

UNIVERSITÉ DE MONTRÉAL

**PROBLÈMES DE LA MODÉLISATION ET DU FLOODING
DANS LES RÉSEAUX MOBILES AD-HOC**

OTHMAN ISSAAD
DÉPARTEMENT DE GÉNIE INFORMATIQUE
ÉCOLE POLYTECHNIQUE DE MONTRÉAL

MÉMOIRE PRÉSENTÉ EN VUE DE L'OBTENTION
DU DIPLÔME DE MAÎTRISE ÈS SCIENCES APPLIQUÉES
(GÉNIE INFORMATIQUE)

JUIN 2004



Library and
Archives Canada

Bibliothèque et
Archives Canada

Published Heritage
Branch

Direction du
Patrimoine de l'édition

395 Wellington Street
Ottawa ON K1A 0N4
Canada

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file *Votre référence*

ISBN: 0-612-97954-7

Our file *Notre référence*

ISBN: 0-612-97954-7

NOTICE:

The author has granted a non-exclusive license allowing Library and Archives Canada to reproduce, publish, archive, preserve, conserve, communicate to the public by telecommunication or on the Internet, loan, distribute and sell theses worldwide, for commercial or non-commercial purposes, in microform, paper, electronic and/or any other formats.

The author retains copyright ownership and moral rights in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

AVIS:

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, publier, archiver, sauvegarder, conserver, transmettre au public par télécommunication ou par l'Internet, prêter, distribuer et vendre des thèses partout dans le monde, à des fins commerciales ou autres, sur support microforme, papier, électronique et/ou autres formats.

L'auteur conserve la propriété du droit d'auteur et des droits moraux qui protègent cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.


Canada

UNIVERSITÉ DE MONTRÉAL

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

Ce mémoire intitulé :

**PROBLÈMES DE LA MODÉLISATION ET DU FLOODING
DANS LES RÉSEAUX MOBILES AD-HOC**

présenté par : ISSAAD Othman

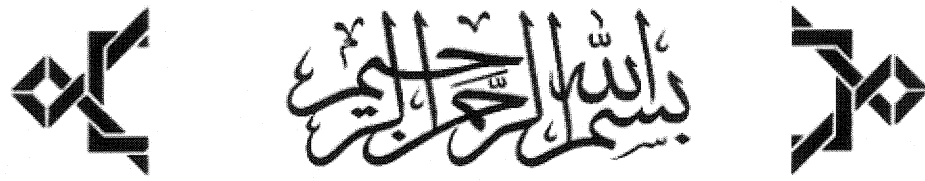
en vue de l'obtention du diplôme de : maîtrise ès sciences appliquées

a été dûment accepté par le jury d'examen composé de :

M. GALINIER Philippe, Doct., président

M. PIERRE Samuel, Ph. D., membre et directeur de recherche

M. QUINTERO Alejandro, Doct., membre



بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

REMERCIEMENTS

Au terme de ce travail, je tiens à présenter un remerciement bien distingué à mon professeur et encadrant M. Samuel Pierre pour son soutien, son aide, sa grande disponibilité, ses précieuses remarques et orientations tant sur le plan éducatif que sur celui de ma vie courante et ce, durant mes deux années d'études à l'École Polytechnique, notamment durant mon projet de maîtrise et la rédaction de ce mémoire.

Mes sincères remerciements aux professeurs MM. Philippe Galinier et Alejandro Quintero pour l'honneur qu'ils me font en acceptant de juger ce travail.

Je tiens à présenter ma grande reconnaissance et gratitude à mes deux anciens professeurs MM. Azzedine Boulmakoul et François Soumis pour tout ce que j'ai appris auprès d'eux.

Je ne pourrais oublier de présenter mes sincères remerciements, ma reconnaissance et gratitude à mes très chers parents : Zahra et El Yazid, à ma tante Rachida, et à mes sœurs Nadia et Safaa pour tout le soutien qu'ils m'ont apporté depuis mon jeune âge.

Mes profonds remerciements vont également à ma très chère Marylène pour son amour et tout le soutien qu'elle m'a fourni, ses grands efforts et sa formidable écoute et disponibilité.

Merci également à mon cher ami Mohamed Bakillah pour son soutien et ses encouragements.

Je remercie aussi tous les membres du LARIM, notamment le personnel technique.

RÉSUMÉ

Depuis les années soixante-dix, les réseaux sans fil ont suscité un engouement extraordinaire de la part des utilisateurs, des chercheurs et de l'industrie. Cet engouement s'explique en grande partie par le fait que ces réseaux répondent à deux besoins essentiels de la vie courante de chacun, à savoir la mobilité et la communication. Depuis, le développement de ces réseaux n'a cessé de s'accroître, donnant ainsi naissance à plusieurs générations de réseaux sans fil. Chacune de ces générations offre un meilleur traitement de la voix, la possibilité d'intégrer un nouveau flux de données, une autonomie plus grande, etc. Toutefois, elles étaient toutes limitées à la présence de points d'accès. En effet, la communication via les réseaux sans fil repose essentiellement sur des infrastructures fixes tels que les stations de base, les commutateurs, etc.

Pour pallier cet inconvénient, une nouvelle génération de réseaux sans fil a vu le jour, en l'occurrence les réseaux Ad-hoc. Ces réseaux se distinguent des autres formes de réseaux sans fil par une absence totale d'infrastructure fixe. Un nombre quelconque de terminaux mobiles équipés de cartes d'interface radio et des couches protocolaires adéquates suffisent pour former un réseau Ad-hoc. Les réseaux Ad-hoc présentent une solution technologique prometteuse qui répond parfaitement à des exigences ponctuelles et temporelles en télécommunications. En effet, ces réseaux se déploient automatiquement et ne nécessitent ni infrastructure, ni administration centralisée. Il n'est pas alors surprenant de constater l'intérêt au niveau de la recherche scientifique pour ces réseaux. Malheureusement, cet intérêt est en partie expliqué par les différents problèmes dont souffrent ces réseaux dans les limitations des batteries, l'accès au médium, la faible puissance radio.

Dans le cadre de notre recherche, nous nous intéressons dans un premier temps au problème de la modélisation des réseaux Ad-hoc. Nous étudions ainsi l'approche de modélisation classique des réseaux Ad-hoc et en dévoilons les principales limitations et

inconvenients. En l'occurrence, nous constatons que cette approche est inadéquate pour modéliser les réseaux Ad-hoc, surtout lorsqu'il s'agit de tenir compte des caractéristiques de la couche physique lors de la modélisation. En effet, tenir compte de ces caractéristiques s'avère indispensable quand il s'agit de développer des solutions à des problèmes de haut niveau pour lesquels la couche physique n'offre pas de garanties. C'est ainsi que nous avons conçu et proposé une nouvelle approche de modélisation pour ces réseaux basée sur la construction d'ensembles flous. Cette approche permet effectivement de rallier les aspects de la couche physique à la modélisation de ces réseaux.

Dans un second temps, nous nous sommes intéressés au problème du *flooding* dans ces réseaux. Le *flooding* consiste en fait à envoyer un message à partir d'une source à tous les autres nœuds du réseau. De par son principe, le *flooding* exige des terminaux du réseau un grand nombre de retransmissions presque simultanées pour accomplir sa fonctionnalité. Ceci est un inconvénient car, utilisé d'une manière peu pensée, et même en un nombre limité de fois, le *flooding* peut conduire à la saturation. D'où l'importance d'en offrir une bonne gestion. Pour ce faire, nous avons modélisé le problème en utilisant la nouvelle approche de modélisation et avons donné un algorithme dit $\gamma+$ pour le résoudre.

Par la suite, nous nous sommes proposé d'étudier les performances de cet algorithme et de le comparer par rapport à l'un des meilleurs algorithmes de *flooding* dans la littérature, à savoir l'algorithme PDP. Les tests de performance conduits dans l'espace libre réduisent l'incertitude reliée à la couche MAC à la présence physique de l'unité réceptrice dans la portée de transmission de l'émetteur. Il résulte de ces tests que l'algorithme $\gamma+$ assure de meilleures performances que le PDP. En effet, l'algorithme que nous avons proposé garantit une meilleure couverture, tout en réduisant la charge du réseau. Ceci est certainement dû au fait de considérer la nouvelle approche qui rallie les aspects de la couche physique à la résolution de problèmes des couches supérieures.

ABSTRACT

Since the seventies, the wireless networks aroused an extraordinary craze on behalf of the users, the researchers and the industry. This craze is largely explained by the fact that these networks address two essential needs of the common life of each : the need to mobility and communication. So, the development of these networks did not stop increasing giving birth to several generations of wireless networks. Each of these generations offered an improved treatment of the voice, the possibility of integrating a new data stream, a bigger autonomy, etc. However, they were quite limited to the presence of an access points. Indeed, the communication using wireless networks is based essentially on fixed infrastructures such as base stations, switches, etc.

To mitigate this inconvenience, a new generation of wireless known as Ad-hoc wireless networks was born. These networks differ from the other forms of wireless networks by a total absence of fixed infrastructure. An unspecified number of mobile terminals equipped with wireless cards and adequate protocol layers can form an Ad-hoc network. Ad-hoc networks present a promising technological solution which perfectly answers punctual and temporal requirements in telecommunications. Indeed, these networks are deployed automatically and do not require either infrastructure, or centralized administration. Then, it is not surprising to notice the interest of the scientific research to these networks. Regrettably, this interest is partially explained by the various problems from which these networks suffer like the batteries' limited autonomy, the medium access problems and the restricted radio power.

Within the framework of our research, we are interested at first in the problem of modeling Ad-hoc networks. Thus, we study the classical approach of modeling Ad-hoc networks and reveal its main limitations and inconveniences. We notice that this approach is inadequate to model Ad-hoc networks, especially when it concerns taking into account characteristics of the physical layer during the modeling. Indeed, to take into

account these characteristics turns out indispensable when developing solutions of high-level problems for which the physical layer offers no or not enough guarantees. So we conceived and proposed a new approach for modeling for these networks based on the construction of fuzzy sets. This approach allows to take into account the aspects of the physical layer in the modelling of these networks.

Secondly, we were interested in the flooding problem in these networks. The flooding consists in fact in sending a message from a source to all other nodes of the network. Due to its principle, the flooding requires a big number of almost simultaneous broadcasts to carry out its feature. This is an inconvenience because, used in a little thought way, even a limited number of times, the flooding can lead to the saturation. Then it is indispensable to offer a good management for the flooding. So, we modeled the problem by using the new approach of modeling and gave an algorithm called $\gamma+$ to resolve it.

Afterward, we studied the performances of this algorithm and compared it with one of the best algorithms of flooding in the literature, known as PDP algorithm. Benchmarks driven in the free space, reduce the uncertainty of the MAC layer to the physical presence of a receiver in the transmission range of the transmitter. The tests conclude that the $\gamma+$ algorithm insures better performances than the PDP. Indeed, the algorithm which we proposed guarantees a better coverage, while reducing the load of the network. This is certainly due to the fact of considering the new approach which takes into account the aspects of the physical layer in the resolution of problems of superior layers.

TABLE DES MATIÈRES

DÉDICACE	iv
REMERCIEMENTS	v
RÉSUMÉ	vi
ABSTRACT	viii
TABLES DES MATIÈRES	x
LISTE DES FIGURES	xii
LISTE DES TABLEAUX	xv
CHAPITRE 1 : INTRODUCTION	1
1.1 Définitions et concepts de base.....	1
1.2 Éléments de la problématique.....	4
1.3 Objectifs de recherche.....	6
1.4 Plan du mémoire.....	6
CHAPITRE 2 : CARACTÉRISATION DES RÉSEAUX AD-HOC MOBILES	8
2.1 Rappel et définitions de base.....	8
2.1.1 Notions de la théorie des ensembles flous.....	8
2.1.2 Propagation radio électrique.....	8
2.2 Nature et caractéristiques des réseaux Ad-hoc.....	10
2.3 Routage dans les réseaux Ad-hoc.....	13
2.3.1 Protocoles de routage proactifs.....	17
2.3.2 Protocoles de routage réactifs.....	20
2.3.3 Analyse des deux approches.....	26
2.4 Qualité de service dans les réseaux Ad-hoc.....	27
2.4.1 Qualité de service: définitions et métriques.....	27
2.4.2 Modèles de qualité de service.....	28
2.4.3 Couche MAC.....	29
2.4.4 Routage avec qualité de service.....	30

2.4.5	Protocoles de signalisation.....	33
CHAPITRE 3 : MODÉLISATION DES RÉSEAUX AD-HOC: APPLICATION AU <i>FLOODING</i>.....		
3.1	Modélisation des réseaux Ad-hoc.....	35
3.1.1	Approche classique et limitations.....	35
3.1.2	Nouvelle approche proposée.....	38
3.2	Problème du <i>flooding</i>	39
3.3	Modélisation du problème.....	42
3.4	Analyse du problème.....	46
3.5	Stratégie et algorithme de résolution.....	51
3.6	Mise en œuvre de l'algorithme.....	53
CHAPITRE 4 : IMPLÉMENTATION ET RÉSULTATS.....		
4.1	Détails de l'implémentation.....	58
4.1.1	Implémentation et structures de données.....	58
4.1.2	Critère γ	69
4.1.3	Critère d'arrêt.....	73
4.1.4	Environnement matériel et logiciel.....	74
4.2	Plan d'expériences.....	76
4.3	Analyse des résultats d'expériences.....	77
4.4	Comparaison avec l'algorithme PDP.....	83
CHAPITRE 5 : CONCLUSION.....		
5.1	Synthèse des travaux.....	92
5.2	Limitations des travaux.....	94
5.3	Orientations et travaux futurs.....	95
BIBLIOGRAPHIE.....		
		96

LISTE DES FIGURES

Figure 1.1	Exemple de nombre flou.....	2
Figure 2.1	Exemple de nombre flou.....	9
Figure 2.2	Dégradation de la puissance tenant compte des phénomènes : l'atténuation, le masquage log-normal et les évanouissements de Rayleigh.....	11
Figure 2.3	Problème de la station exposée.....	15
Figure 2.4	Problème de la station cachée.....	15
Figure 2.5	Problème de sommabilité des bruits.....	16
Figure 2.6	Acheminement d'un message suivant le protocole CGSR.....	20
Figure 2.7	Propagation du RREQ dans le protocole AODV.....	21
Figure 2.8	Acheminement du RREP dans le protocole AODV.....	22
Figure 2.9	Formation de la route lors d'une découverte d'itinéraire DSR.....	24
Figure 2.10	Propagation de la réponse à une demande d'itinéraire DSR.....	25
Figure 3.1	Approximation de la dégradation de la qualité du signal.....	36
Figure 3.2	Calcul de la portion de l'espace couverte par une transmission additionnelle.....	40
Figure 3.3	Définition de la puissance floue \tilde{P}	45
Figure 3.4	Exemple de réseau Ad-hoc.....	47
Figure 3.5	Exemple de <i>flooding</i> sans tenir compte de l'information de voisinage.	48
Figure 3.6	Exemple de <i>flooding</i> en tenant compte du voisinage immédiat.....	48
Figure 3.7	Exemple de <i>flooding</i> en tenant compte du voisinage à deux sauts.....	49
Figure 3.8	Sélection des voisins à deux sauts à couvrir.....	50
Figure 3.9	Ensemble des chemins possibles pour acheminer un paquet de <i>flooding</i> dans une stratégie à deux sauts.....	52
Figure 3.10	Définition du seuil de tolérance S_γ	52

Figure 3.11	Algorithme de <i>flooding</i> proposé.....	54
Figure 3.12	Exemple de réseau Ad-hoc.....	55
Figure 4.1	Interface de la classe <i>SimulationScenario</i> représentant une série de simulations.....	59
Figure 4.2	Interface de la classe <i>Adhoc</i> représentant un réseau mobile Ad-hoc....	60
Figure 4.3	Définitions des paramètres de la simulation pour un réseau Ad-hoc...	61
Figure 4.4	Méthode et attributs privés de la classe <i>Mobile</i>	63
Figure 4.5	Méthodes et attributs publics de la classe <i>Mobile</i>	65
Figure 4.6	Définitions des paramètres de fonctionnement d'un mobile Ad-hoc ...	66
Figure 4.7	Interface de la classe <i>PDP-Algorithm</i>	67
Figure 4.8	Interface de la classe <i>PDP-Algorithm</i>	68
Figure 4.9	Interface de la classe <i>Queue</i>	69
Figure 4.10	Définition du critère γ	71
Figure 4.11	Exemple de <i>flooding</i> infini.....	73
Figure 4.12	Les différents indicateurs de performance de $\gamma+$ selon la valeur du facteur γ de tolérance aux pertes.....	79
Figure 4.13	Les différents indicateurs de performance de $\gamma+$ selon le nombre de mobiles dans le réseau	80
Figure 4.14	Les différents indicateurs de performance de $\gamma+$ selon la vitesse des mobiles en Km/h	81
Figure 4.15	Les différents indicateurs de performances de $\gamma+$ selon la puissance des mobiles exprimée par la couverture en mètre.....	82
Figure 4.16	<i>%Couverture</i> pour $\gamma+$	84
Figure 4.17	<i>%Couverture</i> pour PDP	84
Figure 4.18	Moyenne cumulative du <i>%Couverture</i> pour PDP et $\gamma+$	85
Figure 4.19	Pourcentage de fréquence du <i>%Couverture</i> pour PDP et $\gamma+$	86
Figure 4.20	Moyenne cumulative de la charge du réseau pour PDP et $\gamma+$	87
Figure 4.21	Moyenne cumulative de la densité d'échec des transmissions pour PDP et $\gamma+$	88

Figure 4.22	Les différents indicateurs de performance selon la vitesse des mobiles en Km/h pour PDP et $\gamma+$	89
Figure 4.23	Les différents indicateurs de performance selon le nombre de mobiles dans le réseau des transmissions pour PDP et $\gamma+$	90
Figure 4.24	Les différents indicateurs de performance selon la puissance d'émission des mobiles exprimée par la couverture en m pour PDP et $\gamma+$	91

LISTE DES TABLEAUX

Tableau 2.1	Coefficient α selon différents environnements.....	12
-------------	---	----

CHAPITRE 1

INTRODUCTION

Dans les réseaux de communications, la mobilité des usagers a introduit diverses nouvelles contraintes corrélées présentant parfois des aspects aléatoires difficiles à prendre en compte. De ce fait, plusieurs solutions ont vu le jour, dont la téléphonie cellulaire et les réseaux mobiles Ad-hoc. Ces derniers, aussi appelés MANET (*Mobile Ad-hoc Networks*), constituent l'objet de ce mémoire particulièrement axé sur la modélisation et le *flooding* dans ces réseaux. Dans le présent chapitre, nous allons d'abord définir certains concepts pertinents à notre travail. Puis, nous introduisons les éléments de base liés au problème de la mobilité et de la modélisation dans les réseaux Ad-hoc. Ensuite, nous énonçons les principaux objectifs de notre recherche ainsi que le plan directeur de ce mémoire.

1.1 Définitions et concepts de base

Les communications sans fil traditionnelles reposent essentiellement sur des infrastructures à architecture hiérarchique fixes tel qu'illustré à la Figure 1.1 (Pierre, 2003). L'ensemble de ces infrastructures se complète et interagit afin d'offrir les principales fonctionnalités d'un système de communication sans fil, à savoir : la localisation des usagers, la gestion des profils, le cheminement des messages et le relais des appels et des usagers. Pour ce faire, plusieurs points d'accès au réseau, ou stations de base (BS), sont connectés à d'autres composantes de gestion des usagers (MSC, VLR, HLR) ; ils sont répartis et fixés dans des endroits stratégiques afin de laisser libre cours à

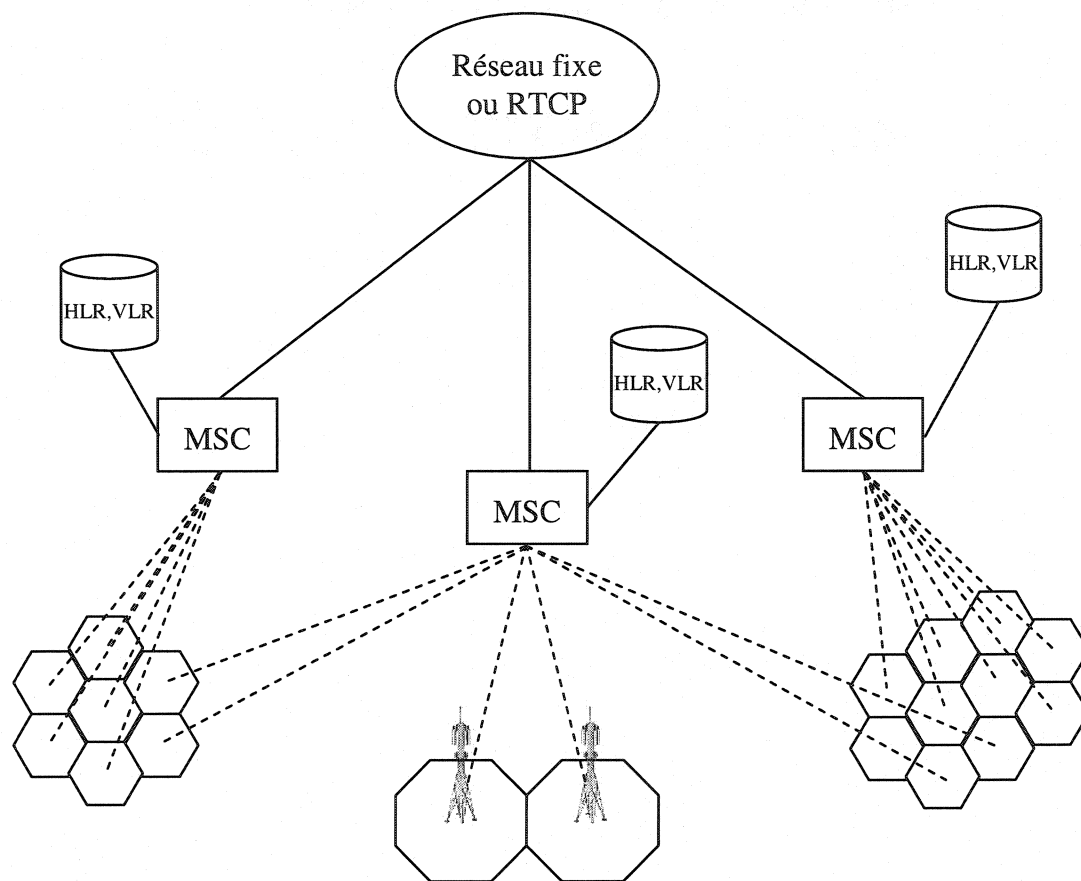


Figure 1.1 Architecture de base des réseaux sans fil 2G

la mobilité des usagers. Toutefois, il se trouve bien des cas où il n'existe presque pas d'infrastructure dédiée à la communication, ou que cette infrastructure, existante, est mal adaptée ou peu fonctionnelle. Malheureusement, dans de telles situations, il arrive souvent que l'échange d'information, ainsi que la fiabilité, la sécurité et la qualité de cet échange soient vitaux et d'une extrême urgence. Ces situations peuvent avoir lieu en cas de guerre, d'exploration spatiale ou de désastre majeur. Et dans ce genre de condition, les usagers peuvent se fier aux réseaux Ad-hoc.

Les réseaux Ad-hoc se distinguent des autres formes de réseaux sans fil par une absence totale d'infrastructure fixe. Un nombre quelconque de terminaux mobiles équipés de cartes d'interface radio et des couches protocolaires adéquates suffisent pour former un

réseau Ad-hoc. Aucun administrateur n'est donc requis pour la gestion, la configuration et la maintenance de tels réseaux. En effet, les nœuds mobiles jouent à la fois le rôle de terminaux et de routeurs et parfois le rôle de commutateurs ou d'administrateurs locaux, formant ainsi une architecture de communication totalement répartie. Généralement, un réseau Ad-hoc constitue un réseau isolé mais peut parfois (à l'aide des cartes d'interface adéquates) être intégré à un réseau fixe ou même faire le relais entre deux ou plusieurs réseaux fixes (Pierre, 2003).

Les réseaux Ad-hoc sont avant tout des réseaux formés par des nœuds mobiles utilisant l'interface radio pour acheminer des messages. Ils exploitent ainsi la portée radio de leurs cartes d'interface pour créer, de bout en bout, des liens avec les autres nœuds du réseau. En fait, les liens radio constituent les canaux de communication possibles et opèrent sur la bande ISM (*Industrial Scientific Medical Band*) des fréquences comprises entre 2,4 et 2,5 GHz (www.itu.int). La portée radio des terminaux est généralement variée et soumise à un certain nombre de contraintes spécifiques. En effet, les ondes hertziennes sont très sensibles à leur environnement. Cette sensibilité peut être décrite par certaines lois de la propagation radio que nous verrons au cours du chapitre suivant. Ces lois décrivent spécialement la valeur de la puissance du signal reçu par les terminaux Ad-hoc, autrement dit, la qualité des liens radio existants entre les terminaux. Malheureusement, cette qualité subit aussi l'impact de phénomènes de nature aléatoire. En effet, rappelons que les terminaux dans un réseau Ad-hoc sont mobiles. En conséquence, il est très difficile de prévoir si un lien existant à une certaine date sera toujours valide à une date ultérieure. Il est encore plus incertain qu'il conserve les mêmes propriétés. La topologie peut donc évoluer de façon imprévisible. D'où l'importance d'effectuer des opérations de découverte et/ou de mise à jour de route et de voisinage par les protocoles utilisés, en l'occurrence ceux dits d'inondation (*flooding*).

Le *flooding* consiste essentiellement à envoyer un message d'un nœud source à tous les autres nœuds du réseau. Ceci se fait de proche en proche, chaque nœud retransmettant le même message vers ses propres voisins et ainsi de suite jusqu'à ce que tous les terminaux le reçoivent. Ce type de mécanisme est très fréquemment utilisé dans

les réseaux Ad-hoc à cause des changements aléatoires de la topologie. En effet, il est particulièrement utile, lors de l'envoi d'un message de découverte de route, de recherche de correspondant (*paging*), ou de contrôle. Cependant, il reste un mécanisme très coûteux en ressources et doit être utilisé avec précaution.

1.2 Éléments de la problématique

Comme nous l'avons vu, les réseaux Ad-hoc suscitent un grand intérêt grâce à leur simplicité de déploiement et d'exploitation. De plus, ils offrent une mobilité totale aux usagers. Malheureusement, l'ensemble de ces atouts et surtout la mobilité totale posent de sérieux problèmes techniques particulièrement liés à l'établissement des liens radio. En effet, dû à la mobilité, la topologie de tels réseaux est en perpétuel changement sans préavis. En conséquence, les liens radio deviennent encore plus volatiles, ce qui compromet la garantie de consistance et de présence dans le temps des réseaux Ad-hoc. Ceci limite énormément les performances de ces réseaux à cause du grand impact de la mobilité sur les protocoles de routage, de *paging*, de *flooding*, etc. En effet, en plus de leurs opérations habituelles, ces protocoles doivent effectuer, des opérations de découvertes et/ou de mise à jour de routes et/ou de voisinage. Ainsi, il paraît indispensable d'implémenter des mécanismes de gestion de la mobilité afin de réduire la fréquence de telles opérations et améliorer, à terme, les performances de ces réseaux.

Notons que la notion de *voisinage* est particulièrement intéressante dans le cas des réseaux Ad-hoc. Elle est à la base de toute représentation graphique et modélisation mathématique qui fournissent elles-mêmes une base et un outil puissant pour la conception et le test des protocoles reliés à ces réseaux. Généralement, les réseaux Ad-hoc sont modélisés par un ensemble de nœuds représentant les unités mobiles. Chacun de ces nœuds définit un sous-ensemble constitué de ses nœuds voisins, autrement dit, constitué de tous les nœuds qui reçoivent son signal radio. Ainsi, un réseau Ad-hoc peut être modélisé par un graphe dont l'ensemble des nœuds représente les unités mobiles et l'ensemble des arcs du graphe les liens radio existants. Cette modélisation nous semble peu adaptée pour tenir compte de l'aspect volatile ou de l'importance des liens radio ainsi

que de l'importance des nœuds eux-mêmes. En effet, dans un tel réseau l'existence des liens radio ne peut être établie d'une manière figée et la disponibilité des nœuds à exécuter certaines tâches reliées aux protocoles utilisés n'est pas statique non plus. Dans ce contexte, une solution probabiliste pourrait être envisagée, en l'occurrence pour attribuer des poids aux arcs. Toutefois, cette approche, étant de nature probabiliste, est relativement peu adaptée pour représenter des phénomènes qui ne sont pas purement aléatoires mais bel et bien déterministes, certes peu bien cernés à cause de leur complexité (Dubois et Prade, 1978). Ceci laisse grandes ouvertes les portes à des recherches scientifiques intensives et nous incite à concevoir et tester de nouvelles approches de modélisation plus adaptées pour représenter les réseaux Ad-hoc.

Dans le même contexte, nous notons la grande importance de la notion de voisinage dans la gestion de la mobilité dans les réseaux Ad-hoc, c'est-à-dire dans le sens de réduire l'impact de cette mobilité sur les protocoles utilisés. Notamment, nous soulignons que la disponibilité de certaines informations spécifiques sur les nœuds et leurs voisinages nous permettrait d'évaluer la mobilité de chaque nœud. Ceci a pour effet de réduire la fréquence et la nécessité des opérations de découverte et de mise à jour de voisinage.

La plupart du temps, la mobilité a été considérée d'un point de vue spatial. Autrement dit, les déplacements des nœuds d'un réseau Ad-hoc sont entrepris uniquement comme des déplacements dans l'espace des trois dimensions spatiales, ce qui est tout à fait naturel. Cependant, particulièrement dans le cas de ces réseaux, le déplacement dans l'espace ne pose de problème que parce que la portée des signaux émis est limitée. D'où l'importance de considérer et de tester de nouvelles approches pour résoudre le problème de la mobilité dans les réseaux Ad-hoc.

Malheureusement, ce problème est d'une ampleur telle qu'il ne peut être abordé d'une manière globale. En effet, outre l'aspect aléatoire qu'il introduit, il affecte sérieusement tous les protocoles utilisés, réduisant ainsi leurs performances. Parmi ces protocoles, ceux dit d'inondation sont particulièrement critiques. En fait, de par son principe, le *flooding* exige des terminaux du réseau un grand nombre de retransmissions

presque simultanées pour accomplir sa fonctionnalité. Ainsi, utilisé d'une manière peu efficace, et même en un nombre limité de fois, le *flooding* peut vite conduire à la saturation. Ajouté à cela le nombre de retransmissions nécessaires à cause des problèmes de partage du médium, le *flooding* est particulièrement vulnérable. D'où l'importance de considérer et d'évaluer des solutions pour réduire l'impact de la mobilité sur les performances de ce protocole.

1.3 Objectifs la recherche

L'objectif principal de notre recherche est de proposer une nouvelle approche de modélisation pour les réseaux Ad-hoc, permettant de tenir compte de l'aspect volatile des liens et des nœuds dans un tel réseau. Par la suite, nous mettrons en application et sous test cette approche en vue de résoudre le problème du *flooding* couplé à la mobilité. L'ensemble de ces objectifs, analyses et tests qui constituent l'essence de ce travail sera atteint en suivant un processus visant spécifiquement à :

- analyser l'approche classique de modélisation des réseaux Ad-hoc dans le but d'identifier les principales faiblesses et limitations ;
- concevoir une nouvelle approche de modélisation des réseaux Ad-hoc pour remédier aux faiblesses de l'approche classique ;
- proposer un algorithme pour résoudre le problème du *flooding* couplé à la mobilité ;
- procéder à une série de tests et de simulations pour évaluer les performances de notre algorithme par rapport à l'un des meilleurs algorithmes déjà connus dans la littérature.

1.4 Plan du mémoire

Le présent mémoire est divisé en cinq chapitres. Dans le deuxième chapitre, nous faisons une synthèse des réseaux Ad-hoc ainsi que des principaux problèmes qui leur sont associés. Au cours du troisième chapitre, nous proposons une nouvelle approche pour traiter les réseaux Ad-hoc. Cette approche est ensuite mise en application dans le contexte

bien particulier de la gestion de la mobilité et de la minimisation des retransmissions dans le cas d'une opération de *flooding*; nous y présentons un nouvel algorithme pour résoudre le problème en question. Le quatrième chapitre décrit l'implémentation, les simulations, les tests et l'évaluation des performances pour notre algorithme. Finalement, le cinquième chapitre, en guise de conclusion, présente une synthèse des résultats obtenus et des indications sur les travaux futurs.

CHAPITRE 2

CARACTÉRISATION DES RÉSEAUX AD-HOC MOBILES

Depuis leur émergence dans les années soixante-dix, les réseaux sans fil sont devenus très populaires surtout durant la dernière décennie. À cette époque, le grand défi technologique était de permettre une mobilité à grande échelle pour les usagers, quoique limitée à la présence de point d'accès au réseau. Très vite, cette mobilité s'est accrue pour donner naissance aux réseaux Ad-hoc. Dans le présent chapitre, nous nous proposons de faire une synthèse des principaux travaux de recherche concernant les réseaux Ad-hoc. Tout au long de ce chapitre, nous parlerons de station hôte, de nœud ou de nœud mobile pour signifier un terminal Ad-hoc. Nous commençons par rappeler quelques notions mathématiques et donner un aperçu sur les lois qui régissent la propagation des ondes radio. Par la suite, nous donnons une définition des réseaux Ad-hoc et présentons leurs propriétés avant de passer en revue les principaux protocoles de routage et les travaux liés à la qualité de service.

2.1 Rappels et définitions de base

Dans cette section, nous fournissons quelques notions mathématiques principalement reliées à la théorie des ensembles flous. Ensuite, nous nous intéressons au phénomène et aux lois de la propagation des ondes radio.

2.1.1 Notions de la théorie des ensembles flous

La théorie des ensembles flous date des années soixante et a été introduite pour représenter des données et des phénomènes imprécis ou mal connus, pour remplacer les

modèles probabilistes qui conduisent souvent à d'énormes calculs. Dans cette section, nous en présentons les notions pertinentes à notre recherche.

Sous-ensemble flou: Soit E un ensemble classique utilisé comme référentiel. On appelle *sous-ensemble flou* A de E l'ensemble des couples $\tilde{A} = \{(x, \mu_A(x))\}, \forall x \in E$ où $\mu_A : E \longrightarrow M$ exprime le degré d'appartenance de x à A prenant des valeurs dans un ensemble M . Typiquement, on choisit : $M = [0,1]$, tel que $\mu_A : E \longrightarrow [0,1]$. Dans ce cas, μ_A est aussi dite fonction caractéristique.

Nombre flou: On appellera un *nombre flou* de valeur moyenne n , notée \tilde{n} , un sous-ensemble flou de \mathfrak{R} dont la fonction caractéristique $\mu_{\tilde{n}}$ est croissante au sens large sur $]-\infty, n]$, vaut 1 en n , est décroissante au sens large sur $[n, +\infty[$. La Figure 2.1 en est une illustration.

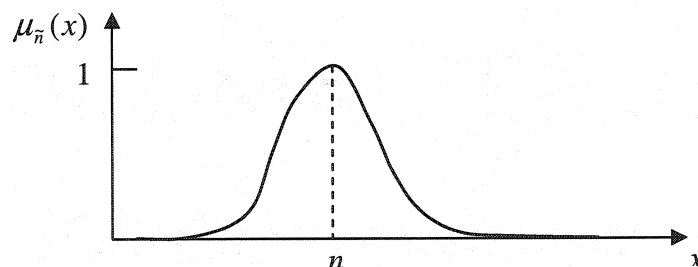


Figure 2.1 Exemple de nombre flou

Fonction de variables floues: À partir de toute fonction f de \mathfrak{R}^n dans \mathfrak{R} qui, à $(x_1, x_2, \dots, x_n) \in \mathfrak{R}^n$ associe $y = f(x_1, x_2, \dots, x_n)$, on peut induire une fonction \tilde{f} de $\wp(\mathfrak{R})^n$ dans $\wp(\mathfrak{R})$ où $\wp(\mathfrak{R})$ est l'ensemble des parties floues sur \mathfrak{R} , dont la restriction à l'ensemble des nombres réels classiques coïncide avec f , de la façon suivante : Au n -uplet de sous-ensembles flous $(\tilde{x}_1, \tilde{x}_2, \dots, \tilde{x}_n)$, on associe le sous-ensemble flou $\tilde{y} = \tilde{f}(\tilde{x}_1, \tilde{x}_2, \dots, \tilde{x}_n)$ de fonction caractéristique $\mu_{\tilde{y}}$ définie par :

$$\forall t \in \mathfrak{R}, \mu_{\tilde{y}}(t) = \max_{t=f(s_1, s_2, \dots, s_n)} \min(\mu_{\tilde{x}_1}(s_1), \mu_{\tilde{x}_2}(s_2), \dots, \mu_{\tilde{x}_n}(s_n))$$

Grphe flou: Soit $G(N, A)$ un graphe avec N l'ensemble des nœuds et A l'ensemble des arcs (ou arêtes). À partir de G on peut construire un graphe flou $\tilde{G}(\tilde{N}, \tilde{A})$ où \tilde{N} et \tilde{A} sont des ensembles flous sur N et A respectivement, tel que :

$$(\forall (x, y) \in \tilde{N} \times \tilde{N}) \quad \mu_{\tilde{A}}(x, y) \leq \min(\mu_{\tilde{N}}(x), \mu_{\tilde{N}}(y))$$

Notons que l'on peut aussi associer des poids flous aux arcs (arêtes) et aux nœuds.

2.1.2 Propagation radio électrique

La théorie caractérisation du phénomène de la propagation des signaux radio sert de fondements à la conception des réseaux radio mobiles. En effet, elle permet de quantifier les pertes en puissance du signal propagé et les propriétés des signaux reçus. En fait, le signal radio subit au cours de son trajet divers phénomènes de réflexion, de réfraction et de diffusion dus principalement aux caractéristiques de l'environnement et aux déplacements des unités réceptrices. Ces phénomènes induisent trois effets majeurs affectant la propagation, à savoir : l'atténuation globale du signal en fonction de la distance, le masquage du signal par des obstacles majeurs sur son trajet et les évanouissements du signal dus aux interférences (Conan, 2003). Ces effets sont illustrés à la Figure 2.2 selon un niveau de détail de plus en plus fin (Rappaport, 2002). Les deux premiers effets à savoir l'atténuation et le masquage log-normal, sont des effets à grande échelle qui décrivent la dégradation de signal lors de mouvements sur de grandes distances. Les évanouissements de Rayleigh, par contre, régissent les atténuations dues aux déplacements à petite échelle sur des distances de l'ordre de 1 m.

En fait, lorsqu'une unité Ad-hoc est en mouvement, le signal radio $r(t)$ qu'elle reçoit peut être décomposé en deux composantes de nature aléatoire (Skalar, 1997):

$$r(t) = m(t) \cdot r_0(t)$$

où $m(t)$ représente les fluctuations du signal dues à l'effet de grande échelle et $r_0(t)$ les évanouissements du signal, c'est-à-dire l'effet à petite échelle.

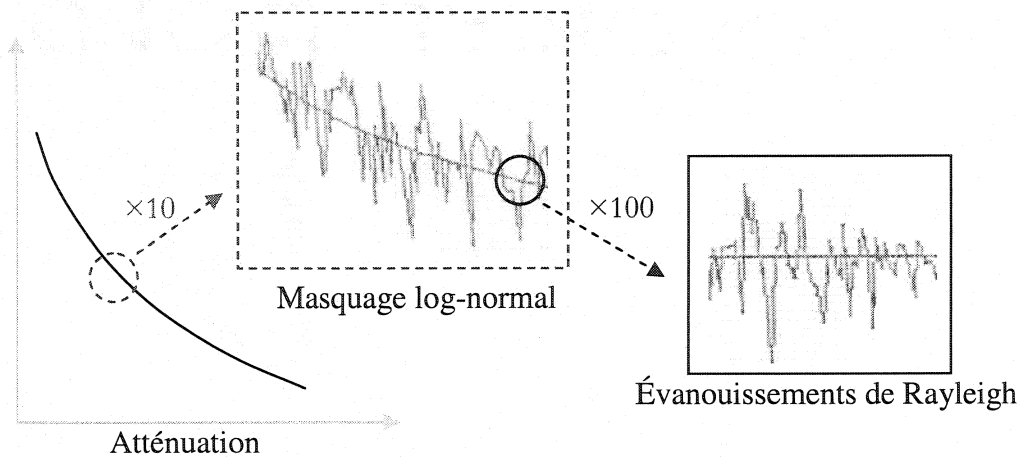


Figure 2.2 Dégradation de la puissance tenant compte des trois phénomènes : l'atténuation, le masquage log-normal et les évanouissements de Rayleigh

Modèles de la propagation radio

Considérons deux unités Ad-hoc mobiles u et v et supposons ces mobiles munis d'antenne omnidirectionnelles de surfaces effectives A_u^e et A_v^e , de gains G_u et G_v , et pouvant émettre sur une fréquence f avec des puissances P_u et P_v respectivement. Vu que les deux unités mobiles sont omnidirectionnelles, la puissance transmise dans l'espace libre est répartie uniformément sur toute sphère de rayon ρ .

Ainsi, dans l'espace libre, la puissance du signal reçue de v par u peut être exprimée par :

$$P_v(u) = \frac{A_u^e G_v P_v}{4\pi\rho^2}$$

Or $A_u^e = \frac{\lambda^2}{4\pi} G_u$ avec $\lambda = \frac{c}{f}$ la longueur d'onde d'opération

Donc, l'expression générale de la puissance dans l'espace libre s'exprime comme suit :

$$P_v(u) = G_u \left(\frac{c}{4\pi f} \right)^2 \cdot \frac{G_v P_v}{\rho^2} \quad (2.1)$$

Selon cette formule, la puissance reçue est inversement proportionnelle à la distance entre l'émetteur et le récepteur. Toutefois, elle suppose qu'il existe un chemin direct entre ces deux unités. En réalité, l'existence d'un chemin direct est rarement assurée et la puissance est alors inversement proportionnelle à ρ^α , avec $2 \leq \alpha \leq 6$ suivant la densité des constructions environnantes. Le Tableau 2.1 donne des valeurs empiriques pour α pour différents environnements (Rappaport, 2002).

Tableau 2.1 Coefficient α selon différents environnements

Environnement	α
Espace libre	2
Urbain	2.7 à 3.5
Urbain ombré	3 à 5
Intérieur dégagé	1.6 à 1.8
Intérieur obstrué	4 à 6
Intérieur d'une usine	2 à 3

Ainsi, la formule (2.1) devient :

$$P_v(u) = G_u \cdot \left(\frac{c}{4\pi f} \right)^2 \cdot \frac{G_v P_v}{\rho^\alpha}$$

En passant à l'expression de la puissance en décibel, on a :

$$P_v(u)_{[dB]} = 10 \cdot \log_{10} \left(G_u \cdot \left(\frac{c}{4\pi f} \right)^2 \cdot G_v P_v \right) - \alpha \cdot 10 \log(\rho) \quad (2.2)$$

Toutefois, le modèle ci-dessus ne tient nullement compte des détails de l'environnement de propagation. En effet, pour des environnements de propagations de même nature (urbain, forestier...) caractérisés par la même valeur α , ainsi que pour la même distance émetteur-récepteur ρ , les valeurs de la puissance mesurée peuvent être largement différentes de celles prédites par la formule (2.2). En réalité, l'expérience montre que la valeur de la puissance pour un couple (ρ, α) donné est aléatoire et suit une

distribution log-normal autour de la valeur moyenne obtenue par la formule (2.2). En fait, l'expression de la puissance tenant compte des effets de l'atténuation et du masquage log-normal suit l'équation suivante :

$$P_u(v)_{[dB]} = 10 \cdot \log_{10} \left(G_v \cdot \left(\frac{c}{4\pi f} \right)^2 \cdot G_u P_u \right) - \alpha \cdot 10 \log(\rho) + X_\sigma \quad (2.3)$$

où X_σ est une variable aléatoire gaussienne de moyenne nulle et de dispersion σ en décibel. Notons aussi que cette formule ne tient pas compte des évanouissements dus à l'effet à petite échelle mentionné précédemment.

2.2 Nature et caractéristiques des réseaux Ad-hoc

Depuis l'apparition et le grand succès commercial de la téléphonie cellulaire, le développement d'équipements mobiles n'a cessé de prendre de l'ampleur. Grâce aux assistants personnels et aux ordinateurs portables, l'utilisateur devient de plus en plus nomade. Dans un tel contexte, il n'est pas surprenant de voir apparaître les réseaux Ad-hoc, dont nous donnons ici une caractérisation. Ces réseaux, de par leur nature, présentent des propriétés et des particularités qui les mettent à part de tous les autres genres de réseaux sans fil.

La communication via les réseaux sans fil traditionnels repose essentiellement sur des infrastructures fixes tels que les stations de base, les commutateurs, etc. Toutefois, il se trouve bien des cas où l'infrastructure dédiée à la communication est inexistante. Dans d'autres cas, cette infrastructure existe mais demeure mal adaptée ou peu fonctionnelle. Malheureusement, dans de telles situations, il arrive souvent que l'échange d'information, ainsi que la fiabilité, la sécurité et la qualité de cet échange soient vitaux et d'une extrême urgence. C'est ce qui se produit en cas de guerre, d'exploration spatiale ou de désastre majeur. Dans ce contexte, les usagers peuvent se fier aux réseaux Ad-hoc.

Les réseaux Ad-hoc se distinguent des autres formes de réseaux sans fil par une absence totale d'infrastructure fixe. Un nombre quelconque de terminaux mobiles équipés de cartes d'interface radio et des couches protocolaires adéquates suffisent pour former un réseau Ad-hoc. Aucun administrateur n'est donc requis pour la gestion, la configuration

et la maintenance de tels réseaux. En effet, les nœuds mobiles (terminaux) jouent à la fois le rôle de terminaux et de routeurs, et parfois le rôle de commutateurs ou d'administrateurs locaux, formant ainsi une architecture de communication totalement répartie (Pierre, 2003).

Les caractéristiques principales des réseaux Ad-hoc découlent de la nature même de ces réseaux et peuvent être résumées dans les points suivants. D'abord, les nœuds mobiles étant libres de se déplacer arbitrairement, la topologie des réseaux Ad-hoc peut changer aléatoirement et de façon rapide. Ces mêmes nœuds combinent les rôles de routeurs et de station hôte permettant ainsi, à la fois, d'échanger de l'information et d'acheminer les messages d'un usager à un autre. Ensuite, ces réseaux ne requièrent l'utilisation d'aucune infrastructure fixe et d'aucune administration centralisée. Finalement, du fait que la mobilité des nœuds est arbitraire, les stations hôtes sont alimentées par des sources autonomes d'énergie ce qui engendre ainsi de fortes limitations au niveau de la puissance.

Les réseaux Ad-hoc sont avant tout un ensemble d'unités mobiles qui exploitent la portée de leurs interfaces radio afin de créer de bout en bout des liens avec les autres nœuds du réseau. Dans la majorité des cas, ces interfaces fonctionnent selon le standard IEEE 802.11 qui permet de créer des canaux de communications de type Ethernet avec un certain nombre de propriétés intrinsèques aux ondes radio.

Comme nous l'avons entrevu au début de ce chapitre, les ondes radio sont très sensibles à leur environnement de propagation. Cette sensibilité devient de plus en plus prononcée quand il s'agit de transmission à faible puissance comme il est le cas dans les réseaux Ad-hoc. En fait, une variation à priori anodine dans l'environnement de propagation comme fermer ou ouvrir une porte par exemple peut conduire au bris d'un lien radio existant (Chaudet et Lassous, 2003). De plus, les réseaux Ad-hoc sont des réseaux totalement mobiles. Ceci veut dire que la topologie de ces réseaux peut évoluer aléatoirement et sans préavis. Ainsi il devient très difficile, voir pratiquement impossible de prévoir avec certitude si un lien radio établi à un instant donné existera à une date ultérieure ou pas. En conséquence, il est encore plus difficile de déterminer les

caractéristiques qu'un lien radio peut avoir dans le future. Ainsi, du fait de la mobilité, la qualité des liens (en terme de bande passante et de délai) peut varier sans qu'il soit possible de le prévoir à l'avance à un coût raisonnable (Chaudet et Lassous, 2003).

Par ailleurs, le canal radio est un medium partagé ce qui crée des problèmes d'accès. Pour gérer ces problèmes, les réseaux Ad-hoc utilisent un protocole d'accès de type CSMA/CA. Ce protocole a été conçu afin de réduire les collisions de paquet au prix du délai d'accès au medium. En effet, ce protocole impose l'utilisation de mécanismes d'acquiescement afin de détecter les éventuelles collisions ce qui introduit des transmissions supplémentaires et augmente le délai. En fait, lorsqu'un mobile veut émettre un paquet, il commence par scruter le canal radio. Si celui-ci est occupé, la transmission est remise à plus tard. Cette solution pose à elle-même plusieurs défauts. Le premier est dit le problème de la station exposée et illustré à la Figure 2.3 (Chaudet et Lassous, 2003).



Figure 2.3 Problème de la station exposée

Dans ce cas de figure, les deux émetteurs peuvent transmettre en même temps sans risque de collision à la destination, mais ceci est impossible à cause des spécificités du protocole d'accès. Un autre défaut est celui connu sous le nom de la station cachée (Wu et Harms, 2001) et est présentée dans l'exemple de la Figure 2.4.

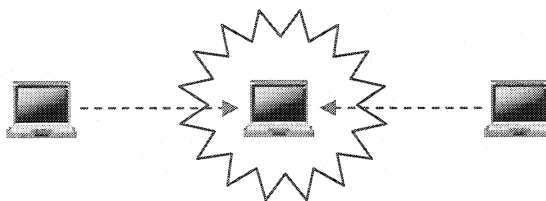


Figure 2.4 Problème de la station cachée

Dans cet exemple, les deux émetteurs considèrent que le médium est libre et commencent ainsi à émettre. Or, cette situation conduit à des collisions au niveau de la station destination.

D'un autre côté, l'exploitation de médium radio pose le problème de la sommabilité des interférences. Comme on peut le voir à la Figure 2.5 (Chaudet et Lassous, 2003), les transmissions des mobiles hors de portée de transmission ne gênent pas individuellement, par contre l'ajout des bruits dus à ces transmissions crée une interférence à la destination.

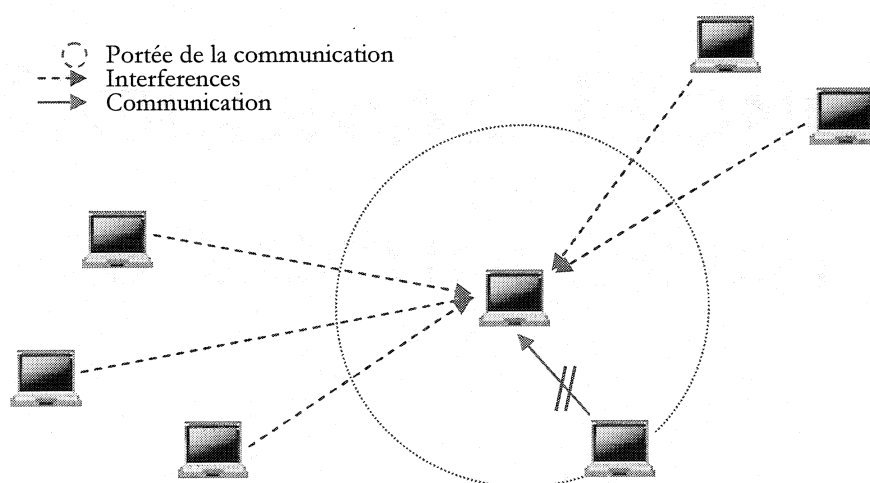


Figure 2.5 Problème de sommabilité des bruits

2.3 Routage dans les réseaux Ad-hoc

Comme tout système de communication, le but d'un réseau Ad-hoc est de transmettre une information d'une origine vers une destination donnée. Or, nous l'avons vu, les réseaux Ad-hoc sont construits sur un médium changeant et peu stable, avec un fort délai de transmission de paquets et une bande passante relativement faible. Afin de s'adapter à ces contraintes, plusieurs travaux ont été réalisés dans le domaine du routage. Ces derniers se sont principalement partagés suivant deux approches opposées : proactive et réactive. Dans ce qui suit, nous présentons une description des deux approches

évoquées ainsi que celle des principaux protocoles de routage qui les implémentent. Finalement, nous présentons une comparaison entre les deux approches mettant en phase respectivement leurs avantages et inconvénients.

2.3.1 Protocoles de routage proactifs

Dans cette catégorie, se classent tous les protocoles qui essaient de maintenir consistante et mise à jour l'information concernant le routage à partir de chaque nœud vers n'importe quel autre nœud du réseau. Ces protocoles requièrent d'une part que chaque nœud maintienne une ou plusieurs tables de routage, et d'une autre part que chaque nœud propage des informations sur les changements de topologies qu'il détecte afin que tous les nœuds gardent une vue cohérente et mise à jour sur la structure du réseau. Toutefois, ces protocoles diffèrent les uns des autres dans le nombre de tables de routage à conserver et sur la manière de les entretenir (Royer et Toh, 1999). Parmi les nombreux protocoles de cette catégorie, nous en présentons deux : DSDV et CGSR.

Protocole DSDV

Le protocole DSDV (Perkins et Bhagwat, 1994), pour *Destination-Sequenced Distance-Vector Routing*, est un protocole proactif basé sur une amélioration du mécanisme de routage de Ford-Bellman. Cette amélioration consiste essentiellement à éliminer les cycles qui peuvent être construits à cause de la symétrie des liens radios.

Dans le cas du DSDV, chaque nœud du réseau doit maintenir une table de routage dans laquelle sont enregistrées toutes les destinations que le nœud détecte, le nombre de sauts pour les atteindre, ainsi que le premier nœud, sur la route à prendre pour les atteindre. Chaque entrée, dans cette table, est identifiée par un numéro de séquence qui est attribué par les nœuds destinations. Ce numéro de séquence permet, en fait, de distinguer les itinéraires récents des anciens, évitant ainsi la formation de boucles lors de l'acheminement des paquets. Ainsi, un paquet utilisant une route marquée d'un numéro de séquence n ne peut emprunter que des portions de routes dont le numéro de séquence est supérieur ou égal à n . Afin de maintenir à jour les tables de routage, celles-ci sont

transmises périodiquement, dans tout le réseau. Malheureusement, ceci peut augmenter considérablement le trafic, dans le réseau. Dans le but d'alléger la quantité du trafic reliée à cet échange, deux types de paquets peuvent être utilisés selon les circonstances. Le premier, dit *Full Dump* (mise à jour complète), permet de diffuser toutes les informations de routage disponibles dans la table et peut exiger l'utilisation de plusieurs *NPDU* (*Network Packet Data Unit*). Ce genre de paquet n'est utilisé qu'occasionnellement. En effet, lors de changements de la topologie, de plus petits paquets de tailles généralement 1 *NPDU* sont transmis, afin de signaler seule l'information de routage changée depuis le dernier *Full Dump*.

Quant aux diffusions d'itinéraires, elles contiennent notamment l'adresse de la destination, le nombre de sauts pour l'atteindre, le numéro de séquence de l'information de routage, ainsi qu'un numéro unique qui identifie la diffusion. De ce fait, l'itinéraire marqué du numéro de séquence le plus récent est toujours utilisé, et, dans le cas où deux mises à jour auraient le même numéro de séquence, l'itinéraire ayant la métrique la plus petite est utilisé afin de raccourcir le chemin.

Protocole CGSR

Le protocole CGSR (Chiang, 1997), pour *Clusterhead Gateway Switch Routing*, est un protocole de routage proactif qui diffère du protocole précédent dans l'adressage et l'organisation du réseau. En effet, au lieu de garder une vue horizontale du réseau, CGSR propose d'organiser le réseau en petites régions et d'effectuer le routage selon ces régions. L'idée majeure est de décomposer le réseau en petites régions connexes, désignant chacune un nœud chef appelé *Cluster Head*. L'avantage de cette approche est que chaque région est « localement administrée » par le chef de région, qui s'occupe d'établir un certain ordre suivant certaines règles. À priori, ceci permet d'avoir certaines garanties sur l'allocation de la bande passante, l'accès au médium, etc. Un algorithme de choix de *Cluster Head* est alors utilisé pour élire un nœud comme chef de région, en appliquant un algorithme réparti. Cependant, l'organisation du réseau sous forme de régions connexes comporte plusieurs inconvénients. En effet, des changements fréquents

de la topologie induisent plusieurs opérations d'élection de *Cluster Head*, ce qui compromet l'exécution du protocole de routage. En fait, dans ce cas, les nœuds seront occupés dans le choix du *Cluster Head* plutôt que dans le relais des paquets. Pour cela, au lieu de re-élire un *Cluster Head*, à chaque fois que des changements dans une région surviennent, un algorithme nommé LCC (Least Cluster Change) est introduit. En utilisant LCC, les *Cluster Head* changent seulement dans deux cas. Le premier, se réalise quand deux *Cluster Head* entrent en contact, c'est-à-dire dans la portée l'un de l'autre. Le deuxième surgit quand un nœud se déplace hors du contact de tous les *Cluster Head*.

Comme protocole de routage, CGSR emploie DSDV. Par conséquent, il utilise les mêmes entêtes de message que dans DSDV. Cependant, il diffère de DSDV en employant une approche hiérarchique de cheminement, pour conduire le trafic de la source à la destination. En effet, les paquets sont toujours transmis du *Cluster Head* vers un nœud passerelle, puis vers un *Cluster Head*. Quant à un nœud passerelle, il est dit d'un nœud qui se trouve dans la portée de communication de deux *Cluster Head* ou plus. D'une façon générale, le schéma de livraison de message est le suivant : un paquet, envoyé par un nœud, est d'abord conduit à son *Cluster Head*, ensuite le paquet est conduit du *Cluster Head* à un nœud passerelle puis à un autre *Cluster Head*, et ainsi de suite jusqu'à ce que le *Cluster Head* du nœud destinataire soit atteint. Le paquet est alors transmis à la destination. La Figure 2.6 illustre un exemple de ce genre de cheminement (Royer et Toh, 1999).

Pour utiliser cette méthode, chaque nœud doit maintenir une table d'association contenant, pour chaque nœud mobile du réseau, son couple (*Nœud, Cluster Head*). Ces tables sont diffusées périodiquement en utilisant DSDV ce qui permet aux nœuds de mettre à jour leur propre table à la réception des informations de leurs voisins.

Outre les tables d'association *nœud/Cluster Head*, chaque nœud doit également construire et maintenir consistante une table de routage qui est employée pour déterminer le prochain nœud, afin d'atteindre une destination donnée. Ainsi, à la réception d'un paquet, un nœud consultera ses deux tables pour déterminer le *Cluster Head* le plus proche pour acheminer le paquet vers la destination.

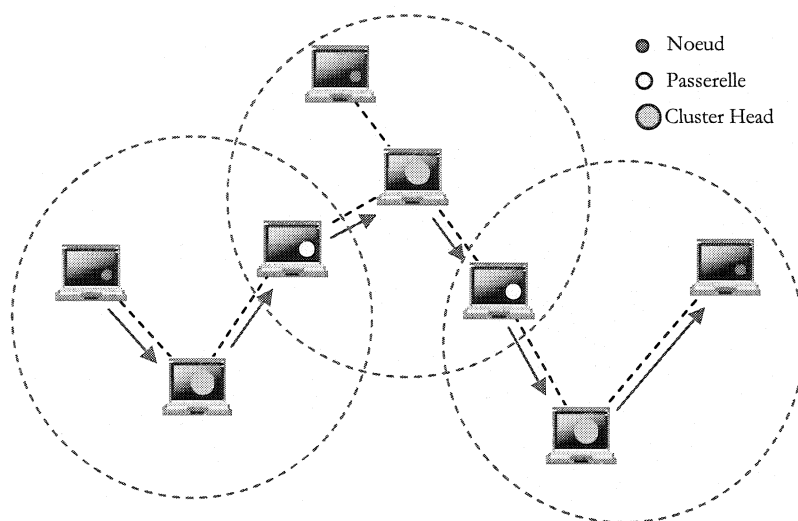


Figure 2.6 Acheminement d'un message suivant le protocole CGSR

2.3.2 Protocoles de routage réactifs

L'autre approche de routage, concurrente à l'approche proactive, est celle réactive. Dans cette approche, les routes ne sont créées que si elles sont demandées par un nœud désirant transmettre un paquet vers un autre nœud destination. En effet, lorsqu'un nœud source désire avoir un chemin vers un autre nœud destination, il lance en diffusion un processus de découverte d'itinéraires. Ce processus se termine lorsque l'on trouve le chemin désiré ou, si toutes les combinaisons de chemins possibles ont été examinées. Une fois un itinéraire établi, il est maintenu consistant grâce à un processus d'entretien d'itinéraires. Ce processus est utilisé jusqu'à ce que la destination devienne inaccessible en utilisant tous les chemins à partir de la source ou, jusqu'à ce que l'itinéraire ne soit plus désiré.

Protocole AODV

Le protocole de routage AODV (Perkins et Royer, 1999), pour *Ad-hoc On-Demand Distance Vector Routing*, est, en fait, une amélioration de l'algorithme DSDV, précédemment décrit. AODV est une amélioration de DSDV car, il réduit le nombre de

diffusions en créant des itinéraires à la demande, par opposition à DSDV qui maintient une liste complète de tous les itinéraires à toutes les destinations.

Quand un nœud source désire envoyer un message à un certain nœud destinataire et n'a pas un itinéraire valide pour cette destination, il lance un processus de découverte de chemin, pour localiser le nœud destination. Il diffuse ainsi un paquet de demande d'itinéraires (RREQ) à ses voisins, lesquels font suivre la demande à leurs voisins, et ainsi de suite. Le processus se termine lorsque la destination est atteinte ou, lorsqu'un nœud intermédiaire avec un itinéraire assez récent, à la destination, est localisé. La Figure 2.7 (Royer et Toh, 1999) illustre la propagation de la diffusion du RREQ à travers le réseau.

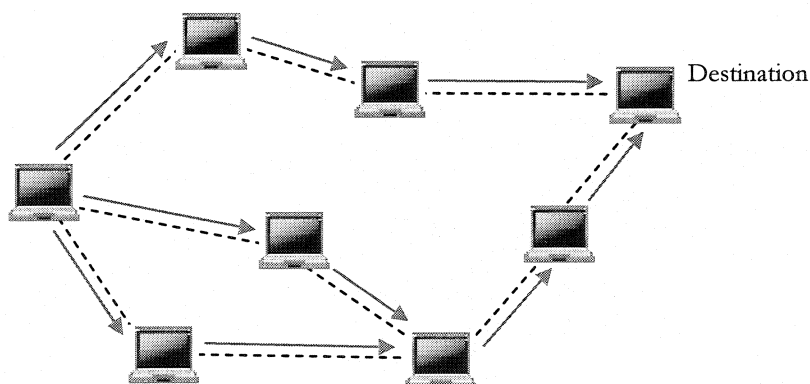


Figure 2.7 Propagation du RREQ dans le protocole AODV

À l'instar de DSDV, AODV utilise des numéros de séquence de destination, pour assurer que tous les itinéraires sont sans boucle et contiennent l'information d'itinéraire la plus récente. Chaque nœud conserve son propre compteur de numéro de séquence, aussi bien qu'un numéro d'identification de la diffusion. Ce dernier est incrémenté pour chaque requête RREQ que le nœud source lance. Cette identification, ajoutée à l'adresse IP du nœud, identifie totalement un message RREQ. En plus du numéro de séquence et l'identification de la diffusion, le nœud source inclut, dans le RREQ, le numéro de séquence le plus récent qu'il a pour la destination. De cette manière, on est assuré d'obtenir l'information de routage la plus récente. En effet, les nœuds intermédiaires ne peuvent répondre au RREQ seulement dans le cas où ils ont un itinéraire à la destination,

dont le numéro de séquence correspondant est supérieur ou égal à celui contenu dans le RREQ.

Pendant le processus de transmission des RREQ, les nœuds intermédiaires, qui reçoivent le RREQ, enregistrent dans leurs tables l'adresse du premier voisin duquel ils ont reçu la première copie du RREQ. Ce mécanisme permet, en fait, d'établir des chemins inverses qui seront utilisés par la suite. Si des copies du même RREQ sont reçues plus tard, ces paquets sont détruits. Une fois que le RREQ atteint la destination ou un nœud intermédiaire ayant un itinéraire assez récent vers la destination, ce nœud répond en envoyant un paquet de réponses d'itinéraires (RREP) en utilisant le chemin inverse établi. Ce mécanisme est illustré à la Figure 2.8 (Royer et Toh, 1999).

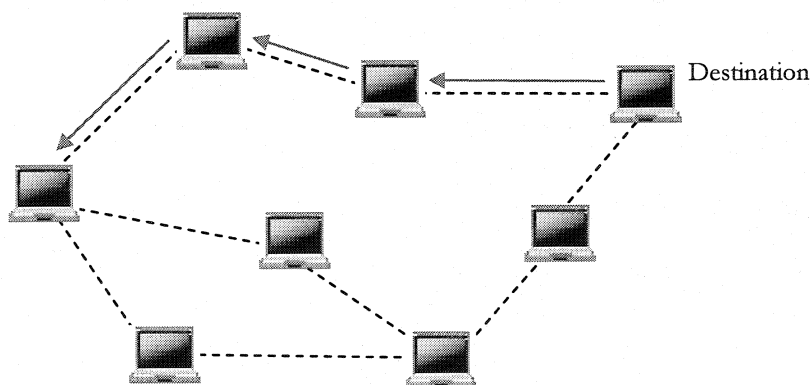


Figure 2.8 Acheminement du RREP dans le protocole AODV

Lors du processus d'acheminement du RREP vers le nœud source, tous les nœuds intermédiaires, qui se trouvent sur le chemin inverse, enregistrent les entrées d'itinéraires qui les mènent directement vers le nœud, qui a lancé le RREP. Ces entrées indiquent l'itinéraire actif. À chaque itinéraire est associé un temps d'expiration d'itinéraires, qui permettra la suppression de ce dernier s'il n'est pas employé ou demandé durant un certain temps de vie. À ce point, il faut noter que, du moment que le RREP est expédié à travers le chemin inverse, établi par le RREQ, AODV ne supporte que l'utilisation des liens symétriques.

Une fois établis, les itinéraires sont maintenus de la manière suivante : si le mouvement du nœud source cause la rupture du chemin à la destination, il peut re-initier le protocole de découverte d'itinéraires pour trouver un nouvel itinéraire. Par contre, si un nœud intermédiaire se déplace, quittant alors l'itinéraire, son voisin ascendant (le premier voisin de la portion de route vers la source) note le mouvement et propage un message de notification d'échec de lien (un RREP avec une métrique infinie) à chacun de ses voisins ascendants actifs, pour les informer de la suppression de cette portion de route. Ces nœuds propagent à leur tour l'avis d'échec de lien à leurs voisins ascendants, et ainsi de suite, jusqu'à ce que le nœud source soit atteint. Le nœud source peut alors choisir de relancer le processus de découverte d'itinéraires pour la destination, si un itinéraire est encore désiré.

Un autre aspect du protocole est l'utilisation par chaque nœud des messages *hello*, à l'aide de diffusions locales et périodiques, pour informer chaque nœud de son voisinage de sa présence et de la présence des autres nœuds dans son voisinage. Cependant, l'utilisation des messages *hello* n'est pas exigée. En effet, les nœuds écoutent les transmissions des paquets de données et s'assurent ainsi que les nœuds responsables sont toujours atteignables. Si une telle retransmission n'est pas entendue, le nœud peut employer n'importe quelle technique, y compris la réception des messages *hello*, pour déterminer si le nœud voisin est toujours dans la portée de la communication. Les messages *hello* peuvent aussi énumérer les autres nœuds dont un mobile a entendu une transmission, rapportant ainsi une plus grande connaissance de la connectivité du réseau.

Protocole DSR

Le protocole DSR (Johnson et Maltz, 1996), pour *Dynamic Source Routing*, est un protocole de routage sur demande basé sur le concept du routage de source. Les nœuds mobiles doivent maintenir des caches qui contiennent les itinéraires (vers des sources) que le mobile arrive à capter. Les entrées dans le cache d'itinéraire sont continuellement mises à jour si de nouveaux itinéraires sont découverts.

Le protocole se compose de deux phases principales : la découverte des itinéraires et leur entretien. Quand un nœud mobile désire envoyer un paquet à une certaine destination, il consulte d'abord son cache d'itinéraire pour déterminer s'il a déjà une route à la destination, qu'il utilisera pour envoyer le paquet. Si le nœud n'a pas un tel itinéraire, il lance la procédure de découverte d'itinéraire en diffusant un paquet de demande d'itinéraire. Cette demande d'itinéraire contient l'adresse de la destination, ainsi que l'adresse du nœud source et un numéro d'identification. Chaque nœud qui reçoit le paquet de demande vérifie s'il a une route à la destination. Si ce n'est pas le cas, il ajoute sa propre adresse aux enregistrements contenant le chemin pris par le paquet et l'expédie le long de ses liens sortants. Pour limiter le nombre de demandes d'itinéraire propagées, un mobile expédie la demande d'itinéraire seulement s'il l'a reçu pour la première fois et si son adresse n'apparaît pas dans les enregistrements contenant le chemin du paquet. Une réponse d'itinéraire est produite quand la demande d'itinéraire atteint la destination elle-même, ou un nœud intermédiaire qui dispose dans son cache d'itinéraire d'une route récente à la destination. Lorsque le paquet atteint la destination ou un tel nœud intermédiaire, le paquet contient déjà le chemin pris. La Figure 2.9 (Royer et Toh, 1999) illustre la formation de l'itinéraire pendant que la demande d'itinéraire est propagée dans le réseau.

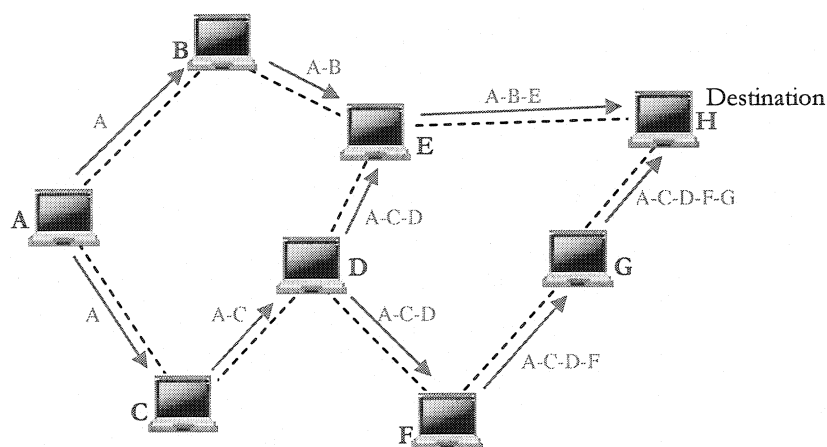


Figure 2.9 Formation de la route lors d'une découverte d'itinéraire DSR

Si le nœud produisant la réponse d'itinéraire est la destination, il place la route établie dans la demande d'itinéraire dans la réponse. Si le nœud répondant est un nœud intermédiaire, il joindra son itinéraire à la destination au chemin établi et puis enverra la réponse. Pour renvoyer cette réponse, le nœud répondant doit avoir une route vers le nœud source. S'il a cette route dans son cache, il peut l'utiliser. Autrement, si les liens sont symétriques, le nœud peut utiliser l'inverse de l'itinéraire établi dans la demande de route. Si les liens symétriques ne sont pas supportés, le nœud peut lancer sa propre demande de découverte d'itinéraire et y associer la réponse d'itinéraire qu'il a en main. La Figure 2.10 (Royer et Toh, 1999), illustre la transmission au nœud source de la réponse d'itinéraire à laquelle est associé le chemin établi.

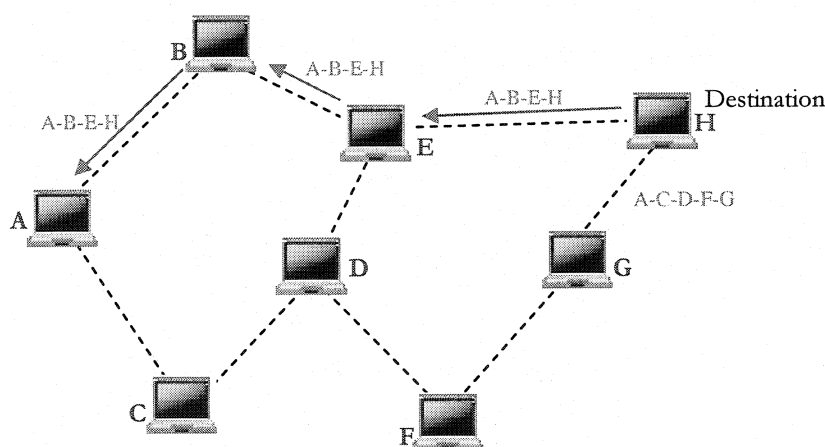


Figure 2.10 Propagation de la réponse à une demande d'itinéraire DSR

L'entretien des itinéraires est accompli à l'aide de paquets d'erreur d'itinéraire et des acquittements. Les paquets d'erreur d'itinéraire sont produits à un nœud quand la couche liaison de données rencontre un problème fatal de transmission. Quand un paquet d'erreur d'itinéraire est reçu, le nœud à l'origine de l'erreur est enlevé du cache d'itinéraire et toutes les routes contenant ce nœud sont coupées au niveau de ce dernier. En plus des messages d'erreur d'itinéraire, des acquittements sont employés pour vérifier la validité des liens. De tels acquittements incluent les acquittements passifs, où un mobile peut écouter la transmission d'un paquet le long d'une route active.

2.3.3 Analyse des deux approches

La différence essentielle entre les approches de routage proactif et réactif est que dans le premier, chaque nœud maintient, localement, une table de routage qui donne, en fonction de l'adresse de la cible, l'adresse du voisin auquel il faudra transmettre le message. Par contre, le deuxième ne nécessite pas le maintien de tables de routage dans chaque nœud, car il est basé sur une construction dynamique des routes, lorsqu'on en a besoin. Les protocoles réactifs génèrent à priori un volume plus faible de signalisation mais, en contrepartie, engendrent un délai lors de la construction (ou de la reconstruction) des routes et produisent plus difficilement des routes optimales quel que soit le critère (Chaudet et Lassous, 2003). Les protocoles proactifs, quant à eux, essaient de maintenir en permanence une route pour chaque destination dans le réseau ce qui engendre un volume de signalisation important. Une comparaison des deux approches montre que la performance de chacune dépend énormément des conditions d'utilisation. Les protocoles construisant des routes par diffusion sont plus performants que leurs concurrents lorsque le nombre de routes actives est faible ou lorsque la mobilité dans le réseau est élevée (Chaudet et Lassous, 2003).

Par ailleurs, notons l'existence d'une approche de routage hybride, pour laquelle le routage est proactif jusqu'à un certain nombre de sauts fixés, et réactif au delà. Ainsi, chaque nœuds dispose d'informations sur la topologie du réseau jusqu'à un certain nombre de sauts et effectue une recherche de route au delà. Il existe aussi certains protocoles de routage exploitant les informations obtenues grâce aux systèmes de géolocalisation GPS. Ces systèmes fournissent une localisation en trois dimensions pour l'unité mobile qui peut la transmettre à ces voisins fournissant ainsi une information plus précise sur la topologie du réseau.

Toutefois, notons que toutes ces approches souffrent de certaines faiblesses dues aux aléas du protocole d'accès sous-jacent. En fait, le standard IEEE 802.11 prévoit des débits de transmission différents selon la nature du paquet transmis. En effet, les paquets de données sont transmis à 11 Mb/s tandis que les paquets de contrôle tel que la mise à jour de la topologie ou la recherche de routes sont transmis à 2 Mb/s. Ceci veut dire que

ces derniers peuvent atteindre des distances plus grandes que celles atteintes par les paquets de données. L'information sur le voisinage se trouve ainsi erronée ce qui peut affecter considérablement les performances des protocoles de routage. D'un autre côté, les paquets de contrôle envoyés en diffusion ne sont ni protégés par un échange RTS/CTS, ni acquittés. Ce qui pourrait causer de nombreuses pertes au niveau de la signalisation et avoir un impact négatif sur le routage.

2.4 Qualité de service dans les réseaux Ad-hoc

Comme nous l'avons vu auparavant, les débits atteints aujourd'hui dans les réseaux Ad-hoc frôlent les 11 Mb/s ce qui est fort encourageant. En effet, ceci rend possible le transfert de flux multimédia soumis à de fortes contraintes. Dès lors, il est légitime de chercher à fournir aux applications des garanties de qualité de service sur le délai, les taux de pertes, la bande passante, etc. Malheureusement ceci n'est pas une tâche facile et il faut attendre encore quelques années avant de pouvoir parler d'une vraie qualité de service.

Toutefois, les travaux dans ce domaine sont souvent classés en quatre grandes catégories (Chaudet et Lassous, 2001):

- Les modèles de qualité de service définissent des architectures globales dans lesquelles des garanties peuvent être fournies ;
- Les protocoles d'accès au médium cherchent à ajouter des fonctionnalités aux couches basses du modèle OSI afin de pouvoir offrir des garanties ;
- Les protocoles de routage avec qualité de service recherchent les routes ayant suffisamment de ressources disponibles pour satisfaire une requête ;
- Les protocoles de signalisation cherchent à offrir des mécanismes de réservation de ressources indépendants du protocole de routage sous jacent ;

2.4.1 Qualité de service : définition et métriques

Habituellement, la qualité de service est définie comme un ensemble de critères qui doivent être remplis par le réseau afin qu'il puisse acheminer correctement un flux de

données entre une paire de nœuds. Ces critères sont dictés par les applications usager qui s'apprêtent à utiliser le réseau. Ainsi, le réseau est supposé garantir, en terme de performance point à point, un ensemble déterminé de paramètres quantifiables tels que la bande passante, la probabilité de perte de paquet, la gigue, etc. Un ou plusieurs de ces paramètres, munis de différents poids, peuvent être pris en compte dans une métrique pour la qualité de service.

Les métriques pour la qualité de service peuvent être concaves ou additives. En fait, si l'on considère $m(i,j)$ comme une métrique pour la qualité de service relative à un lien (i, j) , alors pour un chemin $P=(s, i, j, \dots, l, d)$, la métrique m est concave si $m(P)=\min\{m(s,i), m(i,j), \dots, m(l,d)\}$. Et la métrique m est additive si $m(P)=m(s,i)+m(i,j)+\dots+m(l,d)$. Il a été prouvé que, si la qualité de service rassemble au moins deux critères à métriques additives, alors les problèmes de routage avec qualité de service sont NP-difficiles (Wu et Harms, 2001).

2.4.2 Modèles de qualité de service

Un modèle de qualité de service définit les types de service qui peuvent être fournis par un réseau et certains mécanismes utilisés afin d'offrir ces services. Ces mécanismes couvrent les fonctionnalités que doit fournir le protocole de routage et l'architecture des nœuds, etc

FQMM

Le modèle de qualité de service pour les réseaux Ad-hoc proposé par Xiao et al. (2000) est le modèle FQMM (*Flexible QoS Model for MANET*). Ce dernier tient compte des caractéristiques des réseaux Ad-hoc et s'inspire des modèles de qualité de service du monde filaire. Il se situe entre les deux approches *IntServ* et *DiffServ*. Ceci lui permet de tirer profit des avantages des deux modèles et d'offrir une plus grande granularité. Ainsi, il considère les deux types de services *par flot* comme dans *IntServ* et *par classe* comme dans *DiffServ*. À l'instar de *DiffServ*, FQMM définit trois types de nœuds : les nœuds d'entrée ou émetteurs, les nœuds intermédiaires qui transmettent les données et les nœuds

finaux à qui les données sont destinées. Toutefois, notons que compte tenu du fait que dans les réseaux Ad-hoc chaque nœud mobile assure la fonction de routeur, chaque mobile joue différents rôles suivant les flux. Ceci dit, le conditionnement du trafic est laissé à la charge des émetteurs qui s'occupent du marquage du trafic, de l'annulation ou de la retransmission des paquets suivant son profil. Aussi, FQMM requiert l'utilisation d'un protocole de routage capable d'offrir une certaine qualité de service, c'est-à-dire capable de rechercher des routes satisfaisant certaines contraintes.

2.4.3 Couche MAC

Comme nous l'avons mentionné précédemment, les réseaux Ad-hoc reposent sur un medium peu fiable. Ainsi, l'utilisation d'un protocole d'accès au medium efficace est primordiale. En effet, ce protocole prend en charge plusieurs opérations vitales pour les réseaux Ad-hoc. Il permet d'éviter les collisions, d'assurer un bon partage de la bande passante et de résoudre les problèmes spécifiques aux transmissions radio. Toutefois, il serait bien plus favorable que ce protocole puisse gérer une certaine qualité de service. Et c'est dans cette optique que le protocole MACA/PR a été développé.

a. MACA/PR

Le protocole MACA/PR (Gerla et Lin, 1997) pour *Multiple Access Collision Avoidance with Piggyback Reservation*, fournit une transmission rapide et fiable des datagrammes ordinaires ainsi qu'une garantie de bande passante pour le trafic en temps réel.

Pour la transmission des datagrammes ordinaires dans le protocole MACA/PR, un hôte ayant un paquet à envoyer doit d'abord acquérir une fenêtre libre dans la table de réservation (RT), qui enregistre toutes les fenêtres d'envoi et de réception de n'importe quelle station dans l'étendue de transmission du nœud. Il attend alors un temps aléatoire additionnel de l'ordre de l'envoi et la réception d'un message vers un nœud voisin. Si le canal est libre, il procède à une transmission à l'aide d'un dialogue RTS/CTS-PKT/ACK.

Si le canal est occupé, il attend jusqu'à ce que le canal devienne libre et répète le procédé précédent.

Pour la transmission des paquets en temps réel, le comportement de MACA/PR est différent. Afin de transmettre le premier paquet de données d'un échange en temps réel, l'expéditeur S initie un dialogue RTS/CTS et poursuit alors avec un dialogue PKT/ACK si le CTS est reçu. Pour les paquets suivants de données d'un échange en temps réel, seulement le mécanisme PKT/ACK est nécessaire. Notons que si l'expéditeur ne reçoit pas plusieurs ACK, il rétablit la connexion en utilisant un dialogue RTS/CTS. En temps réel, MACA/PR ne retransmet pas les paquets après collision.

Afin de réserver la bande passante pour le trafic en temps réel, l'information d'établissement d'un échange en temps réel est diffusée dans les en-têtes des PKT et des ACK. L'expéditeur S inclut l'information de réservation pour sa prochaine transmission de paquets de données dans le paquet courant de données (PKT). Le récepteur D insère la réservation dans sa table de réservation (RT) et confirme ceci avec un ACK à l'expéditeur. Les voisins du récepteur D reporteront leurs transmissions une fois l'ACK reçu. En outre, à partir du ACK, ils savent également la prochaine période prévue pour une réception d'un paquet par D et évitent ainsi la transmission au moment où D reçoit le prochain paquet de données de S . Les paquets en temps réel sont protégés contre les stations cachées par la propagation et l'entretien des tables de réservation pour les voisins, et non par les dialogues de RTS/CTS. Ainsi, en utilisant ce mécanisme de réservation et d'entretien des tables de réservation, la largeur de bande est réservée et garantie pour le trafic en temps réel.

2.4.4 Routage avec qualité de service

Les protocoles de routage avec qualité de service recherchent des routes qui disposent d'assez de ressources pour satisfaire certains critères. Plusieurs de ces critères, appelés aussi métriques, sont utilisés tels que le délai, la bande passante ou encore le coût de transmission. Une combinaison de ces métriques est aussi souvent utilisée. Dans ce

contexte, plusieurs protocoles ont été développés. Dans ce qui suit, nous en présentons deux parmi les plus connus, à savoir TBR et CEDAR.

Le protocole TBR

Compte tenu du coût d'accès au médium élevé dans les réseaux Ad-hoc, la recherche de route par exploration exhaustive peut devenir très coûteuse. Pour cela, le protocole TBR (*Ticket Based Probing*) développé par Chen et Nahrstedt (1999) propose de limiter le nombre de routes candidates (à explorer) tout en tenant compte des garanties sur la qualité de service. Ainsi, la découverte de route est limitée car l'émetteur va associer à chaque demande de route un certain nombre de tickets qui va restreindre la diffusion des requêtes. Le nombre de tickets est basé sur l'information sur les caractéristiques des liens d'un nœud vers ses voisins immédiats. Un ticket correspond à une recherche de chemin; et une demande de route devrait porter au moins un ticket. Ainsi, le nombre maximum de chemins recherchés est lié par les tickets publiés par la source. Quand un nœud intermédiaire reçoit une demande de route avec n tickets, il décide, en se basant sur ses informations locales, s'il doit et comment il doit dédoubler ces n tickets, et où retransmettre les requêtes. Quand la destination reçoit une demande de route, un chemin possible de la source à la destination est trouvé.

Dans ce cas, deux problèmes sont alors étudiés. D'abord, établir des routes les plus proches de l'optimal possible, de moindre coût et avec des contraintes de délai, ce qui est un problème NP complet. Ensuite, trouver des routes de moindre coût avec des contraintes de débit, ce qui peut être résolu en un temps polynomial. Et afin d'augmenter la probabilité de trouver une route satisfaisante, on utilise deux types de tickets : les tickets jaunes qui permettent de rechercher des chemins respectant la contrainte imposée et les tickets verts permettant d'obtenir des solutions de faible coût.

Malgré le fait que les nœuds ne connaissent que leur voisinage immédiat, *Ticket Based Probing* est efficace puisqu'il permet de trouver des routes avec une probabilité proche de celle algorithmes basés sur l'inondation du réseau et meilleure que celle

d'algorithmes recherchant un plus court chemin. Il permet en outre de trouver des routes de plus faible coût que ces deux types d'algorithmes.

Le protocole CEDAR

Le protocole CEDAR (*Core Extraction Distributed Ad-Hoc Routing*) a été proposé par Sinha et al. (1999), comme un protocole de routage pour assurer une certaine qualité de service dans les petits et moyens réseaux Ad-hoc, c'est-à-dire comportant entre une dizaine et une centaine de nœuds. Il établit dynamiquement un réseau cœur puis propage des informations sur les liens les plus stables et disposant d'une bonne bande passante à travers les nœuds de ce réseau cœur. Le processus de découverte de route est un processus réactif qui est effectué par les nœuds du réseau cœur en utilisant seulement des informations locales. CEDAR consiste principalement à effectuer trois opérations essentielles. La première permet d'établir le réseau cœur. Il s'agit dans ce cas d'élire un ensemble de nœuds qui formeront le réseau cœur. Ces nœuds maintiendront à jour l'information topologique sur leurs domaines et effectueront les recherches de route. L'ensemble des nœuds du réseau cœur doit approximer un ensemble dominant minimum. La deuxième consiste en la propagation des états des liens. En effet, la qualité de service offerte par le protocole CEDAR est garantie grâce à la propagation à travers le réseau cœur d'informations sur la disponibilité de la bande passante dans des liens relativement stables. L'idée majeure est que l'information sur ce genre de liens doit être connue et propagée le plus possible dans le réseau, tandis que celle concernant des liens volatiles peut rester locale. Quant à la troisième fonctionnalité, elle concerne la découverte des routes. Cette découverte commence d'abord par la mise au point d'un chemin formé de nœuds du réseau cœur, reliant la source à la destination. En utilisant les informations sur ce chemin, CEDAR tente d'une manière itérative de trouver le meilleur chemin satisfaisant les contraintes de bande passante.

2.4.5 Protocoles de signalisation

Afin d'assurer le bon fonctionnement des réseaux Ad-hoc, plusieurs informations de contrôle doivent être transmises à travers tous le réseau. Ces informations peuvent être de différentes natures. En effet, il peut s'agir d'informations topologiques, de demande de recherche de route ou encore de rapport sur l'état du réseau et des ressources disponibles. Une manière de transmettre toutes ces informations est d'utiliser un protocole de signalisation. Celui-ci sert à définir les données à échanger afin de remplir une tâche particulière ainsi que la manière de les échanger. Dans ce contexte, nous présentons le protocole de signalisation INSIGNIA (Lee et al., 2000).

INSIGNIA (*In-band Signaling Support for QoS in Mobile Ad-hoc Networks*) est le premier protocole de signalisation développé pour les réseaux Ad-hoc. Ce protocole est de type *in-band*, ce qui veut dire que la signalisation permettant d'effectuer des réservations de bande passante dans les réseaux Ad-hoc est incluse dans les entêtes des paquets de données.

INSIGNIA fonctionne sur la base d'une granularité par flot. Il concerne principalement les applications adaptatives capables de modifier leur comportement en fonction de la quantité de bande passante qui leur est allouée. Ainsi, chaque application est tenue de spécifier deux niveaux de qualité de service. Le premier niveau, dit de base, permet de spécifier la bande passante minimale nécessaire au trafic. Quant au deuxième, dit amélioré, il permet de spécifier le débit optimal à atteindre lorsque les ressources sont disponibles. En outre, INSIGNIA n'est pas lié à l'utilisation d'un protocole de routage particulier et supporte des changements relativement rapides au niveau de la topologie.

L'idée majeure dans INSIGNIA est d'utiliser une option de l'entête IPv4 afin d'y inclure les informations de signalisation. Ce champ de 26 bits indique si un paquet fait partie d'un flux privilégié ou non, le niveau de qualité de service requis ainsi que l'importance de chaque paquet dans le flux de données. Ce champ est spécifié par l'application émettrice du flux de données. Il pourra être par la suite modifié tout au long du chemin afin d'acheminer des informations sur l'état actuel du réseau jusqu'au destinataire. D'autre part, des informations sur l'état de la route, établies pour un flux donné sont

périodiquement émises par le récepteur vers l'émetteur afin de permettre à ce dernier d'adapter son transfert suivant l'état de la route. Les demandes de réservation de bande passante sont effectuées dans l'entête du premier paquet du flux. De plus, INSIGNIA comporte un mécanisme de reconstruction locale de routes. Il offre ainsi des performances encourageantes pour des réseaux dans lesquels la mobilité est moyenne.

CHAPITRE 3

MODÉLISATION DES RÉSEAUX AD-HOC : APPLICATION AU *FLOODING*

Comme nous l'avons souligné au cours du chapitre d'introduction, l'approche classique de modélisation des réseaux Ad-hoc, inspirée de celle utilisée dans le monde filaire, laisse à désirer. C'est ainsi que nous tentons, dans le présent chapitre, d'analyser ses faiblesses et de recenser ses limitations. Puis, nous présentons une nouvelle approche plus adaptée pour modéliser les réseaux Ad-hoc en remédiant aux limitations identifiées. À la suite de cela, nous essayerons de mettre en application cette approche pour résoudre le problème du *flooding* couplé à la mobilité. Pour ce faire, nous rappelons le concept du *flooding* en vue de mettre en évidence les problèmes que pose l'application de ce concept et la nécessité d'en offrir une bonne gestion. Finalement, nous modélisons le problème en question et nous présenterons un algorithme pour le résoudre.

3.1 Modélisation des réseaux Ad-hoc

Dans cette section, nous présentons dans un premier temps l'approche traditionnelle de modélisation des réseaux Ad-hoc et nous en dévoilons les principaux inconvénients et limitations. Dans un deuxième temps, nous proposons une nouvelle approche pour la modélisation des réseaux en question, approche qui permet de pallier les problèmes de l'approche classique.

3.1.1 Approche classique et limitations

La modélisation des réseaux Ad-hoc et l'existence même de ces réseaux reposent sur la définition et l'application de la notion de voisinage qu'établit chaque nœud mobile. En effet, généralement les réseaux Ad-hoc sont modélisés par un ensemble de nœuds

$U = \{u_i\}$ représentant les unités mobiles. Chacun de ces nœuds définit un sous-ensemble $V(u_i) \subseteq U$ constitué de tous les nœuds v_j voisins de u_i , autrement dit, constitué de tous les nœuds de U qui reçoivent une puissance minimale seuil du signal radio de u_i . Cette puissance seuil dépassée, le lien radio entre les deux unités mobiles u_i et v_j est établi, ce qui se traduit par un arc liant les deux mobiles dans la représentation graphique du réseau Ad-hoc. Et dans le cas contraire où la puissance seuil n'est pas atteinte, la liaison radio est rompue et l'arc en question est enlevé. Ainsi, un réseau Ad-hoc est généralement modélisé par un graphe $G(N, A)$ où N désigne l'ensemble des nœuds du graphe représentant les unités mobiles et A l'ensemble des arcs du graphe représentant les liens radio établis : $A = \{(u_i, v_j) / u_i \in U, v_j \in V(u_i)\}$.

Ceci étant, un lien radio liant deux unités mobiles u_i et v_j , modélisé par un arc (u_i, v_j) dans la représentation graphique d'un réseau Ad-hoc signifie que le mobile v_j appartient à l'ensemble des voisins du mobile u_i . Cette appartenance considérée comme une relation logique (vrai ou faux) implique qu'un nœud v_j appartient au voisinage d'un autre nœud u_i ($v_j \in V(u_i)$), soit qu'il n'y appartient pas ($v_j \notin V(u_i)$). Or, cette approche à deux états (Figure 3.1.a) est loin de représenter la dégradation d'un signal radio, qui elle implique l'état d'un lien Ad-hoc.

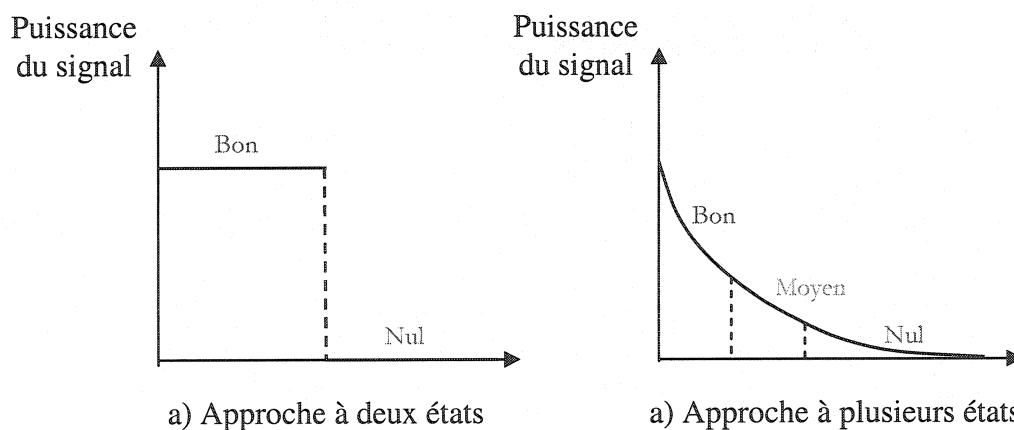


Figure 3.1 Approximation de la dégradation de la qualité du signal

En effet, dû à divers facteurs dont la mobilité, on ne perd pas un signal d'un coup, tel que laisse à croire l'approche traditionnelle de modélisation, mais le signal passe par plusieurs états allant d'un niveau bon, puis moins bon, moyen, faible, et nul, comme on peut le voir à la Figure 3.1.b.

Pour remédier à cela, il est d'usage d'attribuer des valeurs réelles aux arcs représentant les liaisons radio entre les nœuds Ad-hoc. Ces valeurs, dites poids, permettent de quantifier la qualité des liaisons radio et donnent une allure plus réaliste à la modélisation. Cette allure se traduit par la courbe continue de la Figure 3.1.b qui s'approche relativement du phénomène de la dégradation de la qualité du signal représenté à la Figure 2.2.

Cependant, le phénomène réel de dégradation des liaisons radio est encore plus complexe. En effet, tel que nous l'avons vu au chapitre II, les signaux radio subissent des phénomènes d'atténuation superposés suivant les trois niveaux de détails illustrés à la Figure 2.2. Ces phénomènes introduisent des facteurs de dégradation à allures aléatoires qui engendrent des fluctuations importantes autour des valeurs moyennes. Notons alors, et ceci est important, que ces fluctuations introduisent l'incertitude dans l'évaluation de la qualité des liaisons radio. Et ceci est définitivement impossible à représenter à l'aide de l'approche classique et des moyens qu'elle offre, vu l'énorme calcul qu'engendrerait une tentative dans ce sens. Malheureusement, cette incertitude, ou imprécision, est aussi majeure qu'elle ne peut être négligée lors de l'évaluation de la qualité des liens radio. En effet, cette dernière peut subir des dégradations importantes pouvant aller jusqu'à 30 dB (Sklar, 1997).

Par ailleurs, l'incertitude en question peut prendre d'autres aspects d'une envergure toute aussi importante. En effet, dans l'approche classique, l'existence d'un lien radio qui serait établi à un instant donné ne peut être remise en cause que dans le temps. Ainsi, un lien radio établi à un instant donné est potentiellement existant et doit être pris en compte quel que soit son poids, ce qui compromet l'aspect volatile des liens radio. En effet, à proximité des valeurs critiques de la puissance, l'existence des liens radio est encore plus incertaine. Pour pallier ce problème, une solution probabiliste peut

être envisagée pour affecter des probabilités d'existence aux liaisons radio. Or, cette approche, étant de nature probabiliste, est relativement peu adaptée pour représenter un phénomène qui n'est pas de nature purement aléatoire mais bel et bien déterministe, certes peu cerné à cause de sa complexité (Dubois et Prade, 1978).

3.1.2 Nouvelle approche proposée

La nouvelle approche que nous préconisons pour la modélisation des réseaux Ad-hoc repose sur une généralisation de la vue ensembliste sur laquelle est basée l'approche traditionnelle. En effet, au lieu de considérer, tel qu'il est le cas dans l'approche classique, qu'un terminal mobile v_j appartient à l'ensemble des voisins d'un autre mobile u_i ou qu'il n'y appartient pas, nous considérons qu'un mobile v_j appartient à l'ensemble des voisins d'un autre u_i avec un certain degré d'appartenance qui reflète la qualité du signal radio reçu de u_i . Ainsi, au lieu de considérer un réseau Ad-hoc comme un ensemble de terminaux définissant chacun un ensemble de voisins basé sur une appartenance logique, nous construisons autour de chaque unité mobile un ensemble flou de voisins en considérant une fonction d'appartenance continue. Et ceci nous permettra, comme nous le verrons plus loin, de représenter l'aspect volatile inhérent aux réseaux Ad-hoc.

Concrètement, il s'agit de modéliser un réseau Ad-hoc par un graphe flou $\tilde{G}(\tilde{N}, \tilde{A})$ orienté ou non suivant la nature des liens radio, tel qu'il a été défini au cours du deuxième chapitre. Dans ce graphe, à chaque arc (u_i, v_j) est attribué un degré $\mu(u_i, v_j)$ d'importance ou d'existence qui traduira la force de la liaison radio entre u_i et v_j . En effet, plus un lien radio est important (fort), plus il est durable et son existence est établie. De même, plus un lien radio est faible, plus il est volatile. Notons que l'on peut aussi attribuer à chaque nœud u_i un degré $\mu(u_i)$ dans le cas où les nœuds du réseau n'ont pas tous la même importance. Cette importance peut refléter, par exemple, la disponibilité de chaque nœud à exécuter certaines tâches reliées aux protocoles utilisés en tenant compte de propriétés locales telles que le débit, la gigue, etc. De plus, en imposant aux fonctions

d'appartenance de prendre des valeurs floues, nous pouvons introduire d'une manière tout à fait simple et naturelle l'imprécision et par la suite l'aspect volatile reliés aux réseaux Ad-hoc.

En conséquence, l'approche de modélisation que nous proposons permet d'une façon naturelle de tenir compte des deux aspects les plus importants et les plus caractéristiques des réseaux Ad-hoc, à savoir l'imprécision et la volatilité. Elle permet en fait de rallier les aspects de la couche physique à la modélisation de problèmes de haut niveau. Ceci s'avère indispensable surtout lorsque la couche physique n'offre pas du tout ou pas suffisamment de garanties comme la livraison de paquet par exemple. Toutefois, tenir compte des deux aspects évoqués repose fondamentalement sur la construction de fonctions d'appartenance adéquates, ce qui n'est malheureusement pas toujours une tâche facile.

3.2 Problème du *flooding*

Les protocoles d'inondation, dit de *flooding*, consistent essentiellement à envoyer un message d'un nœud source à tous les autres nœuds du réseau. Ceci se fait de proche en proche, chaque nœud retransmettant le même message vers ses voisins et ainsi de suite jusqu'à ce que tous les nœuds reçoivent le message en question. Ce type de mécanisme est très fréquemment utilisé dans les réseaux Ad-hoc à cause des changements aléatoires de topologie. En effet, le *flooding* est particulièrement utile lors d'une opération de découverte de route, de recherche de correspondant (*paging*) ou d'acheminement de message de contrôle. Malgré son importance, ce mécanisme ne bénéficie d'aucune garantie des couches basses comme la livraison des paquets par exemple. En fait, un paquet de *flooding* est ni acquitté ni retransmis ce qui le rend très vulnérable. Le *flooding*, est aussi considéré comme un mécanisme très coûteux en ressources qui doit être utilisé avec précaution. En fait, l'utiliser d'une manière peu pensée et même un nombre limité de fois peut vite conduire à la saturation.

Pour évaluer cela, nous procédons à une généralisation à l'espace de l'analyse en deux dimensions proposée par Ni et al. (1999). En effet, considérons le cas simple de la

Figure 3.2 où un mobile u transmet un message par *flooding* et que son voisin retransmet le même message. Dans ce cas, notons E_u et E_v les portions de l'espace couvertes par les transmissions de u et de v respectivement. Notons R le rayon des sphères E_u et E_v que nous supposons de même volume et ρ la distance entre u et v . Étant donné que les deux mobiles retransmettent exactement le même paquet, le volume d'espace utile que nous permet de couvrir la transmission de v se trouve alors réduite de E_v à $E_{v \setminus u}$ qui est représenté par la partie noircie de la Figure 3.2. Évaluons alors l'utilité $E_{v \setminus u}$ de la retransmission de v . On a : $E_{v \setminus u} = E_v - E_{u \cap v}$. Avec : $E_v = \frac{4}{3}\pi R^3$ et $E_{u \cap v}$ peut être donné par la formule suivante :

$$E_{u \cap v}(\rho) = 8 \cdot \int_{\rho/2}^R \int_0^{\sqrt{R^2 - y^2}} \sqrt[3]{R^3 - x^3 - y^3} \cdot dx \cdot dy$$

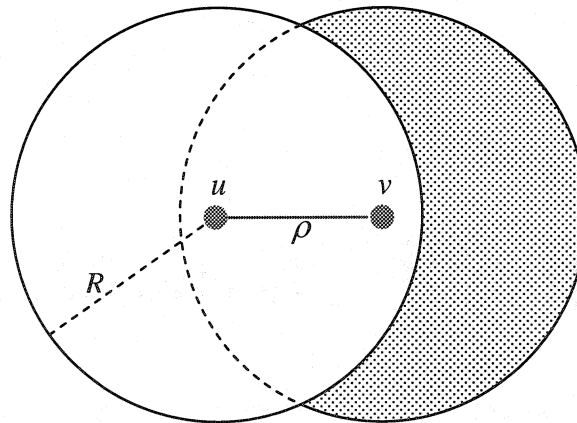


Figure 3.2 Calcul de la portion de l'espace couverte par une transmission additionnelle

Notons que lorsque $\rho = R$, la portion de l'espace hachurée atteint son maximum qui est alors égale à : $E_{v \setminus u}^{\max} \approx 0.54 \times \frac{4}{3}\pi R^3$. Ceci montre, contre toute attente, qu'une retransmission ne permet de couvrir qu'un espace utile de 0 à 54% de l'espace qu'elle

pourrait couvrir. De plus, étant donné que le mobile v peut être, à priori, d'une manière équiprobable à n'importe quelle distance ρ ($0 \leq \rho \leq R$), il est intéressant d'évaluer l'espérance mathématique de l'espace utile couvert $E(E_{v \setminus u})$ plutôt que la valeur maximale $E_{u \cap v}^{\max}$. Cette espérance est obtenue par la formule suivante :

$$E(E_{v \setminus u}) = \int_0^R p(\|u - v\| = \rho) \cdot E_{v \setminus u}(\rho) \cdot d\rho$$

avec :

$$\begin{aligned} p(\|v - u\| = \rho) &= \frac{4\pi\rho^2}{\frac{4}{3}\pi R^3} \\ &= \frac{3}{R^3} \cdot \rho^2 \end{aligned}$$

D'où :

$$\begin{aligned} E(E_{v \setminus u}) &= \frac{3}{R^3} \cdot \int_0^R \rho^2 \cdot E_{u \setminus v}(\rho) \cdot \partial\rho \\ &\approx 34 \times \frac{4}{3} \pi R^3 \end{aligned}$$

Donc, après une première transmission, une retransmission ne permet de couvrir en moyenne que 34% de ce qu'elle pourrait couvrir. Par conséquent, il est indispensable d'effectuer un très grand nombre de retransmissions afin de compléter une seule opération de *flooding*. Ainsi, utilisé de manière peu efficace et même un nombre limité de fois, le *flooding* peut vite conduire à la saturation. Ajouté à cela le nombre de retransmissions additionnelles nécessaires à cause d'un échec de livraison de message dû aux problèmes d'accès au medium, le *flooding* est particulièrement vulnérable.

Dans ce sens, plusieurs solutions ont été proposées dans la littérature afin de réduire le nombre de retransmissions nécessaires à une opération de *flooding*. Parmi ces solutions, nous retenons celle de Lim et Kim (2001) basée sur un algorithme glouton qui tente de sélectionner à chaque itération un ensemble de nœuds retransmettant, approximant un ensemble de recouvrement minimum. Malgré les bons résultats de cette

méthode, elle reste peu adaptée pour tenir compte de l'impact de la mobilité des unités sur le *flooding*, problème que nous tentons de résoudre par la suite.

3.3 Modélisation du problème

Pour modéliser le problème du *flooding* couplé à la mobilité, nous utilisons l'approche de modélisation basée sur les sous-ensembles flous que nous avons décrite auparavant. En effet, l'utilisation de cette approche pour modéliser le problème de *flooding* s'avère indispensable et se justifie par le fait que la couche physique n'offre pas de garantis de livraisons de paquet pour le *flooding*. Il est alors important de tenir compte lors de la résolution du *flooding* de l'incertitude inhérente à la couche physique. Toutefois, dû à la complexité du problème, nous nous intéressons dans le cadre de cette recherche à la seule incertitude reliée aux phénomènes de la propagation radio. Dans ces conditions, les problèmes de collision reliés spécifiquement à la couche MAC sera alors ignorés supposant ainsi l'existence d'une couche MAC presque idéale. Notant alors N l'ensemble des nœuds du réseau Ad-hoc et $A = \{(u, v) / u \in N, v \in N\}$ l'ensemble de tous les arcs ou liens radio possibles, nous considérons le réseau Ad-hoc comme un graphe flou $\tilde{G}(\tilde{N}, \tilde{A})$, où \tilde{N} est un ensemble flou sur N et \tilde{A} est une relation floue sur $N \times N$ tel que :

$$\forall (u, v) \in \tilde{N} \times \tilde{N} \quad \mu_{\tilde{A}}(u, v) \leq \min(\mu_{\tilde{N}}(u), \mu_{\tilde{N}}(v)) \quad (3.1)$$

où μ désigne la fonction d'appartenance que nous proposons de définir par la suite. Cependant, dans le contexte de notre problème, nous supposons que $\forall u \in \tilde{N} \quad \mu_{\tilde{N}}(u) = 1$. Ceci veut dire que nous considérons que tous les nœuds ont la même importance dans le réseau Ad-hoc par rapport à une opération de *flooding*. Autrement dit, tous les nœuds sont supposés capables de déclencher ou de participer à un *flooding* nécessaire à un instant donné. Dans ces conditions, on a $\tilde{N} = N$ et l'inégalité (3.1) peut être supprimée. Il ne nous reste ainsi qu'à définir $\mu_{\tilde{A}}(u, v), \quad \forall (u, v) \in \tilde{N} \times \tilde{N}$.

Nous avons vu que représenter un réseau Ad-hoc par un graphe flou revient à construire autour de chaque nœud $u \in \tilde{N}$ un ensemble flou $\tilde{V}(u)$ de voisins. Celui-ci peut être construit en imposant la condition suivante : $\forall (u, v) \in \tilde{N} \times \tilde{N}, \mu_{\tilde{V}(u)}(v) = \mu_{\tilde{A}}(u, v)$. Ainsi, chaque nœud $v \in \tilde{N}$ appartient à l'ensemble $\tilde{V}(u)$ des voisins d'un autre nœud $u \in \tilde{N}$ avec le degré d'appartenance $\mu_{\tilde{V}(u)}(v)$. Ce degré, particulièrement pour notre problème, reflète (uniquement) la qualité de la liaison radio entre u et v , autrement dit, la puissance du signal de u reçue par v . Donc, on peut écrire que :

$$\forall (u, v) \in \tilde{N} \times \tilde{N}, \quad \mu_{\tilde{V}(u)}(v) = \frac{P_u(v)}{P_u} \quad (3.2)$$

où P_u est la puissance en décibel (dB) du signal émise par u et $P_u(v)$ la puissance en dB du signal de u reçue par v .

Concrètement, considérons deux unités mobiles u et v supposées munies d'antennes omnidirectionnelles de gains G_u et G_v , et pouvant émettre sur une fréquence f avec des puissances P_u et P_v respectivement. Notons ρ la distance radiale entre u et v , et $r(t)$ le signal radio reçu de u par v . Nous avons vu au chapitre II que, dans le cas où les unités u et v sont mobiles, $r(t)$ peut être décomposé en deux composantes de nature aléatoire :

$$r(t) = m(t) \cdot r_0(t)$$

où $m(t)$ caractérise les fluctuations du signal radio dues au phénomène de la mobilité sur de longues distances pouvant aller jusqu'à quelques centaines de mètres, ou sur de longues périodes ; $r_0(t)$ représente les évanouissements du signal dus à la mobilité sur de très petites distances (aussi petites qu'une longueur d'onde 0.12 m), ou sur de petites périodes de l'ordre d'une seconde. Or dans le cadre de ce mémoire, où il s'agit de réduire le nombre de retransmissions dues au *flooding* et à la mobilité, nous nous intéressons moins aux mouvements des unités sur d'aussi petites distances. En effet, il est plus important pour notre cas d'étude de caractériser les fluctuations du signal dues aux mouvements des usagers sur des distances relativement grandes de l'ordre de 10 m plutôt

que sur de petites distances de l'ordre de 0.12 m. Ainsi, nous incombe-t-il d'utiliser des méthodes adéquates pour éliminer la composante $r_0(t)$ de tout calcul. Cette composante ne peut malheureusement pas être tout simplement négligée. En effet, rappelons-le, les mouvements sur de petites distances provoquent de très forts évanouissements pouvant aller de 30 à 40 dB (Rappaport, 2002). Pour cela, nous faisons appel à une heuristique proposée par Narasimhan et Cox (2001) pour estimer la composante $m(t)$ ainsi que la vitesse du mobile. En utilisant cette méthode, nous pouvons dorénavant écrire que : $r(t) \approx m(t)$. En passant à l'expression de la puissance, on a, selon la formule (2.3) :

$$P_u(v)_{[dB]} = 10 \cdot \log_{10} \left(G_v \cdot \left(\frac{c}{4\pi f} \right)^2 \cdot G_u P_u \right) - \alpha \cdot 10 \log(\rho) + X_\sigma$$

où α ($2 \leq \alpha \leq 6$) est un facteur qui décrit l'atténuation de la puissance avec la distance suivant la densité des obstacles séparant les unités mobiles communicantes, X_σ est une variable aléatoire gaussienne de moyenne nulle et de dispersion σ en décibel. Posons alors :

$$h(\rho, \alpha) = 10 \cdot \log_{10} \left(G_v \cdot \left(\frac{c}{4\pi f} \right)^2 \cdot G_u P_u \right) - \alpha \cdot 10 \log(\rho)$$

Donc :

$$P_u(v)_{[dB]} = h(\rho, \alpha) + X_\sigma$$

Et dans ce cas, la relation (3.2) s'écrit :

$$\mu_{\tilde{v}(u)}(v) = \frac{1}{P_u} \cdot (h(\rho, \alpha) + X_\sigma)$$

Or, la fonction d'appartenance telle que définie ci-dessus ne tient nullement compte des fluctuations qui apparaissent dans la réalité et de l'incertitude qui en résulte. En effet, la fonction $v \mapsto \mu_{\tilde{v}(u)}(v)$ prend ses valeurs dans l'ensemble des nombres réels, ce qui la rend rigide. En fait, $\mu_{\tilde{v}(u)}(v)$ peut être approximée par sa valeur moyenne $h(\rho, \alpha)/P_u$, ce qui néglige les fluctuations autour de cette moyenne. Pour remédier à cela, nous définissons

$\tilde{P}_u(v)$, la puissance floue du signal de u reçue par v par sa moyenne $\tilde{P}_u(v) = h(\rho, \alpha)$ et sa fonction d'appartenance :

$$\mu_{\tilde{P}}(x) = e^{-\frac{(x-h(\rho, \alpha))^2}{2\sigma^2}}$$

Ceci revient à construire autour de la puissance moyenne de valeur réelle $h(\rho, \alpha)$, un ensemble flou $\tilde{P}_u(v)$ de fonction d'appartenance $\mu_{\tilde{P}}$ illustrée à la Figure 3.3.

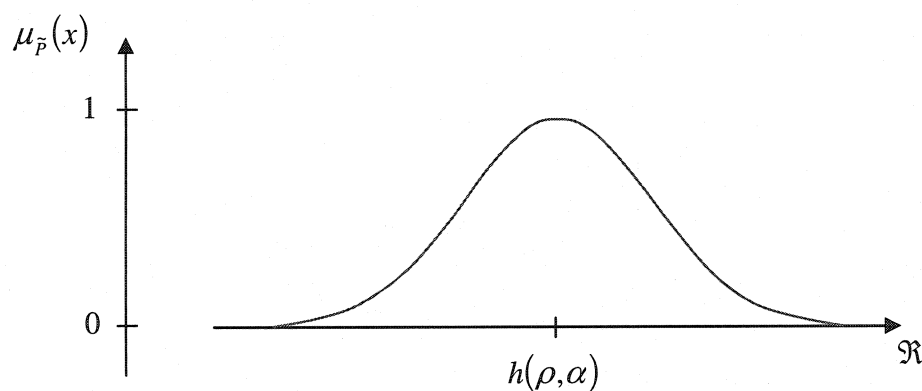


Figure 3.3 Définition de la puissance floue \tilde{P}

Le degré d'appartenance à cet ensemble de chaque valeur $h(\rho, \alpha) + X_\sigma$ de la puissance reflète la probabilité d'occurrence de cette valeur selon la fonction de densité de probabilité qui caractérise σ . Et finalement, on a :

$$\mu_{\tilde{P}(u)}(v) = \frac{\tilde{P}_u(v)}{P_u} \quad (3.3)$$

Or, la fonction d'appartenance définie ci-haut fait appel aux paramètres inconnus ρ , α et σ , soient la distance radiale entre les deux unités mobiles communicantes u et v , et le facteur d'atténuation de la puissance suivant la densité des obstacles environnants et la dispersion de la variable aléatoire. Ces deux derniers dépendant de la fréquence d'opération, de la hauteur des antennes, de la distance ρ ainsi que de l'environnement de propagation doivent être estimés via des mesures sur le terrain. À ce propos, nous pouvons trouver dans la littérature différentes valeurs pour ces facteurs suivant différents

scénarios d'environnement, en l'occurrence celles proposées par Bach Anderson (1997). Toutefois, notons que pour les radio-systèmes de communications personnels, notamment Ad-hoc, les travaux d'estimations de ces facteurs se sont orientés vers des environnements intérieurs limités (Anderson et al. 1995).

Alors que les deux facteurs α et σ sont fixes pour un environnement donné, la distance radiale ρ change tout le temps et doit être calculée chaque fois qu'il est question qu'un mobile u calcule le degré d'appartenance $\mu_{\bar{v}(u)}(v)$ d'un nœud voisin v . À cet égard, rappelons que grâce à une stratégie de rafraîchissement de l'information de voisinage, tous les nœuds Ad-hoc sont amenés à transmettre à chaque intervalle de durée maximale de `HELLO_INTERVAL` = 1 sec un message *hello* à leur voisinage afin de notifier leur existence (Perkins et al., 2002). Ainsi, à la réception d'un message de son voisin v , le mobile u utilisera la puissance reçue pour estimer la distance qui les séparent. En fait, l'utilisation de la méthode d'estimation de la composante $m(t)$ du signal donnée par Narasimhan et Cox (2001) nous permet d'obtenir la valeur moyenne de la puissance. À partir de cette valeur, et en utilisant la formule 2.2, nous déduisons la distance radiale ρ comme suit :

$$\rho = \left(G_u \cdot \left(\frac{c}{4\pi f} \right)^2 \cdot \frac{G_v P_v}{P_v(u)} \right)^{1/\alpha}$$

Cette distance qui datera au pire des cas de `HELLO_INTERVAL` * (1+`ALLOWED_HELLO_LOSS`) sec, où `ALLOWED_HELLO_LOSS` représente le nombre de fois qu'un message *hello* peut être perdu, peut être soit supposée constante sur cet intervalle de temps dans le cas où les unités mobiles sont lentes, soit approchée par $\rho \pm v \cdot \Delta t$, où v est la vitesse du mobile obtenue grâce à la méthode de Narasimhan et Cox (2001).

3.4 Analyse du problème

Comme nous l'avons souligné dans la première section de ce chapitre, la notion de voisinage est fondamentale dans la modélisation des réseaux Ad-hoc. Cette notion est encore plus importante dans la résolution de notre problème consistant à optimiser une

opération de *flooding* en tenant compte de la mobilité des usagers. Nous verrons que l'exploitation des informations sur le voisinage nous permettra de réduire significativement le nombre de retransmissions nécessaire à une opération de *flooding*. En effet, considérons le réseau Ad-hoc illustré à la Figure 3.4. Ce réseau est constitué de 31 nœuds que nous supposons fixes et établissant chacun avec ses voisins immédiats un certain nombre de liens radio (statiques) représentés par les arêtes entre les nœuds. Supposons aussi que le nœud central déclenche un *flooding*.

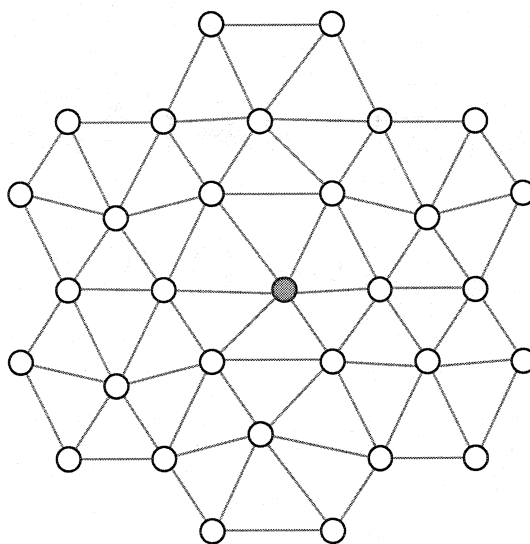


Figure 3.4 Exemple de réseau Ad-hoc

Alors, dans le cas où les nœuds du réseau ne considèrent aucune information sur leur voisinage, chaque nœud qui reçoit pour la première fois un paquet de *flooding* devra le retransmettre à son tour. Ainsi, 31 retransmissions seront nécessaires afin de compléter une telle opération, comme il est illustré à la Figure 3.5.

Par contre, si chaque nœud connaît son voisinage immédiat, alors un nœud ne pourra retransmettre le message du *flooding* s'il le reçoit de tous ses nœuds voisins. En conséquence, un nœud ne pourra retransmettre un message objet de *flooding* que dans le cas où il ne le reçoit pas d'au moins un de ses nœuds voisins. Et dans ce cas, il considérera que ce voisin (ou ces voisins) n'a pas encore reçu le paquet de *flooding*.

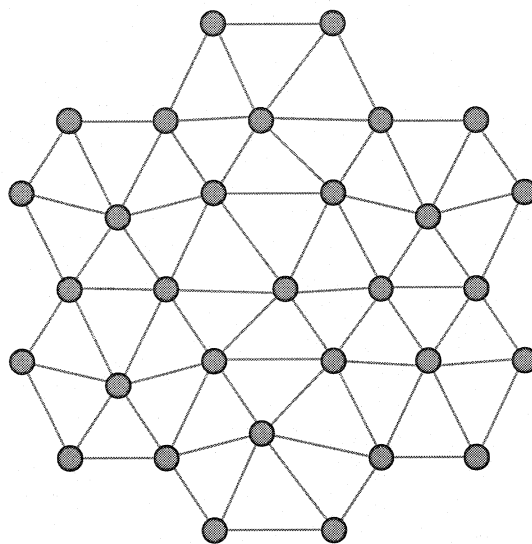


Figure 3.5 Exemple de *flooding* sans tenir compte de l'information de voisinage

Dans ces conditions, seulement 22 transmissions seront nécessaires pour compléter le *flooding*, comme on peut le voir à la Figure 3.6. Ainsi, la connaissance du voisinage immédiat nous permet de réduire le nombre de retransmissions d'à peu près 30% .

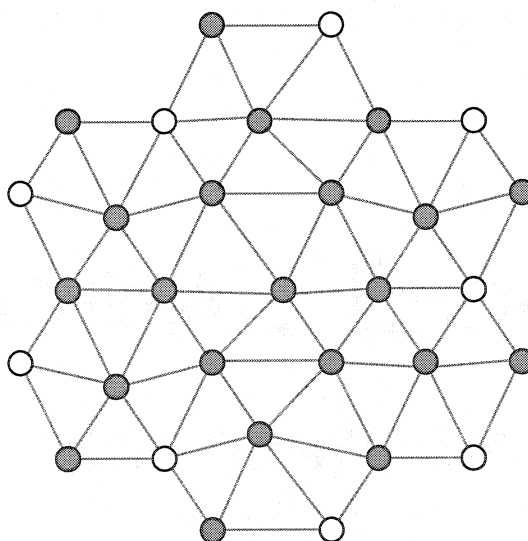


Figure 3.6 Exemple de *flooding* en tenant compte du voisinage immédiat

Toutefois, si l'on considère que chaque nœud dispose de l'information sur le voisinage à deux sauts, c'est-à-dire que chaque nœud connaît son propre voisinage ainsi que le voisinage de chaque nœud de son voisinage, alors chaque nœud retransmettant pourra sélectionner parmi les nœuds de son voisinage immédiat un ensemble minimum afin de couvrir tous les voisins de ses voisins. Et dans ce cas de figure, seulement 13 retransmissions seront nécessaires pour compléter le *flooding*, tel qu'illustré à la Figure 3.7. Ainsi, l'exploitation de l'information de voisinage à deux sauts nous permet de réduire le nombre de retransmissions d'à peu près 58%. Par conséquent, l'on remarque que le *flooding* est amélioré au fur et à mesure que l'on dispose davantage d'informations sur la topologie. L'optimum est alors atteint lorsque chaque nœud prend connaissance de la topologie complète du réseau. Toutefois, notons que la disponibilité de telles informations, surtout sur un grand nombre de sauts, est très coûteux. En effet, cela nécessite des retransmissions additionnelles pour l'échange et la mise à jour de celles-ci. Pour cette raison, nous limitons la connaissance de la topologie à deux sauts. Cette hypothèse a été déjà testée par Lim et Kim (2001) et a donné de bons résultats.

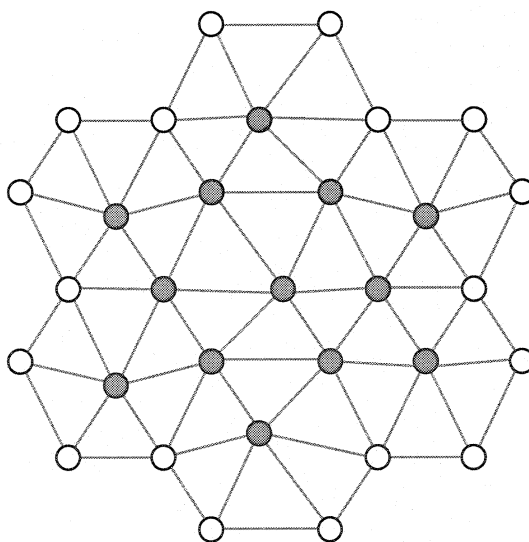


Figure 3.7 Exemple de *flooding* en tenant compte du voisinage à deux sauts

Ceci dit, nous considérons dorénavant que chaque nœud mobile dispose de l'information de voisinage jusqu'à deux sauts. Ainsi, chaque nœud participant au *flooding* devra sélectionner parmi l'ensemble de ses propres voisins un ensemble de nœuds retransmettant afin de couvrir son voisinage à deux sauts. De ce fait, et comme le *flooding* est une opération itérative, nous ne traitons à chaque itération qu'un réseau partiel constitué d'un voisinage à deux sauts autour du nœud prenant part à cette opération.

Supposons maintenant qu'à une itération donnée, un nœud u reçoit un paquet de *flooding* d'un autre nœud m et que u a été sélectionné pour effectuer une retransmission. Comme nous l'avons expliqué précédemment, dans le cadre d'une stratégie à deux sauts, le nœud u doit sélectionner, parmi tous les nœuds de son voisinage immédiat, un certain ensemble de nœuds retransmettant afin de couvrir l'ensemble $U = V(V(u))$ des voisins de ses voisins. Cependant, notons qu'il n'est pas nécessaire de couvrir tous les nœuds de $V(V(u))$. En effet, plusieurs de ces nœuds ont déjà reçu le paquet de *flooding* par d'autres nœuds, en l'occurrence les nœuds appartenant à $V(m) \subset V(V(u))$ voisins de m ont été couverts par la transmission de ce dernier. Mieux encore, l'ensemble U peut être restreint à $U = V(V(u)) - V(u) - V(m) - V(V(u) \cap V(m))$ comme il est montré dans (Lou et Wu, 2002) et illustré à la Figure 3.8.

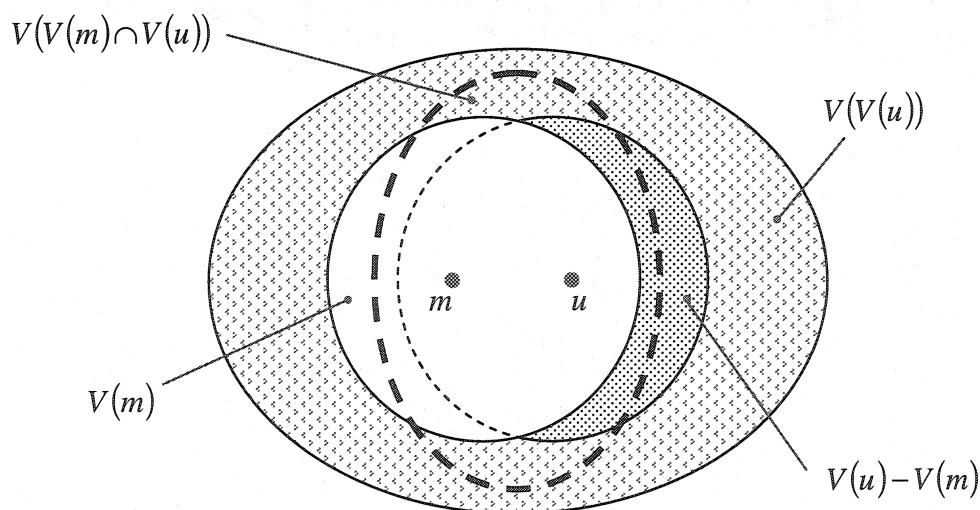


Figure 3.8 Sélection des voisins à deux sauts à couvrir

Tel que modélisé précédemment, le problème à résoudre s'exprime alors de la manière suivante: Choisir parmi les nœuds de l'ensemble flou $\tilde{V}(u)$ des voisins de u , un ensemble retransmettant minimum afin de couvrir tous les nœuds de U en considérant autant que possible les meilleurs liens radio.

Ce problème est un problème NP-difficile. En effet, en posant $\mu_{\tilde{V}(u)}(x) = 1, \forall x \in \tilde{V}(u)$, notre problème se transforme en un problème de recouvrement minimum qui est NP-difficile (Lim et Kim, 2001). Par conséquent, nous favorisons la recherche de bonnes solutions plutôt que des solutions optimales.

3.5 Stratégie et algorithme de résolution

Pour le résoudre, nous scindons notre problème en deux sous-problèmes selon ses deux principaux aspects, à savoir la mobilité et le *flooding*. Le premier sous-problème consiste alors à réduire le nombre de retransmissions échouées dues à la mobilité des usagers, aux changements de topologies et de qualité des liens radio qui en résultent. Pour cela, nous tentons de sélectionner un ensemble de nœuds dits critiques pour lesquels il est primordial d'utiliser les meilleurs liens radio afin de leur livrer le paquet de *flooding*. En effet, un nœud $w \in U$ est dit critique, si le meilleur *chemin flou possible* $\tilde{C}(u, v, w)$ (i.e., v vérifies : $\min(\mu_{\tilde{V}(u)}(v), \mu_{\tilde{V}(v)}(w)) = \max_{v \in \tilde{V}(u)} \{ \min(\mu_{\tilde{V}(u)}(v), \mu_{\tilde{V}(v)}(w)) \}$) pour acheminer un paquet de *flooding* de u à w contient un lien radio faible. Quant à un chemin flou $\tilde{C}(u, v, w)$, il est dit possible s'il appartient à l'ensemble des chemins possibles pour acheminer un paquet de *flooding* dans une stratégie à deux sauts, c'est-à-dire si $v \in \tilde{V}(u)$ ($\mu_{\tilde{V}(u)}(v) \neq 0$), tel qu'illustré à la Figure 3.9. Ainsi construit, l'ensemble des nœuds critiques et la définition même de ces nœuds justifient l'utilisation des meilleurs liens radio disponibles afin de leur acheminer les paquets de *flooding*. Les nœuds critiques sont en fait des nœuds pour lesquels on n'est pas sûr que le paquet de *flooding* parvienne même en utilisant les meilleurs chemins possibles. En conséquence, nous tenterons de diversifier les tentatives d'acheminements des paquets vers ces nœuds. Pour ce faire,

nous changeons la définition de l'ensemble U des nœuds à couvrir. En fait, nous posons : $U = U \cup S$ où S est l'ensemble des nœuds critiques de m inclu dans $V(V(u))$.

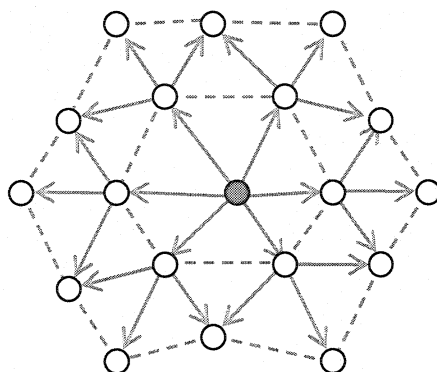


Figure 3.9 Ensemble des chemins possibles pour acheminer un paquet de *flooding* dans une stratégie à deux sauts

Or, un lien radio (u, v) est dit faible si la puissance minimale que peut recevoir le mobile v est supérieure au seuil S_γ illustré à la Figure 3.10. Ce seuil défini par la donnée de la quantité γ détermine en fait la tolérance de notre algorithme aux pertes de messages de *flooding* et peut être ajusté suivant les contextes d'utilisation.

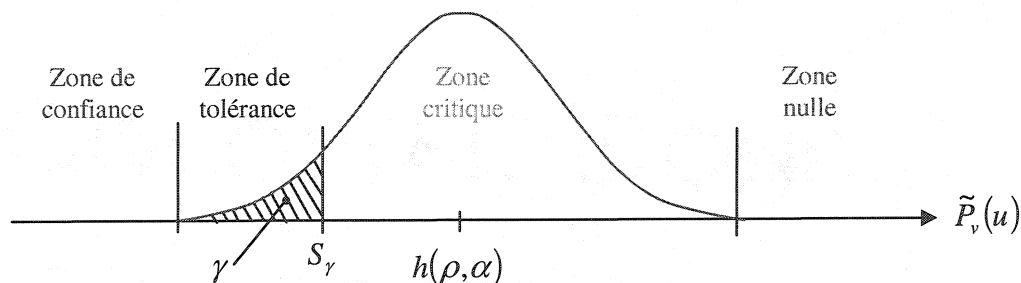


Figure 3.10 Définition du seuil de tolérance S_γ

Ce premier sous-problème peut être résolu en utilisant un algorithme de plus court chemin flou, notamment celui présenté par Boulmakoul (2002). Quant au deuxième, il

consiste à réduire le nombre de retransmissions nécessaires pour couvrir les nœuds non critiques. Pour ces nœuds, nous nous permettons d'utiliser des liens radio moins fort, mais non critiques, quitte à minimiser le nombre de retransmissions nécessaires. Pour cela, nous tentons d'approcher un ensemble de recouvrement minimum (*Minimum Connected Dominating Set*). Ceci peut être fait en utilisant, par exemple, l'algorithme présenté par Guha (1998) sur lequel est basé l'algorithme PDP de *flooding* (Lou et Wu, 2002) que nous utiliserons pour les études de performances au chapitre suivant.

Notons alors $\tilde{G}(\tilde{N}, \tilde{A})$ le graphe flou orienté représentant le voisinage à deux sauts autour du nœud u , construit de la manière suivante :

$$\tilde{N} = V(u) \cup U \text{ avec } \mu_{\tilde{N}}(x) = 1; \forall x \in V(u) \cup U \text{ et}$$

$$\tilde{A} = \{(x, y) \in \tilde{N} \times \tilde{N} / \mu_{\tilde{A}}(x, y) = \mu_{\tilde{V}(x)}(y)\}.$$

Considérons la restriction de \tilde{G} sur l'ensemble des chemins possibles et notons la dorénavant $\tilde{G}(\tilde{N}, \tilde{A})$. Pour ce graphe, nous définissons la fonction \tilde{p} poids des arcs de la manière suivante : $\tilde{p}(x, y) = \mu_{\tilde{V}(x)}(y)$. Dans ces conditions, l'algorithme que nous proposons est décrit à la Figure 3.11.

3.6 Mise en œuvre de l'algorithme

Considérons l'ensemble N des mobiles présenté à la Figure 3.11 évoluant dans l'espace libre. Dans ces conditions, selon la formule (2.1), la relation (3.3) s'écrit :

$$\mu_{\tilde{V}(u)}(v) = \frac{\tilde{P}_u(v)}{P_u}$$

avec :

$$\tilde{P}_u(v)_{[dB]} = h(\rho, 2) \quad (\in \mathfrak{R})$$

Considérons aussi les liens radio établis entre les unités Ad-hoc représentés par des arêtes à la Figure 3.12.

INITIALISATION :

$$F = \emptyset ;$$

$$(\forall v \in V(u))(\forall w \in U) \quad \tilde{p}(v, w) = \max(0, \tilde{p}(u, v) - \tilde{p}(v, w))$$

PHASE I :

- Calculer les PLCF (plus long chemin flous) partant de u vers les nœuds de U .

Pour chaque chemin possible (u, v, w) critique, faire :

$$F = F \cup \{v\} \text{ et } U = U - \{w\}$$

- Considérer la restriction du graphe \tilde{G} sur l'ensemble des nœuds : $F \cup U \cup \{u\}$

Calculer les PLCF allant de u vers les nœuds de Z

Pour chaque chemin possible (u, v, w) non critique, faire :

$$U = U - \{w\}$$

PHASE II :

- 1 Considérer la restriction du graphe \tilde{G} sur l'ensemble des nœuds : $U \cup V(u) - F$

- 2 Appliquer un algorithme pour déterminer un ensemble G de recouvrement minimum

- 3 Pour chaque chemin possible (u, v, w) critique avec $v \in G$, faire :

$$\mu_{\tilde{A}}(v, w) = 0$$

- 4 Retourner à 2 si des chemins critiques sont détectés

RESULTAT :

Retourner $F \cup G$.

Figure 3.11 Algorithme de *flooding* proposé

À chaque arête (u,v) on associe un poids $\mu_{\bar{v}(u)}(v)$ qui caractérise la qualité du lien radio. Par souci de simplification du calcul, nous multiplions ce poids par le facteur P qui représente la puissance d'émission des unités mobiles. Supposons finalement que $S_\gamma = 20$ et que le mobile u a participé à une opération de *flooding* et qu'il sélectionne le nœud v pour effectuer une retransmission.

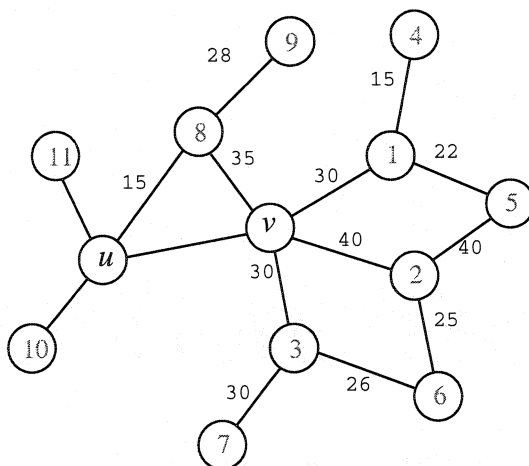
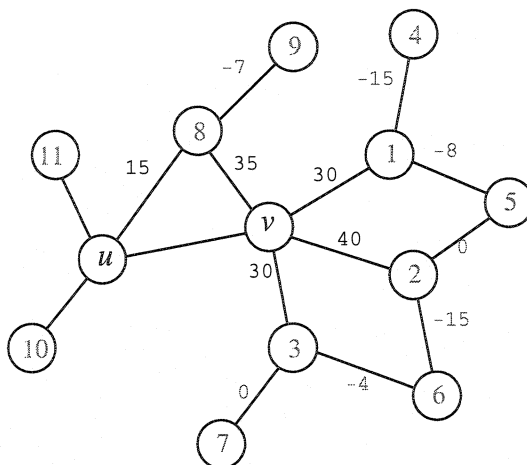


Figure 3.12 Exemple de réseau Ad-hoc

Initialisation :

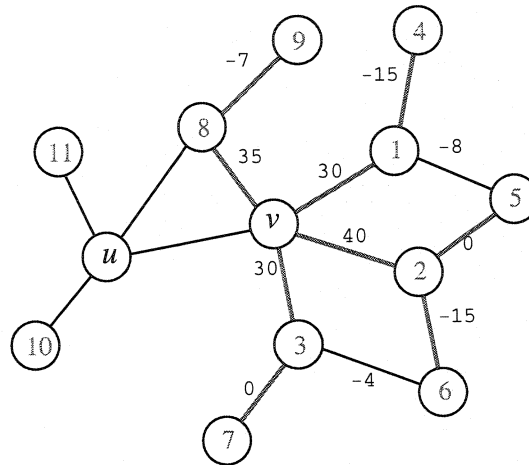
$$U = \{4,5,6,7,9\}$$

$$F = \emptyset$$



Phase I :

- Plus longs chemins :

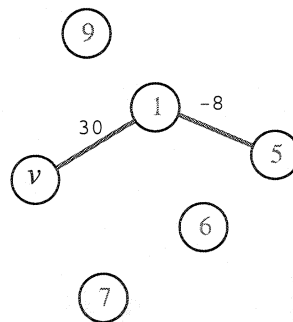


$L(v, 1, 4) = 15$
$L(v, 2, 6) = 25$
$L(v, 8, 9) = 28$
$L(v, 3, 7) = 30$
$L(v, 2, 5) = 40$

La fonction L représente la longueur du chemin. Dans ces conditions, un seul chemin est critique : $(v, 1, 4)$. En effet : $L(v,1,4) < S_\gamma$. D'où :

$$F = F \cup \{1\} = \{1\}; U = U \setminus \{4\} = \{5,6,7\}$$

- Considérer la restriction du graphe sur l'ensemble des nœuds $F \cup U \cup \{v\}$ et calculer les PLCF :



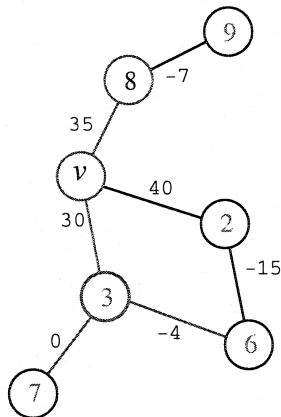
$L(v, 1, 5) = 22$

Un seul chemin est détecté et il est non critique; $L(v,1,5) \geq S_\gamma$.

$$U = U \setminus \{5\} = \{6,7\}$$

Phase II :

- Considérer la restriction du graphe sur l'ensemble des nœuds $U \cup V(v) \setminus F$ et trouver un ensemble de recouvrement minimum en utilisant l'algorithme de Guha (1998).

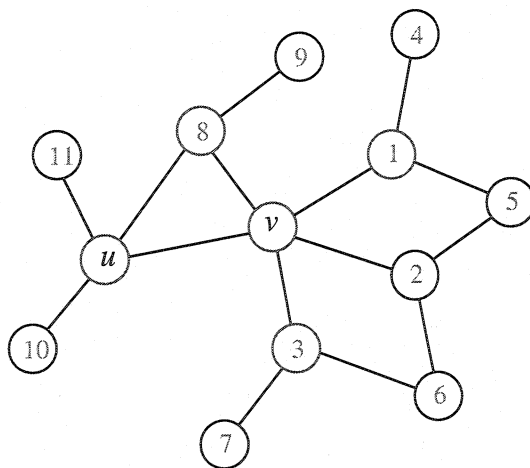


$L(v, 3, 6) = 26$ $L(v, 8, 9) = 28$ $L(v, 3, 7) = 30$

Aucun chemin n'est critique, et $G = \{3,8\}$

Résultat :

Retourner $F \cup G$; ($= \{1,3,8\}$)



CHAPITRE 4

IMPLÉMENTATION ET RÉSULTATS

Dans le précédent chapitre, nous avons traité du problème de la modélisation des réseaux mobiles Ad-hoc dans sa forme la plus générale. Nous avons également abordé le problème du *flooding* et avons proposé un algorithme basé sur le calcul des plus courts chemins et la construction d'ensembles de recouvrement minimum pour le résoudre. Dans le présent chapitre, nous nous proposons d'implémenter cet algorithme et de conduire une série de tests et d'expériences afin d'étudier ses performances en le comparant à l'un des meilleurs algorithmes de *flooding* existant dans la littérature, à savoir le PDP (Lou et Wu, 2002). Ainsi, nous décrivons d'abord l'environnement de l'expérience et l'implémentation. Puis, nous nous attardons sur la nature des tests menés et donnons un plan d'expériences. Finalement, nous présentons les résultats obtenus et les analysons.

4.1 Détails d'implémentation

Dans cette section, nous nous intéressons aux choix d'implémentation faits afin de mener à terme notre travail et conduire les séries de tests désirées. Nous décrivons, dans un premier temps, les structures de données adoptées avant d'aborder les détails techniques de l'algorithme lui-même. Nous étudions ainsi la manière de construire le critère γ et un autre critère d'arrêt que nous justifierons. Ensuite, nous précisons le choix d'environnement matériel et logiciel utilisé.

4.1.2 Implémentation et structures de données

Dans le but de réaliser les tests désirés, nous avons dû implémenter une série de classes inscrites dans un schéma hiérarchique allant du scénario de la simulation, au

réseau Ad-hoc, au mobile Ad-hoc, puis la queue de transmission et l'algorithme de *flooding*. Dans ce qui suit, nous donnons une description de ces différentes classes ainsi que l'interaction entre elles.

- **La classe *SimulationScenario***

Comme nous l'avons avancé plus haut, nous avons choisi de conduire plusieurs séries de simulations suivant le plan d'expériences décrit à la section suivante. Chaque série est en fait constituée de 500 simulations de même nature, à savoir utilisant les mêmes valeurs pour les variables de décisions pertinentes. Le but est de pouvoir calculer une moyenne significative pour chaque variable de décision à la fin de chaque série de simulations. Nous avons implémenté alors la classe *SimulationScenario* dont l'interface est donnée à la Figure 4.1 et qui représente une série de simulation appelée scénario. Ce dernier est défini par la donnée du nombre de simulations à conduire ainsi que le nom du fichier de statistiques à produire. Le pointeur *Network* identifie le réseau Ad-hoc créé et utilisé pour la simulation conduite à un instant donné. La méthode principale dans cette classe est *Simulate* qui permet de conduire *NbSimulation* itérations et de créer un réseau Ad-hoc pour chaque simulation.

```
class SimulationScenario
{
    public:
        SimulationScenario(void);
        ~ SimulationScenario(void);
        void Simulate(CView *View);
        void Display(CDC *pDC);
        void CancelSimulation();

    private :
        char StatFileName[100];
        int NbSimulation;
        Adhoc *Network;
        bool Cancel;

};
```

Figure 4.1 Interface de la classe *SimulationScenario* représentant une série de simulations

- La classe *Adhoc*

La classe *Ad hoc* dont l'interface est fournie à la Figure 4.2 représente un réseau mobile Ad-hoc. Ce dernier est défini par la donnée de la collection *Network* constituée de *NB_MOBILE* unités mobiles. Il définit aussi le cadre général de la simulation en fournissant certains paramètres relatifs au déroulement de la simulation, comme l'illustre la Figure 4.3.

```
class Adhoc
{
public:
    Adhoc(char StatFileName[]);
    ~Adhoc(void);
    bool Simulate(CView *View);
    void SaveStatistics();
    void Display(CDC *pDC);
    void DisplayLink(CDC *pDC);
    void DisplayLowLink(CDC *pDC);
    void DisplayNoLink(CDC *pDC);
    void FreeMemory(void);
    int DoEvents();
    void Cancel();
    void SaveNet();
    void LoadNet(CMobile Network[]);

private :
    Mobile Network[NB_MOBILE];
    char StatFile[100];
};
```

Figure 4.2 Interface de la classe *Adhoc* représentant un réseau mobile Ad-hoc

Le paramètre *NB_MOBILE* dans la Figure 4.3 définit le nombre d'unités mobiles constituant le réseau Ad-hoc évoluant dans un espace de dimensions $X_{MAX}-X_{MIN} * Y_{MAX}-Y_{MIN}$ m. La constante *HELLO_INTERVAL* en msec définit l'intervalle de temps maximum que peut attendre une unité mobile Ad-hoc avant d'envoyer un message *Hello*. Quant à *ALLOWED_HELLO_LOSS*, elle spécifie la tolérance aux pertes des

messages *HELLO*. En effet, une unité mobile qui ne reçoit pas un message *HELLO* d'une de ses unités voisines après l'écoulement d'un intervalle de temps évalué à : $HELLO_INTERVAL(1+ALLOWED_HELLO_LOSS)$ décidera que l'unité mobile en question ne fait plus partie de son voisinage.

```
#define NB_MOBILES 70

#define X_MIN 0.0
#define Y_MIN 0.0
#define X_MAX 1000.0
#define Y_MAX 1000.0

#define HELLO_INTERVAL 1000
#define ALLOWED_HELLO_LOSS 2
#define HELLO_JITTER 0.5

#define TEST_START      5000
#define SIMULATION_TIME (TEST_START + 1000)
```

Figure 4.3 Définitions des paramètres de la simulation pour un réseau Ad-hoc

Le paramètre *HELLO_JITTER* est relatif au mécanisme de traitement des collisions. En effet, une unité Ad-hoc qui essaie de transmettre un message *HELLO* et qui se rend compte que le medium radio est occupé se retiendra d'envoyer son paquet pendant une durée de temps aléatoire dont la valeur maximale est de $HELLO_JITTER*HELLO_INTERVAL/100$. Notons que les valeurs utilisées pour les paramètres relatifs aux messages *HELLO* sont celles recommandées pour le protocole AODV (Perkins et al., 2000). Finalement, les constantes *SIMULATION_DURAITION* et *SIMULATION_START* spécifient respectivement la durée de la simulation (soit celle de vie du réseau Ad-hoc) et l'instant à partir duquel on essaie de déclencher une opération de *flooding* pour récolter les statistiques inhérentes.

La méthode principale de la classe *Ad hoc* est *Simulate*. Celle-ci s'occupe principalement de la gestion du médium de transmission partagé, du décompte du temps

de simulation, du déclenchement du *flooding*, ainsi que de la récolte des statistiques. En fait, dans une première phase de durée *SIMULATION_START*, cette méthode laisse le temps aux unités mobiles d'envoyer plusieurs messages *HELLO*, d'écouter ceux de leurs voisins afin de construire des tables de voisinage, ainsi que de briser et/ou établir des liens de communications point à point. Une fois cette phase terminée, cette méthode testera la connexité du réseau. Si celle-ci est établie, dans le sens où aucune unité Ad-hoc ne se trouve isolée, une unité Ad-hoc sera aléatoirement désignée pour déclencher un *flooding*. Par contre, si la connexité n'est pas établie avant *SIMULATION_DURATION* msec, aucun *flooding* n'est effectué. Dans ce cas, la simulation est considérée en échec, ce qui nécessite de la reprendre à zéro. Notons également que la méthode *Simulate* établit un mécanisme d'exclusion mutuelle pour l'accès au medium radio partagé. Ce mécanisme assure qu'une seule unité mobile dans un voisinage donné peut transmettre un paquet. Toutes les autres sont alors obligées de retarder leurs retransmissions jusqu'à ce que le medium soit libre. Ce mécanisme permet de résoudre les problèmes d'accès au medium évoqués au chapitre 2 et fournit une plate-forme presque idéale pour la couche MAC. En fait, une couche MAC idéale serait celle où tous les mobiles le désirant peuvent transmettre en même temps.

- **La classe *Mobile***

Bien entendu, un réseau Ad-hoc est formé d'un ensemble d'unités mobiles Ad-hoc. Une unité mobile est alors représentée par un objet de la classe *Mobile* dont l'interface est illustrée en deux temps aux figures 4.4 et 4.5. Cette classe introduit aussi certains paramètres relatifs au fonctionnement du mobile donnés à la Figure 4.6.

Le paramètre *MAX_QUEUE_SIZE* définit la taille des queues de transmission et de réception des mobiles ainsi que la taille des tables de voisinage que les mobiles entretiennent. Les valeurs données pour *V_MIN* et *V_MAX* en Km/h définissent l'intervalle de vitesse que prend aléatoirement une unité Ad-hoc. Quant à *DV_MAX* en Km/h et *DDir_MAX* en degrés, ils spécifient respectivement les valeurs absolues de

l'accélération ainsi que le changement de direction que le mobile peut effectuer aléatoirement pendant la simulation.

```

class Mobile
{
  private:

    static int PacketId;

    int Id;
    float V;
    int Dir;
    float X, Y;
    unsigned int StartTime;
    float Power;
    float MinRxPower;
    int **Neighborhood;
    unsigned int *NRefreshTime;

    CQueue<int> RxPacket[4];
    CQueue<int> TxPacket[3];
    CQueue<int> TxHistory;
    CQueue<int> RxHistory;

    int StatNbRx, StatNbTx, StatNbTxFail;

    void RefreshSpeed();
}

```

Figure 4.4 Méthode et attributs privés de la classe *Mobile*

La Figure 4.4 fournit un aperçu des attributs et méthodes membres privés de la classe *Mobile*. Celle-ci est constituée, entre autres, des attributs *Id* et *PaquetId* qui identifient d'une manière unique le mobile et chaque paquet qu'il envoie respectivement dans le réseau Ad-hoc. Les attributs *V*, *Dir*, *X* et *Y* représentent respectivement la vitesse du mobile qui doit être inscrite dans l'intervalle $[V_MIN, V_MAX]$, la direction que le mobile prend à un instant donné et les coordonnées de l'unité Ad-hoc inscrites respectivement dans $[X_MIN, X_MAX]$ et $[Y_MIN, Y_MAX]$. *StartTime* identifie l'instant

où le mobile rejoint le réseau et qui régule par conséquent les dates de transmissions des messages *HELLO*.

Dans toutes les séries de simulation, nous avons imposé aux mobiles de rejoindre le réseau dans les deux premières minutes après le début de la simulation. Les paramètres *Power* et *MinRxPower* définissent respectivement la puissance d'émission du mobile ainsi que la puissance minimale que le mobile peut recevoir. Le pointeur *Neighborhood* identifie la table de voisinage que le mobile entretient, formant ainsi une architecture partielle du réseau couvrant la topologie à deux sauts à partir du mobile en question. Le tableau *NRefreshTime* permet d'associer chaque voisin du mobile à une date de péremption de l'information de voisinage reçue de ce voisin. Cette date est obtenue en ajoutant la valeur *HELLO_INTERVAL(1+ALLOWED_HELLO_LOSS)* à l'instant où le mobile reçoit un message *HELLO* de son voisin.

Les tableaux *RxPacket* et *TxPacket* représentent respectivement les queues de réception et de transmission utilisées lors de la communication avec d'autres mobiles. Lors de la réception, un paquet est constitué de son identité *PacketId*, de l'identité de l'expéditeur et de l'objet du message. À ces paramètres, le mobile ajoute la date de réception. L'objet du message peut être soit un message *HELLO*, soit un paquet de *flooding* demandant au récepteur de retransmettre le paquet, ou encore un paquet de *flooding* ne demandant pas de retransmission. Par ailleurs, un message transmis consiste en l'identité du paquet, la liste des récepteurs et l'objet du paquet. Les queues *TxHistory* et *RxHistory* gardent un historique des messages reçus ou transmis. Les attributs *StatNbTx*, *StatNbRx* et *StatNbTxFail* permettent respectivement de garder des statistiques du nombre de paquets de *flooding* transmis, ceux reçus et le nombre de transmissions qui ont échouées.

La Figure 4.5 fournit la portion publique de la classe *Mobile* dont nous décrivons les principales méthodes et attributs. La variable statique *Time* est une variable partagée par tous les mobiles et représente ainsi le temps absolu. Les méthodes *GetN1* et *GetN2* permettent d'obtenir respectivement les voisinages à un saut et à deux sauts.

```

class Mobile
{
    public:

        static CRandom RGen;
        static unsigned long Time;

        CMobile(void);
        ~CMobile(void);

        CQueue<int> GetN1();
        CQueue<int> GetN2();
        CQueue<float> GetMembership(CQueue<int> &X);

        void SetXY(float X, float Y);
        void SetId(int Id);
        void SetStartTime(int Time);
        void RefreshPosition();
        void ResetNeighborhood();
        void RefreshNeighborhood(CMobile &Mobile);
        void RemoveNeighbor(int MobileId);
        void ReceivePacket(int PacketId, int SenderId,
                           int Time, int PacketObject);
        void UpdateNeighborhood();
        void StartFlooding(int RxPacketId, int SenderId);
        void GetStatistics(int &NbRx, int &NbTx,
                           int &NbTxFail);
        void StartFlooding();
        void Display(CDC *pDC);
        void FreeMemory();

        int GetId();
        float GetX();
        float GetY();
        float GetPower();
        float GetMinRxPower();
        float ReceivedPower(CMobile &Mobile);

        bool IsActive();
        bool SayHello(CMobile Network[], bool MediumBusy);
        bool TreatReceivedPackets(CMobile Network[]);
        bool Broadcast(CMobile Network[]);
};

```

Figure 4.5 Méthodes et attributs publics de la classe *Mobile*

La méthode *GetMembership* permet d'obtenir le degré d'appartenance d'un mobile à l'ensemble des voisins d'un autre. Ce degré d'appartenance est utilisé uniquement par l'algorithme que nous avons proposé, basé sur la modélisation en ensembles flous. La méthode *RefreshPosition* permet de mettre à jour les coordonnées de l'unité mobile dans le temps, étant donné sa vitesse, son accélération et sa direction.

```
#define MAX_QUEUE_SIZE NB_MOBILES

#define V_MIN 60.0
#define V_MAX 80.0
#define DV_MAX 10.0
#define DDir_MAX 15
```

Figure 4.6 Définitions des paramètres de fonctionnement d'un mobile Ad-hoc

Les méthodes *ResetNeighborhood*, *RefreshNeighborhood*, *RemoveNeighbor* et *UpdateNeighborhood* permettent de construire et de mettre à jour la table de voisinage, étant données les contraintes temporelles auxquelles sont soumises les informations contenues dans cette table. En ce qui concerne le *flooding*, il est effectué grâce aux méthodes *StartFlooding*. La première, ne prenant aucun argument, permet de déclencher une opération de *flooding*. Elle concerne donc le nœud source de l'opération. Par contre, l'autre méthode *StartFlooding*, prenant comme argument l'identité du paquet de *flooding* ainsi que celle de l'expéditeur, est utilisée lorsqu'un mobile reçoit l'ordre de retransmettre le paquet qu'il reçoit. Ces deux méthodes utilisent alors l'information sur le voisinage pour déterminer l'ensemble des nœuds à couvrir et calculer l'ensemble des nœuds qui doivent effectuer la retransmission. Ceci s'effectue en utilisant l'algorithme de *flooding* approprié à la simulation. En fait, ces deux méthodes créent selon le cas un objet instance de la classe *Algorithm* afin d'exécuter notre algorithme de *flooding* ou de la classe *PDP-Algorithm* pour utiliser l'algorithme PDP. Les trois dernières méthodes *SayHello*, *TreatReceivedPackets* et *Broadcast* permettent respectivement d'envoyer un paquet *HELLO*, de scruter la queue de réception afin de traiter les messages reçus, et vérifier la queue de transmission afin d'envoyer le premier paquet qu'elle contiendrait. Notons que

ces trois méthodes renvoient un booléen pour signifier si elles se sont exécutées ou non. En fait, dans l'affirmative, ceci voudrait dire qu'un paquet vient d'être envoyé ou reçu, ce qui implique que le médium radio a été utilisé. Cette information est nécessaire afin de mener à bien la tâche d'exclusion mutuelle qui conditionne l'accès au médium radio.

- **La classe *PDP-Algorithm***

L'étude de performance de notre algorithme se fera en partie en le comparant avec l'algorithme PDP. D'ailleurs, notre algorithme se base sur ce dernier et essaie de l'améliorer. Ainsi, il est important de conduire les mêmes séries de tests et épreuves pour les deux algorithmes. Dans cette optique, nous créons la classe *PDP-Algorithm* qui implémente le PDP. Comme il est montré à la Figure 4.7, cette classe offre principalement un constructeur qui permet d'initialiser cet algorithme en déterminant surtout l'ensemble des voisins à couvrir en éliminant ceux ayant "certainement" déjà reçu le paquet de *flooding*. Ensuite, la méthode *Execute* permet de déterminer un sous-ensemble minimum de voisins qui doivent effectuer la retransmission pour compléter le *flooding*. Cette méthode renvoie la liste des nœuds retransmettant sous forme d'une queue.

```
class PDP-Algorithm
{
    CQueue<int> B;
    CQueue<int> U;
    CQueue<int> Sf[MAX_MOBILES];
    int **Network;

public:
    PDP-Algorithm();
    PDP-Algorithm(int Id, int SenderId, int **N);
    CQueue Execute();
    ~CPDPAlgorithm(void);
};
```

Figure 4.7 Interface de la classe *PDP-Algorithm*

- **La classe *Algorithm***

Cette classe nous permet de conduire les tests désirés en utilisant l'algorithme développé. Comme on peut le voir à la Figure 4.8 donnant l'interface de cette classe, en plus du constructeur et de la méthode *Execute* dont le fonctionnement reste le même que celui décrit dans le paragraphe précédent, cette classe offre la méthode *IsGoodLink*.

La méthode *IsGoodLink* est une implémentation du critère γ qui nous permet de conclure si un lien radio risque de se briser ou s'il est de confiance. Cette méthode prend en paramètre le degré d'appartenance d'un mobile à l'ensemble des voisins d'un autre, soit la qualité du lien radio liant les deux mobiles, tel que nous l'avons définie au chapitre précédent. Notons que cette méthode est importante puisqu'elle nous permet d'identifier les nœuds critiques par rapport aux autres et donc de garantir autant que possible l'opération de livraison des paquets. Étant donné l'importance de ce critère, nous nous attarderons un peu plus loin sur la manière de le construire et de l'implémenter.

```
class Algorithm
{
    CQueue<int> B;
    CQueue<int> U;
    CQueue<float> Sf[MAX_MOBILES];
    int **Network;
    int Source;

    bool IsGoodLink(float MembershipDegree);

public:
    Algorithm(int Id, int SenderId, int **N);
    ~Algorithm(void);
    CQueue Execute();
};
```

Figure 4.8 Interface de la classe *PDP-Algorithm*

- **La classe *Queue***

Parmi toutes les classes que nous avons évoquées, la classe *Queue* représente la structure de donnée proprement dite, dans le sens où cette classe est totalement dédiée à

construire une structure de données. Cette classe sert principalement de queue de transmission ou de réception mais aussi pour sauvegarder l'historique des paquets transmis ou reçu, ou en tant que simple queue pour stocker diverses données. Cette classe a été implémentée d'une manière générique afin qu'elle puisse accueillir divers types de données. Son interface est fournie à la Figure 4.9.

```

Template <class Object>
class Queue
{
    public:
        Queue(void);
        void operator = (Queue<Object>);
        void Push(Object Element);
        void Remove(Object Element);
        Object Pull();
        void Reset();
        int GetSize();
        Object GetElement(int Position);
        void ExclusivePush(Object Element);
        bool Exist(Object Element);
        ~CQueue(void);
    private:
        Object Q[MAX_QUEUE_SIZE];
        int Size;
};

```

Figure 4.9 Interface de la classe *Queue*

4.1.2 Critère γ

Tel que nous l'avons vu au chapitre précédent, l'amélioration principale qu'apporte notre algorithme de *flooding*, que nous appelons dorénavant « l'algorithme $\gamma+$ » (car basé sur un critère nommé γ), par rapport au PDP est qu'il se base sur l'approche de modélisation en ensembles flous. Celle-ci permet de tenir compte de l'incertitude quant à la quantification de la qualité des liens radio et donc permet de faire de meilleurs choix quant aux routes que les paquets de *flooding* doivent emprunter.

Rappelons aussi que la modélisation de cette incertitude repose sur la donnée d'une série de valeurs pour la dispersion σ (Formule 2.3) qui est relative à l'environnement de propagation. Nous avons vu également que σ dépend de la distance entre l'émetteur et le récepteur et doit être quantifiée à l'aide de mesures sur le terrain.

Dans la littérature, plusieurs travaux de mesures sur le terrain ont été effectués afin d'estimer les valeurs de σ pour différents scénarios d'environnements. Citons notamment les travaux de Cox et al. (1984), de Seidel et al. (1991) et de Liberti et Rappoport (1992). Pour les réseaux Ad-hoc, ces travaux ont été orientés vers des mesures dans un environnement intérieur limité (Anderson et al. 1995), où la valeur de σ est supposée fixe et indépendante de la distance émetteur/récepteur. Dès lors, il est difficile de caractériser la propagation dans un environnement extérieur, surtout lorsque la mobilité atteint de grandes vitesses tel qu'il est prévu dans notre plan d'expériences. Par contre, on sait parfaitement caractériser la propagation dans l'espace libre que nous utilisons dorénavant comme milieu de propagation. Cet environnement, comme il est le cas lorsque l'on fixe σ , élimine la composante de l'incertitude reliée à l'environnement de propagation telle que décrite lors de l'établissement du critère γ au chapitre précédent. Par contre, l'autre forme d'incertitude reliée essentiellement à la mobilité des usagers et qui agit aussi sur la quantification de la qualité d'un lien radio persiste. En effet, comme on l'a vu au chapitre 3 la quantification de la qualité des liens radio se fait à l'aide des messages *HELLO* envoyés périodiquement par les nœuds Ad-hoc. Or, entre le moment où un nœud reçoit un message *HELLO* et le moment où il décide d'exploiter cette information pour des fins de *flooding*, la qualité des liens a fort probablement changé à cause des mouvements des usagers.

Pour pallier ce problème, nous construirons le critère γ adéquat qui permet de tenir compte de cette incertitude. Pour ce faire, considérons un réseau Ad-hoc dont les unités mobiles émettent à la même puissance, évoluant dans l'espace libre avec des vitesses variables inscrites dans l'intervalle $[V_MIN, V_MAX]$. Considérons particulièrement deux unités Ad-hoc u et v , illustrées à la Figure 4.10, qui ont pu s'échanger des messages *HELLO* auparavant. Mettons-nous dans le contexte où l'unité

mobile u aimerait quantifier la qualité du lien radio qui la relie à v , ce qui revient à déterminer si v est un nœud critique par rapport à u et, en d'autres termes, si le lien radio qui la relie à v est toujours de confiance ou a eu suffisamment de chance de se briser depuis que u a reçu le dernier message *HELLO* de v .

Notons alors R le rayon de couverture de u , Δt le temps écoulé entre l'instant t présent et l'instant t_0 où u a reçu le dernier message *HELLO* de v , et ρ la distance entre les deux unités à l'instant t_0 . Notons Δx la distance qu'a pu alors parcourir v durant Δt . Au pire des cas, on a :

$$\Delta x = 2 \cdot V_MAX \cdot \Delta t$$

Δt étant de valeur maximale $\Delta t = 3$ sec, nous négligeons l'effet de l'accélération dans cette évaluation.

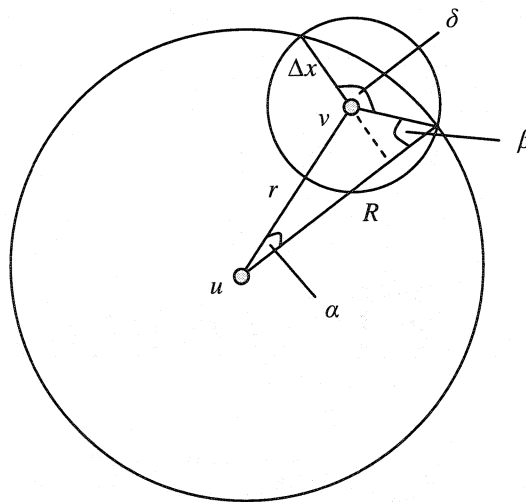


Figure 4.10 Définition du critère γ

Traçons alors le cercle de centre v et de rayon Δx . Ce cercle inscrit la région où peut se trouver le mobile v après Δt . Au pire des cas, le mobile v se retrouvera sur le périmètre du cercle tracé. La probabilité que v ne reçoit pas le paquet de u est évaluée à :

$$p = \delta / 2\pi$$

avec :

$$\delta = 2(\alpha + \beta)$$

et :

$$r \cdot \sin(\alpha) = \Delta x \cdot \sin(\beta)$$

$$r \cdot \cos(\alpha) = R - \Delta x \cdot \cos(\beta)$$

si :

$$r + \Delta x \leq R$$

En fait, dans le cas contraire, on est sûr que le mobile v se trouve dans l'étendue de la région de transmission de u et qu'il pourra recevoir le message. Dans cette hypothèse, on obtient :

$$\cos(\alpha) = \frac{R^2 - \Delta x^2 + r^2}{2 \cdot R \cdot r}$$

$$\cos(\beta) = \frac{R^2 + \Delta x^2 - r^2}{2 \cdot R \cdot \Delta x}$$

On peut alors construire le critère γ de la manière suivante :

$$p \cdot (r + \Delta x) \geq (1 - p) \cdot R \quad \Rightarrow \quad \text{le lien } (u, v) \text{ est faible.}$$

Notons que ce critère peut être relaxé ou serré suivant la tolérance que l'on désire aux pertes des paquets de *flooding*, ou plutôt, précisons-le, selon la tolérance que l'on désire à la livraison de paquet de *flooding*. En fait, tel que notre algorithme a été construit, serré de plus en plus le critère γ correspond non pas à diminuer la perte de paquets, dans une dynamique point à point mais plutôt à diminuer la non livraison de paquet ce qui revient à augmenter la couverture du réseau ou le nombre d'unités mobiles qui reçoivent le paquet de *flooding*. Pour ajuster le critère γ , il suffit de diviser Δx par un facteur $0 < \gamma \leq 1$. Pour le cas de nos expériences, nous avons utilisé la formule suivante :

$$\Delta x = \gamma \times \Delta t \times (V_MIN + V_MAX) / 2 \quad (4.1)$$

$$0 < \gamma \leq 3 \text{ et } \Delta t = HELLO_INTERVAL \cdot (1 + ALLOWED_HELLO_LOSS)$$

ceci correspond à utiliser des vitesses moyennes sur un intervalle de temps plus étendu.

4.1.3 Critère d'arrêt

Tel que nous l'avons présenté jusque là, l'algorithme $\gamma+$ peut conduire à des situations très délicates. En effet, dans certaines circonstances, effectuer le *flooding* à l'aide de notre algorithme risque d'entraîner le réseau dans une boucle infinie de demandes de transmissions, même si tous les nœuds ont déjà reçu le paquet de *flooding*. Une telle situation, illustrée à la Figure 4.11, peut vite conduire le réseau à la saturation.

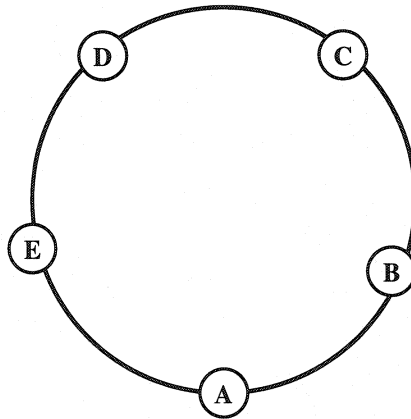


Figure 4.11 Exemple de *flooding* infini

Considérons, dans le réseau Ad-hoc de la Figure 4.11, que le mobile A aimerait effectuer du *flooding*. Dans ce cas là, selon notre algorithme $\gamma+$, le nœud A demandera aux nœuds B et E de retransmettre le paquet afin de couvrir C et D respectivement. À leur tour, les nœuds B et E demanderont aux nœuds C et D de retransmettre afin de couvrir D et C respectivement. L'étape qui suit consistera dans le fait que les nœuds C et D demanderont aux nœuds B et E respectivement de retransmettre afin de couvrir A. Ensuite, les nœuds B et E demanderont au nœud A d'effectuer la retransmission du paquet afin de couvrir les nœuds E et B respectivement. Ainsi, nous revenons à la case de départ et le *flooding* continuera indéfiniment même si tous les nœuds ont reçu le paquet de *flooding*.

Dans ces conditions, un critère d'arrêt est absolument nécessaire pour mettre fin à cette boucle infinie. Ce critère, malgré son importance, peut conduire à des situations non désirées. En effet, un critère d'arrêt trop sévère pourrait limiter la diffusion du paquet de

flooding et réduire ainsi la couverture. Donc, un critère d'arrêt devrait être construit avec précautions, sous peine de briser le *flooding*. Les auteurs du PDP préconisent le critère suivant : *Un nœud ne peut retransmettre un paquet de flooding qu'il a déjà retransmis auparavant, même si un de ces voisins le demande* (Lou et Wu, 2002). Ce critère d'arrêt est effectivement efficace. En effet, son application à l'exemple précédent arrêtera la boucle du *flooding* au niveau de la deuxième retransmission des nœuds *B* et *E*, après que tous les nœuds aient reçu le paquet. Les tests effectués par les auteurs en utilisant ce critère confirment sa performance.

Toutefois, remarquons que si un nœud u reçoit un premier message de flooding lui demandant de retransmettre, c'est que la retransmission de u est nécessaire afin de couvrir son voisinage et il doit l'effectuer. Par contre, si u reçoit le premier paquet de *flooding* sans demande de retransmission, cela veut dire que la retransmission de u n'est pas nécessaire et que son voisinage a été couvert par d'autres nœuds. Dans les deux cas, si u reçoit une deuxième fois le même paquet de *flooding* lui demandant de le retransmettre, nous considérons que cette retransmission est inutile car, dans les deux cas, le voisinage de u est déjà couvert. Ainsi, le critère d'arrêt qui nous paraît adéquat serait : *Un nœud ne peut retransmettre un paquet de flooding qu'il a déjà reçu auparavant, même si un de ces voisins le demande*. L'application de ce critère à l'exemple précédent permet effectivement d'arrêter le *flooding* après que tous les nœuds aient reçu le paquet. Ceci dit, sa performance reste à vérifier à l'aide des simulations afin de s'assurer qu'il ne pose pas de problèmes secondaires indétectables à ce niveau, notamment en terme de couverture.

4.1.4 Environnement matériel et logiciel

Tel que décrit au chapitre précédent, le but de notre algorithme est d'optimiser une opération de *flooding*. Nous nous sommes ainsi préoccupés de maximiser la couverture du réseau en minimisant sa charge. Dans le contexte du *flooding*, ceci revient à maximiser le nombre de nœuds qui reçoivent le paquet objet du *flooding*, tout en minimisant le nombre de retransmissions effectuées. Ainsi, deux mesures de performance

pertinentes pour notre solution au problème du *flooding* sont le nombre de nœuds retransmettant et le nombre de nœuds couverts.

Le choix du logiciel de la simulation a balancé entre deux outils : le simulateur de réseaux OPNET dont on dispose au sein du LARIM des licences d'exploitation et Visual C++ également disponible. Nous avons commencé par scruter les possibilités que nous offre l'un et l'autre. Pour Visual C++, nous ne disposons d'aucun "package" spécial nous permettant de faciliter l'implémentation et il fallait définir le réseau et les échanges de messages. Par contre, OPNET offre une architecture complète pour un nœud Ad-hoc incluant la couche MAC mais ne fournit pas de protocoles pour l'exploitation de ce modèle. En fait, le seul protocole implémenté que nous avons pu trouver est AODV qui n'offre pas les possibilités de *flooding* appropriées, dans le sens où il est basé sur la construction et le maintien de tables de routage, alors que notre approche est totalement différente. Une autre raison qui nous a conduit à écarter AODV est qu'un protocole de *flooding* doit être léger et très simple pour ne pas alourdir le réseau, étant donné que ce genre de protocole n'est pas dédié à la communication mais qu'il fonctionne pour faciliter la tâche aux protocoles plus lourds, notamment ceux de routage qui l'utilisent. Donc, utiliser un protocole de *flooding* par-dessus un protocole de routage est exclu. D'un autre côté, nous avons remarqué que les nœuds mobiles Ad-hoc dans OPNET doivent suivre un modèle de mobilité préétabli. Or pour notre séquence de tests, nous aimerions laisser aléatoires les déplacements des mobiles, surtout que cela semble infaisable s'il s'agit de construire un modèle de mobilité pour chaque nœud lors d'une expérience utilisant une centaine de nœuds, tel que notre plan d'expériences l'impose.

Afin d'évaluer les performances de notre algorithme et de conduire les séries de simulation désirées, nous avons finalement choisi d'implémenter notre algorithme sur Visual C++.NET en utilisant l'architecture d'application MS Framework. Les simulations quant à elles ont été conduites au sein du LARIM sur une machine Pentium III à 800 MHz et 256 Mo de mémoire vive.

4.2 Plan d'expériences

Dans notre approche de solution du problème du flooding, nous avons proposé un algorithme utilisant le calcul des plus courts chemins et une version améliorée du PDP en tenant compte de l'état de liens dans un réseau Ad-hoc. Nous avons ainsi modélisé l'incertitude résultante de la quantification de la qualité d'un lien radio. Par la suite, nous nous proposons de mener une série de tests afin d'évaluer les performances du nouvel algorithme par rapport au PDP, l'un des meilleurs algorithmes de *flooding* existant dans la littérature.

Pour optimiser le *flooding*, nous devons surtout essayer d'augmenter la couverture du réseau en minimisant sa charge. Ceci revient à augmenter le nombre de nœuds recevant le paquet objet du *flooding* en minimisant le nombre de retransmissions qui doivent être effectuées pour atteindre cette couverture. Ainsi, deux mesures de performance pertinentes au contexte du *flooding* sont: le nombre de nœuds recevant le paquet objet du *flooding* et le nombre de retransmissions devant être effectuées pour arriver à cette couverture. Ajouté à cela, on peut penser à évaluer le nombre de paquets perdus à cause d'une mauvaise décision de routage ou d'une mauvaise quantification de la qualité d'un lien radio.

Toutefois, notons que ces mesures de performance dépendent de plusieurs paramètres ou variables de décisions qui peuvent y avoir un impact important. Notamment, mentionnons que le nombre de nœuds couverts et le nombre de nœuds retransmettants dépendent étroitement du nombre NB_MOBILE de nœuds dans le réseau, de l'étendue de l'espace $[X_MIN, X_MAX] * [Y_MIN, Y_MAX]$ que peut parcourir les unités mobiles, de la valeur du facteur γ dans la formule (4.1) et de la puissance P de transmission des mobiles. Notons aussi que le nombre de paquets perdus dépendra de la vitesse des unités mobiles, soit de l'intervalle $[V_MIN, V_MAX]$ des vitesses que prennent les unités Ad-hoc.

Par ailleurs, soulignons l'étroite corrélation entre le nombre NB_MOBILE de mobiles dans le réseau, la puissance P d'émission de ces mobiles et l'étendue de l'espace où évolue le réseau Ad-hoc donné par les paramètres : $X_MAX - X_MIN$ et $Y_MAX -$

Y_{MIN} . En réalité, ces trois paramètres expriment une seule propriété, à savoir la densité du réseau. Effectivement, augmenter par exemple la puissance de transmission des unités Ad-hoc reviendrait à diminuer l'étendue $E = (X_{MAX}-X_{MIN})*(Y_{MAX}-Y_{MIN})$ ou augmenter NB_MOBILE et vice-versa. En fait, nous pouvons exprimer cette densité par une fonction à trois variables : $D = f(E, NB_MOBILE, P)$. Ainsi, varier une des trois variables E , NB_MOBILE ou P , les autres étant fixes, nous permettrait de balayer l'espace des valeurs pour la densité. Toutefois, par souci de précision, nous varions les deux paramètres NB_MOBILE et P indépendamment, E étant fixé, ce qui nous permet de réaliser un meilleur échantillonnage de l'espace des valeurs de la densité et surtout de pouvoir projeter les résultats des expériences sur deux axes différents exprimant chacun une propriété à part entière.

Ainsi, des cinq variables de décision citées plus haut, nous retenons quatre comme étant explicitement indépendantes à savoir : le facteur γ qui intervient dans la sélection des nœuds critiques, l'intervalle $[V_{MIN}, V_{MAX}]$ des vitesses que peuvent prendre les mobiles, le nombre NB_MOBILE de mobiles dans le réseau Ad-hoc et la puissance P d'émission des mobiles. Par la suite, nous nous proposons de fixer $X_{MAX}-X_{MIN}$ et $Y_{MAX}-Y_{MIN}$ à 1000 m et de varier les quatre autres variables de décision indépendamment les uns des autres : γ dans l'ensemble $\{0.25, 0.50, 0.75, 1.0, 1.25, 1.50, 1.75, 2.0, 2.25, 2.50\}$, $[V_{MIN}, V_{MAX}]$ dans l'ensemble $\{[0, 20], [20, 40], [40, 60], [60, 80], [80, 100], [100, 120], [120, 140]\}$, NB_MOBILE dans l'ensemble $\{20, 30, 40, 50, 60, 70, 80, 90, 100, 110, 120\}$ et la puissance P dans l'ensemble $\{100, 125, 150, 175, 200, 225, 250, 275, 300, 325, 350\}$. Ainsi, nous ferons varier toutes les variables de décision à la fois et nous projetterons les résultats sur un axe représentant une d'entre elle afin d'évaluer son impact sur les performances des deux algorithmes.

4.3 Analyse des résultats d'expériences

Dans la présente section, nous conduisons des séries de tests afin d'évaluer les performances de notre algorithme. Nous nous préoccupons surtout d'étudier le comportement de l'algorithme $\gamma +$ dans différents cas de tests régis par les variables de

décision établies à la section précédente. Nous présenterons aussi les mesures relatives aux trois indicateurs de performance recensés à savoir : le nombre de nœuds ayant reçu le paquet du *flooding*, le nombre de retransmissions effectuées ainsi que le nombre de paquets perdus.

Toutefois, afin de pouvoir mieux interpréter les performances de notre algorithme et surtout les comparer à ceux du PDP, nous procédons à la normalisation des indicateurs de performance présentés. Ainsi, au lieu de calculer le nombre *NbNoeudCouvert* de nœuds ayant reçus le paquet de *flooding* parmi le nombre total des nœuds *NB_MOBILE*, nous évaluerons le pourcentage de couverture noté *%Couverture*, qui est calculé de la manière suivante :

$$\%Couverture = 100 * NbNoeudCouvert / NB_MOBILE$$

De la même manière, au lieu de considérer le nombre *NbTx* de retransmissions effectuées par l'un des deux algorithmes, nous évaluerons le nombre de transmissions effectuées pour couvrir *NbNoeudCouvert*. Cette nouvelle mesure que nous noterons *%Tx* évalue en fait la densité des transmissions par rapport aux nœuds couverts, ce qui revient aussi à évaluer la charge du réseau. On a :

$$\%Tx = 100 * NbTx / NbNoeudCouvert$$

Dans la même optique, au lieu de considérer le nombre *NbTxFail* de transmissions qui ont échoué, nous préférons évaluer la densité des échecs de transmission par rapport aux nœuds couverts. Cette densité sera dorénavant appelée *%TxFail* et est calculée de la manière suivante :

$$\%TxFail = 100 * NbTxFail / NbNoeudCouvert$$

Considérons alors le scénario global où nous fixons l'étendue de l'expérience à 1000 m² et faisons varier toutes les variables de décision indépendamment dans leurs intervalles respectifs. Dans un premier temps, intéressons-nous à l'évaluation de la meilleure valeur à donner au facteur γ de la formule (4.1) portant sur la tolérance au perte de paquets ou plutôt sur la tolérance à la non livraison de paquets. Dans ces conditions, évaluons les différents indicateurs de performance en fonction de la variation du facteur γ

dans l'intervalle $[0.25, 2.50]$. Les résultats de l'expérience sont présentés à la Figure 4.12.

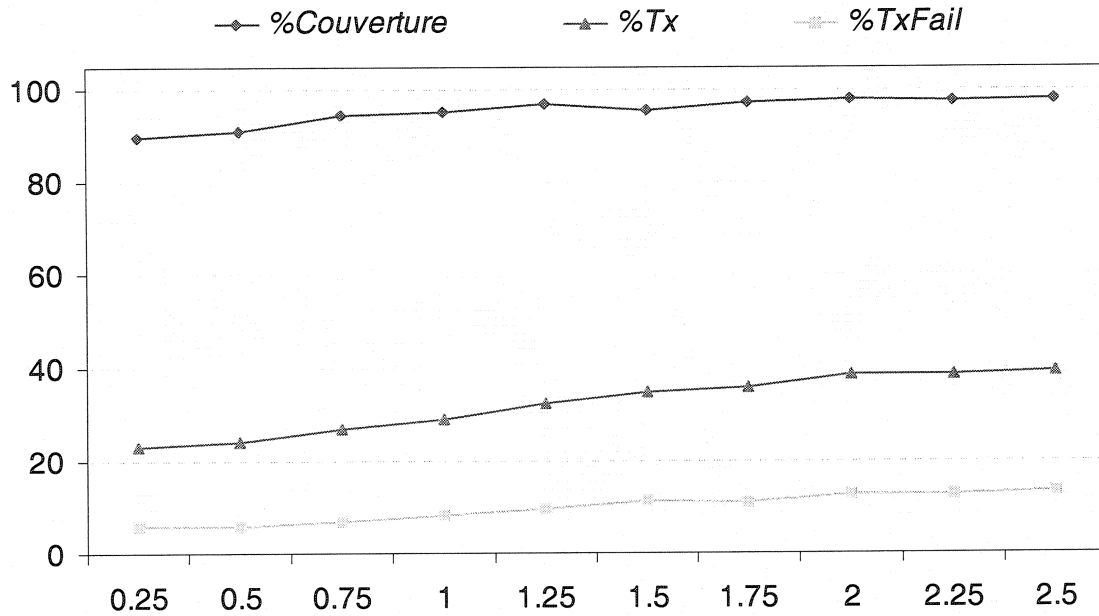


Figure 4.12 Les différents indicateurs de performance de $\gamma+$ selon la valeur du facteur γ de tolérance aux pertes

Comme on pouvait s'y attendre, on peut constater à la Figure 4.12 qu'au fur et à mesure que l'on serre le critère γ , la couverture du réseau augmente. En fait, ceci est réalisé en augmentant la charge du réseau et la densité de pertes de paquet. En effet, augmenter le critère γ conduit à utiliser des chemins de plus en plus courts pour acheminer les paquets de *flooding* et donc utiliser de plus en plus de retransmissions, ce qui explique l'augmentation de la charge du réseau. Augmenter le critère γ fait aussi en sorte d'augmenter l'ensemble des nœuds que nous considérons critiques et auxquels on va essayer de transmettre les paquets par les meilleurs chemins même si ces derniers sont critiques et cela à plusieurs reprises, ce qui explique l'augmentation de la densité de pertes des paquets. Ceci dit, notons que quelle que soit la valeur du facteur γ , l'algorithme $\gamma+$ assure en moyenne des couvertures supérieures à 90%. Notons aussi, qu'à partir de $\gamma=1$, les couvertures assurées varient de 2.66% (95.49% pour $\gamma=1$ et 98.15 pour $\gamma=2.5$);

ainsi, l'apport en couverture devient marginale. Par contre, la charge du réseau et la densité des pertes dans le même intervalle varient respectivement de 10 et de 6%, ce qui n'est pas négligeable. Pour cette raison, nous favorisons $\gamma=1$ comme la valeur qui nous permet d'obtenir les meilleurs résultats. Ainsi, dans tous les tests qui suivent, nous fixerons γ à 1 et ferons varier les autres paramètres de décision.

À présent, évaluons le comportement de notre algorithme face au changement de nombre de mobiles. Pour cela, nous fixons la valeur de *NB_MOBILE* dans l'intervalle $[20, 120]$ et faisons varier les autres variables de décision indépendamment, γ étant fixé à 1. La Figure 4.13 présente les résultats obtenus.

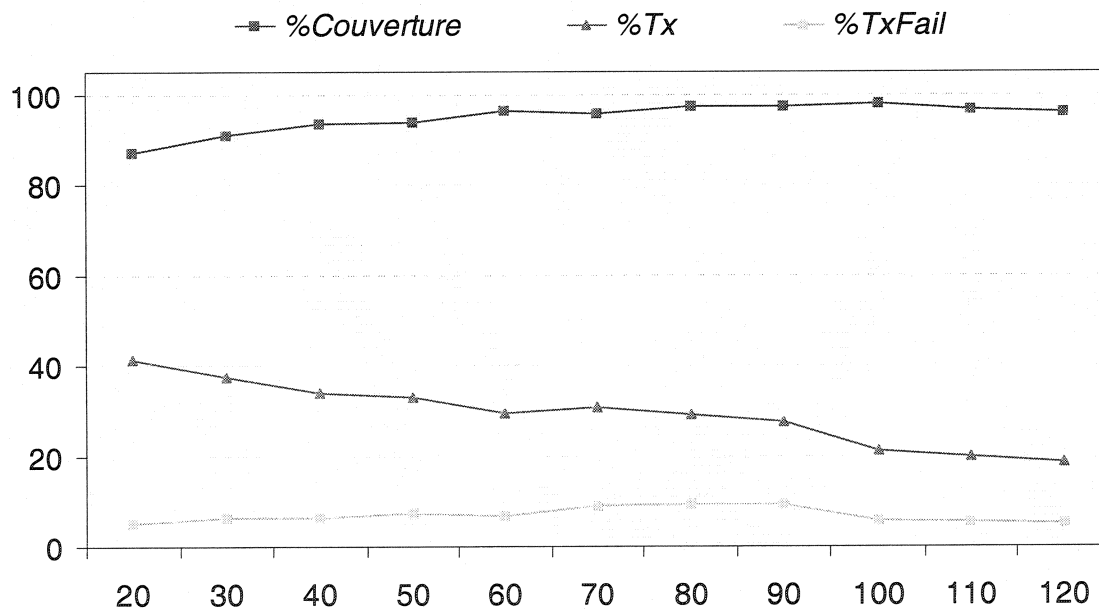


Figure 4.13 Les différents indicateurs de performance de $\gamma+$ selon le nombre de mobiles dans le réseau

À la Figure 4.13, nous remarquons que les performances de l'algorithme $\gamma+$ s'améliorent avec le nombre de mobiles dans le réseau. Ainsi, $\gamma+$ réalise de meilleures performances lorsque la densité du réseau croît. Notamment, la couverture dépasse 96% à partir de 60 nœuds mobiles et se situe entre 87 et 94% pour 50 nœuds et moins. Notons aussi qu'au fur et à mesure que la densité du réseau augmente, la couverture s'améliore et

la charge du réseau diminue fortement puisqu'elle passe de 41% pour 20 mobiles à 18% pour 120 mobiles. Ceci est tout à fait logique car, lorsque la densité du réseau augmente, chaque transmission permet de couvrir de plus en plus de nœuds. Notons aussi, et ceci est intéressant, que la densité des pertes de paquets reste presque stable du moment que sa variation maximale est de 3.66%.

Considérons maintenant, le scénario où l'on se propose d'évaluer le comportement de notre algorithme dans le cas d'une variation de la vitesse des mobiles. Pour cela, nous fixons les vitesses des mobiles dans les intervalles $[0, 20]$, $[20, 40]$, $[40, 60]$, $[60, 80]$, $[80, 100]$, $[100, 120]$ et $[120, 140]$, et faisons varier les autres paramètres de décision dans leurs intervalles respectifs. Ceci nous permet de calculer les moyennes des indicateurs de performance pour ce scénario que nous retrouvons à la Figure 4.14.

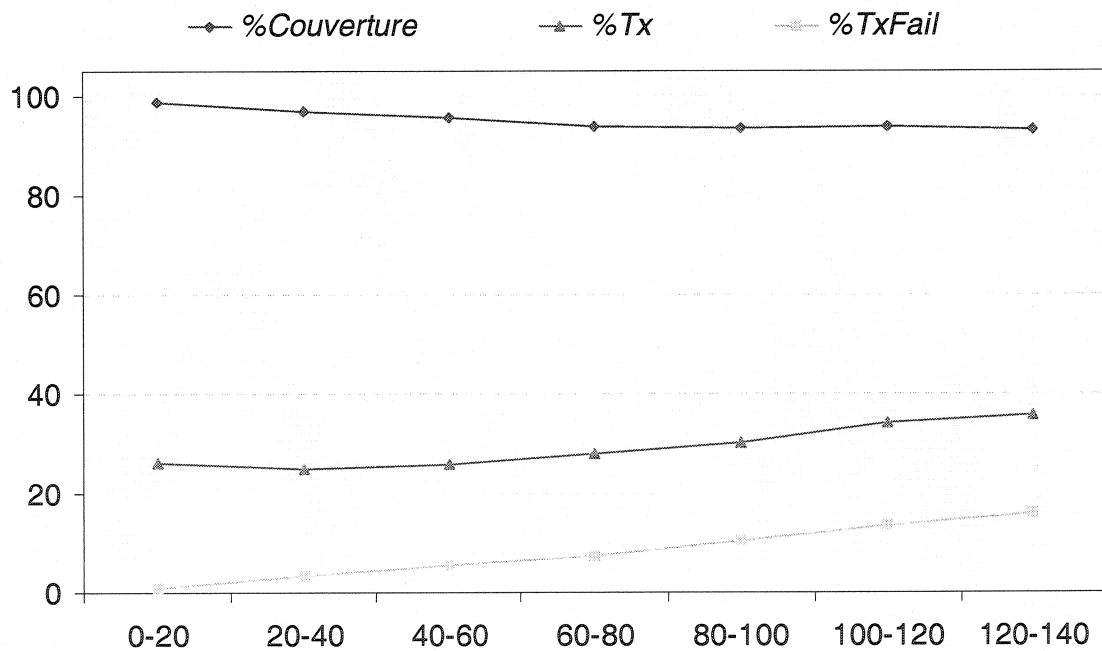


Figure 4.14 Les différents indicateurs de performance de $\gamma+$ selon la vitesse des mobiles en Km/h

À la Figure 4.14, nous constatons que la couverture moyenne qu'assure l'algorithme $\gamma+$ diminue avec la vitesse des mobiles. Toutefois, celle-ci reste supérieure à

93% même pour de très grandes vitesses frôlant les 140 Km/h. Ainsi, $\gamma+$ tend à conserver une bonne couverture moyenne malgré l'augmentation des vitesses d'opération des mobiles. Pour ce faire, $\gamma+$ augmente en même temps, comme on peut le constater à la Figure 4.13, la charge du réseau ainsi que la densité des pertes de paquets, ce qui est tout à fait logique. En effet, plus les vitesses des mobiles augmentent, plus rapidement les liens radio risquent de se briser, ce qui augmente le nombre de nœuds critiques et ainsi la charge du réseau et la densité de perte de paquets.

À ce point, il nous reste à évaluer le comportement de notre algorithme face au changement de la puissance d'émission des mobiles. Pour ce faire, considérons le scénario où nous faisons varier la puissance d'émission des mobiles dans l'intervalle $[150, 350]$ et calculons pour chaque valeur de la puissance la moyenne des indicateurs de performance, toutes les autres variables de décision variant dans leur intervalle respectif. Les résultats de l'expérience sont présentés à la Figure 4.15.

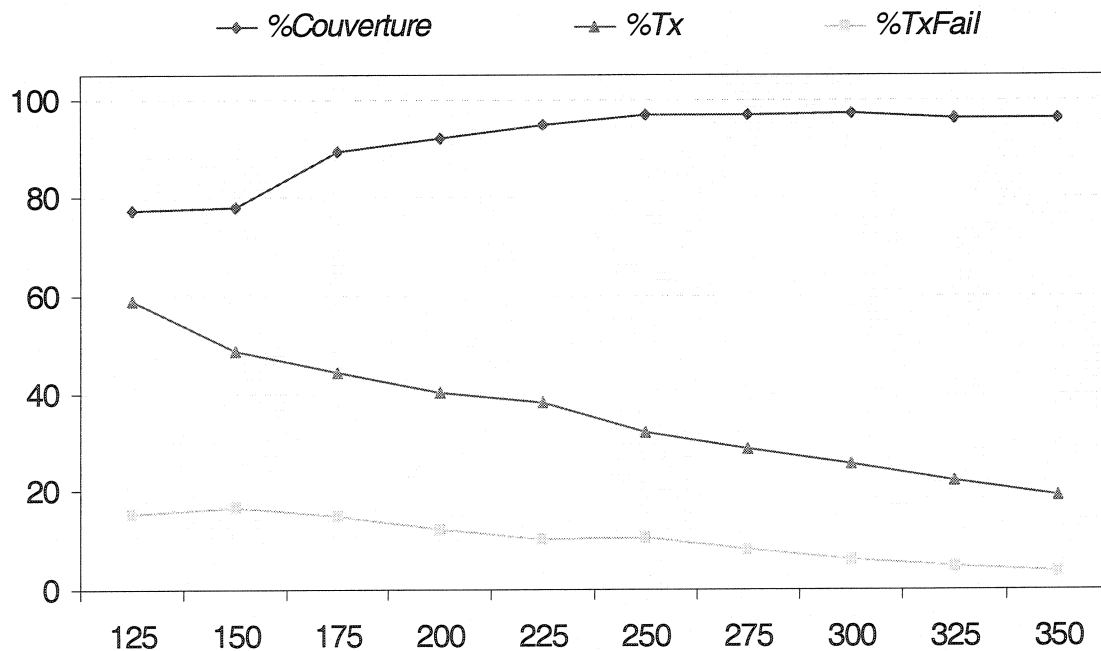


Figure 4.15 Les différents indicateurs de performances de $\gamma+$ selon la puissance des mobiles exprimée par la couverture en mètres

À la Figure 4.15, nous pouvons constater que les performances de l'algorithme $\gamma+$ s'améliorent avec la puissance d'émission des mobiles. En fait, nous remarquons, tel qu'il a été le cas pour la Figure 4.11, que les performances suivent la densité du réseau. Toutefois, notons qu'ici la couverture descend au dessous de 80% lorsque la puissance des mobiles est faible (125 à 150 m). Nous remarquons aussi qu'à partir de 250 m de puissance la couverture se stabilise aux alentours de 97%. Par contre, la charge du réseau ne cesse de chuter allant de 32% pour $P=250$ m à 19% pour $P=350$ m. Il est intéressant de noter qu'à l'encontre du scénario de la variation du nombre de mobiles dans le réseau où l'on a constaté que la densité de perte des paquets reste relativement stable avec la variation de la densité du réseau, dans ce scénario-ci, nous remarquons que la densité de perte de paquets ne cesse de chuter avec la densité du réseau. En effet, augmenter le nombre de mobiles augmente la densité du réseau, augmente du même coup le nombre de retransmissions nécessaires pour le *flooding* et par la suite augmente la densité de perte des paquets. Par contre, augmenter la puissance des mobiles augmente la densité du réseau en diminuant le nombre de paquets nécessaires pour accomplir une opération de *flooding*.

4.3 Comparaison avec l'algorithme PDP

Dans cette section, nous nous proposons de confronter les résultats des simulations portant sur les deux algorithmes $\gamma+$ et PDP. Nous nous intéressons alors à effectuer les mêmes séries de tests pour les deux algorithmes et recenser les résultats des simulations afin de les comparer.

Considérons alors le scénario global où nous fixons l'étendue de l'espace de l'expérience à 1000 m² et nous faisons varier les variables de décision P , NB_MOBILE et $[V_MIN, V_MAX]$ dans leurs intervalles respectifs tel que décrit dans le plan d'expériences. Commençons d'abord par évaluer $\%Couverture$ à l'aide des Figures 4.16 et 4.17. Notons que l'axe des abscisses réfère au numéro de l'expérience. Celui-ci a été et sera supprimé dans tous les graphiques où il intervient, à cause du peu d'intérêt qu'il présente pour les analyses.

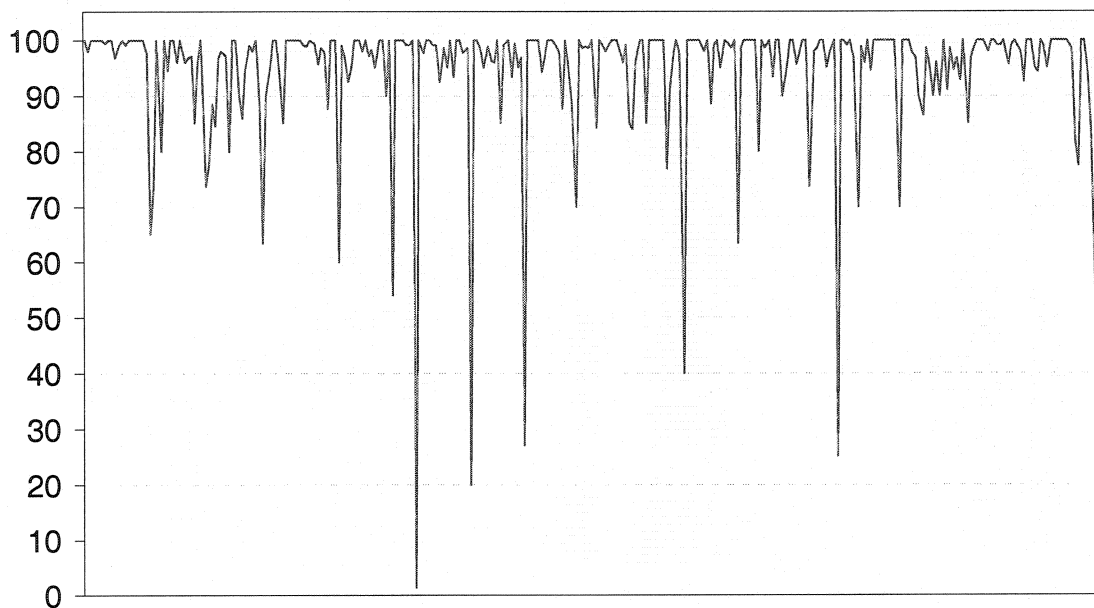


Figure 4.16 %Couverture pour $\gamma+$

La Figure 4.16 nous permet d'évaluer le pourcentage de couverture réalisé par notre algorithme $\gamma+$. Dans ce cas, nous pouvons constater que $\gamma+$ tend à assurer des couvertures globalement comprises entre 80% et 100%.

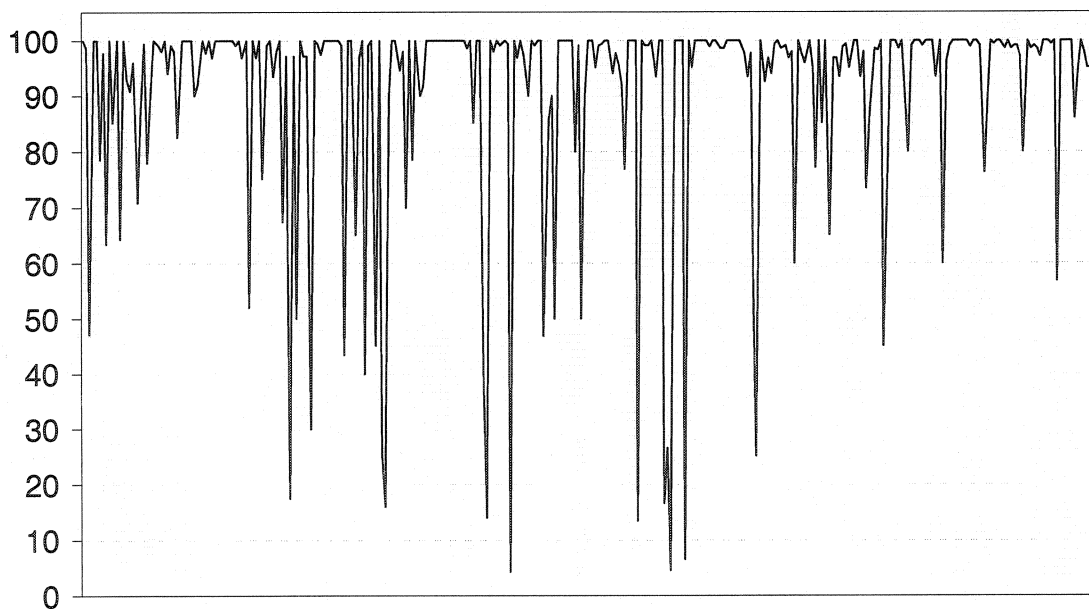


Figure 4.17 %Couverture pour PDP

À partir des deux figures précédentes, nous constatons que les deux algorithmes ont tendance à assurer des couvertures qui se situent globalement entre 80% et 100%, avec un avantage net pour $\gamma+$ qui descend peu de fois au-dessous de la barre des 80% de couverture. À ce niveau, nous aimerions calculer la couverture moyenne que garantit chacun des algorithmes. Pour cela, représentons les courbes des moyennes cumulatives.

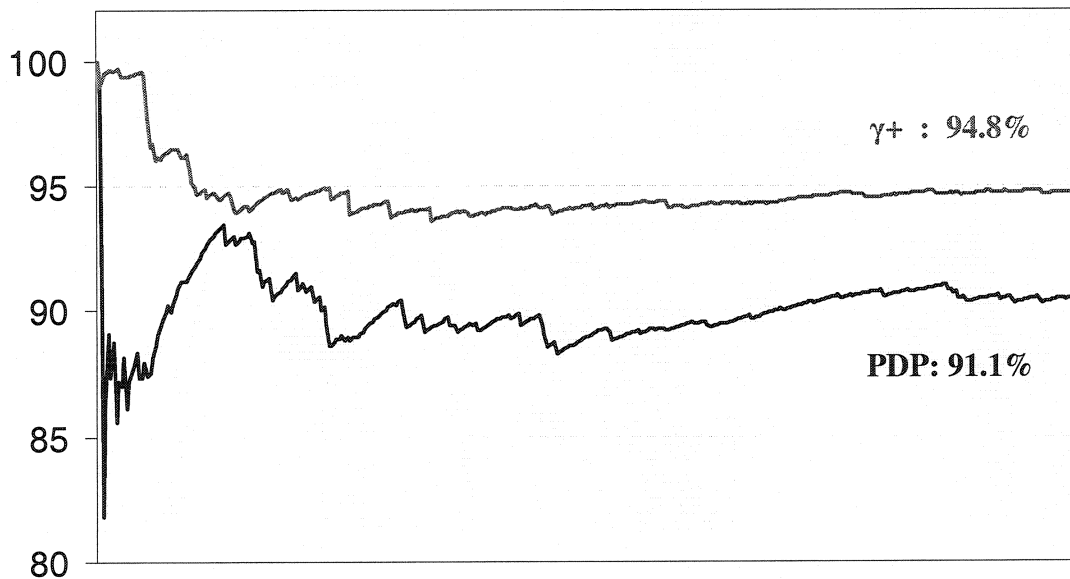


Figure 4.18 Moyenne cumulative du %Couverture pour PDP et $\gamma+$

La Figure 4.18 nous permet de conclure qu'en moyenne, les performances de $\gamma+$ sont meilleures que celles du PDP en terme de pourcentage de couverture. En effet, alors que le PDP garantit une moyenne de couverture de 91%, $\gamma+$ permet de couvrir 95% du réseau. À cette même figure, nous constatons que la courbe de la moyenne cumulative relative à $\gamma+$ semble tendre beaucoup plus rapidement vers la valeur moyenne, alors que celle du PDP fluctue énormément avant de tendre vraiment vers sa valeur moyenne. À priori, ceci voudrait dire que l'algorithme $\gamma+$ est plus stable que le PDP. Pour mieux illustrer cela, représentons les pourcentages concernant les fréquences d'occurrences des valeurs de %Couverture.

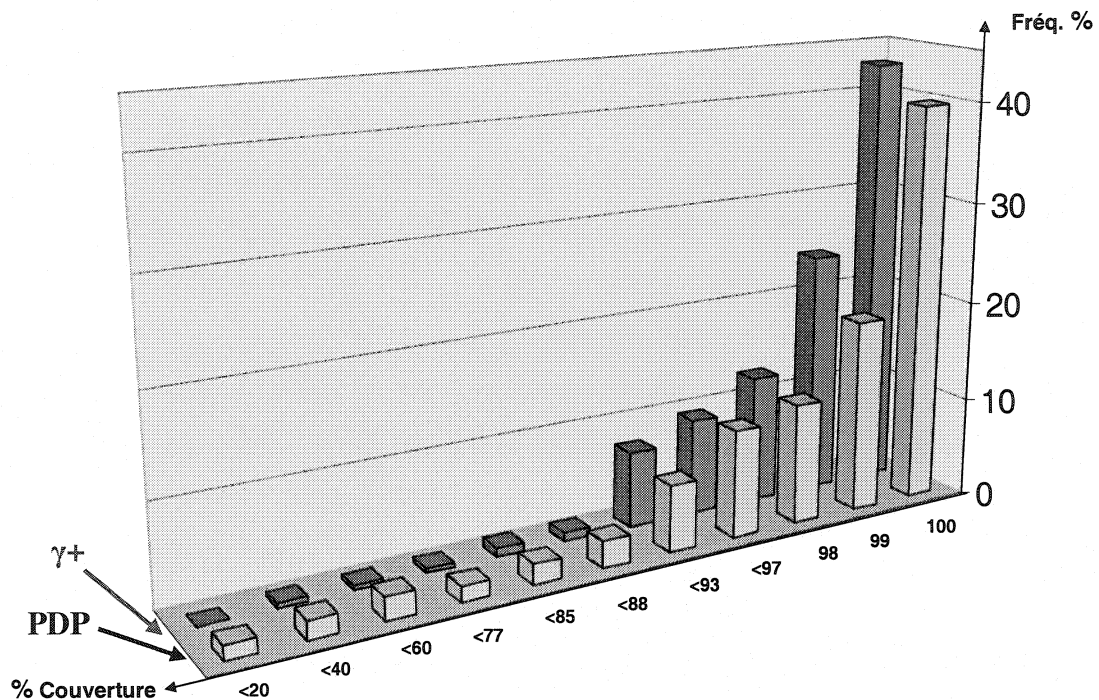


Figure 4.19 Pourcentage de fréquence du %Couverture pour PDP et $\gamma+$

À la Figure 4.19, nous remarquons que l'algorithme $\gamma+$ dépasse le PDP pour de très bons pourcentages de couverture de l'ordre de 98, 99 et 100%. Par contre, dès qu'il s'agit de couvertures moyennes ou basses, évaluées globalement à moins de 88% de couverture, le PDP offre plus de couverture dans cette gamme là, alors que $\gamma+$ tend à frôler le zéro. Ceci confirme effectivement le fait que $\gamma+$ soit plus stable que le PDP, chose que nous avons aussi relevée à partir des Figures 4.16 et 4.17. Pour évaluer cet apport en stabilité, nous pouvons calculer l'écart type standardisé. Celui-ci est évalué à 21,12% pour le PDP et à 12,68% pour $\gamma+$. Ainsi, nous concluons que $\gamma+$ est à peu près deux fois plus stable que le PDP.

Jusque là, nous avons vu que l'algorithme $\gamma+$ garantit une couverture plus grande et offre une meilleure stabilité que le PDP. Toutefois, il nous incombe de nous demander à quel prix cela est réalisé. En effet, si $\gamma+$ augmente en même temps la charge du réseau ou la densité des transmissions échouées, alors ses avantages tourneront plutôt en

inconvenients. Calculons alors la charge du réseau ainsi que la densité d'échec de transmission.

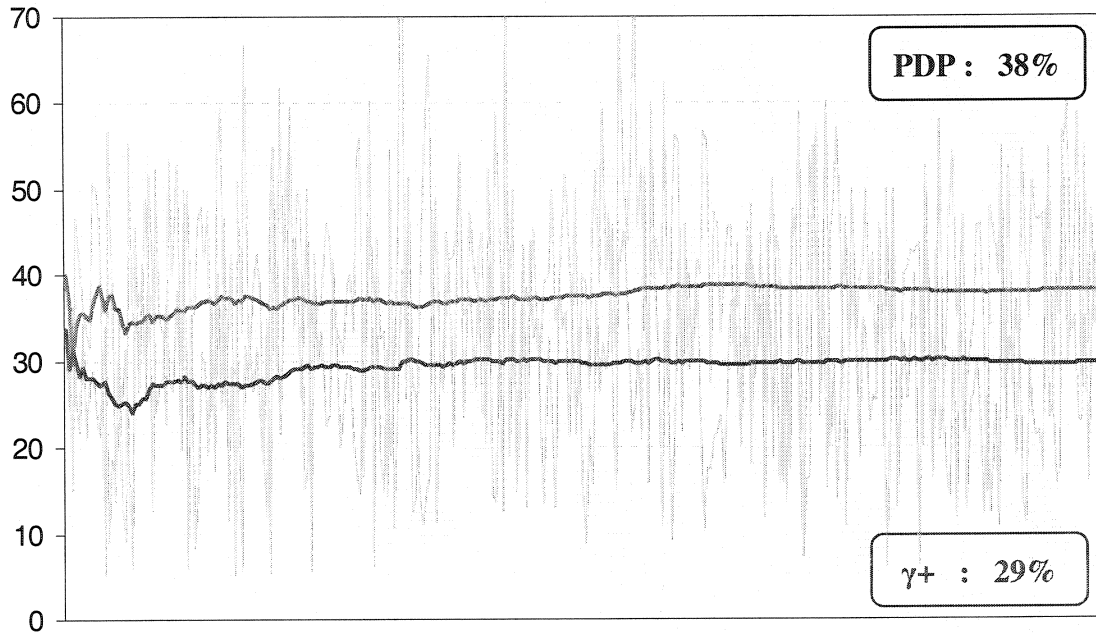


Figure 4.20 Moyenne cumulative de la charge du réseau pour PDP et $\gamma+$

À la Figure 4.20, nous avons produit en même temps les courbes représentant la charge du réseau ainsi que leurs moyennes cumulatives. Nous en concluons que l'algorithme $\gamma+$ permet de réduire sensiblement la charge du réseau. Ainsi, en plus d'augmenter la couverture et d'améliorer la stabilité, $\gamma+$ permet de réduire la charge du réseau, ce qui lui assure de très bonnes performances à tous les niveaux. Jetons alors un coup d'œil sur la densité des transmissions échouées illustrées à la Figure 4.19. Notons que ce critère perd un peu de sa valeur car, quels que soient les résultats d'évaluation par rapport à ce critère, $\gamma+$ assure de meilleures performances que le PDP. À partir de cette figure, nous pouvons conclure que $\gamma+$ assure une meilleure livraison des paquets de *flooding* que le PDP. Donc, en définitif pour ce cas de scénario, $\gamma+$ garantit de meilleures performances que le PDP sur tous les plans.

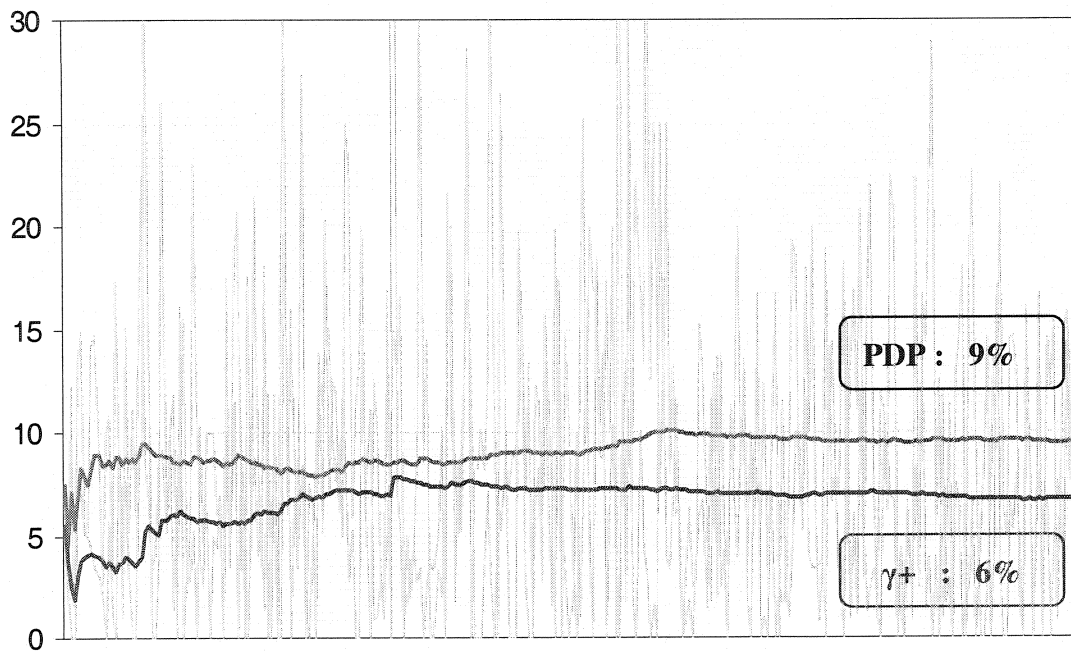


Figure 4.21 Moyenne cumulative de la densité d'échec des transmissions pour PDP et $\gamma+$

Évaluons alors les performances de $\gamma+$ par rapport au PDP en considérant l'impact relié au changement de vitesses des mobiles. Dans ce contexte, fixons les valeurs du couple de variables (V_{MIN}, V_{MAX}) dans l'ensemble $\{(0, 20), (20, 40), (40, 60), (60, 80), (80, 100), (100, 120)\}$ et évaluons les indicateurs de performance tel que nous l'avons effectué dans ce qui précède. Présentons alors ces différents indicateurs à la même Figure 4.22 selon l'intervalle pris par les vitesses.

À la Figure 4.22, nous pouvons constater, tous les critères confondus, que quelle que soit la vitesse des mobiles, les performances de l'algorithme $\gamma+$ dépassent celles du PDP. Nous pouvons aussi constater que $\gamma+$ tend à conserver une très bonne couverture moyenne, malgré l'augmentation des vitesses d'opération des mobiles. Pour ce faire, $\gamma+$ augmente la charge du réseau ainsi que la densité des pertes de paquets ce qui est tout à fait logique. Par contre, nous remarquons qu'au fur et à mesure que les vitesses augmentent, la couverture assurée par le PDP chute, la charge du réseau chute aussi, alors

que la densité des pertes augmente. Ceci signifie que le PDP est instable et est inadéquat pour de grandes vitesses d'opération.

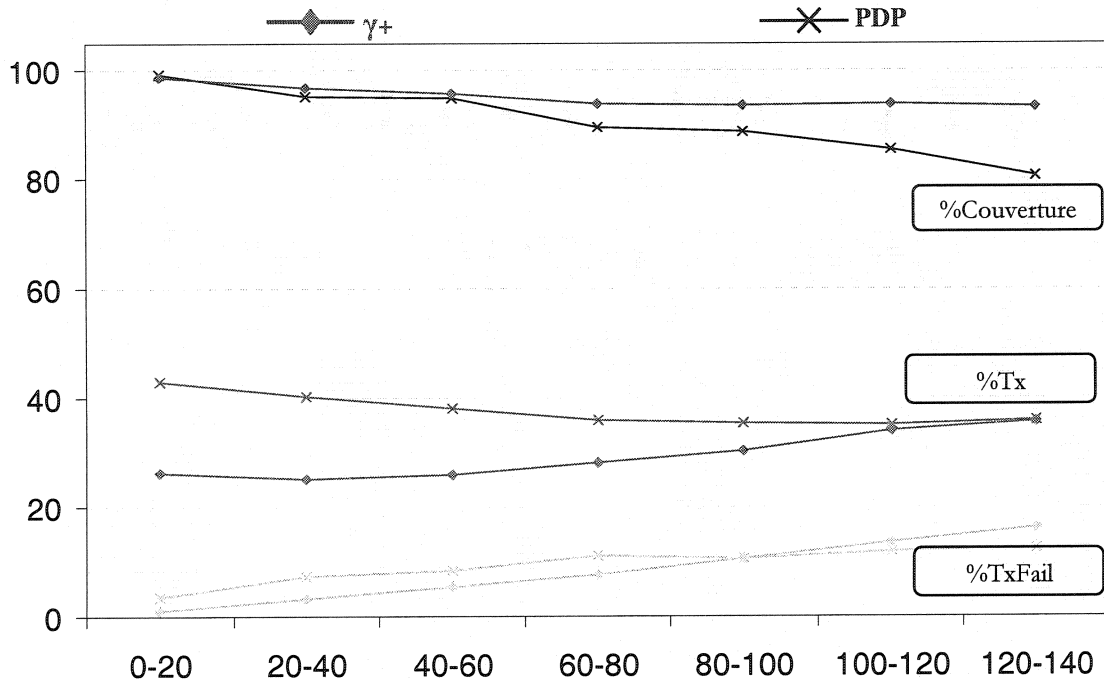


Figure 4.22 Les différents indicateurs de performance selon la vitesse des mobiles en Km/h pour PDP et $\gamma+$

Tel que décrit au plan d'expériences, l'étape suivante consiste à évaluer les performances des deux algorithmes face à une variation de la densité du réseau que nous réalisons d'abord en variant le nombre de mobiles et ensuite en variant la puissance d'émission des mobiles. Dans cette optique, fixons le nombre NB_MOBILE dans l'intervalle $[20, 120]$ et évaluons les indicateurs de performance de la même manière que nous l'avons effectué précédemment. La Figure 4.23 récapitule les résultats des simulations obtenus en fixant le nombre des mobiles.

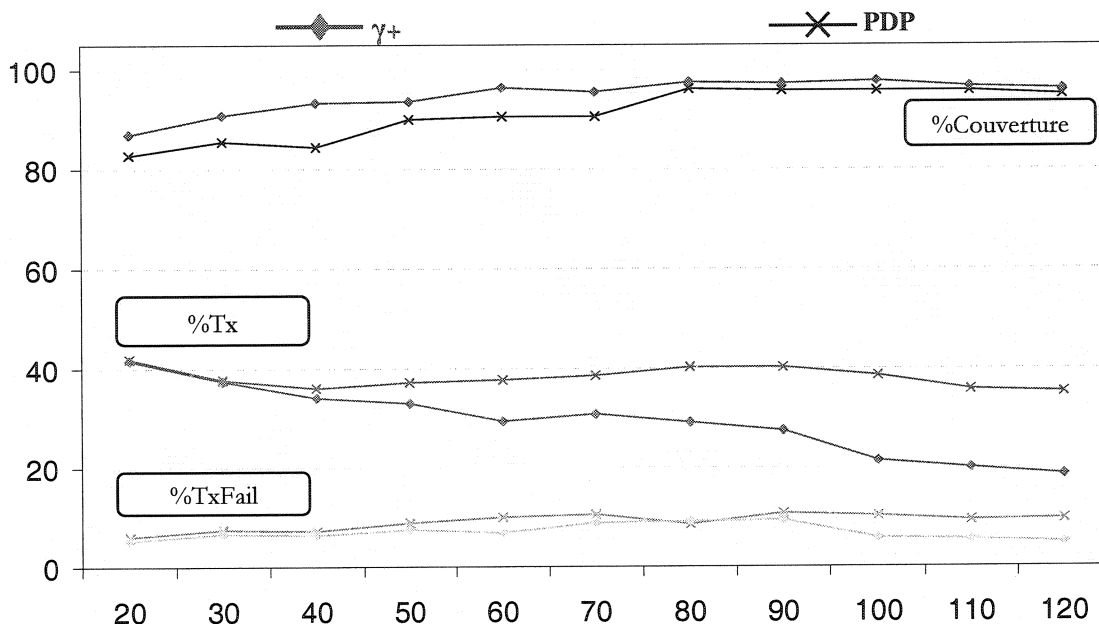


Figure 4.23 Les différents indicateurs de performance selon le nombre de mobiles dans le réseau des transmissions pour PDP et $\gamma+$

La Figure 4.23 présente les différents indicateurs de performance selon le nombre des mobiles formant le réseau. À cette figure, nous remarquons que les performances des deux algorithmes s'améliorent avec la densité du réseau. Cependant, nous pouvons constater une fois de plus que $\gamma+$ garantit de meilleures performances que le PDP, tous les critères confondus. Notamment, remarquons que lorsque la densité du réseau augmente, l'algorithme $\gamma+$ réduit énormément la charge du réseau, alors que le PDP semble la garder stable pour une couverture inférieure à celle obtenue par notre algorithme. Par conséquent, nous notons en général les bonnes performances de $\gamma+$ par rapport au PDP selon le nombre de mobiles dans le réseau.

La dernière étape dans cette étude de performance consiste à comparer les performances des deux algorithmes selon la puissance d'émission des mobiles. Pour ce faire, nous fixons la puissance des mobiles dans l'intervalle $[125, 350]$ et nous déroulons les séries de simulation en variant toutes les autres variables de décision. Les résultats obtenus sont résumés à la Figure 4.24.

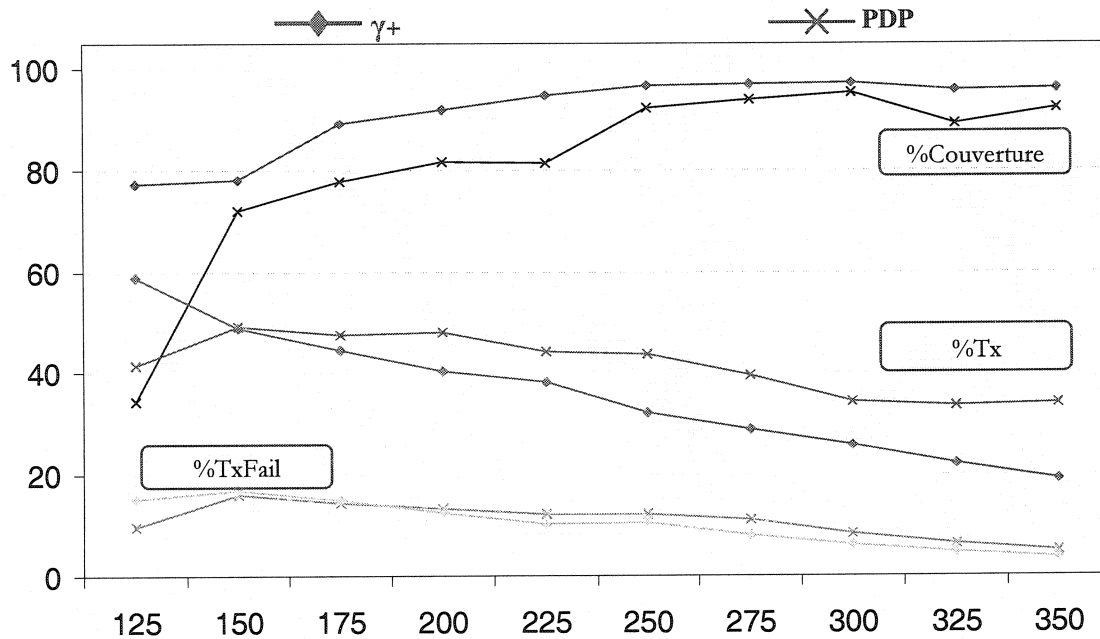


Figure 4.24 Les différents indicateurs de performance selon la puissance d'émission des mobiles exprimée par la couverture en m pour PDP et $\gamma+$

À la Figure 4.24, nous pouvons constater que les deux algorithmes donnent de meilleurs résultats pour de grandes valeurs de la puissance. Par contre, nous notons que l'algorithme PDP est plus vulnérable à la variation de la puissance, ce qui confirme le fait qu'il soit moins stable que $\gamma+$. En effet, pour de petites valeurs de la puissance (125 m), l'algorithme PDP n'assure que 30% de couverture du réseau alors que $\gamma+$ offre en moyenne 78% de couverture. Notons également que notre algorithme donne de meilleurs résultats pour les trois différents critères de performance et ce, quelle que soit la valeur de la puissance des mobiles.

CHAPITRE 5

CONCLUSION

5.1 Synthèse des travaux

Dans le cadre de notre recherche, nous nous sommes intéressés aux problèmes de la modélisation et du *flooding* dans les réseaux mobiles Ad-hoc. Dans un premier temps, nous avons présenté les inconvénients et les limitations de l'approche classique de modélisation des réseaux Ad-hoc. Entre autres, nous avons montré que l'approche classique ne permet pas de tenir compte des aspects d'incertitude et de volatilité inhérents à ces réseaux. Nous avons ainsi conçu et proposé une nouvelle approche basée sur le concept des ensembles flous. Cette approche consiste principalement à construire autour de chaque mobile un ensemble de voisins flous avec des fonctions d'appartenances adéquates. Cette approche permet en fait de rallier les aspects de la couche physique à la modélisation de ces réseaux. Ce rapprochement s'avère indispensable lors de la modélisation ou du développement de solutions pour des problèmes de type applicatif par exemple, surtout quand la couche physique n'offre pas de garantie de livraison de paquets. Notre approche fournit également un modèle de base pour introduire d'une manière simple et naturelle les aspects de la couche physique sans introduire d'énormes calculs.

Dans un second temps, nous avons abordé le problème du *flooding* dans les réseaux Ad-hoc. Nous avons d'abord défini le *flooding* puis montré la pertinence et la nécessité d'en offrir une bonne optimisation. Pour ce faire, nous avons commencé par proposer une modélisation du problème en tenant compte de l'incertitude liée aux phénomènes de la propagation radio et à la quantification de la qualité des liens radio dans ces réseaux. Tenir compte de ces problèmes physiques lors de la modélisation du *flooding* s'avère indispensable et se justifie par le fait que la couche physique n'offre pas

de garantie de livraison de paquets pour le *flooding*. En effet, comme on le sait, les paquets de *flooding* ne sont ni acquittés, ni retransmis. Ensuite, nous avons proposé un algorithme glouton de *flooding* appelé $\gamma+$ pour le résoudre. Cet algorithme basé sur la définition d'un critère γ pour l'évaluation de la fiabilité d'un lien radio se déroule en deux phases. La première consiste à déterminer à l'aide du critère γ un ensemble de voisins dit critiques pour lesquels nous utilisons les plus courts chemins, soit les plus fiables afin de leur acheminer les paquets de *flooding*. La deuxième est basée sur un mécanisme de sélection d'ensembles de recouvrement minimum et utilise une adaptation de l'algorithme PDP de *flooding* (Wei et Lu, 2002).

L'implémentation a été effectuée sous Visual C++.NET. Dans cet environnement nous avons considéré un ensemble de nœuds mobiles évoluant dans l'espace libre et avons testé le *flooding* en utilisant les deux algorithmes $\gamma+$ et PDP. Notons que considérer la propagation dans l'espace libre présente une simplification de l'algorithme $\gamma+$ proposé dans la forme la plus générale. En effet, considérer la propagation radio dans l'espace libre réduit l'incertitude relative à la quantification de la qualité des liens radio, à la présence physique de voisins.

Plusieurs séries de tests ont été effectuées afin d'étudier les performances de l'algorithme $\gamma+$ et le comparer au PDP. Trois mesures de performances pertinentes ont été retenues à savoir : la couverture du réseau ou le pourcentage de nœuds qui reçoivent le paquet de *flooding*, la charge du réseau ou le nombre de retransmissions nécessaires, et les pertes de paquets dans le réseau. D'une manière globale, nous avons remarqué que $\gamma+$ assure de meilleures performances que le PDP et ceci, quel que soit le critère de performance considéré. En effet, $\gamma+$ garantit une meilleure couverture tout en réduisant la charge du réseau et les pertes de paquets. Nous avons aussi noté que $\gamma+$ est à peu près deux fois plus stable que le PDP, ce qui constitue un apport considérable. Par ailleurs, nous avons également noté que $\gamma+$ préserve ses performances même pour des vitesses de mobiles élevées, alors que le PDP tend à perdre en performance. D'un autre côté, les performances des deux algorithmes s'accroissent avec la densité du réseau et nous constatons encore une fois que, quelle que soit la densité, $\gamma+$ est meilleur que le PDP. En

conséquence, nous notons les meilleures performances de l'algorithme $\gamma+$ par rapport au PDP, ce qui est certainement dû au fait de considérer la nouvelle approche qui rallie les aspects de la couche physique à la résolution de problèmes issus des couches supérieures, notamment routage, qualité de service et application. Nous avons également vu que l'algorithme $\gamma+$ définit un critère γ de tolérance aux pertes de paquets qui, quand il est serré assure une meilleure couverture. Pour ce faire, notre algorithme tend à augmenter la charge du réseau en utilisant en effectuant plus de retransmissions de paquets afin d'assurer la couverture. En fait, ce critère agit comme un critère de tolérance à la livraison des paquets de *flooding*.

5.2 Limitations des travaux

Au terme de notre recherche, nous notons deux limitations de nos travaux dont une touche l'application de la nouvelle approche de modélisation des réseaux Ad-hoc et l'autre concerne l'implémentation et les résultats.

La première touche essentiellement l'approche floue. En effet, malgré que cette approche permet, de manière simple et efficace, de modéliser l'incertitude chose qui entraînerait beaucoup de calcul si on devait utiliser une autre approche, probabiliste par exemple, elle repose essentiellement sur la construction des fonctions d'appartenance ce qui n'est pas tout le temps évident. En fait, la mise en œuvre de fonctions d'appartenance adéquates repose principalement sur l'expertise que l'on a du domaine d'application. Ainsi, dans le cas de la propagation radio, nous devons disposer d'études du problème de la propagation dans l'étendue du terrain où évolue le réseau Ad-hoc.

La deuxième limitation découle de l'implémentation. En effet, nous avons implémenté et étudié les performances des deux algorithmes dans le cas d'une propagation dans l'espace libre en faisant abstraction de la couche MAC. Nous avons noté de très bonnes performances pour $\gamma+$. Ces performances devraient en réalité se voir baisser si on tient compte de la couche physique. Par contre, elles resteraient inchangées si on fait la comparaison par rapport au PDP dans les mêmes conditions.

5.3 Orientations et travaux futurs

Compte tenu des limitations recensées ci haut, nous distinguons deux principales orientations pour les travaux futurs. La première serait de tester notre algorithme dans un cas de propagation réel où l'on disposerait d'une série de valeurs pour les facteurs de propagations caractéristiques du terrain de l'expérience. Dans le même sens, nous pourrions essayer de modéliser et d'inclure à notre approche de modélisation l'incertitude reliée au phénomène de la collision, ainsi qu'à la vraisemblance de l'information du voisinage au fil du temps. En effet, ceci nous permettrait de créer un critère γ plus fiable pour la sélection des nœuds et liens radio critiques.

Par ailleurs, une amélioration technique de notre algorithme pourrait être effectuée au niveau du calcul des plus courts chemins. En effet, l'algorithme de plus courts chemins, de par sa définition, fait grandement la différence entre un chemin de longueur 0.152 et 0.153 par exemple, alors qu'en terme de puissance radio, ces deux nombres seraient pratiquement les mêmes. Il serait ainsi très souhaitable si l'on peut considérer une certaine fragmentation du domaine de définition de la puissance de sorte que notre définition des nœuds et liens radio critiques soit plus réaliste. Ceci nous permettrait même de réduire encore plus la charge du réseau du moment que celle-ci dépend étroitement du calcul des plus courts chemins et de la sélection des nœuds critiques.

BIBLIOGRAPHIE

Andersen J.B.; Rappaport T.S.; Yoshida S.; "Propagation measurements and models for wireless communications channels", *Communications Magazine, IEEE*, Vol. 33, No. 1, January 1995 pp. 42-49.

Boulmakoul A.; "Generalized path-finding algorithms on semirings and the fuzzy shortest path problem"; *Journal of Computational and Applied Mathematics*, Vol. 162, No. 1, January 2004, pp. 263-272

Chaudet C.; *Qualité de service et réseaux ad-hoc - un état de l'art*. Rapport de recherche de l'INRIA, November 2001, RR-number : 4325.

Chaudet C.; Lassous I.G.; "Routage QoS et réseaux ad hoc : de l'état de lien à l'état de nœud". Rapport de recherche de l'INRIA, January 2003, RR-number : 4700.

Chen S.; Nahrstedt K.; "Distributed quality-of-service routing in ad hoc networks". *IEEE Journal on Selected Areas in Communications*, Special issue on Wireless Ad Hoc Networks, Vol. 17, No. 8, August 1999, pp. 1488-1505.

Chiang C.C.; "Routing in clustered multihop, mobile wireless networks with fading channel". *Proc. IEEE SICON '97*, April 1997, pp.197-211.

Conan J.; "Radiocommunications mobiles". Notes de cours, École Polytechnique de Montréal, 2003.

Cox D.; Murray R.; Norris A.; "800 Mhz attenuation measured in and around suburban houses". AT&T Bell Laboratories Tech. J., Vol. 673, No. 6, 1984, pp. 921-954.

Dubois D.; Prade, H.; "Algorithme de plus courts chemins pour traiter des données floues". RAIRO / Operations Research, Vol. 2, No. 2, 1978, pp. 213-227.

Gerla M.; Lin C.R.; "MACA/PR : An asynchronous multimedia multihop wireless network". In IEEE INFOCOM 97, Japan, 1997.

Guha S.; Khuller S.; "Approximation algorithms for connected dominating sets". Algorithmica, Vol. 20, 1998, pp. 374-387.

Johnson D.B.; Maltz D.A.; "Dynamic source routing in Ad-hoc wireless networks". Mobile Computing, Imielinski T. and Korth H., Edission : Kluwer, 1996, pp. 153-181.

Lee S.B.; Ahn G.S.; Zhang X.; Campbell A.T.; "INSIGNIA : An ip-based quality of service framework for mobile ad hoc network". Journal of Parallel and Distributed Computing (Academic Press); Special issue on Wireless and Mobile Computing and Communications, Vol. 60, No. 4, 2000, pp.374 - 406.

Liberti, J.C.; Rappaport, T.S.;"Statistics of shadowing in indoor radio channels at 900 and 1900 MHz"; Military Communications Conference, 1992. MILCOM '92, Conference Record. 'Communications - Fusing Command, Control and Intelligence', IEEE, 11-14 Oct., Vol. 3, 1992, pp.1066 - 1070

Lim H.; Kim C.; "Flooding in Wireless Ad Hoc Networks". Computer Communications, Vol. 24, No. 3-4, 2001, pp. 353-363.

Lou W.; Wu J.; "On reducing broadcast redundancy in ad hoc wireless networks". IEEE Transactions on Mobile Computing, Vol. 1, No. 2, April-June 2002, pp.111-122.

Narasimhan R.; Cox D.C.; "Estimation of mobile speed and average received power in wireless systems using best basis methods". IEEE Transactions on Communications, Vol. 49, No. 12, December 2001, pp.2172-2183.

Ni S.-Y.; Tseng Y.-C.; Chen Y.-S.; Sheu J.-P.; "The broadcast storm problem in a mobile ad hoc network". ACM MOBICOM '99, August 1999, pp. 153-167.

Perkins C.E.; Bhagwat P.; "Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers". Proceeding Conference on Communications, Architectures, Protocols, and Applications, September 1994, pp. 234-244.

Perkins C.; Royer E.M.; "Ad-Hoc On-Demand Distance Vector Routing". Proc. Second IEEE Workshop Mobile Computing Systems and Applications (WMCSA), February 1999, pp. 90-100.

Perkins C.; Royer E.M.; Chakeres I.D.; "Ad-Hoc On-Demand Distance Vector Routing". Mobile Ad Hoc Networking Working Group – Internet Draft, January 2004.

Pierre S.; "Réseaux et systèmes informatiques mobiles: Fondements, Architectures et Applications". Presses Internationales Polytechnique (PIP), April 2003.

Pierre S.; Couture M.; "Télécommunications et transmission de données". Éditions Eyrolles, Paris, 1992.

Rappaport T.S.; "Wireless Communications: Principles and Practice". Prentice Hall Edition, December 2001.

Royer E.M.; Toh C.K.; "A Review of Current Routing Protocols for Ad Hoc Mobile Wireless Networks," IEEE Personal Communications, Vol. 6, No. 2, 1999, pp. 46-55.

Seidel S.Y.; Rappaport T.S.; Jain S.; Lord M.L.; Singh R.; "Path loss, scattering and multipath delay statistics in four European cities for digital cellular and microcellular radiotelephone" IEEE Transactions on Vehicular Technology, Vol. 40, No. 4, November 1991, pp. 721-730.

Sinha P.; Sivakumar R.; Bharghavan V.; "CEDAR : a core extraction distributed ad hoc routing algorithm". IEEE Journal on Selected Areas in Communications, special issue on Wireless Ad Hoc Networks, Vol.17, No. 8, August 1999, pp. 1454-1465.

Sklar B.; "Rayleigh fading channels in mobile digital communication systems: I. Characterization". IEEE Communications Magazine, Vol. 35, No. 7, July 1997, pp. 90-100.

Sklar B.; "Rayleigh fading channels in mobile digital communication systems: II. Mitigation". IEEE Communications Magazine, Vol. 35, No. 7, July 1997, pp. 102-109.

Wu K.; Harm J.; "QoS Support in Mobile Ad Hoc Networks". Crossing Boundaries- the GSA Journal of University of Alberta, Vol. 1, No. 1, November 2001, pp. 92-106.

Xiao H.; Seah W.K.G.; Lo A.; Chua K.C.; "A flexible quality of service model for mobile ad hoc networks". IEEE Vehicular Technology Conference, Tokyo, Japan, Mai 2000, pp. 445-449.