



République Algérienne Démocratique et Populaire

وزارة التعليم العالي والبحث العلمي

*Ministère de l'Enseignement Supérieur et de la Recherche
Scientifique*

**UNIVERSITE DES SCIENCES ET DE LA TECHNOLOGIE
d'ORAN Mohamed Boudiaf**

Faculté des Mathématiques et informatique

Département Informatique

Spécialité : Informatique

Option : Informatique

THESE

Présentée par

Mr. Omar Smail

Pour l'obtention du diplôme de Doctorat en Sciences
en Informatique

Thème

**ROUTAGE MULTIPATH DANS
LES RESEAUX AD HOC**

SOUTENUE LE :

Devant le jury composé de :

<u>Qualité</u>	<u>Nom et prénom</u>	<u>Grade</u>	<u>Etb. d'origine</u>
Présidente	Mme. H.Belbachir	Professeur	USTO-MB
Directrice de thèse	Mme. R.Mekki	MCA	USTO-MB
Examineur	Mr. A.Chouarfia	Professeur	USTO-MB
Examineur	Melle. F.Debbat	MCA	Université de Mascara
Examineur	Mr. B.Kechar	MCA	Université d'Oran
Examineur	Mr. KM.Feraoun	MCA	Université de SBA

Année universitaire 2014-2015

Remerciements

Avant tout, le grand et le vrai merci à DIEU qui m'a donné la force et la vie pour accomplir cette modeste thèse.

Mon parcours durant cette thèse n'aurait jamais pu aboutir sans l'intervention de plusieurs personnes, auxquelles j'adresse ma gratitude la plus sincère. Tout d'abord, j'exprime mes profonds remerciements à mes directeurs de thèse.

En premier lieu Mme. R. Mekki pour son encouragement, sa disponibilité, ses conseils et sa sympathie qui m'ont permis de mener à bien mon travail. Je tiens à exprimer toute ma gratitude à Mme Z. Mekkakia et à Mr. B. Cousin, merci pour m'avoir encadré, avoir cru en moi en me confiant ce sujet de thèse et pour leurs accompagnements, encouragements et conseils au cours de ces années tant sur le plan technique que humain. Je remercie vivement Mr B. Messabih pour sa disponibilité hors pair, qui a toujours répondu présent dans les moments les plus difficiles.

J'adresse mes sincères remerciements à Mme H. Belbachir pour l'honneur qui nous a été accordé en présidant le jury. J'exprime ensuite ma plus profonde gratitude à Mr. A. Chouarfia qui a accepté de participer au jury, dont les connaissances suscitent énormément de respect. Je tiens à remercier également Mr B. Kechar d'avoir examiné ma thèse et de nous faire part de ces discussions de recherche dans le domaine des réseaux. J'ai beaucoup apprécié sa participation au jury. Mes vifs remerciements

à Mr. CM. Feraoune pour avoir accepté d'être membre de jury, pour ces remarques pertinentes concernant ma thèse et pour son immense preuve de disponibilité. J'aimerai aussi souligner la gentillesse particulièrement touchante de Melle F. Debbat qui m'a fait l'honneur de participer à mon jury de thèse, Merci.

Merci également à tous ceux qui m'ont accompagné et soutenu dans cette grande expérience de la vie que constitue une thèse. Mes remerciements vont aussi à tous mes amis et acteurs de mon quotidien Rebbah Mohamed, Debakla Mohamed et Mokhtari Chokri, qui ont contribué de près ou de loin dans ce travail, par leur amitié et l'aide qui règnent dans notre groupe. Un grand merci à tous mes collègues enseignants et membres administratifs de l'Université de Mascara.

Un égal merci à ma famille, parents, frère et sœurs, pour leur soutien et leur encouragement ; ils ont été le pilier de ma réussite, surtout ma mère et mon père qui n'ont cessé de croire en moi, je leur dédie cette thèse. Merci à ma femme d'avoir été à mon côté. A notre ange Kaouter qui me fait oublier la fatigue et le stress de tous les jours

J'adresse enfin mes remerciements à tous ceux qui, par leur courtoisie et sympathie, m'ont aidé, encouragé et soutenu.

Résumé

Les réseaux ad hoc sont des réseaux sans fil fonctionnant sans infrastructure et sans gestion centralisée du réseau ; ils sont constitués de nœuds mobiles avec des batteries à énergie limitée ; en conséquence, la conservation de l'énergie est une condition essentielle dans la conception de protocoles de routage pour ce type de réseaux. Afin de palier le problème d'énergie, de nouvelles formes de routage sont utilisées comme le routage multichemin. Le routage multichemin permet d'envoyer les paquets de données sur des chemins multiples et semble être une solution efficace pour ces réseaux.

Dans cette thèse, nous avons étudié les techniques de routage multichemin existant dans la littérature. Ensuite, nous avons proposé une extension du protocole de routage AODVM appelé AODVME+ (Ad hoc On-demand Distance Vector Multipath Energy+). Le but est d'améliorer la stratégie de routage par un routage multichemin avec une classification de chemins. Les chemins ayant un niveau d'énergie consistant sont sélectionnés pour la transmission des données.

Notre deuxième solution est un protocole de routage multichemin efficace en énergie, appelé AOMR-LM (Ad hoc On-demand Multipath Routing with Lifetime Maximization), qui conserve l'énergie résiduelle des nœuds et équilibre l'énergie consommée pour prolonger la durée de vie du réseau. Pour atteindre ce but, nous avons utilisé l'énergie résiduelle des nœuds pour calculer leurs niveaux d'énergie. Le mécanisme de sélection des chemins utilise ce niveau d'énergie pour classer les chemins dans le processus de découverte inverse. Nos deux solutions proposées améliorent la performance des réseaux mobiles ad hoc en prolongeant leurs durées de vie. Ces nouveaux protocoles sont comparés avec d'autres protocoles : AOMDV (Ad hoc On-demand Multipath Distance Vector), MMRE-AOMDV (Maximal Minimal Residual Energy-AOMDV) et ZD-AOMDV (Zone Disjoint-AOMDV). La performance des protocoles a été évaluée en termes de durée de vie du réseau, la consommation d'énergie, et le délai de bout en bout.

Mots clés : Réseau ad hoc, Routage multichemin, Efficacité énergétique.

Abstract

Ad hoc networks are wireless networks that can operate without infrastructure and without centralized network management ; they may contain nodes with limited battery power ; hence, energy conservation is a critical requirement in the design of routing protocols for ad hoc networks. To overcome the energy problem, new forms of routing are used as multipath routing. The multipath routing is used to send data packets over multiple paths and seems to be an effective solution for these networks. In this thesis, we studied the existing multipath routing techniques used in ad hoc networks. Then, we proposed an extension of the well-known routing protocol AODVM called AODVME+(Ad hoc On-demand Distance Vector Multipath Energy+). We propose to improve the multipath routing strategy with a path classification to allow the paths with the best energy level to be chosen.

Our second solution is an energy-efficient multipath routing protocol, called AOMR-LM (Ad hoc On-demand Multipath routing with Lifetime Maximization), which preserves the residual energy of nodes and balances the consumed energy to increase the network lifetime. To achieve this goal, we used the residual energy of nodes for calculating the node energy level. The multipath selection mechanism uses this energy level to classify the paths in the inverse discovery process. Our two proposed solutions improve the performance of mobile ad hoc networks by prolonging the lifetime of the network. These new protocols are compared with other protocols : AOMDV (Ad hoc On-demand Multipath Distance Vector), MMRE-AOMDV (Maximal Minimal Residual Energy-AOMDV) and ZD-AOMDV (Zone Disjoint-AOMDV). The protocols performance has been evaluated in terms of network lifetime, energy consumption, and end-to-end delay.

Keywords : Ad hoc network, Multipath routing, Energy efficiency.

Table des matières

Remerciements	ii
Résumé	iv
Abstract	v
Table des matières	vi
Table des figures	ix
Liste des tableaux	xi
1 Introduction	1
1 Contexte : Réseaux ad hoc	1
1.1 Représentation formelle d'un réseau ad hoc	2
1.2 Caractéristiques d'un réseau ad hoc	2
1.3 Applications des réseaux ad hoc	3
1.4 Avantages et inconvénients des réseaux ad hoc	3
2 Problématique : Le routage dans les réseaux ad hoc	4
3 Contributions	5
4 Plan du manuscrit	7
2 Routage dans les réseaux ad hoc	8
1 Taxonomie des protocoles de routage pour les réseaux ad hoc	8
1.1 Protocoles de routage proactifs	10
1.1.1 Optimized Link State Routing (OLSR)	10
1.2 Protocoles de routage réactifs	12
1.2.1 Le protocole AODV	12
1.3 Protocoles de routage hybride	16
1.3.1 Le protocole ZRP (Zone Routing Protocol)	16
2 Comparaison entre routage proactif et réactif	17
3 Conclusion	18
3 Routage multichemin dans les réseaux ad hoc	19
1 Principe du routage multichemin	20
1.1 Types de chemins	20
2 Objectifs du routage multichemin	21
2.1 Fiabilité	22
2.2 Équilibrage de charge et la consommation d'énergie	22
2.3 Surcoût de routage	22

2.4	Qualité de service (QoS)	23
2.5	Sécurité	23
3	Description des principaux protocoles de routage multichemin dans les réseaux ad hoc	24
3.1	Le protocole de routage multichemin AOMDV	24
3.1.1	Découverte de chemin	24
3.1.2	Constructions des chemins multiples	26
3.1.3	Maintenance des chemins	26
3.1.4	Critiques du protocole d'AOMDV	27
3.2	Ad hoc on Demand Distance Vector Multipath : AODVM	28
3.2.1	Critiques du protocole AODVM	29
3.3	Le protocole MP-OLSR (MultiPath Optimized Link State Routing)	29
3.3.1	Méthode de calcul des chemins dans MP-OLSR	30
3.3.2	Critiques du protocole MP-OLSR	32
4	Caractéristiques des protocoles de routage multichemin	32
5	Critères de performances	33
6	Justification de notre choix de la métrique de routage	33
7	Conclusion	34

4 Routage multichemin avec conservation d'énergie dans les réseaux

ad hoc		35
1	Consommation d'énergie dans les réseaux ad hoc	35
2	Nœud mobile et consommation d'énergie	36
2.1	Consommation d'énergie dans les différents modes d'opération des interfaces de communication sans fil	36
2.2	Sources de perte d'énergie	37
3	Notion de durée de vie d'un réseau	38
4	Conservation d'énergie	39
5	Conservation d'énergie dans les protocoles de routage à chemin unique	39
6	Les protocoles de routage multichemin avec conservation d'énergie	41
6.1	Le protocole MMRE	41
6.1.1	Calcul de l'énergie résiduel minimale des nœuds	41
6.1.2	Tri des chemins multiples	43
6.2	Le protocole REER	43
6.2.1	Fonctionnement du protocole REER	44
6.2.2	Découverte des chemins	44
6.2.3	La maintenance des chemins	45
6.2.4	Transmission de données	46
6.3	Le protocole ZD-AOMDV	46
6.3.1	Les modifications nécessaires d'AODV pour l'implémentation de ZD-AOMDV	47
6.3.2	Principe de fonctionnement du protocole ZD-AOMDV	47
6.3.3	Processus de découverte et de sélection des chemins	48
6.4	Critiques des différentes techniques exposées	51
7	Conclusion	52

5	Protocole de routage multichemin AODVME+	53
1	Protocole de routage multichemin AODVME+	53
1.1	Hypothèses et définition du problème	54
1.2	Découverte des chemins dans AODVME+	55
1.3	Sélection des chemins et la transmission des paquets de données	57
1.4	Maintenance des chemins	58
2	Evaluation des performances du protocole AODVME+	59
2.1	Paramètres de performance	59
2.2	Environnement de simulation	59
2.3	Evaluation des performances	59
2.3.1	Consommation d'énergie	61
2.3.2	Délai de bout en bout	63
2.3.3	Surcout du routage	63
3	Conclusion	64
6	Protocole de routage multichemin AOMR-LM	65
1	Justification du choix du protocole de base : AOMDV	65
2	Le protocole AOMR-LM : Ad hoc On-demand Multipath Routing with Lifetime Maximization	66
2.1	Définition du problème	67
2.2	Découverte des chemins multiples	68
2.3	Sélection des chemins multiples	69
2.4	Transmission de données	74
2.5	Maintenance des chemins	74
3	Evaluation des performances du protocole AOMR-LM	75
3.1	Paramètres de performance	75
3.2	Analyse du coefficient α	75
4	Evaluation des performances	76
4.1	Durée de vie du réseau	77
4.2	Ecart type et coefficient de variation de la durée de vie vs le nombre de nœuds épuisés	78
4.3	Consommation moyenne d'énergie	79
4.4	Delai moyen de bout en bout	80
5	Conclusion	81
7	Conclusion et perspectives	82
1	Conclusion	82
2	Perspectives	84
	Bibliographie	85

Table des figures

2.1	Taxonomie des protocoles de routage pour les réseaux ad hoc.	9
2.2	Utilisation des MPR dans le routage.	11
2.3	Diffusion des paquets <i>RREQ</i> : création de chemin inverse.	14
2.4	Établissement de chemin entre une source <i>s</i> et une destination <i>d</i>	15
2.5	Zone de routage du nœud <i>s</i> (nombre de sauts = 2)	17
3.1	Routage à chemin unique et routage multichemin.	20
3.2	Chemins à nœuds disjoints.	21
3.3	Chemins à liens disjoints.	21
3.4	Découverte de chemins à liens disjoints dans AOMDV	25
3.5	Structure des entrées des tables de routage de AODV et AOMDV.	26
3.6	Le format du paquet RREQ du protocole AODVM.	28
3.7	Une entrée de la table de routage d'AODVM.	29
4.1	Structure d'une entrée de la table de routage de MMRE.	43
4.2	Structure du paquet Hello.	45
4.3	Structure du paquet RREQ.	45
4.4	Structure du paquet KeepAlive.	46
4.5	Exemple d'une topologie d'un réseau.	49
4.6	Première phase de l'algorithme ZD-AOMDV.	49
4.7	Deuxième phase de l'algorithme ZD-AOMDV.	50
4.8	Les chemins sélectionnés (suivant l'exemple).	51
5.1	AODVME+ : Exemple de diffusion des paquets <i>RREQ</i> et <i>RREP</i>	55
5.2	Structure de la table de routage du protocole AODVME+	57
5.3	AODVME+ : Exemple de découverte des chemins.	58
5.4	Consommation d'énergie au cours du temps. ($\alpha=0.2$)	61
5.5	Consommation d'énergie au cours du temps. ($\alpha=0.4$)	61
5.6	Consommation d'énergie au cours du temps. ($\alpha=0.6$)	62
5.7	Consommation d'énergie au cours du temps. ($\alpha=0.8$)	62
5.8	Délai de bout en bout vs la taille du réseau.	63
5.9	Surcout du routage vs la taille du réseau.	64
6.1	Structure de la table de routage du protocole AOMR-LM.	70
6.2	Réception des <i>RREQs</i> par la destination	70
6.3	Le choix des nœuds voisins pour l'envoi des <i>RREPs</i>	73
6.4	Les chemins découverts par le protocole AOMR-LM.	73

TABLE DES FIGURES

6.5	α_{min} versus K	76
6.6	Durée de vie du réseau vs nombre de nœuds épuisés.	77
6.7	Coefficient de variation de la durée de vie.	79
6.8	L'énergie consommée vs temps de simulation.	79
6.9	délai de bout en bout vs nombre de nœuds. (a) Mobilité [0 , 2.5] m/s	80
6.10	délai de bout en bout vs nombre de nœuds. (b) Mobilité [2.6 , 5] m/s	81

Liste des tableaux

4.1	Valeurs de la consommation d'énergie (en mW) de quelques interfaces de communication sans fil IEEE 802.	37
5.1	AODVME+ : Paramètres de simulation.	60
6.1	Mesure d α_{min} relative au nombre de nœuds du réseau K	76
6.2	Paramètres de simulation du protocole AOMR-LM.	77
6.3	moyenne, l'écart type et le coefficient de variation de la durée de vie vs le nombre de nœuds épuisés.	78

Chapitre 1

Introduction

Ces dernières années, le domaine de la télécommunication a pris un nouvel envol grâce à l'évolution des technologies sans fil et l'apparition des unités mobiles de calculs. Les unités mobiles possédant toutes des caractéristiques différentes (indépendance, capacité de stockage, vitesse de calcul...) accèdent désormais aux réseaux à travers une interface de communication sans fil. La technologie sans fil offre une grande flexibilité d'emploi et de mise en réseau des unités dont le câblage serait trop onéreux à réaliser dans leur totalité, voire même impossible. Parmi les caractéristiques de ces nouveaux réseaux, nous pouvons citer l'architecture communicante qui est sans fil, la spontanéité de leur mise en place, leur auto-configuration, l'hétérogénéité des équipements les composant, leur mobilité, leur consommation d'énergie, qui doit être contrôlée, et la possibilité de supporter un grand nombre d'utilisateurs. Les réseaux sans fil se divisent en deux types d'architectures. Les architectures avec infrastructure ou avec point d'accès et les architectures sans infrastructure. Dans les réseaux avec point d'accès, autrement dits réseaux avec infrastructure, les unités mobiles se connectent à une station de base, qui partage la bande passante disponible, une unité mobile peut communiquer avec les autres unités à travers la station de base à laquelle elle est directement rattachée. Chaque station de base possède une cellule au sein de laquelle les unités mobiles peuvent émettre et recevoir des paquets. La deuxième architecture des réseaux sans fil est sans infrastructure (sans station de base).

Nous décrivons, plus en détails, dans la suite de ce chapitre les réseaux ad hoc, la problématique de routage, et nous finissons par nos contributions dans le domaine.

1 Contexte : Réseaux ad hoc

Cette thèse s'inscrit dans le contexte des réseaux sans fil sans infrastructure, précisément les réseaux ad hoc, appelés aussi MANET (Mobile ad hoc network). Un réseau ad hoc est une collection d'unités mobiles, interconnectées par une technologie sans fil formant un réseau temporaire sans l'aide de toute infrastructure préexistante ou administration centralisée. Les unités mobiles (nœuds) interagissent et peuvent

coopérer pour s'échanger des services. Un nœud peut à la fois communiquer directement avec d'autres nœuds ou servir de relai. Un relai permet à des nœuds se trouvant hors de portée radio les uns des autres de communiquer. En conséquence, on utilise le terme réseaux (multi-sauts) dans la littérature pour désigner les réseaux ad hoc ou MANET. Ces réseaux peuvent exister temporairement pour répondre à un besoin ponctuel de communication. Ils peuvent être déployés à bon escient dans des situations d'urgence telles que les missions de secours en cas d'incendies ou de catastrophes naturelles (inondations, tremblements de terre, etc.), ainsi que les missions d'exploration. Nous décrivons dans la suite de cette section la représentation formelle, les caractéristiques, les applications, les avantages et les inconvénients des réseaux ad hoc.

1.1 Représentation formelle d'un réseau ad hoc

Un réseau ad hoc comprend des unités mobiles, appelées nœuds, qui sont libres de se déplacer sans contrainte. Un réseau ad hoc est donc un système autonome de nœuds mobiles. Ce système peut fonctionner d'une manière isolée ou s'interfacer à des réseaux fixes au travers de passerelles [1]. Formellement, un réseau ad hoc peut être représenté par un graphe non orienté $G_t = (V_t, E_t)$ où : V_t : représente l'ensemble des nœuds (les unités mobiles) du réseau à l'instant t . E_t : modélise l'ensemble des connexions (arcs) qui existent entre ces nœuds à l'instant t . Soit i et j deux nœuds de V_t , l'arc (i, j) existent, si et seulement si, i peut envoyer directement un paquet à j à l'instant t , on dit alors que j est voisin de i . Les couples appartenant à E_t dépendent de la position des nœuds et de leur portée de communication. Si on retient l'hypothèse que tous les nœuds ont une portée R identique, et si $d(i, j)$ désigne la distance entre les nœuds i et j , alors l'ensemble E_t peut-être défini comme suit : $E_t = \{(i, j) \in V_t \mid d(i, j) \leq R\}$ [2].

1.2 Caractéristiques d'un réseau ad hoc

Plusieurs caractéristiques différencient un réseau ad hoc d'un réseau filaire, on cite [3] :

- Une topologie dynamique : les unités mobiles du réseau se déplacent d'une façon libre et arbitraire. Par conséquent, la topologie du réseau peut changer, à des instants imprévisibles, d'une manière rapide et aléatoire.
- L'absence d'infrastructure : les réseaux mobiles ad hoc se distinguent des autres réseaux mobiles par la propriété d'absence d'infrastructures préexistantes et de tout genre d'administration centralisée.
- Un routage multi-saut : cela signifie que des communications entre deux nœuds doivent pouvoir s'effectuer même si ceux-ci sont hors de portée de communication directe.
- Chaque nœud du réseau est à la fois station et routeur : chaque nœud du réseau contribue au bon acheminement des informations dans le réseau. Pour ce faire, chaque nœud essaie de posséder une connaissance partielle du réseau la plus étendue possible pour jouer le rôle de routeur pour lui-même et pour tout autre nœud qui le lui demandera.

- Bande passante limitée[2] : une des caractéristiques primordiales des réseaux basés sur la communication sans fil est l'utilisation d'un médium de communication partagé (ondes radios). Le partage fait que la bande passante réservée à une station sera modeste.
- Contraintes d'énergie : les unités mobiles sont alimentées par des sources d'énergie autonomes comme les batteries électriques ou les autres sources consommables. Le paramètre d'énergie doit être pris en considération dans tout contrôle fait par le système.
- Sécurité physique limitée : les réseaux mobiles ad hoc sont plus touchés par le paramètre de sécurité que les réseaux filaires.
- Transmission peu fiable : les erreurs de transmission radio sont plus fréquentes dans les réseaux sans fil que dans les réseaux filaires.
- Interférences : les liens radios ne sont pas isolés, deux transmissions simultanées sur une même fréquence ou, utilisant des fréquences proches peuvent interférer.

1.3 Applications des réseaux ad hoc

Les applications ayant recours aux réseaux ad hoc couvrent de nombreux domaines, incluant les applications militaires, les bases de données réparties, l'enseignement à distance, les systèmes de fichiers répartis, la simulation distribuée interactive, les applications de calcul distribué (ou méta computing), etc.

D'une manière générale, les réseaux ad hoc peuvent être utilisés dans toutes les applications où le déploiement d'une infrastructure réseau filaire est trop pénible, soit parce que difficile à mettre en place, soit parce que la durée d'installation du réseau ne justifie pas de câblage à demeure. Ils seront alors d'un grand apport lors des catastrophes naturelles ou des incendies, où il sera indispensable de disposer rapidement d'un réseau pour organiser les secours et les opérations de sauvetage.

1.4 Avantages et inconvénients des réseaux ad hoc

Les réseaux ad hoc ont de nombreux avantages mais ils présentent tout de même quelques inconvénients.

Avantages :

- Financier : le réseau ad hoc permet d'éviter l'obligation d'un câblage coûteux qui peut se révéler rapidement obsolète ou inutile en cas de déménagements de locaux.
- Tolérance aux pannes : un réseau ad hoc continue à fonctionner même si quelques nœuds tombent en panne, ceci est dû au fait qu'il ne comporte pas de nœuds centraux.
- Mobilité : le réseau sans fil permet aux stations d'être entièrement mobiles, tant qu'elles restent dans la portée radio.
- Evolutivité : les réseaux ad hoc peuvent être dimensionnés au plus juste et suivre simplement l'évolution des besoins.
- Simplicité et rapidité de configuration et d'installation du matériel dans le cadre du réseau sans fil.

Inconvénients :

- Étroitesse et la diminution de la bande passante allouée du fait de la formation de goulots d'étranglement occasionnés par le partage du canal de communication.
- Les changements de topologie du réseau mobile ad hoc, causés notamment par des déconnexions volontaires ou involontaires des utilisateurs, ainsi que par des obstacles et des interférences dans les liaisons radio, qui perturbent les communications.
- L'utilisation des batteries, en tant que source d'énergie par les unités mobiles, affecte la durée de vie du réseau mobile ad hoc.
- En raison des limitations du milieu, il n'est pas possible de transmettre plusieurs paquets sur le même canal radio, donc il y a des possibilités plus élevées des collisions de paquet. Les collisions et les interférences provoquent des pertes de paquet plus probablement.

Dans les réseaux ad hoc, les unités mobiles sont autonomes, elles se déplacent d'une façon libre et arbitraire. Ceci influe sur la procédure de gestion de l'acheminement des données, autrement dit le routage.

2 Problématique : Le routage dans les réseaux ad hoc

Le routage dans un réseau est un mécanisme par lequel plusieurs chemins sont sélectionnés pour acheminer les paquets de données d'un nœud source jusqu'à un nœud de destination. Il consiste à assurer une stratégie qui garantit, à n'importe quel moment, un établissement des chemins qui soient correctes et efficaces entre n'importe quelle paire de nœud appartenant au réseau, ce qui assure l'échange des paquets d'une manière continue.

Le routage dans les réseaux ad hoc constitue une tâche difficile à réaliser. En effet, en l'absence d'infrastructure, le changement de topologie et la limitation des ressources des nœuds, l'acheminement d'un paquet d'un nœud vers la destination n'est pas aisé. Il s'agit de découvrir un chemin entre les deux, un chemin qui est amené à changer dans le temps et qu'il faudra redécouvrir lors d'une prochaine communication. Le routage, opération qui permet l'acheminement des paquets, peut être réalisé de différentes façons, en fonction des caractéristiques du réseau et des périphériques utilisés. En général, les protocoles de routage existants peuvent être, soit proactifs, réactifs ou hybrides. Les protocoles proactifs établissent les chemins qui seront disponibles avant leurs sollicitations. Ces protocoles reposent sur le principe du routage basé sur l'état des liens (Link-State) ou sur le vecteur de distance (Distance Vector) déjà utilisés dans les réseaux filaires. Le problème avec un protocole proactif est que les changements de chemins peuvent être plus fréquents que l'envoi de paquet vers un destinataire. Dans ce cas, la plupart des informations de routage ne sont jamais utilisées, ce qui gaspille les ressources du réseau en termes d'espace mémoire et de temps de calcul. La deuxième classe est les protocoles réactifs qui créent et maintiennent les chemins selon les besoins. Ainsi, aucun chemin ou information de routage ne sera

calculée tant qu'un nœud n'a pas initié une communication pour envoyer un paquet vers le nœud destinataire. La faiblesse de ces protocoles est que le délai nécessaire pour déterminer un chemin peut être significatif. De plus, la procédure de recherche de chemin du protocole réactif génère un trafic de contrôle important. Une solution pour pallier aux faiblesses des protocoles de routage proactifs et réactifs, est de combiner les avantages de ces deux classes dans un schéma hybride. La mobilité des nœuds d'un réseau ad hoc, les ressources limitées des nœuds ainsi que la bande passante réduite provoquent une forte dynamique de la topologie du réseau. Un chemin ainsi établi pour l'acheminement d'un paquet, à un moment donné peut ne plus être exploitable quelques instants plus tard. Les ruptures des liens déclenchent les processus de découverte des chemins, en abusant des ressources du réseau. En plus peu de nœuds sont impliqués dans le processus d'acheminement des paquets, alors que d'autres sont fortement congestionnés. Ceci pénalise les protocoles de routage classiques utilisant un seul chemin pour le routage et dégrade les performances du réseau. Puisque tous les nœuds dans un réseau ad hoc peuvent être connectés de façon dynamique, il est généralement possible d'établir plus d'un chemin entre une source et une destination. Lorsque cette propriété des réseaux est utilisée dans le processus de routage alors on parle du routage multichemin (multipath). Le routage multichemin semble être une solution efficace dans les réseaux ad hoc à forte mobilité avec des ressources limitées, en permettant de prémunir contre le problème de rupture des chemins. Un bon routage multichemin permet aussi d'économiser la bande passante, de conserver l'énergie et de distribuer le trafic à écouler entre une source et une destination sur plusieurs chemins.

3 Contributions

Les réseaux sans fil peuvent contenir différents types d'unités mobiles (agendas électroniques, téléphones cellulaires, capteurs, ordinateurs portables etc.). Les unités mobiles sont alimentées par des batteries dont la durée de vie est cruciale pour le maintien des communications sans fil. L'évolution de la technologie des batteries a pris du retard par rapport aux progrès réalisés en matière de communication et de technologie informatique. Jusqu'à maintenant les capacités des batteries n'ont pas pu être améliorées significativement. C'est pourquoi les efforts se concentrent sur la conception d'équipements et de protocoles prenant en compte l'économie d'énergie. Une unité mobile est composée de plusieurs équipements qui consomment l'énergie de sa batterie (moniteur, processeur, mémoire principale, disque dur, carte de réseau sans fil). Cependant la consommation de la carte réseau sans fil peut atteindre plus de la moitié de l'énergie consommée. Ce qui implique, d'une part, que les applications logiciels doivent essayer de réduire au maximum les transmissions inutiles et d'autre part, que les protocoles prenant en compte l'économie d'énergie doivent réduire la consommation l'énergie au minimum requis. Des techniques optimisant la consommation d'énergie ont fait l'objet de plusieurs travaux de recherche dans les réseaux mobile ad hoc. Les contraintes d'énergie corrélées avec le problème de déploiement d'un grand nombre de nœuds ont mis en évidence la nécessité d'utiliser des protocoles de routage prenant en compte les problèmes de limitation liés à la consommation des ressources énergétiques dans les réseaux mobiles ad hoc. Mais,

les politiques de routage classiques, comme celle du "plus court chemin", peuvent avoir un impact négatif sur la consommation d'énergie dans un réseau mobile ad hoc. C'est à partir de ces constats qu'a émergé la classe des protocoles de routage dite "à basse consommation d'énergie". Toutefois, la conservation d'énergie vient souvent au prix d'une consommation plus élevée de la bande passante (généralisé par le trafic de contrôle). Par conséquent, des compromis doivent être trouvés pour que les performances du réseau mobile ad hoc restent acceptables, tout en diminuant les pertes d'énergie.

Cette thèse est consacrée au routage multichemin (multipath) avec conservation de l'énergie dans les réseaux ad hoc. notre travail présenté dans cette thèse vise à améliorer la stratégie du routage existante en proposant de nouvelles techniques de routage multichemin avec conservation d'énergie, permettant l'amélioration des performances des réseaux ad hoc.

Notre première contribution concerne un nouveau protocole de routage multichemin à nœuds disjoints pour les réseaux ad hoc, appelé AODVME+ (Ad hoc On-demand Distance Vector Multipath Energy+). Il exploite l'énergie des nœuds efficacement (prolonge la vie du réseau), contrôle le nombre de paquets de routage (overhead), minimise les délais de bout en bout, et minimise la consommation d'énergie des nœuds du réseau. Notre protocole est conçu principalement pour les nœuds ayant une batterie à capacité limitée, et pour les réseaux où les ruptures des liens et les défaillances des chemins se produisent fréquemment (à cause de la mobilité des nœuds ou de leur épuisement énergétique). L'épuisement de l'énergie d'un nœud provoque l'interruption de l'acheminement des paquets de données. L'idée principale de notre protocole est de découvrir plusieurs chemins entre une source et une destination dans un réseau mobile ad hoc, en exploitant l'énergie résiduelle des nœuds pour empêcher un nœud critique d'épuiser sa réserve d'énergie. Après le processus de découverte de chemins, les chemins multiples découverts peuvent avoir des valeurs minimales d'énergie résiduelle des nœuds similaires (ou proches). Un nouveau facteur peut être combiné avec cette valeur minimale. Ce facteur est l'énergie résiduelle moyenne de batteries des nœuds de chaque chemin.

Nous avons ensuite proposé un deuxième protocole de routage multichemin à liens disjoints, baptisé AOMR-LM (Ad hoc On-demand Multipath Routing with Lifetime Maximization), ayant comme objectif principal le prolongement de la durée de vie d'un réseau ad hoc. Nous avons utilisé les niveaux d'énergie des nœuds pour classer les nœuds des chemins entre une source et une destination. Le niveau d'énergie d'un nœud est calculé en fonction de l'énergie résiduelle du nœud ; cette information est collectée lors du processus de découverte des chemins. Pour définir la classe d'un nœud un seuil d'énergie des nœuds et un coefficient doivent être définis. Notre protocole sélectionne les nœuds de même classe pour construire un chemin donné. A la fin de l'opération de découverte des chemins nous aurons un ensemble de chemins homogènes en terme d'énergie. Les chemins les mieux classés sont sélectionnés afin d'équilibrer l'énergie consommée pour une meilleure conservation d'énergie. La technique proposée ignore les nœuds avec un niveau d'énergie critique et fait en sorte qu'ils ne participent pas au routage, sinon on peut avoir plusieurs ruptures des liens à cause de l'épuisement de la batterie des nœuds.

Nous avons implémenté ces deux protocoles sous le simulateur ns-2 et nous les avons comparé avec les principaux protocoles de routage multichemin existants à savoir :

AODVM, AOMDV, MMRE et ZD-AOMDV.

4 Plan du manuscrit

La suite de cette thèse est organisée en six chapitres. Le deuxième chapitre propose un état de l'art sur le routage dans les réseaux ad hoc. Nous présentons en détail, dans le chapitre 3, le routage multichemin et les différentes solutions existantes dans la littérature. Le quatrième chapitre est consacré au routage multichemin avec conservation d'énergie. Dans le chapitre 5, nous présentons notre protocole multichemin AODVME+. Le chapitre 6 est dédié à notre deuxième solution qui est le protocole AOMR-LM. Les deux derniers chapitres sont finalisés par des évaluations de performance. Enfin, le chapitre 7 clôt cette thèse, analyse les résultats obtenus et décrit les travaux futurs.

Chapitre 2

Routage dans les réseaux ad hoc

Les communications dans les réseaux ad hoc constituent une tâche difficile à réaliser. En effet, en l'absence d'infrastructure, l'acheminement d'un paquet d'une source vers son nœud de destination n'est pas aisé. Il s'agit de découvrir un chemin entre les deux nœuds, un chemin qui est amené à changer dans le temps et qu'il faudra redécouvrir lors d'une prochaine communication. Le routage, dans les réseaux ad hoc, peut être réalisé de différentes façons, en fonction des caractéristiques du réseau et de ces périphériques utilisés.

Dans ce chapitre, nous allons aborder le routage dans les réseaux ad hoc, en regroupant ces protocoles en trois catégories : les protocoles proactifs, réactifs et hybrides. Chaque catégorie est illustrée par un protocole typique.

1 Taxonomie des protocoles de routage pour les réseaux ad hoc

Plusieurs protocoles de routage destinés aux réseaux mobiles ad hoc ont été réalisés [4-8], ces protocoles peuvent être classés selon plusieurs critères. Ils peuvent être classés selon le type de vision qu'ils ont du réseau et les rôles attribués à ses nœuds ; si tous les nœuds du réseau jouent le même rôle pour la fonction de routage ou non (plat ou hiérarchique), ou encore selon l'information utilisée pour calculer les chemins selon une vision locale ou globale du réseau (vecteur de distance ou état de liens). Un autre critère peut être utilisé pour différencier entre les protocoles de routage, il s'agit de la méthode utilisée pour construire un chemin entre un nœud source et un autre nœud destination (réactive ou proactive).

Pour maintenir leur table de routage, les protocoles proactifs recherchent dans un intervalle régulier les différents chemins disponibles dans le réseau. Quand un paquet doit être transmis, son chemin est alors connu à l'avance et peut ainsi être immédiatement utilisé. Les protocoles réactifs commencent la recherche d'un chemin uniquement avant de transmettre un paquet. Avec un protocole proactif, les chemins sont disponibles immédiatement. Cependant, le trafic induit par les mes-

sages de contrôle (de mise à jour des tables de routage) peut être important et partiellement inutile du fait qu'il ne mis pas à jour les informations des chemins pré-découverts. De plus, la taille des tables de routage croît linéairement en fonction du nombre de nœuds. A l'opposé, dans le cas d'un protocole réactif, aucun message de contrôle ne charge le réseau pour des chemins inutilisés. Mais, pour ces derniers, le processus de recherche du chemin (par inondation) peut être coûteux et provoquer des délais importants avant la découverte du bon chemin.

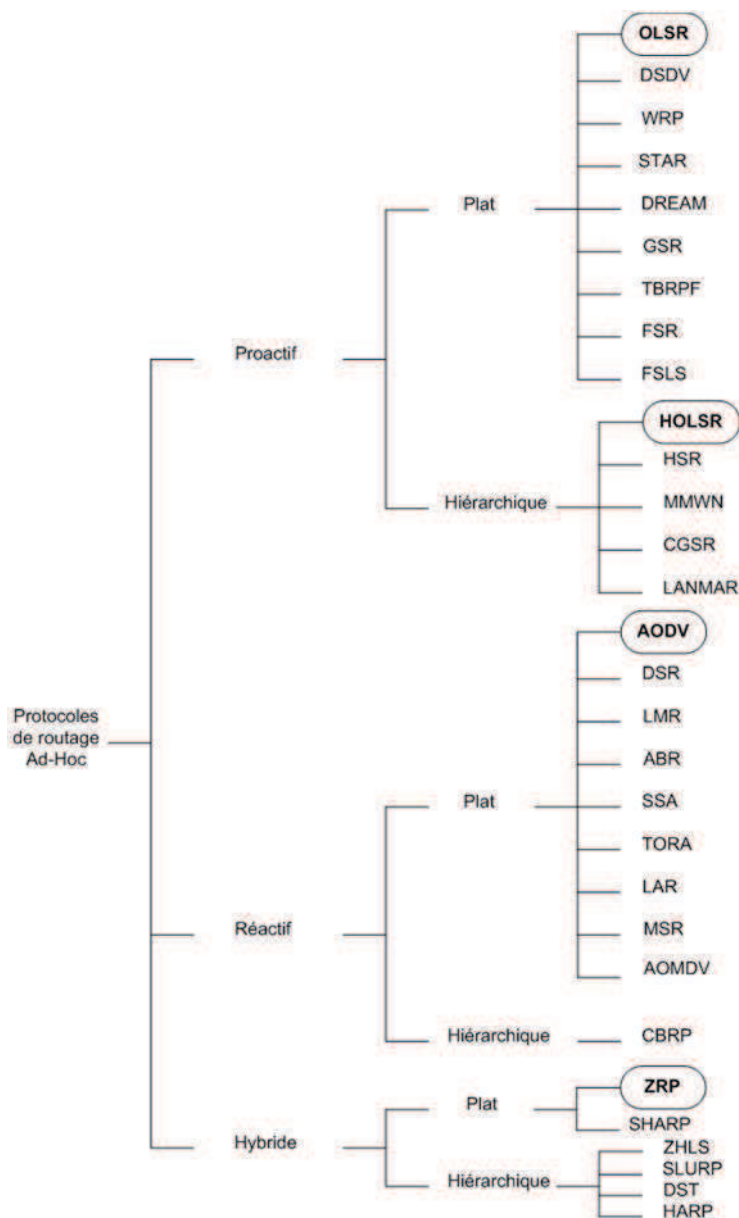


FIGURE 2.1 – Taxonomie des protocoles de routage pour les réseaux ad hoc.

En terme de performances, les protocoles orientés topologie (dit à états de liens) convergent plus rapidement que les protocoles orientés destination (dit à vecteur de distance). Cependant, dans le cas de réseaux à forte mobilité, le trafic induit par les

fréquents messages de contrôle est souvent pénalisant [9].

La figure 2.1 donne une nomenclature et une classification des principaux protocoles proactifs, réactifs et hybrides développés ces dernières années [10]. Chaque classe de protocole peut être de type plat où tous les nœuds sont égaux dans les décisions de routage, ou hiérarchiques, dans ce type de routage les nœuds se distinguent selon leurs rôles confiés et leurs prises de décision dans le mécanisme de routage.

1.1 Protocoles de routage proactifs

Les protocoles de routage ad hoc proactifs sont inspirés des protocoles conçus pour réseaux filaires. Ces mécanismes, tels que DSDV [11] ou OLSR [12], échangent des informations de routage de façon périodique. Les tables de routage sont donc continuellement mise à jour et chaque nœud connaît à tout moment un chemin vers un autre nœud. Les proactifs sont par leur anticipation de la découverte des chemins, appropriés pour les réseaux avec une large répartition des nœuds source et destination ou des applications nécessitant de la qualité de service [13]. Les performances du routage proactif se caractérisent par :

- Les latences les plus faibles, puisque les applications souhaitant émettre peuvent supposer l’existence de chemins à jour et valides,
- Une consommation d’énergie importante et une utilisation importante de la bande passante, dues au fait que le protocole de mise à jour des chemins est employé en permanence, même si les chemins ne sont jamais exploités.

Parmi les protocoles proactifs de base les plus connus : DSDV et OLSR. Dans ce qui suit nous présenterons le protocole OLSR, qui est plus performant dans un réseau dense et à forte mobilité que DSDV [14].

1.1.1 Optimized Link State Routing (OLSR)

OLSR [12] (*Optimized Link State Routing*) est un protocole à états de lien qui se base sur l’utilisation de nœuds relais multipoint (*MultiPoint Relay : MPR*) pour inonder le réseau de façon efficace en réduisant le nombre de paquets redondants au sein d’une même région. Pour cela, chaque nœud i sélectionne, indépendamment des autres nœuds, un ensemble minimal de nœuds relais multipoint ($MPR(i)$) parmi ses voisins directs. L’ensemble des voisins directs des nœuds $MPR(i)$ doit contenir tous les nœuds situés à deux sauts du nœud i . Ainsi, lors d’une diffusion, tous les voisins reçoivent et traitent le message mais seulement les nœuds choisis comme MPR le retransmettent, ce qui diminue considérablement le nombre de messages émis dans le réseau. La figure 2.2 illustre le cas de routage avec et sans $MPRs$, ces derniers réduisent considérablement le nombre de paquets de contrôle (qui sont diffusés). Les nœuds OLSR échangent périodiquement des messages de contrôle et maintiennent des chemins pour atteindre tous nœud OLSR du réseau. Le protocole OLSR réalise principalement deux fonctionnalités : détection de voisinage et gestion de la topologie.

Détection de voisinage Chaque nœud doit détecter toutes les interfaces de ses voisins ayant un lien direct (à 1 saut) et symétrique avec l’une de ses interfaces.

En raison de l'incertitude de la propagation radio, le lien entre deux voisins peut être unidirectionnel. Les liens doivent donc être vérifiés dans les deux sens avant de les considérer comme valides. Chaque nœud doit diffuser périodiquement un message *Hello* dans son voisinage. Ce message contient les informations relatives aux interfaces entendues par ce nœud : la liste des adresses des interfaces des nœuds voisins avec l'état de chaque lien. Un lien peut avoir l'un des quatre états suivant : symétrique, asymétrique, MPR ou perdu.

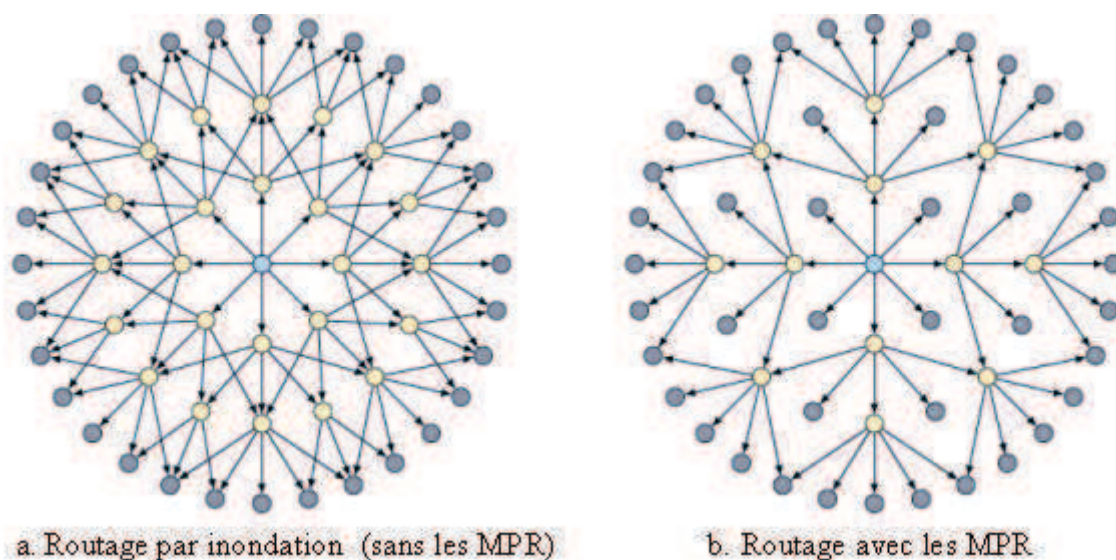


FIGURE 2.2 – Utilisation des MPR dans le routage.

a. Routage par inondation (sans les MPR) b. Routage avec les MPR

Gestion de la topologie Chaque nœud sélectionné comme MPR dans le réseau diffuse périodiquement à tous les nœuds du réseau des messages de topologie, appelés Topology Control (TC) avec une fréquence déterminée par un intervalle défini ($Tc_Interval$). Un message TC contient les nœuds ayant sélectionnés le nœud s origine du TC comme relai multipoint et un numéro de séquence associé à l'ensemble des sélecteurs de relai multipoint ; numéro qui sera incrémenté à chaque changement de cette ensemble. Seuls les nœuds MPRs ont le droit de rediffuser les messages TC. La réception de ces messages TC, permet à chaque nœud dans le réseau de maintenir une base contenant les informations topologiques du réseau qui donne une vision globale de la topologie du réseau. Dans cette topologie, on distingue les nœuds MPR de ceux qui ne le sont pas.

Chaque nœud dans le réseau calcule un chemin pour atteindre tout autre nœud présent dans le réseau. Ce calcul est basé sur les informations de voisinage et de topologie rafraichies périodiquement. Un chemin pour chaque destination, sous la forme du prochain nœud (next hop) est stocké dans la table de routage. Un nœud utilise l'algorithme de Dijkstra, un algorithme de plus court chemin, sur le graphe de topologie partielle construit à partir des liens existants dans la table de voisinage.

Avantages et inconvénients du protocole OLSR Le protocole OLSR réduit sensiblement la surcharge due aux messages de contrôle en utilisant un processus de découverte des chemins par inondation optimisé basé sur un sous-groupe de nœuds (*Relai multipoint*).

Il s'adapte parfaitement aux protocoles de l'Internet et fournit à chaque nœud une topologie du réseau à tout instant [15]. L'inconvénient majeur du protocole OLSR est l'immense volume d'information stockée dans chaque nœud du réseau : la liste des voisins, la liste des MPR, la topologie du réseau et la table de routage. Ceci épuise les ressources des nœuds et engendre un calcul lors des mises à jour des listes, induisant une dégradation des performances du réseau.

1.2 Protocoles de routage réactifs

Dans cette catégorie de protocoles, la découverte d'un moyen d'acheminement des messages ne se fait que lorsqu'un nœud désire communiquer avec un autre. La découverte de chemin se fait donc à la demande. Grâce à cette méthode, les nœuds du réseau ne génèrent aucun trafic de contrôle sans qu'il soit nécessaire. Ceci permet de réduire la charge du trafic dans le réseau. Les protocoles de routage réactifs peuvent être classés en deux catégories [16] :

- **Routage par la source** : Dans les protocoles à routage par la source, les paquets de données portent dans leurs entêtes les adresses de tous les nœuds constituant le chemin à partir du nœud source jusqu'au nœud destination. De ce fait, les nœuds intermédiaires acheminent les paquets selon les informations qui se trouvent dans l'entête de chaque paquet de données. Cela veut dire que les nœuds intermédiaires n'ont pas besoin de maintenir des informations sur les chemins actifs. De plus, ils n'ont pas besoin de maintenir la connectivité avec leurs voisins.
- **Routage saut par saut** : Dans le routage saut par saut chaque paquet de données porte uniquement l'adresse du nœud destination et celle du saut prochain. De ce fait, chaque nœud intermédiaire utilise sa table de routage pour acheminer chaque paquet de données. L'avantage de ce type de routage est qu'il permet une meilleure adaptabilité aux changements fréquents de topologie, car chaque nœud utilise les informations les plus fraîches dans le routage.

Parmi les protocoles de routages réactifs, nous pouvons citer AODV [17] et DSR [18], ils sont les plus connus. Dans la section suivante, nous présenterons le protocole réactif AODV, vu sa performance en terme de gestion des liens brisés, comparativement au protocole DSR.

1.2.1 Le protocole AODV

Le protocole de routage avec vecteur de distance à la demande AODV (Ad hoc On Demand Distance Vector) [17], est un protocole de routage réactif unicast. Il est considéré comme une amélioration du protocole DSDV [11]. Il crée un chemin vers un nœud destination à la demande, contrairement à DSDV, qui maintient un chemin vers la totalité des destinations connues. Une fois le chemin découvert et validé, les nœuds qui ne se trouvent pas sur le chemin actif ne maintiennent aucune

information de routage et ne participent à aucun échange de mise à jour. A cause de la mobilité des nœuds dans les réseaux ad hoc, les chemins changent fréquemment ce qui fait que les chemins maintenus par certains nœuds, deviennent invalides. Un numéro de séquence permet d'utiliser les chemins les plus récents (fresh routes), afin de forcer les mises à jour si nécessaire et d'éviter la formation de boucles de routage. AODV maintient les chemins d'une façon distribuée en gardant une table de routage, au niveau de chaque nœud intermédiaire appartenant au chemin découvert. Une entrée de la table de routage contient essentiellement [17] :

- Adresse du nœud destination à atteindre
- Adresse du nœud suivant (prochain saut)
- Nombre de sauts nécessaire pour atteindre le nœud de destination
- Numéro de séquence du nœud destination
- Durée de vie pour laquelle le chemin reste à la disposition du nœud source
- Liste des voisins qui utilisent ce chemin (adresses d'éventuels nœuds prédécesseurs qu'utilise le nœud courant comme un prochain saut pour atteindre le nœud destination)

Le protocole AODV définit deux types de procédures : la découverte des chemins et la maintenance des chemins. La découverte des chemins permet de trouver un chemin pour atteindre un nœud destination et cela en diffusant par inondation un paquet *RREQ* (*Route REQuest*) dans tout le réseau. La maintenance des chemins permet de détecter et signaler les coupures de chemins provoqués éventuellement par la mobilité des nœuds (ou un affaiblissement de leur batterie). AODV n'utilise pas des mises à jour périodiques, les chemins sont découverts et maintenues selon les besoins.

Procédure de découverte des chemins Lorsqu'un nœud source veut transmettre un paquet vers un nœud destination, il vérifie tout d'abord sa table de routage, s'il y a un chemin préexistant vers ce nœud destination, alors il l'utilise pour envoyer le paquet.

Autrement, il initie la procédure de découverte de chemin en inondant un paquet de type *RREQ* dans le réseau. Cela peut arriver si le nœud destination n'est pas connue au préalable, ou si la durée de vie du chemin existant vers le nœud destination est expirée ou si ce chemin est devenu défaillant [19]. Un paquet *RREQ* émis d'un nœud source vers un nœud destination contient les champs suivants [19] :

- Adresse du nœud destination à atteindre
- Numéro de séquence du nœud destination
- Adresse du nœud source
- Numéro de séquence du nœud source
- Nombre de sauts du nœud source jusqu'au nœud courant
- Identificateur de diffusion (Broadcast ID)

Lorsqu'un nœud retransmet un paquet *RREQ*, il établit un chemin inverse de ce nœud courant vers le nœud source en enregistrant l'adresse du voisin duquel il a reçu la première copie du *RREQ*. AODV utilise la notion de numéro de séquence pour éviter les boucles de routage et pour assurer la sélection du récent chemin en supprimant les chemins invalides. Le numéro de séquence du nœud source est utilisé pour maintenir les informations récentes sur le chemin inverse vers le nœud source, et le numéro de séquence du nœud destination est utilisé pour maintenir les infor-

mations récentes sur le chemin vers le nœud destination. Si un nœud intermédiaire reçoit un paquet *RREQ*, et s'il y a un chemin vers le nœud destination demandé par le nœud source, alors il compare le numéro de séquence du nœud destination dans sa propre entrée de table de routage avec celui dans le paquet *RREQ*, si le numéro de séquence du nœud destination dans le paquet *RREQ* est plus grand que celui dans le nœud intermédiaire, alors le nœud intermédiaire n'utilise pas son chemin enregistré pour répondre au *RREQ*. Il met à jour l'entrée de numéro de séquence du nœud destination dans sa propre table de routage et il rediffuse ce paquet. Le nœud intermédiaire peut répondre seulement quand il a un chemin avec un numéro de séquence supérieur ou égal à celui du paquet *RREQ* [19]. Chaque entrée de table de routage a une durée de vie (Life time). Si un chemin n'est pas utilisé dans cette période, elle est considérée comme invalide.

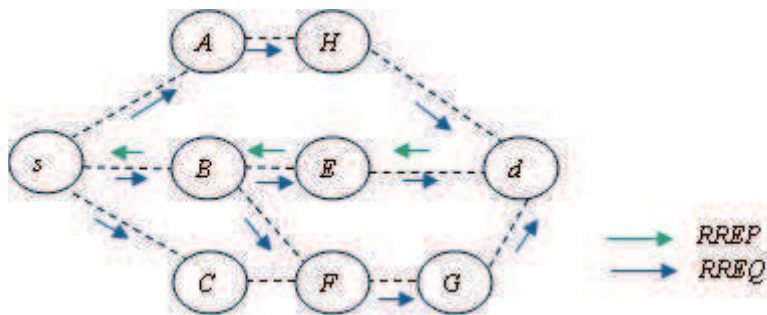
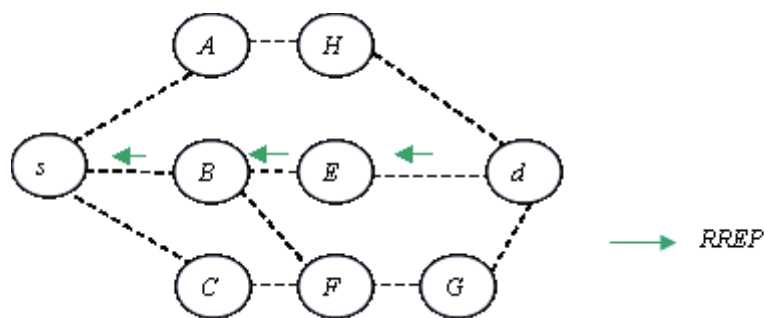


FIGURE 2.3 – Diffusion des paquets *RREQ* : création de chemin inverse.

La figure 2.3 montre la phase de recherche du nœud destination en diffusant le paquet *RREQ*. Une fois trouvé, le nœud destination répond à un nœud voisin pour valider un chemin inverse parmi les chemins découverts. Le champ Identificateur de diffusion (Broadcast ID) du paquet *RREQ* s'incrémente à chaque fois que le nœud source lance un nouveau paquet *RREQ* (initialisation de la procédure de découverte de chemin). Supposons qu'un nœud intermédiaire a reçu un paquet *RREQ*, s'il reçoit un autre paquet *RREQ* avec le même Broadcast ID et la même adresse source, il éliminera le paquet *RREQ* répété.

Lorsque le nœud destination est trouvé, il émet un paquet de réponse de chemin *RREP* (*Route REPLY*), qui parcourt le chemin en sens inverse de manière unicast, en modifiant les tables de routage des nœuds, par lesquels il passe jusqu'au nœud source (voir la figure 2.4). Un paquet *RREP* contient [19] :

- Adresse du nœud source
- Adresse du nœud destinataire
- Numéro de séquence du nœud destination
- Nombre de sauts du nœud destination jusqu'au nœud source
- Durée de vie du paquet

FIGURE 2.4 – Établissement de chemin entre une source s et une destination d .

Après la diffusion du paquet $RREQ$, le nœud source attend un paquet $RREP$, si ce dernier n'est pas reçu durant une certaine période, appelée $RREP_Wait\ Time$, le nœud source peut rediffuser un nouveau paquet $RREQ$ [19]. Une fois que le nœud source reçoit le $RREP$, il peut utiliser le chemin pour transmettre des paquets de données. AODV définit aussi un autre paquet ACK_RREP (*Route Reply Acknowledgment*) accusant la réception des paquets $RREP$.

Procédure de maintenance des chemins Dès qu'un chemin est établi entre un nœud source et un nœud destination, un mécanisme de maintenance est déclenché automatiquement. Ce mécanisme gère essentiellement les ruptures de liens entre les nœuds. Il est basé sur la transmission périodique des messages *Hello* [20]. Si trois messages *Hello* ne sont pas reçus consécutivement à partir d'un nœud voisin, le lien en question est considéré défaillant [19]. Lorsque un nœud détecte une rupture de lien dans un chemin actif, ce dernier envoie un paquet d'erreur $RERR$ (*Route ERROR*) au nœud qui le précède. Quand un nœud reçoit un paquet $RERR$ il recherche dans sa table de routage un chemin alternatif vers le nœud destination, sinon il consulte son nœud prédécesseur, ce mécanisme est répété jusqu'au nœud source, dans le cas échéant il lance une nouvelle découverte.

Avantages et inconvénients du protocole AODV Le protocole AODV a l'avantage de réduire le nombre de paquets de routage échangés, étant donné que les chemins sont créés à la demande. Il utilise un numéro de séquence pour éviter les boucles de routage et garder le chemin le plus frais.

Cependant, l'exécution du processus de découverte de chemin occasionne des délais importants avant la transmission de données. De plus, un problème supplémentaire se pose lors de changement de topologie : ce type de protocole réagit rapidement pour trouver une autre alternative au chemin endommagé. En conséquence, ces mécanismes de maintenance sont lourds à gérer et créent des sursauts de trafic de contrôle à chaque tentative de découverte ou de réparation de chemin. Ces mécanismes entraînent un délai d'attente et une mémorisation des paquets de données dû au défaut de chemin.

1.3 Protocoles de routage hybride

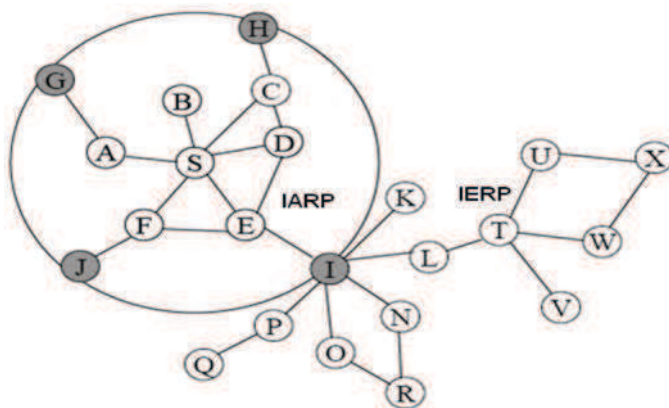
Les protocoles hybrides combinent les deux approches. Ils utilisent un protocole proactif, pour découvrir le proche voisinage (par exemple voisinage à deux ou trois sauts); ainsi ils disposent des chemins immédiats dans leur voisinage. Au-delà de cette zone prédéfinie, le protocole hybride fait appel aux techniques des protocoles réactifs pour chercher des chemins. Avec ce découpage, le réseau est partagé en plusieurs zones, et la découverte de chemins en mode réactif peut être améliorée. A la réception d'une requête de découverte réactive, un nœud peut indiquer immédiatement si le nœud destination est dans son voisinage ou non, et par conséquent savoir s'il faut orienter la requête vers les autres zones sans déranger le reste de sa zone. Ce type de protocole s'adapte bien aux grands réseaux, cependant, il accumule aussi les inconvénients des protocoles réactifs et proactifs : messages de contrôle périodiques, plus le coût de découverte d'un nouveau chemin [21]. Parmi les protocoles hybrides les plus connus, on présente ZRP [22].

1.3.1 Le protocole ZRP (Zone Routing Protocol)

ZRP [22] est un protocole hybride avec des composants proactif et réactif fonctionnant en cohésion. C'est une technique de la catégorie hybride cherchant à allier les avantages des approches proactives et réactives. ZRP introduit la notion de zone afin de délimiter l'utilisation de chaque type de protocole. Celles-ci sont définies par le nombre de sauts maximum que l'on autorise au delà duquel on estime que la requête n'est plus adressée à un nœud local. Il existe donc deux protocoles utilisés au sein de ZRP :

- IARP (Intrazone Routing Protocol) [23] est un protocole proactif à état de lien permettant à chaque nœud de connaître la distance qui le sépare de chaque autre nœud présent dans sa zone de routage. La découverte d'un chemin prend donc très peu de temps mais nécessite une utilisation régulière de la bande passante pour maintenir à jour la table de routage.
- IERP (Interzone Routing Protocol) [24] est un protocole réactif fonctionnant par inondation. Il s'agit donc de découvrir des chemins à la demande en évitant les boucles grâce à l'identifiant de requête. Comme tout protocole réactif, le délai de découverte du chemin par *RREQ/RREP* est globalement long mais réservé aux requêtes sortant d'une zone locale. Le protocole BRP (Bordercast Routing Protocol) [25], permet de minimiser les effets d'une inondation *RREQ* au sein d'une zone en exploitant les chemins connus par IARP dans la zone locale.

Lorsqu'un nœud implémentant ZRP veut envoyer un paquet de données vers un nœud destination, il commence sa recherche dans la zone à laquelle il appartient. S'il trouve le nœud destination, il peut l'atteindre immédiatement étant donné que cette entrée est maintenue dans sa table grâce au protocole proactif (IARP). Sinon, il envoie une requête de découverte de chemin aux nœuds se situant en périphérie de sa zone (nœuds périphériques), grâce au protocole réactif (IERP) en utilisant BRP pour l'acheminement de ces requêtes. Ainsi, les nœuds périphériques recevant la requête de découverte de chemin, recherchent dans leurs zones respectives et ainsi de suite jusqu'à atteindre le nœud destination. Voir la figure 2.5, les nœuds périphériques de la zone 1 sont : *I*, *H*, *G* et *J*.

FIGURE 2.5 – Zone de routage du nœud s (nombre de sauts = 2)

Avantages et inconvénients du protocole ZRP Ce protocole présente l'avantage de diminuer le nombre de messages de contrôle qui transitent sur le réseau comparé aux protocoles proactifs ou réactifs. De plus, il permet de diminuer le temps de latence pour la découverte de nouveaux chemins.

Cette hybridation cumule aussi quelques faiblesses des deux approches proactive et réactive (messages de contrôle périodique, le coût d'ouverture d'une nouvelle route...), surtout dans un réseau à forte mobilité. En plus dans ZRP il n'y a pas de coordination entre les nœuds, il en résulte que les zones se chevauchent, et un nœud peut être à la fois membre d'une zone et nœud frontière de plusieurs zones.

2 Comparaison entre routage proactif et réactif

Depuis que les recherches sur le routage dans les réseaux ad hoc se sont affînées, la supériorité d'une approche par rapport à l'autre a été débattue. Aucune d'elles n'a vraiment pris le dessus bien que la méthode réactive donne un résultat similaire voire meilleur que la méthode proactive dans la plupart des cas. Le délai d'acheminement d'un paquet dépend du temps de découverte du chemin, le temps de chaque saut et le nombre de sauts. Quand le trafic est léger, le temps d'attente et le temps d'accès au canal sont négligeables. Dans ce cas, les protocoles proactifs sont plus performants du fait qu'ils établissent un chemin optimal entre chaque paire de nœuds. Le routage réactif s'adapte naturellement à la diversité du trafic alors que le routage proactif est indépendant de la diversité du trafic. Un protocole proactif maintient les chemins vers toutes les destinations possibles, ce qui n'est pas nécessaire quand la majorité du trafic est destinée à une partie des nœuds, alors que les protocoles réactifs conservent uniquement les chemins en cours d'utilisation. Les protocoles réactifs réagissent seulement quand un lien est rompu et que cela affecte le chemin actuellement utilisé tandis que les protocoles proactifs réagissent même si le lien rompu n'affecte en rien le chemin utilisé. Les protocoles réactifs n'effectuent une mise-à-jour que quand un lien est rompu alors que les protocoles proactifs cherchent toujours le chemin optimal. Ceci dit, le fait de ne pas utiliser le chemin

optimal risque à la longue de congestionner le réseau [26].

3 Conclusion

Un réseau ad hoc est un réseau qui peut évoluer spontanément au gré de l'apparition, de la disparition, et des déplacements des unités mobiles qui le constituent. De ce fait le routage est l'un des plus importants problèmes dans de tels réseaux. Différentes approches sont utilisées dans la conception des algorithmes de routage des réseaux ad hoc. Nous avons présenté les trois principales classes des protocoles de routage qui sont : les protocoles proactifs qui tentent de maintenir à jour une représentation actuelle du réseau, et les protocoles de routage réactifs qui déterminent un chemin uniquement en cas de besoin. Il existe aussi les protocoles associant les deux approches, ce sont les protocoles de routage hybrides. Des tests de performances ont été effectués dont notamment des comparaisons entre des implémentations de ses trois approches au sein de réseaux de densité, mobilité et trafic variables. Une étude ayant mené la comparaison de ces trois approches constate que les performances des trois approches au sein d'un réseau à densité faible étaient relativement stables avec un trafic faible. Par ailleurs, les protocoles réactifs présentent des performances régulières en milieu dense [27]. En raison des caractéristiques des réseaux ad hoc (topologie dynamique, bande passante et batteries limitées...), les chemins reliant des nœuds source aux nœuds destination peuvent être très instables et rompus à tout moment. Un routage multichemin peut être une alternative au routage classique à chemin unique, pour remédier aux faiblesses du routage dans les réseaux ad hoc. Dans le prochain chapitre, nous nous intéressons en détail au routage multichemin avec un souci de conservation de l'énergie afin d'améliorer les performances des réseaux ad hoc.

Chapitre 3

Routage multichemin dans les réseaux ad hoc

Un réseau ad hoc est caractérisé par les changements fréquents de topologie du réseau, la limitation de la bande passante disponible, et la limitation de la puissance des batteries des nœuds. La topologie du réseau ad hoc change fréquemment car les nœuds dans un réseau ad hoc sont capables de se déplacer collectivement ou individuellement et de manière imprévisible. Ces caractéristiques rendent le routage dans ce type de réseau complexe. Le problème de routage a toujours suscité un vif intérêt dans la communauté des chercheurs. De nombreux travaux de recherche se sont intéressés au routage dans les réseaux ad hoc. Bien que les protocoles proposés présentent certaines caractéristiques pertinentes, ils présentent certaines limites, surtout si on considère une forte mobilité des nœuds ou une forte densité du réseau. Les chercheurs se sont intéressés plus particulièrement ces dernières années aux problèmes d'amélioration du routage ad hoc, ce qui a donné naissance à plusieurs mécanismes de routage. Parmi ces mécanismes, le multichemin (multipath) est utilisé par plusieurs protocoles pour éviter des retards inutiles lors des ruptures des liens.

Le concept du routage multichemin est de donner au nœud source le choix, à un moment donné, entre de multiples chemins pour atteindre une certaine destination. Les chemins multiples peuvent être utilisés alternativement ou de manière concurrente en fonction des critères choisis. Les stratégies de routage à chemin unique dans les réseaux ad hoc ne sont pas satisfaisantes pour répartir la charge du réseau et pour prémunir contre les ruptures de liens. Ainsi, pour maximiser le flux de données et obtenir une grande bande passante, les paquets des données d'un flux entre une source et une destination peuvent être répartis entre les multiples chemins [28]. Un autre avantage de protocoles de routage multichemin est la réduction du surcoût du routage. Plusieurs solutions de multichemin ont été développées [29-32], afin d'améliorer les performances des réseaux ad hoc.

1 Principe du routage multichemin

Le principe du routage à multichemin consiste à considérer non plus un seul chemin entre une source et une destination pour la transmission des paquets de données, mais plusieurs chemins (figure 3.1). Les nœuds s et d veulent échangés des paquets, les paquets doivent passer par les nœuds intermédiaires. Le routage à chemin unique choisit un chemin (ici, $s, 3, 6, 5, d$). Le routage multichemin choisit plusieurs chemins (ici, $s, 1, 6, 5, d$; $s, 2, 4, 8, d$; $s, 3, 7, 9, d...$). Si le nœud e épuise sa batterie ou s'il s'éloigne hors de contact du nœud a alors le chemin unique est rompu. Ce qui n'est pas le cas si on considère le multichemin.

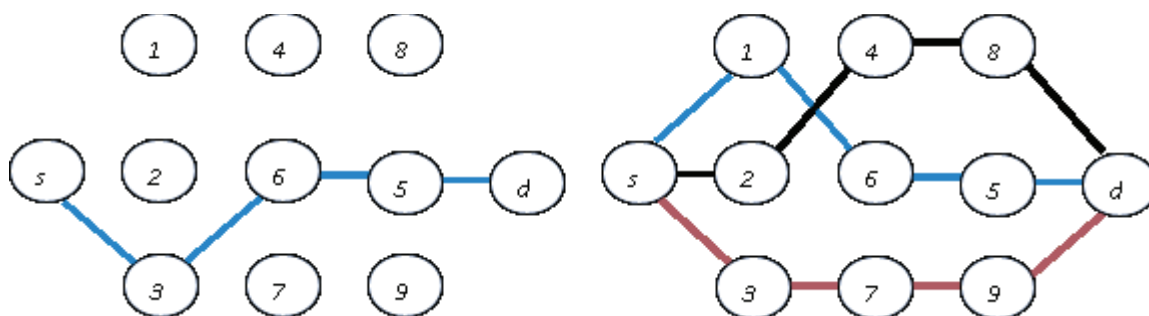


FIGURE 3.1 – Routage à chemin unique et routage multichemin.

Le routage multichemin est basé sur trois mécanismes : la découverte des chemins, la maintenance des chemins et la distribution du trafic [29].

- La découverte des chemins : c'est le processus de recherche des chemins (multiples) entre deux nœuds.
- La maintenance des chemins : c'est le processus de recherche des chemins en panne, de les réparer et de chercher un nouveau chemin valide.
- L'allocation de trafic : c'est de décider comment les paquets de données à transmettre vont être distribués sur les chemins disponibles entre une source et une destination dans le réseau.

1.1 Types de chemins

L'utilisation de chemins multiples diminue l'effet des ruptures des liens. Ceci contribue vivement à l'amélioration des performances du réseau. néanmoins le routage multichemin reste une technique plus difficile que le routage à chemin unique. Sa difficulté réside dans la recherche des (meilleurs) chemins multiples.

L'approche multichemin permet de calculer des chemins de manière distribuée et indépendante dans chaque nœud. Elle est basée sur le principe de disjonction des chemins (la disjonction peut être partielle [33]). Ce principe a pour but d'assurer l'indépendance des chemins du multichemin, c'est à dire si un lien d'un des chemins subi une rupture, cette rupture n'affectera pas les autres chemins du multichemin.

Notons qu'il y a deux types de disjonction de chemins : pour les liens ou pour les nœuds. Il existe trois approches dans le routage multichemin, selon l'utilisation des chemins soit : nœuds disjoints, liens disjoints ou partiellement disjoints [34] :

- **Nœuds disjoints** : Les chemins n'ont aucun nœud en commun, sauf la source et la destination (voir figure 3.2).

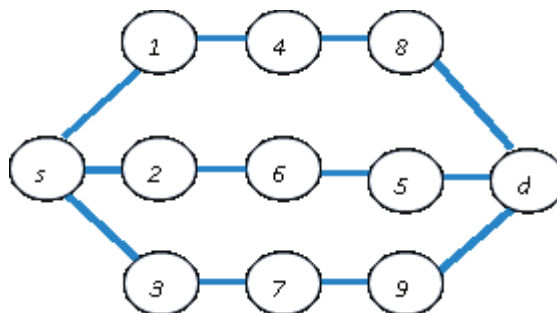


FIGURE 3.2 – Chemins à nœuds disjoints.

- **Liens disjoints** : Les chemins, en revanche, n'ont aucun lien en commun. Il peut, cependant, avoir un ou plusieurs nœuds communs (voir figure 3.3).

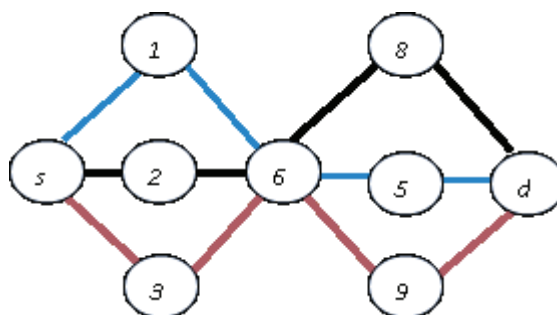


FIGURE 3.3 – Chemins à liens disjoints.

- **Partiellement disjoints** : Les chemins peuvent partager quelques nœuds et/ou liens.

Le terme disjoint peut en fait désigner deux situations : la disjonction par les liens (aucun lien commun) et celles par les nœuds (aucun nœud commun mis à part s et d). Chaque type de chemin a ses points forts et ses points faibles. En effet, l'avantage de la construction de chemins à nœuds disjoints est que lorsque les nœuds se déplacent, on garantit que la coupure des liens se fait d'une manière indépendante ce qui peut rendre plus simple et plus efficace la mise en œuvre du routage. Cependant, le contexte des nœuds disjoints est plus strict que celui des liens disjoints ce qui restreint le nombre de chemins disjoints.

2 Objectifs du routage multichemin

Le routage multichemin a été étudié pour répondre à différents objectifs, notamment la gestion du réseau. Dans ce qui suit, nous décrivons certaines caractéristiques du routage multichemin qui améliorent les performances des réseaux ad hoc.

2.1 Fiabilité

La fiabilité est un défi important dans les réseaux ad hoc parce qu'un paquet transmis peut être perdu facilement dû aux changements fréquents de la topologie du réseau, aux interférences, etc.

A l'origine, le routage multichemin dans les réseaux ad hoc a été développé, d'une part, pour assurer des chemins alternatifs en cas de rupture d'un chemin emprunté (C'est le cas des protocoles DSR [18], TORA [35]), d'autre part, des extensions multichemins ont été apportées sur des protocoles existants tels que AODV-BR [36], APR [37], et SMR [38]. Dans ces cas, les chemins multiples ne sont pas utilisés simultanément, le trafic prend un seul des chemins multiples à la fois. Le routage multichemin concurrent est développé pour améliorer la fiabilité et assurer un équilibrage de charge. Une technique a été proposée dans [39] avec l'ajout d'information de contrôle aux données transmises. Une approche intéressante [40], a combiné le transport multichemin (MPT) et le codage par descriptions multiples (MDC) afin d'envoyer des vidéos ou des images dans un réseau ad hoc. Les informations (descriptions) sont envoyées par de multiples chemins, ceci assure une qualité d'image acceptable, tandis que n'importe quelle description additionnelle, lorsqu'elle est cumulée, augmente la qualité d'image.

La transmission des descriptions multiples par l'intermédiaire de chemins multiples a deux avantages, elle aide à fournir une bande passante exigée par la transmission de vidéos ou images, et assure une transmission fiable de bout en bout.

2.2 Équilibrage de charge et la consommation d'énergie

Les nœuds dans les réseaux ad hoc dépendent fortement de leurs batteries qui ont une énergie limitée. La perte d'un nœud à cause du manque d'énergie affecte la topologie du réseau, son fonctionnement et sa durée de vie.

L'un des buts des protocoles de routage utilisant des chemins multiples, c'est de choisir le meilleur chemin de telle sorte que l'énergie consommée par le réseau soit réduite au minimum [41], de ce fait on maximisera la durée de vie du réseau. L'inconvénient de ces protocoles est que les nœuds choisis pour le chemin épuiseront rapidement leur énergie.

A cet effet, le routage multichemin a montré son efficacité dans la distribution de la charge parmi les nœuds proportionnellement à leur énergie résiduelle. Ainsi, lorsque la consommation d'énergie est équilibrée entre les nœuds, la durée de vie du réseau est prolongée considérablement [42-48].

2.3 Surcoût de routage

Un autre avantage du routage multichemin est la réduction du surcoût engendré par les paquets de contrôle et de leur traitement généré par le routage ("routing overhead"). Il existe trois classes de protocoles de routage : proactif, réactif et hybride. La majorité des études montre que les protocoles réactifs réduisent considérablement le surcoût du routage [44-45]. Ceci produit un impact positif sur la bande passante du réseau sans fil. En général trois types de paquets de routage sont employés :

demande d'un chemin (*RREQ : Route Request*), réponse de routage (*RREP : Route Reply*) et erreur de routage (*RERR : Route Error*).

Les protocoles réactifs multichemin trouvent les chemins multiples entre une source et une destination par une simple découverte de chemins. Une nouvelle découverte de chemins est lancée seulement quand tous les chemins trouvés lors d'une étape précédente tombent en panne.

Dans un cas du routage à chemin unique, quand un nœud n'arrive pas à transmettre un paquet au nœud prochain, un message d'erreur de routage est renvoyé à la source indiquant la rupture du chemin. Lorsque de multiples chemins disjoints sont disponibles, les nœuds peuvent activement retransmettre le paquet en l'envoyant par un autre chemin. Dans ce cas le message d'erreur de routage est émis seulement quand tous les chemins disponibles échouent, ainsi le nombre total de messages de routage échangés est considérablement réduit. Des résultats similaires ont été rapportés dans [46].

2.4 Qualité de service (QoS)

Un objectif important du routage multichemin est de fournir la qualité du service, plus exactement, de réduire les latences, d'éviter la congestion, etc. Des travaux [46] ont montré que le routage multichemin contribue de manière significative à la QoS. Il existe deux types de latences provoquées par les protocoles de routage des réseaux ad hoc. Le premier lors de la recherche d'un chemin. Ce type de latence est dû aux messages de demande de chemins du protocole de routage. Le routage multichemin réduit efficacement la fréquence de recherche de chemins, ainsi la latence induite sera réduite.

Le deuxième type de latence est dû aux ruptures des chemins. Dans ces cas, trois tranches de temps sont additionnés : le temps de transmission du paquet de données jusqu'au dernier nœud avant le point de rupture, le temps de la détection de la rupture du chemin et le temps d'envoi du message d'erreur de routage à la source. Encore une fois, le routage multichemin réduit l'occurrence des erreurs de chemins donc la latence dû aux ruptures de chemin est réduite.

Parmi les problèmes des réseaux ad hoc, le partage des canaux radios rendent les chemins dépendants (à cause des interférences), malgré la disjonction des nœuds [37]. Le choix des chemins multiples qui causent moins de dépendance est donc un défi important dans la conception des protocoles de routage multichemin.

2.5 Sécurité

Le routage multichemin apporte un plus à la sécurité dans les réseaux ad hoc. Il permet de combiner des codes cryptographiques aux données transmises par des chemins multiples. Une approche [47-48] est basée sur le principe de clef probabiliste partagée. À l'aide du partage probabiliste des clefs, chaque nœud dans le réseau sera préchargé avec un certain nombre de clefs initiales. Chaque paire de nœuds du réseau partagera une ou plusieurs clefs communes. Le nœud source produit une clef secrète et la divise en parties, ces parties sont transmises à la destination en utilisant le routage multichemin. Le multichemin dans ce cas est logique, les parties peuvent

emprunter les mêmes chemins physiques sauf qu'elles sont chiffrées par des clefs communes différentes.

3 Description des principaux protocoles de routage multichemin dans les réseaux ad hoc

Un protocole multichemin ne peut donner satisfaction à tous les objectifs cités ci-dessus. Chaque protocole est conçu pour remplir certains des objectifs, selon leurs spécifications propres (par exemple au niveau de la gestion de ressources et l'indépendance des chemins). La plupart des protocoles offre une transmission fiable, une charge du réseau réduite, un minimum de stockage et de consommation des ressources et des chemins optimaux. La plupart des protocoles de routage multichemin proposés dans la littérature sont des extensions de protocoles existant tels que AODV et OLSR (voir chapitre précédent). Dans ce qui suit, nous décrirons leurs versions multichemin à savoir : AOMDV, AODVM et MP-OLSR.

3.1 Le protocole de routage multichemin AOMDV

AOMDV (Ad hoc On demand Multipath Distance Vector) [20] est une extension du protocole AODV qui permet la découverte de plusieurs chemins sans boucle et à liens disjoints. AOMDV est plus performant que AODV en terme surtout de routage et fiabilité. Pour garder la trace de plusieurs chemins, à chaque destination est associée une liste de prochains nœuds avec la distance (exprimée en nombre de sauts) pour atteindre la destination en passant par ce nœud. Pour chaque destination, chaque nœud maintient le "nombre de sauts à annoncer" qui est défini comme la distance minimale de tous les chemins qu'il maintient vers chaque destination. Pour éviter la formation des boucles, un nœud n'accepte que les chemins alternatifs (définis par des *RREQ* ou des *RREP*) dont le nombre de sauts est inférieur à celui annoncé.

Dans AOMDV, les chemins multiples sont formées pendant la phase de découverte du chemin, pour cela les auteurs définissent un nouveau concept, qui est le "*Advertised_hopcount*" remplaçant le paramètre "*hopcount*" du protocole de base AODV, ce paramètre représente le maximum "*hopcount*" des chemins disponibles d'un nœud i vers la destination d . Dans une tentative d'obtenir des chemins disjoints en lien, le nœud de destination répond avec un *RREP* aux requêtes arrivant depuis des voisins distincts. Chaque *RREP* prend un chemin inverse, les chemins de ces *RREP* peuvent partager certains des nœuds intermédiaires mais chaque *RREP* prend un chemin différent vers la source. Ceci assure que les chemins seront disjoints en lien.

3.1.1 Découverte de chemin

Dans AODV, chaque requête reçue par un nœud pendant le processus de découverte de chemin définit potentiellement un chemin alternatif vers la source. Par

exemple, chaque copie du paquet de requête *RREQ* arrivant à un nœud définit un chemin alternatif vers la source. Cependant, accepter naïvement toutes ces copies va mener à la formation de boucles dans les chemins. Afin d'éliminer la possibilité de formation de boucles, AOMDV utilise la même condition que pour AODV. Cette condition consiste à n'accepter une nouvelle requête que si le nombre de sauts est inférieur au nombre de sauts courant, et que le numéro de séquence est supérieur au numéro de séquence courant.

AOMDV permet de construire des chemins multiples à liens disjoints, c'est-à-dire des chemins multiples qui n'ont pas des liens en commun entre les différents chemins qui mènent d'une source vers une destination bien précise [2].

AOMDV ajoute un nouveau champ appelé *firsthop* pour chaque paquet *RREQ*. Ce champ indique le premier saut qui l'a acheminé. En plus, chaque nœud maintient une liste, *firsthop_list*, pour chaque *RREQ* afin de garder une trace de la liste des voisins de la source à partir desquels une copie de *RREQ* a été reçue.

Les nœuds intermédiaires, doivent garder les copies dupliquées de *RREQ* contrairement à AODV. Chaque copie est examinée pour voir si elle procure un nouveau chemin disjoints vers la source, cette vérification est assurée grâce au champ *firsthop* du paquet *RREQ* et la liste *firsthop_list* du nœud. Si on s'assure que *RREQ* apporte un nouveau chemin, l'AOMDV invoque la règle de mise à jour pour vérifier si le chemin inverse peut être mis en place ou non. Si le chemin inverse peut être mise en place et le chemin vers la destination est valide, le nœud intermédiaire envoie un *RREP* vers la source.

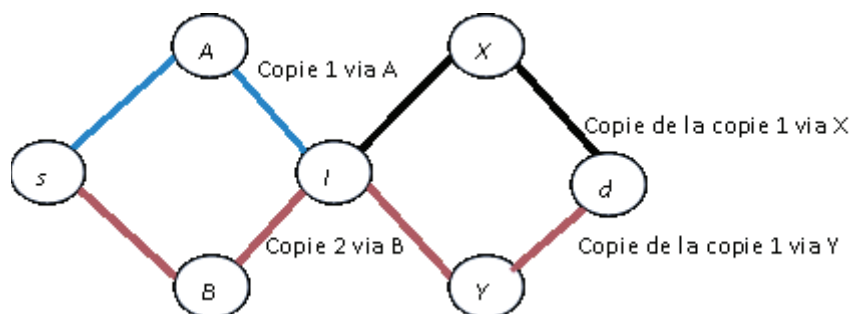


FIGURE 3.4 – Découverte de chemins à liens disjoints dans AOMDV

La figure 3.4 illustre l'utilité supplémentaire de *RREPs*, concernant la deuxième copie de *RREQ* envoyé *B* et non ré-envoyé par le nœud intermédiaire. Les deux copies de la première copie (via *A*) atteignent la destination *d*. La destination répond à tous les deux. Les deux réponses vont fusionner à *I* qui les transmettra le long de deux chemins disjoints (par *A* et *B*). Ainsi *s* obtiendra deux chemins à liens disjoints jusqu'à la destination *d*.

Comme pour les nœuds intermédiaires, la destination doit vérifier que les chemins inverses peuvent être mises en place. Elle doit garantir que les liens sont disjoints, uniquement avec ces voisins. Au-delà du premier saut, le *RREP* suit les chemins inverses qui ont été mises en place et qui sont déjà formées. Chaque *RREP* arrivant à un nœud intermédiaire peut suivre plus qu'un chemin inverse lorsqu'une multitude de chemins sont déjà disponibles. La destination répond à *k* copies de *RREQ*. Le paramètre *k* est utilisé pour contrôler le nombre de *RREP* émis par la destination.

3.1.2 Constructions des chemins multiples

Afin éliminer les éventuelles boucles de routage, AOMDV maintient la même règle de AODV. AOMDV est basé sur le principe du "*Advertised hopcount*" [20]. L'*advertised hopcount* d'un nœud i pour une destination d représente le *hopcount*

<i>Destination</i>
<i>Seq_num</i>
<i>Advertised_hopcount</i>
<i>Route_list</i> {(<i>nexthop1</i> , <i>hopcount1</i>), (<i>nexthop2</i> , <i>hopcount2</i>), ... }
<i>Expiration timeout</i>

Entrée de la table de routage d'AOMDV

<i>Destination</i>
<i>Seq_num</i>
<i>Hopcount</i>
<i>Nexthop</i>
<i>Expiration timeout</i>

Entrée de la table de routage d'AODV

FIGURE 3.5 – Structure des entrées des tables de routage de AODV et AOMDV.

maximum des chemins multiples disponibles pour i vers la destination d . Le *hopcount* maximum est considéré comme le nombre de sauts qui ne change jamais pour le même numéro de séquence. Le protocole permet d'accepter seulement les chemins alternatifs ayant un *hopcount* inférieur à l'*advertised hopcount*. Cette condition est nécessaire pour garantir des chemins sans boucle de routage. La figure 3.5, représente la structure des tables de routage du protocole multichemin AOMDV et celui à chemin unique AODV.

L'*advertised_hopcount* de AOMDV remplace le *hopcount* de AODV. L'entrée *Route_list* remplace *nexthop* et définit essentiellement les sauts multiples suivants le *hopcount*.

Dans AOMDV, un nœud i invoque les règles décrites dans l'algorithme 1, chaque fois qu'il reçoit une demande de découverte d'un chemin pour une destination d depuis un voisin j . Les variables $Seq_num_i^d$, $Advertised_hopcount_i^d$, et $Route_list_i^d$ représentent respectivement le numéro de séquence du nœud i , le nombre maximum de sauts du nœud i vers d , et la liste des chemins depuis le nœud i vers la destination d .

3.1.3 Maintenance des chemins

Le protocole AOMDV utilise le paquet *RERR* (Route ERRor) pour signaler les ruptures des liens. Lorsqu'un nœud détecte une rupture d'un lien, il envoie un paquet *RERR* énumérant les nœuds voisins perdus. Le nœud envoie le *RERR* en amont vers le nœud source. Si il ya plusieurs nœuds précédents qui utilisent ce lien, le nœud diffuse le paquet *RERR* à tous ces nœuds. Ces chemins seront effacés des tables de routage des nœuds récepteurs de ce paquet d'erreur. Une fois la source reçoit la *RERR*, elle ré-initie la découverte des chemins uniquement lorsqu'aucun des chemins menant à la destination précédemment établis n'est valide.

Seq_num = numéro de séquence de la destination d au nœud i .

$Advertised_hopcount_i^d$ = le maximum de nombre de sauts.

$Route_list_i^d$ = liste des routes depuis un nœud i vers la destination d .

Begin

If ($seqnum_i^d < seqnum_j^d$) **then**

$seqnum_i^d \leftarrow seqnum_j^d$

If ($i \neq d$) **then**

$Advertised_hopcount_i^d \leftarrow \infty$;

$route_list_i^d \leftarrow NULL$;

 insert (j , $advertised_hopcount_j^d + 1$) into $route_list_i^d$;

else

$advertised_hopcount_i^d \leftarrow 0$;

end If

else

If ($(seqnum_i^d = seqnum_j^d)$ and

$((advertised_hopcount_i^d) > (advertised_hopcount_j^d))$) **then**

 insert (j , $advertised_hopcount_j^d + 1$) into $route_list_i^d$;

end If

end If

End.

Algorithm 1: Algorithme de découverte des chemins d'AOMDV.

3.1.4 Critiques du protocole d'AOMDV

Dans AOMDV lors d'une rupture de liens au niveau du chemin, à travers lequel on transmet les données, la source change ce chemin par un des chemins alternatifs au lieu de déclencher une nouvelle phase de découverte des chemins. Ainsi une nouvelle phase de découverte des chemins n'est déclenchée que lorsque tous les chemins multiples alternatifs sont rompus. Par contre, pendant une même phase de découverte des chemins, plusieurs chemins qui mènent entre une source et une destination sont établies mais uniquement le meilleur chemin en termes de nombre de sauts est sélectionné pour la transmission des données entre une source et une destination. Le

second plus court chemin en termes de nombre de sauts parmi les autres chemins ne sera utilisé que lorsque le premier chemin sélectionnée sera invalidé. En effet, le calcul et le maintien de plusieurs chemins entre une source et une destination induit une occupation plus importante de la table de routage, consomme de la mémoire du nœud et accroît la taille des entêtes des paquets de contrôle alors qu'on ne bénéficie que d'un seul chemin pour la transmission ce qui constitue un handicap dans notre contexte. Un autre inconvénient de AOMDV est qu'il utilise des chemins à liens disjoints, la défaillance d'un nœud commun entre plusieurs chemins provoque la rupture de tous les chemins passant par ce nœud.

3.2 Ad hoc on Demand Distance Vector Multipath : AODVM

AODVM [32] est la première version modifiée du protocole AODV qui est capable de construire de multiples chemins à nœuds disjoints entre une source et une destination. Les procédées de calcul de chemins avec des nœuds disjoints font l'objet de plusieurs articles, notamment grâce à la propriété d'indépendance fournie par ce type de chemins. AODVM est dit plus fiable et obtient une meilleure performance par rapport à AODV. AODVM est un protocole de routage multichemin qui a comme objectif de trouver des chemins avec des nœuds disjoints. Il est basé sur AODV. La diffusion du paquet *RREQ* suit la même règle que l'AODV. Chaque nœud maintient une table contenant tous les voisins ainsi que le coûts des chemins correspondant (en nombre de sauts) jusqu'à la source (voir la figure 3.6).

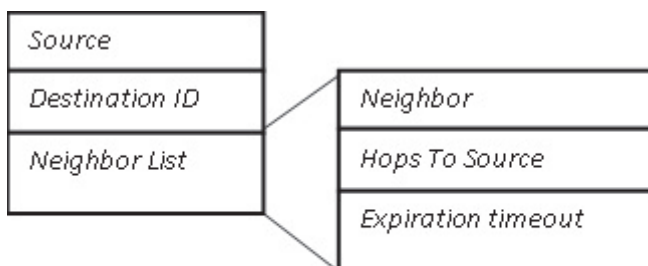


FIGURE 3.6 – Le format du paquet *RREQ* du protocole AODVM.

Quand un *RREQ* atteint la destination, un paquet de *RREP* est produit et renvoyé au nœud amont duquel la destination a reçu le *RREQ*. Le paquet de *RREP* contient un champ "*Last_hop_ID*" indiquant l'identificateur du nœud amont (le voisin à partir duquel la copie de *RREQ* a été reçu). Le paquet de *RREP* peut ne pas suivre le chemin inverse : c'est-à-dire le chemin suivi par paquet *RREQ* qui l'a généré. À la place, un nœud intermédiaire détermine à quel nœud amont le *RREP* devrait être envoyé en se basant sur l'information enregistrée dans sa table de *RREQ*. Quand un nœud intermédiaire reçoit un *RREP*, il trouve dans son paquet *RREQ* un chemin menant à la source et envoie le *RREP* au prochain nœud correspondant. Lorsque un nœud reçoit le premier paquet *RREP* vers une destination donnée, il crée une entrée du chemin vers le nœud d'où il a reçu le paquet *RREP*, et met à

jour le champ `Route_list` de sa table de routage. Chaque nœud maintient un champ `Route_list` ayant une structure décrite dans la figure 3.7.

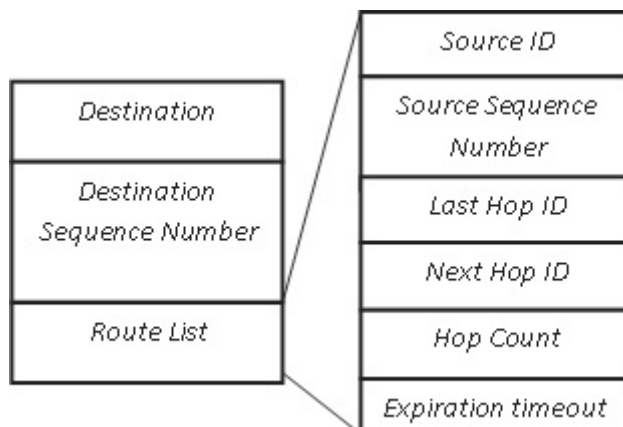


FIGURE 3.7 – Une entrée de la table de routage d’AODVM.

Dans AODVM, afin de s’assurer qu’un même nœud ne participe pas aux différents chemins d’un chemin multiple, quand un nœud détecte la transmission par son voisin d’un paquet de *RREP*, le nœud supprime l’entrée correspondant de ce voisin de sa table de *RREQ*, afin d’éviter d’utiliser ce voisin pour un *RREP*. Si un nœud intermédiaire reçoit un *RREP* et ne trouve aucun voisin à qui l’expédier (c.-à-d. sa table de *RREQ* est vide), il produit un paquet de *RDER* (Route Discovery Error) et l’envoie de nouveau au nœud duquel il a reçu le *RREP*. Le voisin, lors de réception du *RDER* essaiera d’expédier le *RREP* à un voisin différent. Reste à noter que le nombre des *RDER* peut être limité.

3.2.1 Critiques du protocole AODVM

Le protocole AODVM, est un protocole multichemin à nœuds disjoints. La défaillance d’un nœud dans un chemin donné, n’a aucune influence sur les autres chemins découverts. Sauf que, comme tout protocole multichemin à nœuds disjoints, il y aura moins de chemins dans le processus de découverte comparativement aux protocoles à liens disjoints. Naturellement ceci mènera à moins de surcoûts dans un réseau. Mais AODVM a un surcoût plus élevé que d’autres protocoles multichemins, puisqu’AODVM envoie beaucoup de paquets lors de son procédé de découverte de chemins [49].

3.3 Le protocole MP-OLSR (MultiPath Optimized Link State Routing)

Le protocole MP-OLSR [50] est un protocole de routage multichemin, basé sur OLSR [12]. Une autre variante multichemin du routage OLSR a été adaptée aux réseaux de capteurs dans [51]. Les deux solutions multichemins sont similaires à

OLSR dans les premières phases, dans la mesure où elles utilisent l'envoi périodique et proactif de paquets *HELLO* afin de mettre à jour les voisinages et les relais multipoints, ainsi que les paquets TC (Topology Control) afin de transmettre l'information de voisinage au reste du réseau. Ces derniers paquets permettent à chaque nœud de maintenir une vision globale sur les informations de topologie dans le réseau à chaque instant. Ces solutions diffèrent d'OLSR au niveau de la méthode de calcul des chemins et au niveau du routage.

3.3.1 Méthode de calcul des chemins dans MP-OLSR

A la différence d'OLSR, MP-OLSR [50] ne maintient pas les tables de routage en permanence mais il calcule les chemins multiples en présence de données à transmettre vers une destination donnée. MP-OLSR s'apparente donc à un algorithme de recherche réactif : l'information de topologie est bien stockée en permanence mais elle ne sera exploitée que si des chemins sont effectivement requis. MP-OLSR attribue des poids à chaque lien. Les poids sont ensuite envoyés au reste du réseau par les paquets TC. Ainsi, la table de topologie de chaque nœud dispose d'attributs supplémentaires relatifs au poids des liens. Deux poids sont utilisés : un poids correspondant au coût réel reçu dans le paquet TC et un poids correspondant au coût virtuel qui représente un coût évoluant au cours de l'algorithme de sélection de chemins. Puisque le routage dans MP-OLSR se base sur la métrique du nombre de sauts, le poids initial d'un lien vaut 1. L'algorithme de calcul des chemins multiples utilise l'algorithme de Dijkstra, noté par MP-Dijkstra et est donné dans l'algorithme 2.

L'algorithme proposé est appliqué à un graphe G ; deux noeuds s et d et un entier strictement positif N . Il fournit un N -uplet (P_1, P_2, \dots, P_N) chemins de $(s$ à $d)$. $Dijkstra(G, n)$ est un algorithme standard de Dijkstra, qui fournit l'arbre source des plus courts chemins de nœud n dans le graphe G ; $GetPath(SourceTree, n)$ est la fonction qui extrait le plus court chemin jusqu'à n à partir de l'arbre source $SourceTree$; $Reverse(e)$ donne l'arête inverse de e ; $Head(e)$ fournit l'arête du nœud à laquelle des points de e .

Le principe général de cet algorithme consiste en la recherche, à l'itération i , du plus court chemin P_i pour une destination d . Les coûts des liens du chemin P_i augmentent afin d'éviter le plus possible la sélection d'un chemin similaire dans la prochaine itération. Les fonctions incrémentales f_p et f_e de l'algorithme sont utilisées à chaque itération pour obtenir des chemins disjoints entre la source s et la destination d . f_p est utilisée pour augmenter les coûts des liens de l'ancien chemin P_i . Ceci permet de sélectionner des prochains chemins qui auront tendance à utiliser des liens différents. f_e est utilisé pour augmenter les coûts des liens qui mènent aux sommets du chemin précédent P_i . Il existe trois réglages possibles :

1. si $id = f_e < f_p$ alors les chemins ont tendance à être à liens disjoints,
2. si $id \leq f_e = f_p$ alors les chemins ont tendance à être à noeuds disjoints,
3. si $id < f_e < f_p$ alors les chemins ont tendance à être à noeuds disjoints, mais quand ce n'est pas possible, les chemins ont tendance à être à liens disjoints.

Avec id représentant la fonction identité. Ces fonctions de coût assurent la diversité des chemins multiples recherchés dans une topologie réseau.

```

MultiPathDijkstra(s, d, G, N) Begin

c1 ← c
G1 ← G

For i from 1 to n do
    sourceTreei ← Dijkstra(Gi, s)
    Pi ← GetPath(SourceTreei, d)
    If (e or Reverse(e) ∈ Pi) then
        | ci+1(e) ← fp(ci(e))
    else
        If (Head(e) ∈ Pi) then
            | ci+1(e) ← fe(ci(e))
        else
            | ci+1(e) ← ci(e)
        end If
    end If
end For
Gi+1 ← (V, E, ci+1)

Return (P1, P2, ..., PN)

End.
    
```

Algorithm 2: L'algorithme MP-Dijkstra.

Routage à la source : MP-OLSR accorde un rôle important au nœud source, ce qui n'est pas le cas dans OLSR. En effet, afin de garantir que le choix des chemins se fait de manière disjointe, ceux-ci sont déterminés par un seul et même nœud, à savoir la source. À l'inverse, dans OLSR, tout nœud (la source ou un nœud intermédiaire quelconque) ne se préoccupe que du prochain saut du paquet pour atteindre une destination donnée. Dans MP-OLSR, il faut en conséquence ajouter des entêtes spécifiques au paquet de données, contenant le chemin à parcourir. Une fois le paquet de données envoyé sur le réseau, chaque nœud intermédiaire lit l'entête en question et détermine quel est le prochain nœud destiné à recevoir le paquet.

3.3.2 Critiques du protocole MP-OLSR

Le protocole MP-OLSR améliore le taux de livraison des paquets de données (Packet Delivery Ratio PDR), de messages de contrôle générés (le surcoût c'est-à-dire l'"*overhead*"), et le délai de bout en bout des paquets de données par rapport à OLSR. La faiblesse du protocole MP-OLSR est qu'il manque de réactivité face aux ruptures des liens des chemins découverts. Un temps considérable s'écoule avant la détection de la rupture.

4 Caractéristiques des protocoles de routage multichemin

Tous les travaux cités ci-dessus ont montrés que l'approche multichemin dans les réseaux ad hoc améliore les performances des protocoles de routage. Le routage multichemin a pour but de construire plusieurs chemins entre une source et une destination. La construction des chemins se fait lors du processus de découverte des chemins lancé par le nœud source, ce processus doit éviter les boucles. Pour cela, plusieurs solutions sont proposées selon le type de routage appliqué. Les protocoles réactifs tel que AOMDV et AODVM utilisent un numéro de séquence (le même que celui utilisé par AODV : voir chapitre 2). Les protocoles proactifs utilisant le routage à la source, la séquence des nœuds du chemin est enregistrée dans l'entête de chaque paquet de données. Ainsi ces protocoles peuvent détecter et éviter les boucles avant l'utilisation de ces chemins. On peut citer MP-OLSR comme exemple.

Un autre critère important à tenir compte lors de la phase de découverte des chemins est d'établir des chemins disjoints. Car le partage des ressources entre plusieurs chemins maximisera le problème des collisions la surexploitation des ressources communes. Ce qui dégrade les performances du réseau. Deux types de chemins peuvent être mises en place dans les protocoles de routage à chemins multiples : à liens disjoints et à nœuds disjoints. Les chemins à nœuds disjoints d'un multichemin sont plus robustes vu que les chemins sont totalement indépendants. Donc la panne d'un chemin n'entraîne pas la panne d'un autre chemin du même multichemin. Dans la littérature, la disjonction des chemins est défini par rapport à la topologie du réseau et non par rapport à la zone radio de l'interface sans fil du nœud. Pour acheminer le trafic, les chemins d'un même multichemin peuvent être alternativement ou simultanément, selon le type de trafic et les besoins exigés. Les protocoles AOMDV et AODVM utilisent les chemins alternativement, à l'opposé de MP-OLSR qui les utilisent simultanément.

De nombreuses métriques de sélection d'un chemin peuvent être utilisées : le nombre de sauts d'un chemin, le délai de chaque chemin, la qualité de service du chemin... D'autres métriques comme, par exemple, la robustesse aux interférences, la qualité d'un lien ou l'énergie résiduelle des batteries des nœuds semblent tout aussi intéressantes à exploiter. Concevoir un protocole de routage avec une métrique de sélection adéquate est important pour la fiabilité du protocole selon les critères de performance exigés. Nous donnerons dans ce qui suit les critères de performance les plus connus.

5 Critères de performances

Parmi les critères à en tenir en compte dans la conception d'un protocole de routage multichemin dans les réseaux ad hoc sont [2] :

- **Robustesse** : le protocole doit être assez robuste pour supporter la mobilité des nœuds et offrir un large taux de livraison de paquets.
- **Efficacité** : le routage efficace utilise un nombre de paquets de contrôle relativement faible par rapport au nombre de paquets de données correctement reçus par les récepteurs.
- **Charge de contrôle** : le nombre de paquets de contrôle par unité de temps qui sont transmis pour le maintien des chemins multiples doit être le plus faible possible ce qui évite la consommation de la bande passante.
- **Qualité de service** : le routage multichemin permet de fournir la qualité du service, plus exactement, de réduire les délais, d'éviter la congestion, etc. Plusieurs travaux ont montré que le routage multichemin contribue de manière significative à la QoS, en réduisant le délai de transmission des paquets de bout en bout.
- **Gestion de ressources** : comme les nœuds ad hoc sont soumis à une contrainte de ressources limitées, le protocole multichemin doit conserver l'énergie, minimiser la bande passante, réduire le nombre de transmission de paquets et réduire le stockage des informations.
- **Adaptabilité** : le protocole doit être capable de s'adapter à un grand nombre de nœuds et à une forte charge du réseau.
- **Indépendance des chemins multiples** : le protocole doit être capable de fournir des chemins multiples indépendants pour minimiser le problème des ruptures des liens et des collisions.

6 Justification de notre choix de la métrique de routage

La sélection des chemins multiples dépend de la métrique choisie. Cette métrique doit être adaptée non seulement au contexte de l'application mais aussi à celui du réseau. La majorité des protocoles de routage, présentée précédemment, cherche essentiellement à trouver des chemins possibles entre une source et une destination sans prendre en considération la qualité des liens. Plusieurs métriques doivent être considérées pour le routage ad hoc et ces métriques ont un impact sur les performances [52].

Dans les réseaux ad hoc, chaque nœud dispose d'une batterie d'alimentation électrique et d'un approvisionnement énergétique limité. Au fil du temps, les nœuds épuisent leur réserve d'énergie et sont finalement retirés du réseau, ce qui limite la capacité d'acheminement de paquets de tout le réseau. Certains nœuds épuisés peuvent être critiques pour la transmission de paquets s'ils font partie du seul chemin actif reliant le nœud source au nœud destination. Pour résoudre ce problème, il est utile de trouver des chemins multiples sélectionnés sur la base de l'énergie résiduelle de leurs nœuds. Notre contribution dans cette thèse est de développer des

protocoles multichemins utilisant l'énergie comme métrique de sélection, ceci permet de mieux conserver l'énergie et donc maximiser la durée de vie du réseau et accroître ses performances.

7 Conclusion

Dans ce chapitre, nous avons présenté les principes de base d'un routage multichemin dans les réseaux ad hoc, ainsi que ses principaux objectifs, on cite : la fiabilité, l'équilibrage de charge, réduction du surcout généré des paquets de contrôle, etc. Les chemins découverts par un protocole de routage multichemin doivent être disjoints si on veut qu'ils bénéficient d'une certaine indépendance vis-à-vis des pannes. Cette disjonction peut être totale ou partielle. Nous avons dégagé trois approches de disjonction : chemins à nœuds disjoints, à liens disjoints ou à disjonction partielle. Puis, nous avons vu les principales solutions de routage multichemin dans les réseaux ad hoc présentes dans la littérature. Nous avons étudié les techniques de routage selon leur principe de découverte de chemins à savoir AOMDV, AODVM et PM-OLSR. L'étude réalisée nous a permis de dégager les besoins exprimés par ces techniques et de faire une comparaison entre celles-ci.

La majorité des protocoles de routage existants se base sur le critère du nombre de sauts pour sélectionner les chemins. Les autres critères possibles se basent principalement sur la qualité des liens suivant un besoin applicatif. En fin nous avons justifié notre choix de l'énergie comme métrique de sélection des chemins. Une bonne gestion de l'énergie permet d'éviter sa consommation inutile ce qui prolongera la durée de vie du réseau et améliorera ses performances. Nous présenterons dans le prochain chapitre, les principaux protocoles multichemins réactifs ayant comme métrique l'énergie des nœuds dans un réseau ad hoc. On s'intéresse au routage réactif vu sa performance prouvée dans les réseaux sans fils, comparativement à la classe proactive.

Chapitre 4

Routage multichemin avec conservation d'énergie dans les réseaux ad hoc

L'objectif pour un réseau ad hoc est que les nœuds mobiles soient opérationnelles au maximum n'importe où et n'importe quand. Cependant, l'une des grandes limitations de cet objectif concerne le support énergétique. En effet, la principale contrainte dans les communications sans fil est la durée de vie limitée des nœuds mobiles dont le support énergétique est souvent une batterie électrique dont la capacité est limitée. Cette contrainte est beaucoup plus importante dans les réseaux ad hoc, où les nœuds mobiles consomment leur propre énergie en routant des paquets de données pour d'autres nœuds mobiles. Dans les réseaux ad hoc, les ressources sont limitées en énergie et la consommation d'énergie est un facteur primordial pour la durée de vie du réseau et des nœuds mobiles.

Dans ce chapitre, nous allons présenter la contrainte d'énergie dans le routage multichemin, avec la description des principaux protocoles de routage de la littérature qui se basent sur la contrainte d'énergie. Ces protocoles ont pour principal objectif d'améliorer les performances d'un réseau ad hoc en prolongeant la durée de vie du réseau aussi longtemps possible.

1 Consommation d'énergie dans les réseaux ad hoc

La consommation d'énergie [53] est un critère primordial dans la conception des protocoles de routage pour les réseaux ad hoc ; les nœuds mobiles opèrent grâce à des batteries électriques dont la capacité est limitée. En outre, dans des environnements critiques (par exemple : champs de batailles, zones sinistrées, etc.) remplacer ou recharger les batteries est souvent impossible. Dans les réseaux ad hoc, les nœuds

mobiles utilisent généralement des équipements de stockage d'énergie autonomes pour se procurer de l'énergie et par conséquent ont une durée de vie limitée.

En réalité, dans les réseaux ad hoc l'épuisement de l'énergie d'un nœud n'affecte pas uniquement sa capacité à recevoir ou émettre, mais également sa capacité à acheminer les données pour les autres nœuds ce qui peut diminuer les performances du réseau ou carrément isoler certains segments du réseau.

Actuellement, il existe trois axes de recherche dans le domaine de conservation de la consommation d'énergie spécifiques aux réseaux ad hoc :

- L'économie d'énergie qui minimise la dissipation d'énergie lorsqu'un nœud mobile est inutilisée. On introduit un état de veille, plus économe en énergie que l'état actif et on tente de maximiser la durée que passe les nœuds mobiles en veille.
- Le contrôle de la puissance de transmission qui consiste à maintenir la capacité du réseau et à acheminer le trafic de données avec un coût énergétique minimal. On permet aux nœuds de déterminer la puissance de transmission minimale suffisante pour maintenir la connectivité du réseau.
- La distribution de la charge d'acheminement des données dont l'objectif principal est d'équilibrer la consommation d'énergie entre les nœuds mobiles.

Bien entendu, les trois approches ne sont pas exclusives, d'ailleurs certains protocoles de routage à basse consommation d'énergie combinent ces différentes approches.

2 Nœud mobile et consommation d'énergie

Un nœud mobile possède typiquement plusieurs composants matériels qui consomment de l'énergie, à savoir : le processeur, le disque, l'écran et l'interface de communication sans fil etc. Selon [54-55], l'interface sans fil consomme jusqu'à 50% de l'énergie globale du nœud mobile. Les protocoles de routage à basse consommation d'énergie proposés dans la littérature pour les réseaux ad hoc, cherchent soit à minimiser l'énergie dissipée lors des communications actives (durant les opérations d'émission et de réception, acheminement inclus) ou celle consommée dans les périodes inactives (quand l'interface sans fil n'effectue aucune communication).

2.1 Consommation d'énergie dans les différents modes d'opération des interfaces de communication sans fil

Plusieurs études [54] ont mesuré la consommation d'énergie dans les interfaces de communications sans fil. Dans ces études, les différents modes d'opération d'une interface ont été examinés. Il a été trouvé que l'interface sans fil d'un équipement mobile ne consomme pas uniquement de l'énergie en communiquant avec les autres équipements, mais aussi quand elle n'effectue aucune opération de communication. Cela confirme que "énergie" et "bande passante" sont deux métriques différentes [56]. Les modes d'opération examinés sont les suivants :

- **Le mode transmission** : Correspond à l'état de transmission des paquets.

Interface de communication sans fil (mW)	Transmission	Réception	Inactif	Veille
Aironet PC 4800	2500	900	110	20
Aironet 350 PCI	1870	1620	1440	910
Lucent Bronze	1300	910	840	66
Lucent WaveLAN	1400	1200	1000	150
Cabletron Roamabout	1400	1000	830	130

TABLE 4.1 – Valeurs de la consommation d'énergie (en mW) de quelques interfaces de communication sans fil IEEE 802.

- **Le mode réception** : Correspond à l'état de réception des paquets.
- **Le mode inactif** : C'est le mode par défaut dans un réseau ad hoc, où l'équipement mobile écoute le canal en attendant de recevoir ou d'émettre des paquets.
- **Le mode veille** : Correspond à l'état où l'interface sans fil est éteinte. Dans ce mode, l'équipement mobile ne peut ni émettre ni recevoir. C'est le mode où l'interface est le plus économe en énergie, toutefois un certain délai est nécessaire pour passer dans un des trois autres modes.

Le taux de consommation d'énergie dans chaque mode dépend de l'implémentation matérielle de l'interface sans fil. Le Tableau 4.1 donne un aperçu sur la consommation d'énergie dans chaque mode d'opération pour différentes gammes d'interfaces de communications sans fil [57]. A partir du Tableau 4.1, nous notons que l'énergie consommée en mode émission est la plus élevée suivie par l'énergie consommée en mode réception. Bien que l'équipement mobile n'effectue pas d'opérations de communication dans le mode inactif, il gaspille une quantité d'énergie considérable qui s'approche de celle consommée en mode réception, en effet il doit écouter le canal régulièrement. L'interface sans fil consomme la plus basse énergie quand elle est en mode veille.

Dans [56], il a été trouvé que les émissions de type unicast et broadcast dans IEEE 802.11 ont des coûts de consommation différents. Cela a été justifié par le fait que les émissions unicast requièrent l'échange de paquets de contrôle (RTS, CTS, etc.) entre les nœuds émetteurs et les nœuds récepteurs, tandis que les émissions de type broadcast ne requièrent pas ce genre d'échange.

Tableau 4.1

2.2 Sources de perte d'énergie

Ils existent plusieurs sources de consommation d'énergie en communications dans un réseau ad hoc. Quelques sources sont utiles tandis que d'autres sont considérées

comme des pertes qui doivent être réduites ou éliminées. Certaines pertes d'énergie lors des communications sont dues aux facteurs suivants [58] :

- **Le mode inactif** : L'interface sans fil de communication gaspille de l'énergie sans effectuer aucune tâche utile. Une solution possible est de mettre périodiquement les nœuds en veille.
- **Les collisions** : Elles surviennent surtout dans des conditions de trafic élevé. Les paquets affectés par les collisions ne sont pas correctement reçus et l'énergie consommée lors de leurs émissions et réceptions est gaspillée. Une solution possible à ce problème est d'améliorer par les stratégies de retransmission des protocoles MAC ou par une meilleure gestion du trafic.
- **Le surcoût des protocoles** : Cela fait référence aux paquets de contrôle que génèrent les différents protocoles de communication, et qui imposent une consommation d'énergie supplémentaire à ce qui est strictement nécessaire à la transmission des données. Une solution possible à ce problème est les protocoles de communication restreignent au maximum les paquets de contrôle qu'ils génèrent.
- **Le taux d'erreurs** : Ce phénomène est dû à la qualité de la transmission de l'interface de communication sans fil. Les données impliquées lors des transmissions erronées ne sont pas correctement reçues ce qui fait que l'énergie consommée lors de leurs transmissions est perdue. De plus, les protocoles de routage réactifs interprètent les retards des acquittements comme une rupture du lien, ce qui provoque la diffusion de paquets de requête dans tout le réseau pour rétablir les chemins considérés invalides. Cela constitue une grande source de dissipation d'énergie. Une solution possible à ce problème est d'augmenter la robustesse des protocoles de routage, par exemple par le choix de chemins stables ou par l'utilisation du routage multichemin.

3 Notion de durée de vie d'un réseau

Il existe différentes définitions pour la durée de vie d'un réseau mobile ad hoc (fondées sur la fonctionnalité désirée). Elle peut être définie par la durée jusqu'au moment où le premier nœud est épuisé énergétiquement. Dans ce premier cas on considère que tout nœud du réseau est indispensable. Elle peut être également définie par le temps jusqu'au moment où une certaine proportion de nœuds est épuisée énergétiquement. Si la proportion de nœuds épuisés dépasse un certain seuil, cela peut avoir comme conséquence la non couverture de certaines régions ou le partitionnement du réseau mobile ad hoc. Les principales définitions de la durée de vie proposées dans la littérature sont les suivantes :

- La durée jusqu'à ce que le premier nœud épuise toute son énergie [57-58].
- La durée pendant laquelle le réseau satisfait continuellement les besoins de l'application [60-61].
- La durée jusqu'à ce que tous les nœuds épuisent leur énergie [62].
- La demi-vie du réseau mobile ad hoc : la durée jusqu'à ce que 50% des nœuds épuisent leur énergie [63].

Finalement, nous constatons plusieurs définitions convergent puisque certaines d'entre elles ne sont que des relaxations des autres, par contre la deuxième solution est trop

dépendante du type d'application, elle n'est pas précise donc on ne pas l'utilisé.

4 Conservation d'énergie

La conservation d'énergie est importante dans un réseau mobile ad hoc pour augmenter la durée de vie des nœuds et, par conséquent, celle de tout le réseau. Cette tâche est particulièrement difficile du moment que les actions de conservation d'énergie doivent être coordonnées [64] (c'est-à-dire l'échange d'information entre nœuds et une prise de décision cohérente). Par exemple un nœud avant d'acheminer les paquets à autre nœud, il est inutile d'économiser l'énergie du second si l'énergie du premier est épuisé. Ils doivent coordonner leurs décisions et leurs actions.

Plusieurs techniques de conservation de l'énergie dans les réseaux ad hoc ont été développées [65-69]. En général, ces techniques se concentrent sur deux parties : la partie réseau (i.e., la gestion d'énergie est prise en compte dans les opérations de chaque nœud, ainsi que dans la conception de protocoles du réseau), et la partie détection des données (i.e., des techniques de réductions des données et l'efficacité énergétique au niveau d'acquisition). Les protocoles de conservation de l'énergie peuvent être à chemin unique ou multichemin, nous présenterons ces deux types ci-dessous, et nous donnerons nos justifications de notre choix du multichemin en fin de section.

5 Conservation d'énergie dans les protocoles de routage à chemin unique

Pour minimiser la consommation de l'énergie au niveau routage, une première technique propose de choisir le meilleur chemin en se basant sur les contraintes d'énergie comme facteur de performance. Ceci se traduit par le fait de choisir le chemin garantissant la consommation d'énergie la plus faible ou encore de choisir le chemin passant par les nœuds intermédiaires ayant les batteries les plus chargées. Il faut aussi garantir une diminution de l'énergie consommée par la procédure de découverte des chemins.

Une technique pour diminuer la consommation de l'énergie consiste à minimiser le nombre de messages de contrôle engendrés par le protocole de routage. Divers protocoles sont proposés pour assurer la conservation de l'énergie au niveau routage :

- **Minimum Total Transmission Power Routing (MPTR)** [41] : Permet de choisir le chemin minimisant la consommation totale d'énergie en considérant que le meilleur chemin est celui assurant la consommation minimale des capacités des batteries. Il effectue le choix du chemin en calculant la puissance de transmission des nœuds. L'inconvénient est que ce protocole ne considère pas l'énergie restante des nœuds ainsi il ne prolonge pas la durée de vie d'un nœud donné et qu'il ne tient pas compte des nœuds qui se réveillent régulièrement et redécouvrent leurs voisins.

- **Minimum Battery Cost Routing (MBCR)** [70] : Considère que la capacité restante des batteries reflète mieux la durée de vie d'un nœud et choisit le chemin qui maximise la capacité restante de la batterie. L'avantage de ce protocole est qu'il protège les nœuds contre l'épuisement et qu'il garantit le choix du meilleur chemin dans le cas où les nœuds présentent les mêmes caractéristiques (même capacité de stockage d'énergie, même vitesse de consommation d'énergie). Son inconvénient par contre est qu'un chemin contenant un nœud avec une capacité faible peut quand même être choisi.
- **Min-Max Battery Cost Routing (MMBCR)** [70] : Choisit le chemin qui maximise la capacité résiduelle du goulot d'étranglement du chemin. Le goulot d'étranglement est le nœud ayant le minimum de capacité résiduelle. Ce protocole a pour avantage d'équilibrer l'utilisation des batteries entre les nœuds du réseau. Par contre, il ne garantit pas que le chemin choisi soit le chemin ayant la plus faible consommation d'énergie.
- **Conditional Min-Max Battery Capacity Routing (CMMBCR)** [70] : Propose de sélectionner un ensemble de chemins dont tous les nœuds proposent une capacité résiduelle minimale puis d'appliquer MPTR. Ceci assurera le choix d'un chemin vérifiant que la capacité résiduelle minimale est au-dessus d'une certaine borne et que ce chemin minimise la consommation d'énergie.

L'idée fondamentale de ces protocoles est de router les paquets en fonction de la minimisation d'un critère relatif à la consommation des batteries électriques. Cela peut être de deux types : (a) minimiser de manière globale, (b) minimiser de manière locale.

a) Minimiser de manière globale

La première approche consiste à minimiser l'énergie consommée pour atteindre le nœud de destination, en cherchant le chemin de plus faible consommation. Par exemple les auteurs dans [71] fournissent un algorithme (PSR-Power-aware Source Routing), qui reprend le protocole DSR [18], et assigne à chacun des liens un cout qui est fonction de l'énergie nécessaire pour transmettre un paquet sur ce chemin. Le routage se fait donc préférentiellement suivant les chemins de plus faible cout, en agrégeant le cout de l'ensemble des liens composant le chemin. Ce protocole a le mérite de réduire globalement la consommation du réseau mobile ad hoc.

b) Minimiser de manière locale

La deuxième approche tente d'augmenter la durée de vie du réseau mobile ad hoc, en améliorant la durée de vie des nœuds. La prise de décision est distribuée sur tous les nœuds et ne requiert pas l'échange d'informations globales sur l'état des autres nœuds du réseau. Par exemple le protocole LEAR (*Load Energy-Aware Routing*) proposé dans [66] met en cause la "bonne volonté" de chaque nœud mobile à participer au processus de sélection de chemin et de relayer des paquets de données au nom d'autres nœuds. Lors de la recherche d'un chemin, chaque nœud utilise les informations locales sur le niveau de sa propre batterie pour décider de participer ou non au processus de sélection du chemin.

Le routage multichemin est un paradigme compatible avec l'objectif de la conservation de l'énergie dans les réseaux ad hoc, permettant d'étendre la durée de vie du réseau, d'assurer la fiabilité, et de réduire le surcout du routage. Nous présentons dans ce qui suit quelques protocoles multichemin assurant la conservation de l'énergie.

6 Les protocoles de routage multichemin avec conservation d'énergie

Le principal but d'un protocole de routage est d'assurer la fonction du routage dans un réseau le plus longtemps possible. Ce but peut être accompli en réduisant au minimum l'énergie du nœud mobile non seulement pendant la communication active mais également en mode inactif [72].

Dans ce qui suit, nous présentons les principaux protocoles de routage multichemin réactifs, dont la métrique de routage est basée sur l'énergie. D'une part, ces protocoles ont pour principaux objectifs de garantir le maintien de la connectivité du réseau ad hoc, et de mieux gérer la consommation d'énergie du réseau entier. D'autre part, ils améliorent la durée de vie du réseau, ils sont fiable et par conséquent ils améliorent les performances du réseau. Ces protocoles sont : MMRE [73], REER [74] et ZD-AOMDV [75].

6.1 Le protocole MMRE

MMRE-AOMDV (*Maximal Minimal Residual Energy Ad hoc On Demand Multipath Distance Vector*) [73] est un protocole de routage qui exploite le maximum des minimums de l'énergie résiduelle des nœuds. Il établit un équilibre de la consommation d'énergie pour entre tous les nœuds du réseau. Le protocole MMRE a deux principaux objectifs :

- a) Trouver l'énergie résiduelle minimale des nœuds pour chaque chemin pendant la découverte des chemins
- b) Trier l'énergie résiduelle des nœuds par un tri décroissant et utilise le chemin avec l'énergie résiduelle maximum pour transmettre les paquets des données.

6.1.1 Calcul de l'énergie résiduel minimale des nœuds

Pour trouver le minimum de l'énergie résiduelle des nœuds, plusieurs modifications sont faites au protocole AOMDV. Cela permet le calcul du minimum d'énergie résiduelle des nœuds pour chaque chemin entre un nœud source et un nœud de destination. Chaque paquet *RREQ* et paquet *RREP* porte un champ supplémentaire appelé *min_re_energy* pour indiquer le minimum de l'énergie résiduelle des nœuds d'un chemin. Il est utilisé chaque fois qu'un nœud i reçoit une requête de découverte d'un chemin à une destination d du voisin j .

Les variables $Seq_num_i^d$, $Advertised_hopcount_i^d$, $Route_list_i^d$ et $re_energy_i^d$ représentent respectivement le numéro de séquence du nœud i , le nombre maximum de sauts du nœud i vers d , la liste des chemins depuis le nœud i vers la destination d et l'énergie résiduelle du nœud i (voir figure 4.2).

Au niveau du nœud source, la valeur du champ *min_re_energy* a une valeur maximale, égale à l'énergie initiale d'un nœud source. Chaque fois qu'un nœud intermédiaire reçoit un paquet *RREQ*, MMRE applique les règles décrites dans l'algorithme 3.

Seq_num = numéro de séquence de la destination d au nœud i .

$Advertised_hopcount_i^d$ = le maximum de nombre de sauts.

$Route_list_i^d$ = liste des routes depuis un nœud i vers la destination d .

Begin

If ($seqnum_i^d < seqnum_j^d$) **then**

$seqnum_i^d \leftarrow seqnum_j^d$

If ($i \neq d$) **then**

$min_re_energy_j^d \leftarrow re_energy_i$

$Advertised_hopcount_i^d \leftarrow \infty$;

$route_list_i^d \leftarrow NULL$;

 insert (j , $advertised_hopcount_j^d + 1, min_re_energy_j^d$) into

$route_list_i^d$;

else

$advertised_hopcount_i^d \leftarrow 0$;

end If

else

If ($(seqnum_i^d = seqnum_j^d)$ and

$((advertised_hopcount_i^d) > (advertised_hopcount_j^d))$) **then**

 insert (j , $advertised_hopcount_j^d + 1, min_re_energy_j^d$) into

$route_list_i^d$;

end If

end If

End.

Algorithm 3: Processus de découverte de chemins dans le protocole MMRE.

Lorsqu'un nœud intermédiaire reçoit un paquet $RREQ$, si le numéro de séquence du paquet reçue est plus grand de celui du nœud, son énergie résiduelle est comparée avec le champ min_re_energy de $RREQ$. Si l'énergie résiduelle de ce nœud intermédiaire est inférieure à min_re_energy de $RREQ$, on met à jour le champ min_re_energy avec la nouvelle valeur du nœud, et ceci afin d'avoir la plus faible

valeur des énergies parmi tous les nœuds de chemin. A la destination, le chemin inverse suit le même principe du protocole AOMDV.

<i>Destination</i>
<i>Sequence_Number</i>
<i>Advertised_hopcount</i>
<i>Route_list</i> {(<i>nexthop1</i> , <i>hopcount1</i> , <i>min_re_energy1</i>),(<i>nexthop2</i> , <i>hopcount2</i> , <i>min_re_energy2</i>),...}
<i>Expiration_time_out</i>

FIGURE 4.1 – Structure d’une entrée de la table de routage de MMRE.

6.1.2 Tri des chemins multiples

Dans MMRE, chaque nœud maintient un champ *Route_list*. Son contenu est représenté sur la figure 4.1. Un champ supplémentaire *min_re_nergie* est ajouté dans le champ *Route_list*. La valeur du champ est mise à jour suivant les règles représentées dans l’algorithme 3.

Le champ *Route_list* est trié par ordre décroissant de la valeur de *min_re_nergie*. Le nœud envoie les paquets de données en utilisant le chemin avec l’énergie résiduelle maximale. Lorsque le nœud source reçoit de nouvelles informations concernant un chemin ayant une meilleure énergie résiduelle minimale, il sélectionne ce nouveau chemin pour l’envoi des paquets de données. Ce processus permet d’empêcher un ou plusieurs nœuds critiques de s’épuiser et permet aussi de prolonger la durée de vie du réseau.

Les résultats de simulations ont montré que le protocole de routage MMRE a de meilleurs résultats que AOMDV en termes des délais de livraison de paquets et de durée de vie du réseau.

6.2 Le protocole REER

REER (*Robust and Energy Efficient Multipath Routing Protocol for Wireless Sensor Networks*) [74] est un protocole robuste multichemin économe en énergie. Il réalise un équilibrage de charge grâce à la distribution du trafic sur un ensemble de chemins à nœud disjoint disponibles. REER utilise l’énergie résiduelle des batteries,

la taille disponible du tampon, et le rapport signal sur bruit (SNR) pour prédire le prochain saut lors de la phase de découverte de chemin.

6.2.1 Fonctionnement du protocole REER

Le protocole REER utilise un seul chemin parmi les chemins découverts pour le transfert de données. Lorsque le coût de chemin utilisé est en dessous d'un certain seuil, il passe au chemin alternatif suivant.

a) Hypothèses

Les auteurs de REER supposent N nœuds identiques et distribués aléatoirement. Tous les nœuds ont le même niveau de puissance de transmission, et assez d'énergie pour communiquer. Le réseau est entièrement connecté. Chaque nœud dans le réseau a un identifiant unique. Par ailleurs, les auteurs supposent que chaque nœud est capable de calculer son énergie résiduelle, et sa taille du tampon disponible, mais aussi d'enregistrer les performances de la liaison entre lui et ses nœuds voisins en termes du rapport signal-bruit (SNR).

b) Coût d'une liaison

Une fonction de coût de la liaison est utilisée par le nœud qui doit sélectionner le prochain saut pendant la phase de la découverte de chemins. Le calcul de cette fonction est basé sur les travaux de [71], avec quelques modifications. Avec N_x est l'ensemble des voisins du nœud x . La fonction de coût inclut un facteur énergétique, le facteur de tampon disponible, et les interférences avec des pondérations appropriées (α , β , et γ) :

$$N_{exthop} = \max_{y \in N_x} \{ \alpha E_{resd,y} + \beta B_{buffer,y} + \gamma I_{interference,xy} \} \quad (4.1)$$

$E_{resd,y}$ est l'énergie résiduelle du nœud y , où $y \in N_x$, $B_{buffer,y}$ est la taille de la mémoire tampon disponible du nœud y , et $I_{interference,xy}$ est le SNR du lien entre les nœuds x et y . Cette fonction de coût ne considère que l'énergie résiduelle du nœud y . Parce que le nœud y consomme l'énergie en réception et en émission, s'il est choisi comme prochain saut du nœud x jusqu'à la destination.

Le coût total ($C_{total,P}$) pour un chemin P constitué d'un ensemble de K nœuds est la somme des coûts de liaison individuels $l_{(xy)i}$, $i \in K$ le long du chemin. Nous avons :

$$C_{total,P} = \sum_{i=1}^{K-1} l_{(xy)i} \quad (4.2)$$

6.2.2 Découverte des chemins

Dans REER, basé sur l'idée de la diffusion dirigée, le nœud puit (sink) commence la phase de découverte des chemins multiples pour créer un ensemble de voisins qui peuvent transmettre des données vers le nœud puit depuis le nœud source. Les chemins découverts sont des chemins à nœuds disjoints. Dans un routage multichemin, les chemins à nœuds disjoints sont généralement préférés car ils sont les plus tolérants aux pannes (toutefois cela nécessite en général plus de ressources). Si un nœud intermédiaire dans un ensemble des nœuds disjoints est énergétiquement épuisé, seul le chemin contenant ce nœud est affectée, il y a donc un impact minimum sur les chemins du multichemin. La procédure de découverte des chemins est exécutée conformément aux phases suivantes :

a) Phase d'initialisation

Chaque nœud de capteur diffuse un paquet *HELLO* afin d'avoir suffisamment d'informations sur ses voisins. Un nœud de capteur met à jour sa table des voisins. La table des voisins contient des informations sur la liste des voisins. Figure 4.2 illustre la structure de paquet *HELLO*. Le champ *Link quality* est exprimé en termes de rapport signal-bruit (SNR) pour la liaison entre le nœud et son voisin. *Hop count* donne la distance entre le nœud voisin émetteur du paquet et le nœud identifié par le champ *Source ID* du paquet.

Source ID	Hop count	Residual energy	Free buffer	Link quality
-----------	-----------	-----------------	-------------	--------------

FIGURE 4.2 – Structure du paquet Hello.

b) Phase de découverte de chemin

Après la phase d'initialisation, chaque nœud a suffisamment d'informations pour calculer la fonction de coût pour toutes des destinations qui passent par ses nœuds voisins. Ensuite, le nœud sélectionne localement son nœud prochain préféré pour une destination donnée en utilisant la fonction de coût du lien, et envoie un paquet *RREQ* à son prochain nœud préféré (Figure 4.3 montre la structure du paquet *RREQ*). De même, grâce à la fonction de coût du lien, le nœud prochain préféré calcule localement ses nœuds prochains préférés vers le nœud source, et envoie un paquet *RREQ* à ses prochains nœuds, l'opération se poursuit jusqu'au nœud source.

Source ID	Dest ID	Route ID	Residual energy	Free buffer	Link quality	Route cost
-----------	---------	----------	-----------------	-------------	--------------	------------

FIGURE 4.3 – Structure du paquet RREQ.

c) Phase de découverte des chemins alternatifs

Pour le deuxième chemin alternatif, le nœud puit envoie un paquet *RREQ* au nœud prochain préféré suivant. Pour éviter d'avoir des chemins avec des nœuds communs, chaque nœud accepte un seul *RREQ* provenant du même nœud. Pour les nœuds qui reçoivent plus d'un paquet *RREQ*, seul le premier paquet *RREQ* est accepté.

6.2.3 La maintenance des chemins

Pour garder les chemins multiples actifs, le nœud source doit faire une inondation périodique du paquet *KeepAlive* sur les chemins alternatifs afin de vérifier s'ils ont valides. La fréquence du paquet *KeepAlive* détermine la rapidité avec laquelle le

protocole de routage détecte les ruptures sur le chemin. La figure 4.4 décrit le paquet *KeepAlive*.

Source ID	Next hop ID	Residual energy	Free buffer	Link quality
--------------	----------------	--------------------	----------------	-----------------

FIGURE 4.4 – Structure du paquet *KeepAlive*.

6.2.4 Transmission de données

Après la découverte de plusieurs chemins, le nœud source commence à transmettre les paquets de données au nœud puit. Le protocole REER permet d'acheminer les données vers le nœud puit, en ajoutant un seuil pour comparer les chemins. Quand le nœud source reçoit une requête du nœud puit, le nœud source commence la collecte des informations, et puis les envoie par le meilleur chemin valide. Chaque nœud consulte sa table de voisins pour transmettre les paquets de données au prochain nœud. Ce processus est répété jusqu'à ce que les paquets de données atteignent le nœud puit. Quand un nœud ne parvient pas à transmettre le paquet de données à son prochain nœud, il considère ce lien rompu et envoie un paquet d'erreur vers le nœud source. Après la réception du paquet d'erreur, le nœud source supprime le chemin courant de sa table de routage et sélectionne immédiatement le prochain meilleur chemin valide, et reprend le processus de transmission des paquets de données. Si tous les chemins sont rompus, le nœud source diffuse une requête au nœud puit pour initier une nouvelle découverte de chemins. Une version améliorée du protocole REER utilise un sous-ensemble de chemins valides pour transférer les paquets de données afin de distribuer la charge sur les nœuds, et pour éviter l'épuisement rapide des batteries.

Les résultats des simulations ont montré que le protocole REER réalise plus d'économie d'énergie, de délais réduits par rapport à d'autres protocoles de la littérature.

6.3 Le protocole ZD-AOMDV

ZD-AOMDV (Zone-Disjoint Ad-hoc On-demand Multi-path Distance Vector)[75], découvre les chemins disjoints entre une source et une destination, pour envoyer des paquets de données à travers ces chemins simultanément. Les chemins disjoints ne sont pas totalement indépendants les uns des autres à cause des mécanismes d'accès au canal partagé dans les réseaux sans fil. A cet effet, le nombre de voisins actifs est calculé suivant une stratégie bien déterminée. La technique proposée compte le nombre de voisins actifs pour chaque chemin, et choisit les chemins avec moins de voisins actifs pour l'envoi des données. Les voisins actifs d'un nœud sont définis comme des nœuds qui ont déjà reçu la requête *RREQ*. ZD-AOMDV est basé sur le protocole de routage de base AODV.

6.3.1 Les modifications nécessaires d'AODV pour l'implémentation de ZD-AOMDV

Dans AODV, les nœuds intermédiaires maintiennent une table des chemins découverts au cours du processus de découverte des chemins. Ainsi, si un nœud intermédiaire reçoit une requête de demande de chemin provenant de la source, il renvoie un paquet de réponse par un chemin inverse à la source. Cependant, dans l'algorithme proposé, les nœuds intermédiaires n'ont pas besoin de gérer ces tables. Par conséquent, plusieurs requêtes de demande de chemins atteindront la destination. En fait, toutes les requêtes de demande de chemins envoyées par la source sont transmises à la destination. Dans ZD-AOMDV, chaque nœud doit mettre les spécifications de la requête *RREQ* reçue dans une nouvelle table appelé *RREQ_Seen*, afin de répondre aux demandes des nœuds voisins. En outre, pour compter les voisins actifs dans chaque chemin, dans la table *RREQ_Seen*, chaque nœud a un champ nommé "le nombre de voisins actifs après l'envoi *RREQ*" brièvement appelé *After_A_N_C*. En outre, le champ *ActiveNeighborCount* est ajouté à l'entête de *RREQ* et *RREP* pour comptabiliser les nœuds suivants du chemin courant. Enfin, deux nouveaux paquets *RREQ_Query* et *RREQ_Query_Reply* sont ajoutés au processus de découverte de chemin. Plus en profondeur, le nœud initiateur de la requête met les informations de *RREQ* courant dans *RREQ_Query* et l'envoie à ses voisins. Si les nœuds eux-mêmes attendent la réponse du processus de requête, ils envoient une *RREQ_Query_Reply* au nœud initiateur.

6.3.2 Principe de fonctionnement du protocole ZD-AOMDV

Quand un nœud est sur le point d'envoyer des données vers une destination spécifique et il ne trouve pas un chemin valide vers sa destination, il lance le processus de découverte de chemin par la production et l'envoi de requêtes *RREQ* à ses voisins. Dans cette requête, la valeur initiale de zéro sera attribuée au champ *ActiveNeighborCount*. Par conséquent, les nœuds voisins reçoivent le paquet *RREQ*, et fixent leurs noms comme les fondateurs de l'un des chemins et mettent les informations du chemin dans la table du chemin. Mais avant de renvoyer le paquet *RREQ*, ils envoient une requête pour avoir des informations sur leurs voisins. En fait, ils demandent à tous leurs voisins : "*Avez-vous vu un RREQ à cette spécification ?*", ensuite, ils augmentent la valeur de *ActiveNeighborCount* dans le paquet *RREQ* pour les voisins qui ont une réponse positive à la requête. Pour cette dernière, les nœuds utilisent les paquets *RREQ_Query* et *RREQ_Query_Reply*. En fait, le nœud de requête envoie le paquet *RREQ_Query* à ses voisins et après la période de temps spécifique attend les réponses des voisins à la question. Si les nœuds voisins ont déjà reçu *RREQ*, ils répondent positivement au nœud initiateur de la requête. La réponse à la requête est effectuée par la production et la transmission des paquets *RREQ_Query_Reply*. Enfin, après l'expiration du temps, le nœud initiateur de la demande de recherche, diffuse le paquet *RREQ* pour poursuivre le processus de découverte. Si un nœud reçoit une deuxième requête *RREQ* il initie le processus de recherche mais juste pour des nouveaux voisins. Après le rejet de la requête *RREQ*, si le nœud a été demandé par un *RREQ*, avant l'arrivée des *RREPs*, il incrémente la valeur de champ *After_A_N_C* pour chaque requête. Lorsque le *RREP* est envoyée de la source vers

la destination, le long du chemin, chaque nœud ajoute la valeur de *After_A_N_C* de sa table *RREQ_Seen* au champ *ActiveNeighborCount* du paquet *RREP*. Ainsi, quand un paquet *RREP* atteint la source, son champ *ActiveNeighborCount* contient le nombre exact des voisins actifs de ce chemin. À ce stade, la source peut choisir les *RREPs* de celles reçues qui ont la plus faible valeur de *ActiveNeighborCount* et envoient les paquets des données simultanément par ces chemins.

6.3.3 Processus de découverte et de sélection des chemins

Dans ZD-AOMDV, si le nœud source a des paquets de données à envoyer, il diffuse le paquet *RREQ* et attend l'arrivée des paquets *RREP*. Chaque nœud intermédiaire suit les règles décrites dans le pseudo code décrit ci-dessous.

1. If you received the *RREQ* packet and this packet is acceptable, do the following steps.
Otherwise, dismiss the packet.
 - (a) Put this packet's specification into the *RREQ_Seen* table.
 - (b) Prepare the *RREQ_Query* packet and assign it a value.
 - (c) There is a question on this packet that asks : Have you seen such a request packet before ?
 - (d) Send the *RREQ_Query* packet to your neighbors.
 - (e) Wait a specific period of time for your neighbors to reply
 - (f) Increase the *ActiveNeighborCount* with regard to the number of accepted replies.
 - (g) Rebroadcast the *RREQ* packet.
2. When you received the *RREQ_Query*, perform the following actions :
 - (a) With regard to the *RREQ_Seen* table, if you have not seen this *RREQ* before, dismiss the packet and don't consider it.
 - (b) According to the *RREQ_Seen* table, if you have seen this *RREQ* before, inform the query node by sending a *RREQ_Query_Reply* packet then add one unite to the *After_A_N_C* field of the corresponding *RREQ* in its *RREQ_Seen* table.
3. If you have received the *RREQ_Query_Reply* packet, add one unite to this *RREQ*'s *ActiveNeighborCount* field.
4. When you receive the *RREP* packet, add the corresponding *After_A_N_C* to *ActiveNeighborCount* field of *RREP* packet and send it.

Pour une meilleure compréhension de ZD-AOMDV, on considère le réseau présenté dans la figure 4.5. Dans ce réseau, la ligne entre deux nœuds signifie que ces deux nœuds sont dans la même portée par radio.

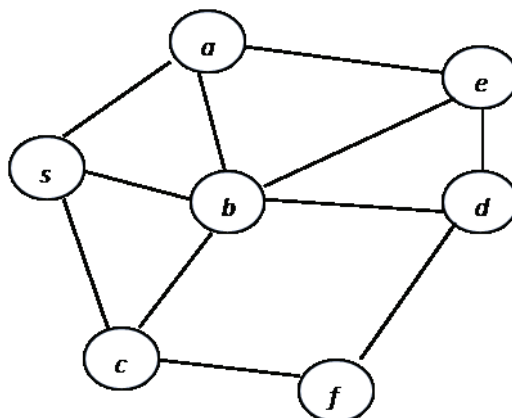


FIGURE 4.5 – Exemple d'une topologie d'un réseau.

Supposons dans cet exemple, le nœud s comme un nœud source veut envoyer des données au nœud de destination d . Le nœud s diffuse la requête $RREQ$ avec le champ $ActiveNeighborCount$ mise à zéro. Dans la première étape, a , b et c reçoivent la requête et insèrent ces spécifications dans la table $RREQ_Seen$ avec la valeur initiale de zéro pour le champ $After_A_N_C$. Puis ils ajoutent leurs adresses comme des fondateurs d'un chemin dans la requête $RREQ$. En outre, ils commencent la procédure d'interrogation selon l'algorithme proposé. A cet effet, ils envoient le paquet $RREQ_Query$ à leurs voisins et attendent que leurs voisins réagissent. Après cette demande, les nœuds a et c ne reconnaîtront que le nœud b dans leurs voisinage et ajoutent une unité au champ $ActiveNeighborCount$ dans la requête $RREQ$, mais le nœud b reconnaît le voisin du nœud a et c . En outre, il ajoute deux unités au champ $ActiveNeighborCount$ de la requête $RREQ$. Ensuite, les trois nœuds a , b et c diffuse la requête $RREQ$ pour terminer le processus de découverte. Ce processus est représenté sur la figure 4.6.

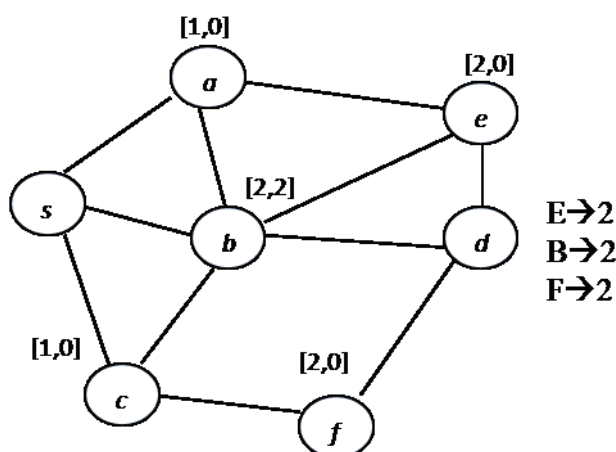


FIGURE 4.6 – Première phase de l'algorithme ZD-AOMDV.

Chaque nœud a deux nombres écrits en ci-dessus, le nombre à gauche indique la

valeur du champ *ActiveNeighborCount* et le nombre de droite montre la valeur du champ *After_A_N_C* exactement avant la re-diffusion des *RREQs* dans le tabe de *RREQ_Seen* correspondante. Dans l'étape suivante, ces *RREQs* atteignent d'autres nœuds (en raison de simplification de l'exemple les copies de *RREQ* qui ont été livrés aux nœuds par d'autres chemins, ne sont pas indiquées). Le nœud *b* délivre le premier *RREQ* à *d*, et la destination *d* répond en envoyant un paquet *RREP* à la source. Le champ *ActiveNeighborCount* de ce chemin est égal à 2. Toutefois, *RREQ* atteint *e* et *f* à travers le chemin correspondant. Ces deux nœuds commencent le processus de découverte séparément, reconnaissent le nœud *b* dans leurs voisinage, et ajoutent une unité à *ActiveNeighborCount* existant dans *RREQ*. Dans les faits, les nœuds *e* et *f* envoient *RREQ_Query* au nœud *b* effectuant le processus d'interrogation. En outre, parce que le nœud *b* a propagé ce *RREQ* avant, il aura une réponse positive pour les deux requêtes et il envoie *RREQ_Query_Reply* pour chaque requête. Le nœud *b* trouve deux nouveaux voisins (*e* et *f*) après cette découverte.

Par conséquent, après l'envoi de la réponse de la requête, le nœud *b* augmente par une unité le champ *After_A_N_C* du chemin courant dans la table *RREQ_Seen* pour chaque requête. Cette procédure est représentée dans la figure 4.6, et nous pouvons voir que la valeur du champ *After_A_N_C* du nœud *b* est égale à 2. La figure 4.7 montre l'état du réseau où tous les *RREQs* ont atteint la destination. Comme il est indiqué sur la figure, les trois *RREQs* ont été reçus par la destination avec une valeur de 2 du champ *ActiveNeighborCount*. A ce stade, la destination reçoit le *RREQ*, crée les *RREPs*, remplit les champs *ActiveNeighborCount* de *RREP* par les valeurs des *RREQs* correspondantes, puis envoie les paquets *RREP* à la source.

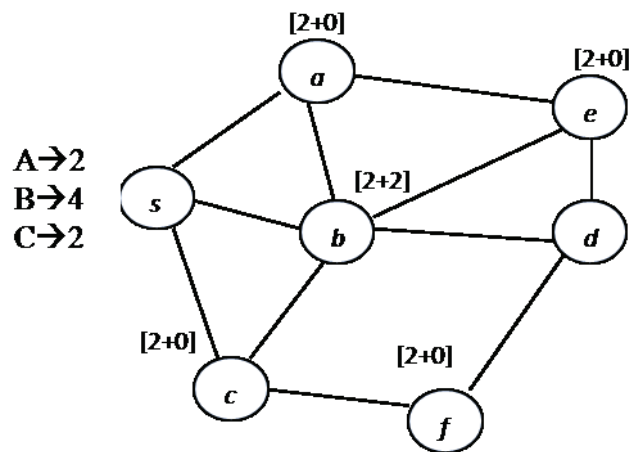


FIGURE 4.7 – Deuxième phase de l'algorithme ZD-AOMDV.

Après avoir reçu chacun de ces *RREPs*, les nœuds intermédiaires doivent ajouter la valeur *After_A_N_C* de la table *RREQ_Seen* dans le paquet *RREP*. Ceci est représenté sur la figure 4.7. Sur cette figure, l'action de la somme correspondante de chaque nœud est représentée au-dessus des nœuds. Comme on peut le voir, les nœuds *a*, *e*, *f* et *c* dans ce scénario n'ajoute pas de valeur dans le champ de *ActiveNeighborCount* dans *RREP*, mais le nœud *b* ajoute deux unités à *ActiveNeighborCount*

dans son propre *RREP*. Enfin de compte, les paquets *RREP* sont livrés à la source (nœud *s*).

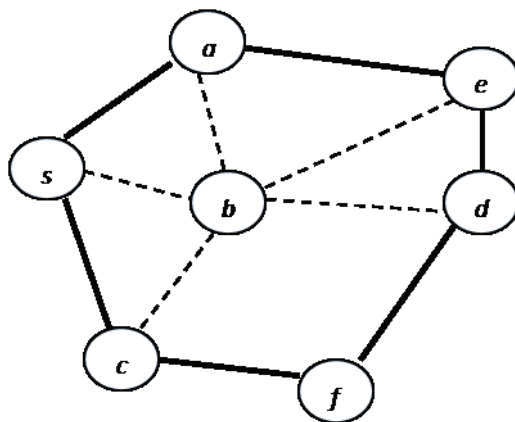


FIGURE 4.8 – Les chemins sélectionnés (suivant l'exemple).

Après expiration du temporisateur, le nœud source trie les *RREPs* reçus dans un ordre croissant. Puis il sélectionne le nombre requis de *RREPs* depuis la file d'attente et commence à envoyer les paquets de données simultanément dans les chemins sélectionnés. Supposons qu'on sélectionne deux chemins pour l'envoi simultané, dans ce cas, les deux chemins $s-a-e-d$ et $s-c-f-d$ sont sélectionnés (comme il est montré dans la figure 4.8). ZD-AOMDV consomme moins d'énergie et améliore le délai de bout en bout comparativement à d'autres protocoles multichemin.

6.4 Critiques des différentes techniques exposées

Tout les protocoles multichemins traitées dans cette section permettent de résoudre le problème de la conservation de l'énergie. Cependant les mécanismes d'économie d'énergie basées uniquement sur l'énergie résiduelle ne peuvent être utilisés pour établir le meilleur chemin entre un nœud source et un nœud de destination. D'une part, si un nœud est prêt à accepter toutes les demandes de route seulement parce qu'il a une énergie résiduelle suffisante dans la batterie du nœud, ceci mène à un acheminement chargé de trafic via ce nœud. D'autre part, l'économie d'énergie excessive néglige la consommation de l'énergie au niveau des nœuds individuels, ce qui peut accélérer le partitionnement du réseau en épuisant les batteries de nœuds qui seraient critiques dans le réseau, ceci réduit les performances du réseau. Par conséquent, équilibrer la consommation d'énergie est un remède pour ces types de problèmes. Enfin, la majorité de ces protocoles n'ont été comparés qu'avec leurs protocoles originaux (AODV, AOMDV, DSR, SMR, etc.) qui ne prennent pas explicitement la contrainte d'énergie, et donc ces évaluations sont peu crédibles.

7 Conclusion

La durée de vie d'un réseau ad hoc est étroitement liée à la vie des nœuds constituant ce réseau. Ces nœuds dépendent essentiellement de leur consommation d'énergie. Nous avons présenté dans ce chapitre quelques approches de conservation d'énergie dans les réseaux ad hoc. Le premier axe des approches de conservation d'énergie vise à minimiser la consommation de l'énergie au niveau du routage à chemin unique, ceci passe par un bon choix du chemin. Ce chemin doit garantir une meilleure consommation de l'énergie lors de la transmission des paquets des données. Il faut aussi garantir une minimisation de l'énergie consommée par la procédure de découverte ou de maintenance des chemins. La politique de minimisation de la consommation de l'énergie peut être locale ou globale. Parmi les faiblesses du routage à chemin unique, on remarque la rupture du chemin (un seul chemin validé entre chaque source et chaque destination par ces protocoles de routage à chemin unique). Cette rupture entraîne le lancement d'une nouvelle découverte de chemin qui s'avère très coûteuse en termes de surplus d'énergie consommée et de surcoût du nombre de messages de contrôle échangés. Un routage multichemin avec conservation d'énergie peut être une technique alternative pour remédier aux faiblesses du routage à chemin unique.

La seconde partie s'intéresse au routage multichemin avec conservation d'énergie. Plusieurs solutions ont été proposées. Aucune approche de protocoles multichemin de conservation d'énergie n'est la meilleure dans tous les scénarios mais chaque approche se révèle mieux adaptée à certaines situations qu'à d'autres (débit, densité et mobilité). Une comparaison directe entre les différentes approches est difficile car chacune d'elles possède un but différent, et elles sont fondées sur des hypothèses différentes. En réalité, effectuer un routage multichemin avec conservation d'énergie est un problème multi aspects. Focaliser sur un seul aspect du problème peut conduire à des performances sous optimales. Conséquemment, la combinaison de plusieurs approches semble plus appropriée. La conception d'un protocole de routage multichemin à conservation d'énergie dans les réseaux ad hoc est un domaine de recherche où les évolutions sont permanentes et où beaucoup de progrès ont été et restent à réaliser.

Les prochains chapitres de notre travail consistent justement à présenter des solutions protocolaires de routage multichemin avec conservation d'énergie. Ils viseront à prolonger la durée de vie du réseau et à améliorer ses performances.

Chapitre 5

Protocole de routage multichemin

AODVME+

Dans ce chapitre, nous proposons un nouveau protocole de routage réactif pour les réseaux ad hoc basé sur le principe du multichemin. Ceci afin d'exploiter efficacement l'énergie des nœuds, contrôler le surcout généré par les messages de contrôle, minimiser les délais de bout en bout, et minimiser la consommation d'énergie d'un chemin.

Notre protocole est conçu principalement pour les nœuds ayant des batteries électriques limitées, où les ruptures des liens et les défaillances des chemins se produisent fréquemment. L'idée principale de notre protocole est de découvrir plusieurs chemins entre un nœud source et un nœud de destination dans un réseau mobile ad hoc, en exploitant l'énergie résiduelle minimale de nœuds pour empêcher qu'un nœud critique épuise sa réserve d'énergie et se déconnecte du réseau. Après le processus de découverte, les chemins multiples peuvent avoir des énergies minimales très proches. Un nouveau facteur peut être combiné avec cette valeur minimale qui est l'énergie moyenne des batteries des nœuds constituant les chemins.

1 Protocole de routage multichemin AODVME+

Notre protocole AODVME+ (*Ad hoc On-demand Distance Vector Multipath Energy Plus*) [76], est un protocole de routage multichemin réactif. Le but de AODVME+ est de conserver l'énergie des nœuds constituant les chemins. AODVME+ est basé sur les protocoles AODV [17] et AODVM [32]. Ces protocoles utilisent le même principe de fonctionnement et les mêmes types de messages. Dans les réseaux ad hoc, une défaillance d'un nœud peut être causée par un problème physique (par exemple, les dommages physiques ou l'épuisement de la batterie) ou une forte congestion au niveau du nœud, ce qui provoque un échec de réception du paquet. AOMDV [20] et AODVM sont deux protocoles multichemin basés sur le protocole (AODV), l'un des protocoles de routage réactif les plus populaires pour les réseaux ad hoc.

Dans AOMDV la défaillance d'un nœud peut provoquer la rupture d'un multichemin lorsque ce nœud est partagé par tous les chemins du chemin multiple. Ainsi, Ye et al. [32] ont apporté des modifications au protocole AODV, pour découvrir des chemins à nœuds disjoints d'un nœud source vers un nœud de destination. Bien qu'il y ait eu un travail préalable sur la modification AODV pour calculer les chemins à nœuds disjoints [20], le protocole de AODVM est la première version modifiée de AODV qui a la capacité de trouver des chemins à nœuds disjoints. Cependant, AODVM ne prend pas en compte le niveau d'énergie des batteries des nœuds : un nœud peut être incapable d'assurer le transfert de paquets de données en raison de l'épuisement de son énergie, provoquant la rupture du chemin. Notre extension AODVME+ exploite l'énergie résiduelle des batteries afin d'améliorer les performances des réseaux ad hoc. Cette performance est approuvée par la simulation de AODVME+, et sa comparaison avec d'autres protocoles tels que AODVM et MMRE[73].

Nous définissons d'abord quelques hypothèses et nous donnerons ensuite les détails des processus de découverte, de sélection et de maintenance du multichemin. Ces processus sont suivis par une évaluation de notre protocole. Enfin nous concluons ce chapitre.

1.1 Hypothèses et définition du problème

Un réseau sans fil est représenté par un graphe non orienté, $G = (V, E)$, où V est l'ensemble des nœuds et E est l'ensemble des liens bidirectionnels. Soit $w(u)$, $u \in V$ représente l'énergie résiduelle du nœud u .

Soit $c(u, v)$, $(u, v) \in E$ l'énergie nécessaire pour transmettre un paquet de nœud u au nœud v . Nous supposons que $c(u, v) = c(v, u)$ pour tout $(u, v) \in E$. Soit $P_i(u_0, u_k) = u_0^i, u_1^i, \dots, u_k^i$ le i ème chemin dans G entre les deux nœuds $u_0 = u_0^i$ et $u_k = u_k^i$.

Soit $e_{min}(P_i(u_0, u_k))$, l'énergie résiduelle minimale des nœuds constituant le chemin $P_i(u_0, u_k)$ pour un nœud source u_0 vers le nœud de destination u_k , peut être exprimée comme suit :

$$e_{min}(P_i(u_0, u_k)) = \min\{w(u_j^i), 0 \leq j \leq k\} \quad (5.1)$$

L'énergie résiduelle totale du chemin $P_i(u_0, u_k)$, notée par $e_{sum}(P_i(u_0, u_k))$, est donnée comme suit :

$$e_{sum}(P_i(u_0, u_k)) = \sum_{j=0}^k w(u_j^i) \quad (5.2)$$

Soit $e_{average}(P_i(u_0, u_k))$, l'énergie résiduelle moyenne d'un chemin, est exprimée comme suit :

$$e_{average}(P_i(u_0, u_k)) = \frac{e_{sum}(P_i(u_0, u_k))}{k + 1} \quad (5.3)$$

Entre un nœud source s , et un nœud destination d , nous pouvons définir deux formulations du problème :

i) Le maximum des minimums des énergies résiduelles des chemins : c'est-à-dire trouver un chemin $P_i(s, d)$ avec $e_{min}(P_i(s, d))$ maximale.

ii) Le maximum de la moyenne des énergies résiduelles des chemins : c'est-à-dire trouver un chemin $P_i(s, d)$ avec $e_{average}(P_i(s, d))$ maximale.

Trouvez les minimums des énergies des chemins et sélectionner le maximum, réduit les ruptures des chemins et contribue à l'amélioration de la durée de vie des réseaux, mais si ces minimums sont très proches (ou égaux) la sélection du meilleur chemin en terme énergie peut être contrainte. La moyenne des énergies des chemins est parmi les indicateurs pour la sélection des chemins pour le transfert des paquets de données. La combinaison des deux formulations est la base de sélection des chemins de notre protocole. Cette formulation est développée dans ce qui suit.

1.2 Découverte des chemins dans AODVME+

Les protocoles de routage multichemin à nœuds disjoints sont généralement préférés car ils utilisent les ressources du réseau de manière efficace et sont plus tolérants aux pannes.

Pour permettre la découverte de plusieurs chemins disjoints entre un nœud source et un nœud de destination, au lieu de rejeter les paquets de demandes d'une route (*RREQ*) dupliquée, dans AODVME+, un nœud intermédiaire doit traiter les paquets *RREQ* dupliqués de la même manière que le premier paquet de *RREQ* (*Route REQuest*) conservé. Chaque nœud intermédiaire diffuse le paquet *RREQ* reçu depuis ces voisins directs (cette opération continue jusqu'au nœud destination, voir la figure 5.1). Lorsqu'un nœud reçoit un message de *RREQ* dupliqué, il établit toujours un chemin inverse vers le nœud source en enregistrant l'adresse du prochain nœud à partir duquel il a reçu la première copie du paquet de *RREQ*. Les paquets de réponse *RREPs* parcourent les chemins inverse jusqu'au nœud source.

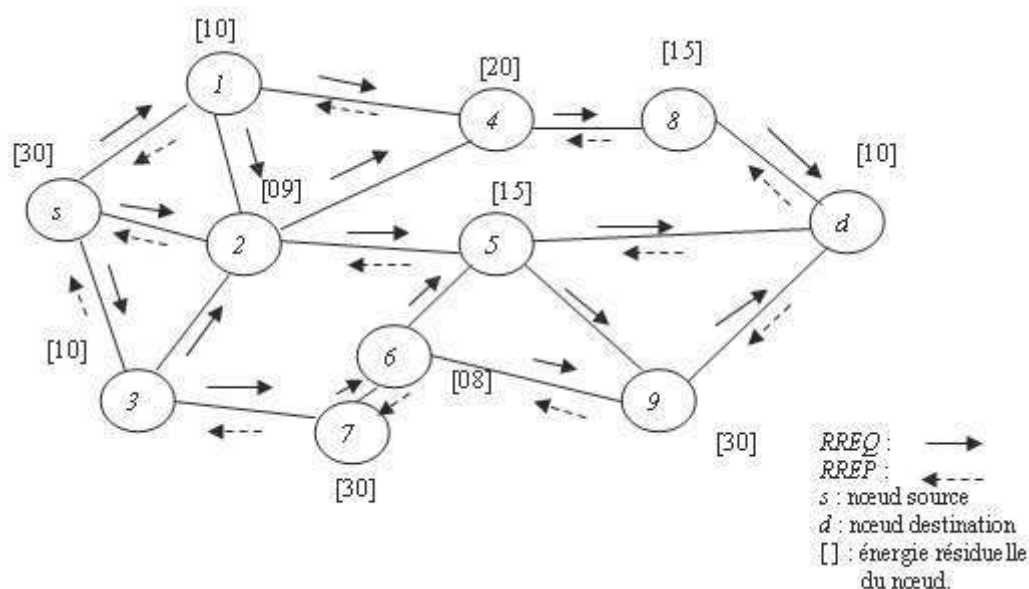


FIGURE 5.1 – AODVME+ : Exemple de diffusion des paquets *RREQ* et *RREP*.

Nous avons modifié le format des paquets *RREQ* et *RREP* du protocole de AODVM en ajoutant deux nouveaux champs : le champ *min_re_energy* et le champ *sum_re_energy*. Lorsque le nœud intermédiaire *i* reçoit un paquet *RREQ* du

nœud j , il compare son énergie résiduelle avec la valeur du champ de min_re_energy ; si elle est inférieure, le nœud remplace la valeur min_re_energy par sa propre valeur et augmente la valeur du champ sum_re_energy par la valeur de son énergie résiduelle. Ce traitement doit tenir compte du numéro de séquence (Seq_num) du paquet $RREQ$ afin d'assurer la fraîcheur des chemins [32]. Le même processus est répété jusqu'à ce que le paquet $RREQ$ atteigne sa destination finale. Nous proposons un algorithme (Algorithme 4) pour le calcul du minimum et la somme des énergies résiduelles des nœuds d'un chemin :

```

Begin
If ( $seq\_num_i^d < seq\_num_j^d$ ) then
     $seq\_num_i^d \leftarrow seq\_num_j^d$ 
    If ( $re\_energy_i < min\_re\_energy_j^d$ ) then
         $min\_re\_energy_j^d \leftarrow re\_energy_i$ 
         $sum\_re\_energy_j^d \leftarrow sum\_re\_energy_j^d + re\_energy_i$ 
         $route\_list_i^d \leftarrow NULL$ ;
         $insert (source_i^d, Seq\_num_s^i, lasthop_i^d, j, hopcount_j + 1,$ 
         $sum\_re\_energy_j^d, min\_re\_energy_j^d, ExpirationTimer) into$ 
         $Route\_list_i^d$ 
    end If
else
    If ( $(seqnum_i^d = seqnum_j^d) and (re\_energy_i < min\_re\_energy_j^d)$ ) then
         $min\_re\_energy_j^d \leftarrow re\_energy_i$ 
         $sum\_re\_energy_j^d \leftarrow sum\_re\_energy_j^d + re\_energy_i$ 
         $insert (source_i^d, Seq\_num_s^i, lasthop_i^d, j, hopcount_j + 1,$ 
         $sum\_re\_energy_j^d, min\_re\_energy_j^d, ExpirationTimer) into$ 
         $Route\_list_i^d$ 
    end If
end If
End.
    
```

Algorithm 4: Algorithme de calcul du minimum et de la somme des énergies.

Avant le lancement du processus de découverte, les champs min_re_energy et sum_re_energy sont initialisées par l'énergie résiduelle du nœud source et la valeur zéro, respectivement.

1.3 Sélection des chemins et la transmission des paquets de données

Après la réception du premier paquet *RREQ*, le nœud de destination attend une certaine période de temps (*RREQ_Wait_Time*) avant de commencer à répondre aux paquets *RREQ* issus de la même requête (même source, même destination et même numéro de séquence). Cette période d'attente est nécessaire pour recevoir les paquets *RREQ*. Le nœud de destination doit répondre par l'envoi d'un paquet *RREP* (*Route REPLY*) vers chaque nœud à partir duquel il a reçu un paquet *RREQ*. Lorsqu'un nœud reçoit son premier message de *RREP*, il crée une entrée de chemin vers le nœud à partir duquel il a reçu le message *RREP* et met à jour le champ *Route_list*. Chaque nœud maintient une structure *Route_list* décrite dans la figure 5.2. Nous avons ajouté deux champs, *min_re_energy* et *sum_re_energy* au champ original *Route_list* de AODVM.

Le champ *min_re_energy* et *sum_re_energy* sont mis à jour selon les règles indiquées dans l'algorithme proposé.

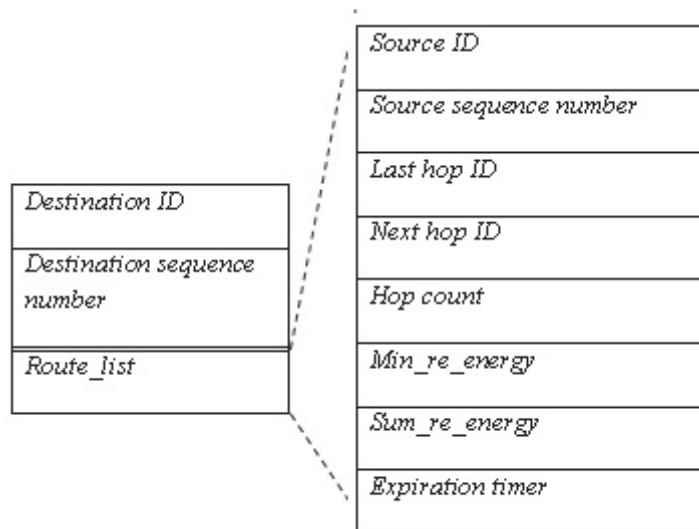


FIGURE 5.2 – Structure de la table de routage du protocole AODVME+

Lorsque le nœud source reçoit le premier message *RREP*, il attend une certaine période de temps (*RREP_Wait_Time*) pour recevoir les *RREPs* avant de sélectionner le meilleur chemin. Le choix du meilleur chemin entre un nœud source s et le nœud de destination d , dépend de deux valeurs, $e_{min}(P_i(s, d))$ et $e_{average}(P_i(s, d))$, qui sont définis par les équations (5.1) et (5.3). Dans cette approche, le nœud source calcule $e_{ma}(P_i(s, d))$, basé sur $e_{min}(P_i(s, d))$ et $e_{average}(P_i(s, d))$ en utilisant une somme pondérée ; formellement présentée comme suit :

$$e_{ma}(P_i(s, d)) = \alpha e_{min}(P_i(s, d)) + (1 - \alpha) e_{average}(P_i(s, d)) \quad (5.4)$$

Le coefficient α représente le degré de pondération. Notre idée est basée sur le tri de tous les chemins à nœuds disjoints entre un nœud source s et le nœud

de destination d suivant la valeur décroissante de $e_{ma}(P_i(s, d))$. Le chemin ayant le maximum (le premier chemin de la liste triée) de $e_{ma}(P_i(s, d))$ est choisi pour transmettre les paquets de données. La figure 5.3 montre le résultat de la découverte

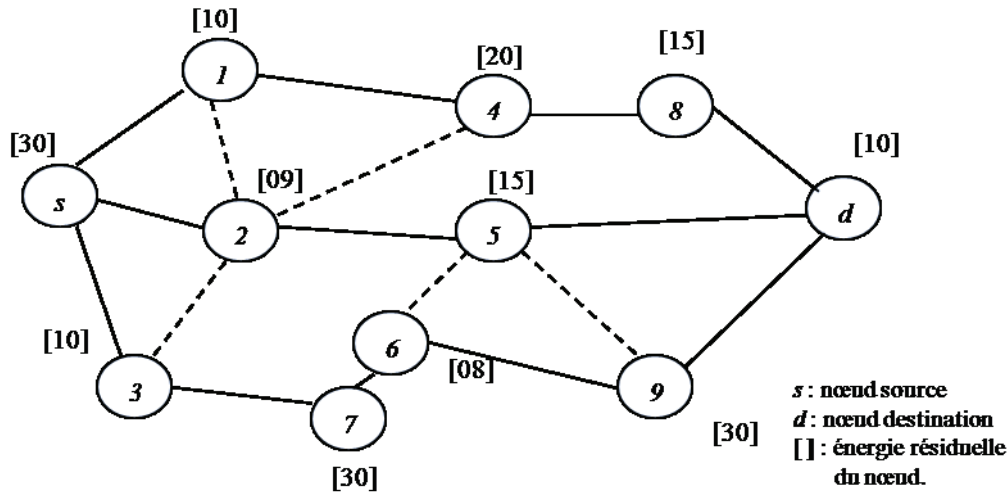


FIGURE 5.3 – AODVME+ : Exemple de découverte des chemins.

des chemins multiples du protocole AODVME+. A partir de la source s vers la destination d , trois chemins à nœuds disjoints sont découverts. Ce sont :

- Chemin 1 : $s-1-4-8-d$
- Chemin 2 : $s-2-5-d$
- Chemin 3 : $s-3-7-6-9-d$

A la fin du processus de découverte, le nœud source doit sélectionner un seul chemin pour la transmission des paquets de données. En appliquant l'algorithme 4, le nœud recevra pour chaque chemin sa moyenne des énergies et le minimum des énergies de ces nœuds. Suivant l'exemple de la figure 5.3, les chemins découverts auront des moyennes d'énergie et des minimums respectivement : Chemin1 (15 et 10), le Chemin2 (12 et 09), le Chemin3 (19.5 et 08). Le Chemin3 sera sélectionné en premier pour la transmission des paquets de données, puisqu'il a la meilleure somme pondérée suivant la formule 5.4, cette formule dépend de la valeur de α , fixée à 0,4 pour les raisons d'illustration de l'exemple.

1.4 Maintenance des chemins

Le processus de la maintenance des chemins est basé sur le principe utilisé par AODVM. Une fois que le chemin est sélectionné, il est utilisé par le nœud source pour acheminer les paquets des données. Quand un nœud intermédiaire détecte une défaillance de lien tout en essayant de transmettre un paquet au prochain saut, il envoie un message *RERR* (Route ERRor) au nœud source, pour signaler la rupture du lien intermédiaire sur le chemin. Quand le nœud source reçoit un message *RERR*, il supprime le chemin de sa table de routage et cherche un chemin alternatif dans sa liste pré-triée des chemins vers le nœud destination s'il est disponible ; sinon, il

déclenche un nouveau processus de découverte de chemin pour reprendre la transmission des paquets des données. Le nouveau chemin alternatif est sélectionné selon le principe décrit à la section 5.1.3.

2 Evaluation des performances du protocole

AODVME+

Dans cette section, nous présentons les résultats de simulation pour démontrer l'efficacité de notre protocole. Nous présentons, en premier lieu, les paramètres utilisés pour l'évaluation des performances, puis nous analysons l'influence des valeurs du coefficient α , afin de sélectionner la valeur la plus appropriée pour le reste des simulations. Nous évaluons notre protocole en le comparant avec deux protocoles de la littérature, à savoir AODVM [32] et MMRE [73]. Cette évaluation est accompagnée d'une analyse et une discussion des résultats.

2.1 Paramètres de performance

Nous allons évalué trois paramètres de performance. La consommation d'énergie : c'est la moyenne de l'énergie consommée par les nœuds participant à un transfert de paquets à partir du nœud source au nœud de destination durant toute la durée de la simulation. Le délai de bout en bout : c'est le délai moyen de transmission des paquets de données bien reçues durant toute la durée de la simulation. Le surcoût des paquets de routage : est mesuré comme étant le nombre de messages de contrôle transmis par chaque nœud durant toute la durée de la simulation.

2.2 Environnement de simulation

Les simulations présentées ont été réalisées avec le simulateur *ns-2* (*Network Simulator*) [77]. Le choix s'est porté sur cette version en raison des aides trouvées pour implémenter AODVME+ sur cette version. Les simulations ont été effectuées sur un ordinateur "dual core" d'une capacité de 1.5 GHZ pour chaque core, doté d'une mémoire RAM de 1 Go.

2.3 Evaluation des performances

Dans notre protocole, le choix du chemin est pris par le nœud source pour transmettre les paquets de données vers le nœud de destination. Cette décision est basée sur l'énergie minimale de nœuds et l'énergie moyenne d'un chemin (voir l'équation 5.4). L'équation 5.4 est principalement affectée par la valeur de α . Afin de trouver les valeurs appropriées de α , nous avons analysé l'impact de différentes valeurs de α sur les performances de notre protocole AODVME+. On a fait varier la valeur de α de 0,2 à 0,8.

Modèle du trafic	Constant bit rate (CBR)
Type d'interface réseau	WirelessPhy
Type MAC	IEEE 802.11
Modèle de mobilité	Random Waypoint
Surface de simulation	840mx840m
Portée de la transmission	250 m
Nombre de nœuds	30-50-70-90
Taille du paquet	512 octets
<i>RREQ_Wait_Time</i>	1.0 s
<i>RREP_Wait_Time</i>	1.0 s
Durée de simulation	300 s

TABLE 5.1 – AODVME+ : Paramètres de simulation.

Nous avons effectué nos simulations pour déterminer l'efficacité de notre protocole. Le principal objectif de ces simulations est d'analyser notre protocole en le comparant avec d'autres protocoles, principalement AODVM et MMRE. Les valeurs des paramètres de simulation sont résumées dans le tableau 5.1.

Pour évaluer AODVME+, chaque simulation a une durée de 300 secondes. Cette durée est suffisante pour atteindre un état stable et de négliger la phase de démarrage transitoire. Au cours de chaque simulation, des flux à débits constants (CBR) sont générés. Chaque connexion CBR produit 4 paquets par seconde avec une taille de paquet de 512 octets. Notre réseau est de 90 nœuds dans un environnement 840 m * 840 m. Le modèle Random Waypoint est utilisé pour simuler le mouvement des nœuds, et chaque nœud se déplace avec une vitesse choisie aléatoirement de 0 à 5 m/s, sans aucune pause. Le modèle radio utilisé a des caractéristiques similaires à une interface radio commerciale, de type Lucent's Wave LAN. Wave LAN est une interface sans fil de communication (média partagé) avec un débit binaire de 2 Mbit/s et une portée radio de 250 mètres [78].

Chaque simulation est effectuée dans un réseau avec un nombre différent de nœuds. Les mesures des paramètres (métriques) des performances sont obtenues en faisant 10 simulations en moyenne pour chaque résultat. Nous supposons qu'un nœud consomme 281,8 mW en réception d'un paquet, et 281,8 mW pour une l'émission d'un autre.

Il a été montré dans [79] qu'aucune optimisation de l'énergie de nœud ne peut être réalisée en présence d'overhearing ou dans l'état de repos. Pour cette raison, la consommation d'énergie pendant le temps d'inactivité ou overhearing n'est pas considérée dans ce modèle. Dans nos simulations, nous avons initialisé les énergies des nœuds aléatoirement entre 10 et 60 Joules (avec une distribution uniforme).

2.3.1 Consommation d'énergie

Les figures ci-dessous montrent l'énergie consommée dans différents tranches de temps de par les protocoles AODVME +, MMRE et AODVM. AODVME+ n'est pas performant au début de la simulation, mais il s'améliore avec le temps.

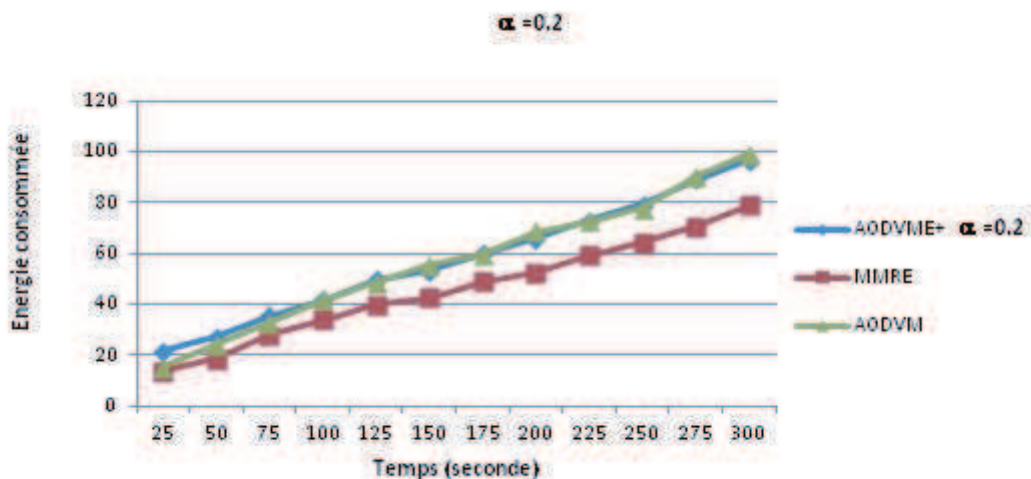


FIGURE 5.4 – Consommation d'énergie au cours du temps. ($\alpha=0.2$)

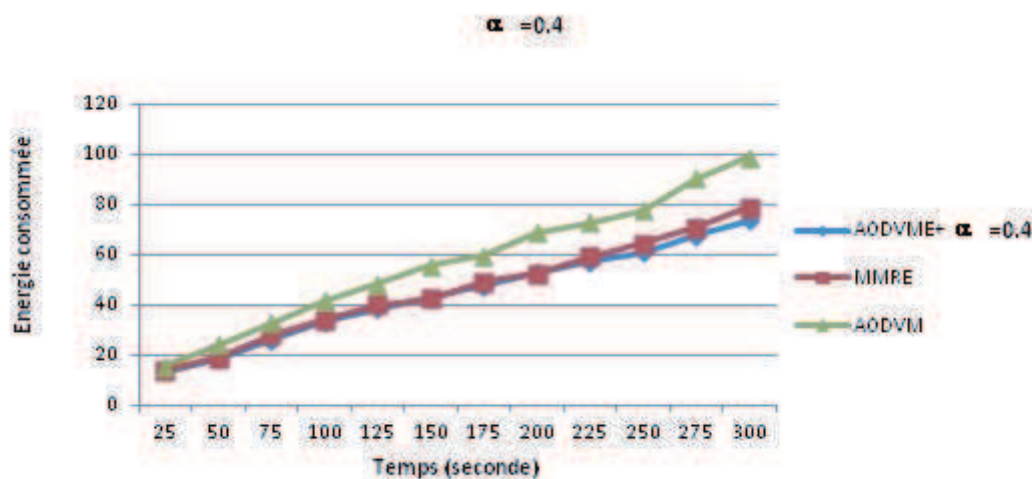


FIGURE 5.5 – Consommation d'énergie au cours du temps. ($\alpha=0.4$)

Dans un premier temps, il n'est pas meilleur que les autres protocoles, car initialement la majorité des paquets de données ne sont pas transmis, donc l'énergie totale de l'envoi et la réception des paquets n'est pas importante. Dans la figure 5.4,

on peut observer que, quand $\alpha = 0.2$, le protocole MMRE a la plus faible consommation par rapport à AODVME + et AODVM. Lorsque $\alpha = 0.4$ (figure 5.5) le protocole AODVME + consomme moins d'énergie, et offre une meilleure performance en termes de consommation d'énergie moyenne. La raison est que AODVME + se focalise non seulement sur la capacité d'énergie d'un nœud, mais aussi sur l'énergie moyenne du chemin. Cette combinaison garantit que les chemins avec l'énergie la plus élevée soient sélectionnés pour la transmission. Ainsi l'énergie est équilibrée à travers le réseau, ce qui réduit la consommation d'énergie inutile.

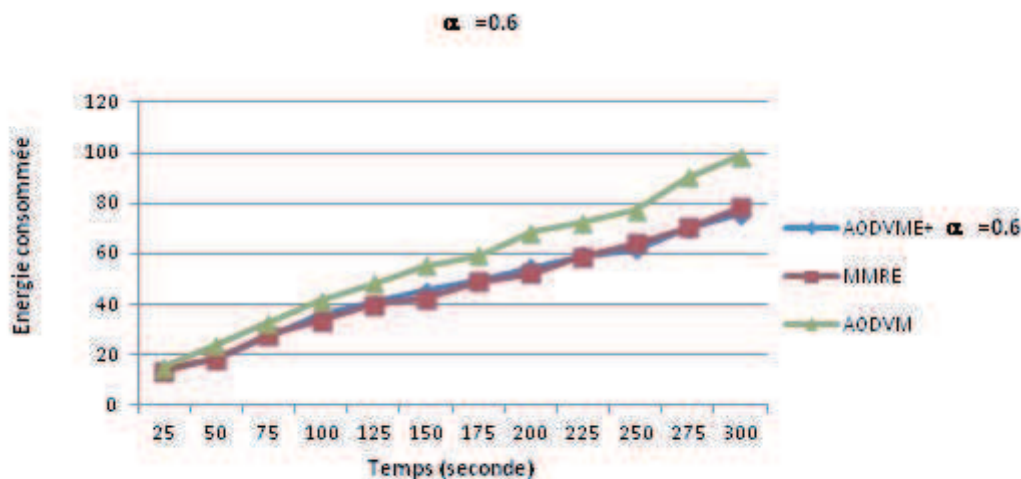


FIGURE 5.6 – Consommation d'énergie au cours du temps. ($\alpha=0.6$)

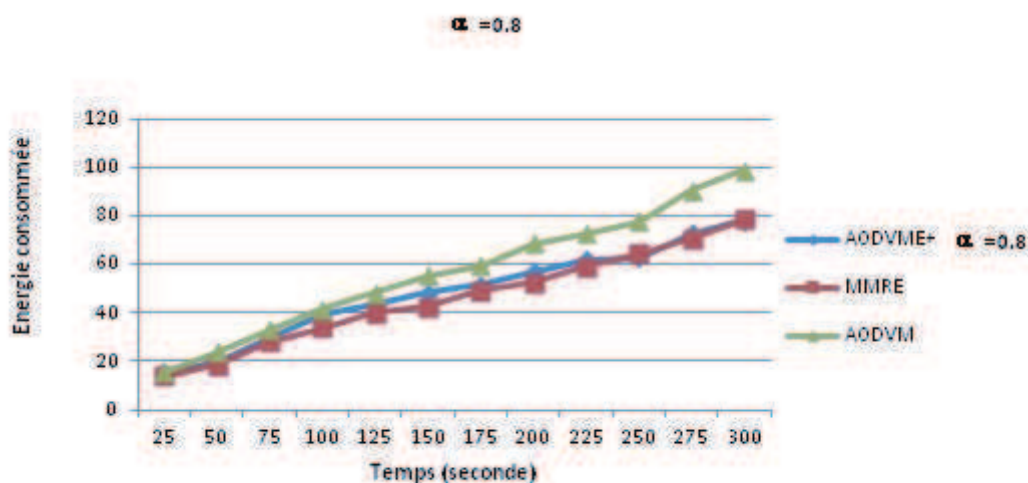


FIGURE 5.7 – Consommation d'énergie au cours du temps. ($\alpha=0.8$)

Enfin, nous notons que lorsque $\alpha=0.6$ ou 0.8 , la consommation d'énergie de AODVME + est proche de MMRE, car lorsque $\alpha=0.6$ ou 0.8 (voir figure 5.6 et 5.7), AODVME + favorise l'énergie résiduelle maximale des minimums des nœuds dans la sélection des chemins, ce qui est le principe du protocole MMRE.

2.3.2 Délai de bout en bout

La figure 5.8 présente le délai moyen de bout en bout des protocoles simulés avec une valeur de α fixé à 0.4 . Le délai de bout-en-bout moyen pour tous les protocoles testés augmente avec l'augmentation de la taille du réseau, mais le délai moyen de bout-en-bout de AODVME + est inférieure à celle de MMRE et AODVM.

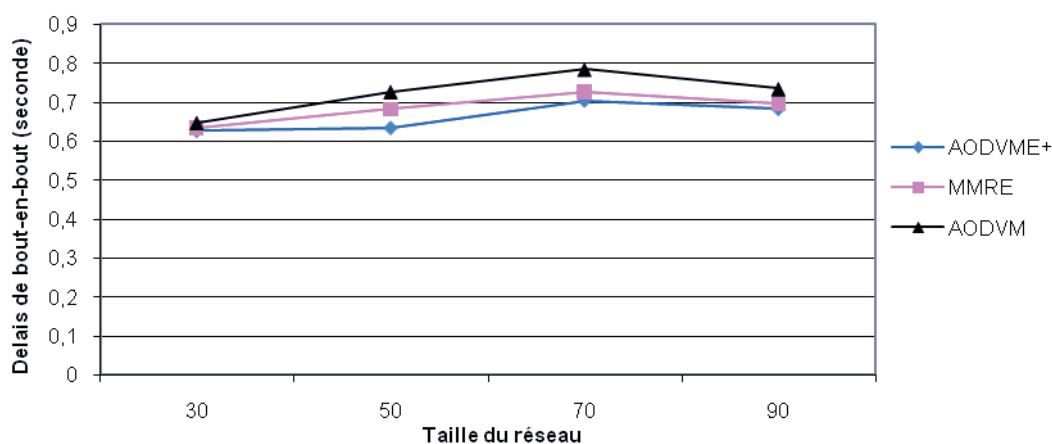


FIGURE 5.8 – Délai de but en bout vs la taille du réseau.

Lorsque le nombre de nœuds d'un réseau est compris entre 50 et 70, le délai du protocole AODVME + est (en moyenne) 11% plus bas que le délai du protocole AODVM et encore plus bas que celui du protocole de MMRE. Ceci parce que le protocole AODVME + sélectionne le chemin en fonction de l'énergie résiduelle de nœuds combinés avec l'énergie moyenne du chemin ; ce qui affecte considérablement le délai de bout en bout.

2.3.3 Surcout du routage

De la figure 5.9, nous constatons que le surcout de contrôle de routage est plus important pour le protocole AODVM. En effet, 590 messages de routage (dans le cas du réseau de 70 nœuds) ont été transmis dans le réseau au cours de la simulation en 300 secondes, par rapport à 430 messages pour le protocole AODVME+. Les protocoles AODVME+ et MMRE génèrent moins de trafic de contrôle par rapport à AODVM. L'augmentation du rendement de ces protocoles est due à la performance du processus de sélection des chemins alternatifs lorsqu'un lien échoue. Les protocoles AODVME+ et MMRE améliorent le nombre de paquets de contrôle par rapport à AODVM. Comme on le remarque, dans tous les cas, AODVME+ a le

plus petit nombre de paquets de contrôle par rapport à MMRE, ceci est justifié par l'équilibrage de l'utilisation des batteries des nœuds par le protocole AODVME+.

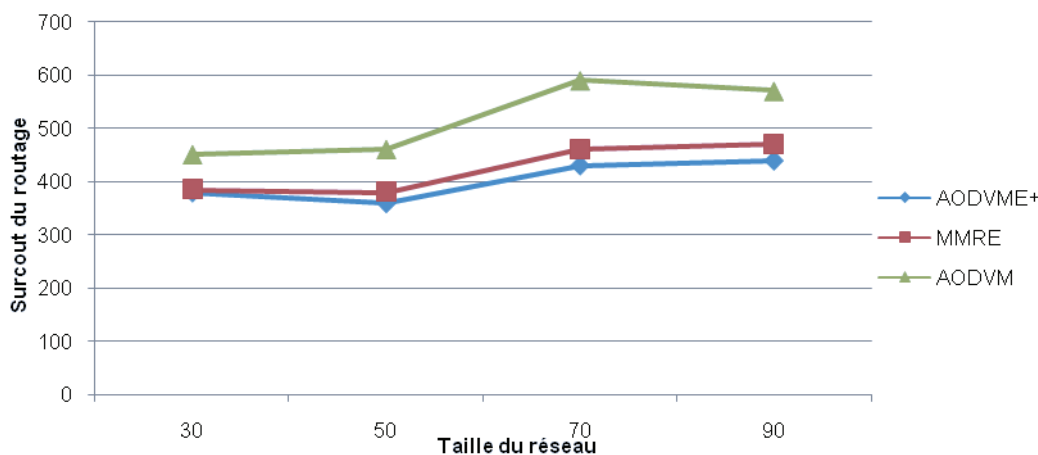


FIGURE 5.9 – Surcout du routage vs la taille du réseau.

3 Conclusion

A travers ce chapitre, nous avons proposé un nouveau protocole de routage pour les réseaux sans fil ad hoc, où les échecs des liens et les ruptures des chemins se produisent fréquemment. Cependant, les fréquents changements de la topologie épuisent rapidement les batteries des nœuds qui ont une puissance limitée, et cela peut provoquer une partition du réseau et diminue les performances du réseau. Notre approche consiste en une extension du protocole de routage AODVM. Les chemins alternatifs sont découverts à l'avance, et quand il y a une rupture d'un lien, l'un des chemins alternatifs est sélectionné (si au moins un chemin alternatif est disponible) pour transmettre les paquets de données. Le choix du chemin dépend d'une fonction pondérée concernant tous les chemins découverts. L'énergie résiduelle minimale et l'énergie moyenne du chemin, sont les critères utilisés pour calculer cette fonction pondérée. Nous avons évalué et étudié les performances de notre protocole de routage AODVME+ en le comparant avec les protocoles AODVM et MMRE. Les résultats de simulation ont démontré que notre protocole consomme moins d'énergie que AODVM et MMRE en l'améliorant à 21% et 4% respectivement. AODVME+ produit un délai de bout en bout moyen plus faible et minimise le surcout du routage en générant moins de messages de contrôle. Nous concluons que notre protocole AODVME+ conserve l'énergie consommée et améliore les performances d'un réseau ad hoc. Toutefois les solutions multichemins à nœuds disjoint produisent un nombre de chemins découverts réduits constituant une contreperformance des protocoles utilisant ce principe. La disjonction au niveau des liens est une problématique à explorer sur des réseaux mobiles ad hoc. Dans le chapitre suivant nous présenterons un nouveau protocole multichemin à liens disjoints avec conservation d'énergie pour les réseaux ad hoc.

Chapitre 6

Protocole de routage multichemin

AOMR-LM

Dans les réseaux ad hoc, chaque nœud dispose d'une batterie d'alimentation électrique et donc d'un approvisionnement énergétique limité. Au fil du temps, les nœuds épuisent leurs réserves d'énergie et sont finalement retirés du réseau, ce qui limite l'acheminement des paquets. Certains nœuds épuisés peuvent être critiques pour la transmission de paquets quand ils appartiennent au seul chemin du nœud source vers le nœud de destination. Pour résoudre cette inefficacité, nous avons proposé un protocole qui prolonge la durée de vie du réseau. Nous avons utilisé les niveaux de nœuds d'énergie pour classer les nœuds des chemins entre un nœud source et un nœud destination. Le niveau d'énergie d'un nœud est calculé en fonction de l'énergie résiduelle du nœud ; ces informations sont collectées lors de la découverte de plusieurs chemins en utilisant le même principe que la découverte d'un chemin unique. Deux paramètres sont introduits, un seuil d'énergie et un coefficient, pour définir la classe du nœud. Notre protocole sélectionne les nœuds de la même classe pour construire un chemin donné. A la fin de l'opération de découverte des chemins, le nœud source aura un ensemble de chemins homogènes en termes d'énergie. Les chemins les mieux classés sont sélectionnés pour équilibrer l'énergie consommée entre les différents chemins. La technique proposée supprime les nœuds avec un niveau d'énergie critique et fait en sorte qu'ils ne participent pas au routage, sinon on pourrait avoir plusieurs liens brisés causés par l'épuisement ces nœuds.

1 Justification du choix du protocole de base :

AOMDV

Les protocoles de routage existants dans les réseaux ad hoc tels que les protocoles proactifs et réactifs peuvent être modifiés pour intégrer une fonction contrôlant

l'énergie qui prolonge la durée de vie du réseau et optimise la consommation d'énergie. Nous avons choisi AOMDV (Ad hoc On demand Multipath Distance Vector) [20] comme protocole de base. Il a été prouvé que AOMDV peut être un bon protocole qui utilise le multichemin ; tous ses chemins sont disjoints et peuvent garantir un routage sans boucle et permet d'utiliser seulement les chemins alternatifs ayant le nombre de sauts le plus bas. Le protocole de AOMDV lance le processus de découverte de chemin entre le nœud source et le nœud destination, lorsque la source a besoin d'envoyer un premier paquet de données. Dans ce processus, les paquets (*RREQ* : *Route REQuest*) sont diffusés par chaque nœud du réseau. La destination envoie un message de réponse (*RREP* : *Route REPLY*) pour tous les paquets reçus *RREQ*. Un nœud intermédiaire transmet un paquet *RREP* reçu à son voisin jusqu'au nœud source. Ce processus de découverte peut être exploité pour collecter des informations des nœuds, tels que l'énergie résiduelle, le niveau de charge, etc.

Plusieurs études [80-82] ont montré que le protocole AOMDV est plus robuste et donne de meilleurs résultats dans la plupart des scénarios simulés. Donc nous avons choisi ce protocole à la place de tous les autres protocoles réactifs (tels que SMR [38]) comme référence.

2 Le protocole AOMR-LM : Ad hoc On-demand Multipath Routing with Lifetime Maximization

Le protocole AOMR-LM [83] est un protocole de routage multichemin réactif, avec un nouveau mécanisme de classification de chemins en fonction du niveau d'énergie des nœuds formant ces chemins, qui peut être élevé, moyen ou bas. L'idée est de construire des chemins homogènes en termes d'énergie, ce qui équilibre la consommation de l'énergie des nœuds et évite les ruptures des liens en raison de l'épuisement des énergies des nœuds. Le protocole définit un seuil d'énergie et un coefficient de pondération, utilisés pour définir la classe de chaque nœud, la détermination de cette classe est nécessaire lors de la transmission du paquet de réponse. Le concept de AOMR-LM prolonge la durée de vie du réseau et améliore la performance énergétique dans les réseaux mobiles ad hoc.

Dans le protocole de routage AODV, la maintenance des chemins se fait par l'envoi périodique de messages courts (appelés *HELLOs*). Si trois messages *HELLO* consécutifs ne sont pas reçus par le nœud voisin, le nœud considère que le lien est rompu. Lorsque le nœud reçoit un paquet de données qui devrait être acheminé via ce lien brisé, un message d'erreur (*RERR*, *Route ERRor*) est renvoyé à la source indiquant la rupture du lien. Dans ce cas, une nouvelle opération de découverte de chemin est nécessaire, ceci ajoute un coût supplémentaire en termes de délai, de débit, de surcôt de messages de contrôle, et de consommation d'énergie pour le transfert de messages supplémentaires de contrôle. Parmi les raisons de défaillance d'un nœud est l'épuisement de l'énergie de sa batterie. En raison de ces défaillances de nœuds, les liens d'un chemin peuvent devenir temporairement indisponibles et rendent le chemin non valide. L'algorithme de routage décide de quel nœud du réseau doit figurer dans un chemin particulier. La sélection de nœuds sans la prise en compte

de l'énergie de nœud lors de la détermination des chemins conduit à un déséquilibre du niveau d'énergie dans le réseau. Les nœuds ayant une énergie réduite peuvent rapidement devenir critiques, tandis que d'autres nœuds restent intacts. Cela se traduira par l'épuisement prématuré de certains nœuds. A cet effet, un routage multichemin peut être efficace, car il répartit la charge de trafic entre plusieurs nœuds et proportionnellement à leurs énergies résiduelles. Lorsque la consommation d'énergie entre les nœuds est plus équilibrée, le temps moyen de défaillance d'un nœud est prolongé et la durée de vie du réseau se prolonge ainsi.

Le protocole AOMR-LM proposé est un protocole multichemin réactif. Notre mécanisme de sélection conserve l'énergie résiduelle de nœuds et équilibre l'énergie consommée. Cela prolonge la durée de vie du réseau et améliore les performances du réseau. Les performances de AOMR-LM sont comparées avec d'autres protocoles à savoir AOMDV[20] et ZD-AOMDV[75]. Notre protocole donne de meilleurs résultats comparativement à ces protocoles. Dans ce qui suit, nous définissons notre problématique, puis nous détaillerons les différents processus de notre solution, tels que la découverte des chemins, la sélection des chemins, la transmission des paquets de données, et la procédure de maintenance des chemins ainsi que une discussion de tous nos résultats.

2.1 Définition du problème

Un réseau ad hoc sans fil est représenté par un graphe non orienté, $G = (V, E)$, où V est l'ensemble des nœuds de réseau et E est l'ensemble des liens bidirectionnelles de réseau. Soit $w(u)$, $u \in V$, l'énergie résiduelle du nœud u , et soit $c(u, v)$, $(u, v) \in E$, l'énergie nécessaire pour transmettre un paquet de nœud u au nœud v . Nous supposons que $c(u, v) = c(v, u)$ pour tout $(u, v) \in E$. Nous considérons que P est un ensemble de chemins de G , donc $P(u_0, u_n)$ est l'ensemble des chemins entre le nœud u_0 et le nœud u_n . Soit $P_i(u_0, u_n) = u_0, u_{i1}, \dots, u_{im}, \dots, u_n$, le i ème chemin dans $P(u_0, u_n)$, le nœud source est noté par u_0 ou u_{i0} et le nœud de destination est noté par u_n ou u_{in} . On définit le nombre de nœuds sur le chemin $P_i(u_0, u_n)$ par $\overline{P_i(u_0, u_n)}$ et le nombre de noeuds de tous les chemins de la phase de découverte par $\overline{P(u_0, u_n)}$. On note $e_{sum}(P_i(u_0, u_n))$, la somme de l'énergie résiduelle d'un chemin $P_i(u_0, u_n)$, est donnée par :

$$e_{sum}(P_i(u_0, u_n)) = \sum_{u_{i0}}^{u_{in}} w(u_{ij}) \quad (6.1)$$

Nous considérons $e_{sum}(P(u_0, u_n))$ comme étant la somme des énergies résiduelles de tous les chemins appartenant à l'ensemble $P(u_0, u_n)$:

$$e_{sum}(P(u_0, u_n)) = \sum_{P(u_0, u_n)} e_{sum}(P_i(u_0, u_n)) \quad (6.2)$$

Soit $e_{average}(P_i(u_0, u_n))$ l'énergie résiduelle moyenne d'un chemin, $e_{average}(P_i(u_0, u_n))$ est donné par :

$$e_{average}(P_i(u_0, u_n)) = \frac{e_{sum}(P_i(u_0, u_n))}{\overline{P_i(u_0, u_n)}} \quad (6.3)$$

L'énergie résiduelle moyenne du réseau est désignée par $e_{averageNet}(P(u_0, u_n))$, qui représente l'énergie résiduelle moyenne des nœuds qui ont participé à la phase de découverte des chemins multiples entre un nœud source u_0 et un nœud de destination u_n , $e_{averageNet}(P(u_0, u_n))$ est donné par :

$$e_{averageNet}(P(u_0, u_n)) = \frac{e_{sum}(P(u_0, u_n))}{P(u_0, u_n)} \quad (6.4)$$

On définit , le niveau d'énergie du nœud u_j pendant un processus de découverte des chemins entre un nœud source u_0 et u_n nœud de destination un, il est donné par :

$$e_{level}^{u_0, u_n}(u_j) = \frac{w(u_j)}{e_{averageNet}(P(u_0, u_n))} \quad (6.5)$$

Etant donné un nœud source s , un nœud de destination d , et un paquet de données qui doit être acheminé, la source initie la découverte des chemins multiples. Cette découverte fournit de multiples chemins disjoints avec des propriétés d'efficacité énergétique. La sélection d'un chemin est basé sur le niveau d'énergie résiduelle de leurs nœuds pour préserver l'énergie des nœuds, et pour équilibrer l'énergie consommée sur l'ensemble des chemins.

2.2 Découverte des chemins multiples

Le processus de découverte des chemins suit le même principe que celui du protocole AOMDV décrit précédemment. Plusieurs changements ont été nécessaires dans la procédure de découverte des chemins du protocole AOMDV, afin de permettre le calcul de la moyenne d'énergie résiduelle du réseau et de définir les niveaux d'énergie de chaque nœud. Chaque paquet *RREQ* a un champ supplémentaire, appelé $e_{sum}(P_i(u_0, u_j))$, qui représente la somme de l'énergie résiduelle du nœud source s au nœud courant j , jusqu'au nœud destination d . Un autre champ $w(u)$ est ajouté, de sorte qu'un nœud ait l'information sur l'énergie résiduelle de ses voisins. Quand un nœud intermédiaire i reçoit un paquet *RREQ* du nœud j , il augmente le champ $e_{sum}(P_i(u_0, u_j))$ par la valeur de son énergie résiduelle. Ce traitement doit prendre en compte le numéro de séquence (*Seq_num*) afin de garantir la fraîcheur des chemins et le nombre maximum des sauts (*Advertised_hopcount*) pour tous les chemins [32]. Tout paquet *RREQ* reçu avec un numéro de séquence inférieur au numéro de séquence du *RREQ* précédent à partir de la même source vers la même destination est mise à l'écart. Le même processus est répété jusqu'à ce que le paquet *RREQ* atteigne sa destination finale ; voir l'algorithme 5.

```

Begin
If ( $seqnum_i^d < seqnum_j^d$ ) then
     $seqnum_i^d \leftarrow seqnum_j^d$ 
    If ( $i \neq d$ ) then
         $e_{sum}(P_i(u_0, u_j)) \leftarrow e_{sum}(P_i(u_0, u_j)) + w(u_i)$ 
         $advertised\_hopcount_i^d \leftarrow \infty$ ;
         $route\_list_i^d \leftarrow NULL$ ;
         $insert(j, advertised\_hopcount_j^d + 1, w(u_i))$  into  $route\_list_i^d$ ;
    else
         $advertised\_hopcount_i^d \leftarrow 0$ ;
    end If
else
    If ( $(seqnum_i^d = seqnum_j^d)$  and
     $((advertised\_hopcount_i^d, i) > (advertised\_hopcount_j^d, j))$ ) then
         $e_{sum}(P_i(u_0, u_j)) \leftarrow e_{sum}(P_i(u_0, u_j)) + w(u_i)$ 
         $insert(j, advertised\_hopcount_j^d + 1, w(u_i))$  into  $route\_list_i^d$ ;
    end If
end If
End.
    
```

Algorithm 5: Processus de découverte de chemins dans le protocole AOMR-LM.

La figure 6.1 représente les entrées de la table de routage d'un nœud. Pour chaque destination connue par le nœud existe une entrée. *Route_list* contient tous les nœuds voisins d'un nœud connus menant vers le nœud de destination. Chaque voisin menant au nœud de destination est identifié par *nexthop*, et le champ *hopcount* est le nombre de sauts nécessaires pour atteindre la destination en utilisant ce voisin. Nous avons ajouté deux nouveaux champs, $w(u)$ et *marked_node*, dans le champ *Route_list*. Le champ $w(u)$ désigne l'énergie résiduelle d'un nœud et le champ *marked_node* indique si un nœud a été sélectionné ou non par un chemin inverse.

2.3 Sélection des chemins multiples

Après la diffusion du paquet *RREQ* par le nœud source, le nœud de destination attend pendant une certaine période de temps *RREQ_Wait_time* (après la récep-

<i>Destination</i>
<i>Sequence_number</i>
<i>Advertised_hopcount</i>
<i>Route_list</i> {(<i>nexthop1</i> , <i>hopcount1</i> , $w(u_1)$, <i>marked_node</i>), (<i>nexthop2</i> , <i>hopcount2</i> , $w(u_2)$, <i>marked_node</i>), ...}
<i>Expiration_timeout</i>

FIGURE 6.1 – Structure de la table de routage du protocole AOMR-LM.

tion du premier *RREQ*) avant de commencer la procédure de sélection des chemins. Cette période d'attente peut générer un délai supplémentaire pour le routage multichemin (voir la figure 6.2).

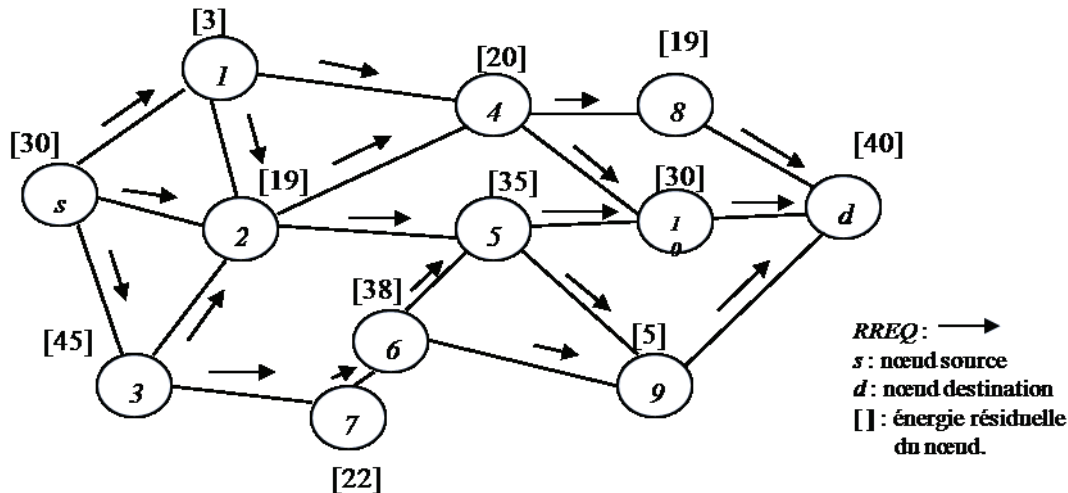


FIGURE 6.2 – Réception des *RREQ*s par la destination

Lorsque cette période de temps expire, le nœud de destination génère un paquet réponse (*RREP*) et l'envoie à la source. Dans le protocole AOMR-LM, l'énergie d'un nœud $w(u_j)$ est prise en considération lors de transmission en sens inverse des paquets *RREP*. Donc, la sélection du nœud suivant vers lequel le paquet *RREP* est transmis, dépend de la classe du nœud. Pour déterminer l'appartenance d'un nœud à l'une des trois classes, on définit deux seuils, noté α et β . α étant le seuil inférieur qui sépare la classe basse de la classe moyenne, β le seuil supérieur qui sépare la classe moyenne de la classe supérieure. Le seuil d'énergie est prédéfinie et identique

pour tous les nœuds du réseau. Notre solution suppose que le seuil est l'énergie résiduelle moyenne du réseau. Si l'énergie de nœud u_j (i.e. $w(u_j)$) est au-dessus du seuil ($w(u_j) \geq \beta$), le nœud u_j a une forte probabilité d'être en mesure de transmettre des paquets de données, et donc il est classé dans un niveau élevé. Lorsque $w(u_j) < \beta$, la définition de la classe de nœud dépend du seuil β et du coefficient α , qui sont nécessaires pour la décision de transmission du paquet réponse. Le coefficient α est introduit afin de déterminer la capacité du nœud de supporter le trafic de données en termes d'énergie. Ce coefficient décide de la participation de nœud nœud dans le routage.

Lorsque $w(u_j) < \beta$ alors $\frac{w(u_j)}{\beta} < 1$. Soit $\alpha < 1$; on obtient :

$$\alpha * \frac{w(u_j)}{\beta} < 1 \quad (6.6)$$

Pour analyser les valeurs du coefficient α , nous définissons $T_{Net}(P(u_0, u_n))$, qui décrit la participation des K nœuds d'un réseau dans le transfert de données sur l'ensemble des chemins $P(u_0, u_n)$. Nous aurons :

$$T_{Net}(P(u_0, u_n)) = \prod_{j=1}^K \alpha * \frac{w(u_j)}{\beta} \leq \left(\frac{\alpha}{\beta}\right)^K * \prod_{j=1}^K w(u_j)$$

En appliquant l'inégalité des moyennes arithmétiques et géométriques :

$$\frac{\alpha_1 x_1 + \alpha_2 x_2 + \dots + \alpha_n x_n}{\alpha} \geq \sqrt[\alpha]{x_1^{\alpha_1} x_2^{\alpha_2} \dots x_n^{\alpha_n}}$$

où $\alpha = \alpha_1 + \alpha_2 + \dots + \alpha_n > 0$

Si on prend $n = k$, $\alpha_i = 1$ and $x_i = w(u_i)$

on obtient

$$\left(\prod_{j=1}^K w(u_j)\right)^{\frac{1}{K}} \leq \left(\frac{1}{K} \sum_{j=1}^K w(u_j)\right)$$

En suite, nous avons

$$\left(\frac{\alpha}{\beta}\right)^K * \prod_{j=1}^K w(u_j) \leq \frac{\alpha^K}{\beta^K} * \left(\frac{1}{K} \sum_{j=1}^K w(u_j)\right)^K$$

ce qui prouve que

$$T_{Net}(P(u_0, u_n)) = \prod_{j=1}^K \alpha * \frac{w(u_j)}{\beta} \leq \left(\frac{\alpha}{\beta}\right)^K * \left(\frac{1}{K} \sum_{j=1}^K w(u_j)\right)^K$$

Pour simplifier l'analyse, on pose $\sum_{j=1}^K \frac{w(u_j)}{K} \approx e_{averageNet}(P(u_0, u_n))$, étant donné que la somme de l'énergie résiduelle de nœuds, divisé par le nombre de nœuds donne une valeur approchée de celle calculée par l'équation 6.4. On considère que si l'énergie résiduelle d'un nœud est plus grande que l'énergie résiduelle moyenne du réseau, alors ce nœud possèdera une énergie suffisante et présente une forte probabilité de transmission de plusieurs paquets de données avant son épuisement. Ce cas correspond à ($w(u_j) \geq \beta$), donc :

$$\beta = e_{averageNet}(P(u_0, u_n)) \quad (6.7)$$

Ainsi :

$$T_{Net} = \prod_{j=1}^K \alpha * \frac{w(u_j)}{\beta} \leq \alpha^K$$

On obtient :

$$\alpha \geq (T_{Net})^{\frac{1}{K}} \quad (6.8)$$

Suivant (6.6), dans le cas où $w(u_j) < \beta$ et on a $\alpha * \frac{w(u_j)}{\beta} < 1$, on obtient :

$$\alpha \leq \frac{\beta}{w(u_j)} \quad (6.9)$$

De (6.8) et (6.9), on aura :

$$\alpha \in \left[(T_{Net})^{\frac{1}{K}}, \frac{\beta}{w(u_j)} \right] \quad (6.10)$$

Après avoir reçu un paquet de réponse *RREP*, le nœud intermédiaire vérifie si le paquet *RREP* a été reçu. Si le message *RREP* a déjà été reçu, le nœud le supprime directement ; sinon, il calcule les niveaux d'énergie de tous les nœuds voisins et détermine leurs classes. Notre idée est de classer les nœuds en fonction de leurs niveaux d'énergie. Lorsque la valeur de $w(u_j)$ du nœud j est élevé, la fonction (6.7) est utilisée, ce cas est réduit à $e_{level}(u_j) \geq 1$. Le choix des niveaux dépend de β et α , parce que si le niveau du nœud d'énergie est supérieure à β (il est démontré que c'est l'énergie moyenne du réseau), la classe du nœud est considéré comme élevé ce qui permet de supporter le routage des paquets et un trafic de données important, pour une durée de temps considérable. Si le niveau du nœud d'énergie est inférieur à β , le coefficient α est introduit afin de déterminer la capacité du nœud de supporter le trafic de données ; donc si le niveau d'énergie du nœud est supérieur à α le nœud est affecté à la classe moyenne ; sinon la classe du nœud est considérée comme basse. Trois classes de nœuds sont définies :

1. **Basse** : Le niveau d'énergie du nœud est inférieure à α .
2. **Moyenne** : Le niveau d'énergie du nœud est compris entre α et β .
3. **Elevée** : Le niveau d'énergie du nœud est supérieure à β .

Dans le protocole AOMDV, le nœud intermédiaire établit un chemin inverse en sélectionnant le premier nœud voisin du champ *Route_list* de la table de routage. Dans notre protocole, le nœud intermédiaire détermine la classe de chaque nœud voisin et transfère le paquet *RREP* au nœud voisin de la même classe (voir la figure 6.3). Si les classes de l'ensemble des nœuds voisins sont différentes de la classe du nœud intermédiaire, le nœud voisin avec la classe la plus proche de la classe du nœud intermédiaire est sélectionnée. Dans l'étape (a) de l'exemple montré dans la figure 6.3, le nœud destination émet le paquet *RREP* aux voisins 8 et 10, mais il ignore le nœud 9, parce qu'il appartient à une classe plus basse (son niveau d'énergie relatif est très bas). Dans l'étape (b), le nœud 10 choisit en premier lieu le nœud 5 pour lui transmettre *RREP*, puisque ils appartiennent à la même classe ; par contre le 8 n'a qu'un seul voisin le nœud 4, donc il le choisit pour lui répondre par *RREP*.

Ce processus est répété par chaque nœud intermédiaire jusqu'au nœud source. A la fin du processus, toutes les classes des nœuds d'un même chemin découvert seront

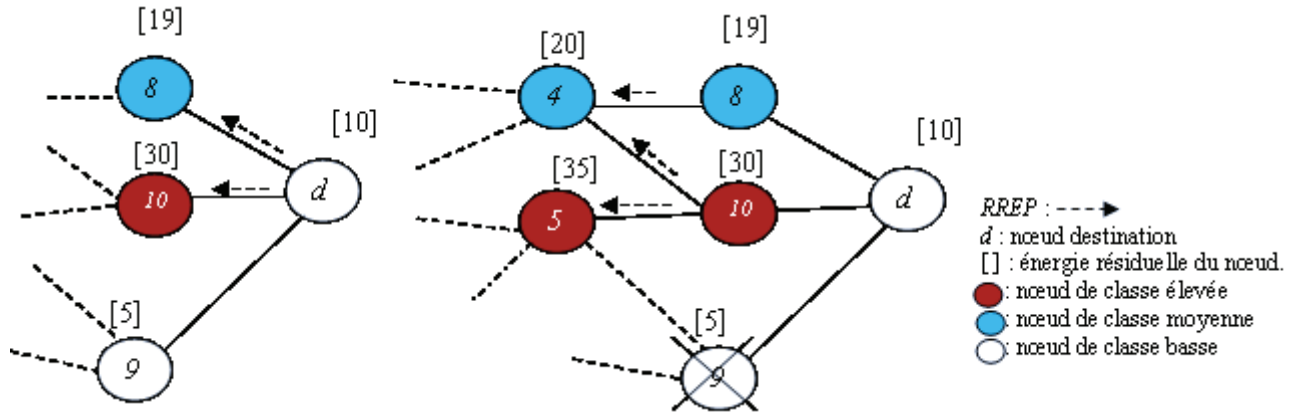


FIGURE 6.3 – Le choix des nœuds voisins pour l’envoi des *RREPs*.

identiques ou proches, donc on obtient des chemins homogènes. La figure 6.4 montre les chemins découverts par notre protocole. Suivant l’exemple l’exemple de la figure 6.5, les deux chemins découverts sont : $s - 3 - 7 - 6 - 5 - 10 - d$ et $s - 2 - 4 - 8 - d$. Le choix des

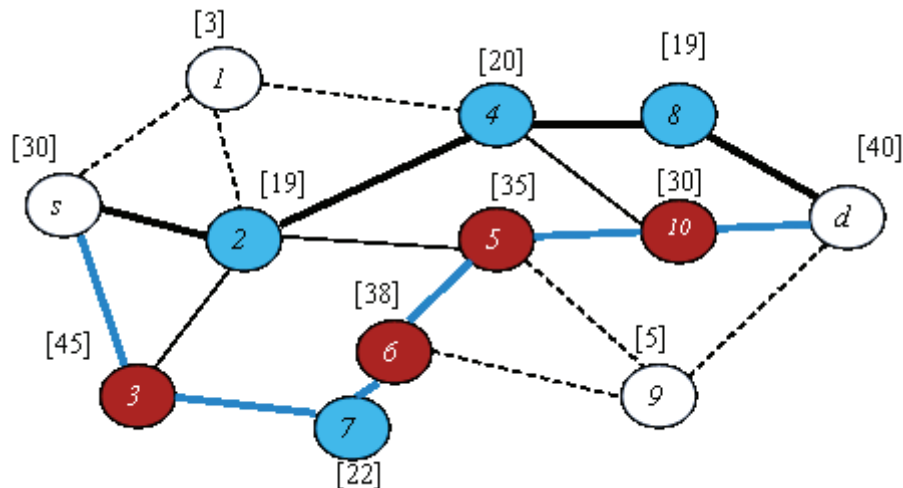


FIGURE 6.4 – Les chemins découverts par le protocole AOMR-LM.

Pour assurer des chemins à liens disjoints, un nœud intermédiaire marque chaque nœud voisin sélectionné par un chemin inverse ; cette propriété est assurée par le champ de la table de routage *marked_node* (voir la figure 6.1) ; en outre, notre solution évalue le niveau de chaque nœud de l’énergie, de sorte que l’épuisement d’un nœud est considéré et n’affecte pas les liens des chemins. Les trajectoires des paquets *RREP* peuvent se croiser dans des nœuds intermédiaires, mais chacun prend un chemin inverse différent vers le nœud source pour assurer la disjonction des liens des chemins. Lorsque le nœud source reçoit le premier paquet *RREP* pour une destination donnée, il attend un temps spécifique, appelé *RREP_Wait_time*,

avant de choisir le meilleur chemin ; ce temps introduit potentiellement un délai supplémentaire pour le routage multichemin.

2.4 Transmission de données

Le choix du meilleur chemin entre un nœud source s et le nœud destinataire d dépend des niveaux d'énergies des nœuds. AOMDV transmet les paquets de données en utilisant le premier chemin dans la liste des chemins disponibles quand plusieurs chemins sont découverts. Dans notre protocole AOMR-LM, les chemins sont regroupés par classe (élevée, moyenne et basse), selon les classes des nœuds qui constituent ces chemins. Si un chemin est constitué de nœuds de classes différentes, il est classé selon le plus grand nombre de type de classe présente sur ce chemin. Les chemins de la classe élevée sont choisis en premier pour transmettre les paquets de données. Dans ce cas, tous les paquets allant vers la même destination suivent le même chemin tant qu'il n'y a pas de défaillance de liens dans ce chemin. Une fois la classe élevée épuisée, l'algorithme sélectionne un chemin de la classe moyenne. Ce processus équilibre la consommation d'énergie entre les nœuds. Ceci prolonge la durée de vie du réseau. Un seul chemin est sélectionné pour une transmission de paquets de données entre un nœud source et un nœud de destination. Si le chemin sélectionné échoue, le nœud source reçoit un paquet d'erreur *RERR* ; dans ce cas, un chemin est sélectionné à partir de la plus haute classe disponible. Une nouvelle découverte de multichemins est lancée, s'il n'y a plus de chemins disponibles ; ce processus de découverte est décrit dans la section 6.2.2.

2.5 Maintenance des chemins

La détection des ruptures des chemins du protocole AOMR-LM est similaire à la détection des ruptures des chemins dans AOMDV. Le processus de maintenance est lancé lorsqu'une liaison échoue entre deux nœuds le long d'un chemin depuis un nœud source vers un nœud de destination.

Quand un nœud voisin ne répond pas aux trois messages *HELLO* successifs envoyés par un nœud, la liaison est considérée comme rompue. Si un nœud détecte une défaillance d'un lien dans un chemin actif, il efface le chemin de sa table, puis envoie un message d'erreur *RERR* (Route ERRor) au nœud source. Chaque nœud intermédiaire renvoie ce message *RERR* le long du chemin inverse jusqu'au nœud source. Quand un nœud source reçoit un message d'erreur *RERR*, il efface le chemin de sa table et sélectionne un autre chemin alternatif vers le nœud de destination s'il en existe ; sinon il déclenche un nouveau processus de découverte de chemin pour reprendre la transmission de données. Un autre chemin est sélectionné, comme décrit à la section 6.2.3.

3 Evaluation des performances du protocole AOMR-LM

Dans cette section, nous présentons les résultats de simulation pour démontrer l'efficacité de notre protocole. Nous présentons d'abord les paramètres utilisés pour l'évaluation des performances, puis nous analysons les valeurs de coefficient α pour sélectionner la valeur la plus appropriée pour le reste des simulations ; cette valeur permet de classer les nœuds en fonction de leurs niveaux d'énergie. Enfin, nous évaluons notre protocole en le comparant avec deux protocoles de la littérature, à savoir AOMDV et ZD-AOMDV. Cette évaluation est accompagnée d'une analyse et une discussion des résultats.

3.1 Paramètres de performance

Nous avons évalué trois indicateurs de performance clés. La durée de vie du réseau, qui peut être définie de trois façons différentes [84] : le temps nécessaire pour épuiser la batterie du premier nœud de réseau, le temps nécessaire pour épuiser la batterie de N nœuds de réseau, et le moment où la batterie du dernier nœud du réseau est épuisée. Nous choisissons la seconde façon.

La consommation d'énergie est la moyenne de l'énergie consommée par les nœuds participant à un transfert de paquets à partir du nœud source au nœud de destination.

Le délai de bout en bout est le délai moyen de transmission des paquets de données recus avec succès. Le délai de transmission est défini comme la différence entre le temps d'arrivée d'un paquet à sa destination et le temps de son émission par le nœud source.

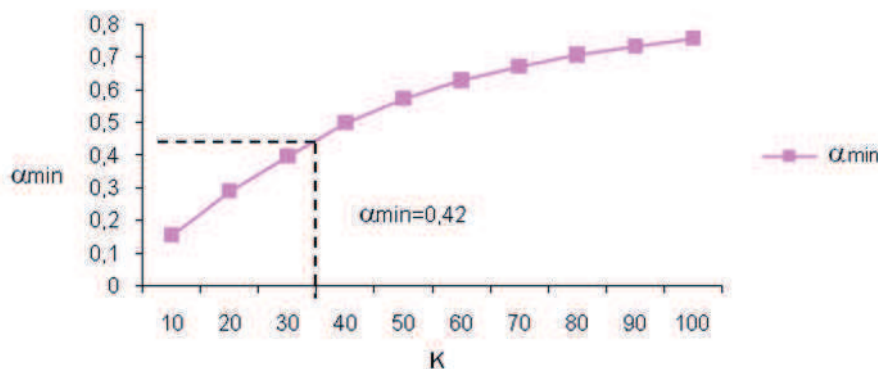
3.2 Analyse du coefficient α

Dans notre protocole le chemin utilisé par un nœud source pour transmettre des paquets de données vers un nœud de destination est basé sur la classe du chemin, qui est fonction des classes de nœuds constituant ce chemin. Dans notre protocole, la définition de classe d'un nœud est principalement affectée par la valeur de α . Afin de trouver les valeurs appropriées de α , nous appliquons l'équation (6.8) pour trouver les valeurs minimales. Nous supposons que si le nombre de nœuds participant à la transmission de données est proche du nombre de nœuds du réseau K , T_{Net} tend vers zéro. Le tableau 6.1 montre les valeurs de α_{min} mesurées en variant K de 10 à 100 nœuds.

K	10	20	30	40	50	60	70	80	90	100
α_{min}	0.155	0.289	0.396	0.499	0.574	0.629	0.672	0.707	0.734	0.757

TABLE 6.1 – Mesure d α_{min} relative au nombre de nœuds du réseau K .

Dans la figure 6.5, nous pouvons constater que α_{min} augmente avec l'augmentation de K , et tend vers zéro lorsque K devient faible. Le nombre de nœuds pour le transfert de paquets de données à partir d'un nœud source au nœud de destination est compris entre 30 et 40 (cela correspond à K du tableau 6.1), pour un réseau de 190 nœuds [85]. Pour la simulation de notre protocole AOMR-LM, pour un réseau de 190 nœuds, notre choix s'est porté sur $\alpha_{min} = 0.42$.

FIGURE 6.5 – α_{min} versus K .

4 Evaluation des performances

Pour évaluer AOMR-LM, nous avons utilisé le simulateur de réseau ns-2 [76]. Le principal objectif de ces simulations est d'analyser notre protocole en le comparant avec d'autres protocoles, principalement AOMDV [20] et ZD-AOMDV [75]. Les nœuds source et destination sont choisis aléatoirement ainsi que les positions des nœuds du réseau. Les valeurs des paramètres de simulation sont résumées dans le tableau 6.2.

Modèle du trafic	Constant Bit Rate (CBR)
MAC type	IEEE 802.11
Marque commerciale de l'interface réseau	Lucent Wave LAN
Modèle de Mobilité	Random Waypoint
Surface de simulation	840 m*840 m
Portée de transmission	250 m
Nombre de nœuds	30 à 190
Taille du paquet	512 bytes
<i>RREQ_Wait_Time</i>	1.0 s
<i>RREP_Wait_Time</i>	1.0 s
Consommation en réception	281.8 mW
Consommation en émission	281.8 mW
Energies des nœuds	Entre 10 et 60 J, affectées aléatoirement

TABLE 6.2 – Paramètres de simulation du protocole AOMR-LM.

4.1 Durée de vie du réseau

La durée de vie du réseau est illustrée dans la figure 6.6,

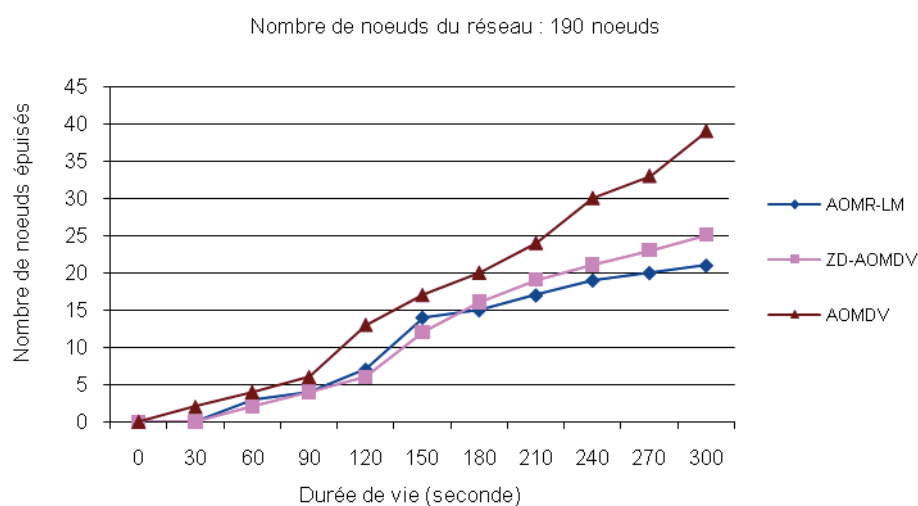


FIGURE 6.6 – Durée de vie du réseau vs nombre de nœuds épuisés.

pour un réseau de 190 nœuds. La durée de vie du réseau de AOMR-LM est plus longue que celles des AOMDV et ZD-AOMDV. Moins de nœuds sont épuisés dans AOMR-LM par rapport aux autres protocoles simulés. La durée de vie du réseau dans AOMR-LM est augmentée de 11% par rapport à ZD-AOMDV et de 34% par rapport à AOMDV, en moyenne. Ce gain en durée de vie, pour AOMR-LM, est le résultat de l'équilibrage de l'énergie parmi tous les nœuds des chemins, ceci prolonge la durée de vie de nœud individuel et par la suite toute la durée vie du réseau.

4.2 Ecart type et coefficient de variation de la durée de vie vs le nombre de nœuds épuisés

Dans nos simulations, chaque point représente une moyenne de plusieurs valeurs obtenues lors de plusieurs simulations (dans notre simulation : 20 runs). Nous avons calculé la moyenne et l'écart type de toutes ces valeurs. Le coefficient de variation ($cv = \text{rapport de l'écart type à la moyenne}$) est calculé pour analyser la dispersion des valeurs. Le résultat du calcul est présenté dans le tableau 6.3.

La durée de simulation (seconde)	Le nombre de nœuds épuisés		
	Moyenne	Ecart type	Coefficient de variation
0	0	0	#
30	0	0,653834842	#
60	2,85	1,152171862	0,40427083
90	4,05	1,395528574	0,34457496
120	6,55	1,745709025	0,26652046
150	14,4	2,289104628	0,1589656
180	14,7	2,451530134	0,16677076
210	16,5	2,889636655	0,17512949
240	19,45	2,635811071	0,13551728
270	20,3	2,812472222	0,13854543
300	21,45	2,539192785	0,11837729

TABLE 6.3 – moyenne, l'écart type et le coefficient de variation de la durée de vie vs le nombre de nœuds épuisés.

La figure 6.7 montre la bonne dispersion de nos valeurs de simulation autour la moyenne, vu les valeurs réduites du coefficient de variation. Ceci montre que la simulation produit des valeurs avec une précision acceptable.

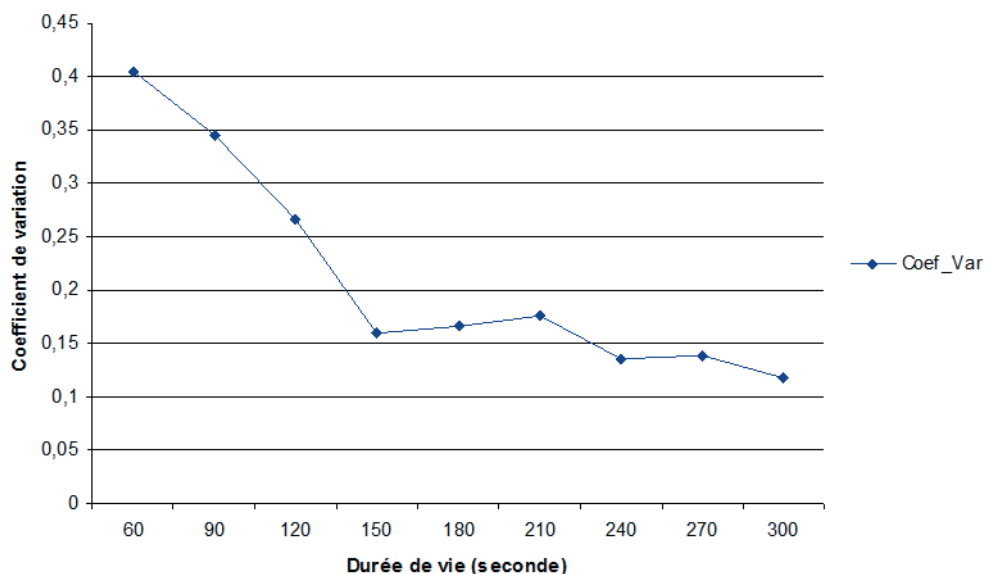


FIGURE 6.7 – Coefficient de variation de la durée de vie.

4.3 Consommation moyenne d'énergie

La figure 6.8 montre l'énergie consommée par les protocoles AOMR-LM, ZD-AOMDV, et AOMDV. AOMR-LM n'est pas performant au début de la simulation.

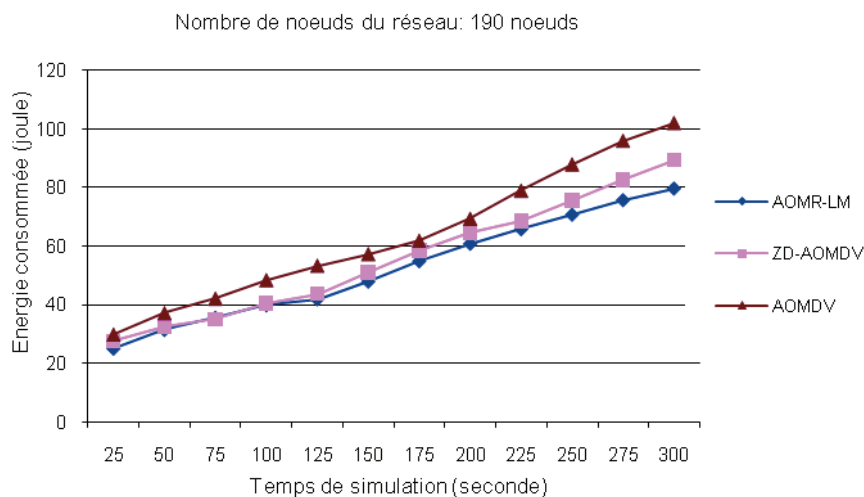


FIGURE 6.8 – L'énergie consommée vs temps de simulation.

Dans un premier temps, il n'est pas meilleur que les autres protocoles, parce que la majorité des paquets ne sont pas encore transmis, donc l'énergie totale de l'envoi et de la réception des paquets n'est pas importante. Une fois qu'on avance dans le

temps de la simulation, on constate un déséquilibre de la consommation de l'énergie. Nous pouvons voir que l'énergie consommée dans AOMR-LM est inférieure à celle consommée par ZD-AOMDV ou AOMDV. AOMR-LM consomme moins d'énergie que ZD-AOMDV ou AOMDV, en premier lieu parce AOMR-LM est en mesure d'équilibrer l'énergie entre les différents chemins. Ainsi, l'énergie est équilibrée à travers le réseau, ce qui réduit la consommation d'énergie inégale. En second lieu, AOMR-LM est en mesure d'éviter les nœuds qui sont faibles en énergie pour la construction des chemins multiples. Cela signifie que les chemins à énergie plus élevée sont identifiés et sélectionnés pour la transmission des paquets de données.

4.4 Délai moyen de bout en bout

La figure 6.9 représente le délai moyen de bout en bout des protocoles comparés. Le délai de bout-en-bout moyen pour tous les protocoles testés augmente avec l'augmentation de la taille du réseau, mais le délai moyen de bout-en-bout de AOMR-LM est inférieur à celui des protocoles ZD-AOMDV et AOMDV, pour différentes vitesses de mobilité des nœuds (voir les Figures 6.9 et 6.10).

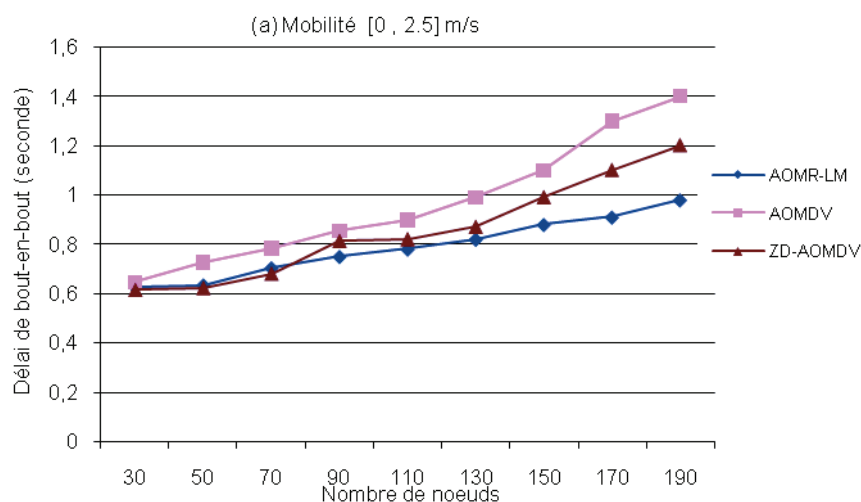


FIGURE 6.9 – délai de bout en bout vs nombre de nœuds. (a) Mobilité [0 , 2.5] m/s

Lorsque le nombre de nœuds du réseau est inférieur à 70, le protocole ZD-AOMDV a un délai égal ou meilleur que AOMR-LM due à l'utilisation simultanée des chemins. Une fois que la taille du réseau augmente, notre protocole produit une meilleure performance. La raison est que notre protocole AOMR-LM favorise les nœuds ayant un niveau d'énergie élevé lors de l'acheminement des paquets des données, et empêche les nœuds critiques en énergie de participer à la transmission des données. Ce qui produit moins de ruptures de liens et réduit considérablement le délai de bout en bout.

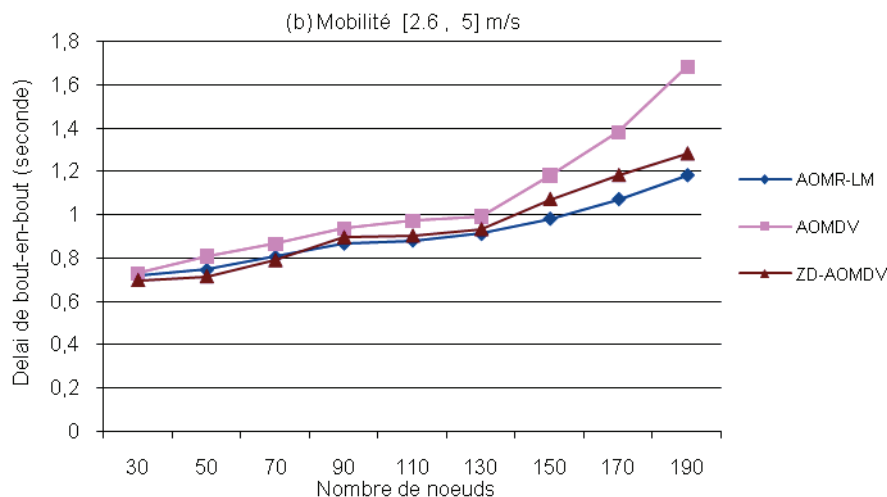


FIGURE 6.10 – délai de bout en bout vs nombre de nœuds. (b) Mobilité [2.6 , 5] m/s

5 Conclusion

Dans ce chapitre, nous avons présenté une solution aux problèmes de routage dans un réseau ad hoc. Les réseaux mobiles ad hoc sont caractérisés par leur absence d'infrastructures et de leur dynamique : ruptures des liens et coupures de chemins sont fréquents. En outre, les changements fréquents de topologie épuisent les batteries des nœuds, ce qui diminue les performances du réseau. Un nouveau protocole de routage multichemin, AOMR-LM, a été proposé dans ce chapitre, effectuant un routage performant en énergie dans les réseaux mobiles ad hoc. Il a été démontré que AOMR-LM préserve l'énergie résiduelle des nœuds et équilibre la consommation d'énergie entre les chemins. Le protocole proposé utilise un mécanisme qui exploite l'énergie résiduelle de nœuds pour sélectionner et classer les chemins en fonction des niveaux d'énergie de leurs nœuds. Ce concept prolonge la durée de vie du réseau et permet de préserver l'énergie du réseau en comparaison avec d'autres solutions connues dans la littérature. Le seuil d'énergie β et le coefficient α sont analysés afin de trouver les valeurs appropriées qui sont nécessaires pour définir la classe d'un nœud pendant le processus de découverte inverse des chemins. La comparaison des performances de AOMR-LM avec celles des protocoles AOMDV et ZD-AOMDV, montre que AOMR-LM permet d'équilibrer l'énergie consommée. AOMR-LM améliore la durée de vie du réseau de 11% et 34% comparativement à ZD-AOMDV et AOMDV respectivement. Il consomme moins d'énergie et a un délai de bout en bout moyen plus faible, la cause est que dans notre solution les chemins sont construits en fonction du niveau d'énergie de leurs nœuds, ainsi le meilleur chemin en terme d'énergie est sélectionné.

Chapitre 7

Conclusion et perspectives

1 Conclusion

L'émergence récente des petits périphériques communicants sans fil a rendu la connectivité disponible partout et à tout moment. De façon générale, les réseaux sans fil peuvent être basés sur une infrastructure ou être complètement indépendants. Alors que dans le premier cas, des routeurs, des serveurs de noms et adresses ainsi que des contrôleurs d'autorité gèrent l'infrastructure, dans le second, aucun support n'existe. Les nœuds qui vivent dans le réseau doivent s'auto-organiser et gérer eux mêmes les communications avec les autres nœuds. Ces réseaux sont appelés ad hoc ou MANET (Mobile Ad hoc NETWORKS).

A cause de la nature des nœuds dans un réseau ad hoc, les protocoles de communication doivent se conformer à certaines caractéristiques : consommation d'énergie, unité de calcul de faible puissance, bande passante limitée et changements de topologie fréquents dus à la mobilité des nœuds. Dans ce contexte, le routage est un véritable défi à relever. Étant donné que la plupart des solutions classiques, issues des protocoles de routage filaires, ne sont pas vraiment adaptées, un large éventail de nouveaux protocoles de routage spécialement dédiés à ce type de réseaux a vu le jour.

Parmi les contraintes citées ci-dessus, celle d'énergie pose le plus de problème de manière générale. Effectivement, en raison de la capacité limitée des batteries des nœuds, la consommation de l'énergie devrait être un critère fondamental lors de la conception de tels protocoles de routage. L'épuisement de l'énergie d'un nœud n'affecte pas uniquement sa capacité de recevoir ou émettre, mais également sa capacité d'acheminer les données pour les autres nœuds ce qui peut diminuer les performances du réseau ou carrément isoler certains segments du réseau. Le routage utilisé est l'un des principaux facteurs agissant sur le taux de consommation d'énergie dans ces réseaux mobiles ad hoc. Il est à noter que les protocoles de routage actuels, normalisés dans le groupe MANET de l'IETF, ne prennent pas en compte la métrique consommation de l'énergie pendant le processus de découverte de route. Ils s'intéressent, en revanche, à découvrir le plus court chemin. La métrique consommation de l'énergie, peut s'avérer plus efficace, d'autant que cette nouvelle métrique n'est pas forcément

un frein à la rapidité du processus de routage. Autrement dit, une métrique basée sur la consommation de l'énergie permettrait d'augmenter la durée de vie du réseau mobile ad hoc ; c'est probablement le facteur principal pour pouvoir communiquer.

Cette dernière décennie a marqué l'émergence d'une nouvelle famille de protocoles de routage pour les réseaux mobiles ad hoc dite "à basse consommation d'énergie". L'objectif le plus important de cette famille de protocoles est d'assurer une durée de fonctionnement du réseau maximale. Cela parce que les nœuds mobiles sont alimentés en énergie par des batteries dont la durée de vie est limitée. Le routage multichemin semble être une solution efficace pour palier les problèmes d'énergie dans les réseaux ad hoc.

C'est dans cet objectif que s'inscrivent les travaux de cette thèse. Nous avons commencé par une étude des principales solutions multichemins dans les réseaux ad hoc présentées dans la littérature et avons donné une taxonomie des différents protocoles de routage selon leur principe de découverte de chemins. L'état de l'art détaillé sur les différentes stratégies multichemins existantes nous a permis de dégager les besoins exprimés par ces dernières et aussi de faire une comparaison entre celles-ci. Cette étude nous a aidés à analyser les différents enjeux et défis posés par une stratégie multichemin durant chacune de ses phases (découverte des chemins, maintenance de chemins et la transmission des données).

La disjonction des chemins figure est une des propriété importante pour la performance du routage multichemin. Ces chemins peuvent être à nœud disjoints, à lien disjoints ou partiellement disjoints.

En se basant sur l'approche des chemins à nœuds disjoints, nous avons donc proposé AODVME+ (Ad hoc On-demand Disatance Vector Multipath Energie Plus), notre protocole est conçu principalement pour les nœuds à batteries limitées, où les ruptures des liens et les défaillances des chemins se produisent fréquemment. Notre protocole est une extension du protocole multichemin réactif AODVM, il permet de découvrir plusieurs chemins à nœuds disjoints entre un nœud source et un nœud de destination. AODVME+ exploite l'énergie résiduelle minimale des nœuds pour empêcher un ou plusieurs nœuds critiques d'épuiser leurs énergies et se déconnecter du réseau. Après le processus de découverte des chemins, une énergie minimale de chaque chemin est enregistrée, les valeurs des énergies minimales de ces chemins peuvent être très proches ou similaires. Un nouveau facteur qui peut être combiné avec cette valeur minimale est l'énergie moyenne des batteries des nœuds constituant les chemins. La sélection des chemins se base sur cette combinaison en favorisant l'énergie minimale.

Les tests d'évaluation ont été effectués avec le simulateur réseau NS-2. L'analyse des résultats de simulation a montré que notre protocole consomme moins d'énergie, a un délai de bout en bout moyen plus faible et minimise le surcout du routage. Nous concluons que notre protocole AODVME+ conserve l'énergie consommée et améliore les performances d'un réseau ad hoc.

Nous avons par la suite proposé une deuxième solution protocolaire AOMR-LM (Ad hoc On-demand Multipath Routing with Lifetime maximization) qui prolonge la durée de vie du réseau. Notre protocole est réactif multichemin à liens disjoints. Chaque nœud est classé suivant son niveau d'énergie, cette classification est reprise pour les chemins. Le niveau d'énergie d'un nœud est calculé en fonction de l'énergie résiduelle du nœud lors du processus de découverte des chemins. Notre protocole

sélectionne les nœuds de même classe pour construire un chemin donné. En fin de découverte, le nœud source aura un ensemble de chemins homogènes en termes d'énergie. La source sélectionne les chemins appartenant à la classe élevée pour la transmission des données. Notre solution ignore les nœuds avec un niveau d'énergie critique, pour qu'ils ne participent pas au routage. L'évaluation a montré que notre protocole AOMR-LM, améliore la durée de vie du réseau, consomme moins d'énergie et a un délai de bout en bout moyen plus faible en comparant avec les protocoles simulés, la raison est que notre solution construit les chemins en fonction du niveau d'énergie de leurs nœuds, ainsi le meilleur chemin en terme d'énergie est sélectionné.

2 Perspectives

Les perspectives pour les travaux présentés dans cette thèse sont nombreuses :

- Un premier axe de recherche intéressant serait d'utiliser des métriques prédictives dans les protocoles multichemins,
- Exploitation des solutions proposées afin de réaliser l'équilibrage de charge entre les chemins découverts,
- Extension des protocoles multichemins pour des réseaux où les nœuds sont multi-interfaces et multi-radios,
- Étude de l'interaction entre les couches de transport et MAC, et son impact sur la performance des protocoles de routage multichemin ;
- Application des protocoles multichemins dans un environnement scalable et dense éventuellement clusterisé (organisé en groupes),
- Etudier l'effet de différentes applications sur les performances de routage multichemin tels que : la vidéo streaming, les applications industrielles etc.

Bibliographie

- [1] S. Corson, J. Macker, *Mobile ad hoc networking (MANET)*, RFC 2501, Routing Protocol Performance Issues and Evaluation Considerations, IETF, June 1999.
- [2] N. Mansouri, *Protocole de routage multichemin avec quilibrage de charge dans les réseaux mobiles ad hoc*, Rapport de Projet de fin dtudes: Réseaux et Services mobiles. 2007.
- [3] P. Meskauskas, *Mobile ad hoc networking*, Seminar on Telecommunications Technology, Helsinki, October 12, 1998.
- [4] D Murthy, J J Garcia-Luna-Aceves, *A routing protocol for packet radio networks*, In Proc. of the 1st ACM International Conference on Mobile Computing and Networking (Mobicom95), pp. 8695, 1995.
- [5] T H Duyen, W Benjapolakul, D P Minh, *Performance evaluation and comparison of different ad hoc routing protocols*, Elsevier Computer Communications. Vol. 30, No. 11-12, pp. 2478-2496, Sep 2007.
- [6] G Jayakumar, G Gopinath, *Performance comparison of two on demand routing protocols for ad-hoc networks based on random way point mobility model*, American Journal of Applied Sciences, Vol. 5, No. 6, pp. 659-664, June 2008.
- [7] H Rivano, F Theoleyre, F Valois, *A Framework for the Capacity Evaluation of Multihops Wireless Networks*, Ad Hoc And Sensor Wireless Networks (AHSWN), Vol. 9, No. 3, pp. 139-162, January 2010.
- [8] A Radwan, T M Mahmoud, E H Houssein, *Performance measurement of some mobile ad hoc network routing protocols*, International Journal Computer Science Issues. Vol. 8, No. 1, pp. 107-112, January 2011.
- [9] J. M. Percher, B. Jouga, *Dtection dintrusions dans les réseaux ad hoc*, in Proc. of 1er Symposium sur la Scurit des Technologies de l'Information et de la Communication (SSTIC03), juin 2003.
- [10] M. M. Islam, R. D. Pose, C. Kopp *Routing protocols for ad-hoc networks*, in *Mobile Multimedia Communications: Concepts, Applications and Challenges*, Eds Gour Karmakar and Laurence S. Dooley, Information Science Reference, Hershey PA USA, pp. 178-221, 2008.
- [11] C. Perkins and P. Bhagwat, *Highly dynamic Destination-Sequenced Distance-Vector routing (DSDV) for Mobile Computers*, Association for Computing Machinery's Special Interest Group on Data Communication94, pp. 234244, 1994.
- [12] T. Clausen and P. Jacquet, *Optimized Link State Routing Protocol*, RFC 3626, October 2003.
- [13] A. Munaretto, H. Badis, K. A. Agha, and G. Pujolle, *A link-state qos routing protocol for ad hoc networks*, in Proc. of the 4th IFIP IEEE International Conference on Mobile and Wireless Communication Networks (MWCN02), Stockholm, Sweden, Sept. 2002.
- [14] R. Kumawat, V. Somani, *Comparative analysis of DSDV and OLSR routing protocols in MANET at different traffic load*, in Proc. published by International Journal of Computer Applications (IJCA), CSI- COMNET, 2011.

- [15] K. Chebira, *Etude et analyse de la stabilit des protocoles de routage dans les rseaux ad hoc*, Mmoire de magistre: Informatique industrielle. 2007.
- [16] M. Abolhasan , T. Wysocki , E. Dutkiewicz, *A review of routing protocols for mobile ad hoc networks*, Ad Hoc Networks, vol. 2, no. 1, pp. 1-22, 2004 .
- [17] C. E. Perkins, E. M. Royer, S. Das. *Ad hoc on-demand distance vector (AODV) routing*, RFC 3561, July 2003.
- [18] D. B. Johnson, D. A. Maltz, J. Broch, *DSR: The Dynamic Source Routing Protocol for Multihop Wireless Ad Hoc Networks*, In Ad Hoc Networking, C. E. Perkins (Ed.), Addison-Wesley, pages 139172, 2001.
- [19] C. E. Perkins, E. M. Royer, *Ad hoc On-Demand Distance Vector Routing*, in Proc. 2nd IEEE Wksp. Mobile Comp. Sys. and Apps., pp. 90100, Feb. 1999.
- [20] M. Marina, S. Das, *On-demand multipath distance vector routing in ad hoc networks*, Ninth Int. Conf. Network Protocols (ICNP), California, IEEE Computer Society Press, pp. 1423, November 2001.
- [21] V. Gauthier, *Cross layer et QoS dans les rseaux ad hoc sans fils*, Rapport bibliographique, Institut National des Tlcommunications, Evry, France, GET-INT-RR, 2004.
- [22] Z. Haas, M. Pearlman, *ZRP: A hybrid framework for routing in ad hoc networks*, Book Chapter in Ad Hoc Networks, Editor C. Perkins, Addison Wesley, pp. 221-254, 2001.
- [23] Z. J. Haas, M. R. Pearlman, P. Samar, *Intrazone Routing Protocol (IARP)*, IETF Internet Draft, <http://www.ietf.org/internet-draft-draft-ietfmanet-iarp-02.txt>, July 2002.
- [24] Z. J. Haas, M. R. Pearlman, P. Samar, *Interzone Routing Protocol (IERP)*, IETF Internet Draft, <http://www.ietf.org/internet-draft-draft-ietfmanet-ierp-02.txt>, July 2002.
- [25] Z. J. Haas, M. R. Pearlman, P. Samar, *Bordercasting Resolution Protocol (BRP)*, IETF Internet Draft, [draft-ietf-manet-brp-02.txt](http://www.ietf.org/internet-draft-draft-ietf-manet-brp-02.txt), July 2002.
- [26] P. Mohapatra, S. Krishnamurthy. *Ad hoc networks technologies and protocols*, Book Springer, 2005.
- [27] P. Kuppusamy, K. Thirunavukkarasu, B. Kalaavathi, *A Study and Comparison of OLSR, AODV and TORA Routing Protocols in Ad Hoc Networks*, 3rd International Conference on Electronics Computer Technology (ICECT 2011), 2011.
- [28] S. Chen, K. Nahrstedt, *Distributed quality-of service routing in ad hoc networks*, IEEE Journal on Selected Areas in Communications, vol. 17, no. 8, pp. 14881505, 1999.
- [29] S. Mueller, P. Tsang, D. Ghosal, *Multipath routing in mobile ad hoc networks: issues and challenges. Performance Tools and Application to Networked Systems*, Lecture Notes in Computer Science, M. C. Calzarossa, E. Gelenbe, Eds., vol. 2965, pp. 209234, Apr. 2004.
- [30] M. Tarique, K. E Tepe, S. Adibi, S. Erfani, *Survey of multipath routing protocols for mobile ad hoc networks*, Journal of Network and Computer Applications, Elsevier, vol. 32, no. 6, pp. 1125-1143, Nov. 2009.

- [31] P.Periyasamy, E.Karthikeyan, *Survey of Current Multipath Routing Protocols for Mobile AD Hoc Networks*, I. J. Computer Network and Information Security, vol. 12, pp. 68-79, 2013.
- [32] Z. Ye, S. V Krishnamurthy, S. K Tripathi, *A framework for reliable routing in mobile ad hoc networks*, in Proc. of the IEEE INFOCOM 2003, Vol. 1, pp. 270280, 2003
- [33] S. Yin, X. Lin, *MALB: MANET adaptive load balancing*, in Proc. of Vehicular Technology Conference, 2004 vol. 4, pp. 28432847, Sept. 2004
- [34] A. Boursier, S. Dahlen, *Cross-layer approach in mobile as hoc routing*, Project report, Dec. 2004.
- [35] V.D. Park, M.S.Corson, *A Highly Adaptive Distributed Routing Algorithm for Mobile Wireless Networks*, In Proceedings of 1997 IEEE conference on computer Communications (Infocom 97), Apr. 1997.
- [36] S. J. Lee, M. Gerla, *AODV-BR: backup routing in ad hoc networks*, IEEE Wireless Communications and Networking Conference (WCNC00), Sept. 2000.
- [37] M.R. Pearlman, Z.J. Haas, P. Sholander, S. S. Tabrizi, *On the impact of alternate path routing for load balancing in mobile ad hoc networks*, The ACM Symposium on Mobile Ad Hoc Networking and Computing (MobiHOC00), Boston, MA, Aug. 2000.
- [38] S. J. Lee, M. Gerla, *Split multipath routing with maximally disjoint paths in ad hoc networks*, International Conference on Communications (ICC01), Helsinki, Finland, June 2001.
- [39] A. Tsirigos, Z. J. Haas, *Multipath routing in the presence of frequent topological changes*, IEEE Communication Magazine, vol. 39, no. 11, pp. 132-138, Nov. 2001.
- [40] N. Gogate, S. S. Panwar, *Supporting video/image applications in a mobile multihop radio environment using route diversity*, IEEE International Conference on Communications (ICC99), Vancouver, Canada, June 1999.
- [41] S. Singh, M. Woo, C.S. Raghavendra, *Power aware routing in mobile ad hoc networks*, in Proc. of the Fourth Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom 98), 1998.
- [42] D. Ganesan, R. Govindan, S. Shenker, D. Estrin, *Highly-resilient, energyefficient multipath routing in wireless sensor networks*, Mobile Computing and Communication Review, 5(4):10-24, Oct. 2001.
- [43] K. Wu, J. Harms, *Performance study of a multipath routing method for wireless mobile ad hoc networks*, 9th international symposium on modeling, analysis and simulation of computer and telecommunication system (MASCOTS 01), Cincinnati, Ohio, Aug. 2001.
- [44] J. Broch, D. Maltz, D. Johnson, Y-C. Hu, J. Jetcheva, *A performance comparison of multi-hop wireless ad hoc network routing protocol*, The 4th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom98), pp. 85-97, Dallas, TX, Oct. 1998.

- [45] P. Johansson, T. Larsson, N. Hedman, B. Mielczarek, M. Degermark, *Scenario-based performance analysis of routing protocols for mobile ad hoc networks*, The 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom99), pp.195-206, Seattle, WA, August 1999.
- [46] S. K. Das, A. Mukherjee, S. Bandyopadhyay, D. Saha, and K. Paul, *An Adaptive Framework for QoS Routing through Multiple Paths in Ad Hoc Wireless Networks*, Journal of Parallel and Distributed Computing, vol. 63, no. 2, pp. 141153, February 2003.
- [47] H. Chan, A. Perrig, D. Song, *Random key predistribution schemes for sensor networks*, IEEE Symposium on Security and Privacy (SP03), Oakland, CA, May 2003.
- [48] S. Zhu, S. Xu, S. Setia, S. Jajodia, *Establishing pairwise keys for secure communication in ad hoc networks: A probabilistic approach*, 11th IEEE International Conference on Network Protocols (ICNP03), Atlanta, GA, Nov. 2003.
- [49] A. Gecyasar, T. Braun, *Ad-hoc multipath routing protocol*, Bachelor, Institute of Applied Informatics and Mathematics, 2006.
- [50] J. Yi, E. Cizeron, S. Hamma, B. Parrein, *Simulation and Performance Analysis of MP-OLSR for Mobile ad-Hoc Networks*, IEEE Wireless Communications and Networking Conference (WCNC 2008), Page(s) : 2235 - 2240, 2008.
- [51] X. Zhou, Y.Lu, B. Xi, *A novel routing protocol for ad hoc sensor networks using multiple disjoint paths*, 2nd IEEE international Conference on Broadband Networks, Boston, Oct. 2005.
- [52] L. Chen, W. B. Heinzelman, *A survey of routing protocols that support qos in mobile ad hoc networks*, IEEE Network, pages 30-3, 2007.
- [53] S. Chettibi, M. Benmohammed, *A Multipath Energy-Aware On demand Source Routing Protocol for Mobile Ad-Hoc Networks*, 1st Workshop on Next Generation Networks: Mobility, WNGN,, Maroc, 2009.
- [54] E. Shih, S. H. Cho, N. Ickes, R. Min, A. Sinha, A. Wang, A. Chandrakasan, *Physical Layer driven protocol and algorithm design for energy-efficient wireless sensors Networks*, in Proc. ACM Mobicom, Rome, Italy, 2001.
- [55] W. R. Heinzelman, A. Chandrakasan, H. Balakrishnan, *Energy-Efficient Communication Protocol for Wireless Microsensor Networks*, in Proc. of the 33rd Hawaii International Conference on System Sciences, 2000.
- [56] M. Ali, A. Bhm, M. Jonsson, *Wireless sensor networks for surveillance applications - A comparative survey of MAC protocols*, in Proc. of the 4th International Conference on Wireless and Mobile Communications (ICWMC '08), pages 399-403, Washington, DC, USA, 2008.
- [57] K. Holger, A. Willig, *Protocols and Architectures for Wireless Sensor Networks*, Wiley, 2005.
- [58] W. Ye, J. Heidemann, D. Estrin, *Medium access control with coordinated adaptive sleeping for wireless sensor networks*, IEEE/ACM Transactions on Networking, 2004.

- [59] J. H. Chang, L. Tassiulas, *Energy conserving routing in wireless ad-hoc networks*, in Proc. of the 19th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM'00), volume 1, pages 22-31, 2000.
- [60] C. Chiasserini, I. Chlamtac, P. Monti, A. Nucci, *Energy efficient design of wireless Ad Hoc networks*, in Proc. of the 2nd International IFIP-TC6 Networking Conference on Networking Technologies, Services, and Protocols, pp. 376-386, London, UK, 2002.
- [61] S. Soro, W. B. Heinzelman, *Prolonging the lifetime of wireless sensor networks via unequal clustering*, in Proc. of 19th IEEE International Parallel and Distributed Processing Symposium (IPDPS'05), Denver, Colorado, USA, 2005
- [62] D. Tian , N. D. Georganas, *A coverage-preserving node scheduling scheme for large wireless sensor networks*, in Proc. of the 1st ACM international workshop on Wireless Sensor Networks and Applications (WSNA'02), pages 32-41, New York, USA, 2002.
- [63] C. Yu, B. Lee, H. Youn, *Energy efficient routing protocols for mobile Ad Hoc networks*, Wireless Communications and Mobile Computing. vol. 3, no. 8, pp. 959-973, 2003.
- [64] H. Idoudi, W. Akkari, A. Belghith, M. Molnar *A alternance synchrone pour la conservation d'nergie dans les rseaux ad hoc*, Institut nationale de recherche en informatique et en automatique, Rapport de recherche N6045, Dc. 2006.
- [65] V. Rodoplu, T. H. Meng, *Minimum energy mobile wireless networks*, IEEE Journal on Selected Areas in Communications, vol. 17, No. 8, pp. 1333-1344, August 1999.
- [66] K. Woo, C. Yu, H. Y. Youn, B Lee, *Non-blocking, localized routing algorithm for balanced energy consumption in mobile ad hoc networks*, International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS 2001), pp. 1171-1174, 2001.
- [67] Q. Zhao, L. Tong, *Energy efficiency of large-scale wireless networks: proactive versus reactive networking*, IEEE Journal on Selected Areas in Communications, vol. 23, no. 5, pp. 1100-1113, May 2005.
- [68] L. Bao, J. J. Garcia-Luna-Aceves, *Stable energy-aware topology management in ad hoc networks*, in Proc. of Ad Hoc Networks, vol. 8, no. 3, pp. 313-327, 2010.
- [69] F. De Rango, F. Guerriero, P. Fazio, *Link-stability and energy aware routing protocol in distributed wireless networks*, IEEE Transactions on Parallel and Distributed Systems, vol. 23, no. 4, pp. 713-726, Apr. 2012.
- [70] C.K. Toh, *Maximum Battery Life Routing to Support Ubiquitous Mobile Computing in Wireless Ad Hoc Networks*, IEEE Communications Magazine, 2001.
- [71] M. Maleki, K. Dantu, M. Pedram *Lifetime Prediction Routing in Mobile Ad Hoc Networks*, IEEE Wireless Communication and Networking Conf, 2003.
- [72] V. Hau, V. Moraru, *Communications mobiles sans fil dans le contexte d'conomie d'nergie*, projet de recherche AROUND-SCOUT de l'equipe IFI-MSI, Hanoi, 2007
- [73] Y. Liu, L. Guo, H. Ma, T. Jiang, *Energy efficient on demand multipath routing protocol for multihop ad hoc networks*, in Proc. of IEEE 10th International Symposium on Spread Spectrum and Applications, pp. 592-597, Aug. 2008.

- [74] B. Yahya, J. Ben-Othman, *REER : Robust and Energy Efficient Multipath Routing Protocol for Wireless Sensor Networks*, in Proc. IEEE Communications Society subject matter experts for publication in the IEEE (GLOBECOM), 2009.
- [75] H. Nasehi, N. T. Javan, A. B. Aghababa, Y. G. Birgani, *Improving energy efficiency in manets by multi-path routing*, International Journal of Wireless And Mobile Networks (IJWMN), Vol. 5, No. 1, Feb. 2013.
- [76] O. Smail, Z. Mekkakia, B. Messabih, R. Mekki, B. Cousin, *Energy Conservation for Ad Hoc On-Demand Distance Vector Multipath Routing Protocol*, International Journal of Computer Network and Information Security (IJCNIS), Vol. 6, No. 6, pp. 1-8, 2014.
- [77] The Network Simulator ns-2, available online at <http://www.isi.edu/nsnam/ns>.
- [78] L. M. Feeney, and M. Nilsson, *Investigating the energy consumption of a wireless network interface in an ad hoc networking environment*, IEEE INFOCOM, pp. 1548-1557, 2001.
- [79] K. Dongkyun, J. J. Garcia-Luna-Aceves, K. Obraczka, J.-C. Cano, P. Manzoni, *Routing mechanisms for mobile ad Hoc networks based on the energy drain rate*, IEEE Trans. Mobile Computing, vol. 2, no. 2, pp. 161-173, 2003.
- [80] G. Parissidis, V. Lenders, M. May, B. Plattner, *Multipath routing protocols in wireless mobile ad hoc networks: a quantitative comparison*, in Proc. of 6th NEW2AN, pp. 313326, 2006.
- [81] S. Malini, E. Kannan, A. Valarmathi, C. D. S. Rajan, *Traffic based energy consumption analysis of AOMDV protocol in a mobile ad hoc network*, International Journal of Computer Applications, Vol. 14, No. 6, pp. 43-46, Feb. 2011.
- [82] P. Periyasamy, E. Karthikeyan, *Performance evaluation of AOMDV protocol based on various scenario and traffic patterns*, International Journal of Computer Science, Engineering and Applications (IJCSEA), Vol. 1, No. 6, pp. 33-48, 2012.
- [83] O. Smail, B. Cousin, R. Mekki, Z. Mekkakia, *A Multipath Energy-Conserving Routing Protocol for Wireless Ad Hoc Networks Lifetime Improvement*, EURASIP Journal on Wireless Communications and Networking (Section: Green Wireless Communications, Theory and Practice), Vol. 2014, No. 139, pp. 1-22, 2014.
- [84] M. Bheemalingaiah, M. M. Naidu, D. S. Rao, G. Varaprasad, *Energy aware node disjoint multipath routing in mobile ad hoc network*, Journal of Theoretical and Applied Information Technology, Vol. 2, No.2, pp. 416-431, 2009.
- [85] N. Meghanathan, *A node-disjoint multi-path routing protocol based on location prediction for mobile ad hoc networks*, International Journal of Engineering, Science and Technology, Vol. 2, No. 5, pp. 66-80, 2010.