Université Libre de Bruxelles
Faculté des Sciences Appliquées

Année académique
2005-2006

# Hybridation entre les modes ad-hoc et infrastructure dans les réseaux de type Wi-Fi

*Promoteur du Mémoire :*
Pr. Esteban Zimanyi
*Co-promoteur :*
Jean-Michel Dricot

Mémoire de fin d'études présenté par
Van der Meerschen Jérôme en vue de l'obtention
du grade d'Ingénieur Civil Informaticien
en Sciences Appliquées.

# Remerciements

Un MFE est une entreprise longue et périlleuse (plus pour certains que pour d'autres), au final ce n'est pas un projet que l'on réalise si l'on est pas entouré.

Je tiens à remercier tout particulièrement le professeur Esteban Zimanyi ainsi que M. Jean-Michel Dricot pour l'aide et les conseils qu'ils m'ont apportés tout au long de ce travail. Je voudrais aussi remercier mes compagnons d'infortune pour cette année qui restera dans toutes nos mémoires (et dans tous nos mémoires ! ! !). Je profite aussi de l'occasion pour dire un grand merci à ma "petite" famille, et tout particulièrement ma maman, pour leur soutien tout au long de ces années.

Je ne peux pas terminer ces remerciements sans te remercier, toi, pour tout ce que tu m'apportes quotidiennement, merci.

# Table des matières

# Table des figures

# Liste des tableaux

# Première partie

# État de l'art

# Chapitre 1

# Introduction

Derrière ce titre complexe se cache un concept relativement simple : la combinaison de différents types de réseaux. Dans notre cas, il s'agit des deux modes de fonctionnement des réseaux WiFi : infrastructure et ad-hoc. Le mode infrastructure est celui que tout le monde connaît, dans lequel les ordinateurs du réseau communiquent via un point d'accès. Le mode ad-hoc quant à lui permet aux machines de s'envoyer des messages directement, sans passer par une quelconque infrastructure.

## 1.1 Contexte

Les communications sans fils prennent de plus en plus d'importance dans la société actuelle, la plupart de nos habitudes se basent sur ces communications, que nous en soyons conscients ou non. En effet, tout un chacun se sert d'un GSM et les sociétés investissent de plus en plus dans des réseaux sans fils.

Les réseaux sans fils fonctionnant en mode ad-hoc sont flexibles et faciles à déployer. Ces propriétés en font de parfaits candidats pour le futur des réseaux cellulaires et des WLAN. De plus, avec l'arrivée de systèmes *wireless* intégrant diverses technologies, comme la 4G des réseaux cellulaires, l'interconnexion de MANETs avec les réseaux fixes est un des domaines de recherche les plus en vogue du moment. Dans ce type de réseaux, dits hybrides, les noeuds mobiles peuvent êtres utilisés comme une extension de l'infrastructure existante. Certains des noeuds ad-hoc ont le rôle de *gateway* et sont utilisés comme relais entre les noeuds mobiles et le réseau fixe. Le challenge dans ces cas d'interconnexion entre des réseaux ad-hoc et Internet

est de réussir à informer les noeuds mobiles de la présence des *gateways* sans gaspiller les ressources du réseau.

Dans les réseaux ad-hoc, les recherches actuelles sont dirigées vers les algorithmes de routage. En effet, la plupart des protocoles existants ne se préoccupent que d'un paramètre pour trouver le chemin d'une source vers une destination : la longueur du chemin. Or, spécialement dans des situations *indoor*, cette mesure ne traduit pas l'efficacité du chemin. Une des solutions possibles pour tenir compte de la qualité des routes est de se baser sur les informations provenant de la couche physique comme, par exemple, la puissance d'émission ou la valeur du SNR.

## 1.2 Objectifs

Afin de voir comment il serait possible de combiner des réseaux ad-hoc et des réseaux avec infrastructure, il faut comprendre les différents mécanismes à la base des réseaux sans fils et les caractéristiques de ces types de réseaux. Nous devons notamment comprendre les différents algorithmes de routage afin de parvenir à identifier la meilleure approche du point de vue de la transmission des paquets.

Notre objectif est de modifier un protocole de routage existant afin de prendre en compte l'intégration des deux types de réseaux. Nous voulons donc un protocole permettant à un réseau ad-hoc de communiquer avec Internet, par exemple. De plus, nous voulons parvenir à un protocole ayant un bon comportement en *indoor*. Pour cela, nous allons nous baser sur la puissance nécessaire pour transmettre sur les différents liens.

L'implémentation de ce protocole sera faite dans un simulateur de réseau appelé NS2. Ce simulateur permet de modifier rapidement les protocoles existants. Il permet aussi de lancer facilement diverses simulations qui nous permettrons de tester notre protocole afin de vérifier que les apports effectués ont porté leurs fruits.

## 1.3 Approche

Nous commencerons par décrire, dans le chapitre 2, les principes propres aux communications sans fil comme les aspects de propagation et les différentes sources d'affaiblissement.

Dans le chapitre 3, nous parlerons des réseaux sans fils proprement dits. Nous introduirons les réseaux cellulaires et détaillerons les réseaux d'ordinateurs. A la fin de ce chapitre, nous

parlerons des réseaux ad-hoc, en expliquant leurs avantages et inconvénients. Nous parlerons aussi du routage dans ce type de réseaux, en donnant des exemples de protocoles existants.

Le chapitre 4 traite de l'intégration de différents réseaux, en donnant les intérêts de ces combinaisons. Certaines solutions existantes y seront également introduites.

Au chapitre 5, nous présenterons nos choix, en expliquant pourquoi nous les avons faits. Nous donnerons les grandes lignes de l'implémentation effectuée dans le simulateur NS2.

Le chapitre 6 reprend les simulations qui ont été effectuées ainsi que les résultats et conclusions qui en découlent.

# Chapitre 2

# Communications sans fil

Dans tout système de transmission de données, on définit le support de transmission comme le chemin physique entre l'émetteur et le récepteur. Ce support, aussi appelé média, peut être :

- guidé : le long d'un support physique (câble en cuivre, fibre optique,...)
- non guidé : la transmission se fait à travers le milieu ambiant et nécessite l'emploi d'antennes.

Parmi les différentes plages de fréquences utilisables par les communications sans fils, c'est celle correspondant aux ondes radio (30 MHz - 1 GHz) qui est la plus utilisées dans les réseaux sans fils.

## 2.1 Ondes radio

Les ondes radio sont omnidirectionnelles. Les ondes électromagnétiques sont donc propagées dans toutes les directions et peuvent être captées par de nombreuses antennes. La transmission de tels signaux ne requiert donc pas d'alignement spécifique des antennes. La plage de fréquences 30 MHz - 1 GHz est très efficace pour une diffusion omnidirectionnelle. En effet, l'atmosphère ne réfléchit pas les ondes de fréquence supérieure à 30 MHz (et les communications via satellite sont donc peu perturbées). De plus, ces fréquences ne sont que peu atténuées par la pluie.

Les interférences les plus importantes, pour ce type de signaux, résultent de la propagation multitrajet, c'est à dire la création de différents chemins de propagation par la réflexion des

signaux sur le sol, l'eau et les différents obstacles. A chaque plage de fréquences correspondent un ou plusieurs modes de propagation. Nous nous intéresserons à des ondes de fréquence supérieure à 30 MHz, on parle alors de propagation en vue directe.

## 2.2 Propagation en vue directe

Dans ce type de transmission, l'antenne émettrice et l'antenne réceptrice doivent être situées en vue directe l'une de l'autre, comme illustré à la figure 2.1.

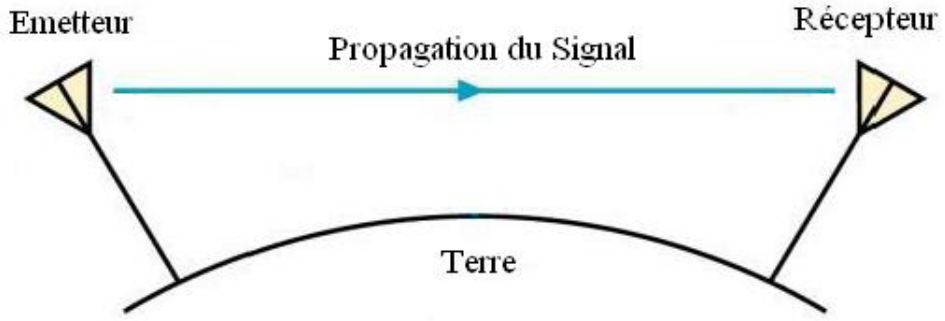


FIG. 2.1 – Propagation en vue directe

Le signal reçu sera différent du signal émis étant données les dégradations qui se produisent durant la transmission. Pour une communication sans fils en vue directe, les sources de perturbation sont [1] :

1. Affaiblissement :
   La puissance d'un signal décroît avec la distance. Pour un média sans guide physique, l'atténuation est fonction de la distance et des conditions atmosphériques. Cet affaiblissement est plus important aux hautes fréquences ce qui entraîne des distorsions plus importantes.
2. Affaiblissement en espace libre :
   Dans un système de transmission sans fils le signal se disperse avec la distance, ce qui conduit à une diminution de puissance supplémentaire. Cet effet s'ajoute à l'affaiblissement et une antenne recevra donc moins de puissance quand elle est éloignée de l'émetteur. On peut calculer cet affaiblissement grâce à :

$$PL = UnitLoss + 10nlog(d) \tag{2.1}$$

où :
$PL$ = Path Loss,
$UnitLoss$ = puissance perdue à 1m de distance (30dB),
$n$ = constante de path loss,
$d$ = distance par rapport à l'antenne.

3. Le bruit :
   Pendant le trajet, le signal est altéré par des signaux perturbateurs, provenant par exemple d'autres antennes. Ce sont ces signaux qui sont appelés bruit. Ce bruit peut être réparti en plusieurs catégories :

   **le bruit thermique :** provoqué par l'agitation des électrons, il est présent dans tous les équipements électroniques.

   **le bruit d'intermodulation :** résultant du partage d'un média par des signaux de fréquences différentes.

   **la diaphonie :** il s'agit d'un couplage perturbateur de trajets de signaux voisins. L'exemple classique, que tout un chacun a déjà expérimenté lors d'une communication téléphonique, est de percevoir une autre conversation.

   **le bruit impulsif :** bruit changeant, apparaissant sous forme de pics irréguliers et dont les causes sont diverses. Il peut provenir de défauts internes à l'antenne ou de perturbations extérieures comme la foudre.

4. Absorption atmosphérique :
   La vapeur d'eau et l'oxygène sont deux éléments intervenant fortement dans l'affaiblissement d'un signal, en absorbant une partie de celui-ci.

5. Propagation multitrajet :
   Rappelons les trois effets de propagation [2] qui entrent en ligne de compte lors d'une transmission et dont la représentation peut être trouvée à la figure 2.2 :
   – La réflexion : survient lorsqu'une onde rencontre une surface qui est plus grande que sa longueur d'onde.

– La diffraction : se produit quand l'onde frappe le coin d'un obstacle plus grand que sa longueur d'onde. Des ondes se propagent alors dans différentes directions à partir de ce coin.
– La dispersion : a lieu quand la taille de l'objet est de l'ordre de la longueur d'onde du signal. Celui-ci est alors dispersé en plusieurs signaux plus faibles.

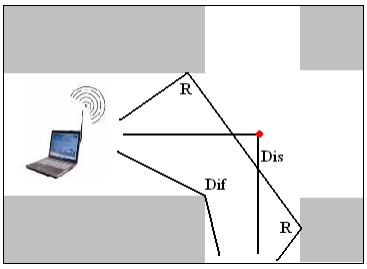


FIG. 2.2 – Effets de propagation : réflexion (R), dispersion (Dis), diffraction (Dif).

Dans la plupart des cas, on trouve une multitude d'obstacles entre l'émetteur et le récepteur. Le signal peut donc être réfléchi un grand nombre de fois et plusieurs copies du signal original peuvent exister. Le récepteur capte alors un signal qui est la résultante du signal principal et de tous les signaux réfléchis qui sont captés par son antenne. Le signal peut être renforcé ou atténué (voir même annulé) par ces différentes composantes et les effets de la propagation multitrajet ont donc une importance considérable sur les transmissions sans fils.

En plus des perturbations dont nous venons de discuter, il y a un autre problème qui surgit dans un environnement mobile, c'est le phénomène d'évanouissement (*fading*). Ce terme désigne la variation dans le temps de la puissance du signal reçu, due à des changements dans le support ou dans le chemin de transmission emprunté.

## 2.3 Evanouissement

Quand une des deux antennes se déplace par rapport à l'autre, l'emplacement relatif des obstacles entre elles est modifié et l'effet de ces obstacles sur la transmission évolue. Comme nous l'avons vu, ces phénomènes vont participer à la propagation multitrajet et avoir une influence constructive ou destructive sur le signal. Le déplacement du mobile va donc induire une variation dans l'apparition des signaux secondaires et influe donc sur le signal reçu.

## 2.4 Propagation indoor

Dans le cadre d'une communication *indoor* la situation est empirée : les murs, le plafond et le sol participent à la réflexion et à la dispersion. Les coins des couloirs participent, eux, à la diffraction, sans oublier les meubles qui jouent aussi un rôle. La propagation multitrajet devient plus importante, ce qui rend les transmissions beaucoup plus difficiles en *indoor*.

De plus, le mouvement des personnes à l'intérieur des bâtiments augmente également la propagation multitrajet. La distance de transmission maximale est donc nettement réduite et sans l'utilisation de relais, il est impossible de couvrir un bâtiment tout entier avec un seul émetteur.

Dans le cas où la station émettrice se situe en dehors du bâtiment où se trouve le récepteur, il faut prendre en compte la pénétration du signal dans le bâtiment. Celle-ci dépend de plusieurs facteurs :

- Les matériaux de construction employés.
- L'orientation du bâtiment.
- La hauteur du bâtiment. En effet la qualité du signal est meilleure en hauteur car les interférences urbaines sont moindres.
- Le pourcentage de fenêtres. La perte de signal à travers une vitre est plus faible qu'à travers un mur. La différence se chiffre aux alentours de 6dB.
- La fréquence de transmission : les pertes dues à la pénétration diminuent avec l'augmentation de la fréquence.

Maintenant que nous sommes familiarisés avec les principes à la base des communications sans fils, nous pouvons nous intéresser au réseaux sans fils à proprement parler.

# Chapitre 3

# Architecture des réseaux cellulaires et sans fils

Un réseau sans fils est un réseau dans lequel les différents éléments participants ne sont pas raccordés entre eux par un média physique. Ceci permet aux utilisateurs de se déplacer dans un périmètre de couverture. La transmission des données se fait via les ondes hertziennes avec les caractéristiques que nous avons détaillées précédemment.

L'avantage d'un tel réseau est, outre la mobilité de ses utilisateurs, qu'il ne nécessite pas de travaux de câblage et les coûts de mise en place sont donc réduits. Cependant, il existe encore certains problèmes dus à la difficulté de contrôle de la propagation des signaux, qui conduit à des pertes ou à une portée assez faible. De plus, il faut respecter les réglementations relatives aux transmissions radio. Nous allons maintenant introduire les réseaux cellulaires, avant de passer aux réseaux WiFi.

## 3.1 Réseaux cellulaires

Les réseaux cellulaires [3] sont, comme leur nom l'indique, divisés en cellules (voir figure 3.1). Chacune d'elles est une zone géographique dont les points peuvent être atteints à partir d'une même antenne. Chaque cellule possède une station de base, BS (*Base Station*), assurant la couverture radio au sein de la zone. Chaque BS est connectée à une BSC (*Base Station Controller*) dont le rôle consiste à gérer les admissions des mobiles, l'allocation des canaux et le *handover* (HO). Ce dernier désigne le changement de cellule au cours d'une communication

et implique donc le réacheminement de l'information vers une nouvelle cellule.

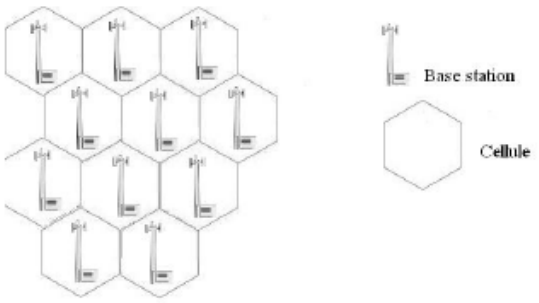


FIG. 3.1 – Réseaux cellulaires

Une certaine plage de fréquences est attribuée à chaque cellule, avec comme contrainte que des cellules adjacentes disposent de plages différentes. Cela permet de réduire les interférences et la diaphonie. Cependant, deux cellules éloignées peuvent avoir la même bande de fréquences. En effet, quand la distance entre deux cellules augmente, les interférences qu'il peut y avoir entre elles diminuent.

Le choix de la distance minimale entre deux cellules utilisant la même bande est une caractéristique essentielle dans la conception d'un réseau cellulaire. L'organisation des cellules peut suivre divers regroupements appelés motifs (voir figure 3.2).

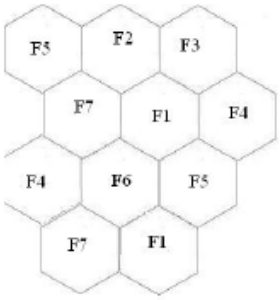


FIG. 3.2 – Réutilisation de fréquence dans les réseaux cellulaires

Au cours du temps, plusieurs technologies de communication ont existé et ont donné naissance à différentes familles de réseaux cellulaires, appelées génération.

### 3.1.1 Première Génération, 1G

La première génération est apparue à la fin des années 70 et était caractérisée par des terminaux analogiques dotés d'une faible mobilité et de services limités. En effet, en dehors du téléphone, les services fournis étaient pratiquement inexistants.

A cause du coût élevé de leur infrastructure, ces réseaux n'ont eu que peu de succès. Ils ont malgré tout été les premiers à permettre à un utilisateur mobile de communiquer de façon continue.

### 3.1.2 Deuxième Génération, 2G

Le passage à la deuxième génération s'est fait avec l'introduction de la technologie numérique. C'est cette famille qui a permis l'essor des téléphones mobiles, notamment grâce à la technologie GSM (*Global System for Mobile communication*).

Dans les systèmes GSM, le terminal mobile comprend deux parties : celle permettant la communication radio, et celle responsable de l'identification de l'abonné. La mobilité est généralement gérée grâce à deux bases de données : le HLR (*Home Location Register*) qui garde à jour les données de l'abonné et le VLR (*Visitor Location Register*), qui gère le client dans la cellule où il se trouve.

Pour permettre à d'avantage d'utilisateurs d'être connectés à une station de base sans saturer le réseau, deux modes d'accès sont utilisés :

- TDMA (*Time Division Multiple Acces*), illustré à la figure 3.3, pour lequel le temps est divisé en tranches auxquelles un seul terminal peut accéder. Une fréquence peut donc être utilisé par plusieurs abonnés simultanément.
- FDMA (*Frequency Division Multiple Access*), illustré à la figure 3.4,découpage en bande de fréquences de manière à attribuer une partie du spectre à chaque utilisateur. De cette manière, chaque utilisateur se voit attribuer une bande de fréquences distincte.

Les spécifications GSM permettent un transfert de données avec un débit supérieur à celui offert par la première génération mais celui-ci reste trop faible (moins de 10kbits/s) pour envisager l'arrivée de services multimédia.

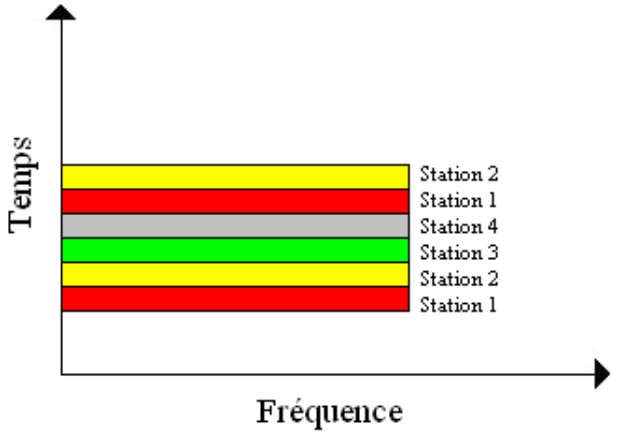


FIG. 3.3 – TDMA

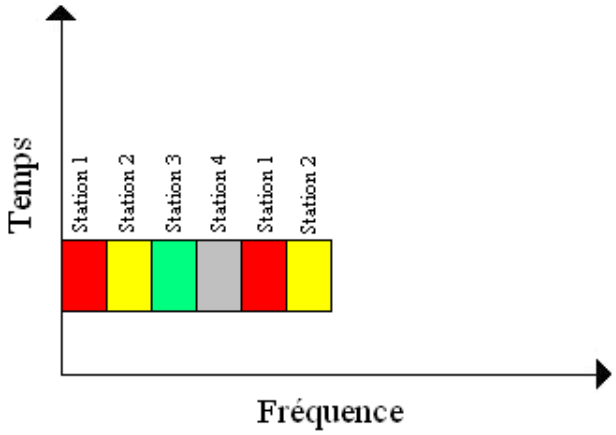


FIG. 3.4 – FDMA

### 3.1.3 Deuxième Génération et demi, 2,5G

De nouveaux systèmes étaient donc nécessaires afin de prendre en compte les besoins des utilisateurs. La transition entre 2G et 3G s'est faite en passant par une génération intermédiaire dans laquelle le réseau est séparé en deux sous réseaux : un réseau pour la parole et un réseau pour le transfert des données sous forme de paquets. Cette génération pouvait donc se baser sur l'infrastructure déjà en place pour la deuxième génération, en y ajoutant le réseau pour le transfert de paquets. La technologie 2,5G utilisée en Europe est le GPRS (*General Packet Radio Service*) qui est une évolution du GSM ayant un débit jusqu'à huit

fois plus élevé que celui de son prédécesseur.

### 3.1.4 Troisième Génération, 3G

Cette troisième génération vise donc à apporter de nouveaux services à la précédente mais sa mise en place est complexe car elle demande une mise à jour des infrastructures.

Ces réseaux sont caractérisés par des débits plus importants et ils doivent être capables de supporter des services audio, vidéo, texte, etc. essentiels aux appels multimédia. Il faut donc un système intégrant plusieurs environnements fonctionnant de concert pour offrir des services adéquats à l'utilisateur.

La norme européenne de la 3G est l'UMTS (*Universal Mobile Telecommunication System*). Elle est caractérisée par une technique d'accès W-CDMA (*Wideband-Code Division Multiple Acces*). Le CDMA (voir figure 3.5) est une autre technique utilisée afin de faire passer plusieurs canaux sur la même fréquence porteuse. Elle consiste à étaler le spectre afin de faire passer une information supplémentaire : un code alloué à chaque communication.

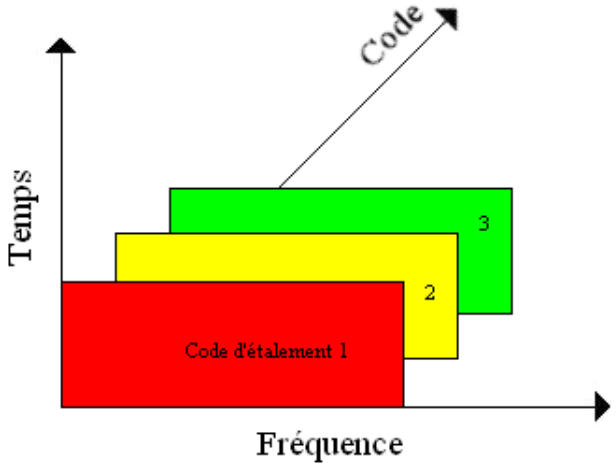


FIG. 3.5 – CDMA

L'élargissement du spectre a débouché sur le W-CDMA. De plus, l'UMTS a un débit pouvant atteindre les 2Mbits/s (en faible mobilité), vitesse nettement supérieure au débit de base des GSM.

### 3.1.5 Quatrième Génération, 4G

La 3G est à peine sur le marché que l'on parle déjà de son successeur. Les raisons de cette discussion sont doubles [4] : d'une part, la prolifération des réseaux sans fils est très importante ces dernières années mais on réalise qu'il y a une certaine anarchie dans l'utilisation des différentes technologies précitées. De plus, avec l'évolution du tout en un, par exemple l'intégration des PDA (*Personal Digital Assistant*)dans les téléphones mobiles et vice-versa, plusieurs technologies sans fils (WLAN ou Bluetooth) sont intégrées dans le même appareil sans pour autant offrir un usage optimal de ces différents réseaux. Le développement de la 4G devrait régler ces problèmes en offrant une compatibilité non seulement d'accès mais également de *roaming*, nom donné au passage d'un réseau de communication sans fils à un autre, en toute transparence.

D'autre part, les prédictions veulent qu'aux alentours de 2010 le maximum d'utilisateurs et de bande passante de la 3G sera atteint et il faudra donc passer au niveau supérieur. Pour cela, une des possibilités est d'utiliser une technique d'accès OFDM (*Orthogonal Frequency Division Multiplexing*) qui consiste à diviser sur un grand nombre d'ondes porteuses le signal numérique que l'on veut transmettre.

La 4G sera donc dotée d'une bande passante plus importante, dans le but d'offrir toujours plus de services aux utilisateurs, comme par exemple un accès rapide à Internet depuis son téléphone mobile. De plus, la 4G sera un réseau complet : le réseau cellulaire sera entièrement rattaché à Internet et aux réseaux WiFi. Elle impliquera donc l'hybridation de deux réseaux différents.

Un réseau supportant un grand nombre d'utilisateurs nécessitera une bonne organisation des ressources. En effet, malgré une bande passante importante, celle-ci restera limitée et il se pourrait que certains services soient surchargés. Une des possibilités pour augmenter la fluidité est que les terminaux soient capables de faire transiter l'information entre eux, sans passer par la station, en formant donc un réseau ad-hoc (voir section 3.3). Nous voyons ici une des grandes applications de ce mémoire, car la 4G sera basée sur la combinaison d'un réseau infrastructure (le réseau cellulaire) et d'un réseau ad-hoc.

## 3.2 Réseaux sans fils

Le terme réseau sans fils désigne des réseaux d'ordinateurs communicant sans support filaire.
Il existe différents types de réseaux, caractérisés par la taille de leur zone de couverture (voir figure 3.6) :

- *Body Area Network* (BAN) : Réseaux sans fils de très faible portée, limitée à la taille du corps humain.
- *Personal Area Network* (PAN) : Catégorie de réseaux à faible portée dont le but est de relier différents périphériques à une unité centrale.
- *Local Area Network* (LAN) : Portée d'environ 100m, correspondant à un réseau d'entreprise. Leur but est de connecter plusieurs machines situées dans un périmètre restreint.
- *Metropolitan Area Network* (MAN) : Couverture de l'envergure d'un campus, dans le but d'interconnecter différents réseaux fixes ou sans fils.
- *Wide Area Network* (WAN) : Parfois appelé "réseau cellulaire mobile", s'étendant sur plusieurs kilomètres.

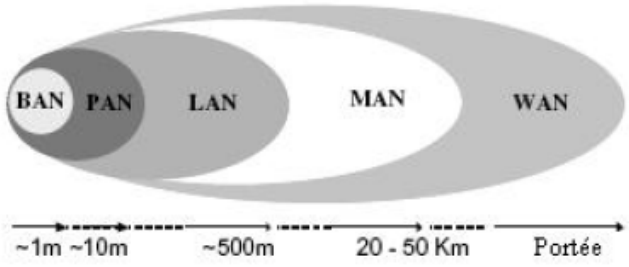


FIG. 3.6 – Types de réseaux sans fils [5]

Ces différentes appélations sont également valables pour les réseaux sans fils, en y ajoutant le préfixe *Wireless*. Ils deviennent alors WBAN, WPAN, WLAN, WMAN, WWAN. Il y a différentes technologies utilisées dans le monde sans fils. Les plus importantes sont :

- Bluetooth : Offre un débit d'environ 1Mbps pour une portée d'une trentaine de mètres.
- Hyperlan2 : Technologie WLAN, c'est une norme européenne offrant un débit de 54Mbps et une zone de couverture d'environ 100 mètres.
- WiFi (*Wireless Fidelity*) : Autre technologie WLAN, offrant les même débit que l'Hyperlan2 mais une portée de plusieurs centaines de mètres.

Dans le cadre de ce mémoire, nous nous intéressons à la technologie WiFi, norme de la WECA (*Wireless Ethernet Compatibility Alliance*), que nous allons détailler par la suite.

### 3.2.1 Réseaux IEEE 802.11

#### Modèle OSI

Le modèle OSI (*Open System Interconnect*) est un modèle de référence pour les réseaux informatiques. Ce modèle décrit les concepts et la démarche à suivre pour normaliser l'interconnexion de plusieurs systèmes. Ce modèle est constitué de sept couches effectuant chacune une tâche bien définie et fournissant des services à ses couches voisines, voir figure 3.7.

1. Application : comprend les programmes d'application qui utilisent le réseau. Comme la messagerie électronique, ou le transfert de fichiers.
2. Présentation : est destinée à supporter les fonctions dont beaucoup de programmes ont besoin, comme la compression de texte ou la conversion d'images graphiques.
3. Session : gère les connexions entre les applications.
4. Transport : assure un contrôle de bout en bout, en permettant à un processus destinataire de communiquer directement avec le processus source
5. Réseau : définit l'unité de données de base transférée sur le réseau entre deux sites extrêmes et inclut les concepts d'adressage et de routage.
6. Liaison de données : s'occupe de l'acheminement de trames de données entre deux équipements voisins.
7. Physique : s'occupe de la connexion physique d'une machine avec le réseau.

Un autre modèle existant est le modèle TCP/IP. Les deux modèles présentent des similitudes mais diffèrent au niveau du nombre de couches. Comme illustré à la figure 3.7 il n'y a que 5 couches dans le modèle TCP/IP. Les trois couches supérieures du modèle OSI ont été combinées pour n'en faire qu'une seule.

Les concepteurs du modèle OSI pensaient qu'il serait à la base de nombreux développements qui deviendraient les standards du secteur des communications informatiques. Malheureusement pour eux ce ne fut pas le cas et aujourd'hui, c'est le modèle TCP/IP qui domine en pratique. Le modèle OSI quant à lui reste le modèle de base pour toute la théorie dans le domaine des réseaux d'ordinateurs.

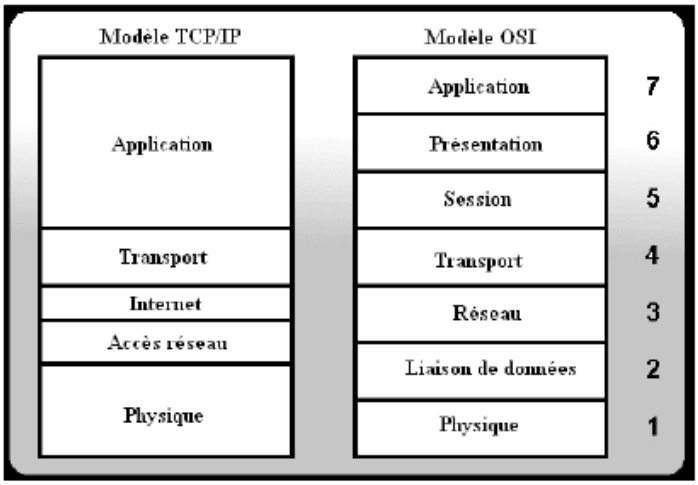


FIG. 3.7 – Modèle OSI et Modèle TCP/IP

**WiFi**

Les spécifications WLAN les plus importantes ont été développées par le groupe 802.11 de l'IEEE (*Institute of Electrical and Electronics Engineers*). C'est en 1997 que ce groupe a donné naissance au standard IEEE 802.11 qui est utilisé pour définir les réseaux locaux hertziens.

Le standard IEEE 802.11 définit les deux premières couches du modèle OSI, à savoir la couche physique et la couche liaison de données (voir figure 3.8).

| | | | | |
|---|---|---|---|---|
| **Couche 2 OSI Liaison de données** | **802.11 Logical Link Control (LLC)** | | | |
| | **802.11 Medium Acces Control (MAC)** | | | |
| **Couche 1 OSI Physique (PHY)** | **DSSS** | **FHSS** | **IR** | **...** |

FIG. 3.8 – Couches 1 et 2 du 802.11

Trois couches physiques étaient définies dans la norme initiale 802.11 :

– DSSS (*Direct Sequence Spread Spectrum*) : étalement de spectre à séquence directe.

Technique opérant dans la bande des 2,4GHz à des débits de 1 et 2 Mbits/sec. Comme nous pouvons le voir sur la figure 3.9, elle divise la bande des 2,4 GHz en 13 ou 14 canaux de 22 MHz. Ces canaux fournissent un signal très bruité, car les canaux adjacents ont des bandes passantes qui se recouvrent partiellement et peuvent donc se perturber mutuellement.

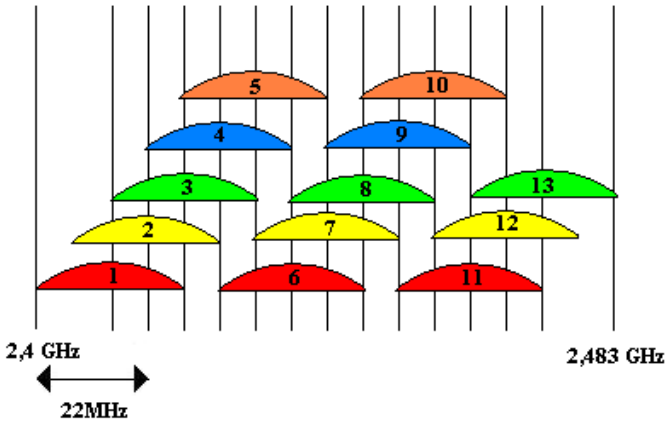


FIG. 3.9 – DSSS (Direct Sequence Spread Spectrum)

- FHSS (*Frequency-Hopping Spread Spectrum*) : étalement de spectre par saut de fréquence. Le nombre de canaux disponibles est plus grand que dans le DSSS. En effet, la bande passante est divisée en un minimum de 75 canaux d'une largeur de 1MHz, la transmission se faisant en utilisant une combinaison de canaux connue de toutes les stations de la cellule.
- Infrarouge (IR) : mêmes débits que pour les autres couches mais utilisant des longueurs d'onde différentes.

Une nouvelle couche physique a été ajoutée par la suite : OFDM. Le principe de l'OFDM consiste à diviser le signal que l'on veut transmettre sur différentes ondes porteuses, comme si l'on combinait ce signal sur un grand nombre d'émetteurs indépendants, fonctionnant à des fréquences différentes. Pour que les fréquences des porteuses soient les plus proches possibles et ainsi maximiser la quantité d'information transmise sur un plage de fréquences donnée, l'OFDM utilise des porteuses orthogonales entre elles. Les signaux des différentes porteuses se chevauchent mais, grâce à l'orthogonalité, n'interfèrent pas entre eux.

La méthode d'accès de la couche MAC est basée sur le CSMA/CA (*Carrier Sense Multiple*

*Acces with Collision Avoidance*). Cette méthode consiste en une écoute du canal de transmission avant l'envoi. Si le canal est libre, l'envoi est immédiat. Autrement, le noeud attend un temps aléatoire avant de transmettre (*back-off value*). La probabilité que deux noeuds choisissent le même *back-off* étant faible, le risque de collision l'est aussi.

Ceci étant, il n'y a pas de mécanisme de détection des collisions. En effet, les WLAN sont généralement munis d'une seule antenne et les noeuds ne sont donc pas capables d'écouter pendant qu'ils envoient. Un système de confirmation (ACK) est donc mis en place entre le récepteur et l'émetteur pour confirmer la réception d'un paquet.

La demande en matière de réseaux sans fils opérants à des fréquences et des débits différents est très importante. Pour répondre à ces demandes, le groupe 802.11 à publié une série de normes que nous allons décrire.

- **802.11a** : La norme IEEE 802.11a utilise une bande de fréquences appelée UNII (*Unlicensed National Information Infrastructure*) divisée en trois parties :

  1. UNII-1 (de 5,150 à 5,250 GHz), exploitée en intérieur.
  2. UNII-2 (de 5,250 à 5,350 GHz), exploitée en intérieur et extérieur.
  3. UNII-3 (de 5,725 à 5,825 GHz), exploitée en extérieur.

  Son avantage par rapport aux normes 802.11b/g est qu'elle dispose d'une plus grande bande passante et des débits plus importants que 802.11b (54Mb/s). De plus, la bande de fréquences qu'elle utilise est relativement peu encombrée. IEEE 802.11a utilise une technique de modulation OFDM.
  Ses inconvénients sont sa faible portée (15m) et son incompatibilité avec 802.11b.
- **802.11b** : Le terme WiFi (*Wireless Fidelity*) fait référence à cette norme qui fut la première norme des WLAN utilisée par un grand nombre d'utilisateurs. A l'heure actuelle, la norme 802.11b est remplacée par la 802.11g, plus rapide. La norme WiFi permet l'interopérabilité entres les différents matériels existants, elle offre des débits de 11Mb/sec, avec une portée de 300m, dans un environnement dégagé. Elle fonctionne dans la bande des 2,4GHz, séparée en plusieurs canaux. Son inconvénient est le risque d'interférence avec les appareils fonctionnant aux mêmes fréquences (four à micro onde, matériel sans fils, ...).
- **802.11g** : Elle étend la norme 802.11b, en augmentant le débit jusqu'à 54Mb/sec. Elle fonctionne aussi à 2,4GHz, ce qui rend les deux normes parfaitement compatibles. Grâce à cela, les équipements 802.11b sont utilisables avec les points d'accès 802.11g et

vice-versa. Cependant, 802.11g utilise la technique de modulation OFDM.

- **802.11c** : Traite du fonctionnement de pont, équipement reliant deux LAN. Elle analyse les procédures de connexion entre les points d'accès.
- **802.11d** : S'occupe de l'actualisation des réglementations, différentes dans chaque pays.
- **802.11e** : Apporte des modifications à la couche MAC afin d'améliorer la Qualité de Service (QoS). Elle prévoit des communications planifiées, dans des intervalles de temps ou aucun trafic n'est transmis. Ces optimisations visent à utiliser des services de téléphonie sur IP et de diffusion de vidéo en continu.
- **802.11f** : Gère l'interopérabilité entre des points d'accès de différents fabricants. Cette norme facilite le *roaming*.
- **802.11h** : Traite de la gestion du spectre et de la puissance afin de les rendre conformes aux normes européennes.
- **802.11i** : Gère le mécanisme d'authentification et de sécurité au niveau de la couche MAC. Elle résout notamment les faiblesses du protocole WEP (*Wired Equivalent Privacy*) conçu pour la couche MAC 802.11.
- **802.11k** : Apporte des améliorations dans le domaine de la mesure des ressources radio, dans le but d'arriver à une meilleure gestion du réseau. Elle définit quelles sont les informations qu'il faut rendre disponibles pour la gestion et la maintenance des WLAN.
- **802.11n** : La norme 802.11n est attendue pour avril 2007. Le débit théorique atteint 540 Mbit/s (débit réel de 100 Mbit/s dans un rayon de 90 mètres) grâce aux technologies MIMO (*Multiple-Input Multiple-Output*) et OFDM.

Depuis la création de la norme WiFi, permettant l'interopérabilité entres les différents matériels existants, l'installation de ce type de réseaux sans fils a augmenté. Cette évolution s'explique par la demande croissante de mobilité des utilisateurs, ainsi que le faible coût des équipements.

Dans un réseau WiFi, les machines communiquent entre elles grâce à des signaux émis par leurs antennes. Les machines doivent donc être équipées d'une antenne qui fait office de carte réseau. L'opération consistant à acheminer un paquet d'un noeud à un autre s'appelle le routage. Il est effectué par des éléments matériels appelés routeurs qui contiennent une table de routage, représentant l'ensemble des connexions du routeur avec les autres noeuds du réseau.

Il y a deux modes de fonctionnement pour un réseau WiFi, ayant chacun des caractéristiques propres que nous allons décrire : les modes infrastructure et ad-hoc. Les cartes réseau peuvent

être configurées de façon à fonctionner dans un mode ou dans l'autre, mais pas dans les deux simultanément.

### 3.2.2 Mode infrastructure

Ce mode de fonctionnement est très semblable au protocole Ethernet des réseaux filaires. Les machines se connectent à un point d'accès (AP) qui partage la bande passante disponible, ce mode de fonctionnement est illustré à la figure 3.10.

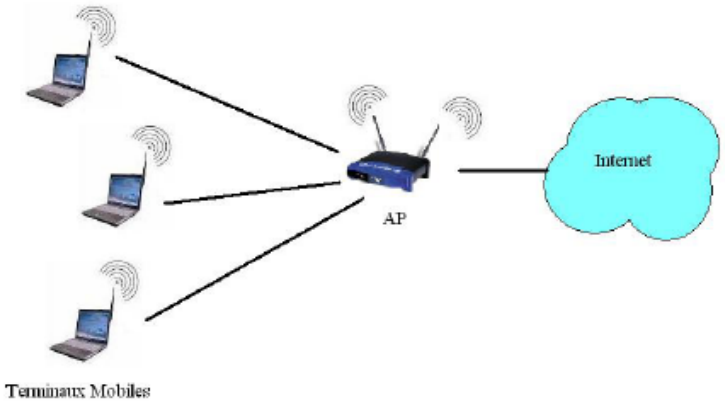


FIG. 3.10 – Mode infrastructure

L'AP donne l'accès au réseau aux machines qui le désirent, on peut le comparer aux concentrateurs (hub) des réseaux fixes. Les AP sont en général câblés entre eux afin de créer un réseau de bornes d'accès. Chaque station (machine dans le domaine *Wireless*) se connecte à l'AP. Le domaine formé par celui-ci et les stations qu'il couvre est appelé *Basic Service Set* (BSS). Il est évidement possible de créer un réseau reliant plusieurs BSS entre eux, ils sont différenciés via leur *BSS Identifier* (BSSID) de 6 octets correspondant à l'adresse MAC de l'AP. Contrairement aux réseaux téléphoniques mobiles, il n'y a pas de mécanisme de gestion du changement d'AP. Si une station se déplace, elle cherche le meilleur AP accessible afin de s'y connecter, la connexion sera cependant interrompue si elle change d'AP au cours de son déplacement.

### 3.2.3 Mode ad-hoc

Ce mode n'a pas besoin d'AP pour fonctionner, ce sont les stations elles-mêmes qui entrent en communication sans s'appuyer sur un équipement extérieur (voir figure 3.11). Tous les noeuds d'un réseau de ce type se comportent comme des routeurs et prennent part à la découverte et à la maintenance des chemins de communication entre les différentes machines. Il existe un grand nombre de protocoles de routage et nous en détaillerons certains dans la suite de ce rapport.

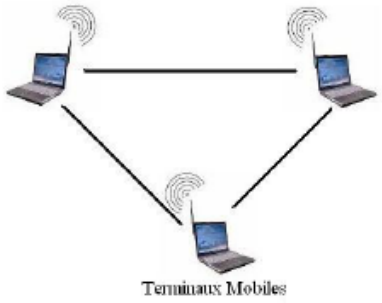


FIG. 3.11 – Mode ad-hoc

L'avantage de ces réseaux réside dans la facilité de mise en place et d'ajout de nouvelles stations sur le réseau. L'absence de structure fixe diminue aussi le coût de leur mise en oeuvre.

## 3.3 Réseaux ad-hoc

Un réseau mobile ad-hoc, appelé MANET (*Mobile Ad-hoc NETwork*), consiste donc en un grand nombre d'unités mobiles se déplaçant dans un environnement quelconque en utilisant, comme moyen de communication, des interfaces sans fils sans infrastructure préexistante. Historiquement, l'utilité première des réseaux ad-hoc fut d'améliorer les communications sur le terrain dans le domaine militaire. Cependant, au fil du temps, leur utilité dans d'autres domaines a fait son chemin et nous sommes aujourd'hui à l'aube de leur avènement, avec la 4G.

Les réseaux ad-hoc sont formés dynamiquement par des stations mobiles (noeuds) qui se connectent sans utiliser d'infrastructure existante. Ces noeuds sont donc libres de se déplacer et de s'organiser arbitrairement, impliquant une grande variabilité de la topologie du réseau (voir figure 3.12).

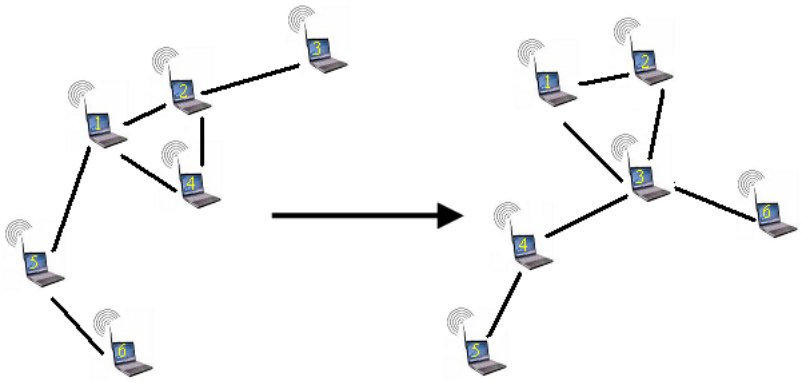

FIG. 3.12 – Changement de topologie dans les réseaux ad-hoc

Généralement, les routes entre les noeuds d'un tel réseau sont constituées de plusieurs sauts (*hops*). Chaque noeud est capable de communiquer directement avec ses voisins (se trouvant dans la zone de portée de leur antenne), voisins par lesquels ils passent pour communiquer avec des noeuds plus éloignés.

Les caractéristiques de ces réseaux ajoutent des contraintes à leur mise en oeuvre [5] :

- Sans infrastructure : Les MANET ne dépendent donc pas d'une infrastructure préétablie, chaque noeud opère comme un routeur indépendant. L'organisation du réseau doit donc être distribuée à tous les noeuds, ce qui rend la détection d'erreur et la gestion du réseau complexes.
- Topologie dynamique : Les noeuds se déplaçant arbitrairement, la topologie change fréquemment et de façon aléatoire. Cela implique que les routes entre les noeuds changent et des paquets peuvent ainsi être perdus.
- Connexions variables : Les noeuds peuvent avoir plusieurs interfaces radios, présentant des propriétés de débit ou de fréquences différentes. Ces variations donnent naissance à des connexions asymétriques.
- Contraintes d'énergie : Les batteries utilisées par les noeuds ne sont pas illimitées, les services supportés par ces noeuds sont donc restreints. C'est un problème d'autant plus important que les noeuds sont responsables du routage des paquets dans le réseau, ce qui consomme beaucoup d'énergie.
- Taille : A l'heure actuelle, la plupart des algorithmes utilisés pour les réseaux ad-hoc sont optimisés pour des petits réseaux. Il y a donc des améliorations à apporter dans

certains domaines (sécurité, routage,...) pour pouvoir passer à une échelle supérieure.

Dans les réseaux ad-hoc basés sur un protocole d'accès avec détection de porteuse (comme le 802.11), les caractéristiques de l'accès au support génèrent des problèmes supplémentaires : le problème de la station cachée et celui de la station exposée.

### 3.3.1 Problème de la station cachée

Ce problème survient quand une ou plusieurs stations ne peuvent se détecter (A et C sur la figure 3.13), car elles se trouvent hors de leurs portées respectives, mais leurs zones de transmission ne sont pas disjointes. Une collision peut se produire quand les stations A et C envoient des informations à la station B simultanément.

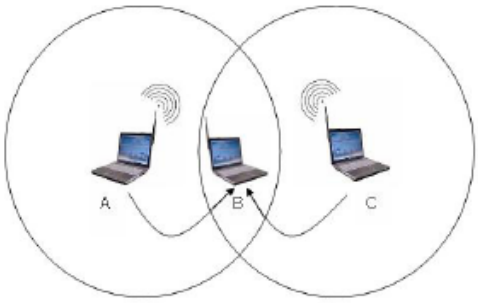


FIG. 3.13 – Problème de la station cachée

Un mécanisme a été pensé afin d'éliminer ce problème : avant l'envoi d'information, l'émetteur transmet un paquet RTS (*Request To Send*) au récepteur, lui annonçant ainsi une demande de transmission. Le récepteur renvoie un paquet CTS (*Clear To Send*) s'il est libre. Cette technique permet donc d'obtenir une certaine visibilité de la station cachée.

### 3.3.2 Problème de la station exposée

Ce problème survient quand une station veut établir une transmission avec une deuxième mais doit la retarder car il y a une transmission en cours entre deux autres stations se trouvant dans son voisinage. La figure 3.14 décrit un scénario typique : supposons que les stations A et C peuvent entendre les transmissions de B, mais que la station A n'entent pas C (et vice-versa). Supposons aussi que B est entrain d'envoyer des données vers A et que, au même moment, C veut communiquer avec D. En suivant la logique CSMA, la station C va commencer par

déterminer si le support est libre. A cause de la communication entre B et A, C trouve le support occupé et il retarde son envoi bien que celui-ci n'aurait pas causé de collision.

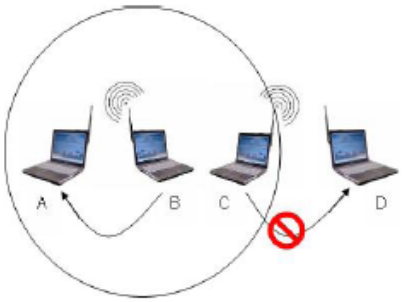


FIG. 3.14 – Problème de la station exposée

A l'heure actuelle, les recherches dans le domaine des MANETs, couvrent plusieurs secteurs, dont :

- Routage : certaines recherches tentent d'utiliser des informations provenant de toutes les couches du modèle OSI afin d'effectuer un routage optimal [6, 7].
- Sécurité : La nature même des réseaux ad-hoc rend leur sécurisation encore plus complexe que dans le ces des réseaux sans fils classiques, de nombreux chercheurs tentent de trouver le meilleur moyen de les rendre robustes aux attaques extérieures [8, 9].

Maintenant que nous avons introduit les réseaux 802.11 fonctionnant en mode ad-hoc, nous allons nous intéresser de plus près aux algorithmes nécessaires à leur mise en oeuvre. Pour établir une communication *end-to-end*, l'émetteur a premièrement besoin de localiser le récepteur au sein du réseau. Il faut donc lier son adresse à sa situation dans le réseau. Une fois la localisation effectuée, il faut router l'information à travers les différents noeuds du réseau. Il existe de nombreux algorithmes de routage, nous en détaillerons certains au point 3.3.4.

### 3.3.3 Localisation

Les services de localisation doivent donc savoir répondre aux requêtes de localisation des noeuds. Une méthode simple pour effectuer cette localisation est d'inonder le réseau de cette requête (*flooding*). Bien entendu cette solution n'est valable que pour des réseaux de taille réduite où ces paquets n'ont pas une grande influence, même en considérant leur fréquence d'envoi élevée. Une amélioration de cette technique consiste à affiner l'inondation, en augmentant le nombre de sauts utilisés pour la propagation du message, et ce jusqu'à ce

que le noeud recherché soit trouvé. Cette technique est dite réactive car la localisation a lieu chaque fois qu'un noeud à besoin d'en trouver un autre.

Une autre possibilité est d'utiliser une méthode proactive, qui construit des tables contenant les informations de chaque noeud et qui les gardent en mémoire. Ces tables sont mises à jour afin de prendre en compte les modifications de la topologie du réseau.

### 3.3.4 Routage

Avant de parler du routage proprement dit, il est bon de rappeler quels sont les principaux modes de communication dans les réseaux mobiles : la communication point à point ou *unicast*, pour laquelle il y a une source et une seule destination, la communication multipoint ou *multicast*, qui permet d'envoyer une message à plusieurs destinataires et la diffusion ou *broadcast*, qui envoie un message à tous les noeuds du réseau. Ces trois modes de communication sont schématisés par la figure 3.15.

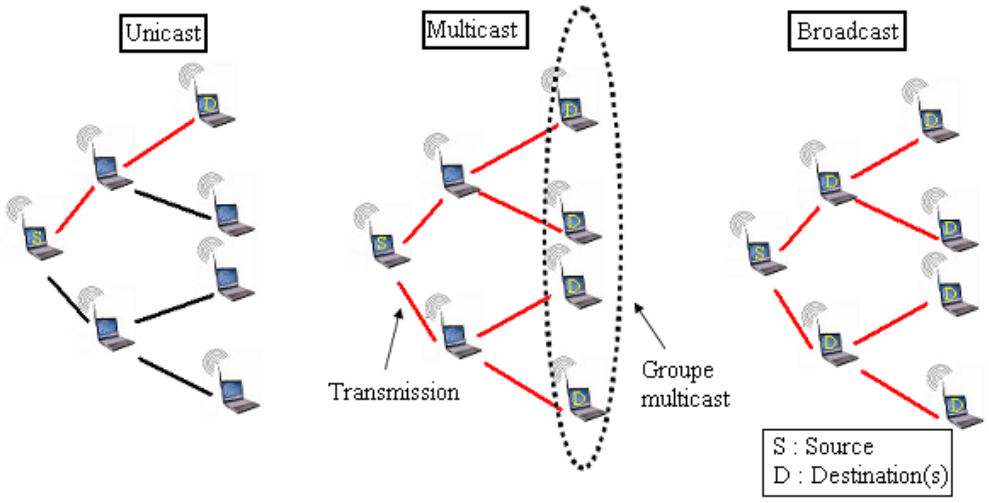


FIG. 3.15 – Modes de communication dans les réseaux mobiles

Le routage dans les réseaux ad-hoc est, comme nous l'avons vu, assez délicat étant donnée la nature changeante de la topologie de ce type de réseaux. De nombreux protocoles et algorithmes ont été proposés, et leurs performances ont été analysées dans différentes situations.

Ces algorithmes peuvent être classés en deux grandes catégories :

1. Protocoles Réactifs : Pour ce type d'algorithme, les routes ne sont créées que quand elles sont demandées. Lorsqu'un noeud désire envoyer un paquet vers un autre noeud

du réseau, celui-ci invoque un mécanisme de découverte des chemins vers la destination. La route ainsi créée reste valide tant que le noeud final est joignable ou jusqu'à ce que la route ne soit plus utilisée.

2. Protocoles Proactifs : Chaque noeud maintient une ou plusieurs tables contenant l'information de routage vers tous les autres noeuds du réseau. Quand la topologie change, les noeuds propagent des messages de mise à jour de ces tables afin de garder une vue consistante et actualisée de l'entièreté du réseau.

Généralement, les algorithmes réactifs sont plus efficaces que les proactifs. Ils permettent en effet de minimiser les messages de contrôle du réseau et de limiter ainsi la consommation d'énergie. L'avantage des protocoles proactifs est que chaque noeud connaît à tout moment l'entièreté du réseau et, quand un message doit être envoyé, aucun temps n'est perdu dans une recherche de chemin. Cependant, ils conservent une grande quantité d'information inutile car tous les chemins ne sont pas utilisés. De plus, si le réseau change fréquemment de topologie, la mise à jour des tables de routage prends un temps considérable et gaspille les capacités du réseau.

En plus des protocoles réactifs et proactifs, il existe des protocoles hybrides qui combinent les deux approches, afin de profiter des avantages des deux méthodes. La figure 3.16 montre une classification non exhaustive de différents protocoles de routage.

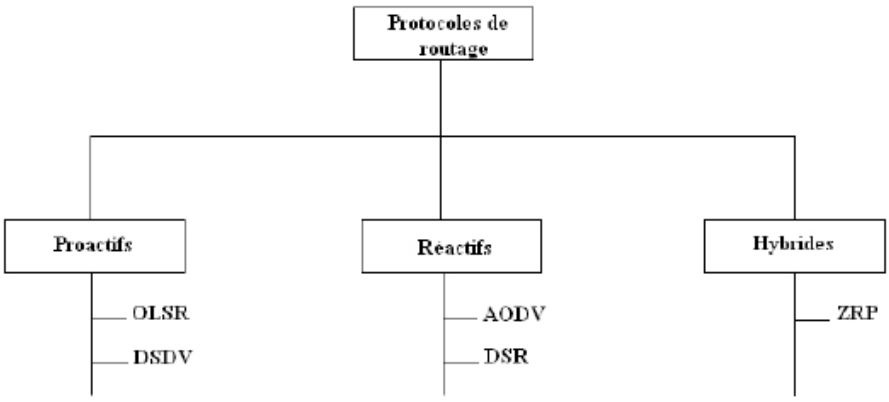


FIG. 3.16 – Classification des protocoles de routage ad-hoc

Nous allons maintenant analyser différents algorithmes des trois familles.

### 3.3.5 Algorithmes Proactifs

#### Destination Sequenced Distance Vector (DSDV)

DSDV [10] est un protocole dans lequel chaque noeud du réseau contient une table de routage dans laquelle sont entrées toutes les destinations accessibles, ainsi que le nombre de noeuds intermédiaires par lesquels transiter pour atteindre la destination. A chaque entrée est associé un numéro de séquence, donné par le noeud destination, permettant de distinguer les nouvelles routes des anciennes et d'éviter la formation de boucles de routage. Ce sont les routes les plus récentes (plus grand numéro de séquence) qui sont utilisées pour les transferts. En cas d'égalité, l'algorithme se base sur la métrique choisie (voir point 3.3.8).

Les mises à jour des tables sont transmises périodiquement à travers le réseau afin de maintenir la consistance des informations. Toutes ces mises à jour génèrent un trafic important qu'il faut absolument limiter. Pour cela, il existe deux types de paquets de mise à jour : les *full dump*, contenant toutes les informations et des paquets plus petits, ne contenant que les informations ayant changé depuis le dernier *full dump*.

### 3.3.6 Algorithmes Réactifs

#### Dynamic Source Routing (DSR)

Dans cet algorithme, chaque paquet contient la liste complète des noeuds par lesquels il doit passer pour arriver à destination. L'avantage de cette méthode est que les noeuds intermédiaires n'ont pas besoin de maintenir les informations sur la route à jour puisque le paquet possède toutes ces informations.

DSR est décomposé en deux parties : la découverte des routes et la maintenance de celles-ci. Pour la découverte du chemin (voir figure 3.17), un noeud 1 envoie un paquet *route request* à destination du noeud 8. Ce paquet se propage dans le réseau jusqu'à arriver au noeud destination ou à un noeud connaissant un chemin vers celui-ci. Le paquet contient l'adresse source, l'adresse de destination, un numéro d'identification, ainsi qu'un champ dans lequel sera accumulée la séquence des noeuds visités durant la propagation de la requête dans le réseau (*route record*).

Quand un noeud reçoit un paquet *route request*, il vérifie s'il connaît un chemin vers la destination. Si ce n'est pas le cas, il ajoute son adresse dans le *route record* et transmet le paquet à ses voisins. Ce transfert n'a lieu que si l'adresse du noeud n'apparaît pas déjà dans

le *route record*, afin d'éviter les boucles et la multiplication des paquets *route request*.

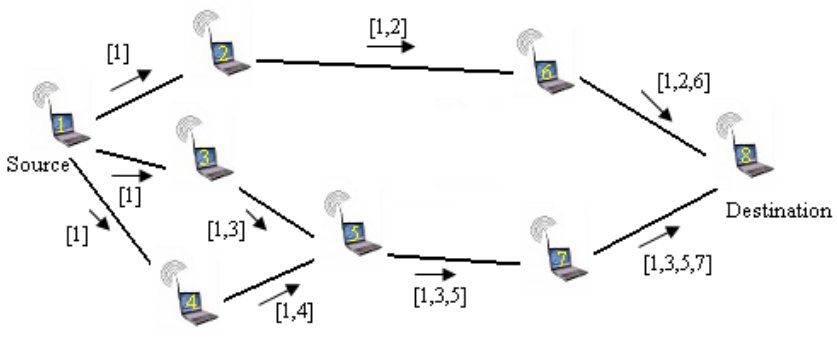


FIG. 3.17 – Découverte de route dans DSR

Quand le paquet atteint sa destination, le noeud ainsi atteint envoie un paquet en réponse via le chemin donné dans le *route record*, si les liaisons sont symétriques, ou via un autre chemin (moyennant éventuellement une découverte de chemin). La figure 3.18 montre le cas de connexions symétriques. Afin de diminuer le coût de la recherche de route, chaque noeud garde en mémoire les routes qu'il a apprises, ce qui diminue le besoin de demande de route.

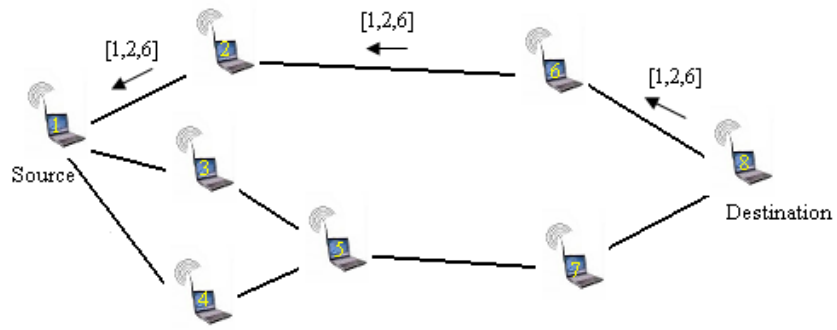


FIG. 3.18 – Renvoi du chemin dans DSR

La maintenance de la route consiste à envoyer un paquet *route error* quand une route est inutilisable. Quand un noeud détecte un problème fatal de transmission, ce paquet, contenant l'adresse du noeud qui a détecté l'erreur et celle du noeud qui le suit dans le chemin, est envoyé à l'émetteur original du message. Lors de la réception du paquet *route error* par la source, le noeud concerné par l'erreur est supprimé du chemin sauvegardé, et tous les chemins qui contiennent ce noeud sont tronqués à ce point là. Par la suite, une nouvelle opération de

découverte de routes vers la destination est initiée par l'émetteur.

### Ad-hoc On Demand Distance Vector (AODV)

AODV [11] peut être considéré comme la combinaison de DSR et de DSDV. En effet, ce protocole emploie les mécanismes de découverte de chemin et de maintenance de route de DSR en y associant le numéro de séquence et les mises à jour périodiques de DSDV.

Envisageons le cas, illustré à la figure 3.19, où le noeud S veut communiquer avec le noeud D. Pour cela, s'il n'existe pas encore de route valide entre eux, le message RREQ (*Route REQuest*) est envoyé dans le réseau afin de trouver le chemin entre les deux noeuds. AODV utilise le numéro de séquence pour éviter les boucles et être sûr d'utiliser les routes les plus récentes. Le RREQ est d'abord envoyé aux plus proches voisins qui vont ensuite propager le message, jusqu'à atteindre le noeud destination (ou un noeud connaissant un chemin vers celui-ci). Chaque noeud intermédiaire enregistre dans sa table de routage l'adresse du noeud qui lui a transmis le RREQ, établissant ainsi le chemin de retour (*Reverse Path*). Si un noeud reçoit plusieurs copies d'un même RREQ, seule la première est conservée.

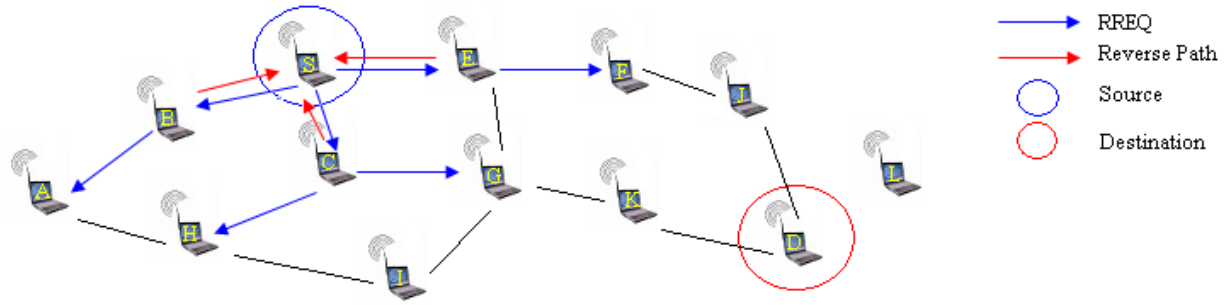


FIG. 3.19 – Découverte de route dans AODV

Une fois que le message atteint le noeud destination (ou un noeud connaissant un chemin vers celui-ci), ce noeud transmet un message RREP (*Route REPly*) vers la source par le *Reverse Path* (voir figure 3.20) : il parcourt donc le chemin en sens inverse en modifiant les tables de routage des noeuds par lesquels il passe. En effet, l'information à conserver dans la table de routage est le noeud suivant dans le chemin (*Forward Path*) et pas le *Reverse Path* précédemment enregistré. Étant donné que le RREP est envoyé par le même chemin que le RREQ, AODV ne supporte que des liens symétriques.

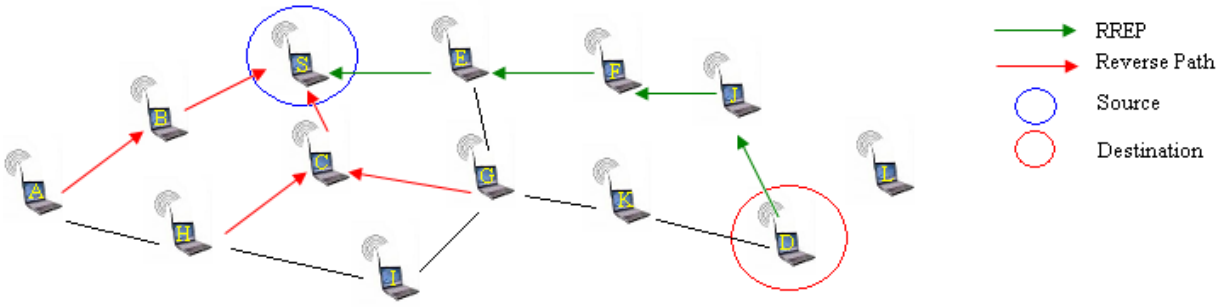


FIG. 3.20 – Réponse de la destination dans AODV

La figure 3.21 montre ce qu'il se passe en cas de cassure dans un chemin : un message RERR (*Route ERRor*) est envoyé dans le réseau. Si un noeud a, dans sa table de routage, un chemin passant par ce lien, il efface la route et recommence une recherche de chemin, s'il en a besoin.

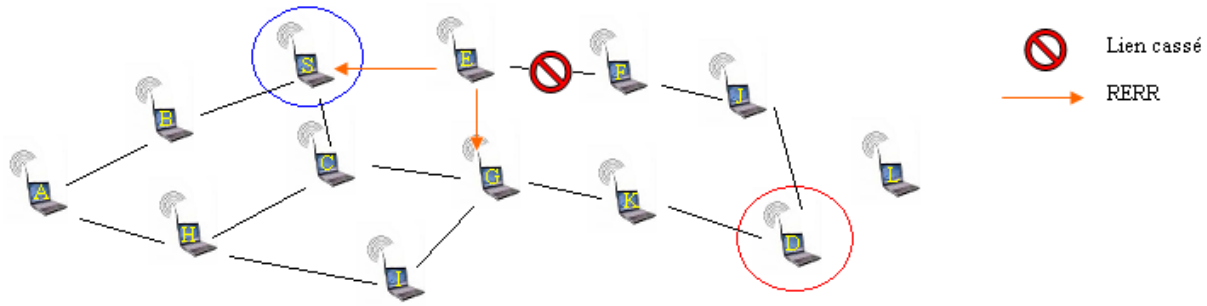


FIG. 3.21 – Erreur dans AODV

AODV permet aussi aux noeuds d'envoyer périodiquement des messages *Hello*, qui sont utilisés par les noeuds afin de signaler leur présence. Ces messages permettent entre autre de détecter des cassures de route.

### Associativity Based Routing (ABR)

ABR constitue une nouvelle approche de routage pour les réseaux ad-hoc. Ce protocole introduit la notion de stabilité de route comme nouvelle métrique de routage. Dans ABR, le choix d'une route ne se fait donc plus uniquement sur le nombre de sauts mais tient compte de la stabilité de la connexion existante entre deux noeuds du réseau. L'objectif d'ABR est donc

de trouver des chemins ayant une grande durée de vie. La formule exacte de ce calcul reste inconnue, l'auteur d'ABR ayant décidé de la garder secrète pour des raisons professionnelles. Cependant, le principe de fonctionnement d'ABR est connu [12].

Les noeuds du réseau génèrent périodiquement des messages de contrôle afin de signaler leur existence au reste du réseau. Quand un noeud reçoit ce message, il met à jour la valeur de l'associativité avec l'émetteur. Ces valeurs d'associativité sont remises à zéro quand la connexion entre les noeuds est perdue. Une grande valeur d'associativité, correspondant à un noeud voisin, indique un état de faible mobilité de ce noeud. Une petite valeur indique un état de forte mobilité du voisin.

Le protocole consiste en trois phases principales : la découverte de routes, la reconstruction des routes, et la suppression des routes. La phase de découverte de routes se passe comme suit : quand un noeud veut trouver un chemin vers une destination, il diffuse un message BQ (*Broadcast Query*) afin de trouver les noeuds qui mènent vers cette destination. Les noeuds intermédiaires ajoutent leur adresse et leurs valeurs d'associativité au paquet de la requête et transfèrent le BQ reçu. Ces noeuds de transit ne maintiennent que l'associativité qui leur est associée et celle du noeud qui les précède dans le chemin. De cette manière, chaque paquet qui arrive à la destination va contenir les valeurs d'associativité de tous les noeuds qui appartiennent au chemin reliant la source et la destination. Le noeud destination peut donc choisir le meilleur chemin en comparant les valeurs des différents paquets reçus. Si plusieurs chemins ont la même valeur, le chemin ayant le plus petit nombre de sauts est choisi. Une fois ce choix effectué, le noeud destination envoie un paquet de réponse (REPLY) au noeud source en utilisant le chemin choisi. Les noeuds qui appartiennent au chemin suivi par le paquet REPLY indiquent que leurs routes sont valides, le reste des routes reste inactif.

La phase de reconstruction des routes (RRC) consiste en plusieurs phases : une découverte partielle de routes, une suppression de routes invalides, une mise à jour des routes valides et une nouvelle découverte de routes. Toutes ces étapes ne sont pas effectuées à chaque fois. Les actions à entreprendre dépendent en effet du noeud dont le mouvement a conduit à une rupture de route. Par exemple, le mouvement du noeud source, implique un nouveau cycle BQ-REPLY de par la nature réactive du protocole. Un message de notification de route RN (*Route Notification*) est utilisé dans le but d'éliminer les routes invalides des noeuds suivants dans le chemin. Si c'est le noeud destination qui s'est déplacé, son prédécesseur supprime sa route et initie une recherche locale LQ (*Localized Query*) afin de savoir si la destination est toujours joignable. Si le paquet LQ est reçu par la destination, celle-ci choisi la meilleure

route partielle existante et envoie un REPLY. Dans le cas contraire, le noeud ayant initié cette LQ envoie un message à son prédécesseur, après avoir attendu un temps déterminé, pour lui signaler l'invalidité de la route et lui dire d'invoquer à son tour une procédure LQ. Cette procédure ne s'étend bien entendu pas jusqu'au premier noeud de la route, arrivé à mi-chemin elle s'arrête pour laisser place à un nouveau cycle BQ-REPLY.

Quand un chemin cesse d'être utilisé par une certaine source, une diffusion de suppression de route RD (*Route Delete*) est lancée. Tous les noeuds appartenant à ce chemin suppriment les entrées correspondantes de leurs tables de routage.

### 3.3.7 Algorithmes hybrides

#### Zone Routing Protocol (ZRP)

ZRP [13, 14] est un protocole de routage dit hybride. Il met en place, simultanément, un routage proactif et un routage réactif, afin de combiner les avantages des deux approches. Pour ce faire, il passe par un concept de découpage du réseau en différentes zones, appelées "zones de routage". Une zone de routage pour un noeud S, est définie par son "rayon de zone". Ce rayon correspond au nombre de sauts maximum qu'il peut y avoir entre un noeud D et le noeud S.

Un exemple est donné à la figure 3.22. On voit que, pour un rayon de zone égal à deux, la zone de routage du noeud S est constituée par tous les noeuds qui sont autour du noeud S avec un maximum de deux sauts les séparants. Sont donc inclus dans la zone de routage, tous les voisins du noeud S ainsi que tous les voisins de ces voisins.

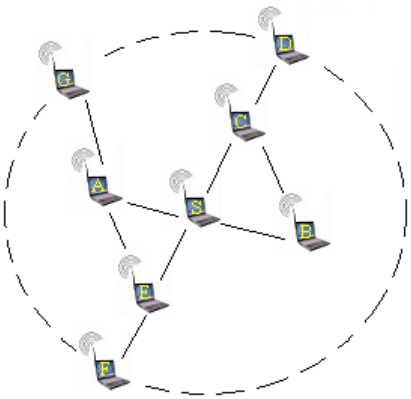


FIG. 3.22 – ZRP : Zone de routage

Le routage au sein d'une zone se fait de manière proactive, via le protocole IARP (*IntrAzone Routing Protocol*) et le routage vers les noeuds extérieurs de la zone se fait de façon réactive, grâce au protocole IERP (*IntErzone Routing Protocol*). En plus de ces deux protocoles, ZRP utilise le protocole BRP (*Bordercast Routing Protocol*). Ce dernier a pour but de construire la liste des noeuds périphériques d'une zone ainsi que les routes permettant de les atteindre, en utilisant les données de la topologie fournies par le protocole IARP. Il est utilisé pour propager des requêtes de recherche de routes de l'IERP dans le réseau.

La recherche des chemins s'effectue comme suit : on vérifie tout d'abord si le noeud destinataire se trouve dans la zone du noeud source (chaque noeud connaissant le contenu de sa zone), auquel cas le chemin est déjà connu. Autrement, une demande d'établissement de route RREQ est initiée vers tous les noeuds périphériques, ces derniers vérifient si la destination existe dans leurs zones. Dans l'affirmative, la source recevra alors un paquet RREP contenant le chemin menant à la destination. Dans le cas contraire, les noeuds périphériques diffusent la requête à leurs propres noeuds périphériques qui, à leur tour, effectuent le même traitement.

Maintenant que nous avons une vue plus précise des protocoles de routage, il faut parler de la façon d'évaluer les performances d'un réseau ad-hoc et donc d'un protocole de routage.

### 3.3.8 Métriques

Comme nous l'avons vu, il existe une multitude de protocoles de routage différents, ayant chacun leurs caractéristiques propres. La question posée est la suivante : comment comparer ces méthodes, étant données leurs différences ? Les métriques de routage sont là pour tenter de répondre à cette question : il s'agit de méthodes employées pour évaluer les performances d'un réseau ad-hoc. Il y a deux familles de métriques de routage : les métriques de performance, qui détaillent le résultat d'une simulation, et les métriques de scénario qui décrivent les paramètres fournis au début de la simulation.

#### Métriques de performance

Ces métriques sont très utiles car elles permettent de montrer ce qu'il s'est réellement passé durant la simulation et donc de décrire au mieux le protocole utilisé. Les métriques les plus utilisées sont :

– *End-to-End Throughput* : moyenne des transmissions réussies, c'est à dire la mesure du nombre de paquets transmis avec succès à leur destination dans un intervalle de temps

donné.

- *End-to-End Delay* : mesure du temps mis pour que les paquets arrivent à destination.
- *Link Utilisation* : probabilité qu'un noeud soit utilisé pour effectuer la transmission
- *Packet Loss* : pourcentage des paquets envoyés qui ne sont jamais arrivés à destination ou ayant été corrompus durant le transfert.
- *Packet Delivery Fraction* : mesure du rapport entre le nombre de paquets reçus par la destination et le nombre de paquets envoyés par la source.
- *Routing Overhead* : métrique très intéressante, qui mesure le nombre total de paquets de routage transmis pendant la simulation. Elle montre donc à quel point un protocole consomme de la bande passante avec ses messages de routage.
- *Path Optimality* : calcul de la différence entre le nombre de noeuds par lesquels un paquet est passé pour arriver à destination et le plus court chemin existant physiquement dans le réseau au moment de l'émission du paquet. Si les paquets passent souvent par des chemins proches du plus court, le protocole est dit être un bon protocole.

**Métriques de scénario**

De telles métriques sont calculées à partir des paramètres de départ de la simulation. Elles ne dépendent donc pas du protocole de routage ou du déroulement de la simulation.

- *Mobility* : évaluation de la mobilité des noeuds du réseau. Le calcul se fait en mesurant le mouvement relatif d'un noeud par rapport aux autres.
- *Pause Time* : temps moyen où les noeuds ne sont pas en mouvement. Plus longues sont les pauses, moins le mouvement est important. Il faut cependant signaler que mesurer le mouvement des noeuds via cette métrique n'est pas fiable. En effet, rien ne dit que si un noeud fait une longue pause, il ne repartira pas de sa position avec une vitesse élevée causant de ce fait de nombreuses pertes de connexions.

Le dernier domaine dont nous devons parler est celui de la combinaison des réseaux ad-hoc et infrastructure. Pour cela nous allons commencer par nous intéresser à l'intégration des WLAN avec les réseaux cellulaires et regarder certaines solutions existantes, pour ensuite passer à l'hybridation des modes ad-hoc et infrastructure dans les réseaux WiFi.

# Chapitre 4

# Intégration

## 4.1 Intégration WLAN - UMTS

Les évolutions et le déploiement important des WLAN a augmenté l'intérêt d'une intégration de ceux-ci avec des réseaux mobiles de troisième génération. En effet, ces deux types de réseaux sont complémentaires : les réseaux cellulaires fournissent une couverture universelle et une forte mobilité tandis que les WLAN offrent un haut taux de transfert (voir figure 4.1).

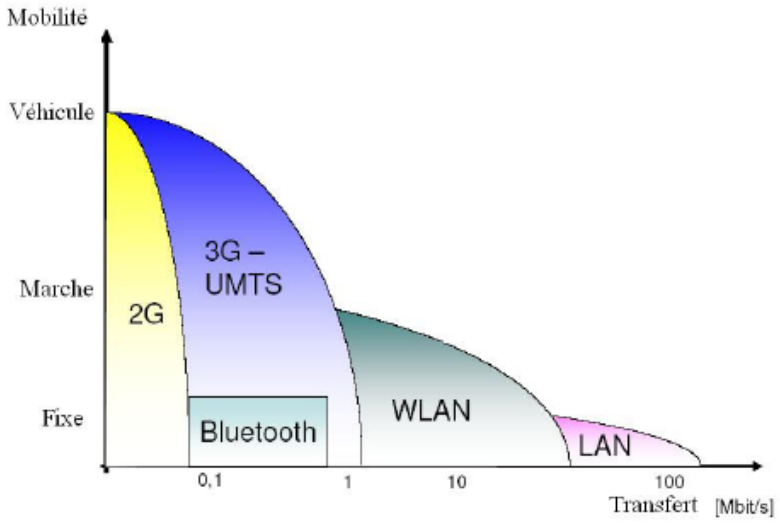


FIG. 4.1 – Mobilité et transfert

Le but d'une telle intégration est de développer un réseau mobile capable de supporter

divers services de données (voir point 3.1.5). Cette intégration est liée à des challenges techniques importants, comme le *handover* vertical, la sécurité, l'authentification, la facturation, la QoS (Quality of Service),...

Le couplage entre les WLAN et l'UMTS peut être implémenté de différentes manières. Le 3GPP (*3rd Generation Partnership Project*) a défini six scénarios de couplage définissant le type et la qualité des services offerts aux utilisateurs. Chacun de ces scénarios rend disponible de nouvelles fonctionnalités et augmente le niveau d'intégration.

Du point de vue architecture cependant, ces scénarios peuvent être réduits à quatre niveaux de couplage [15] :

1. *Open coupling* : Dans ce scénario, l'UMTS et le WLAN utilisent des réseaux d'accès et de transport n'ayant en commun que la facturation.
2. *Loose coupling* : Ce couplage rend possible une authentification commune en créant un lien entre le HLR du réseau UMTS et le serveur AAA (*Authentication Authorization Accounting*) du WLAN qui restent physiquement séparés.
3. *Tight coupling* : Dans ce cas ci, L'AP du réseau WLAN est connectée comme un RNC (*Radio Network Controller*) au réseau. Avec ce couplage, le *handover* entre le WLAN et le réseau cellulaire est possible.
4. *Very tight coupling* : Le WLAN fait partie intégrante du UTRAN (*UMTS Terrestrial Radio Acces Network*), grâce à la connexion de l'AP du WLAN au RNC, ce qui offre la possibilité d'avoir des services homogènes.

Tous les cas discutés ci dessus sont basés sur le mode infrastructure des WLAN. Une autre approche possible est d'utiliser le mode ad-hoc qui permet une extension du réseau UMTS et ce, afin de répondre à une demande croissante de trafic dans les réseaux cellulaires. En effet, cette demande combinée à la capacité limitée du réseau est la cause principale de congestion de celui-ci. Cette congestion implique que, dans certaines cellules, il n'y a plus aucun canal de communication disponible pour les utilisateurs.

L'idée est qu'un terminal peut communiquer avec la BS de façon directe ou indirecte, en passant par un autre terminal servant de relais. Afin de faciliter la séparation des deux réseaux, plusieurs solutions sont envisagées :

- Différentes interfaces, et donc différentes fréquences
- Même interface mais des fréquences différentes

### 4.1.1 Solution existante

**Integrated Cellular and Ad-hoc Relay system : iCAR**

iCAR [16] propose d'intégrer l'infrastructure cellulaire avec des réseaux ad-hoc afin d'atteindre un équilibre de charge entre les différentes cellules. L'idée est de placer un certain nombre de relais ad-hoc, ARS(*Ad-hoc Relay Station*), à des endroits stratégiques. Ces ARS pouvant êtres utilisés entre les mobiles et les BS, ceci afin d'augmenter la couverture mais aussi de rediriger le trafic d'une cellule congestionnée vers une autre ne l'étant pas. Notons que les ARS possèdent deux interfaces, une pour le réseau cellulaire et une pour le réseau ad-hoc, de plus elles sont plus petites et moins coûteuses que les BTS.

## 4.2 Intégration ad-hoc – infrastructure

Un accès à Internet, n'importe où et n'importe quand, devient une chose essentielle à l'heure actuelle. Les AP deviennent de plus en plus nombreux, tant chez les particuliers que dans les grandes sociétés, le problème restant toujours la couverture limitée de ces équipements.

Comme nous l'avons vu, les réseaux ad-hoc sont considérés comme une des meilleures solutions pour étendre cette couverture. En effet, les noeuds qui ne sont pas directement connectés à l'AP peuvent se connecter via leurs voisins. Il y a cependant un certain nombre de contraintes qui entrent en compte quand on veut connecter un réseau ad-hoc à Internet [17].

### 4.2.1 Connexion Internet pour les réseaux ad-hoc

Par définition, les noeuds appartenant à un réseau ad-hoc ne peuvent communiquer qu'avec d'autres noeuds de ce même réseau. Chacun de ces noeuds a une adresse unique qui n'a aucun sens hors du réseau étant donné qu'aucune information ne transite entre celui-ci et l'extérieur. La figure 4.2 montre comment il est possible de fournir un accès Internet à un réseau ad-hoc. Il faut que certains noeuds servent de liaison entre les deux domaines, ces noeuds sont appelés des *gateways* et doivent avoir deux interfaces réseau, une pour les communications à l'intérieur du réseau et l'autre pour les transferts avec Internet.

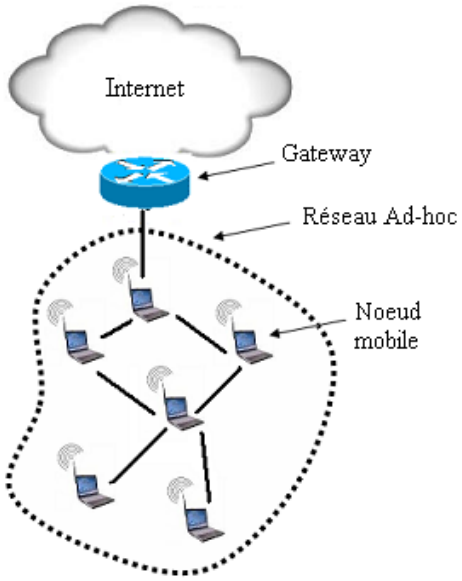


FIG. 4.2 – Accès Internet pour les réseaux ad-hoc

Il y a deux méthodes pour envoyer des messages hors d'un réseau ad-hoc [18, 19] : avec et sans tunnel. Dans ces deux approches, un noeud mobile a besoin de connaître l'adresse du *gateway* et avoir une route vers celui-ci. Il connaît aussi son préfixe réseau qu'il peut comparer avec l'adresse du noeud avec lequel il désire communiquer, afin de savoir si celui-ci appartient ou non au même réseau.

Dans la méthode avec tunnel, quand le destinataire d'un message réside hors du réseau, la source encapsule les paquets destinés au monde extérieur et donne comme adresse de destination celle du *gateway*. Quand celui-ci reçoit un tel paquet, il le désencapsule et le transmet à la destination souhaitée. On dit que les paquets passent par un tunnel entre l'émetteur et le *gateway*, à cause de cette encapsulation.

Avec l'autre technique, si la destination ne se trouve pas dans le réseau, le noeud source envoie ses paquets avec la vraie adresse du destinataire. Ces paquets sont envoyés au *gateway*, en passant par un certain nombre de noeuds intermédiaires. Chaque noeud a donc besoin d'une route "par défaut" permettant d'atteindre l'extérieur. Le *next hop* pour cette route est le celui permettant d'atteindre le *gateway*. Un exemple de table de routage est donné à la table 4.1. La méthode avec tunnel est transparente pour les noeuds intermédiaires mais elle n'est pas nécessaire si chaque noeud peut différencier les adresses externes des internes.

| **Adresse de destination** | **Adresse du *next hop*** |
|---|---|
| Noeud extérieur : A.B.C.D | DEFAULT |
| DEFAULT | Gateway |
| Gateway | Noeud intérieur : E.F.G.H |

TAB. 4.1 – Exemple de table de routage pour un noeud mobile

Concernant la découverte de routes vers les *gateways*, il existe, à l'instar des protocoles de routage, trois grandes familles de méthodes : proactive, réactive et hybride. Dans la méthode proactive, les *gateways* envoient périodiquement des messages qui transitent sur tout le réseau ad-hoc afin d'informer les autres noeuds de son existence. Ces approches offrent une forte connectivité mais conduisent aussi à une surcharge du réseau due aux messages provenant des *gateways*.

Dans les méthodes réactives, ce sont les noeuds qui, quand ils ont besoin de se connecter à Internet, trouvent un *gateway*. Cette découverte se fait en envoyant un message de sollicitation dans le réseau. Ces approches induisent donc moins de surcharge.

Les démarches hybrides combinent les deux précédentes : pour certains noeuds situés dans un périmètre donné autour des *gateways*, la découverte se fait de manière proactive. Pour les autres, c'est la méthode réactive qui est utilisée.

De par la nature des réseaux ad-hoc, un noeud peut avoir accès à plusieurs *gateways*, en passant par des routes différentes. Si un noeud reçoit des messages provenant de différents *gateways*, il doit choisir auquel se connecter et initier éventuellement une procédure de *handover*. Le critère permettant de décider si un *gateway* est meilleur qu'un autre dépend des implémentations. Le choix peut se baser sur le *hop count*, la distance physique, la qualité du signal, ... . La question du choix de la métrique appropriée pour la sélection de route est au centre d'une multitude de recherches dans les MANETs.

### 4.2.2 Solutions existantes

#### AODV+

Le protocole AODV+ est une extension du protocole AODV, proposée par Ali Hamidian [20, 21], permettant l'intégration de la découverte de *gateway* ainsi que la notion de route par défaut dans les réseaux ad-hoc

Dans AODV+, il y a trois méthodes de découverte de *gateway* :

1. *Proactive Gateway Discovery* : Initiée par le *gateway* lui-même, il *broadcast* (envoie à tout le réseau) périodiquement un message GWADV (*GateWay ADVertisement*) avec une période déterminée par un attribut ADVERTISEMENT_INTERVAL. Un noeud recevant ce message crée (ou *update*) une entrée dans sa table de routage et *rebroadcast* le message. Afin d'être sur que tous les noeuds du réseau reçoivent ce message, le nombre de retransmissions est donné par NET_DIAMETER défini par AODV, représentant la taille du réseau. De plus, chaque noeud attend un temps aléatoire avant de retransmettre le message, afin d'éviter d'être en synchronisation avec les *rebroadcasts* d'autres noeuds, ce qui permet d'éviter des collisions.

   L'avantage de cette méthode est qu'il y a une chance qu'un noeud mobile initie un *handover* avant de perdre sa connexion Internet. Le désavantage est l'envoi périodique du message de contrôle, qui utilise beaucoup de ressources du réseau.

2. *Reactive Gateway Discovery* : Initiée par un noeud mobile désirant entrer en communication avec un *gateway*. Il *broadcast* un RREQ avec un *flag* 'I' (RREQ_I) vers une adresse spéciale : ALL_MANET_GW_MULTICAST (cette adresse correspond aux adresses IP de tous les *gateways* du MANET). Un noeud intermédiaire recevant un RREQ_I ne peut y répondre et se contente de le retransmettre. Quand un *gateway* reçoit ce message, il répond, en *unicast* (envoi à un seul destinataire), un message RREP_I contenant entre autre son adresse IP.

   L'avantage de cette approche est que les messages de contrôle ne sont générés que quand ils sont nécessaires. Le désavantage est que le *handover* n'est initié que quand le noeud perd sa connexion et il est possible qu'un noeud soit en train d'utiliser un *gateway* n'étant pas le plus proche.

3. *Hybrid Gateway Discovery* : Afin de minimiser les désavantages des méthodes précédentes, elles sont combinées dans une méthode hybride. Pour des noeuds situés dans une certaine zone autour du *gateway*, la méthode proactive est utilisée. Les autres noeuds utilisent l'approche réactive.

   Périodiquement, le *gateway* envoie donc son message GWADV mais le nombre de retransmissions est cette fois défini par l'attribut ADVERTISEMENT_ZONE, définissant la portée du message. Quand un noeud situé hors de la zone a besoin d'un *gateway*, il envoie un RREQ_I à ALL_MANET_GW_MULTICAST, comme dans la méthode réactive.

En ce qui concerne le *handover*, quand un noeud reçoit des GWADV de plusieurs *gateways* il se connecte à celui qui est le plus proche, en terme de *hop count*.

### Wireless Infrastructure and Ad-Hoc Network Integration : WIANI

WIANI [22] définit une architecture au sein de laquelle les noeuds peuvent se connecter à un AP en passant par d'autres noeuds et communiquer entre eux sans passer par l'AP. Les AP sont connectés entre eux via un réseau d'infrastructure et le trafic entre les AP passe par celui-ci. Les communications entre les noeuds et les AP se font en mode ad-hoc. La figure 4.3 montre un exemple d'architecture WIANI.

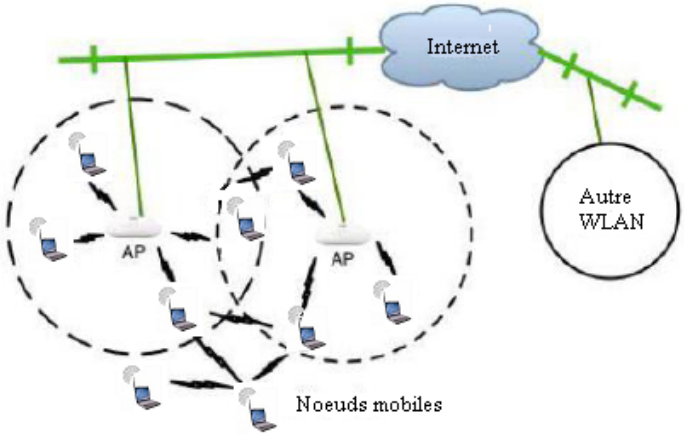


FIG. 4.3 – Architecture WIANI

Les avantages d'une telle architecture sont :

- Extension de la couverture et coût de déploiement réduit : en utilisant le mode ad-hoc, la couverture des AP est en effet augmentée. De plus, moins d'AP sont nécessaires pour couvrir une même zone, d'où la réduction des coûts.
- Amélioration des performances : les noeuds, grâce au mode ad-hoc, ont plus de possibilités de chemins pour envoyer un message à une même destination. Ceci permet aussi à un noeud n'ayant plus beaucoup de batterie d'envoyer un message via un noeud proche plutôt que de devoir faire cela sur une grande distance.
- Meilleure répartition de charge entre les AP.

### Mixed-Mode WLAN :M$^2$WLAN

Le M$^2$WLAN [23] présente une intégration des modes ad-hoc et infrastructure des WLAN. Dans ce réseau (voir figure 4.4), les utilisateurs peuvent communiquer directement avec l'AP (en mode infrastructure) mais aussi initialiser des communications en mode ad-hoc, sous l'administration de l'AP. Un noeud change de mode en fonction des conditions de trafic dans la cellule.

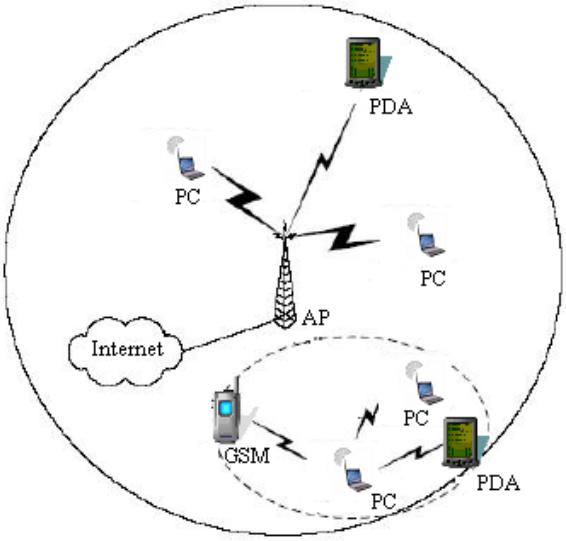


FIG. 4.4 – Architecture M$^2$-WLAN

Dans cette implémentation, c'est l'AP qui gère les modes des différents noeuds. Initialement, tous les noeuds sont dans le mode infrastructure. Si le trafic devient important, l'AP demande à un groupe de noeuds de passer dans le mode ad-hoc Il faut noter que, comme conclusion à ce projet, les auteurs n'ont pas observé d'améliorations importantes liées à l'introduction de connexions ad-hoc.

# Deuxième partie

# Implémémentation du protocole

# Chapitre 5

# Stratégie et outils d'implémentation

Nous allons maintenant décrire la façon dont nous avons implémenté notre protocole, en introduisant d'abord les solutions envisagées, pour ensuite parler de l'outil de simulation utilisé et des améliorations qui y ont été apportées.

## 5.1 Solutions envisagées

Dans le cadre de ce mémoire, nous nous intéressons donc à trouver une solution pour combiner un réseau fonctionnant en mode infrastructure avec un réseau ad-hoc. Pour cela, il faut avoir un protocole de routage adapté à cette situation. Ce protocole doit, de plus, avoir un bon comportement dans des situations *indoor*.

Le premier choix à faire fut celui du protocole de routage sur lequel se baser. En effet, nous avons décidé de ne pas repartir à zéro et créer un tout nouveau protocole mais bien de modifier un protocole existant afin qu'il réponde à nos exigences.

Plusieurs études ont été faites, afin de comparer les différents protocoles de routage [24, 25, 26]. Comme nous l'avons vu, il existe des différences entre les familles de protocoles mais il existe aussi des différences au sein d'une même famille. Notre choix s'est rapidement porté sur AODV. En effet, AODV montre un meilleur comportement en terme de livraison de paquets, à la fois au niveau de la quantité de paquets transmis mais aussi du point de vue du délai moyen de transmission. AODV génère malgré tout une quantité importante de messages de routage quand le nombre de noeuds est important, notamment lors de l'utilisation des messages *Hello*. Les conclusions ressortant de toutes les études sont que AODV est un protocole de routage

ayant un bon comportement dans toutes les situations, notre choix semble donc approprié.

Une fois ce choix fait, il était nécessaire de comprendre comment intégrer les deux types de réseau via ce protocole de routage. Au cours de nos recherches, nous avons trouvé qu'une telle solution avait déjà été proposée, sous le nom AODV+. Le problème d'AODV+ est qu'il ne tient pas compte des conditions spécifiques des situations *indoor*. En effet, son choix de *gateway* ne se base que sur le *hop-count* et, pour citer Douglas S. J. De Couto [27] : "*Shortest Path is not enough*". En effet, dans les situations *indoor*, se baser uniquement sur le *hop count* n'est pas optimal, à cause des problèmes de propagation. C'est pourquoi il a fallu trouver une autre mesure du poids des routes.

Les simulations actuelles dans le domaine des réseaux sans fils, et plus particulièrement celles visant à tester les performances d'un protocole particulier, se contentent de simuler le comportement du protocole en question et de la couche du modèle OSI correspondante. L'aspect des interactions entre les différentes couches est souvent négligé.

L'hypothèse classique est de supposer que les couches inférieures font leur travail correctement et n'interfèrent pas avec le comportement du protocole, ce qui peut ne pas être le cas. Une grande partie des recherches dans le monde des MANETs est dirigée vers l'étude des interactions entre ces couches [6, 7]. Toutefois, ces études sont souvent relatives aux couches directement supérieure ou inférieure. La couche physique est cependant souvent négligée, ce qui n'est pas une bonne solution. En effet, plusieurs études [28, 29] ont montré l'importance de la couche physique sur le comportement du routage. Le calcul du bruit et des interférences est notamment un facteur influençant fortement les communications. Le SIR (*Signal to Interference Ratio*) et le SNR (*Signal to Noise Ratio*) sont tous deux fort corrélés avec le FER (*Frame Error Rate*) sur le canal. La puissance de ces interférences est calculée en sommant les puissances respectives de tous les autres signaux présents. Pour un SNR donné, il existe deux familles de modèles permettant de définir la réception de signaux :

- Modèles *SNR threshold* : utilisent directement la valeur du SNR et la comparent avec un seuil, le SNRT (*SNR Threshold*). Si cette valeur est inférieure au seuil, le signal n'est pas accepté.
- Modèles BER (*Bit Error Rate*) : sur base du BER et de la taille de la trame, les signaux se voient attribuer une certaine probabilité d'acceptation.

Afin de prendre en compte les données de la couche physique, notre idée est d'évaluer la qualité de la connexion entre les noeuds. Pour ce faire, deux démarches nous ont semblés pertinentes : se baser soit sur un calcul du SNR moyen, soit sur un calcul de la puissance nécessaire

pour communiquer entre deux noeuds. Il est apparu que le calcul du SNR nécessiterait la connaissance d'une grande quantité d'information, étant donné qu'il requiert de calculer les puissances de tous les signaux reçus. Nous avons donc opté pour une évaluation de la puissance en calculant :

$$E[P_{i,j}] = E[P_{Tx}] - E[P_{Rx}] \tag{5.1}$$

où :
$E[x]$ = Valeur moyenne de x,
$P_{i,j}$ = Puissance nécessaire pour atteindre le noeud j depuis le noeud i,
$P_{Tx}$ = Puissance d'émission,
$P_{Rx}$ = Puissance de réception.

Pour calculer cette valeur moyenne, il faudra mémoriser une liste de paquets reçus pour chacun des noeuds du réseau.

Cette modification du poids ne se fera que pour les connexions entre les noeuds et les AP car il s'agit de l'opération la plus critique dans un réseau hybride. Concernant les communications inter nodales, le protocole AODV et ses messages *Hello*, combiné avec le protocole de *Link Layer Détection* suffisent à maintenir des performances satisfaisantes.

Le deuxième point à améliorer dans AODV+ est le *handover* (HO). A nouveau, celui-ci n'a lieu que quand un noeud reçoit un message d'un AP plus proche, en terme de *hops*, que celui auquel il est actuellement connecté. Ce qui n'est à nouveau pas optimal en *indoor*. En nous inspirant du HO des réseaux cellulaires, nous pensons nous baser sur une hystérèse pour passer d'un AP à un autre.

Le principe de fonctionnement de cette hystérèse (voir figure 5.1) est le suivant : le *handover* ne se fait que quand la puissance sur la nouvelle station est supérieure d'une valeur H à celle de la station actuelle. Nous allons cependant ajouter une condition supplémentaire à ce *handover*. En effet, supposons que le noeud voulant quitter un AP soit le dernier lien reliant le réseau à cet AP, il pourrait alors être préjudiciable de le quitter. On peut effectivement se demander si cet AP ne permet pas d'atteindre un réseau particulier ou simplement un noeud spécifique, comme un serveur mail.

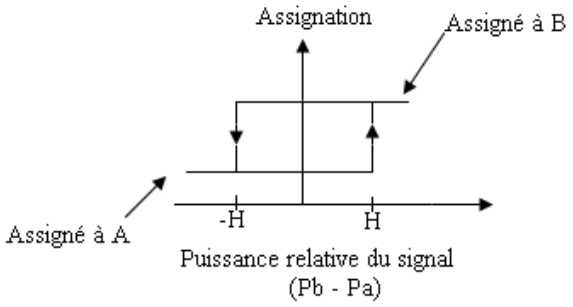


FIG. 5.1 – Hystérèse

Il reste à décider qui, de l'AP ou du noeud, décidera d'entamer une procédure de HO. Si c'est le noeud qui décide, il pourrait y avoir des cas ou il choisit de quitter son AP, car le signal n'est plus assez bon, alors qu'il est le dernier relais vers l'AP en question. Si c'est l'AP qui gère le HO, il faut entamer la procédure quand la communication entre l'AP et le noeud est encore possible. En effet, si l'on s'y prend trop tard, ou si le noeud arrive dans une zone de mauvaise réception, la procédure ne pourra pas être entamée. Dans le cadre de ce mémoire, nous avons décidé de laisser le noeud décider s'il doit effectuer un HO.

## 5.2 Scénario

Nous allons donc nous placer dans le cas d'un réseau ad-hoc *indoor* (dans un bureau, par exemple) relié via des AP au monde extérieur. Il peut évidemment y avoir plusieurs AP qui connectent le réseau à Internet. Comme nous le voyons sur la figure 5.2, il y a deux types de noeuds dans notre scénario : ceux purement ad-hoc (en bleu) et ceux servant de relais entre les AP et le reste du réseau (en rouge). Nous allons considérer comme relais tous les noeuds se trouvant dans la zone de couverture d'un AP. Ce sont ces éléments qui ont la tâche de communiquer directement avec les AP. Tout ces noeuds peuvent soit communiquer entre eux, soit établir une connexion avec Internet. Pour cela, ils ont besoin de passer par un AP et donc de trouver une route vers celui-ci, par le biais d'un relais.

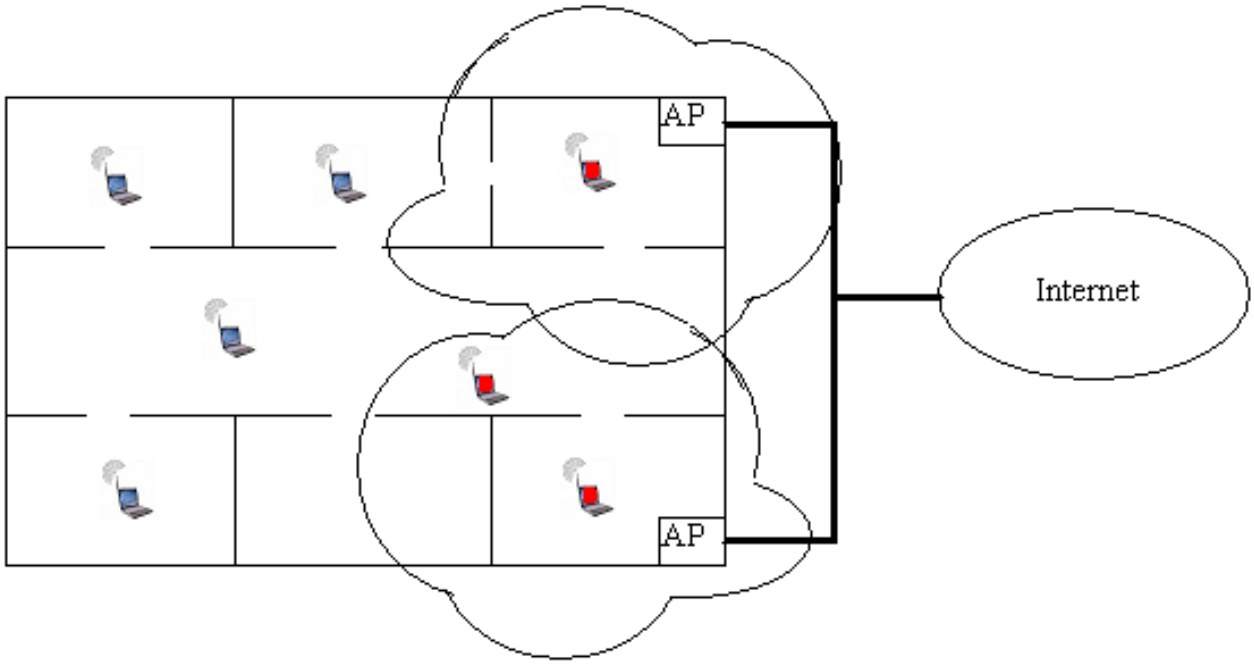


FIG. 5.2 – Scénario envisagé

## 5.3 Techniques de simulation

Plusieurs recherches [30, 31] introduisent les difficultés auxquelles nous sommes confrontés quand il s'agit de simuler des réseaux, et plus particulièrement Internet.

La premier problème est l'hétérogénéité. En effet, dans les réseaux informatiques, il existe une multitude de protocoles, de trafics, d'applications, etc. et les possibilités de combinaison sont nombreuses. La deuxième est l'évolution, les réseaux tendant à devenir de plus en plus grands. De plus, pour les réseaux sans-fils, la topologie évolue au cours du temps.

Dans le cadre de ce mémoire, nous avons utilisé le simulateur NS2 (*Network Simulator 2*). NS2 est un logiciel de simulation de réseaux informatiques qui est devenu la référence dans ce domaine. C'est un logiciel dans le domaine public, disponible sur Internet et dont l'utilisation est gratuite. Le logiciel est exécutable tant sous Unix que sous Windows.

### 5.3.1 NS2

Au départ, la version 1.0 de NS a été développée au Laboratoire National de Lawrence Berkeley (LBNL) par le groupe de recherche réseau. Son développement fait maintenant

partie du projet VINT (*Virtual InterNetwork Testbed*) qui a pour but la construction d'un simulateur réseau offrant des outils et des méthodes novatrices, dans un environnement proche de la réalité. Ce simulateur essaie de répondre aux questions de mise à l'échelle (simulation de grandes topologies) et d'interaction entre divers protocoles.

La dernière version de NS-2, la 2.29, date d'octobre 2005. Les sources sont disponibles sur le site de l'ISI (*Information Science Institute*), dans la section "nsnam". Ces sources se présentent sous deux formes : l'une dite "*all in one*" qui contient le code NS-2 ainsi que tous les autres composants nécessaires (comme OTcl, NAM,...), soit par morceaux. Le *package* comprend aussi des exemples de simulations ainsi que des modèles de mouvement (pour les noeuds mobiles) ou de génération de trafic.

L'architecture réseau de NS-2 est basée sur le modèle des couches OSI (voir section 3.2.1). NS2 est un simulateur à évènements discrets, chaque activité physique sur le réseau est traduite en un évènement qui est mis en file d'attente. A chaque évènement est attribué un instant de traitement, permettant d'ordonnancer cette file.

NS-2 est écrit en C++ et en OTcl (*Object Tool Command Language*) et ces deux parties sont étroitement liées. Quand l'utilisateur crée un nouvel objet via l'interpréteur OTcl, un objet correspondant est aussi crée dans la hiérarchie C++. Bien entendu, les objets peuvent être manipulés aussi bien en OTcl qu'en C++, grâce à la mise en place de procédures de liaison entre les deux langages. La partie codée en C++ est rapide à exécuter mais plus lente à modifier et est utilisée pour l'implémentation des protocoles. Celle en OTcl est, quant à elle, plus lente à l'exécution qu'elle ne l'est à modifier, ce qui la rend optimale pour la configuration des simulations. Un des avantages d'une telle combinaison de langages est qu'elle permet de générer rapidement des scénarios à très grande échelle. L'inconvénient majeur étant que, pour modifier et étendre le simulateur, il faut savoir comprendre et débugger les deux langages.

Il y a trois outils primordiaux dans la familiarisation avec NS2 :

1. *Marc Greis's Tutorial* : le but de ce tutoriel est de guider les premiers pas des nouveaux utilisateurs de NS2. Il introduit l'utilisation du simulateur, indique comment créer ses propres simulations et comment ajouter des nouvelles fonctionnalités.
2. *NS by Example* : ce document énonce les idées à la base du simulateur, explique comment configurer celui-ci pour effectuer des simulations et donne des références afin de trouver des informations supplémentaires. Après le *Marc Greis's Tutorial*, c'est le texte le plus intéressant à lire.

3. *NS Mailing List* : permet de trouver les réponses au questions que l'on se pose encore. Il existe en effet des archives de toutes les questions posées sur cette mailing list et des nouvelles questions peuvent toujours y être posées.

## 5.4 Apports

Comme expliqué précédemment, notre principal apport est le calcul de la puissance moyenne afin de choisir le meilleur AP possible. A cette fin, il a fallu trouver une méthode permettant de mémoriser à la fois les AP connus et les paquets reçus de ces derniers. Le choix que nous avons fait est d'utiliser des listes. En effet, nous n'avons pas besoin de mémoriser une grande quantité d'information et la création d'une nouvelle classe dans NS2 ne semblait pas nécessaire, nous avons donc crée deux nouvelles structures de données. Deux listes ont été ajoutées à chaque noeud : une comprenant les informations des AP qu'il connaît et l'autre contenant, pour chaque AP, les informations relatives aux paquets reçus de cet AP. La taille de cette seconde liste est limitée à 20, afin de ne pas gaspiller les capacités de mémoire des noeuds.

La figure 5.3, représente les relations entre ces structures. Par souci de clarté, un diagramme de classe est utilisé. Comme l'on peut le voir sur cette figure, chaque noeud contient maintenant un pointeur vers la tête de liste des AP. De plus, il y a pour chaque entrée de la liste des AP deux informations importantes : l'identificateur de cet AP et un pointeur vers la tête de la liste des paquets associés à celui-ci. L'identificateur n'est rien d'autre que l'adresse de l'AP dans le réseau. Ce choix est justifié car dans chaque paquet se trouve l'adresse de son expéditeur. De cette façon, il est aisé d'ajouter un nouveau paquet dans la liste appropriée.

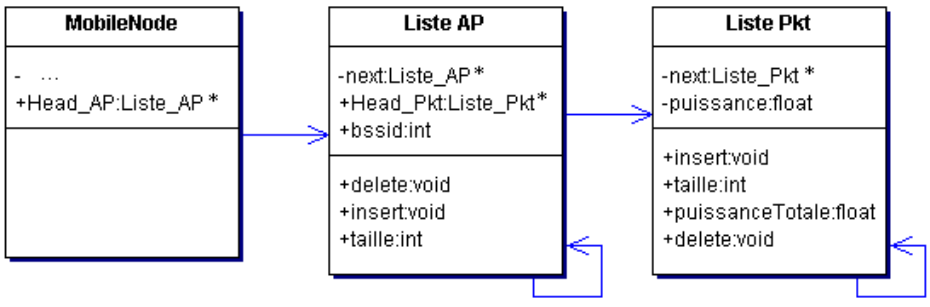


FIG. 5.3 – Représentation des listes

La donnée principale contenue dans la liste des paquets est l'attribut puissance corres-

pondant à $P_{i,j}$ dans l'équation 5.1. Cette valeur est calculée à la réception d'un paquet et est ajoutée dans la liste des paquets. Ce calcul se base sur des données contenues dans le paquet en question. En effet, les puissances auxquelles le paquet a été émis et reçus sont des champs spécifiques contenus dans un paquet. Il faut cependant remarquer que ces puissance, dans NS2, sont exprimées en Watts. A l'émission, cette valeur tourne aux alentours de 0.2818 W. Comme la puissance devient très rapidement petite (la décroissance est exponentielle d'ordre 10), il est plus simple de l'exprimer en dBm. Pour obtenir des dBm, il faut appliquer une conversion du type :

$$P[dBm] = 30 + 10 * log_{10}P[W] \tag{5.2}$$

Une fois ces listes implémentées et la puissance de chaque paquet calculée, il faut définir la fonction qui calculera la puissance moyenne. Il suffit de parcourir la liste des paquets, d'additionner leurs puissances respectives et de diviser cette somme par la taille de la liste.

Il faut maintenant parvenir à utiliser cette nouvelle information dans le processus de routage. Pour ce faire, il faut modifier la table de routage des noeuds du réseau, en lui ajoutant un champ qui contiendra la puissance moyenne calculée. C'est en effet dans la table de routage que sont situées toutes les informations nécessaires au routage des paquets. La table 5.1 montre la forme de la table de routage après modification. Comme nous pouvons le voir, cette table contient toutes les informations nécessaires pour atteindre une destination : l'adresse du noeud suivant par lequel passer, le temps de validité de la route (*Expires*), la longueur de celle-ci (*Hop count*) et, après nos modifications, la puissances moyenne nécessaire pour communiquer avec la destination. Cette dernière information nous permettra de prendre des décisions de routage en utilisant des informations sur la qualité des liens et donc de ne plus se contenter de la longueur du chemin.

| N° de séquence | *Hop count* | *Expire* |
|---|---|---|
| Adresse de la destination | | |
| Adresse du *next hop* | | |
| ... | | |
| Puissance moyenne | | |

TAB. 5.1 – Table de routage après modifications

Un autre ajout qu'il a fallu apporter à AODV+ est un *Timer*. En effet, la question s'est posée de savoir comment réagir quand on ne reçoit plus d'information d'un certain AP. Il est

évident, compte tenu du comportement de propagation *indoor*, que l'on ne peut pas supposer être sorti de la zone de l'AP en se basant uniquement sur le fait de ne pas avoir reçu de GWADV dans le temps imparti. C'est pourquoi un nouveau *Timer* a été implémenté. Ce *Timer* permet de vérifier le temps écoulé depuis la réception du dernier GWADV et, une fois ce temps égal à trois fois le temps ADVERTISEMENT_INTERVAL, on supprime un paquet de la liste. Cette suppression périodique permet de tenir compte de la mobilité en milieu *indoor* et donc de la dégradation du signal qui peut s'en suivre, mais aussi de laisser la possibilité à un noeud sortant de la zone de couverture d'un AP de s'y reconnecter s'il la rejoint rapidement. Un choix que nous avons dû faire, est celui de la méthode de découverte de *gateway*. Comme nous l'avons, vu, il existe trois techniques dans AODV+, chacune ayant des avantages et des inconvénients :

- La méthode proactive est très utile quand on veut que le noeud choisisse lui même quel AP utiliser. Cette méthode consomme bien entendu plus d'énergie et plus de temps mais elle est très efficace pour garder les liaisons en *indoor*. En effet, le fait de recevoir des messages périodiques de l'AP permet d'être sûr que ce dernier est encore dans notre champ de réception.
- L'approche réactive est quant à elle très intéressante étant donné qu'elle utilise moins de ressources du réseau. Cependant son inconvénient principal est que, quand un noeud envoie une requête, c'est l'AP le plus proche qui y répondra et, comme nous l'avons vu, cette proximité ne signifie pas que l'AP est le meilleur.
- C'est le mode hybride qui apporte le meilleur compromis. Il est en effet possible de combiner les avantages des deux méthodes précédentes, il suffit pour cela de bien définir quels noeuds recevront les GWADV des AP. Les noeuds fortement éloignés des AP n'ont pas la puissance nécessaire pour communiquer directement avec eux, ils doivent donc passer par des noeuds intermédiaires, et ne doivent pas choisir à quel AP se connecter. Ce sont uniquement les noeuds relais qui doivent être capables de décider si l'AP est optimal.

Les plus importantes modifications du protocole ont été faites pour les noeuds servant de relais avec les AP. Trois étapes ont dû être implémentées : comment un noeud devient relais, comment il cesse de l'être et comment un relais gère le *handover*.

### 5.4.1 Relais

Nous avons choisis de limiter l'envoi des GWADV aux seuls noeuds se trouvant à l'intérieur de la zone de couverture de l'AP. Cela peut se faire simplement en modifiant le paramètre ADVERTISEMNET_INTERVAL dans AODV+. Les messages provenant des AP ne seront plus transmis que sur une distance d'un saut. Un noeud recevant un tel message devient relais, sauf dans le cas où il occupe déjà cette fonction pour un autre AP. Dans ce cas, il doit tester s'il doit entamer une procédure de HO (voir point 5.4.2). Tant qu'un noeud sert de relais, il continue à recevoir les messages en provenance des AP et continue à mettre à jour sa table de routage afin de garder une vue consistante du réseau. Quand un noeud ne reçoit plus de message, le *Timer* décrit précédemment se charge d'effacer les paquets un à un de la liste des paquets. A chaque suppression, la table de routage est également mise à jour.

Deux cas peuvent alors se présenter : soit la puissance du lien vers cet AP devient plus faible que celle vers un autre AP et une procédure de HO est initiée, soit cette puissance tombe en dessous d'un seuil, fixé à 10 dB, et le relais quitte sa fonction pour devenir un noeud classique. Cette valeur de 10 dB est le seuil de SNR (*Signal to Noise Ratio*), pour les systèmes 802.11b et le simulateur NS2, en dessous duquel aucune transmission ne peut prendre place. Ce seuil nous permet deux choses :

- rester connecté à l'AP suffisamment longtemps que pour prendre en compte les problèmes dus à la propagation *indoor*.
- quitter l'AP quand la qualité du signal n'est plus suffisante.

### 5.4.2 Fonctionnement du handover

Intéressons nous de plus près à la procédure de *handover*. Le HO se déroule comme suit : à la réception d'un message GWADV, le noeud va vérifier si l'émetteur appartient à sa liste d'AP connus. Pour cela, il va récupérer l'adresse de l'AP et vérifier si celle-ci se trouve dans la liste. Si ce n'est pas le cas, il ajoute cet AP dans sa liste. De plus, à chaque fois qu'un paquet provenant d'un AP connu est reçu, celui-ci est ajouté à la liste des paquets, après avoir calculé sa puissance. Une fois cet ajout effectué, l'évaluation de la valeur moyenne est faite et l'entrée correspondant à l'AP dans la table de routage est mise à jour. C'est ici que l'on va tester si l'on doit effectuer le HO : si le paquet reçu provient d'un AP différent de celui avec lequel le noeud est actuellement connecté, nous allons alors comparer les valeurs de leur puissances moyennes respectives. Si cette différence est supérieur à 10 dBm, le HO peut se

faire, à condition que le noeud en question ne soit pas le dernier noeud connecté à cet AP.

Le HO peut se faire dans les cas suivants :

- Quand la qualité du lien avec l'AP actuel se dégrade.
- Quand le noeud perd la connexion avec l'AP actuel. Le *Timer* décrit plus haut aura aussi pour conséquence de diminuer le poids de la liaison.
- Quand la qualité du lien avec le nouvel AP augmente.

Il ne faut pas oublier que le noeud doit vérifier s'il peut quitter l'AP auquel il est actuellement connecté. Nous avons vu que, dans certains cas, il peut être important de rester connecté à un AP. Cette vérification se fait en comptant le nombre de noeuds connectés à l'AP. Dans NS2, les noeuds connaissent l'adresse de leur AP et chaque noeud connaît la liste de ses voisins. Dans NS2, un AP est un noeud mobile comme les autres ayant la particularité d'être son propre AP. Il suffira de compter les voisins de l'AP pour savoir si celui-ci n'a plus qu'un seul noeud avec qui communiquer. Dans ce cas, le HO ne se fera pas.

Les figures 5.4 et 5.5 montrent les fonctions implémentées afin de réaliser ce HO. On peut voir, sur ces figures, que nos modifications on été faites dans la classe AODV de NS2 et pas dans la classe *MobileNode*. Ce choix s'explique par le fait que les informations dont nous avons besoin pour le routage sont des propriétés *private* de la classe AODV et ne sont donc pas accessibles via la classe *MobileNode*. De plus, il y a une correspondance entre les deux classes et à chaque instance de la classe AODV correspond un et un seul noeud mobile.

Analysons les différentes fonctions ainsi implémentées :

- 1 : recvAdvertisement : c'est cette fonction qui est appelée dans la classe AODV quand un message de type GWADV est reçu par un noeud. C'est au sein de cette fonction que sont gérées les connexions avec les AP.
- 1.1 : GetAPAddress : afin de vérifier si l'AP appartient à notre liste d'AP, il fallait récupérer son adresse. Pour cela, nous allons récupérer l'émetteur du paquet qui vient d'être reçu.
- 1.2 : isIn : cette fonction va parcourir notre liste d'AP et vérifier si l'adresse que l'on vient de récupérer correspond à celle d'une des entrées de la liste.

Si l'AP n'appartient pas à la liste, il faut effectuer les étapes suivantes :

- 1.3.1 : puissance : fonction calculant la puissance nécessaire pour communiquer entre l'AP et le noeud en question. Les informations utiles à ce calcul, à savoir, les puissances d'émission et de réception, se trouvent dans la classe *Packet*.
- 1.3.1.1 : insert : fonction qui ajoute l'AP en tête de la liste.

– 1.3.1.2 : insertPkt : afin d'ajouter le paquet reçu dans la liste correspondant au bon AP, il faut invoquer cette fonction, qui se chargera d'insérer le paquet au bon endroit (1.3.1.2.1 : insert ).

Les actions à entreprendre dans le cas où l'AP se trouve dans notre liste d'AP connus sont similaires, nous n'allons détailler que les différences :

– 1.4.1 : getSize : rappelons que nous avons décidé, pour des raisons d'économie de mémoire, de limiter la taille de la liste des paquets à 20. Nous devons donc, avant d'ajouter un nouveau paquet à la liste, vérifier la taille de celle-ci, via la fonction getSize.
– 1.4.3.1 : removePkt : si la taille atteint 20, il faut enlever un paquet de notre liste avant d'en ajouter un nouveau.

Il faut maintenant parler des fonctions propres à la mise à jour de la table de routage :

– 1.5 : puissanceTotale : cette fonction va calculer la puissance moyenne des paquets se trouvant dans la liste des paquets de l'AP qui vient de nous envoyer un message. Pour cela, il suffit de parcourir la liste, en faisant la somme des puissances de chaque paquet et en divisant cette valeur par la taille de la liste.
– 1.6 : update : il faut finalement mettre à jour la table de routage. Après avoir été chercher l'entrée correspondant à l'AP, il faut modifier la valeur de la puissance moyenne précédemment enregistrée dans la table.

Il ne reste plus qu'à déterminer si un HO doit être initié. La séquence de fonctions appelées se trouve sur la figure 5.5.

– 1.1 : comparaisontMetriques : Cette fonction va consulter la table de routage, afin d'y trouver deux informations : la puissance moyenne de l'AP auquel on est actuellement connecté et celle de l'AP qui vient de nous envoyer un message. Si celle du nouvel AP est supérieure à l'autre de 10dBm, le HO est possible.
– 1.2.1 : getPermission : il faut encore vérifier auprès de l'AP actuel si le noeud peut le quitter. Cette requête se fait via l'AP, en le faisant compter le nombre de voisins qu'il a.
– 1.2.2.1 : set_base_stn : une fois que le HO est accepté, il faut associer le noeud au nouvel AP, via cette fonction.

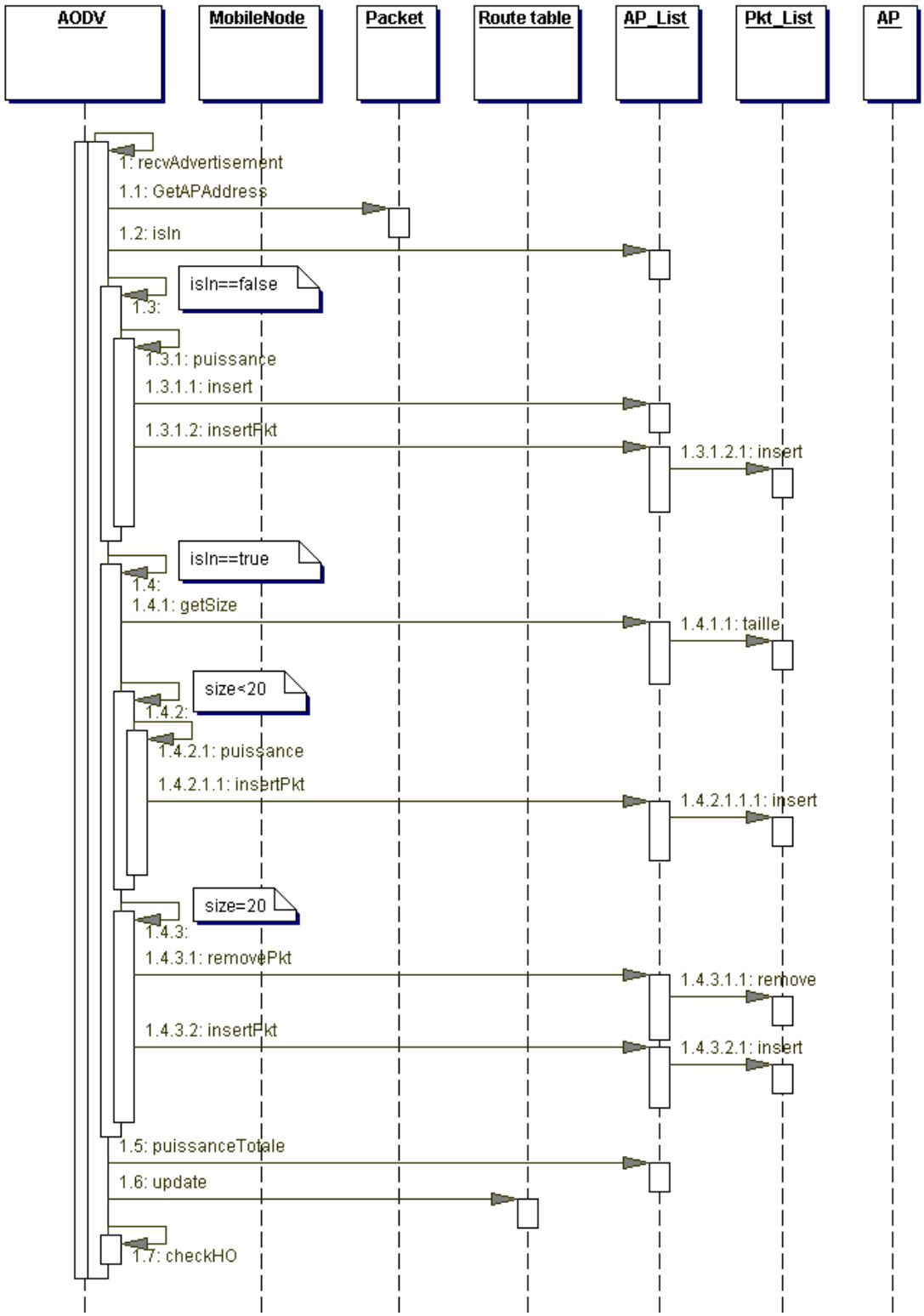


FIG. 5.4 – Diagramme de séquence (1)

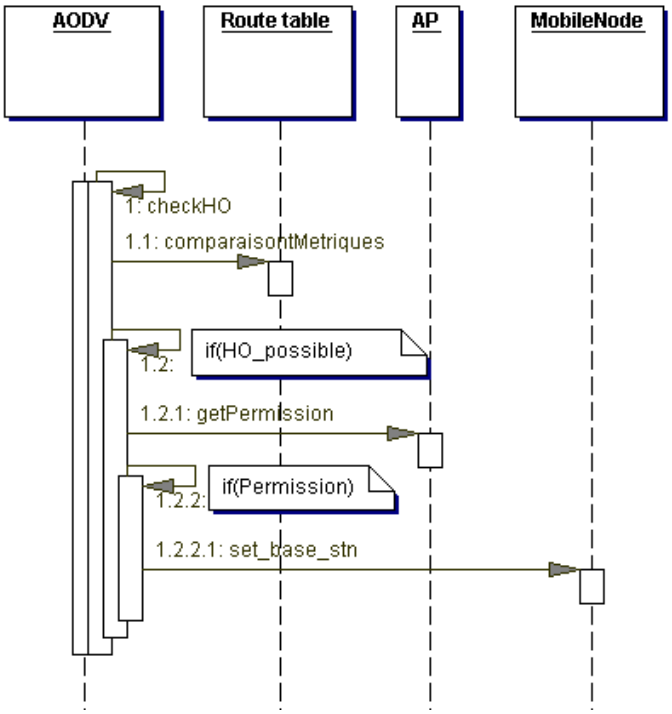


FIG. 5.5 – Diagramme de séquence (2)

# Chapitre 6

# Simulations

Afin de tester le comportement du protocole ainsi implémenté, plusieurs simulations ont été faites dans des conditions différentes. Avant de détailler nos simulations, nous allons introduire les paramètres communs à celles-ci. D'autre part, un certain nombre d'éléments demandent à être définis.

## 6.1 Paramètres de simulation

Dans nos simulations, l'environnement sera un bâtiment de 1500 x 300 m, dans lequel se trouveront 50 noeuds. Ceux-ci seront soit des noeuds mobiles, soit des noeuds fixes reliés aux AP, en fonction de la simulation. Concernant le nombre d'AP, celui-ci variera également au cours de nos simulations, allant de 0 à 3, en fonction du comportement que nous désirons faire ressortir. Nous ferons des simulations d'une durée de 1200 secondes. Ces paramètres peuvent êtres trouvés dans la table 6.1.

| **Paramètre** | **Valeur** |
|---|---|
| Nombre de noeuds | 50 |
| Temps de simulation | 1200 s |
| Topologie | 1500x300 m |

Tab. 6.1 – Paramètres de simulation

De plus, il faut noter que NS2 permet de générer aléatoirement le mouvement des noeuds, ainsi que le trafic entre ceux-ci. Pour chacune de nos simulations, nous avons donc créé des

trafics et des mobilités spécifiques.

### 6.1.1 Mobilité des noeuds

L'impact du mouvement des noeuds sur une simulation est bien entendu très important. En effet, la vitesse, la direction et la fréquence du mouvement ont un impact considérable sur la transmission d'information. Dans nos simulations, nous nous sommes basés sur le *random waypoint model*, utilisé pour la première fois par Johnson et Maltz [32]. Ce modèle est devenu par la suite le standard dans la recherche sur les réseaux sans fils.

Dans ce modèle, les noeuds choisissent une destination au hasard et s'y rendent à une vitesse comprise dans l'intervalle $]0, V_{max}]$, où $V_{max}$ est la vitesse maximale de la simulation. En effet, il est possible de générer une mobilité en spécifiant deux paramètres : $V_{max}$ et le temps de pause. Ce temps de pause est le temps pendant lequel les noeuds restent immobiles, entre chaque mouvement.

Dans nos simulations, nous avons fixé la vitesse maximale à 3m/s. Le temps de pause est, quant à lui, sauf mention explicite, de 300s. Ces paramètres sont repris à la table 6.2.

| **Paramètre** | **Valeur** |
|---|---|
| Nombre de noeuds | 50 |
| Temps de pause | variable |
| Vitesse maximale | 3 m/s |

TAB. 6.2 – Paramètres de mobilité

### 6.1.2 Trafic entre les noeuds

Le paramètre suivant à définir est l'établissement des différents trafics entre les noeuds du réseau. Dans NS2, il existe un générateur de trafic permettant de créer deux types de trafics différents : TCP (*Transport Control Protocol*) ou CBR (*Constant Bit Rate*).

Le protocole TCP n'est pas approprié dans le cas des MANETs [33]. En effet, TCP garantit la livraison des paquets dans l'ordre dans lequel ils ont été envoyés et introduit un mécanisme de renvoi des paquets quand ceux-ci sont perdus. TCP a été développé pour assurer la bonne performance des réseaux fixes, pour lesquels la probabilité de perte de paquets est faible. De plus, dans ce type de réseau, la perte de paquets est principalement due à la congestion du

réseau. Avec TCP, quand un paquet est perdu, le mécanisme de contrôle de congestion est initié, réduisant la vitesse d'envoi des paquets.

Le problème avec les réseaux mobiles est que les pertes de paquets sont essentiellement dues aux cassures de liens, et non à la congestion. TCP interprète souvent les pertes de paquets causées par ces cassures comme de la congestion, ce qui conduit à une réduction du débit et affecte ainsi les performances du réseau (en invoquant les mécanismes de contrôle de congestion quand cela n'est pas nécessaire). De nombreuses études sont faites afin d'améliorer le comportement de TCP pour les réseaux ad-hoc [34, 35, 36].

Pour cette raison, nous nous sommes basés sur le protocole CBR dont le fonctionnement est assez simple [37] : les paquets ont une taille fixe et sont envoyés à un rythme continu, l'intervalle d'envoi entre deux paquets est constant. De plus, la source d'un message n'essaie pas de savoir si son paquet a bien été reçu.

Afin de créer un trafic dans le réseau, certains paramètres doivent être spécifiés : le nombre de noeuds, le nombre maximal de connexions ainsi que la fréquence d'envoi. Dans nos simulations, la taille des paquets est fixée à 512 bytes et la fréquence d'envoi est de 5 paquets/secondes. Le débit de chaque sources est donc de :

$$Debit = 5 * 512 * 8bit/sec = 20kbit/sec \tag{6.1}$$

Ces paramètres sont résumés à la table 6.3.

| **Paramètre** | **Valeur** |
|---|---|
| Type de trafic | CBR |
| Nombre de noeuds | 50 |
| Nombre de sources | variable |
| Taille des paquets | 512 bytes |
| Fréquence d'envoi | 5 paquets/sec |

TAB. 6.3 – Paramètres de trafic

### 6.1.3 Propagation

Il y a trois modèles de propagation implémentés dans le NS2 [38, 2] : *Free space*, *Two-ray ground* et *Shadowing*. Ces modèles sont utilisés pour prédire la puissance avec laquelle chaque paquet sera reçu. Au niveau de la couche physique, il existe une limite inférieure de réception

de message. Si une message arrive à destination avec une puissance inférieure à cette limite, il est considéré comme perdu.

### Free space

Ce modèle idéal suppose que les noeuds participant à la communication se trouvent dans une situation où il n'existe pas d'obstacles entre eux. Le calcul de la puissance à une distance $d$ se fait via l'équation suivante :

$$P_r(d) = \frac{P_e G_e G_r \lambda^2}{(4\pi)^2 d^2 L} \tag{6.2}$$

où :
$P_e$ est la puissance de l'émission du paquet,
$G_e$ et $G_r$ sont les gains des deux antennes,
$\lambda$ est la longueur d'onde,
$L$ la perte dans le système.

Typiquement, ce modèle est représenté par un cercle autour de l'émetteur. Si le récepteur est hors de ce cercle, le paquet est perdu. Ce modèle n'est bien entendu pas réaliste, surtout dans des situations *indoor*, car il n'existe pas toujours de vue directe entre les noeuds communicant.

### Two-ray ground

Ce modèle prend en considération à la fois le signal direct et celui issu de la réflexion sur le sol. Le calcul de la puissance reçue se fait avec l'équation suivante :

$$P_r(d) = \frac{P_e G_e G_r h_e^2 h_r^2}{d^4 L} \tag{6.3}$$

où :
$h_e$ et $h_r$ sont les hauteurs de l'émetteur et du récepteur.
Ce modèle offre de meilleurs résultats que le modèle *Free space* pour des longues distances [39].

### Shadowing

Ce modèle est une version améliorée de la propagation *Free space*, prenant en considération l'existence de trajets multiples entre la source et la destination. Ce modèle introduit une

distance $d_0$ à partir de laquelle les signaux indirects peuvent avoir une interférence destructive sur le rayon direct. La formule du *Shadowing* est :

$$P_{r,Sh}[dBm] = P_{r,Fs}[dBm] - nlog(d/d_0) + N(0,\sigma) \tag{6.4}$$

où :
$P_{r,Fs}$ est la puissance calculée pour le modèle *Free space*(voir l'équation 6.2),
$n$ est appelé exposant de *path-loss*,
$\sigma$ est la déviation de *shadowing*.
Les tables 6.4 et 6.5 donnent les valeurs typiques de $n$ et $\sigma$.

| **Environnement** | | **n** |
|---|---|---|
| *Outdoor* | *Free space* | 2 |
| | Zone urbaine | 2.7 à 5 |
| *Indoor* | Vue directe | 1.6 à 1.8 |
| | Obstacles | 4 à 6 |

TAB. 6.4 – Valeurs typiques de n

| **Environnement** | $\sigma[dB]$ |
|---|---|
| *Outdoor* | 4 à 12 |
| Bureaux | 7 à 9.6 |
| Usines | 3 à 6.8 |

TAB. 6.5 – Valeurs typiques de $\sigma$

Dans nos simulations, nous allons utiliser le modèle de *Shadowing*, avec $n$ et $\sigma$ valant respectivement 3,5 et 3,0.

### 6.1.4 Métriques utilisées

Dans NS2, chaque simulation génère un fichier trace, contenant la liste de tous les paquets envoyés au cours de la simulation. C'est sur base de ces fichiers que nous pouvons analyser les résultats de la simulation. Nous avons décidé de nous baser sur trois métriques pour évaluer le protocole :

– *Routing overhead* : rapport entre le nombre de paquets de routage envoyés et le nombre de paquets reçus dans le réseau. Ce dernier désigne le nombre de paquets utiles réellement reçus par les noeuds du réseau.

$$\eta = \frac{PaquetsDeRoutage}{PaquetsRecus} \tag{6.5}$$

– *Packet Delivery Fraction* : rapport du nombre de paquets envoyés sur le nombre de paquets reçus.

– *End to End delay* : dans les fichiers traces, les instants auxquels les paquets sont envoyés et reçus sont mémorisés. Il suffit donc de faire la différence entre ces deux valeurs et de la diviser par le nombre de paquets.

Nous allons maintenant détailler les simulations effectuées ainsi que les résultats obtenus.

## 6.2 Simulations

### 6.2.1 Influence de l'AP

La première chose dont il fallait s'assurer était que l'ajout d'un AP à un réseau ad-hoc quelconque ne perturbait pas son fonctionnement. Pour cela, plusieurs simulations différentes ont été effectuées. Pour ces simulations, nous avons généré aléatoirement différents trafics CBR (en gardant le nombre de connexions égal à 10) en gardant la même mobilité des noeuds et nous avons analysé nos métriques. Les paramètres utilisés sont repris à la figure 6.1.

1. **Topologie :** simulation dans un environnement de 1500 m x 300m.
2. **Connexions et Trafic** : 10 connexions entre 8 noeuds différents. Trafic de type CBR avec une fréquence d'envoi de 5 paquets/sec.
3. **Noeuds et AP** : 50 noeuds sont déployés dans le réseau, le nombre d'AP varie de 0 à 1. Le temps de pause des noeuds dans leur mouvement est de 300 secondes.
4. **Propagation** : la fréquence du canal est de 2.4 Ghz. Les autres paramètres, comme la puissance du signal sont laissées à leur valeur par défaut. Le modèle de propagation utilisé est le *Shadowing*, avec comme paramètres n = 3.5 et $\sigma = 3.0$.
5. **Simulateur** : ns-2.29 sous Windows XP. Le temps de simulation est de 1200 secondes.

FIG. 6.1 – Paramètres de simulation : Influence de l'AP

Le graphique de la figure 6.2 montre l'évolution du pourcentage de *Routing Overhead* en fonction des différentes simulations. La première chose à remarquer est que, comme nous pouvions nous y attendre, le *Routing Overhead* augmente quand on ajoute un AP. Cet effet est prévisible car cet AP envoie, à intervalle régulier, des messages GWADV aux autres noeuds du réseaux. Ces messages sont des messages typiques du protocole de routage et entrent donc en compte lors du calcul du *Routing Overhead*. De plus, il n'y a pas plus de paquets arrivant à destination qu'avant, étant donné que les trafics sont identiques.

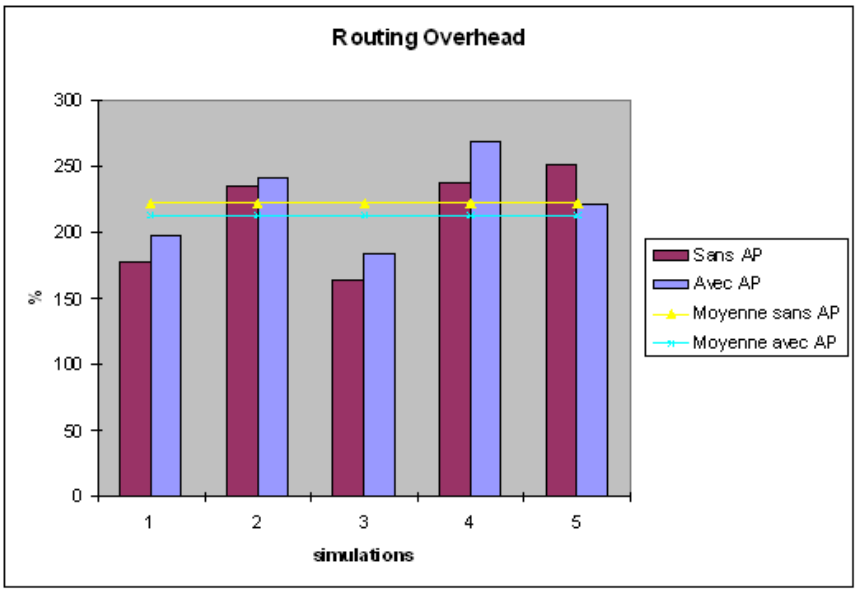


FIG. 6.2 – Influence de l'AP : Routing Overhead

Concernant le pourcentage de PDF (voir figure 6.3), les valeurs avec et sans point d'accès sont quasiment identiques, les variations proviennent probablement du fait que le nombre de collisions dans le réseau a augmenté. Comme nous l'avons vu, le *Routing Overhead* est plus important dans le cas avec AP, ce qui se traduit par la présence d'un plus grand nombre de paquets sur le réseau, augmentant la probabilité de collision entre ces différents paquets.

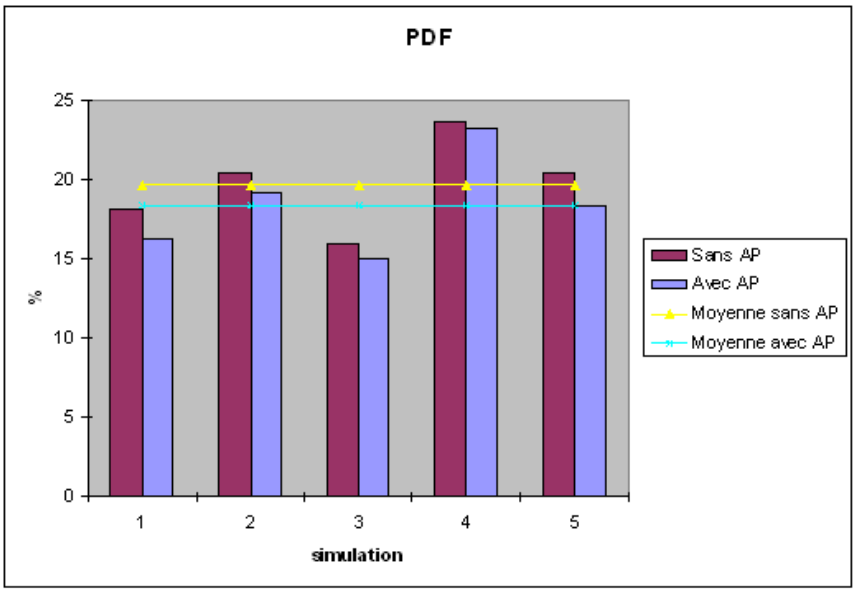


FIG. 6.3 – Influence de l'AP : PDF

Le *End-to-End Delay* moyen est quand à lui inférieur dans le cas des simulations avec AP, comme montré sur la figure 6.4, ce qui signifie que les paquets transmis sont arrivés plus rapidement à leur destination que dans le cas sans AP. Une des explications possibles est que l'AP ajouté participe lui aussi à la transmission des paquets mais uniquement entre ses voisins. Ces transmissions ne passent donc que par un seul noeud intermédiaire, l'AP. Leur délai de transmission est donc moins important que si les paquets devaient emprunter un plus long chemin. L'AP aurait pu devenir un goulot d'étranglement et ainsi perturber le comportement du réseau. C'est pourquoi nous avons regardé débit de l'AP et avons vu que celui-ci n'est pas saturé, l'AP n'est donc pas un goulot d'étranglement.

Nous voyons donc que notre protocole n'induit pas d'influence négative quand on ajoute un AP. De plus, nous voyons que les courbes avec et sans AP sont quasiment identiques pour toutes les simulation. Comme nous pouvions l'espérer, chaque simulation n'a pas un comportement spécifique. Nous allons donc pouvoir, dans la suite de nos analyses, effectuer différentes simulations et effectuer la moyenne des résultats.

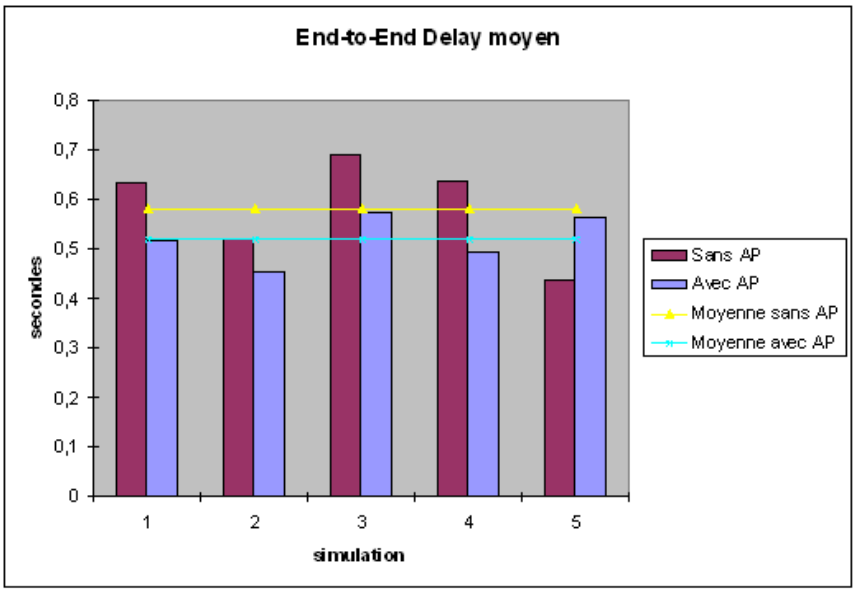


FIG. 6.4 – Influence de l'AP : End-to-End delay

### 6.2.2 Comparaison avec AODV+

Nous avons ici tenté d'évaluer l'apport effectif des modifications apportée à AODV+. Pour cela, nous avons testé le comportement des deux protocoles dans des réseaux comportant plusieurs AP. En effet, c'est dans ces configurations là que nous nous attendons à avoir le plus de différences. Nos modifications sont sensées apporter de meilleurs résultats et ce, de par le fait que nos liaisons avec les AP sont plus robustes. Nous avons effectué nos simulations avec 2 AP et avons fait varier le nombre de noeuds du réseau entrant en communication.

La figure 6.5 montre la différence de *Routing Overhead* entre les deux protocoles. Nous voyons que les valeurs sont plus élevées pour la version non modifiée du protocole. Ceci s'explique par le fait que, grâce aux modifications apportées, les liaisons avec le monde extérieur sont améliorées. La raison principale de cette amélioration est que le HO aura lieu moins souvent. Dans notre version du protocole, ce dernier n'a lieu que quand le signal provenant d'un nouvel AP est meilleur que celui de l'actuel. Moins de paquets seront donc perdus. En effet, si un noeud commence une procédure de HO pendant qu'il est en train d'envoyer des paquets à son AP, ceux-ci sont perdus. Dans nos simulations, la majorité de nos noeuds communiquent avec un noeud extérieur représentant Internet. Ceci implique que, quand les liaisons vers ce noeud sont meilleures, il y a moins de paquets perdus et il y a donc moins de messages d'erreur

de route, de reconstruction de route, etc. qui sont envoyés. Nous voyons ici un des premiers apports concrets de notre protocole.

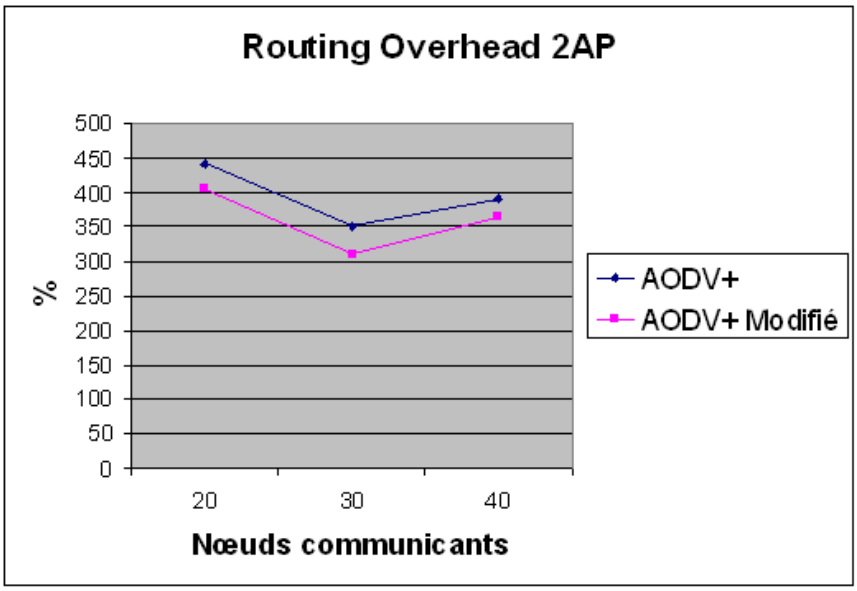


FIG. 6.5 – Comparaison avec AODV+ : Routing Overhead

La figure 6.6, représentant le PDF, met en avant un autre avantage du nouveau protocole, le PDF de la version modifiée est meilleur. A nouveau, cette amélioration s'explique par l'optimisation du choix de l'AP auquel se connecter. En effet, les liaisons étant plus stables, le pourcentage de paquets émis vers l'extérieur et arrivant à destination est plus important. Nous notons une amélioration moyenne de 12% du PDF entre la version non modifiée et la version modifiée du protocole.

Le *End-to-End Delay* est, quant à lui, légèrement inférieur dans la version modifiée du protocole (voir la figure 6.7), ce qui signifie que l'amélioration de la qualité des routes diminue aussi légèrement le délai de transmission.

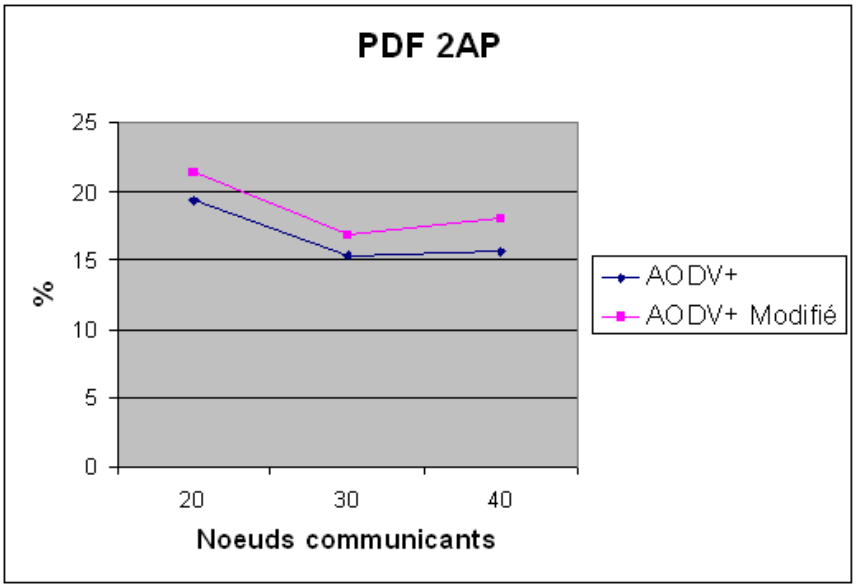


FIG. 6.6 – Comparaison avec AODV+ : PDF

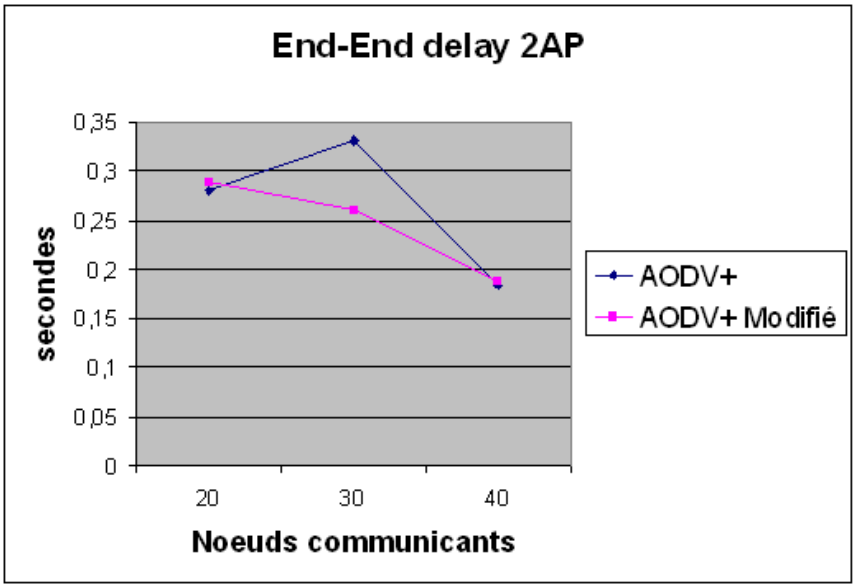


FIG. 6.7 – Comparaison avec AODV+ : End-to-End delay

Nos améliorations ont donc porté leurs fruits. Le protocole modifié se comporte beaucoup mieux en situation *indoor*, ce qui était un des points que nous voulions améliorer.

### 6.2.3 Communication avec Internet

Le but de ces simulations est de voir comment le protocole se comporte en fonction du nombre de noeuds désirant communiquer avec le monde extérieur. Pour cela, nous allons générer deux trafics différents : l'un dans lequel 10 noeuds envoient des messages vers un noeud unique, représentant Internet et un autre avec 40 noeuds. Dans ces simulations, il n'existe qu'un seul AP permettant de se connecter au monde extérieur. Le paramètre que nous allons faire varier est le temps de pause, représentant le temps d'arrêt des noeuds au cours de la simulation. Ces informations sont reprises dans la figure 6.8.

1. **Topologie :** simulation dans un environnement de 1500 m x 300m.
2. **Connexions et Trafic** : Nous avons deux possibilités : 10 connexions entre 10 noeuds du réseau et un noeud représentant le monde extérieur et 40 connexions. Trafic de type CBR avec une fréquence d'envoi de 5 paquets/sec.
3. **Noeuds et AP** : 50 noeuds sont déployés dans le réseau, il y a 1 AP. Le temps de pause des noeuds dans leur mouvement varie : $0, 50, 100, 150, 200, 300, 600$.
4. **Propagation** : la fréquence du canal est de 2.4 Ghz. Les autres paramètres, comme la puissance du signal sont laissées à leur valeur par défaut. Le modèle de propagation utilisé est le *Shadowing*, avec comme paramètres n = 3.5 et $\sigma = 3.0$.
5. **Simulateur** : ns-2.29 sous Windows XP. Le temps de simulation est de 1200 secondes.

FIG. 6.8 – Paramètres de simulation : Communication avec Internet

La figure 6.9 montre deux choses : premièrement, le *Routing Overhead* est sensible au nombre de noeuds communicant avec Internet. Ceci s'explique par le fait que, plus il y a de noeuds désirant communiquer avec Internet, plus grand est le nombre de routes devant être trouvées. Deuxièmement, nous voyons l'effet de la variation du temps de pause sur le *Routing Overhead*. Ce dernier diminue quand le temps de pause augmente car, les noeuds étant moins mobiles, les connexions sont plus stables et il y a moins des messages de routage qui transitent sur le réseau.

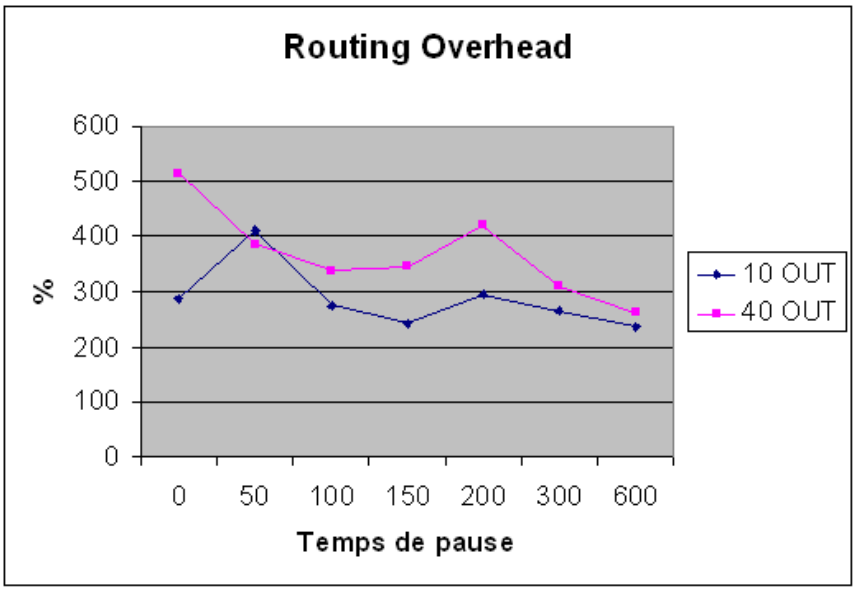


FIG. 6.9 – Communication avec Internet : Routing Overhead

Concernant le PDF (voir figure 6.10), les conclusions à tirer sont doubles également. Le PDF diminue quand le nombre de sources augmente, ce qui s'explique par la présence d'un unique AP. Toutes les transmissions vers l'extérieur passent par ce noeud qui devient donc un goulot d'étranglement. En effet, le risque de collision en ce point du réseau est plus important. D'un autre côté, le PDF diminue avec le temps de pause.

Sur la figure 6.11, on peut voir l'évolution du délai de transmission moyen. Ce dernier diminue avec le temps de pause, ce qui peut être mis en rapport avec la diminution du PDF. En effet, moins de paquets arrivent à destination. La probabilité de perte de paquets est plus grande sur les longues routes et les paquets arrivant à destination sont donc plutôt des paquets effectuant peu de sauts pour atteindre leur destinataire. Leur temps de transmission est donc plus faible.

Nous voyons aussi dans ces graphiques l'influence du temps de pause : pour un temps de pause nul (t=0 s), le réseau est en perpétuelle évolution. Le réseau ne converge donc pas vers une topologie stable, entraînant plus de pertes de paquets et plus de délais car il faut reconstruire des routes fréquemment. Quand le temps de pause augmente (t=600 s), le réseau devient plus stable et les routes, une fois établies, ne changent pas. Cette stabilité permet d'obtenir un meilleur pourcentage de transmission de paquets ainsi que des délais plus faibles.

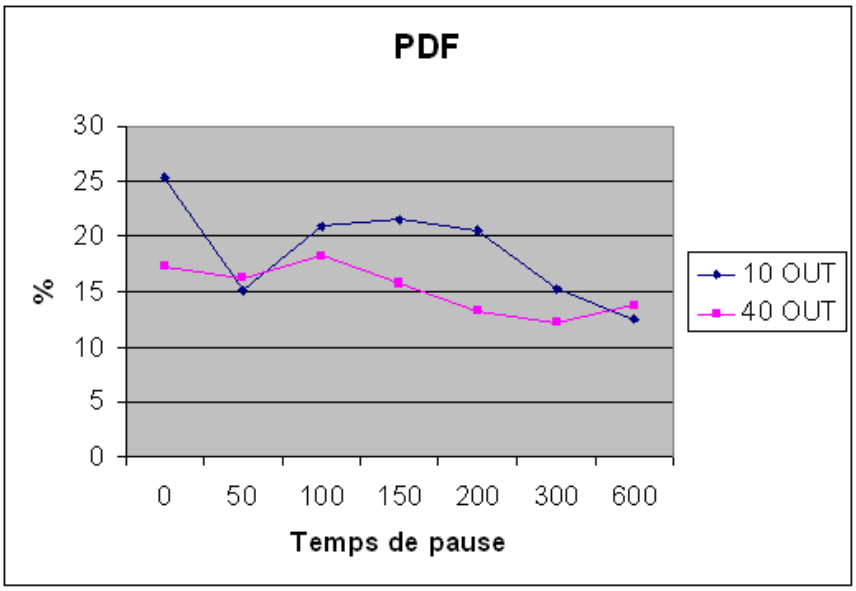


FIG. 6.10 – Communication avec Internet : PDF

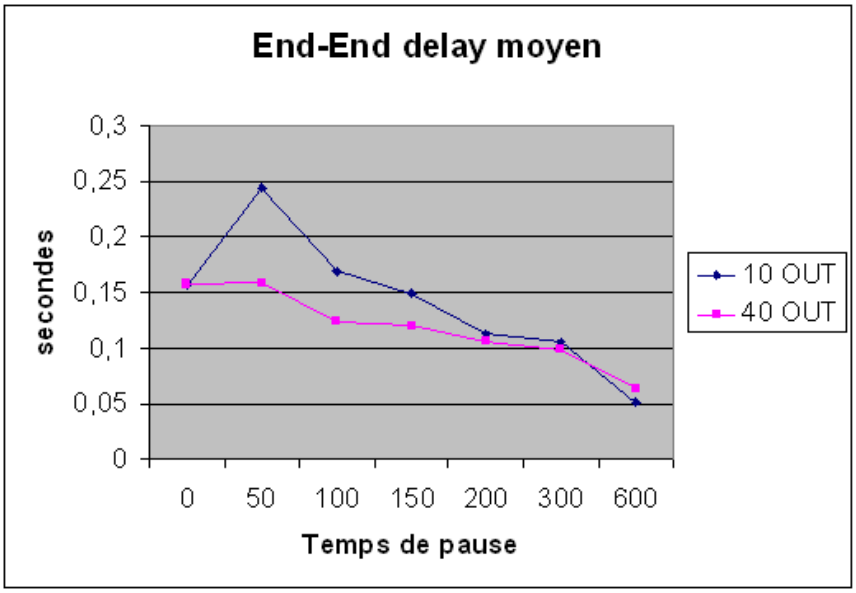


FIG. 6.11 – Communication avec Internet : End-to-End delay

### 6.2.4 Influence du nombre d'AP

Un autre comportement important à tester est celui qui se présente quand plusieurs AP sont présents, rendant notamment le HO possible. Nous faisons donc varier le nombre d'AP

disponibles, de 1 à 3, et ce, pour différents trafics. Les trafics que nous testons ici sont à nouveau des trafics pour lesquels certains noeuds communiquent avec l'extérieur, mais pour un seul temps de pause (ici, t=300 s). Les valeurs données au paramètres sont reprises dans la figure 6.12.

1. **Topologie :** simulation dans un environnement de 1500 m x 300m.
2. **Connexions et Trafic** : Nous avons trois possibilités : $10, 20 ou 40$ connexions entre les noeuds du réseau et un noeud représentant le monde extérieur. Trafic de type CBR avec une fréquence d'envoi de 5 paquets/sec.
3. **Noeuds et AP** : 50 noeuds sont déployés dans le réseau, il y a $1, 2 ou 3$ AP. Le temps de pause des noeuds dans leur mouvement vaut 300 s.
4. **Propagation** : la fréquence du canal est de 2.4 Ghz. Les autres paramètres, comme la puissance du signal sont laissées à leur valeur par défaut. Le modèle de propagation utilisé est le *Shadowing*, avec comme paramètres n = 3.5 et $\sigma = 3.0$.
5. **Simulateur** : ns-2.29 sous Windows XP. Le temps de simulation est de 1200 secondes.

FIG. 6.12 – Paramètres de simulation : Influence du nombre d'AP

En analysant le *Routing overhead* sur la figure 6.13, nous constatons que celui-ci diminue quand le nombre d'AP augmente. Cette diminution s'explique par la présence des AP, créant de nouvelles routes vers l'extérieur. De plus, quand le nombre d'AP augmente, les routes deviennent plus courtes et il y a moins de collisions. Un autre aspect est bien entendu la meilleure qualité de ces routes, nécessitant moins de reconstructions. De plus, nous voyons apparaître une certaine convergence des résultats pour les différentes simulations, quand le nombre d'AP augmente.

La figure 6.14 nous apprend de nombreuses choses sur le comportement du protocole. Tout d'abord, nous voyons que le PDF augmente avec le nombre d'AP présents. Ceci s'explique par le fait qu'il existe plusieurs chemins pour atteindre l'extérieur car il y a plus de noeuds relais au sein du réseau. Ensuite, nous voyons qu'avec 3 AP, les PDF de nos simulations sont quasiment identiques. L'effet de goulot d'étranglement observé précédemment au niveau des AP a disparu, les communications pouvant maintenant passer par 3 noeuds plutôt qu'un seul.

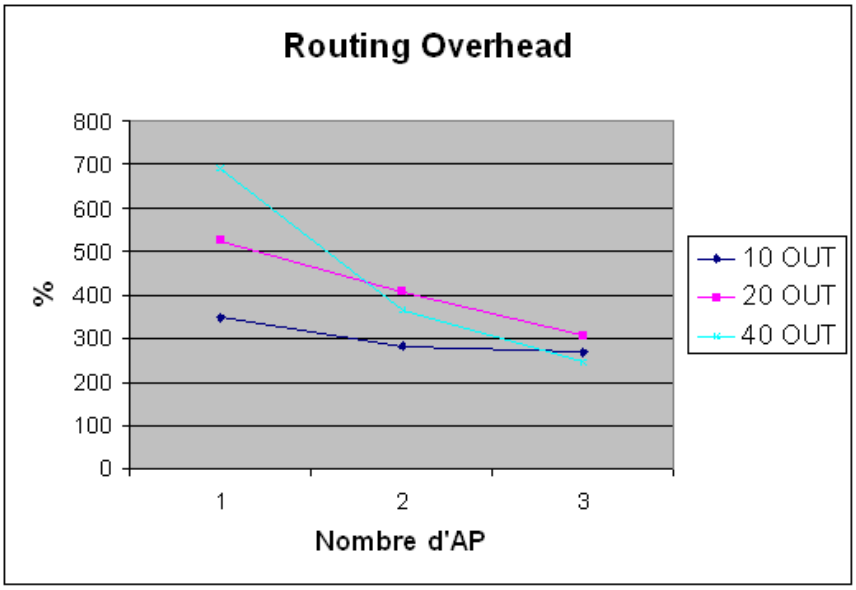


FIG. 6.13 – Influence du nombre d'AP : Routing Overhead

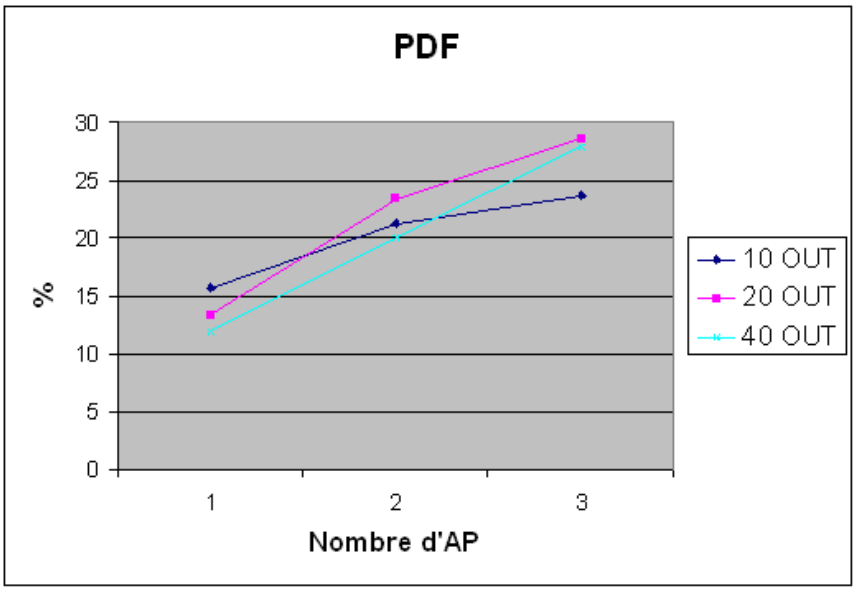


FIG. 6.14 – Influence du nombre d'AP : PDF

Le *End-to-End Delay* moyen décroît, cette diminution peut s'expliquer par la baisse du *Routing Overhead*. En effet, celui-ci étant moins important, il y a d'une part moins de collisions donc moins de retransmissions pour un même paquet et d'autre part, cette diminution traduit une meilleur qualité dans les liaisons inter-nodales, conduisant à une meilleure transmission.

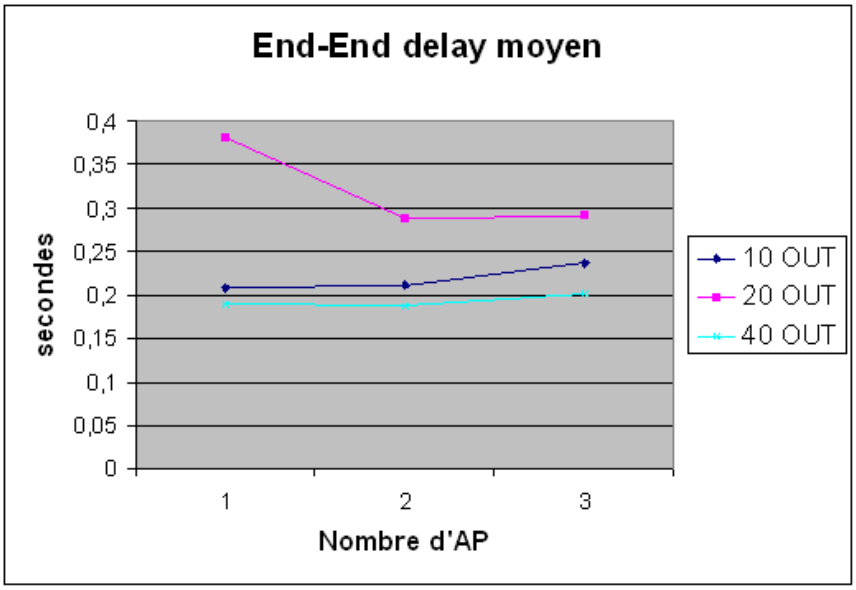


FIG. 6.15 – Influence du nombre d'AP : End-to-End delay

### 6.2.5 Trafic mixte

Après avoir fait tous ces test pour des noeuds ne communicant qu'avec certains types de noeuds, il semblait important de considérer le cas d'un trafic mixte, où chaque noeud est susceptible de communiquer avec tous les autres noeuds.

Comme pour les simulations dans lesquelles les noeuds ne communiquent qu'avec l'extérieur, nous voyons diminuer le *Routing Overhead* avec le nombre d'AP. Cependant, si l'on compare les deux types de simulation, on constate que le RO est plus important dans le cas d'un trafic mixte. Les communications entres noeuds mobiles sont évidemment moins efficaces que celles avec un noeud fixe. En effet, le mouvement relatif des deux noeuds peut conduire à des cassures dans les chemins de transmission et de là, la nécessité d'une plus grande quantité de messages de routage.

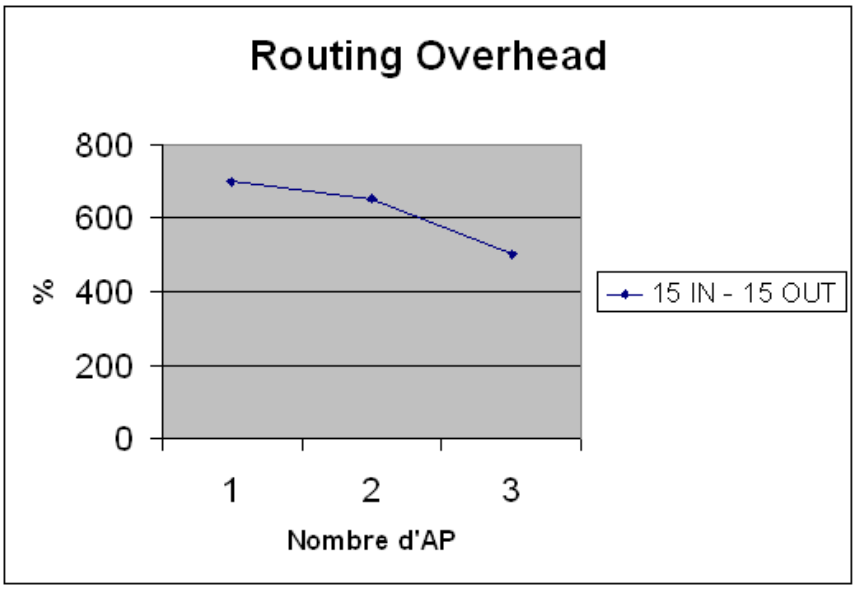


FIG. 6.16 – Trafic mixte : Routing Overhead

Une constatation similaire peut être faite en analysant le PDF (figure 6.17). Le PDF total, bien qu'augmentant grâce à la présence des divers AP, est légèrement inférieur aux cas où les communications sont uniquement dirigées vers l'extérieur. Comme nous l'avons expliqué, la mobilité des noeuds introduit des pertes de paquets et diminue donc le pourcentage de paquets transmis avec succès.

On peut voir sur la figure 6.18 que le *End-to-End Delay* moyen augmente légèrement. Ce comportement peut lui aussi s'expliquer par le fait que, dans ces simulations, nous communiquons avec des noeuds mobiles. En effet, d'après le comportement du PDF et du *Routing Overhead*, nous pourrions nous attendre à voir le délai diminuer. Cependant, de par la nature mobile de nos noeuds, un certain nombre de paquets doivent être renvoyés et prennent donc plus de temps pour arriver à destination.

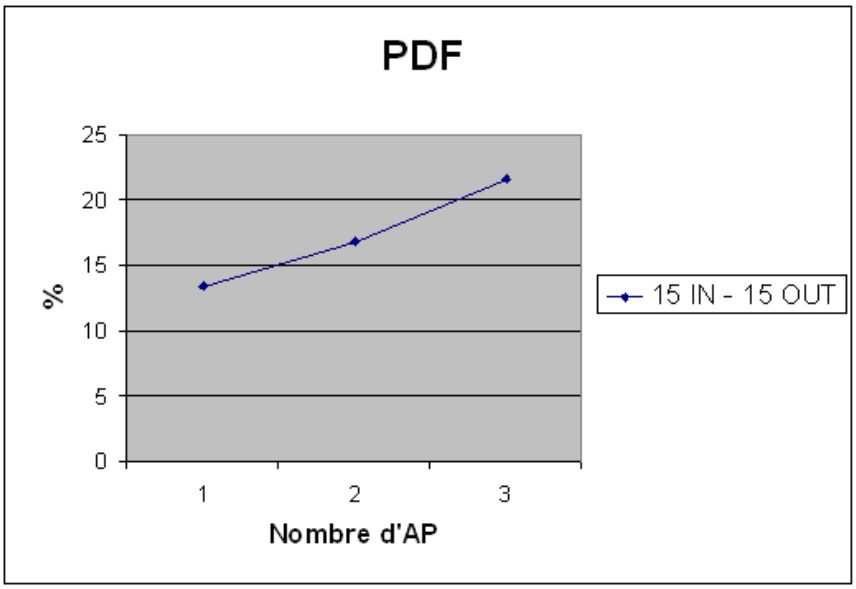


FIG. 6.17 – Trafic mixte : PDF

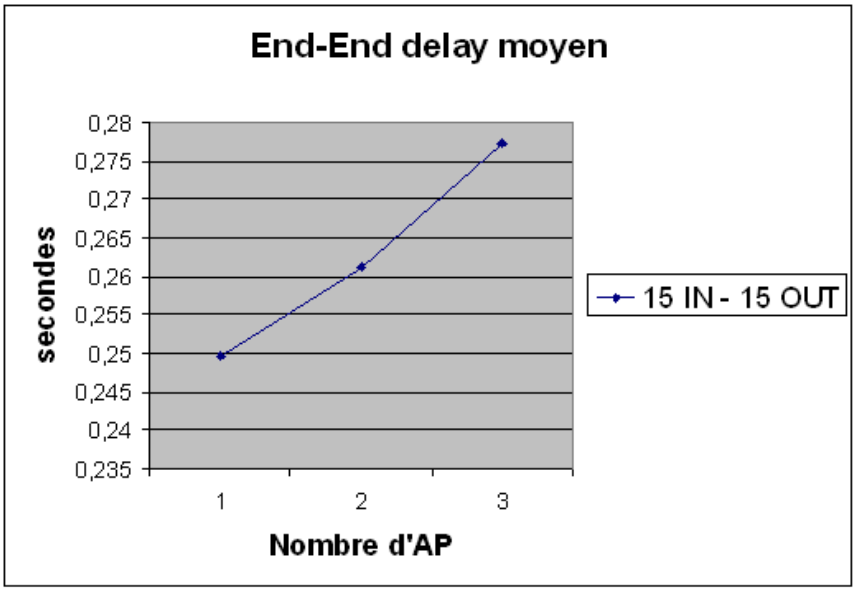


FIG. 6.18 – Trafic mixte : End-to-End delay

### 6.2.6 Influence du rapport ad-hoc/infrastructure

Pour déterminer les conditions optimales d'utilisation de notre protocole, nous avons fait des simulations dans lesquelles nous faisons varier le pourcentage de noeuds fonctionnant en mode infrastructure. Les résultats dans ce domaine sont plus qu'encourageants. Nous avons étudié le comportement d'un réseau constitué de deux types de noeuds, des noeuds ad-hoc et des noeuds infrastructure. Nous avons fait évoluer ces noeuds dans un environnement contenant deux AP différents. La figure 6.19 reprends les principaux paramètres.

1. **Topologie :** simulation dans un environnement de 1500 m x 300m.
2. **Connexions et Trafic** : Nous avons trois possibilités : 25 connexions entre les noeuds du réseau quels qu'ils soient. Trafic de type CBR avec une fréquence d'envoi de 5 paquets/sec.
3. **Noeuds et AP** : 50 noeuds sont déployés dans le réseau, nous faisons varier le rapport de noeuds fonctionnant en mode infrastructure. Il y a 2 AP dans le réseau. Le temps de pause des noeuds dans leur mouvement vaut 300 secondes.
4. **Propagation** : la fréquence du canal est de 2.4 Ghz. Les autres paramètres, comme la puissance du signal sont laissées à leur valeur par défaut. Le modèle de propagation utilisé est le *Shadowing*, avec comme paramètres n = 3.5 et $\sigma = 3.0$.
5. **Simulateur** : ns-2.29 sous Windows XP. Le temps de simulation est de 1200 secondes.

FIG. 6.19 – Paramètres de simulation : Influence du rapport ad-hoc/infrastructure

La figure 6.20 montre l'évolution du *Routing Overhead*. Celui-ci diminue avec l'augmentation du nombre de noeuds en mode infrastructure. Ceci s'explique par le fait que ces noeuds infrastructure ne bougent pas et que les routes vers ceux-ci sont donc plus stables. De plus, comme nous l'avons vu, suite nos apports, les communications avec les AP sont meilleures, rendant les communications beaucoup plus stables.

Concernant le PDF, celui augmente avec la fraction de noeuds infrastructure (voir figure 6.21). A nouveau, cette tendance s'explique par la présence de plus en plus marquée de noeuds immobiles permettant d'arriver à un meilleur pourcentage de paquets livrés.

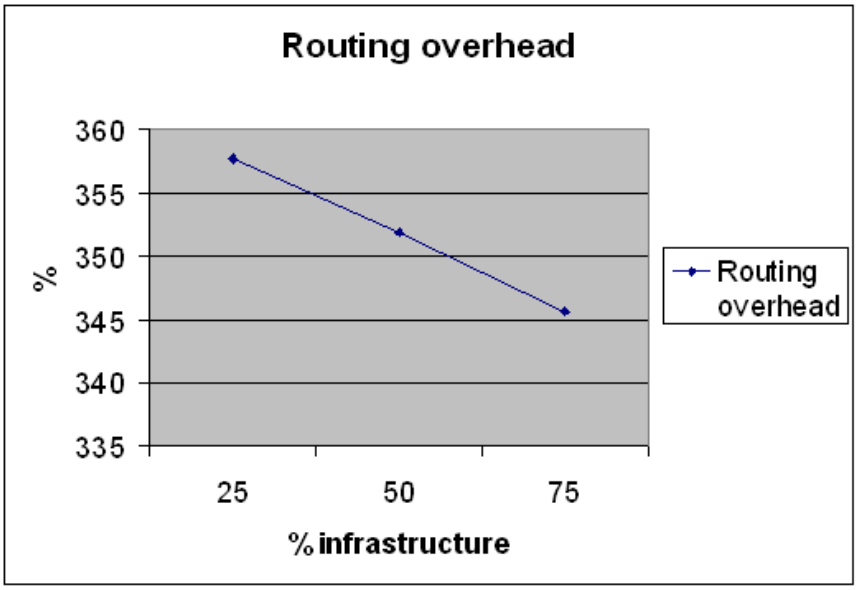


FIG. 6.20 – Influence du rapport ad-hoc/infrastructure : Routing Overhead

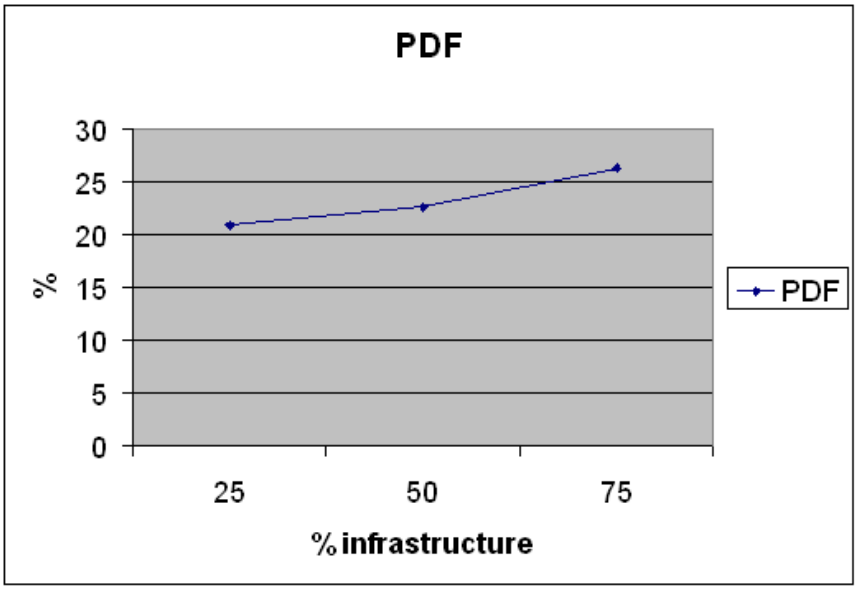


FIG. 6.21 – Influence du rapport ad-hoc/infrastructure : PDF

La figure 6.22 montre que le délai de transmission moyen augmente avec l'augmentation du nombre de noeuds fonctionnant en mode infrastructure. Ceci s'explique par le fait que les noeuds mobiles continuent à participer aux communications mais, ceux-ci étant de moins en moins nombreux, ils sont difficiles à atteindre.

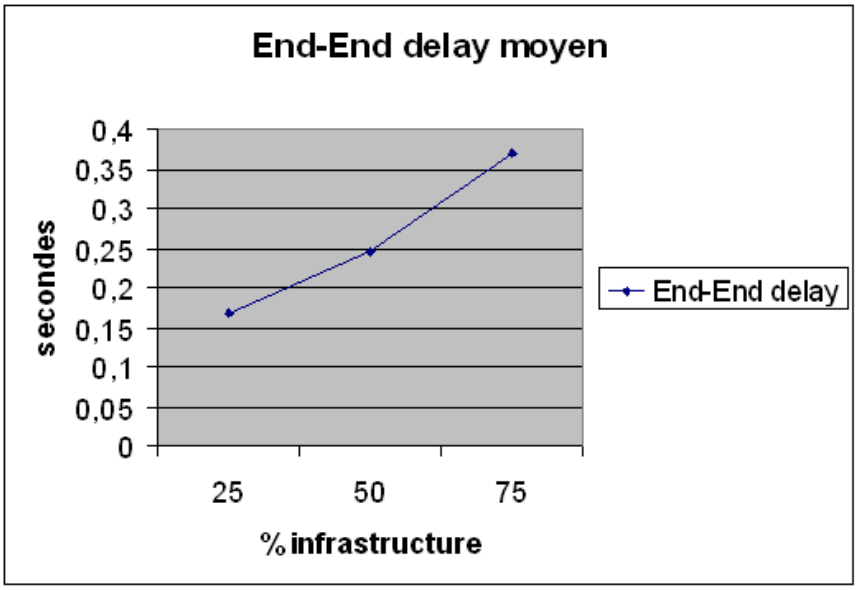


FIG. 6.22 – Influence du rapport ad-hoc/infrastructure : End-to-End delay

# Chapitre 7

# Conclusion

Dans le domaine des réseaux hybrides, utilisant différentes technologies, c'est au niveau des interfaces entre ces diverses composantes que les communications sont les plus critiques. Dans le cas que nous avons étudié, l'intégration des modes ad-hoc et infrastructure, c'est donc au niveaux des points d'accès qu'il faut avoir les meilleures connexions possibles.

Dans le cadre de ce mémoire, nous avons implémenté un nouveau protocole de routage prenant en compte des informations de la couche physique afin d'obtenir de meilleures performances, notamment en milieu *indoor*. Notre idée à été de nous baser sur la puissance nécessaire pour émettre sur une liaison entre deux noeuds, afin de privilégier les connexions les plus robustes. Nous nous sommes basés sur un protocole de routage existant, AODV+, et nous y avons apporté des modifications afin d'améliorer ses performances. Nous avons aussi modifié la façon dont le protocole gère le *handover*, à nouveau dans le but de rendre le réseau plus stable. Nous avons ensuite analysé le comportement de notre protocole grâce à un logiciel de simulation appelé NS2.

Le protocole ainsi implémenté montre de notables améliorations par rapport à AODV+. Nous avons entre autre constaté une hausse de 12% dans la livraison des paquets. De plus, notre protocole montre de très bonnes caractéristiques dans un milieu contenant plusieurs AP, ce qui est habituellement le cas dans des entreprises utilisant le WiFi. Nous entrevoyons ici une des possibilités d'utilisation du protocole : dans un bâtiment dans lequel il existe plusieurs AP, les noeuds fonctionnant en mode ad-hoc permettent une extension du réseau tout en gardant une bonne qualité de transmission de paquets et ce, que les communications soient dirigées principalement vers l'intérieur (Intranet) ou l'extérieur (Internet). Notre protocole

montre aussi de très bons résultats avec une grande proportion de noeuds fonctionnant en mode infrastructure, avec comme seul inconvénient un délai de transmission élevé, excluant des applications de type vidéoconférence, pour lesquels ce délai doit être petit. A nouveau, les bureaux sont propices au déploiement du protocole car le pourcentage d'ordinateurs fixes, utilisant le mode infrastructure, est important. Notons que dans un cas où les noeuds fonctionnant en mode ad-hoc sont majoritaires, comme dans une salle de réunion par exemple, les résultats restent bons, offrant de plus un délai relativement court.

Il reste malgré tout un certain nombre de choses à analyser avant de passer à une implémentation effective. Il faudrait effectuer les simulations dans différentes conditions de propagations, en changeant les paramètres de simulation. De plus, certaines études ont montrés que la couche physique du NS2 n'est pas toujours très réaliste. Il faudrait donc effectuer des simulations en utilisant un outil de *raytracing*, comme celui développé dans le Service d'Informatique et Réseaux. Ces outils offrent plus de précision mais demandent plus de temps de calcul.

La prochaine génération de téléphones cellulaire, la 4G, se basant sur l'intégration de différents types de réseaux est une des principales possibilité d'utilisation d'un tel protocole. Il faudrait maintenant tester ce dernier non plus dans le cadre de réseaux WiFi mais dans celui des réseaux UMTS afin d'analyser le comportement de notre protocole dans le domaine des réseaux cellulaires.

Notre protocole montre donc de très bonnes performances dans les conditions que nous avons analysées. Il reste bien entendu de nombreuses simulation à effectuer mais nous avons ici une très bonne base pour de futures recherches.

# Bibliographie


[1] W. Stallings. *Wireless communication and networks*. Pearson Education Inc., 2nd edition, 2005.

[2] T. S. Rappaport. *Wireless Communications : Principles and Practice*. 2002.

[3] Cours de Réseaux publics de télécommunication, ELEC 321. G. BOCQ, ULB, 2005.

[4] J. Horvth and S. Imre. Examination of the viability of fourth generation mobile networks. *3GIS*, June 2001.

[5] I Chlamtac, M. Conti, and J. J.-N. Liu. Mobile ad-hoc networking : imperatives and challenges. *Ad Hoc Networks*, 1(1) :13–64, July 2003.

[6] P. R. Kumar and V. Kawadia. Principles and protocols for power control in wireless ad hoc networks. *IEEE Journal on Selected Areas in Communications*, pages 76–88, 2005.

[7] P. R. Kumar and V. Kawadia. A cautionary perspective on cross layer design. *IEEE Wireless Communications*, 12(1) :3–11, 2005.

[8] F. Ye S. W. Lu H. Yang, H. Y. Luo and L. Zhang. Security in mobile ad hoc networks : Challenges and solutions. *IEEE Wireless Communications*, 11(1) :38–47, 2004. http ://repositories.cdlib.org/postprints/618.

[9] L. Buttyan J. Hubaux and S. Capkun. The quest for security in mobile ad hoc networks. *Proceeding of the ACM Symposium on Mobile Ad Hoc Networking and Computing*, pages 146–155, 2001.

[10] E.M Royer and C-K Toh. A review of current routing protocols for ad-hoc mobile wireless networks. IEEE Personal Communications, Apr. 1999.

[11] C.E. Perkins and E.M. Royer. Ad-hoc on-demand distance vector routing. in MILCOM '97 panel on Ad Hoc Networks, November 1999.

[12] C.K. Toh. A novel distributed routing protocol to support ad-hoc mobile computing. IEEE International Phoenix Conf. on Computers and Communications.

[13] Z. J. Haas. A new routing protocol for the reconfigurable wireless networks, 1997. In Proc. of the IEEE Int. Conf. on Universal Personal Communications.

[14] Z. J. Haas and M. R. Pearlman. Determining the optimal configuration for the zone routing protocol. *IEEE journal on selected areas in communications*, 17(8), August 1999.

[15] M. Hildebrand J. Diaz R. Sigle G. Cristache, K. David. Aspects for the integration of ad-hoc and cellular networks. *3rd Scandinavian Workshop on Wireless Ad-hoc Networks*, May 2003. http ://wireless.kth.se/adhoc03/Proceedings/MHsession4-2.pdf.

[16] C. Qiao and H. Wu. iCAR : an integrated cellular and ad-hoc relay system. Proc. Int'l Conf. Computer Communications and Networks, Oct. 2000, pp. 154-161.

[17] B. Xu N.Bayer, D. Sivchenko and S. Hischke. Integration of heterogeneous ad hoc networks with the internet. *International Workshop on Wireless Ad-hoc Networks*, 2005. http ://www.ctr.kcl.ac.uk/iwwan2005/papers/26.pdf.

[18] P. Gunningberg E. Nordström and C. Tschudin. Comparison of gateway forwarding strategies in ad hoc networks. *Technical Report 2004-007*. http ://www.it.uu.se/research/publications/reports/2004-007/2004-007-nc.pdf.

[19] P. M. Ruiz and A. F. Gomez-Skarmeta. Adaptive gateway discovery mechanisms to enhance internet connectivity for mobile ad hoc networks. *Ad Hoc and Sensor Wireless Networks*, 1 :159–177, 2005.

[20] U. Körner A. Hamidian and A. Nilsson. Performance of internet access solutions in mobile ad hoc networks. *Dagstuhl-Workshop "Mobility and Wireless in Euro-NGI"*, pages 189–201, 2005.

[21] A. Hamidian. A study of internet connectivity for mobile ad hoc networks in ns 2. Master's thesis, Department of Communication Systems, Lund Institute of Technology, Lund University, January 2003.

[22] S.-H. G. Chan J. Chen, S. Li and J. He. Wiani : Wireless infrastructure and ad-hoc network intergration.

[23] J He J. Chen, S.-H. G. Chan and S.-C. Liew. Mixed-mode wlan : The integration of ad hoc mode with wireless lan infrastructure. *IEEE Globecom 2003*, 2003. http ://www.ie.cuhk.edu.hk/index.php ?id=88.

[24] D. B. Johnson Y.-C. Hu J. Broch, D. A. Maltz and J. Jetcheva. A performance comparison of multi-hop wireless ad hoc network routing protocols. In *Mobile Computing and Networking*, pages 85–97, 1998.

[25] C. E. Perkins S. R. Das and Elizabeth E. Royer. Performance comparison of two on-demand routing protocols for ad hoc networks. In *INFOCOM (1)*, pages 3–12, 2000.

[26] D. O. Jörg. Performance comparison of manet routing protocols in different network sizes. Computer science project, University of Berne, Switzerland, 2003.

[27] Douglas S. J. De Couto, D. Aguayo, B. A. Chambers, and R. Morris. Performance of multihop wireless networks : Shortest path is not enough. In *Proceedings of the First Workshop on Hot Topics in Networks (HotNets-I)*, Princeton, New Jersey, October 2002. ACM SIGCOMM.

[28] J. Dricot, P. De Doncker, E. Zimanyi, and F. Grenez. Impact of the physical layer on the performance of indoor wireless networks. In Proc. of the Int. Conf. on Software, Telecommunications and Computer Networks, 2003.

[29] M. Takai, J. Martin, and R. Bagrodia. Effects of wireless physical layer modeling in mobile ad hoc networks, 2001.

[30] V. Paxon and S. Floyd. Why we don't know how to simulate the internet. In *Winter Simulation Conference*, pages 1037–1044, 1997. http ://citeseer.ist.psu.edu/article/paxson97why.html.

[31] S. Floyd and V. Paxson. Difficulties in simulating the internet. *IEEE/ACM Transactions on Networking*, 9(4) :392–403, August 2001. http ://www.icir.org/floyd/papers.html.

[32] D. B. Johnson and D. A. Maltz. Dynamic source routing in ad hoc wireless networks. In Imielinski and Korth, editors, *Mobile Computing*, volume 353. Kluwer Academic Publishers, 1996.

[33] A. Al Hanbali, E. Altman, and P. Nain. A survey of tcp over mobile ad hoc networks. *IEEE Communications Surveys and Tutorials*, 7(3) :22 36, August 2005.

[34] D. Kim, C.-K. Toh, and Y. Choi. TCP-bus : Improving TCP performance in wireless ad-hoc networks. In *ICC (3)*, pages 1707–1713, 2000.

[35] Ashish Natani, Jagannadha Jakilnki, Mansoor Mohsin, and Vijay Sharma. Tcp for wireless networks. http ://citeseer.ist.psu.edu/natani01tcp.html.

[36] H. Balakrishnan, V. Padmanabhan, S. Seshan, R. H. Katz, and M. Stemm. TCP behavior of a busy internet server : Analysis and improvements. Technical Report CSD-97-966, 9, 1997.

[37] T. Clausen, P. Jacquet, and L. Viennot. Comparative study of CBR and TCP performance of MANET routing protocols. In *Workshop MESA*, 2002. ETSI.

[38] K. Fall and K. Varadhan. The ns manual (formerly ns notes and documentation), 2002. http ://www.isi.edu/nsnam/ns/doc/index.html.

[39] D. M. Nicol J. Liu, Y. Yuan and R. S. Gray. Empirical validation of wireless models in simulations of ad hoc routing protocols. In *SIMULATION*, volume 81. April 2005.