanger les informations relatives à leur énergie : dans EAR par exemple, chaque nœud envoie périodiquement un message contenant son niveau d'énergie à ses voisins.

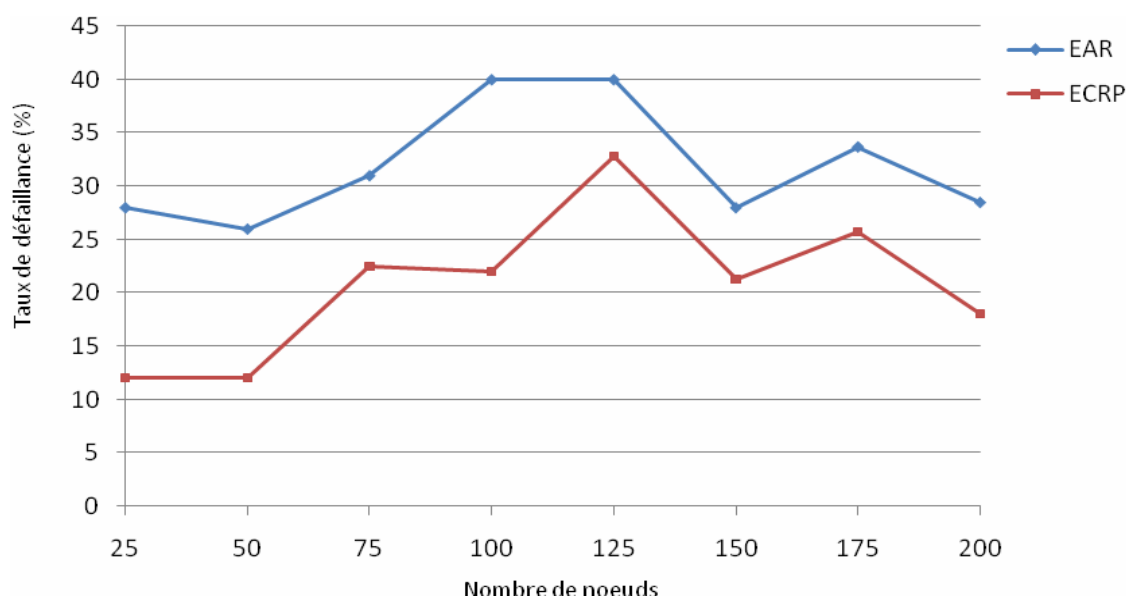


Fig. 5.3 Evolution du taux de défaillance des nœuds.

Taux de succès

La figure Fig.5.4 montre que le taux de livraison de données dans EAR est légèrement meilleur que celui d'ECRP. Ceci est dû au nombre important de paquets de données générés par les nœuds dans ECRP, car le taux de défaillance des nœuds dans EAR est plus important que celui dans notre approche (Nombre de nœuds fonctionnels est plus grand dans ECRP). Par conséquent le rapport entre le nombre de paquets de

données reçus par la station de base et celui généré par l'ensemble des nœuds du réseau est légèrement plus grand dans le cas du protocole EAR. Notons que la quantité de données reçues par la station de base est pratiquement la même pour les deux protocoles.

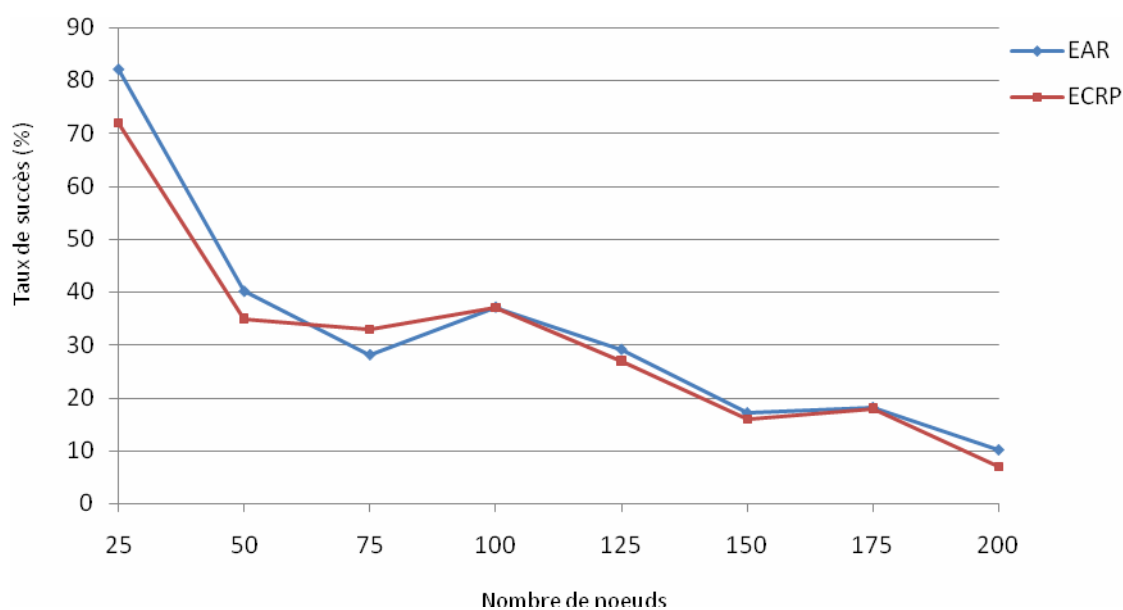


Fig. 5.4 Taux de succès ou de livraison des paquets

Moyenne d'énergie consommée

On peut voir dans la figure Fig.5.5 que dans certain cas (75, 150 et 175 nœuds) la moyenne de l'énergie consommée dans EAR est légèrement plus petite que celle dans ECRP en dépit du fait qu'il est apparu dans ce qui précède que le taux de défaillances d'EAR est plus important (Fig.5.3). Cela s'explique par le fait que le protocole ECRP gère de manière plus efficace la consommation de l'énergie en faisant en sorte que les nœuds qui ont la même distance par rapport à la station de base aient le même taux de consommation d'énergie. Ainsi, une consommation équilibrée est assurée dans le réseau en prenant en considération tous les paramètres qui influent sur la bonne gestion de la ressource énergétique lors du routage des paquets (Energie résiduelle des voisins, coût du lien, nombre de sauts pour atteindre la station de base) alors que dans EAR seul le coût du lien est pris en compte.

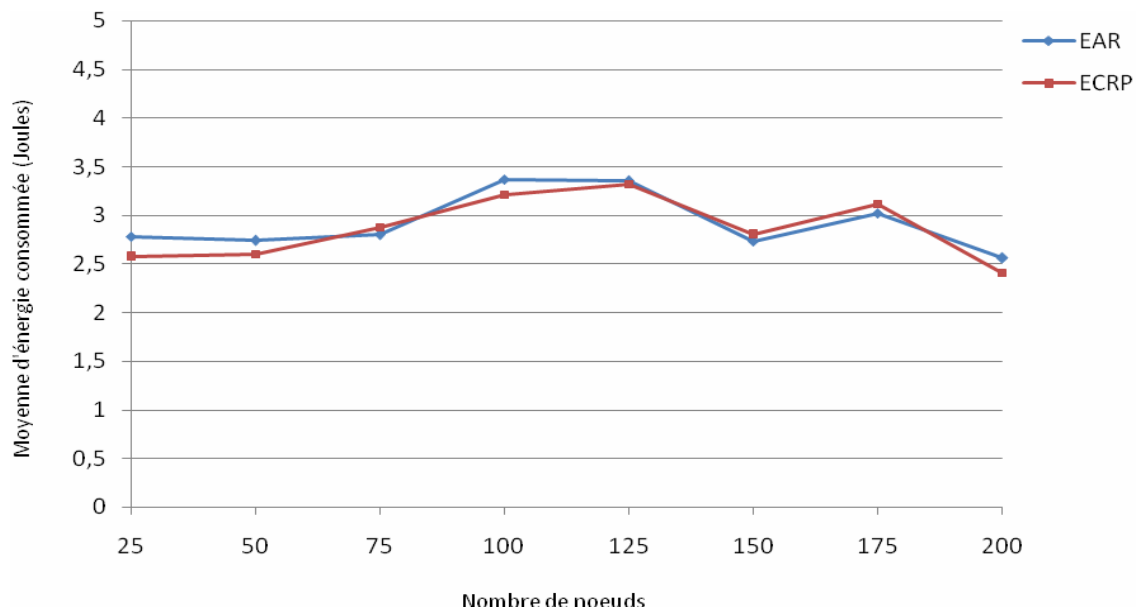


Fig. 5.5 Moyenne d'énergie consommée

Délai Moyen

Dans la figure 5.6, nous pouvons constater que dans le cas d'une faible densité de déploiement des nœuds, les performances en termes de délais d'acheminement sont meilleures dans le protocole EAR. Cependant pour les fortes densités de déploiement l'avantage est pour le protocole ECRP. La raison à cela, est que le protocole ECRP réagit d'une manière réactive à chaque réception d'un paquet de données. Par contre dans le protocole EAR, on connaît à l'avance le prochain nœud qui prendra le relai. Cependant dès que nous avons une forte densité de déploiement, le volume du trafic de contrôle augmente d'une manière très importante dans le cas du protocole EAR, ce qui engendre un retard dans l'acheminement des données.

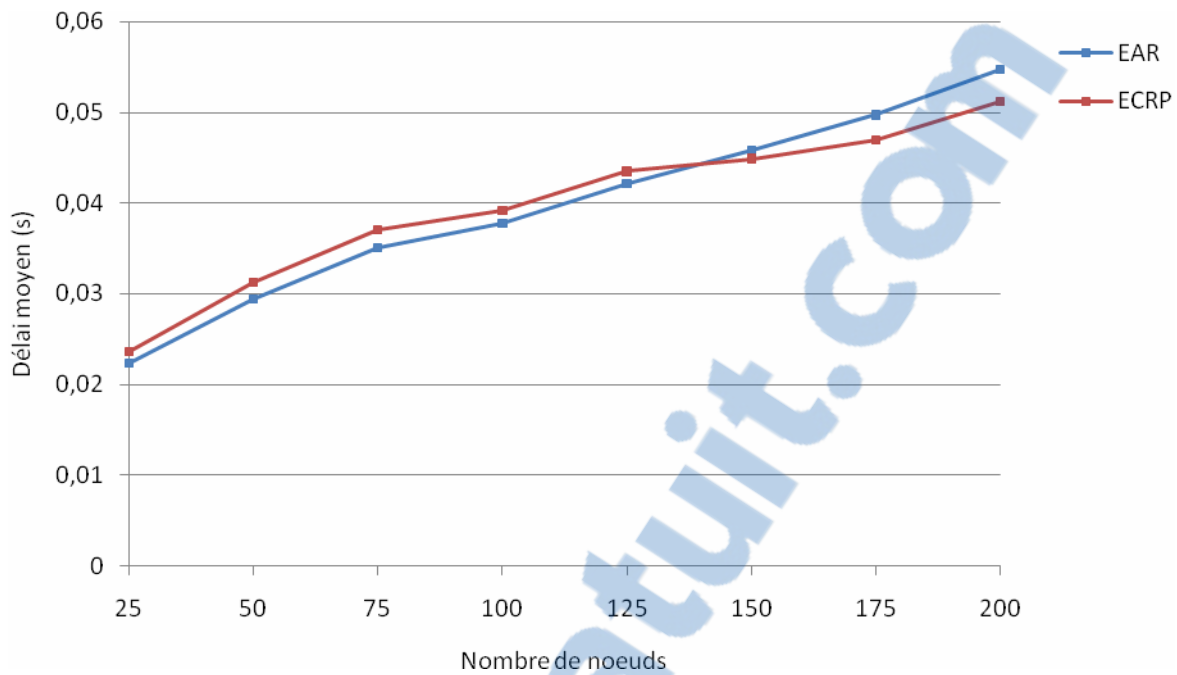


Fig. 5.6 Délai moyen

Fonction énergétique

Il apparaît dans la figure Fig.5.4 qu'EAR est légèrement plus performant en termes de taux de livraison des données. De même, dans la Figure Fig.5.5 certaines simulations présentent une moyenne d'énergie consommée plus petite que celle d'ECRP. Cependant, la figure Fig.5.7 montre que le rapport entre les performances du réseau (Taux de livraison des données) et la dégradation de la connectivité due à la défaillance des nœuds, est en faveur de notre approche. Étant donné que la gestion de l'énergie est l'un des facteurs essentiels à prendre en compte lors de la conception des protocoles de communication dans les réseaux de capteurs, nous pouvons affirmer l'apport de notre protocole en termes de conservation de l'énergie.

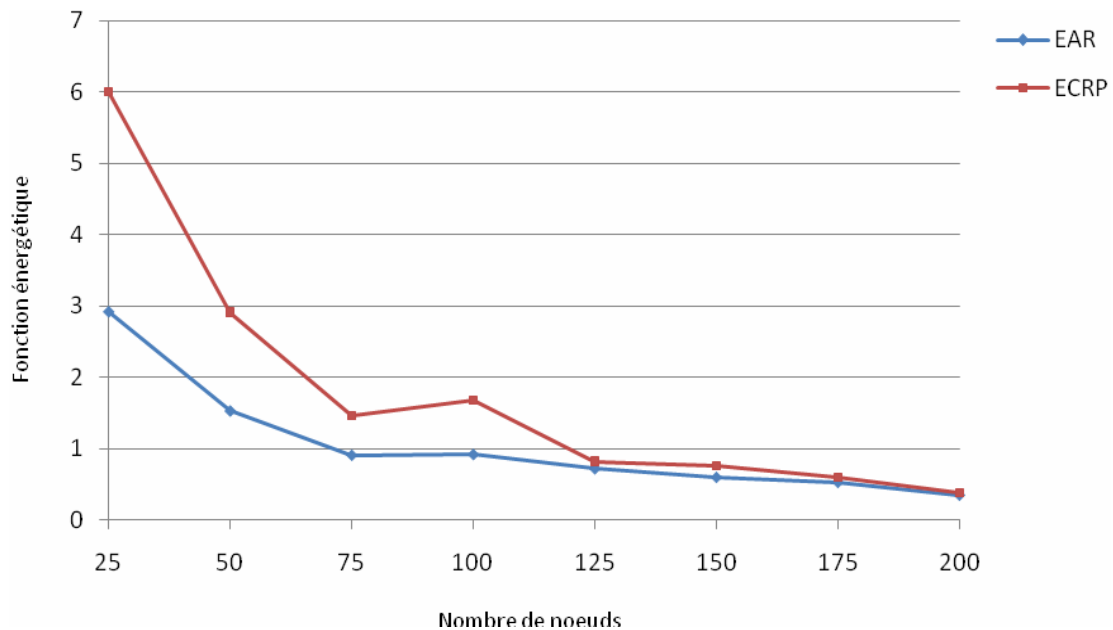


Fig. 5.7 Fonction énergétique

5.4 Conclusion

Notre protocole est conçu de manière à s'adapter aux réseaux ayant une topologie arbitraire, et constitués d'une station de base et d'un ensemble de nœuds capteurs capables de capter l'information et de la transmettre vers la destination en employant le routage multi-sauts. Chaque nœud dans le réseau possède une ressource énergétique limitée, d'où l'importance de prendre en compte les paramètres qui peuvent influencer sur la consommation d'énergie au moment du routage. Notre approche se base sur les facteurs suivants : l'énergie : résiduelle des voisins, le coût du lien et le nombre de sauts pour atteindre la station de base. Elle propose également une méthode permettant l'estimation des niveaux d'énergie des nœuds voisins au lieu de la transmission périodique du niveau exact. Les résultats de la simulation ont montré que notre protocole gère de manière efficace la consommation de l'énergie en permettant un taux de défaillance des nœuds moins important que le protocole EAR auquel il a été comparé, ce qui augmente la durée de fonctionnement du réseau.

Conclusion générale et perspectives

L'objectif de ce mémoire était d'étudier le routage dans les réseaux de capteurs, afin de mettre en lumière les caractéristiques et les insuffisances des protocoles existants et d'être à même d'améliorer ces derniers, voire d'en proposer de nouveaux plus efficaces.

Le problème auquel nous nous sommes intéressés dans ce travail concerne le routage dans les réseaux de capteurs. Il est indispensable de gérer le niveau d'énergie des nœuds de manière économique afin de prolonger la durée de vie du réseau. Dans ce travail, nous avons donc fait, dans un premier temps, une étude relative aux réseaux de capteurs en insistant sur la problématique du routage, afin de proposer, dans un second temps, un protocole permettant une gestion plus efficace en termes énergétiques.

Lors de l'opération de routage, notre approche prend en compte les contraintes liées au nœud (coût du lien). Les contraintes concernant les voisins et le réseau global (l'énergie résiduelle des voisins et le nombre de sauts pour atteindre la station de base respectivement) sont également prises en compte pour sélectionner la route induisant le plus petit coût. Les nœuds utilisent également une méthode leur permettant d'estimer le niveau d'énergie de leurs voisins dans le but de diminuer de façon significative le nombre de messages de contrôle circulant dans le réseau (échange de messages relatifs aux états des nœuds).

La simulation de notre protocole montre que celui-ci donne des résultats très intéressants notamment le prolongement de la durée de vie des nœuds (le taux de défaillance des nœuds est moins important) ce qui implique le prolongement de la durée de fonctionnement du réseau.

Comme très souvent, le travail que nous avons effectué ouvre de nouvelles perspectives, permettant de le compléter de différentes manières.

L'une des principales voies que nous voyons pour améliorer ce travail concerne l'obtention de résultats expérimentaux ne s'appuyant pas sur des simulations. En effet, aussi bon soit-il, un simulateur dépend d'un grand nombre de choix de paramètres et de modèles qu'il ne reflète jamais totalement la réalité d'un réseau sans fil.

Le protocole que nous avons élaboré s'applique sur les réseaux ayant une structure plate (appartenant à la famille de protocoles horizontaux). Ce dernier peut être adapté pour supporter les structures hiérarchiques. Le but est de pouvoir appliquer facilement l'agrégation de données, ce qui réduirait le nombre de messages circulant dans le réseau (élimination des données redondantes) et ainsi obtenir un gain en matière d'énergie.

Enfin, une étude du trafic dans les réseaux de capteurs reste à faire prochainement afin de mieux comprendre sa nature, l'objectif étant d'affiner les estimations des niveaux d'énergie des voisins.

Références

1. [ABO 05] M.Aboelaze, F.Aloul, «*Current and future trends in Sensor Networks : A survey* » IEEE, 2005.
2. [AKK 03] Kemal Akkaya and Mohamed Younis « *A Survey on Routing Protocols for Wireless Sensor Networks* » Department of Computer Science and Electrical Engineering University of Maryland, Baltimore County Baltimore, MD 21250, 2003.
3. [AKY 02] Ian F. Akyildiz, Weilian Su, Yogesh Sankarasubramaniam, and Erdal Cayirci « *A Survey on Sensor Networks* ». Georgia Institute of Technology. IEEE Communications Magazine, August 2002.
4. [ALK 05] Jamal N. Al-Karaki, Ahmed E. Kamal « *Routing Techniques in Wireless Sensor Network : A Survey* » Dept. of Electrical and Computer Engineering Iowa State University, Ames, Iowa 50011, 2005.
5. [AVA 03] Avancha, S. et al. « *Wireless Sensor Networks* ». Kluwer Academic/Springer Verlag Publishers, 2003.
6. [BOU 03] Br. Bougard, Denis C. Daly, A.Chandrakasan, , W. Dehaene, «*Energy Efficiency of the IEEE 802.15.4 Standard in Dense Wireless Microsensor Networks: Modeling and Improvement Perspectives*», 2003.
7. [BRA 02] D. Braginsky and D. Estrin « *Rumor Routing Algorithm for Sensor Network* » in Proceedings of the first Workshop on Sensor Networks and Application, 28 Sept. 2002 Atlanta, USA, 2002, pp. 22-31.
8. [BRA 06] Mohamed BRAHMA. « *Étude de la QoS dans les Réseaux Ad hoc : Intégration du Concept de l'Ingénierie du Trafic* ». Thèse doctorat de l'Université de Haute Alsace des sciences et techniques, décembre, 2006.
9. [BRI 04] Britton, M. et al. The SECOAS project. « *Development of a self organizing,wireless sensor network for environmental monitoring* ». Proc. workshop SANPA, 2004.
- 10.[BUR 04] Burrell, J. et al. « *Vineyard Computing Sensor Networks in Agricultural Production*» IEEE Pervasive Computing, Vol.3, No.1, pp. 38-45, 2004.
- 11.[CHA 00] J. H. Chang and L. Tassiulas. « *Energy conserving routing in wireless ad-hoc networks*». Proceedings. IEEE, INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies, Vol. 1, 2000.
- 12.[CHA 04] Chaudhary, S. et al. « *Architecture of Sensor based Agricultural Information System for Effective Planning of Farm Activities.* » Proc. SCC, 2004.

- 13.[CHE 04] Guillaume Chelius « *Architectures et communications dans les réseaux spontanés sans fil* » Thèse de doctorat, INSA, Lyon, Avril 2004
- 14.[CHE 04] Cheng, X. et al. « *TPS: A Time-Based Positioning Scheme for Outdoor Wireless Sensor Networks* » Proc. Infocom, 2004.
- 15.[ELH 02] A. El-Hoidy, « *Spatial TDMA and CSMA with preamble sampling for low power Ad Hoc wireless sensor net works* », in proceedings of the IEEE Symposium on Computers and Communications (ISCC), 1-4 july 2002, Taormina/Giardini Naxos,Italy,2002,pp.685_692.
- 16.[DAI 04] Dai, H. et al. « *ELF: an efficient log-structured flash file system for micro sensor nodes*» Proc. SenSys, 2004.
- 17.[DAS 03] S. Das, C. Perkins, E. Belding-Royer. RFC 3561, Ad hoc On-Demand Distance Vector (AODV) Routing [en ligne], 2003. Memo. The Internet Society. Disponible sur : <ftp://ftp.rfc-editor.org/innotes/rfc3561.txt>.
- 18.[DAS 00] Samir R. Das, Robert Castañeda and Jiangtao Yan, « *Simulation-base performance evaluation of routing protocols for mobile ad ho networks* » Baltzer Science Publishers BV, 2000.
- 19.[DHO 03] Dominique Dhoutaut. « *Etude du standard IEEE 802.11 dans le cadre des réseaux ad ho de la simulation à l'expérimentation* ». Thèse de doctorat, INRIA ARES, Laboratoire CITI, INSA de Lyon, 2003.
- 20.[GAY 03] Gay, D. et al. «*The nesC language: A holistic approach to networked embedded systems*» Proc. SIGPLAN, 2003.
- 21.[GEI 04] J Geier, « *Wireless networks first step* », Cisco Press, August 2004.
22. [GRE 04] Greenstein, B. et al. « *A sensor network application construction kit (SNACK)* ». Proc. SenSys, 2004.
- 23.[GUP 02] A. Gupta, N.Kumar, «*Term Paper Report on Acquiring information from smart badges in an emergency network* » Advanced Computer Networks, 2002.
- 24.[HAM 07] Hamzi, « *Plateforme basée agents pour l'aide à la conception et la simulation des réseaux de capteurs sans fil* » Mémoire de Magister, INI, Alger, 2007.
- 25.[HE 03] T. He, et al. « *SPEED: A stateless protocol for real-time communication in sensor networks.* » Proc. ICDCS, 2003.
- 26.[HED 88] S. Hedetniemi and A. Liestman « *A Survey of Gossiping and Broadcasting in Communication Networks* » IEEE Network, vol. 18, N° 4 pp. 319-349, 1988.

-
- 27.[HEI 99] W. Heizelman, J. Kulik, and H. Balakrishnan. «*Adaptive protocols for information dissemination in wireless sensor networks*». Proceedings of the 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking, pp. 174-185, 1999.
- 28.[HEI 00] W. Heizelman, A. Chandrakasan and H. Balakrishnan, «*Energy-Efficient Communication Protocol for Wireless Mi-crosensor Networks* » Proceedings of the 33rd Hawaii International Conference on System Sciences (HICSS '00), January 2000.
- 29.[HEM 05] M. Hemel, H. Sharif and P. Raviraj «*HEAR-SN: A New Hierarchical Energy-Aware Routing Protocol for Sensor Networks* » in Proceeding of the 38th Annual Hawaii International Conference on System Sciences (HICSS), January 2005.
- 30.[HIL 00] Hill, J. et al. «*System architecture directions for networked sensors* ». Proc. ACM Int. Conf.on Architectural Support for Programming Languages and Operating Systems, 2000.
- 31.[HU 03] Yih-Chun Hu, David B. Johnson, David A. Maltz. «*The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks (DSR)* » (draft IETF), 2003.
- 32.[ILY 05] M. ilyas , I. mahgoub, «*Handbook of Sensor Networks: Compact Wireless and Wired Sensing Systems* », CRC PRESS, Boca Raton London New York Washington, D.C, 2005.
- 33.[INT 03] K. Intae and R. Poovendran, «*Maximizing static network lifetime of wireless broadcast ad-hoc* », in Proceedings of the IEEE International Conference on Communication, 11-15 May 2003, Anchorage, USA, vol.3, pp. 2256-2261.
- 34.[INT 00] C. Intanagonwiwat, R. Govindan, and D. Estrin. «*Directed Diffusion: A scalable and robust communication paradigm for sensor networks* » in Proceedings of the 6th International Conference in Mobile Computing and Networking (Mobicom), 6-11 Aug. 2000, Boston.
- 35.[JAC 03] Philippe Jacquet, Thomas Clausen. RFC 3626, Optimized Link State Routing Protocol (OLSR), 2003.
- 36.[JI 04] Ji, X. et al. «*Sensor Positioning in Wireless Ad-hoc Sensor Networks with Multidimensional Scaling* », Proc. Infocom, 2004.
- 37.[JIA 04] Q. Jiang and D. Manivannan. «*Routing protocols for sensor networks*». Proceedings of the First IEEE Consumer Communications and Networking Conference, pp. 93-98, June 2004.

- 38.[KAR 04] Karlof, C. et al. «*TinySec : a link layer security architecture for wireless sensor networks*». Proc. SenSys, 2004.
- 39.[KHE 04] L.Khelladi, N.Badache, «*Réseaux de capteurs : état de l'art*», Rapport de recherche Université des sciences et de la technologie houari Boumediene (USTHB), Février 2004.
- 40.[KRI 02] B. Krishnamachari, D. Estrin, S. «*Wicker Modeling Data Centric Routing in Wireless Sensor Networks* »in the *Proceedings of IEEE INFOCOM*, New York, NY, June 2002.
- 41.[LAA 07] Naima Laadjal , Selma Hamouda «*Plateforme pour l'aide à la conception et la simulation des réseaux de capteurs* » Mémoire d'Ingénieur, INI, Alger, 2007.
- 42.[LEU 05] Charl Jaco Leuschner «*The Design of Simple Energy Efficient Routing Protocol to Improve Wireless Sensors Network Lifetime* » Submitted in partial fulfilment of the requirements for the degree Master of Engineering, University of Pretoria, 2005.
- 43.[MAN 01] A. Manjeshwar and D.P. Agrawal, «*TEEN : A protocol for Enhanced Efficiency in Wireless Sensor Networks* » Proceedings of the 1 st Int. Workshop on Parallel and Distributed Computing Issues in Wireless Networks and Mobile Computing, April 2001.
44. [MAR 04] Martinez, K. et al «*Environmental Sensor Networks* » IEEE Computer, Vol. 37, No. 8, pp. 50-56, 2004.
- 45.[MER 05] Amina Meraihi Naimi. «*Délai et Routage dans les réseaux ad hoc 802.11* ». Thèse doctorat de l'Université de Versailles Saint-Quentin-En-Yvelines, 2005.
- 46.[MIL 91] Mills, D.L. «*Internet Time Synchronization: The Network Time Protocol* ». IEEE trans. Communications, 39(10), 1991.
- 47.[MOO04] D.Moore et al. «*Robust distributed network localization with noisy range measurements*». Proc. SenSys, 2004.
- 48.[NIE 03] T. Nieberg, S. Dulman, P. Havinga, Lodewijk v. Hoesel, J. Wu, «*Collaborative Algorithms for Communication in Wireless Sensor Networks*». 2003.
- 49.[POM 04] C. Pomalaza-Rez, «*Wireless Ad Hoc & Sensor Networks* », Université d'Oulu, Finlande, 2004.
- 50.[PRA 05] Prasant Mohapatra and Srikanth V. Krishnamurthy. «*Ad-hoc Networks, Technologies and Protocols*». Book, University of California, Springer Science + Business Media, Inc. Boston, Print © 2005.

-
- 51.[RAM 04] Karthik Ramachandra and Hesham H. Ali. « *Evaluating the Performance of Various Architectures for Wireless Ad Hoc Networks* », IEEE Proceedings of the 37th Hawaii International Conference on System Sciences , 2004.
- 52.[RAB 00] J M. Rabaey et al. « *PicoRadio Supports Ad Hoc Ultra-Low Power Wireless Networking*». IEEE Comp. Mag., 2000.
- 53.[ROD 99] V. Rodoplu and T. H. Meng, « *Minimum Energy Mobile Wireless Networks* » IEEE Journal on Selected Areas in Communications, vol. 17, N° 8, pp. 1333-1344, Aug. 1999.
- 54.[SCH 05] Alexandros Schillings « *A Study on Power-Friendly Routing Protocols for Sensor Networks* » University Of Essex, Department of Electronic Systems Engineering, March 2005.
- 55.[SCH 01] C. Schurgers and M.B. Srivastava « *Energy efficient routing in wireless sensor networks* », in Proceedings of the IEEE Military Communications Conference (MILCOM), 28-31 Oct. 2001, Washington, USA, vol. 1, 2001, pp. 357-361.
- 56.[SCH 01] Loren Schwiebert, Sandeep K. S. Gupta, and Jennifer Weinmann. «*Research challenges in wireless networks of biomedical sensors. In Mobile Computing and Networking*» pages 151. 165, 2001.
- 57.[SER 02] S. Servetto and G. Barrenechea «*Constrained Random Walks on Random Graph Routing Algorithms for Large Scale Wireless Sensor Network*» in Proceedings of the first International Workshop on Wireless Sensor Networks and Applications, 28 Sept. 2002, Atlanta, USA, 2002, pp. 12-21.
58. [SHA 02] R. C. Shah and J. Rabaey «*Energy Aware Routing for Low Ad-Hoc Sensor Networks*» in Proceedings of the IEEE Wireless Communications and Networking Conference, 17-21 Mar. 2002, Orlando, USA, vol. 1, 2002, pp. 350-355.
- 59.[SHA 04] Shang, Y. et al. «*Improved MDS-Based Localization*». Proc. Infocom, 2004.
- 60.[SHU 04] Shum, L. et al. «*Distributed Algorithm Implementation and Interaction in Wireless Sensor Networks*». Proc. workshop SANPA, 2004.
- 61.[SOH 00] K. Sohrabi, et al. «*Protocols for self-organization of a wireless sensor network*». IEEE Personal Communications, Vol.7, No.5, pp.16-27, 2000.
- 62.[SZE 04] Szewczyk, R. et al. «*An analysis of a large scale habitat monitoring application*». Proc. SenSys, pp. 214 - 226, 2004
- 63.[TAN 03] A. Tanenbaum. «*Computer Networks, 4th edition*». Amsterdam Libre university, Pearson Education 2003.

- 64.[VIS 04] Sharat C. Visweswara, «*An Automatic, Adaptive, Ad-hoc Algorithm for Power Conservation in Sensor Networks using Switch-off* ». Master thesis, department of computer science, Raleigh, Juillet 2004.
65. [WOO 02] Wood, A. et al. «*Denial of service in sensor networks* ». IEEE Computer, 35(10):54-62, 2002.
- 66.[XU 04] Y. Xu , et al. «*A wireless sensor network for structural monitoring*». Proc. SenSys, 2004.
- 67.[XU 01] Y. Xu, et al. «*Geography-informed energy conservation for ad hoc routing*». Proc. MobiCom, 2001.
- 68.[YAO 02] Y. Yao and J. Gehrke. «*The cougar approach to in-network query processing in sensor networks*». In SIGMOD Record, September 2002.
- 69.[YE 01] F. Ye, A. Chen, S. Lu and L. Zhang. «*A Scalable Solution to Minimum Cost Forwarding in large Sensor Networks*». In Proceedings of the IEEE International Conference on Computer Communication and Networks (ICCCN), 15-17 Oct. 2001, Pheonix, USA, 2001, pp. 304-309.
- 70.[YON 05] Eiko Yoneki, Jean Bacon. «*A survey of Wireless Sensor Network technologies: research trends and middleware's role* » University of Cambridge, Technical Report Number 646, September 2005.
- 71.[YOU 02] M. Younis, M. Youssef, and K. Arisha. «*Energy-aware routing in cluster-based sensor networks*». Proceedings of the 10th IEEE International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunications Systems (MASCOTS 2002), pp 129-136, 2002.
- 72.[YU 01] Y. Yu, et al. «*Geographical and Energy-Aware Routing : A Recursive Data Dissemination Protocol for Wireless Sensor Networks*». Technical Report, UCLA-CSD TR-01-0023, 2001.
- 73.[ZHA 04] Zhang, W. et al. «*Optimizing Tree Reconfiguration for Mobile Target Tracking in Sensor Networks.* » Proc. Infocom, 2004.
- 74.[OMN web] OMNet++ Discrete Event Simulation System. <http://www.omnetpp.org/>