THÈSE

présentée à
l'Université Louis Pasteur de Strasbourg
Département Informatique
Laboratoire LSIIT, UMR CNRS-ULP N°7005

Pour obtenir le grade de
Docteur de l'Université Louis Pasteur
Mention SCIENCES
Spécialité INFORMATIQUE

par

Nicolas MONTAVONT

Gestion optimisée d'interfaces multiples et prise en compte des déplacements rapides sur un terminal IPv6 mobile

Soutenue publiquement le 16 Septembre 2004 devant le jury composé de :

M. **Andrzej DUDA**, Rapporteur externe
Professeur à l'ENSIMAG de Grenoble
M. **Serge FDIDA**, Examinateur
Professeur au laboratoire LIP6 de Paris
Mme. **Catherine Mongenet**, Rapporteur interne
Professeur à l'Université Louis Pasteur - Strasbourg
M. **Thomas NOEL**, Co-encadrant de thèse
Maître de conférence HDR à l'Université Louis Pasteur - Strasbourg
M. **Jean-Jacques PANSIOT**, Directeur de thèse
Professeur à l'Université Louis Pasteur - Strasbourg
M. **Stéphane UBEDA**, Rapporteur externe
Professeur à l'INSA de Lyon

Remerciements

L'achèvement de cette thèse a été possible grâce au soutien de nombreuses personnes, auxquelles je tiens à exprimer ma gratitude. Sans eux, ce travail n'aurait pas pu aboutir.

Le travail présenté dans ce manuscrit a pu être réalisé grâce au dévouement de plusieurs personnes qui ont fortement contribué aux résultats présentés. Je pense à Alexandre Boeglin, Nicolas Dichtel, Arnaud Frey, Romain Kuntz, Yaël Malka et Jean-Marc Muller qui ont réalisé un travail de fond remarquable.

Je remercie vivement Thomas Noël qui a tout fait pour mettre en place un environnement et des conditions de travail dont tout doctorant souhaiterait bénéficier. Je le remercie également pour les réflexes qu'il a su me donner et la confiance qu'il m'a témoigné.

J'ai également eu la chance d'être encadré par Jean-Jacques Pansiot. Son savoir et son recul se sont toujours révélés précieux.

Je remercie Andrzej Duda, Serge Fdida, Catherine Mongenet et Stéphane Ubéda pour avoir accepté de faire partie de mon jury de thèse.

Je remercie vivement Christophe Jelger qui m'a accompagné tout au long de ces trois dernières années, et qui a consacré beaucoup de son temps à mes travaux. Merci aussi à Jean Lorchat sans qui je n'aurais pas pu aller aussi loin dans mes travaux et qui, malgré toutes ses activités, a toujours été disponible. De manière générale, je remercie l'ensemble de l'équipe Recherche Réseau du LSIIT.

Je remercie Thierry Ernst, qui m'a chaleureusement accueilli à plusieurs reprises au Japon. Il m'a ouvert de nouvelles perspectives et montré des méthodes de travail différentes.

Enfin, je remercie Olivier et Catherine, les américains au grand cœur. Je remercie Sébastien dont l'amitié m'honore. Enfin, je remercie ma famille, dont le soutien m'a toujours poussé en avant.

A François.

Table des matières

Table des figures

Liste des tableaux

Introduction

Les travaux présentés dans cette thèse ont pour objectif d'optimiser les mécanismes de gestion de la mobilité. La mobilité considérée ici est non seulement la mobilité des équipements dans l'Internet Nouvelle Génération, mais également la mobilité des flux au sein d'un même terminal. Plus particulièrement, cette thèse a donné lieu à l'implémentation d'un nouvel outil pour la simulation des réseaux sans fil, et à la conception et au développement d'une nouvelle architecture pour le terminal afin d'optimiser l'utilisation d'interfaces multiples sur un nœud mobile IPv6.

Contexte

La miniaturisation des équipements communiquants et le fort développement des technologies sans fil telles que le Wifi ou le GPRS, permettent aujourd'hui à un utilisateur de se déplacer tout en restant connecté à l'Internet. Cette évolution laisse présager un nouveau modèle d'utilisation de l'Internet où une majorité des utilisateurs seront mobiles. Parallèlement, une nouvelle version du protocole pour l'Internet (IP), dit Internet Nouvelle Génération, est en cours de finalisation. Cette nouvelle version du protocole pour l'Internet propose un cadre générique de mobilité qui s'applique à la majorité des usages. Toutefois, cette généricité a un coût non négligeable sur les performances lors des changements de localisation. Par ailleurs, le terme mobilité dans l'Internet représente aussi bien la mobilité de l'équipement, lorsqu'il se déplace physiquement entre différents réseaux IPv6, que la mobilité entre différentes interfaces réseau, au sein du même terminal.

Le premier type de mobilité est la *mobilité horizontale*. Généralement, elle est occasionnée lorsqu'un terminal mobile équipé d'une interface réseau sans fil se déplace et change son point d'attache à l'Internet. Un tel changement de réseau IPv6 nécessite une gestion particulière, afin de ne pas interrompre les communications

en cours. Mobile IPv6 [84] est le protocole standard utilisé pour gérer ce type de mobilité. Mobile IPv6 propose d'associer à un noeud mobile deux adresses : une adresse mère et une adresse temporaire. L'adresse mère est une adresse permanente allouée au terminal mobile qui lui permet d'être joint à tout moment, quelle que soit sa localisation dans l'Internet. Quand le nœud mobile est en déplacement, son adresse mère est associée à une adresse temporaire obtenue dans le réseau visité.

Le second type de mobilité, dit *mobilité verticale*, est une mobilité interne à l'équipement. Elle consiste en des redirections de communications entre interfaces réseau du terminal. Par exemple, si l'utilisateur se rend compte qu'un de ses flux devient volumineux en terme de bande passante pour l'interface qu'il est en train d'utiliser, il peut demander à ce que ce flux soit redirigé sur une autre interface à plus forte capacité. Le travail réalisé au cours de cette thèse fut donc d'analyser les enjeux d'un terminal à interfaces multiples [61, 115] et de proposer une nouvelle architecture modulaire pour le terminal [120, 20]. Cette architecture permet d'intégrer et d'utiliser des interfaces multiples et d'optimiser les deux types de mobilité cités ci-dessus.

Déclencheurs de niveau 2

D'après les analyses que nous avons pu réaliser sur Mobile IPv6 [119, 121, 116], le temps de *handover* dans Mobile IPv6 reste trop important pour des communications temps réel telles que la téléphonie sur IP ou la vidéo/musique à la demande. Des optimisations sont possibles à aux différentes étapes du protocole, aussi bien au niveau de la détection de déplacement, qu'au niveau de l'enregistrement de l'association entre les adresses mère et temporaire(s) du terminal. Pour cette dernière optimisation, de nombreuses études ont déjà eu lieu ; nous pouvons citer Mobile IPv6 Hiérarchique [168] qui utilise un point d'ancrage qui se charge de masquer les mouvements à l'intérieur d'un domaine, ou Fast Mobile IPv6 [201] qui permet un échange entre l'ancienne et la nouvelle localisation du terminal mobile.

Une des propositions avancées au cours de cette thèse a été d'améliorer les opérations de mobilité par la prise en compte des états et événements de la couche 2 du modèle OSI [144]. Cette proposition provient de la constatation qu'il n'existe à ce jour que très peu d'interactions entre la couche 2 (couche MAC) et la couche 3 (couche IP). Or, avec le développement des technologies sans fil qui ont des caractéristiques bien plus fluctuantes que leurs homologues filaires, il s'avère fort intéressant de mettre en place une interaction entre ces couches afin que la couche 3 soit informée au plus vite des changements observés au niveau 2.

Afin de mettre en place un tel échange, nous avons participé à la définition d'une nouvelle interface d'échanges entre les deux couches MAC et IP. Cette définition est actuellement proposée pour standardisation à l'IETF[1] [35, 195, 124]. Une notion importante de cette définition est que l'abstraction des différents paramètres et événements doit être totalement indépendante de toute technologie. Cette généralisation permet notamment l'intégration simplifiée d'une future technologie dans les différents mécanismes utilisant cette abstraction.

L'ensemble des paramètres défini servira à l'optimisation des handovers horizontaux, à la sélection d'interfaces et à l'optimisation de handovers verticaux, comme nous allons le voir dans les parties suivantes.

Anticipation des handovers horizontaux

Lorsqu'un nœud mobile change de réseau IPv6, un mécanisme tel que Mobile IPv6 est nécessaire pour le maintien des communications en cours. Afin d'accélérer le déclenchement des opérations de Mobile IPv6, l'événement indiquant que le terminal vient de changer de point d'attachement peut être utilisé. La prise en compte d'un tel événement permet de déclencher une procédure de handover sans délai supplémentaire. Cependant, bien qu'inférieur à Mobile IPv6, le temps d'interruption généré reste trop élevé pour permettre la correcte réception d'applications temps réel.

Une nouvelle méthode mise au point au cours de ces trois dernières années est l'anticipation de mouvement basée sur des informations de niveau 2 [118, 117]. Grâce à la définition des paramètres génériques de niveau 2, il nous est possible de connaître à tout instant la qualité du lien entre le terminal mobile et son point d'accès sans fil. En utilisant cette information, le terminal peut anticiper un déplacement imminent, et rechercher son futur point d'attachement. Cette anticipation permettra au terminal de préparer sa configuration et les noeuds du réseau pour son déplacement, avant même d'entamer son mouvement effectif.

Afin de mesurer les performances de ce nouveau mécanisme et pour pouvoir le comparer à d'autres méthodes, quatre optimisations de détection de mouvement ont été implémentées dans le simulateur *SimulX*. Ce simulateur a été développé pendant ma thèse et a pour but de simuler des protocoles de mobilité IPv6.

[1] Internet Engineeting Task Force - Organisme de standardisation des protocoles pour l'Internet

Nouvelle architecture de gestion d'interfaces multiples

Comme nous le mentionnions précédemment, de nombreux équipements mobiles sont actuellement vendus avec plusieurs interfaces réseau. Ces interfaces réseau implémentent des technologies de communication différentes, aussi bien filaires que sans fil. Ces technologies ont des capacités très variables, aussi bien au niveau du débit de données, de la fiabilité que de la portée. De plus, le coût d'utilisation de ces interfaces peut être très différent pour l'utilisateur. Or, dans les systèmes actuels, un utilisateur maîtrise mal la sélection d'interfaces réseau. En effet, il est peu courant de pouvoir utiliser plusieurs interfaces simultanément pour différents flux de données, ou de pouvoir choisir quelle interface utiliser pour un certain type de flux.

Dans une volonté d'intégrer de manière optimale l'ensemble de ces technologies, nous avons proposé une nouvelle architecture pour le terminal [20, 128], appelée ***M**ultiple **I**nterfaces **M**anagement **A**rchitecture* (MIMA). Cette architecture s'intègre dans le modèle classique OSI, par l'ajout de 5 nouveaux modules intermédiaires aux couches standards TCP/IP, aussi bien au niveau utilisateur que dans le noyau du système. La conception de l'architecture en différents modules favorise son intégration dans le modèle existant et permet une évolution simplifiée au cours du temps ; cette modularité permettra également une mise à jour de n'importe quelle partie de l'architecture en fonction de la spécification de nouveaux protocoles, ou d'intégrer de nouvelles technologies. L'objectif premier de cette architecture est de permettre à l'utilisateur ou à une entité administrative de configurer son terminal pour pouvoir utiliser ses interfaces simultanément, de placer des priorités sur les interfaces / flux, et de rediriger de manière efficace les flux entre les différentes interfaces.

Les différents modules composant cette nouvelle architecture sont les suivants : 1. Un *module d'Extraction*, implémenté entre la couche 2 et la couche 3 du système qui représente l'abstraction des déclencheurs de niveau 2, permettant à la couche 3 de connaître les états des différentes interfaces sous-jacentes. 2. Le module *Gestionnaire d'Interfaces*, qui se situe dans la couche 3 et qui interagit fortement avec Mobile IPv6 pour répartir les flux sur les interfaces et effectuer les redirections. Ce module est le plus important de l'architecture puisqu'il prend les décisions finales d'attribution et de redirection selon les préférences des utilisateurs et les états des interfaces. Plusieurs mécanismes de redirection y sont implémentés, selon la configuration du réseau auquel est rattaché le terminal mobile [127, 126, 125]. 3. Le *Démon de Communication*, qui gère la communication entre l'espace noyau et

l'espace utilisateur. 4. Le *module Gestionnaire de Profils*, qui permet à l'utilisateur d'entrer différentes préférences sur les interfaces par rapport aux flux et aux correspondants, ou d'enregistrer des paramètres réseau pour la configuration d'une interface donnée. 5. Le *module d'Adaptation des Applications*, qui permet d'adapter des applications aux conditions courantes du réseau : si les capacités réseau diminuent suite à une chute de débit de données par exemple, un codec vidéo différent sera alors utilisé par l'application de vidéo à la demande pour diminuer la bande passante nécessaire.

Dans le cadre du projet RNRT[2] Cyberté, l'ensemble de cette architecture a été implémenté dans un système d'exploitation Linux. Plusieurs déclencheurs de redirection ont été implémentés : saturation d'une interface, perte d'une interface, gain d'une interface, redirection temporaire durant un handover horizontal, chute du débit de données observée sur une interface, anticipation de la perte de connectivité sur une interface. Plusieurs tests ont été effectués et les temps de redirection varient entre 40 millisecondes et 2 secondes, selon le type du déclencheur, la technologie utilisée, et le fait que le terminal mobile soit rattaché au même réseau avec plusieurs interfaces ou non. Il peut être noté que dans certaines situations où la redirection se fait avant la coupure réelle de l'interface, le nombre de paquets perdus est nul, puisque durant la redirection, les paquets continuent à arriver sur l'ancienne interface.

Plan de la thèse

La suite de ce rapport est organisée comme suit. Le prochain chapitre est consacré à la présentation du contexte de travail, à savoir la mobilité dans l'Internet IPv6. Nous détaillerons comment la mobilité horizontale peut être gérée par les différents niveaux du modèle TCP/IP. Puis, nous analyserons les différentes propositions actuelles sur la gestion de la mobilité verticale. Chacune d'entre elles propose une méthode différente, basée sur une architecture pour le réseau et/ou l'introduction de nouveaux mécanismes sur les nœuds mobiles. Efin, nous étudierons les optimisations possibles à cette gestion de la mobilité, comme les modèles hiérarchiques ou l'accélération de la détection de mouvements.

Ensuite, nous présenterons trois normes de communication sans fil des plus répandues au jour d'aujourd'hui. L'étude des normes IEEE 802.11 [2], Bluetooth [36] et des réseaux cellulaires (GSM, GPRS, UMTS) [1] nous permettra d'identifier les avantages et inconvénients de chacune d'entre elles, et surtout, pourquoi il est im-

[2]Réseau National de la Recherche en Télécommunications

pératif de proposer un mécanisme d'intégration de toutes ces technologies au sein d'un même terminal.

Le chapitre suivant est consacré à l'évaluation des protocoles de gestion de la mobilité, principalement horizontale. Cette évaluation traite aussi bien des performances théoriques que des performances pratiques des protocoles Mobile IPv6, Mobile IPv6 Hiérarchique et Fast Mobile IPv6. Ce chapitre nous permettra notamment d'identifier les points sensibles qui occasionnent des dégradations de service lors des mouvements des nœuds mobiles.

Ensuite, un nouvel outil pour la simulation sera présenté. Il ne s'agit pas ici de détailler toute l'implémentation de ce nouveau simulateur nommé *SimulX*, mais plutôt d'expliquer les raisons de son développement, et les principales fonctionnalités qu'il apporte pour l'évaluation des protocoles de gestion de mobilité dans l'Internet. Nous profiterons de cette présentation pour compléter l'évaluation des performances des réseaux IEEE 802.11b.

Enfin, une première proposition d'optimisation pour les handovers dans les réseaux IEEE 802.11b sera exposée. Il s'agit de la mise en place d'anticipation de mouvements, pour accélérer non seulement le handover de niveau 2, mais également le handover de niveau 3, par la mise en place de la duplication du trafic aux différentes localisations potentielles du nœud mobile. Cette solution de *Handover Anticipé* sera évaluée dans SimulX, et comparer à d'autres méthodes de gestion de la mobilité.

Finalement, nous présenterons le cœur du travail de cette thèse à savoir la conception et l'évaluation d'une nouvelle architecture pour le terminal pour l'intégration d'interfaces multiples. Ce chapitre présentera les intérêts et objectifs d'une nouvelle architecture, avant de présenter chaque composant de celle-ci. La dernière partie du chapitre sera consacrée à l'évaluation des performances de redirection de flux entre interfaces, afin de vérifier l'apport de cette nouvelle architecture et le gain mesuré sur le nœud mobile.

Un dernier chapitre est dédié aux conclusions de ces travaux et donnera les perspectives envisagées.

Chapitre 1

La gestion de la mobilité dans l'Internet IPv6

1.1 Introduction

La mobilité des équipements IP est un comportement relativement récent des nœuds de l'Internet. Les protocoles assurant le fonctionnement de l'Internet n'ont donc pas été définis, ni même pensés dans leur conception initiale pour intégrer un support de la mobilité. La mobilité des équipements est d'abord apparue avec la notion de nomadisme [25]. Le nomadisme représente le déplacement d'un nœud alors qu'il n'est pas connecté au réseau, c'est-à-dire entre ses communications. Supporter le nomadisme revient à permettre à l'utilisateur de ne pas avoir à reconfigurer son équipement lui-même après chacun de ses déplacements ; cette fonctionnalité passe par une gestion centrale d'adressage comme DHCP [58] ou par un protocole d'auto-configuration [176]. En outre, le développement des technologies sans fil a amené la possibilité supplémentaire pour les nœuds de se déplacer en cours de communication. La problématique de cette mobilité est la continuïté des communications courantes : comment le nœud mobile pourra poursuivre ses communications bien que rattaché dans un nouveau sous-réseau. De plus, les terminaux se sont récemment vus équipés de plusieurs interfaces réseau, leur permettant un accès à l'Internet par différentes technologies. L'hétérogénéité de ces technologies permet non seulement une plus grande facilité d'accès, mais également une utilisation simultanée de plusieurs interfaces sur un même nœud. Des mécanismes de redirection seront alors nécessaires pour effectuer la répartition dynamique des communications sur plusieurs interfaces.

Les travaux sur la gestion de la mobilité dans l'Internet Nouvelle Génération ont notamment débuté avec les spécifications du protocole Mobile IPv6 à l'IETF [84]. Il existe cependant de nombreux travaux antérieurs pour la gestion de la mobilité dans l'Internet actuel (IPv4). Le but n'est pas ici de faire un inventaire détaillé de ces protocoles, mais l'on peut tout de même affirmer que les bases de la gestion de la mobilité dans l'Internet reposent principalement sur deux propositions. L'une est devenue Mobile IP [149], l'autre était une proposition de SONY appelée VIP [174, 175]. Il est intéressant de noter qu'à l'heure actuelle on retrouve les bases de ces deux concepts dans l'Internet Nouvelle Génération : l'un s'appelle Mobile IPv6 [84] et l'autre LIN6 [91, 83, 173]. Cependant, comme nous allons le voir, d'autres solutions, à plusieurs niveaux du modèle TCP/IP [144, 171], ont été proposées et toutes ces propositions apportent une particularité intéressante.

La suite de ce chapitre est divisée en quatre parties principales. Nous verrons tout d'abord une introduction au modèle de l'Internet et les concepts de la mobilité des nœuds. Ensuite nous étudierons consécutivement la mobilité horizontale, avec un aperçu de différentes solutions de différents niveaux, et la mobilité verticale, avec un intérêt tout particulier pour les architectures qui ont été proposées. Enfin nous décrirons tout un ensemble de protocoles définis pour l'optimisation des mouvements des nœuds mobiles, par la réduction du temps de handover et la réduction du nombre de paquets perdus.

1.2 Terminologie et architecture

Avant de commencer toute description de protocole, il est nécessaire de rappeler les termes et concepts de base qui permettent à la communauté réseau de s'appuyer sur des définitions universelles. Plusieurs termes ont été définis et un modèle pour l'architecture d'un nœud de l'Internet est devenu une référence depuis les années 1970. Dans cette section, nous nous proposons de rapidement revoir ces fondements. L'ensemble des définitions des termes utilisés dans ce manuscrit est partiellement donné dans le glossaire et peut également être trouvé dans [104, 109].

1.2.1 Modèle en couches

L'Internet est la plus grande interconnexion de réseaux existant à l'heure actuelle, s'étendant dans le monde entier. Il rassemble plusieurs millions d'ordinateurs et d'utilisateurs de manière décentralisée ; toute paire de nœuds peut communiquer ensemble et chaque nœud de l'Internet peut être client ou serveur d'un

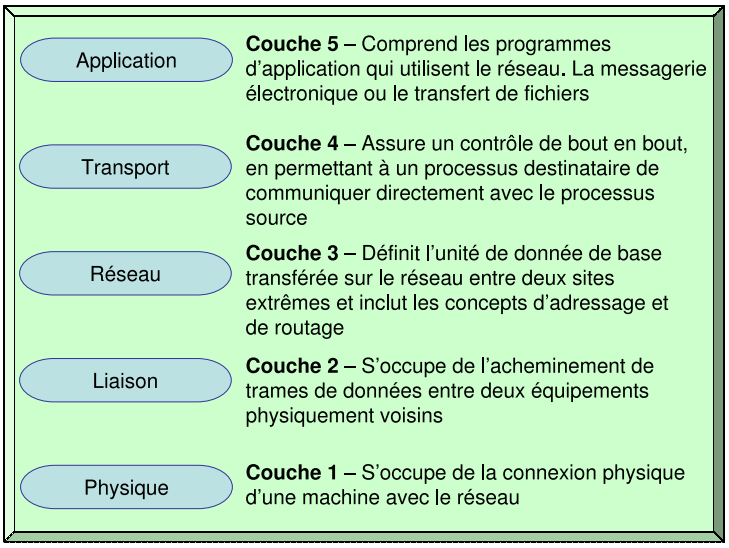


FIG. 1.1 – Modèle en couches TCP/IP

service. Pour que ces machines puissent coopérer, il a été nécessaire de définir un certain nombre de standards. L'architecture des machines de l'Internet repose sur une approche en couches, similaire au modèle OSI[1] (Interconnexion des Systèmes Ouverts) [144]. Ce modèle a été simplifié pour finalement devenir le modèle de référence TCP/IP [171], qui est décrit dans la figure 1.1. Cette séparation en couches permet une forte hétérogénéité d'implémentations et une évolution simplifiée des protocoles : la spécification du protocole utilisé dans une couche peut être complètement modifiée sans qu'il ne faille redéfinir les autres couches du modèle. Nous verrons dans la suite l'importance de cette caractéristique dans nos contributions à la gestion de la mobilité.

Chaque couche du modèle TCP/IP a un rôle particulier. La couche physique établit la connexion physique entre un système et le réseau et encode les données en fonction du support (câbles, ondes radio, etc). A ce niveau, l'unité d'information est le *bit*. La couche de liaison de données construit les trames à envoyer et analyse les trames reçues. Cette couche est également responsable de l'accès au médium, du contrôle et de la correction des données. Les protocoles Ethernet (IEEE 802.3) [11], Token Ring (IEEE802.5) [12] PPP [49], SLIP [156] sont des standards opérant sur cette couche. La couche réseau sert à acheminer les paquets entre n'importe quels nœuds de l'Internet grâce à un adressage hiérarchique des réseaux. Ici interviennent les protocoles IPX [141], IP [151, 54]. La couche transport détermine la séquence des données à transmettre et assure la confirmation de cette transmission. Les

[1] Open Systems Interconnection - Interconnection des Systèmes Ouverts

protocoles couramment utilisés sont UDP [150] et TCP [152]. Enfin, la couche application représente l'interface entre le réseau et le programme ayant émis une requête réseau.

Tout au long de ce manuscrit, nous nous intéresserons plus particulièrement aux couches liaison et réseau : la couche liaison, car elle gère l'accès au médium physique et certaines informations peuvent être très utiles dans la gestion de la mobilité ; la couche réseau, car la gestion de la mobilité à ce niveau de la pile TCP/IP nous semble la plus appropriée. L'objet de ce travail est d'évaluer les mécanismes actuels de gestion de la mobilité et d'en identifier les lacunes. Nous proposerons ensuite un ensemble de solutions visant à réduire les effets de la mobilité sur les nœuds et le réseau. Nous serons également appelés à parler des couches transport et application pour permettre une éventuelle adaptabilité aux conséquences de la mobilité et pour pouvoir tester les effets des protocoles de la couche réseau sur les couches supérieures. Enfin, nous verrons l'impact et l'intégration proposée de l'utilisation d'interfaces réseau multiples sur ce modèle.

1.2.2 La nouvelle version du protocole IP : IP version 6

L'Internet, qui interconnecte plusieurs millions d'ordinateurs et d'utilisateurs, est le plus grand réseau existant. Comme nous l'avons déjà dit, IP est le protocole en charge du routage dans l'Internet. La popularité exponentielle de l'Internet au cours des dix dernières années a révélé les limites de la version actuelle du protocole IP (IP version 4). Les adresses IPv4 sont en effet codées sur 32 bits ce qui permet théoriquement d'adresser 2^{32} machines, soit à peu près 4 milliards. En fait, l'attribution des adresses se fait par réseau, c'est-à-dire que seule une partie des 32 bits sert à identifier une machine, le reste étant attribué à une organisation. Ce découpage limite en pratique le nombre de machines adressables à environ 2 milliards [196] alors qu'on estime en 2003 à plus de 600 millions le nombre d'utilisateurs d'Internet [73]. De plus, l'attribution des adresses IPv4 dans le monde est très inégale, ce qui limite sévèrement le développement de l'Internet dans certains pays. Enfin, les tables de routage des routeurs du maillage central de l'Internet atteignent des tailles très importantes et deviennent de plus en plus difficiles à maintenir. Les adresses ont en effet été attribuées de manière chaotique depuis les années 80, ce qui empêche de réduire la taille de ces tables de manière suffisante en utilisant des techniques d'agrégation.

Le protocole IPv6 propose l'utilisation d'adresses de 128 bits qui permettent d'adresser un nombre important de machines [47]. Avec le plan d'adressage agrégé retenu [78], les 64 premiers bits d'une adresse IPv6 sont utilisés pour indiquer le

préfixe du réseau, alors que les 64 bits restant identifient la machine (ou plutôt l'interface physique) à laquelle cette adresse est attribuée. De plus, un système d'options pouvant être rajoutées à l'en-tête IPv6 des paquets de données permet de mettre en œuvre des extensions, par exemple pour gérer la mobilité des nœuds du réseau. Enfin, une grande force du protocole IPv6 est le développement d'une méthode d'auto-configuration d'adresse sans état, c'est à dire ne nécessitant pas de serveur centralisé.

IPv6 ayant atteint un degré de maturité suffisant, il devient LE protocole de l'Internet Nouvelle Génération. L'apparition de nouveaux usages de l'Internet tels que la mobilité tendent à accélérer le déploiement du protocole IPv6. La multiplication des réseaux d'accès sans fil induit en effet une forte croissance de la population d'utilisateurs mobiles. De plus, l'augmentation très rapide des débits des réseaux rend possible le développement d'applications multimédia telles que la télévision et la radio par Internet. Ces applications sont particulièrement adaptées à l'utilisation de communications multicast qui permettent de mettre en communication un grand nombre d'utilisateurs.

1.2.3 Les nouveaux acteurs de l'Internet sans fil et de la mobilité

L'arrivée des technologies sans fil dans l'Internet a amené un nouvel acteur dans les réseaux : *le point d'accès*. Par définition, il s'agit d'un équipement de niveau 2, c'est-à-dire non muni de couche réseau, établissant un pont entre le réseau filaire et le réseau sans fil. Un point d'accès est donc équipé d'au moins deux interfaces, une filaire et une sans fil, et a pour rôle de relayer les paquets d'une interface sur l'autre (ou éventuellement sur la même interface lorsque les deux stations qui communiquent sont rattachées au même point d'accès). Cependant, la plupart des produits commerciaux sont munis de fonctionnalités de niveau IP, afin d'être adressables, et de permettre le filtrage de paquets. Certaines propositions de gestion de la mobilité s'appuient même sur des fonctionnalités IP au niveau des points d'accès [38].

Par ailleurs, le déploiement des technologies sans fil favorise la mobilité des équipements IP communiquants. Le fait de ne pas avoir de fil permet des déplacements tout en étant en communication avec un pair. Cette nouvelle caractéristique a nécessité le développement de protocoles de gestion de la mobilité. Sans parler d'une proposition spécifique, cette gestion de la mobilité a amené la définition des termes suivants, qui sont représentés dans la figure 1.2 :

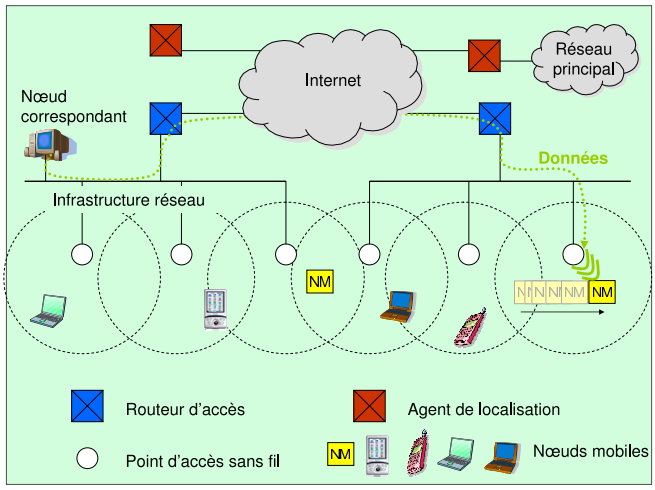


FIG. 1.2 – Les acteurs de l'Internet sans fil et de la mobilité

- **Nœud mobile** : équipement IP qui est capable de se déplacer sur Internet, indépendamment du fait qu'il soit en cours de communication. On peut penser ici à un ordinateur portable ou un assistant personnel muni d'une interface sans fil. Plusieurs types de nœuds mobiles sont représentés dans la figure 1.2.
- **Réseau principal ou réseau mère** : réseau d'attache primaire du nœud mobile, où les paquets à destination du nœud mobile sont envoyés par défaut, indépendamment de la position courante du nœud mobile. Ce réseau pourra éventuellement être virtuel. Dans certaines propositions de gestion de la mobilité, il peut même y avoir plusieurs réseaux ayant la fonctionnalité de réseau mère [187].
- **Réseau visité** : réseau courant auquel est rattaché le nœud mobile et par lequel il accède à Internet.
- **Nœud correspondant** : émetteur et / ou récepteur d'un flux échangé avec un nœud mobile à un moment donné. Il est à noter que le nœud correspondant peut également être mobile.
- **Agent de relais / de localisation** : toute solution de mobilité utilise à un moment donné un agent de relais, c'est-à-dire un agent qui est en charge d'intercepter les paquets à destination d'un nœud mobile. Son rôle est soit de rediriger les paquets du nœud mobile vers sa localisation courante (voir l'agent mère de Mobile IP [199, 84]), soit d'informer les nœuds demandeurs de la localisation courante du nœud mobile [162].

1.2.4 Adressage et multihoming

Le modèle de l'Internet est basé sur un adressage hiérarchique, qui permet une construction de route, un routage et une allocation d'adresses sur les équipements efficaces. Les adresses IP sont l'agrégation de l'identification du réseau d'attache et de l'identification du nœud, comme nous l'avons vu dans la section 1.2.2.

Lorsqu'un nœud est mobile, il peut avoir plusieurs adresses simultanément. L'adresse attribuée à un nœud mobile de manière permanente est dite *adresse principale* ou *adresse mère*. C'est l'adresse qui pourra toujours être utilisée par un nœud correspondant pour joindre le nœud mobile, quelle que soit sa localisation dans l'Internet. En parallèle, le nœud mobile peut détenir une (plusieurs) adresse(s) temporaire(s), en fonction de son point d'attachement dans l'Internet. Cette (ces) adresse(s) est (sont) obtenue(s) par le nœud mobile à chaque entrée dans un réseau visité. Toute la problématique de la gestion de la mobilité consiste à établir l'association entre les différentes adresses, principale(s) et temporaire(s), qu'un nœud mobile possède à un moment donné.

De manière générale, nous parlerons de **multihoming** [61, 115] pour des nœuds ayant plusieurs adresses simultanément. Ce sera le cas lorsque plusieurs préfixes seront annoncés sur le(s) lien(s) au(x)quel(s) le nœud mobile est rattaché, ou alors lorsque le nœud mobile a plusieurs interfaces réseau.

L'adresse utilisée par les applications pour établir une communication avec un nœud mobile sera l'adresse qui identifie cette communication pendant toute sa durée. En effet, les couches transport [72] utilisent notamment les adresses source et destination pour identifier une communication. Cette caractéristique met en avant le double rôle des adresses IP, à savoir un rôle de localisation et un rôle d'identification. Nous verrons dans la suite l'impact de cette double fonctionnalité des adresses IP sur la gestion de la mobilité.

1.2.5 Concepts liés à la mobilité

Comme nous venons de le voir, la mobilité dans l'Internet nécessite un support particulier. La plupart du temps, le temps pris par le protocole de gestion de la mobilité pour mettre à jour la position de l'équipement sera critique, dans le sens où les communications du nœud mobile seront interrompues. Plus particulièrement, nous utiliserons la terminologie suivante (voir [104]) :

– **Handover de niveau 2** : Processus nécessaire pour le changement de point

d'accès.
- **Handover de niveau 3** : Processus nécessaire pour le changement de sous-réseau IPv6.
- **Temps de latence du handover** : Laps de temps compris entre le moment où le nœud mobile pouvait utiliser son interface, jusqu'au moment où la mise à jour de localisation de l'équipement a été effectuée. Par abus de langage, nous utiliserons également le terme *temps de handover*.
- **Temps d'interruption des communications** : Laps de temps pendant lequel le nœud mobile ne peut ni envoyer, ni recevoir des paquets de données.
- **Nombre de paquets perdus** : Nombre de paquets de données perdus pendant que le nœud mobile effectuait un handover entre sous-réseaux.

Dans la suite de ce chapitre, nous allons nous intéresser aux solutions les plus populaires de gestion de la mobilité. Après avoir discuté de la mobilité horizontale et verticale dans les deux sections suivantes, nous présenterons quelques optimisations pour les mécanismes de gestion de la mobilité.

1.3 La mobilité horizontale

La mobilité horizontale représente les déplacements des équipements mobiles entre différents réseaux IPv6. L'exemple type est un nœud mobile ayant une interface réseau sans fil qui se déplace dans un établissement fournissant l'accès à un ensemble de sous-réseaux sans fil. Le déplacement du nœud mobile peut le conduire à un changement de point d'accès, connecté à un sous-réseau différent. Ce changement de point d'attachement occasionnera un changement d'adresse IPv6 sur le nœud mobile. Ce changement de localisation nécessite une gestion particulière pour que le nœud mobile reste joignable par tout nœud de l'Internet et que ses communications n'aient pas à être ré-initialisées.

Lorsqu'on parle de mobilité, on pense principalement à des réseaux IPv6 sans fil, car ils apportent le support de la mobilité aux utilisateurs. Toutefois, tout protocole gérant la mobilité se veut indépendant des couches physiques afin d'offrir un protocole générique, qui fonctionnera quel que soit le support de communication sous-jacent : réseaux filaires, réseaux locaux sans fil, réseaux cellulaires... Cette indépendance a un certain coût comme nous le verrons par la suite.

Les principaux objectifs des protocoles de gestion de la mobilité sont le maintien des communications suite à un changement de localisation sur l'Internet, mais aussi la réduction du temps d'interruption des communications et de manière plus générale leur efficacité (en terme de nombre de messages et de routage des pa-

quets de données dans l'Internet). La mobilité affecte toutes les couches du modèle TCP/IP. La couche physique est affectée par la position du nœud mobile par rapport au point d'accès (contrôle du ratio signal / bruit). La couche liaison doit faire face à différentes méthodes de correction d'erreurs, gérer les terminaux cachés et la mobilité entre points d'accès. La couche réseau est impliquée dans la localisation et le routage pour atteindre l'équipement en mouvement. La couche transport est concernée par le contrôle de congestion et d'erreurs. La couche applicative doit pouvoir être configurée correctement et une découverte des services doit être possible quelle que soit la localisation des équipements. Comme chaque couche est concernée par la mobilité, nous allons voir différentes solutions de différents niveaux se proposant de gérer la mobilité.

1.3.1 Mobile IPv6

La gestion de la mobilité dans l'Internet Nouvelle Génération repose principalement sur les spécifications du protocole Mobile IPv6 [147], qui viennent d'être standardisées à l'IETF [84]. Le protocole Mobile IP provient d'une première proposition pour l'Internet IPv4 [149, 64] où tous les routeurs situés entre le dernier réseau visité et le nœud mobile devaient maintenir un cache de localisation. Les spécifications de Mobile IPv6 ont nécessité plusieurs années de recherche, principalement pour des raisons liées à la sécurité des messages de contrôle entre le mobile et les autres acteurs de la mobilité. A présent, bien que son concept soit simple, le protocole se révèle relativement complexe et repose sur une validation de plusieurs années (la première version du protocole date de juillet 1995). Mobile IPv6 profite de l'héritage du protocole défini pour l'Internet actuel : Mobile IPv4 [199]. Bien qu'historiquement Mobile IPv6 ait été développé pour gérer la mobilité horizontale, nous verrons dans la suite qu'il peut également être utilisé pour la mobilité verticale.

Détail du protocole

La solution Mobile IPv6 s'appuie sur le fait que la mobilité des équipements est un problème de routage. Un routeur spécifique, appelé l'agent mère, situé dans le réseau mère du nœud mobile, est nécessaire au fonctionnement de Mobile IPv6. Cet agent de relais est chargé d'intercepter les paquets à destination du nœud mobile et de les lui retransmettre à sa position courante dans l'Internet. Pour réaliser ces opérations, l'agent mère maintient un cache d'association entre l'adresse mère et l'adresse temporaire du nœud mobile. Les messages Binding Update et Binding

Acknowledgement sont utilisés à cet effet. La sécurisation de ces mises à jour est établie par l'utilisation de IPsec [22].

En ce qui concerne l'échange de paquets de données entre un nœud mobile et un nœud correspondant, Mobile IPv6 définit trois mécanismes distincts. Le premier d'entre eux est celui qui avait été défini dans Mobile IPv4, qui est l'utilisation d'un tunnel IP dans IP [146] ; dans cette méthode, un tunnel bi-directionnel entre le nœud mobile et son agent mère est utilisé pour transporter tous les paquets à destination et en provenance du nœud correspondant. La mobilité est alors complètement transparente au nœud correspondant qui communique avec le nœud mobile exactement comme s'il était rattaché dans son réseau mère. Le routage triangulaire est une alternative au tunnel bi-directionnel. Dans la méthode du routage triangulaire, le nœud correspondant envoie toujours ses paquets de données dans le réseau mère du nœud mobile où ils sont interceptés et redirigés par l'agent mère. Par contre, les paquets en provenance du nœud mobile sont directement routés depuis la localisation courante du nœud mobile vers le nœud correspondant, i.e. sans passer par le réseau mère. Pour cela, un en-tête de routage contenant l'adresse mère doit être utilisé. Sur réception d'un tel paquet, le nœud correspondant doit considérer l'adresse présente dans l'en-tête de routage comme la source du paquet.

Le dernier mécanisme de routage de paquets entre un nœud mobile et son correspondant est l'utilisation d'optimisation de routage. Cette fonctionnalité a été pensée pour Mobile IPv4 [148] et intégrée dans la spécification de base de Mobile IPv6. Pour cela, le nœud mobile doit entamer une procédure de *Return Routability* avec son correspondant. Cette procédure est établie par quatre messages, dont deux doivent passer par l'agent mère du mobile. L'échange de ces messages a pour but d'éviter qu'un nœud malicieux de l'Internet situé entre le nœud mobile et son agent mère ne modifie le cache d'association d'un correspondant, en se faisant passer pour le nœud mobile. Une fois que la procédure de *Return Routability* a terminé avec succès, le nœud mobile peut envoyer une mise à jour d'association au correspondant. Suite à ce message, les paquets de données peuvent directement être envoyés à la position courante du nœud mobile. Afin d'optimiser le temps de handover, [185] propose que le Binding Update puisse être envoyé en parallèle des messages de la procédure de *Return Routability*. Cela semble nécessaire pour réduire le temps de latence des handovers, puisqu'une mise à jour de la localisation d'un nœud mobile, suite à un déplacement, est retardée de deux aller-retours entre le nœud mobile et son correspondant. D'autre part des travaux sont actuellement en cours pour déterminer quand l'optimisation de routage avec un nœud correspondant doit être mise en place [164].

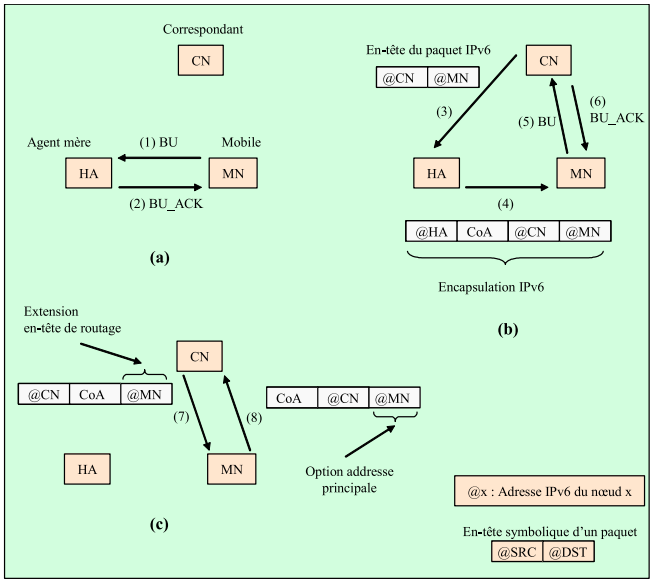


FIG. 1.3 – Mécanismes de gestion de la mobilité avec Mobile IPv6

Détection de liens et configuration d'adresses

La gestion de la mobilité consiste bien évidemment en la mise à jour de la localisation du nœud mobile et en la distribution des paquets qui lui sont destinés, mais aussi en la manière dont le nœud mobile se rend compte qu'il a effectué un mouvement. Cette opération n'est pas triviale : comment découvrir le nouveau lien ? Comment être sûr que l'ancien lien n'est plus disponible ? Bien entendu, le délai de découverte du changement de lien est un facteur important du temps total de handover.

Afin de garantir l'indépendance du protocole par rapport à toute technologie sous-jacente, la détection des mouvements repose principalement sur la réception des *Router Advertisement* (que nous noterons par la suite RA) envoyés par les routeurs situés sur le nouveau réseau visité par le nœud mobile [133]. Afin de minimiser le temps de découverte, l'IETF propose un intervalle d'émission des RA situé entre 0,03 sec et 0,07 sec. Cet intervalle est toutefois relativement contraignant et pénalise fortement les réseaux sans fil à faible débit, car une partie de leur bande passante utile sera sollicitée pour améliorer les détections de mouvements des équipements mobiles. Il pose également problème pour les réseaux de type cellulaire où généralement l'utilisateur est facturé selon le nombre de données reçues (voir la section 1.5.1).

Le protocole de découverte des voisins [133] offert par IPv6 joue un rôle important dans Mobile IPv6. Il permet entre autres à des équipements situés sur le même lien physique de se découvrir mutuellement, de découvrir leurs adresses niveau 2 et de localiser les équipements de routage. Le processus de découverte des routeurs d'accès se déroule de manière similaire au protocole de découverte des agents ; tout routeur d'accès émet périodiquement des RA contenant son (ses) préfixe(s) sur le lien. Un nœud mobile peut éventuellement demander un RA explicitement, par l'émission d'un *Router Solicitation* (message qu'on notera dans la suite RS). En outre, l'information contenue dans ces RA permet aux nœuds mobiles de créer une adresse temporaire par auto-configuration [176]. Ensuite il leur faudra vérifier l'unicité de celle-ci grâce au protocole de détection de duplication d'adresse [176].

L'ensemble du mécanisme de gestion de la mobilité est décrit dans la figure 1.3. Les implications du processus de découverte et de création d'adresses sont analysées et évaluées dans la section 1.5.1.

1.3.2 Approche de l'adressage logique

Comme décrit dans la section 1.2.4, une adresse IP est utilisée aussi bien pour identifier un nœud que pour le localiser puisqu'une partie de celle-ci est construite en fonction du point d'attache à l'Internet. Or la gestion de la mobilité a montré les limites de cette double fonctionnalité des adresses IP. Cette observation a favorisé la conception d'un ensemble de propositions qui se basent sur la séparation entre identificateur du nœud et localisation du nœud. En effet, nous pouvons citer VIP pour IPv4 [174, 175] et LINA [83] avec son application pour IPv6 LIN6 [91, 82], LAR [139, 140], une solution d'adressage logique, ou encore HIP pour *Host Identity Protocol* [137] qui fait l'objet d'un nouveau groupe de travail à l'IETF [79]. Le point clé de ces propositions est la séparation des identificateurs logiques des identificateurs physiques. La couche réseau se voit alors divisée en deux sous-couches : une sous-couche réseau virtuelle et une sous-couche de réseau physique. L'association entre le niveau virtuel et physique est réalisée sur le nœud lui-même.

Dans la suite de cette sous-section, nous allons détailler le fonctionnement de LIN6, solution la plus ancienne (une première spécification nommée VIP pour *Virtual Identification Protocol* pour IPv4 avait été faite début des années 1990) et toujours maintenue à jour actuellement, notamment à l'IETF [173], dans le cadre du groupe de travail sur le "multihoming des site" [130].

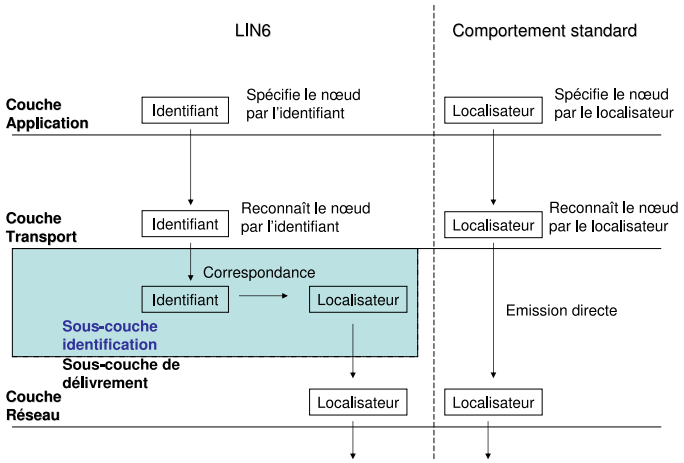


FIG. 1.4 – Séparation en deux de la couche réseau

Détail du protocole LIN6

LIN6, pour *Location Independent Network Architecture for IPv6*, est une mise en œuvre de LINA (*Location Indenpendant Network Architecture*) où la séparation des identificateurs logiques des localisateurs physiques des nœuds de l'Internet est mise en avant. Comme le montre la figure 1.4, la couche réseau est divisée en deux sous-couches : la sous-couche d'identification et la sous-couche de localisation. Généralement les communications sont établies par une adresse LIN6 unique, attribuée au nœud, qui représente l'identification de ce nœud. L'association entre cette adresse d'identification et l'adresse physique de localisation est établie dans la sous-couche d'identification grâce à une table d'association maintenue à jour par tout nœud (nœud final ou routeur) LIN6. Cette table d'association est mise à jour par des paquets de contrôle envoyés par le nœud mobile, à chaque déplacement.

Une adresse LIN6 est construite sur les mêmes principes qu'une adresse IPv6, à savoir 64 bits réservés pour le préfixe et 64 bits utilisés pour identifier le nœud. Comme les adresses LIN6 sont uniquement des adresses servant à identifier un nœud, la partie préfixe d'une adresse LIN6 est non routable. L'utilisation du même format que les adresses IPv6 est à la fois évidente et ingénieuse : cela permet la compatibilité avec les appels socket courants qui s'attachent sur des adresses IPv6 et permet également une association simplifiée avec les localisateurs. En effet, la partie identification du nœud de l'adresse IPv6 est la même que celle de l'adresse LIN6.

La dernière notion importante de cette solution est l'agent de localisation. Chaque nœud mobile a un agent de localisation désigné, qui sera utilisé par défaut par ses correspondants. L'agent de localisation n'effectue pas de transfert de paquets de données, mais maintient simplement à jour la localisation courante des nœuds mobiles. Pour une meilleure robustesse du système et une résistance en cas de panne, l'information de localisation des nœuds mobiles peut être maintenue par plusieurs agents de mobilité.

Initialisation d'une communication

A l'ouverture d'une communication, un nœud correspondant fait une requête DNS [37] pour obtenir l'adresse de l'agent de localisation du nœud mobile, en envoyant l'adresse LIN6 d'identification du nœud destinataire. L'interrogation par un serveur de noms permet d'utiliser des mécanismes de sécurité existant comme AAA [13]. Ensuite le nœud correspondant interroge l'agent de localisation pour connaître la localisation courante du nœud mobile. Puis, à chaque déplacement du nœud mobile, l'agent de localisation et les nœuds correspondants courants du nœud mobile sont informés de la nouvelle localisation par des paquets de contrôle.

Comparaison avec Mobile IPv6

Bien entendu beaucoup de concepts et de fonctionnalités sont très proches dans Mobile IPv6 et LIN6. L'agent de localisation de LIN6 peut être mis en relation avec l'agent mère de Mobile IPv6, la table des associations est présente dans les deux solutions et les messages de mise à jour de localisation sont quasiment identiques. Cependant, les différences entre ces deux méthodes sont importantes, comme nous allons le voir.

Tout d'abord, LIN6 semble proposer une méthode plus adaptée à la mise à l'échelle de l'Internet : comme l'adresse identifiant le nœud n'est pas en relation avec un quelconque réseau, la désignation d'agent de localisation n'est pas liée à une position particulière dans l'Internet. Plusieurs agents de localisation peuvent alors être répartis dans tout l'Internet et la panne d'un agent de localisation peut être gérée aisément par le changement de l'agent désigné. Des travaux récents visent à résoudre le problème de cet unique point de cassure dans Mobile IPv6 [187].

Par ailleurs, dans Mobile IPv6, l'association entre adresse virtuelle (adresse mère) et adresse physique (adresse temporaire) est seulement réalisée sur le nœud destination, c'est pourquoi tout paquet de données échangé entre un nœud mo-

bile et son correspondant doit contenir l'adresse mère du nœud mobile. L'inclusion d'une adresse mère dans un paquet de données est réalisée par l'ajout d'un en-tête. Dans LIN6, l'association est faite aussi bien sur le nœud destinataire que sur le nœud émetteur. Aucune information supplémentaire ne doit donc être transportée dans les paquets de données échangés entre deux nœuds. Un léger temps de traitement supplémentaire (évalué à 1 micro secondes dans [83], ce qui représente 40% du temps de traitement total) permet donc d'alléger la taille des en-têtes des paquets qui traversent le réseau. Si on considère que dans les prochaines années la part des nœuds mobiles dans l'Internet va considérablement augmenter [60], les en-têtes supplémentaires utilisés dans Mobile IPv6 peuvent avoir un impact non négligeable sur la quantité de données échangées dans le réseau.

D'un autre côté, la solution Mobile IPv6 s'intègre mieux dans le modèle existant de l'Internet. Cette solution ne requiert aucun changement dans les nœuds correspondants (par l'utilisation de tunnels) et uniquement une modification dans la couche réseau des nœuds mobiles. LINA non seulement préconise des changements dans les couches réseau, transport et application mais rompt également la modélisation en couches TCP/IP.

1.3.3 Approche au niveau de la couche transport

La gestion de la mobilité dans l'Internet peut également être faite au niveau de la couche transport du modèle TCP/IP. L'ensemble des propositions qui prônent l'utilisation de la couche transport pour la gestion de la mobilité peut être classifié par les trois points suivants [17] :

- Mécanismes adaptés aux particularités des réseaux sans fil
- Utilisation simultanée de plusieurs interfaces
- Contrôle de la mobilité d'une manière bout-à-bout

Généralement, les protocoles adaptés aux particularités des réseaux sans fil ne permettent pas un contrôle de la mobilité bout-à-bout. Comme nous allons le voir, une entité du réseau est nécessaire pour différencier le contrôle de transport sur la partie sans fil du contrôle de transport sur la partie filaire. D'un autre côté, les protocoles permettant l'utilisation simultanée de plusieurs interfaces n'imposent pas l'utilisation d'une entité supplémentaire dans le réseau.

Afin de présenter un aperçu des solutions de gestion de la mobilité au niveau transport, deux solutions parmi les plus reconnues seront présentées. Il s'agit tout d'abord de SCTP (pour *Stream Control Transport Protocol*) [170, 69] et de I-TCP

(pour *Indirect TCP*) [28].

Contrôle bout-à-bout

Stream Control Transport Protocol (SCTP) [170, 69] est un protocole de transport initialement défini pour transporter les messages de signalisation PSTN pour la téléphonie au-dessus de réseaux IP. Au niveau du contrôle du flux, SCTP se rapproche de TCP dans le sens où il assure la non-duplication des paquets de données, garantis sans erreur et le séquencement correct des paquets remontés aux couches supérieures. De plus, SCTP propose deux propriétés supplémentaires qui se révèlent intéressantes, le multihoming et le multi-streaming. Le support du multihoming est assuré par la possibilité pour les deux extrémités d'établir une connexion avec plusieurs adresses (pour chaque extrémité). Cette caractéristique permet notamment de rendre le protocole robuste aux pertes du niveau réseau, puisque l'adresse destination pourra être changée si jamais une adresse n'est plus joignable. Le multi-streaming permet de gérer plusieurs flux indépendamment pour une seule connexion SCTP (un paquet perdu d'un flux n'influera pas sur les tailles des fenêtres de retransmissions des autres flux). La spécification de ce protocole fait l'objet d'un groupe de travail à l'IETF nommé *Transport Area Working Group* (tsvwg) [180].

Avec le besoin grandissant de gérer la mobilité des équipements, SCTP est apparu comme une solution potentielle pouvant gérer la mobilité des nœuds [88]. Seules quelques extensions ont été nécessaires pour permettre la gestion de la mobilité par SCTP, à savoir la possibilité d'ajouter, de détruire et de choisir dynamiquement une adresse dans l'ensemble des adresses utilisées par une connexion [63]. Cependant, une seule composante de la gestion de la mobilité manque dans ce modèle, il s'agit de la localisation initiale des équipements. En effet, comment un nœud correspondant pourra joindre le nœud mobile si ce dernier est rattaché à un réseau visité ?

Aydin et al. [24] propose une solution complète de gestion de la mobilité, en utilisant les mécanismes de l'IETF et la partie localisation de l'équipement fourni par SIP (SIP est détaillé dans la section suivante). Trois modules sont utilisés dans l'architecture d'un terminal pour gérer la mobilité :

- Un agent situé au niveau de la couche réseau est chargé de détecter un nouveau lien et de créer une nouvelle adresse IP si nécessaire.
- Un module cSCTP dans la couche transport est chargé d'envoyer la nouvelle adresse temporaire et d'avertir le nœud correspondant qu'un handover est

en cours.
- Un agent utilisateur SIP est chargé de localiser les nœuds mobiles à l'initialisation d'une communication.

Lorsque le nœud mobile détecte un mouvement, il envoie un message au nœud correspondant afin de lui annoncer sa nouvelle adresse, grâce au message défini dans [63]. En supplément, le nœud mobile indique qu'il est en procédure de handover. Ce message a pour effet la réduction de la fenêtre d'anticipation SCTP et déclenche l'émission des paquets aux deux localisations du nœud mobile, à savoir l'ancienne et la nouvelle. Ce mécanisme appelé *Bi-casting* est utile lorsque le nœud mobile peut continuer à recevoir des paquets sur son ancienne localisation pendant la procédure de handover. La notion de Bi-casting est présentée plus en détails dans la section 1.5.3. Puis lorsque le nœud mobile considère que l'ancienne adresse n'est plus valide, il envoie un message au nœud correspondant pour lui indiquer que le handover est terminé et que l'ancienne adresse doit être effacée. C'est également à ce moment que la fenêtre d'anticipation peut être remise à sa valeur pré-handover.

Gestion par la division de la connexion

D'autres propositions suggèrent une séparation de la connexion de niveau transport en deux parties [28, 29, 103]. L'idée qui émerge de ces mécanismes est la gestion optimisée de connexions TCP sur des liens sans fil. Une connexion TCP est alors séparée en deux connexions, une entre le nœud mobile et un nœud intermédiaire et une autre entre le nœud intermédiaire et le nœud correspondant. L'objectif est bien entendu de rendre cette séparation transparente aux nœuds correspondants, voire également aux nœuds mobiles. L'entité dans le réseau, par laquelle tous les paquets de la communication devront passer, est un point d'accès étendu dans I-TCP (*Indirect - TCP*) [28, 29] ou un proxy dans MSOCKS [103]. La séparation de la connexion permet d'éviter les dégradations de TCP dues à la mobilité des nœuds et aux caractéristiques des interfaces sans fil.

La différence entre les deux méthodes citées ci-dessus est la gestion du déplacement et la prise en compte du "multihoming" : dans I-TCP [31], le point intermédiaire de la division de connexion change en fonction des mouvements du nœud mobile puisque c'est le point d'accès courant du nœud mobile. La connection TCP est divisée en deux connexions sans que le nœud mobile ou le nœud correspondant ne s'en rendent compte. Cette méthode est surtout utilisée pour améliorer les connexions TCP par rapport aux déplacements des nœuds et aux conditions de transmission sur des interfaces sans fil. Cette méthode ne peut donc

pas être utilisée en tant que solution unique et globale à la mobilité, puisqu'une solution comme Mobile IP est tout de même nécessaire pour gérer la mobilité entre différents sous-réseaux.

Par contre, dans le cas de MSOCKS, le même proxy est utilisé quels que soient les mouvements du nœud mobile. Le nœud mobile établit une connexion avec le proxy en lui indiquant les identificateurs du nœud correspondant. Ensuite le proxy établit une connexion vers le nœud correspondant souhaité. Cependant, le fait que la connexion soit divisée en deux ne doit pas influer sur la connexion logique entre le nœud mobile et son correspondant, qui doivent traiter la connexion comme une seule connexion de bout-en-bout. Pour cela, le proxy implémente *TCP Splice* [102] qui rend transparent la division de la connexion aux deux extrémités de la communication. Dans ce cas de figure, la mobilité est gérée par le proxy qui cache la mobilité au(x) nœud(s) correspondant(s). Lorsque le nœud mobile se déplace, il envoie un message *reconnect* au proxy pour mettre à jour sa localisation.

Le déplacement de connexion dans I-TCP se fait par le transfert de l'état de connexion d'un point d'accès à l'autre. L'évaluation de I-TCP [30] a montré que les performances de TCP sont meilleures, quel que soit le scénario, qu'avec un TCP standard. Le débit de données observé est 50% meilleur avec I-TCP lorsque le nœud mobile est en réception et 100% meilleur lorsque le nœud mobile est en émission.

D'un autre côté, [103] a montré que le proxy de MSOCKS résiste bien à la mise à l'échelle, en émulant jusqu'à 250 connexions simultanées sur le même proxy. Le rétablissement de connexion (suite à une redirection ou à un handover) nécessite deux allers-retours et demi entre le nœud mobile et le proxy et est considéré comme le principal temps de latence du handover, sans tenir compte de la détection de mouvements.

Conclusion

Nous venons de voir une autre alternative à Mobile IPv6, à savoir la gestion de la mobilité au niveau de la couche transport. Comme cité ci-dessus, deux grandes familles de protocoles peuvent être distinguées : les protocoles de bout-en-bout et les protocoles utilisant une entité intermédiaire qui intercepte tous les paquets d'une communication.

Les protagonistes de la première solution s'appuient sur le fait qu'une connexion entre deux nœuds de l'Internet doit rester entre ces deux nœuds sans l'interaction d'une tierce partie. Ainsi, il est inutile de déployer des équipements supplémentaires

et leurs protocoles dans l'Internet. Cependant une composante principale manque dans ce modèle : la localisation initiale du nœud mobile.

D'autres pensent qu'il est difficile, voire impossible de changer tous les nœuds de l'Internet pour mettre en place une solution de gestion de mobilité. La présence d'une entité particulière dédiée à gérer les caractéristiques spécifiques aux nœuds mobiles leur semble alors un déploiement nécessaire. Une division de connexion peut se révéler avantageuse pour la séparation du contrôle de flux et du contrôle de congestion entre le domaine sans fil et le domaine de l'Internet. Ces solutions favorisent également le développement de protocoles spécifiques pouvant intégrer des comportements adaptés aux mouvements des nœuds mobiles et à la notification d'évènements. L'utilisation d'une entité spécifique permet même des opérations supplémentaires comme le traitement d'une partie des données sur l'entité intermédiaire pour alléger les opérations sur le nœud mobile, puisque ce dernier a souvent une faible puissance de calcul.

Cependant, les solutions existantes ne sont pas complètes ; I-TCP n'est pas capable de gérer les mouvements des nœuds mobiles entre sous-réseau et SCTP et MSOCKS requièrent une entité supplémentaire pour localiser les nœuds mobiles lors de l'initialisation d'une connexion. De plus, ces mécanismes sont définis pour des communications orientées connexion, alors que les applications de type temps réels utilisent généralement un protocole transport sans contrôle d'erreurs comme UDP.

Par ailleurs, nous pourrons noter qu'il n'y a que très peu d'évaluations de la gestion de la mobilité par le niveau transport au-dessus d'IPv6, car peu de personnes se sont intéressées aux réseaux IPv6 (les évaluations données dans [24, 30, 103] sont basées sur des implémentations pour l'IPv4). Bien que ces solutions de niveaux transport soient indépendantes de la version du protocole IP sous-jacent, le support d'IPv6 dans ces mécanismes n'est pas encore vraiment pris en compte.

1.3.4 Approche au niveau de la couche application : SIP

Une autre méthode de gestion de mobilité est une gestion de niveau applicatif. Le protocole d'initialisation de session (*Session Initiation Protocol* - SIP) [157, 161], servant à créer, modifier et terminer des sessions entre plusieurs participants, peut être utilisé pour gérer la mobilité des nœuds [162, 189]. Dans cette approche, l'infrastructure nécessaire à SIP et les messages utilisés suffisent à une gestion de la mobilité, sans l'utilisation de paquets de contrôle supplémentaires à ceux définis dans SIP ou sans ajout d'en-têtes dans les paquets de données.

SIP comme gestionnaire de session

SIP permet à deux participants ou plus d'initialiser une session, par laquelle ils pourront échanger des données de type multimédia, comme de la vidéo-conférence, ou de la diffusion audio. Une session peut être utilisée pour plusieurs flux multimédia. De nombreuses implémentations de SIP ont déjà été réalisées [166] et SIP a récemment été étendu à la notification d'événements et à la messagerie instantanée [105].

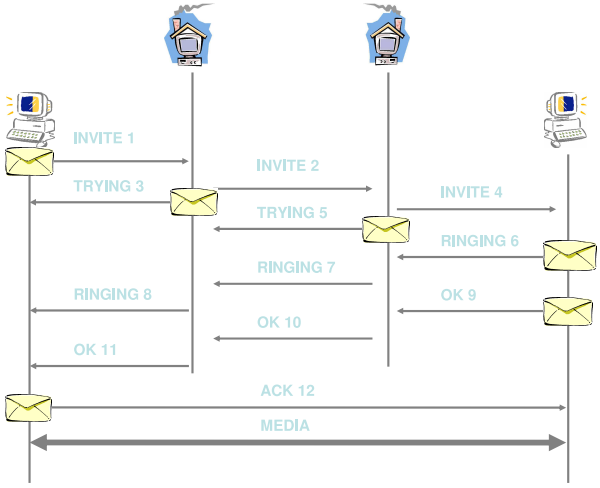
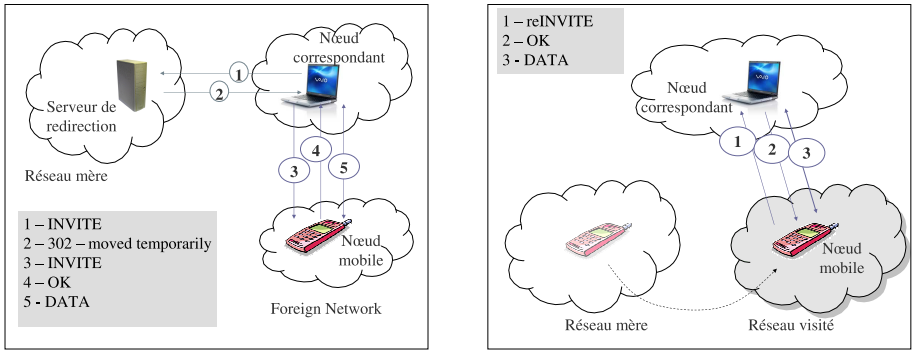


FIG. 1.5 – Etablissement d'une session SIP

Chaque nœud SIP a un identificateur unique de niveau utilisateur qui se présente sous la forme d'une adresse mèl. Chaque extrémité d'une communication est appelée agent utilisateur, qui peut être soit initiateur, soit récepteur d'une demande. C'est la seule entité dans SIP où se rejoignent le média et la signalisation. Des serveurs de redirection et des serveurs de proxy peuvent être utilisés pour rediriger des demandes d'ouverture de session. Certains serveurs proxy conservent des états pour les sessions en cours qu'ils relayent. Ils pourront notamment être employés pour relayer plusieurs copies de paquets de données à différentes localisations. Lorsqu'un nœud désire établir une session, l'identificateur unique est l'adresse destination, déterminée grâce à un mécanisme de résolution de nom [37]. Lorsqu'une demande véhiculée dans un message *INVITE* atteint le domaine d'appartenance du destinataire et que celui-ci est en déplacement, une entité appelée *Registrar* est en charge d'informer l'initiateur de session de la localisation courante du nœud destinataire. Le *Registrar* est donc chargé de maintenir la localisation courante des agents utilisateurs de son domaine. La figure 1.5 retrace l'échange de messages entre deux nœuds pour l'établissement d'une session SIP.

SIP comme gestionnaire de mobilité

Le modèle de gestion de mobilité est très proche de celui de Mobile IPv6. Cependant, bien que Mobile IPv6 fournisse un cadre de gestion de mobilité beaucoup plus générique, SIP permet lui une mobilité plus diversifiée [162, 189] : mobilité du terminal, mobilité d'une session, mobilité personnelle et mobilité de service. SIP permet également l'utilisation de plusieurs adresses de localisation pour un seul identificateur, ce qui s'avère être une caractéristique intéressante de SIP pour la gestion d'interfaces multiples. Peu d'études cependant se sont focalisées sur cet aspect de SIP.



(a) Gestion de la mobilité pré-appel dans SIP

(b) Gestion de la mobilité en cours de session dans SIP

FIG. 1.6 – Gestion de la mobilité dans SIP

En ce qui concerne la mobilité horizontale, c'est-à-dire le déplacement du nœud mobile, SIP distingue deux types de mobilité (voir figure 1.6) : la mobilité pré-appel, qui est la mobilité de l'équipement avant de recevoir un appel et la mobilité milieu-d'appel, où le nœud mobile se déplace en cours de session. Nous avons déjà vu dans la section ci-dessus comment la mobilité pré-appel était gérée par l'utilisation du *Registrar* qui informe tout initiateur de session de la localisation courante du nœud mobile.

La mobilité en cours de session fonctionne de la manière suivante. Après détection de mouvements (algorithme indépendant de SIP, qui peut être le même que celui de Mobile IPv6 par exemple), le nœud mobile envoie un message *re-INVITE* au nœud correspondant. Ce message est utilisé pour renégocier les paramètres de la session existante. Il peut donc être employé pour annoncer au nœud correspondant une nouvelle adresse de destination. Bien entendu, le nœud mobile devra égale-

ment prévenir son *Registrar* afin de mettre à jour sa localisation courante pour les appels futurs. Plusieurs mécanismes de sécurité comme PGP [23] peuvent être mis en place pour empêcher un nœud malicieux du réseau de modifier la position du nœud mobile à sa place.

Des optimisations sont possibles, notamment par la mise en place d'une hiérarchie (le modèle hiérarchique pour Mobile IPv6 est présenté dans la section 1.5.4). Dans un tel modèle, les mouvements des nœuds mobiles à l'intérieur d'un domaine sont cachés aux correspondants. Les serveurs de proxy peuvent être utilisés comme points d'ancrages et facilitent grandement le déploiement de modèles hiérarchiques dans l'Internet puisqu'il n'est pas nécessaire de déployer de nouveaux équipements.

Par contre, dans la version de base de SIP, le mouvement simultané des deux extrémités d'une communication n'était pas géré. Cette constatation a donné lieu à la rédaction d'une extension [57] afin de mettre en place un mécanisme de minuterie de session SIP. Ce protocole impose le rafraîchissement de la session à intervalles réguliers. Si l'extrémité de la communication ne répond plus, il faut repasser par le *Registrar* du nœud correspondant. Bien que ce mécanisme permette de maintenir la session, il est complètement dépendant de l'intervalle de rafraîchissement, qui impose un échange supplémentaire de messages.

Comparaison avec Mobile IPv6

Comme introduit dans la section 1.2, SIP et Mobile IPv6 partagent de nombreux concepts. Néanmoins, une différence majeure entre ces deux protocoles est que SIP permet l'échange des données d'un flux directement entre les deux communicants sans état intermédiaire, alors que l'optimisation de routage dans Mobile IPv6 n'est qu'une option du protocole (et n'est d'ailleurs pas le comportement par défaut). Cette caractéristique permet un dimensionnement plus aisé des agents de localisation dans SIP puisqu'ils ne sont responsables que de la mise à jour de la localisation courante des nœuds mobiles, alors que dans Mobile IPv6, un agent mère est chargé de relayer les paquets de données pour les nœuds mobiles gérés. De plus, SIP n'utilise pas d'en-têtes supplémentaires dans les paquets de données, alors que Mobile IPv6 impose que tous les paquets de données contiennent l'adresse mère du nœud mobile.

L'identificateur utilisé par SIP fournit également un avantage quant au déploiement d'une solution de gestion de mobilité : dans Mobile IPv6, c'est généralement le fournisseur de service Internet qui sera en charge de déployer des agents mères et de fournir des adresses mères aux nœuds mobiles, alors que dans SIP, l'identi-

ficateur unique peut être distribué par n'importe quel domaine de l'Internet aux agents utilisateurs.

En ce qui concerne la performance de SIP par rapport à celle de Mobile IPv6, il est difficile de dire qu'une solution est meilleure que l'autre. Cependant, trois points peuvent être identifiés. Premièrement, la détection de mouvements et la création d'une nouvelle adresse temporaire, qui constitue une part importante du temps de latence d'un handover, peuvent être les mêmes opérations dans SIP que dans Mobile IPv6. Cependant, la détection de mouvements et le changement d'adresse sont des opérations réalisées au niveau de la couche réseau du modèle TCP/IP. La pile Mobile IPv6 est donc informée naturellement des changements de lien. Par contre, dans SIP, il faut mettre en place un mécanisme spécifique afin de remonter l'information de changement de lien au niveau applicatif [132].

Deuxièmement, la mise à jour de localisation dans SIP passe par un message *re-INVITE*. Ce message a été défini pour établir ou modifier la description d'une session, dont l'adresse IP n'est qu'une partie. Lorsqu'on pense à l'utilisation d'interfaces multiples et plus particulièrement à des handovers verticaux (voir la section suivante), ce même message peut également être utilisé pour changer les caractéristiques du flux. En effet, le passage d'une technologie à une autre engendre généralement une baisse ou une augmentation des capacités de réception ou d'émission du nœud en question. Le message de mise à jour de localisation de SIP peut alors également être utilisé pour ralentir ou augmenter le débit du (des) flux concerné(s) par la session. Mobile IPv6 ne fournit pas de telles possibilités pour le moment.

Troisièmement, le temps de latence du handover est supérieur dans le cas de SIP, aussi bien théoriquement [92] que dans la pratique [132]. [92] a montré qu'en moyenne le temps de handover de SIP était 1/3 supérieur au temps de handover de Mobile IPv6, que ce soit pour des handovers intra- ou inter-domaines. [132] a mis en place une plate-forme de tests de petite taille (deux routeurs d'accès, un nœud mobile, deux nœuds correspondants) et a mesuré le temps que met le nœud mobile à recevoir le nouveau flux à la nouvelle localisation. Ce temps est de 30 ms pour Mobile IPv6 contre 400 ms pour SIP. Ces temps utilisent la même méthode de détection de mouvements et n'incluent pas le temps du changement physique de point d'accès.

1.4 La mobilité verticale

La mobilité verticale est un aspect très récent de la mobilité, apparue fin des années 1990. La mobilité verticale est une mobilité interne à l'équipement, dans le sens où il s'agit de redirections entre interfaces réseau du même équipement. L'exemple typique est un utilisateur équipé d'un ordinateur portable ayant une interface Ethernet [11] et une interface sans fil Wifi [2]. Alors qu'il entre dans le bâtiment où il travaille, il accède à Internet par son interface Wifi. Arrivé dans son bureau, il connecte sa prise Ethernet et désire faire basculer ses communications sur cette nouvelle interface disponible, offrant de meilleures caractéristiques techniques (e.g. plus de fiabilité, meilleur débit de données).

Cependant, la mobilité verticale ne se résume pas seulement à une utilisation exclusive des interfaces réseau disponibles sur un équipement ; l'utilisation **simultanée** de plusieurs interfaces est également un des objectifs premiers sur un terminal multi-interfaces. En effet, l'hétérogénéité des technologies de communication est telle aujourd'hui qu'il est difficile de "préférer" l'une d'entre elles pour toutes les applications et pour tous les modèles de mobilité. Nous pouvons remarquer ici que l'équipement n'est pas forcément mobile lui-même. L'utilisation simultanée d'interfaces multiples peut se faire sur un poste fixe. Cependant, lorsque le terminal est lui-même mobile, la disponibilité des interfaces est beaucoup plus dynamique et demande une gestion rapide et optimisée, c'est pourquoi nous nous intéresserons principalement à des nœuds mobiles dans la suite.

Bien que Mobile IPv6 fut initialement développé pour gérer la mobilité horizontale, il peut être étendu pour supporter la mobilité verticale [126, 186]. Cependant, dans cette section nous nous intéresserons plus particulièrement aux architectures de gestion de la mobilité afin d'étudier des modèles plus globaux intégrant une certaine adaptation du terminal par rapport à son environnement. Effectivement, l'utilisation d'interfaces multiples ne consiste pas uniquement à la gestion des redirections, mais également à la sélection d'adresses / d'interfaces, à l'algorithme de redirection et aux possibilités de choix offertes par le système à l'utilisateur.

Les sous-sections suivantes donnent un aperçu des architectures et des protocoles proposés dans la littérature pour la gestion des interfaces multiples sur un terminal mobile.

1.4.1 Le projet Moby Dick et l'architecture Mirai

Le projet Moby Dick [113] - *The Mobile Digital Companion*, qui était initialement un projet européen, regroupe un consortium de trois universités qui travaillent sur plusieurs aspects de l'Internet ubiquitaire. La philosophie de ce projet est d'unifier l'ensemble des fonctionnalités offertes par différents équipements électroniques tels qu'un assistant personnel ou un téléphone portable sur un seul terminal. Les principaux axes de recherche du projet Moby Dick sont la définition d'une architecture de réseau pour intégrer une re-configuration du système liée à la qualité de service, la minimisation de consommation d'énergie sous différentes contraintes comme le type de l'équipement cible (capteur, assistant personnel, ordinateur portable) ou des contraintes de qualité de service, la sécurité, l'étude, le développement et la validation d'une architecture dédiée à des services multimédia et d'une architecture pour le réseau consacrée à des services temps réels pour des réseaux sans fil hétérogènes. Cette dernière thématique nous intéresse tout particulièrement dans le cadre de gestion d'interfaces multiples. Ce projet se montre fort intéressant de par la vision globale qu'il propose de l'intégration de technologies hétérogènes.

L'architecture MIRAI[2] [191, 190] est un projet du plan e-Japan financé par le gouvernement japonais (il est question d'un déploiement pour le grand public de la technologie développée). L'architecture a pour objectif l'intégration transparente pour l'utilisateur final de différents réseaux hétérogènes, pouvant être utilisés simultanément, la découverte de ces réseaux, la gestion de la mobilité, la localisation des nœuds inactifs, la sécurité et le support de qualité de service. Le concept du support de l'ensemble de ces fonctionnalités passe principalement par un module radio sur le terminal, configurable selon l'environnement et par une unique fonction d'accès, ce qui facilite le processus de connexion au réseau par les utilisateurs.

Les éléments constituant l'architecture MIRAI sont les suivants :

- *Multi Service User Terminal* (MUT) : logiciel radio multi-services, configurable pour le terminal
- *Common Core Network* (CCN) : réseau cœur IPv6 reliant plusieurs réseaux d'accès radio hétérogènes entre eux. Il procure de la qualité de service et permet de faire des handovers rapides. Il intègre un gestionnaire de ressources et un gestionnaire de mobilité qui maintiennent un ensemble de paramètres par utilisateur pour coordonner la distribution du trafic et la sélection du meilleur réseau d'accès radio.
- *Basic access network* (BAN) : canal de contrôle et de signalisation pour

[2]Mirai : Multimédia Integrated network by Radio Access Innovation

permettre à tous les MUT d'accéder au réseau. Cette entité assure la mise à jour de localisation des nœuds, la recherche de la localisation des nœuds inactifs et permet des handovers verticaux vers les autres technologies radio. Entre autres, le BAN fournit l'information aux nœuds mobiles sur les réseaux d'accès radio adjacents, sans que le nœud mobile n'ait besoin de mettre en œuvre une procédure de découverte de son environnement radio.

Mobile IPv6 a été désigné comme gestionnaire de la mobilité entre plusieurs CCN. L'adresse du routeur de sortie du CCN est utilisée comme adresse temporaire des nœuds mobiles à l'intérieur du CCN. Cette gestion est proche du modèle hiérarchique de la gestion de la mobilité décrite en 1.5.4. L'ensemble des fonctionnalités décrites a été implémenté sur un prototype en 2001 avec les technologies PHS et IEEE 802.11b [81].

Un autre point intéressant de ce projet est la définition des paramètres à prendre en compte pour la sélection d'interface, c'est-à-dire la sélection de la technologie d'accès. Les paramètres proposés sont les suivants :

- Besoins en qualité de service, exprimés par les applications
- Préférences des utilisateurs
- Capacité du terminal
- Statut du CCN et du réseau d'accès radio
- Localisation du terminal

Ce projet a abordé le problème multi-interfaces de façon très complète et fort intéressante. Cependant, la vision de l'intégration de technologies hétérogènes repose sur une vision du contrôle total par le réseau. Comme nous l'avons vu, des entités supplémentaires sont nécessaires, comme le CCN, et de nombreuses modifications sont à apporter à plusieurs niveaux : modification d'applications, modification du module radio, etc.

1.4.2 Le projet MosquitoNet

MosquitoNet [26] est un projet du centre des télécommunications de l'université de Stanford aux Etats-Unis. MosquitoNet a pour but d'offrir une connectivité Internet continue à des nœuds mobiles IPv4 [27]. Bien que ce projet ait été développé pour l'Internet IPv4, il introduit de nombreux concepts qui seront fortement réutilisés par la suite. Le support d'une connectivité continue passe par deux objectifs : permettre à un nœud mobile de changer d'interface pour ses communications et permettre à un nœud mobile son déplacement dans un réseau ne supportant pas

la mobilité (dans Mobile IPv4 [199], chaque réseau visité doit être muni d'un agent spécial qui permet au nœud mobile de recevoir les paquets qui lui sont destinés). Ce deuxième objectif a été largement atteint avec l'arrivée d'IPv6, puisque Mobile IPv6 intègre maintenant les concepts introduits ici : le nœud mobile obtient une adresse dans le réseau visité et c'est lui qui traite directement les paquets reçus.

Dans la continuité de ce projet, [198] propose au nœud mobile un choix de route pour ses communications et une méthode d'utilisation simultanée d'interfaces multiples. Le choix de route devra se faire entre le routage triangulaire de Mobile IP, le tunnel bi-directionnel de Mobile IP ou alors ne pas utiliser les fonctionnalités de Mobile IP. Dans ce dernier cas, le nœud mobile démarre ses communications directement avec son adresse temporaire et devra ré-initialiser la communication en cas de déplacement. Cette solution est bien adaptée pour des applications à courte durée de vie comme l'accès au World Wide Web.

L'utilisation d'interfaces multiples de manière simultanée passe par la notion de filtrage : le nœud mobile peut ajouter une option dans ses mises à jour de localisation pour indiquer à son agent mère quel flux envoyer sur quelle interface, identifiée par son adresse. De plus amples détails sur les opérations de filtrage sont donnés dans la sous-section 1.4.5.

De plus, la distinction entre redirection à froid (*Cold Switching*) et redirection à chaud (*Hot Switching*) a déjà été introduite dans [27]. Ces concepts fondamentaux de redirection ont un impact essentiel sur les temps de handovers verticaux ; une redirection à froid est la perte de l'interface utilisée (perte de connexion par exemple), la mise en route et la configuration d'une autre interface et enfin la redirection des flux sur cette interface. Cette redirection est très critique par opposition à la redirection à chaud où les deux interfaces sont disponibles au moment de la redirection. Pendant le temps de la redirection, les paquets continuent alors à être reçus sur l'interface initiale. Nous verrons dans le chapitre 6 que nous faisons aussi la distinction entre ces deux types de redirection. Une implémentation sous Linux a été faite (sur le noyau 1.2.1.3), constituant une des premières implémentations de Mobile IP et des résultats sur les performances de redirection peuvent être trouvés dans [45].

1.4.3 Les solutions de séparation entre identification et localisation

Comme décrit dans la sous-section 1.3.2, des alternatives à Mobile IPv6 ont été proposées pour gérer la mobilité des nœuds dans l'Internet en séparant les

notions d'identification et de localisation. Il s'avère que ces solutions, de par leur conception, intègrent de manière naturelle les interfaces multiples et le multihoming. Effectivement, il a suffit d'ajouter la notion d'associations multiples pour le même identificateur de machine pour gérer l'utilisation d'interfaces multiples. Le fait que l'identificateur du nœud mobile ne soit pas basé sur une interface du nœud contribue à la simplicité de ces solutions.

Les détails relatifs à la mise en œuvre de la gestion d'interfaces multiples et de "multihoming" sont donnés dans [106, 173]. Entre autres, une nouvelle interface pour les Sockets permettant de mieux gérer plusieurs localisations est définie. Cette nouvelle interface permet notamment de choisir l'adresse principale à utiliser pour une communication. Cette fonctionnalité permet une sélection d'adresse par communication. La nouvelle interface d'échanges définie offre également un mécanisme de changement d'adresse suite à une faille de l'adresse courante utilisée.

Des modifications mineures au protocole de base permettent finalement d'enregistrer une liste d'adresses de localisation associée à une adresse principale, jouant le rôle d'identificateur et offrent la possibilité de recourir à une autre adresse / interface en cas de faille. Cependant, l'utilisation simultanée d'interfaces multiples n'est pas abordée.

1.4.4 Intégration de réseaux locaux sans fil (WLAN) dans les réseaux cellulaires

La gestion d'interfaces multiples au sein d'un même terminal concerne également la réunion des réseaux locaux sans fil (WLAN) et des réseaux cellulaires. Cette intégration représente la jonction de deux mondes jusqu'ici bien distincts : la téléphonie cellulaire sans fil et l'Internet. Cette convergence est appelée 4G, pour *Réseau Cellulaire Quatrième Génération* [109]. Une plus ample explication de l'évolution des réseaux cellulaires est donnée dans 2.4. Cependant, le futur modèle de cette intégration n'est pas claire à ce jour. [100] fait même apparaître une distinction entre deux visions différentes de la 4G ; une première définition serait une nouvelle interface radio née du mouvement de la 3G vers un standard unique et complet, qui procure des débits plus importants et une meilleure adaptation à l'environnement. L'intégrateur de la convergence est donc vu ici comme un logiciel radio capable de s'adapter à son environnement.

Une deuxième vue sur la 4G est plus simplement un ensemble de réseaux sans fil hétérogènes, basés sur IP. Cette vision représente plutôt une architecture évoluée et hétérogène dont le lien commun serait le protocole IP [59]. Tout le monde

s'accorde à dire que les enjeux de la convergence de ces deux mondes seront la personnalisation du contexte, l'accès rapide et sans coupure, la gestion de la mobilité, la qualité de service et la facturation. Bien que tous ces aspects ne nous intéressent pas directement, les modèles proposés sont intéressants en matière de gestion d'interfaces multiples.

[59] propose trois axes de recherches pour la gestion du handover vertical : l'interopérabilité entre réseaux hétérogènes, la réduction du délai de localisation et la maintenance des paramètres de qualité de service. L'interopérabilité entre réseaux a fortement été étudiée dans [95, 160] qui classifient le couplage entre WLAN et réseaux cellulaires en trois catégories détaillées dans les points suivants :

- Couplage inexistant : dans ce cas, les réseaux WLAN sont développés avec leurs propres méthodes d'authentification, de contrôle d'accès et de gestion de la mobilité. L'utilisation simultanée d'interfaces multiples est laissée à l'utilisateur final qui pourra répartir ses flux en fonction des interfaces qui lui seront disponibles. Aucun lien, aussi bien au niveau authentification que continuité de services, n'est fourni entre les réseaux WLAN et cellulaires.
- Couplage faible : dans le couplage faible, le WLAN est déployé comme un réseau complémentaire, relié au point d'extrémité du réseau de l'opérateur. Cette solution facilite le regroupement de différents opérateurs réseaux pour un même fournisseur du service Internet [15, 95].
- Couplage fort : le WLAN est fourni comme un réseau d'accès radio (RAN dans la terminologie des réseaux cellulaires) classique de l'architecture GPRS. Le réseau cœur de l'opérateur est donc utilisé pour transporter les données échangées sur le WLAN, qui est donc vu comme un réseau d'accès radio semblable aux stations de base GPRS.

Le deuxième axe de recherche concerne la réduction du temps de latence des handovers verticaux. Ce temps de latence dépend de la méthode de gestion de la mobilité choisie : peuvent être utilisés des protocoles orientés circuit (GSM), des protocoles orientés paquets (Mobile IP) ou l'utilisation de tunnels comme dans GPRS [165]. Cependant, comme la *colle* de l'intégration est IP, Mobile IP est fortement considéré comme gestionnaire de mobilité, au moins au niveau de la macro-mobilité. La réduction des délais peut alors passer par la mise en œuvre de handovers rapides et/ou d'architectures hiérarchiques (voir 1.5).

Le troisième point est le maintien de la qualité de service, qui s'avère être difficile étant donnée l'hétérogénéité des technologies ; les débits sont très variables d'une technologie à l'autre et le maintien d'une certaine qualité de service semble alors difficile. Par contre, l'adaptation des applications par rapport à l'environnement du nœud mobile et aux technologies utilisées apparaît comme une approche

appropriée [17].

Bien que la littérature soit très riche en articles traitant du sujet de l'intégration des WLAN dans les réseaux cellulaires, peu d'entre eux s'intéressent à l'utilisation simultanée d'interfaces multiples. La recherche pour l'intégration des WLAN dans les réseaux cellulaires est plutôt focalisée sur la réalisation effective de cette intégration, la convergence vers un monde tout IP et un système de contrôle de l'utilisateur.

1.4.5 Filtrage des flux

La pièce manquante de Mobile IPv6 pour l'utilisation simultanée d'interfaces multiples sur un terminal est sans doute le fait qu'une seule adresse temporaire peut être associée à une adresse principale sur un nœud. Effectivement, lorsque le nœud mobile envoie une mise à jour de sa localisation, l'adresse annoncée vient remplacer la précédente.

Wakikawa et al. analysent la gestion du multi-interfaces sous deux angles : il faut un mécanisme dans Mobile IPv6 qui permette d'utiliser plusieurs adresses temporaires simultanément [186], et d'autre part, il est nécessaire d'établir des règles de routage afin d'utiliser simultanément plusieurs interfaces [188]. La mise en œuvre et le processus de décision doivent donc être indépendants l'un de l'autre. Bien entendu, les règles de routage devront être dynamiques pour prendre en compte le caractère volatile des interfaces sans fil et devront également tenir compte des préférences de l'utilisateur.

La mise en œuvre de filtrage de flux passe par la possibilité pour le nœud mobile d'enregistrer plusieurs adresses temporaires sur un nœud distant, en y insérant des règles de routage par rapport à l'identification du flux. Le choix de l'adresse temporaire déterminera alors l'interface qui sera utilisée. Une implémentation du mécanisme sous FreeBSD a permis une première évaluation de ce mécanisme. La surcharge du traitement d'un paquet sortant sur un nœud correspondant ayant plusieurs règles a été évaluée à 2% de temps de traitement supplémentaire. Les temps de handovers observés sont par contre moins incisifs, se situant entre 4 et 11 secondes pour le passage entre une interface cellulaire et WLAN de type IEEE 802.11 [2]. La différence est due à la fréquence des RA émis sur le lien. Nous verrons plus en détail les caractéristiques de la détection de mouvements dans la section 1.5.1. Une dernière évaluation du nombre de paquets perdus montre que l'utilisation simultanée de plusieurs interfaces (deux dans le cas présent) réduit le nombre de paquets perdus.

Ces notions de règles de routage se rapprochent fortement des travaux réalisés par Fikouras et al. [67, 68, 90] et Soliman, Castelluccia, Zhao et El Malki [167, 198, 197], qui proposent un ensemble d'options pour les messages d'enregistrement de Mobile IP. Ces propositions permettent à tout nœud de l'Internet d'enregistrer un filtre dans le cache d'association des adresses temporaires. Un filtre peut être l'ensemble du quintuplet[3] identifiant une communication ou un sous-ensemble du quintuplet. Lorsque le nœud correspondant désire émettre un paquet, la règle qui correspond le plus au flux en question est sélectionnée.

1.4.6 Les alternatives

D'autres solutions ont été proposées pour gérer les interfaces multiples au sein d'un terminal IPv6. Nous allons voir un aperçu de deux d'entre elles, qui reflètent la recherche faite dans ce domaine.

Une première solution préconise un changement conséquent du modèle TCP/IP [17] : adaptation des applications, modification de la couche transport pour gérer le multihoming, adaptation de la couche liaison et interaction forte entre les couches. L'adaptation du niveau applicatif revient à une adaptation du codage en fonction des conditions du réseau de rattachement, et un marquage des paquets les plus importants afin de rendre leur transmission plus fiable. Un échange direct est donc possible entre les applications et la couche liaison. Le protocole de la couche transport avancé est nommé R^2CP (pour *Radial Reception Control Protocol*) [80, 16]. R^2CP est une extension de TCP/RCP qui maintient un ensemble d'états par session (un état par interface), au lieu de maintenir un seul état pour une connexion de type TCP. Cette gestion permet de réduire l'impact des handovers sur les applications TCP et optimise le ré-ordonnancement des paquets grâce à un contrôle de flux par les extrémités de la communication. La couche MAC adaptative, appelée A-MAC, a pour objectif un accès sans coupure à des média hétérogènes [18]. La couche MAC est divisée en deux sous-couches, une sous-couche "maître" chargée de prendre les décisions d'ordonnancement sur les interfaces, et une sous-couche d'accès qui implémente les protocoles d'accès aux différents média (CSMA/CA, CDMA, TDMA).

Une dernière solution rejoint les études faites dans [100], qui émet l'hypothèse qu'à terme les réseaux d'accès intègreront les accès cellulaires, WLAN, mais aussi les réseaux ad hoc et les réseaux de senseurs. P-Handoff [179] a pour objectif de rendre les connexions sans fil transparentes aux applications, à l'heure de l'intégra-

[3]Le quintuplet identifiant une communication est consituée des adresses source et destination, des numéros de port source et destination et du numéro de protocole.

tion de technologies hétérogènes. Bien que cette solution soit présentée uniquement pour les réseaux ad hoc, c'est-à-dire sans infrastructure, l'intégration proposée se révèle fort intéressante.

Les travaux réalisés dans cette étude proviennent de l'observation que la gestion de plusieurs interfaces au sein d'un terminal est difficile. Chaque technologie a une méthode d'utilisation différente, autant au niveau API (Socket TCP/IP pour 802.11, abstraction de *pipe* en série avec RFCOMM ou des sockets dédicacés pour IrDA (IrSock)), qu'au niveau adressage (Bluetooth utilise les adresses MAC, IrDA n'a pas d'adresse et 802.11 utilise le standard TCP/IP/Ethernet). Pour le moment toute cette gestion est généralement laissée à l'utilisateur qui doit configurer lui-même chaque interface et ensuite lancer son application. Les objectifs de l'intégration de ces interfaces multiples sont donc la transparence d'utilisation et de configuration et le choix optimal de la technologie pour chaque application. Ces objectifs feront partie des thèmes récurrents de nos travaux (voir section 6). Pour atteindre ces objectifs, P-handoff, extension de V-Handoff [169], définit une architecture de diversité de connexion modifiant les couches TCP/IP et liaison pour la mise en place d'un contrôle et d'un échange d'événements. Le modèle est le suivant :

- Adaptation IP : IP est vu comme la couche procurant la transparence pour les applications. Pour les technologies de type PPP (IrDA ou Bluetooth), l'émission de paquets IP n'est pas triviale.
- Le *gestionnaire de connexion* est chargé de "réveiller" une interface (y compris établir sa configuration si nécessaire) lorsqu'il décide qu'un flux doit être émis par telle ou telle interface. Particulièrement, la configuration devra se faire sans l'interaction de l'utilisateur, grâce au module de découverte et d'adaptation.
- La *Couche d'adaptation et de découverte* réalise la correspondance entre noms et adresses IP (rappelons que dans le contexte de recherche, il s'agit de communications en mode ad hoc).
- Le *Gestionnaire de règles* stocke les préférences sur les technologies et les applications.

Les mécanismes de gestion de mobilité verticale sont ceux de Mobile IP, à savoir l'utilisation d'une adresse principale pour les communications. La granularité de mobilité est faite par adresse destination ; pour une adresse destination donnée, l'ensemble des technologies utilisables est sélectionné. La prise en compte des événements de la couche liaison est également intégrée dans le modèle. Cette prise en compte se limite à la détection de perte de lien et d'établissement de lien, ce qui permet une réactivité accrue du système.

Bien que cette solution soit définie pour les réseaux en mode ad hoc, elle introduit déjà une architecture pour le terminal lui permettant la découverte rapide des liens disponibles, la configuration automatique des interfaces, et la sélection des liens suivant l'adresse destination et la prise en compte des préférences utilisateurs. Finalement, une solution telle que P-handoff en conjonction avec une solution de handover vertical pour réseaux à infrastructure offrirait une architecture complète pour le terminal lui permettant une utilisation simplifiée et optimisée de plusieurs interfaces avec plusieurs modes de fonctionnement.

1.5 Optimisation des handovers

La démocratisation des équipements sans fil laisse présager un boulversement des méthodes d'accès à l'Internet dans les prochaines années. Les utilisateurs mobiles demanderont évidemment des niveaux de qualité de service proches, voire même identiques à ceux offerts aux utilisateurs de postes fixes. Or, le protocole le plus couramment utilisé actuellement pour gérer la mobilité IP (Mobile IP, voir section 3) fait cruellement ressentir ses limites dans cette perspective. Les délais de handover, la mise à l'échelle du protocole et le nombre de paquets perdus sont considérés comme les points critiques du contrôle de la mobilité. Une telle vision présente donc un certain nombre de défis techniques pour la mobilité IP.

Cette section est consacrée au traitement spécifique du **handover**. Rappelons qu'un handover est le processus engendré lorsqu'un nœud mobile change son point d'attachement au réseau. Généralement le nœud mobile ne peut pas recevoir ni envoyer de paquets de données durant tout le processus de handover, c'est pourquoi le temps de latence du handover est considéré comme critique. Habituellement deux types de handover sont distingués. Le handover de niveau 2 représente les opérations nécessaires au changement de point d'accès. Il correspond à la fin de connexion entre le nœud mobile et son point d'accès courant puis son rattachement à un nouveau point d'accès. Ce changement n'implique pas forcément un changement dans la configuration IP de l'équipement puisque le nouveau point d'accès peut être dans le même sous-réseau d'accès que l'ancien. S'ils n'appartiennent pas au même sous-réseau, le nœud mobile devra réaliser des opérations supplémentaires pour mettre à jour sa configuration IP. L'intervalle de temps compris entre le détachement de l'ancien point d'accès jusqu'à la possibilité de communiquer dans le nouveau réseau d'attachement est appelé handover de niveau 3. Nous pourrons noter qu'un handover de niveau 2 n'implique pas forcément un handover de niveau 3.

Le processus de handover peut être amélioré soit en réduisant le nombre de paquets perdus, soit en diminuant la charge de la signalisation, soit encore en rendant le processus le plus rapide possible. Les protocoles dit de *micro mobilité* sont conçus pour des environnements où les nœuds mobiles échangent généralement des données de type temps réel et/ou changent leur point d'attachement à l'Internet si fréquemment que les performances de Mobile IP se révèlent insuffisantes : surcharge de la signalisation, perte de paquets, délais de connectivité, livraison des données aux applications retardées. La micro mobilité est la gestion fine de la mobilité, souvent dans une aire ou un domaine limité en taille. Des protocoles distincts peuvent alors être utilisés pour gérer la micro mobilité et la mobilité globale (entre aires ou domaines). Un protocole de micro mobilité raffine la gestion de la mobilité en fonction des besoins spécifiques dans l'aire ou le domaine en question.

Par ailleurs, la signalisation engendrée par la gestion de la mobilité s'accroît avec le nombre d'utilisateurs. [108] a montré que dans les réseaux cellulaires, la gestion de la mobilité nécessitait 4 à 11 fois plus de trafic (que la téléphonie fixe par exemple). Dans les réseaux cellulaires, des techniques spécifiques de localisation d'équipements inactifs (*pagination*) sont employés pour réduire au maximum la signalisation et optimiser les performances de la gestion de la mobilité. Actuellement, Mobile IP supporte la mise à jour de localisation mais pas la pagination. Une caractéristique importante des protocoles de micro mobilité (cf. section 1.5.5) est leur capacité à réduire la signalisation liée aux migrations fréquentes des mobiles et de réduire la consommation des équipements en tenant compte du mode opérationnel de cet équipement (actif ou inactif). Le support d'une « connectivité passive » à l'Internet (par une localisation approximative) s'avère impératif puisqu'il permet de réduire la charge notamment sur les interfaces sans fil.

Dans la suite, nous détaillerons les opérations du processus de handover de niveau 3 afin d'identifier chaque étape du handover. Ensuite, nous présenterons les optimisations pour améliorer la détection du nouveau lien, l'anticipation de mouvements avec le Fast Handover, le modèle hiérarchique de la gestion de la mobilité et enfin les protocoles inspirés des réseaux cellulaires. Nous terminerons par une brève conclusion sur l'ensemble de ces optimisations.

1.5.1 Détection rapide de mouvements

La détection de mouvements est une phase essentielle du processus de handover. Bien entendu, elle va conditionner les performances du handover, et nous allons voir que de nombreuses méthodes existent avec leurs avantages et leurs défauts. La détection de mouvements consiste à déterminer si la configuration IPv6 du nœud

mobile est toujours valide par rapport au sous-réseau sur lequel il est connecté. Cette validation inclue d'une part la détermination que le routeur d'accès courant n'est plus accessible, et d'autre part la découverte d'un nouveau routeur d'accès. Le message permettant la découverte d'un nouveau routeur d'accès est le *Router Advertisement* (RA), qui comprend l'adresse du routeur d'accès et le préfixe utilisé sur le lien. Le fait que plusieurs routeurs d'accès puissent être présents dans un même sous-réseau complique la détection de mouvements ; le fait d'entendre un nouveau RA n'implique pas forcément le changement de son routeur d'accès par défaut.

Le protocole de découverte des voisins (*Neighbor Discovery*), défini dans le RFC 2461 [133], détermine l'émission des RA et des RS. Cette spécification fut standardisée en 1998 sans vraiment tenir compte de la mobilité des nœuds ; des constantes et délais trop longs furent alors définis pour éviter une congestion d'un réseau en cas de démarrage simultané de plusieurs équipements. Les spécifications recommandent d'envoyer les RA aléatoirement dans un intervalle de 200 à 600 secondes. Un nœud ne peut envoyer un RS qu'après avoir attendu trois fois le délai maximum d'émission du RA sans recevoir de RA. Le routeur d'accès choisit ensuite un délai aléatoire entre 0 et 500 ms avant de répondre à la sollicitation. Ce temps de réponse peut encore être décalé de 3 secondes, délai minimum entre deux émissions de RA consécutives. Ces chiffres apparaissent complètement démesurés par rapport à la mobilité d'un nœud. Ils impliquent une détection de mouvements de plusieurs minutes. En respectant pleinement la norme, il faudrait même attendre plus de trente minutes afin de pouvoir envoyer un RS pour être assuré que le routeur d'accès courant ne soit plus accessible.

Mobile IPv4 [199] a déjà proposé des alternatives pour la détection de mouvements. Deux méthodes, nommées *Lazy Cell Switching* et *Eager Cell Switching*, différencient le comportement d'un nœud mobile par rapport à la détection de mouvements. Dans *Lazy Cell Switching*, le nœud mobile attend simplement que l'annonce d'un routeur remplace celle de son routeur courant alors que dans *Eager Cell Switching* le nœud mobile va solliciter explicitement les agents de mobilité. Mobile IPv6 a également pris en compte l'importance de la détection de mouvements en proposant un intervalle d'émission des RA réduit à $[30ms; 70ms]$ et en ramenant le délai avant de pouvoir envoyer un RS à une seconde. Dans ce cas de figure, la détection de mouvements est beaucoup plus rapide, avec une moyenne de 25 ms pour découvrir le nouveau routeur d'accès. La plupart du temps, on considèrera que la découverte d'un nouveau routeur d'accès avec un nouveau préfixe implique un mouvement, sans tester si l'ancien routeur d'accès est encore joignable. Bien que la solution avancée par Mobile IPv6 accélère considérablement la détection de mouvements, elle requiert une forte utilisation de la bande pas-

sante : 20 RA sont envoyés en moyenne toutes les secondes, utilisant au minimum 14 kb/s (avec les RA les plus petits, c'est-à-dire sans option particulière). Bien que rapide, cette méthode semble difficilement adoptable, surtout sur des réseaux de type cellulaire pour des raisons d'utilisation de bande passante. Caceres et al. [38] ont montré l'impact de l'émission d'un message périodique sur la réception d'un flux TCP dans les réseaux sans fil (étude faite sur IPv4, mais qui représente bien le comportement générique). Lorsque la fréquence des messages passe en-dessous de 50 ms, le débit du flux TCP chute considérablement. Au-dessus de 50 ms, en revanche, le débit du flux TCP reste correct : le meilleur rapport fréquence/débit est évalué à 100 ms où le débit du flux observé est à 99% de son maximum.

Aussi, d'autres solutions ont vu le jour, utilisant des informations permettant d'anticiper la détection de mouvements. Cette anticipation, ou du moins la détection rapide d'un mouvement, est basée sur des indications reçues soit sur le nœud mobile, soit sur un élément du réseau (le point d'accès ou le routeur d'accès en général). Ces indications sont appelées déclencheurs de niveau 2, *L2 trigger* en anglais [86, 65, 66]. Un déclencheur de niveau 2 est une information spécifique remontée par la couche liaison aux couches supérieures, en l'occurence à la couche réseau. Des exemples de déclencheurs de niveau 2 utilisés sont la perte de lien avec un point d'accès ou l'établissement d'un lien avec un nouveau point d'accès. Il est important de noter ici qu'un déclencheur de niveau 2 n'est qu'une indication, et n'assure aucunement qu'un événement comme le déplacement entre sous-réseaux a eu lieu. Nous allons voir que plusieurs méthodes utilisent ces déclencheurs de niveau 2.

En ce qui concerne la mobilité IPv4, Fikouras et al. [65, 66] ont introduit les méthodes *Hinted Cell Switching* et *Fast Hinted Cell Switching*. Ces méthodes consistent à solliciter explicitement le nouvel agent de mobilité sur réception d'un déclencheur de niveau 2 indiquant la fin d'un handover de niveau 2. La méthode de Fast RA [87] est son équivalent pour l'IPv6, proposant l'émission immédiate d'un RS sur réception du déclencheur de niveau 2 indiquant que le point d'accès a changé (handover de niveau 2). Cette solution recommande également de supprimer le délai de réponse pour que le routeur d'accès puisse répondre immédiatement. Daley et al [50] avancent que cette solution reste sensible aux paquets perdus : le RS est envoyé en multicast, or la transmission radio d'une trame en multicast est moins fiable qu'une transmission unicast puisque les trames multicast ne sont pas acquittées. De plus, le RA en réponse est envoyé en unicast au nœud mobile, ce qui peut nécessiter des mécanismes supplémentaires de résolution d'adresse (découverte des voisins).

Une alternative au Fast RA est le *RA caching* [46]. Les points d'accès mettent

en tampon le dernier RA reçu et déclenchent son émission dès l'association d'un nœud mobile. Cette solution est très avantageuse puisqu'elle ne requiert aucune intervention de la part du nœud mobile qui reçoit directement le nouveau RA après son handover de niveau 2. Cependant, pour mettre en place ce mécanisme, les points d'accès, ponts de niveau 2, doivent être modifiés.

Daley et al [50] ont comparé ces différentes méthodes avec deux fréquences de handover différentes ; la fréquence la plus rapide permet aux routeurs d'accès de conserver leur tampon pour l'adressage du nœud mobile alors qu'avec la fréquence de handovers moins rapide, les routeurs d'accès doivent à chaque fois demander l'adresse du nœud mobile par un *Neighbor Solicitation*. Dans les deux situations, le temps de handover en respectant le RFC 2461 est de 234 ms (en considérant que la réception d'un nouveau RA implique un mouvement). Le RA caching prend 10 ms contre 16 ms pour le Fast RA pour une fréquence de handover rapide. Avec la fréquence de handover plus lente, le RA caching prend 74 ms contre 26 ms pour le Fast RA lorsqu'il n'y a pas de paquet perdu et 51 ms lorsqu'il y a des paquets perdus. Lorsqu'un paquet est perdu (RS du nœud mobile, *Neighbor Advertisement* du nœud mobile) le délai de handover augmente de 1 seconde, correspondant au délai de retransmission. La perte de paquets a été observée dans 4 tests sur 120. Ces temps rejoignent ceux obtenus par simulation par Fikouras et al. [65, 66] dans IPv4, où le handover avait été évalué à 42 ms avec l'utilisation des déclencheurs de niveau 2, et entre 30 et 500 ms pour différentes fréquences des annonces des agents de mobilité (équivalent de la solution Mobile IPv6).

1.5.2 Fast Handover

Afin de supporter localement des handovers rapides entre sous-réseaux, tout un ensemble de propositions se focalisent sur la réalisation du *Fast Handover* [89, 201, 107, 200, 99]. Le Fast Handover consiste en la réduction du temps de latence du handover et de la perte de paquets pendant un handover. Un certain nombre de choix dans la structure des protocoles influencent les performances du handover, comme l'entité de contrôle du handover, la mise en tampon et les techniques de transfert, le comportement radio, la détection et la prédiction de mouvements et l'accouplement et la synchronisation entre les couches IP et radio. L'objectif sous-jacent à la mise en place du Fast Handover est de combiner tous ces choix dans un même protocole.

L'idée de base du Fast Handover [201] est de permettre au nœud mobile d'obtenir sa nouvelle adresse temporaire avant d'effectuer le handover vers le nouveau sous-réseau afin qu'il puisse immédiatement communiquer lorsque la connexion

avec son nouveau routeur d'accès est établie. L'objectif est donc d'effectuer (du moins initier) un handover de niveau 3 avant que celui de niveau 2 ne soit terminé. De plus, cette technique permet la retransmission des paquets de l'ancien routeur d'accès vers le nouveau.

La création et l'attribution d'une nouvelle adresse temporaire avant que le nœud mobile ne se rattache au nouveau réseau nécessitent une anticipation de mouvements. Celle-ci peut être réalisée à l'aide des messages échangés dans la couche liaison ou grâce à des informations de niveau 2 (mesure de l'intensité de signal).

Les optimisations sont mises en œuvre grâce à des interactions entre le routeur d'accès courant du nœud mobile, le nœud mobile et le routeur d'accès cible. Deux scénarii sont possibles : soit le réseau contrôle le handover en décidant du nouveau point d'attachement pour le mobile, soit le nœud mobile contrôle le handover et déclenche les opérations de fast handover, en indiquant l'identification du réseau cible à son routeur d'accès courant.

Handover contrôlé par le réseau

Lorsque les handovers sont contrôlés par le réseau (voir figure 1.7), le routeur d'accès courant reçoit l'indication que le nœud mobile est sur le point d'effectuer un handover. De même, l'identité du nouveau routeur d'accès sur lequel le nœud mobile va se rattacher lui est envoyée. Ces informations peuvent être transmises par une entité spécifique dans le réseau chargé de contrôler la mobilité des équipements. La façon dont ces informations sont transmises au routeur d'accès courant ne fait pas partie des spécifications du Fast Handover. Lorsque le routeur d'accès courant sait que le nœud mobile doit se rattacher à un nouveau routeur d'accès, il crée une nouvelle adresse temporaire pour le nœud à l'aide de la méthode d'auto-configuration d'adresse sans état [176]. Cette adresse est ensuite envoyée, en plus de l'adresse IP du nouveau routeur d'accès, au nœud mobile, à l'aide du message *Proxy Router Advertisement*.

Dans le même temps, le routeur d'accès courant envoie un message *HI* (Handover Initiate) vers le nouveau routeur d'accès pour lui indiquer l'ancienne et la nouvelle adresse temporaire du nœud mobile. Le nouveau routeur d'accès commence par vérifier si la nouvelle adresse temporaire pour le nœud mobile est valide. Si tel est le cas, alors il ajoute cette adresse dans son cache des voisins et répond par un message *HAck* (Handover Acknowledgement). Par contre, si l'adresse n'est pas valide, le nouveau routeur d'accès ajoute dans le message *HAck* qu'il accepte le

handover, mais que l'adresse n'est pas valide. Le nœud mobile devra alors effectuer un handover "classique".

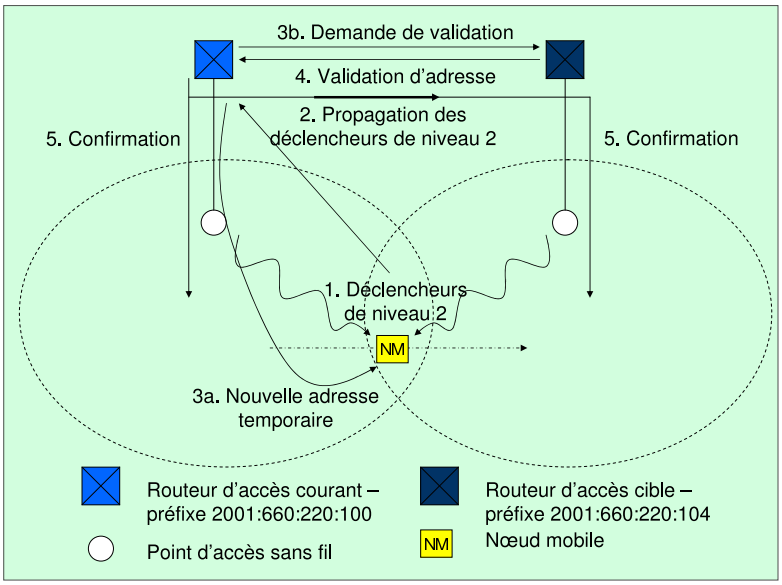


FIG. 1.7 – FMIP - Handover contrôlé par le réseau

Handover contrôlé par le mobile

Dans le cas du handover contrôlé par le nœud mobile, ce dernier envoie un *Router Solicitation For Proxy* à son routeur d'accès courant lorsqu'il détecte qu'un handover va avoir lieu. Encore une fois, la manière dont le nœud mobile découvre qu'il est sur le point de faire un handover n'est pas détaillée dans la spécification du Fast Handover. Cependant, l'utilisation des déclencheurs de niveau 2, déjà citée dans la section précédente et détaillée dans la section 6.3.1, semble être une possibilité pour savoir qu'un handover est imminent. Dans ce message, le nœud mobile doit inclure des informations qui permettront d'identifier le nouveau routeur d'accès. A la réception de ce message, le routeur d'accès courant répond par un message contenant une nouvelle adresse temporaire valide pour le réseau cible. L'échange des messages *HI* et *HAck* se déroule comme dans le cas des handovers contrôlés par le réseau.

Ensuite, le nœud mobile doit enregistrer sa nouvelle adresse temporaire auprès de son agent mère. Pour ce faire, le nœud mobile envoie un *Fast Binding Update*

à son ancien routeur d'accès juste avant d'effectuer le handover. A la réception de ce message, le routeur d'accès courant envoie un *Fast Binding Acknowledgement* (F-BACK) au nœud mobile pour indiquer que la mise à jour est effectuée. De plus, le routeur d'accès courant déclenche le transfert des paquets vers le nouveau routeur d'accès, c'est-à-dire que tous les paquets qui arrivent sur le routeur d'accès à destination du nœud mobile sont envoyés directement au nouveau routeur d'accès. Ce dernier les stocke dans un tampon en attendant de pouvoir les envoyer au nœud mobile lorsqu'il sera connecté.

Lorsque le nœud mobile se connecte au nouveau routeur d'accès, il lui envoie un *Fast Neighbor Advertisement* (F-NA) pour déclencher la retransmission des éventuels paquets en attente sur le routeur d'accès. Enfin, après le déplacement du nœud mobile, ce dernier doit envoyer un *Binding Update* (BU) à son agent mère et à ses correspondants à travers le nouveau routeur d'accès pour enregistrer sa nouvelle adresse temporaire.

Une autre méthode de configuration est possible, appelée auto-configuration à état. Dans ce cas, l'ancien routeur d'accès doit envoyer un message *HI* au routeur cible avant d'envoyer la nouvelle adresse temporaire au nœud mobile. Dans ce cas de figure, ce message est utilisé pour demander une nouvelle adresse temporaire plutôt que pour en valider une. Le message *HAck* contiendra une adresse valide qui pourra être transmise au nœud mobile.

Evaluation du Fast Handover

Des simulations dans le simulateur réseau NS-2 [143] ont montré que le Fast Handover est plus rapide que Mobile IPv6 lorsque moins de trente nœuds mobiles se déplacent simultanément [178]. Alors qu'un handover avec Mobile IPv6 passe de 45 millisecondes pour un seul nœud mobile à 1 seconde pour 30 nœuds mobiles, le handover avec la solution de Fast Handover reste autour de 20 millisecondes. Cependant, lorsque plus de trente nœuds mobiles se déplacent simultanément, le temps de handover avec la solution de Fast Handover augmente parallèlement avec le temps de handover de Mobile IPv6, atteignant jusqu'à 3 secondes pour 50 nœuds mobiles. Une analyse de la bande passante a montré que la baisse du débit disponible pour les stations implique la baisse des performances du Fast Handover pour plus de trente nœuds mobiles. Cette dégradation est due aux flux réceptionnés par les nœuds mobiles. Les messages de signalisation utilisant le même canal que les données, la baisse de la bande passante a un impact important sur les temps de handover.

Koodli et Perkins [89] ont également évalué le Fast Handover, avec le transfert de contexte, sur une plate-forme de taille limitée FreeBSD (mouvements entre deux routeurs connectés l'un à l'autre). Cette étude a montré que 90% de la valeur du temps de handover (mesurée autour de 80 millisecondes) est dû au handover de niveau 2 (mesuré à 66 millisecondes). Nous pouvons noter ici que les temps mesurés sur cette plate-forme sont cohérents avec ceux obtenus par simulation puisque le handover de niveau 2 n'est pas implémenté dans NS-2 (voir chapitre 4). Une première constatation importante, que confirme les simulations décrites ci-dessus, est que le nombre de messages utilisés ne ralentit pas considérablement le handover. Les auteurs ont également évalué les temps du transfert de contexte à posteriori, i.e une fois que le nœud mobile a fini le handover. Le temps de transfert a été mesuré à 1,1 millisecondes. Ce temps relativement petit est notamment dû au temps d'aller-retour très court entre les deux routeurs d'accès qui sont directement relié par un câble. Ce résultat nous montre clairement que c'est le temps d'aller-retour entre les deux routeurs qui déterminera la rapidité du transfert de contexte. Bien que ces mesures n'aient été faites que sur une plate-forme de taille limitée (un nœud mobile, temps d'aller-retour entre les équipements très courts), les résultats sont encourageants pour le développement du Fast Handover avec Transfert de Contexte.

1.5.3 Bi-casting

Le *Bi-casting* [101], ou plus généralement le *n-casting* est la duplication du même trafic destiné au nœud mobile à plusieurs localisations. Cette méthode est utilisée pour réduire le nombre de paquets perdus pendant un handover. Elle peut être utilisée en plus du protocole de Fast Handover décrit ci-dessus pour envoyer des copies du trafic aux localisations potentielles du nœud mobile. De plus, le Bi-casting est une bonne solution contre l'effet "ping-pong" ; quand un nœud mobile se déplace entre deux routeurs d'accès plusieurs fois et fréquemment, Mobile IP requiert que le nœud mobile crée une nouvelle adresse temporaire et l'enregistre à chaque déplacement. Le Bi-casting permet au nœud mobile de s'enregistrer dans les deux sous-réseaux simultanément.

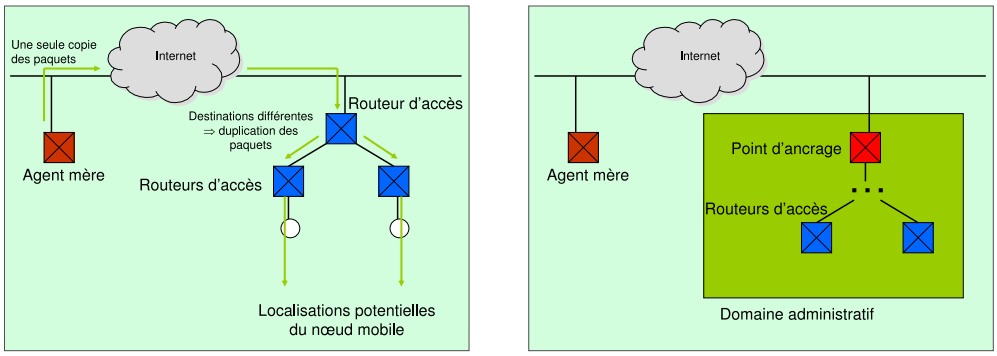
Pour mettre en œuvre le Bi-casting, le nœud mobile doit associer plus d'une adresse temporaire à son adresse principale. Le nœud mobile devra alors ajouter un bit spécifique dans son message d'enregistrement (*Binding Update*) indiquant que l'adresse annoncée ne doit pas remplacer la précédente, mais plutôt être considérée comme une adresse de localisation temporaire supplémentaire. Quand l'agent mère ou un correspondant reçoit ce type de message, il ajoute la nouvelle adresse temporaire sans écraser l'ancienne dans le cache d'association.

Aucun nouveau message n'est donc nécessaire puisque toutes les requêtes et réponses liées au Bi-casting peuvent être mises dans les messages déjà utilisés dans Mobile IP. Si le nœud mobile sollicite le Bi-casting auprès d'un correspondant qui ne supporte pas les associations simultanées, le correspondant ignore l'option et le nœud mobile réalise un handover comme spécifié dans Mobile IP.

Lorsque le nœud mobile est sur le point de faire un handover, il peut demander le Bi-casting à son agent mère. Encore une fois, la détection d'un handover imminent n'est pas définie, mais les mêmes mécanismes que ceux proposés dans le Fast Handover peuvent être utilisés, à savoir une entité dans le réseau qui contrôle le déplacement des nœuds mobiles et/ou l'utilisation des déclencheurs de niveau 2. La demande de Bi-casting peut également se faire dans le message *Fast-Binding Update* du Fast Handover (voir section précédente). Si l'agent mère accepte la requête, il enregistre deux associations pour ce mobile et enverra désormais l'ensemble du trafic aux deux adresses temporaires du nœud mobile. Les deux routeurs d'accès propagent ensuite le trafic dans leur sous-réseau respectif.

Bien que le Bi-casting semble être efficace et réduise sensiblement le nombre de paquets perdus, la manière de le mettre en œuvre est gourmande en terme de ressources utilisées ; effectivement, l'ensemble du trafic pour le nœud mobile est dupliqué sur tout le chemin allant de l'agent mère au nœud mobile alors que les paquets prennent exactement le même chemin, excepté les quelques derniers sauts. Pour remédier à ce problème, on peut penser à envoyer les paquets en multicast [159, 19]. Cependant, la gestion de groupe doit être légère, dynamique et l'ajout ou le retrait d'une adresse destinatrice doit se faire rapidement en fonction des déplacements du nœud mobile. La technique du *Small Group Multicast* [194] répond parfaitement à ces besoins. Small Group Multicast est basé sur le multicast explicite : les datagrammes multicast sont routés d'après les informations de routage unicast et chaque datagramme contient la liste des adresses destination. A chaque saut, le routeur contrôle si pour chaque destination le datagramme peut prendre la même direction. Si une adresse indique que le datagramme doit prendre une autre direction que les autres, le routeur duplique le paquet. Sinon il transmet simplement le paquet sans aucune modification (voir figure 1.8(a)).

D'autres méthodes pour réduire la charge du réseau tout en faisant du Bi-casting existent. L'une d'entre elles est de réaliser le Bi-casting à partir d'un autre point du réseau. Le tunnel mis en place entre l'ancien et le nouveau routeur d'accès par la procédure de Fast Handover (voir section 1.5.2) peut être utilisé pour faire du Bi-casting. Grâce à l'introduction d'une hiérarchie dans le modèle de Mobile IPv6, nous allons voir que d'autres entités encore peuvent réaliser la duplication du trafic pour le nœud mobile à un endroit plus proche de ce dernier.



(a) Xcast : optmisation pour réaliser du Bi-casting

(b) Modèle hiérarchique : la mobilité dans le domaine est cachée

FIG. 1.8 – Modèle hiérarchique et méthode de Bi-casting

1.5.4 Architecture hiérarchique

L'objectif de la mise en place d'une hiérarchie pour la gestion de la mobilité est de cacher certains mouvements des nœuds mobiles à des correspondants. Si un nœud mobile se déplace à l'intérieur d'un domaine, il lui incombe uniquement de faire un enregistrement régional, sans faire part de ce déplacement aux nœuds extérieurs à ce domaine. Par contre, lorsque le nœud mobile change de domaine, il lui faudra faire un enregistrement global. Un domaine est défini comme étant une aire de mobilité locale [104]. Généralement un domaine est indépendant des sous-réseaux et sa taille est choisie par l'opérateur réseau. Généralement un domaine dépend d'une même autorité administrative, c'est pourquoi l'entrée dans un domaine peut nécessiter des mécanismes d'autorisation / d'authentification supplémentaires.

La mise en place d'une hiérarchie permet de minimiser le trafic entre l'agent mère et le nœud mobile, sans pour autant introduire de signalisation supplémentaire entre le nœud mobile et les routeurs d'accès. Un modèle hiérarchique offre également des délais inférieurs en ce qui concerne la mise à jour d'adresse temporaire. Comme nous allons le voir, ce modèle est bien adapté pour réaliser le Fast Handover et le Bi-casting.

Plusieurs modèles hiérarchiques ont été définis dans la littérature [38, 33, 52]. La référence la plus importante reste la spécification de *Mobile IPv6 Hiérarchique* à l'IETF [168] avec l'implémentation la plus utilisée provenant de l'université de Monash [136]. La spécification de Mobile IPv6 hiérarchique provient initialement

des articles de recherche de Castellucia [42, 43]. Dans toutes ces solutions de gestion de mobilité hiérarchique, un nouveau type de nœud est introduit. Il s'agit d'un point d'ancrage, généralement un routeur d'accès, au sommet de plusieurs autres routeurs d'accès, qui sera en charge de cacher les mouvements des nœuds mobiles à l'intérieur de son domaine (voir la figure 1.8(b)).

Mobile IPv6 Hiérarchique

Chaque point d'ancrage d'un domaine est annoncé dans les RA envoyés par les routeurs d'accès. En plus de l'adresse du point d'ancrage, le préfixe du domaine rattaché au point d'ancrage, la distance au nœud mobile ainsi que les préférences du point d'ancrage sont inclus dans les RA. Lorsqu'un nœud mobile entre pour la première fois dans un domaine, il doit s'enregistrer avec son agent mère en indiquant l'adresse du point d'ancrage comme adresse temporaire. Ensuite lorsque le nœud mobile se déplace à l'intérieur du domaine (comme celui de la figure 1.8(b) par exemple), il aura uniquement besoin de faire des enregistrements locaux.

La mobilité hiérarchique de Mobile IPv6 [168, 42, 43] propose deux modes d'enregistrement pour le nœud mobile, à savoir un mode basique et un mode étendu. Ces deux modes diffèrent dans le nombre d'adresses utilisées par le nœud mobile. Dans le mode basique, le nœud mobile a deux adresses : une adresse temporaire régionale basée sur le préfixe du point d'ancrage et une adresse temporaire locale. Dans ce schéma, le point d'ancrage agit comme un agent mère. Il intercepte les paquets à destination de l'adresse temporaire régionale et les retransmet à l'adresse temporaire locale correspondante. Ces opérations sont totalement transparentes à de l'agent mère qui n'a besoin d'aucune modification.

Cependant, dans un souci de mise à l'échelle ou pour des raisons de simplification de gestion d'adressage, tous les nœuds mobiles ne peuvent pas acquérir leur propre adresse régionale. Dans le mode étendu, l'adresse temporaire régionale est celle du point d'ancrage. Le point d'ancrage maintient une table d'associations entre l'adresse principale des nœuds mobiles et leur adresse locale. Quand le point d'ancrage reçoit des paquets à destination d'un nœud mobile, il doit les décapsuler et les ré-encapsuler à destination de l'adresse locale. Ceci implique que chaque paquet doit contenir l'adresse principale du nœud mobile. Le mode étendu supporte aussi bien les nœuds mobiles que les réseaux mobiles.

Dans tous les cas de figure, le nœud mobile doit faire un enregistrement auprès de son agent mère quand il change de domaine. Ce changement de domaine est détecté par le nœud mobile dans les annonces des routeurs d'accès. Effectivement,

chaque routeur d'accès à l'intérieur d'un domaine annonce l'adresse IP du point d'ancrage. Le nœud mobile compare l'adresse annoncée avec celle de son point d'ancrage courant. Si elles diffèrent, il considère qu'il a changé de domaine et doit donc avertir les nœuds extérieurs au domaine.

Toute cette procédure permet tout de même à un nœud mobile n'implémentant pas la solution hiérarchique d'utiliser Mobile IPv6. De manière symétrique, un nœud mobile qui demande à faire un enregistrement local à un routeur d'accès n'implémentant pas la solution hiérarchique de Mobile IPv6, aura la possibilité de faire une enregistrement "classique".

Bi-casting dans une hiérarchie

Comme il a été montré dans la section 1.5.3, le Bi-casting à partir de l'agent mère peut difficilement être mis à l'échelle et peut générer une congestion dans le réseau s'il est appliqué à des millions de nœuds mobiles. Un modèle hiérarchique permet de faire le Bi-casting à partir du point d'ancrage d'un domaine [52, 33, 101]. Quand un nœud mobile se déplace à l'intérieur d'un domaine, il peut demander le Bi-casting dans ses messages d'enregistrement régional. Cette demande est transmise au point d'ancrage qui ajoute une entrée pour le nœud mobile (association simultanée). Ensuite le point d'ancrage propage le trafic à l'ancienne et à la nouvelle localisation du nœud mobile. La duplication du trafic à plusieurs localisations du nœud mobile peut même être faite avant le mouvement effectif du nœud mobile si une anticipation peut être mise en place [52, 33].

Cette solution a l'avantage d'être indépendante de Mobile IPv6 : le Bi-casting peut être mis en place à l'intérieur du domaine sans interaction avec des nœuds à l'extérieur du domaine. Cette caractéristique est d'autant plus forte qu'elle ne s'applique pas uniquement à la mise en place du Bi-casting. Avec la mise en place d'un modèle hiérarchique, la gestion de la mobilité à l'intérieur d'un domaine peut être indépendante au niveau protocolaire de la gestion de la mobilité globale, i.e entre domaines. Ainsi, un protocole de gestion de la mobilité différent de Mobile IP, adapté au domaine dont il a la charge, peut être mis en place. En assurant une compatibilité avec Mobile IP, la mise en place de tels protocoles semble réaliste. La dernière section présente une solution de gestion de micro mobilité.

1.5.5 Protocole de micro mobilité

Typiquement, les nœuds connectés à l'Internet (par exemple, des ordinateurs de bureau connectés à un réseau local) restent en ligne pendant plusieurs heures bien que la plupart du temps ils ne communiquent pas. Être "toujours connecté" de cette manière aboutit à être accessible et à pouvoir accéder aux ressources Internet à tout instant. Les utilisateurs mobiles connectés à l'Internet sans fil s'attendront à un service semblable. Malheureusement, le maintien de l'information de localisation des nœuds mobiles, exige des mises à jour fréquentes, ce qui consomment de la bande passante précieuse et l'énergie électrique des nœuds mobiles. Ce surplus de signalisation et cette consommation électrique peuvent être réduits par l'introduction de la *pagination*. Généralement, les nœuds mobiles fonctionnent sur des batteries de faible autonomie, c'est pourquoi il s'avère impératif d'éviter que les nœuds mobiles inoccupés aient à transmettre des messages de mise à jour de localisation fréquemment. Cela exige l'appui explicite de protocoles réseau, comme la capacité de suivre la localisation des mobiles de manière approximative et la capacité de retrouver des nœuds mobiles inactifs. Des nœuds mobiles inactifs ne doivent pas enregistrer leur déplacement dans la même aire de pagination pour ne signaler que leur changement d'aire de pagination. La pagination a été mise en œuvre dans un certain nombre de protocoles de micro mobilité dont Cellular IP [41, 183, 40, 163] et Hawaii [153]. Cellular IP est présenté ci-dessous.

Cellular IP

Le protocole Cellular IP [41, 183, 40, 163] utilise une architecture hiérarchique ; l'Internet global est divisé en domaines constituant des aires de micro mobilité. Chaque niveau de mobilité est traité par un protocole différent, adapté aux besoins du niveau. Cellular IP, inspiré des systèmes cellulaires, se propose de résoudre la gestion de la mobilité à l'intérieur d'un domaine. Ce protocole a été conçu pour répondre rapidement à un grand nombre de nœuds mobiles qui migrent fréquemment. Par contre, Mobile IP gère la mobilité entre domaines (macro mobilité).

Dans une aire de micro mobilité, l'information de routage est totalement distribuée, c'est-à-dire qu'aucun des nœuds ne garde une information globale sur la topologie du réseau. Ceci rend la solution robuste. De plus, Cellular IP peut être utilisé dans un domaine allant d'un bureau à un réseau s'étendant sur une ville et peut supporter un grand nombre d'hôtes. Le but de Cellular IP est de procurer un handover ne générant aucune coupure dans les communications, à l'intérieur d'un domaine, et de cacher les mouvements des nœuds mobiles au reste de l'Internet.

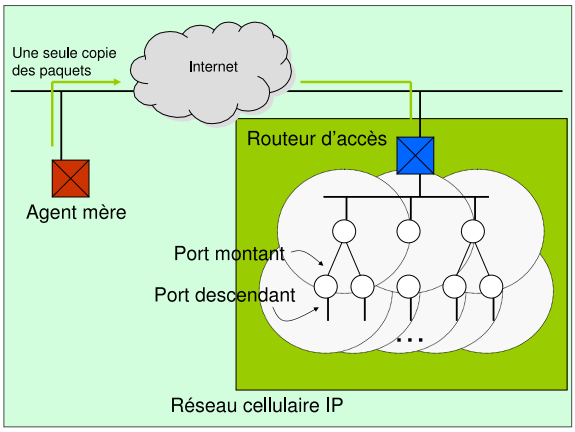


FIG. 1.9 – Architecture Cellular IP

Cellular IP fonctionne avec une hiérarchie composée d'un routeur passerelle et de nœuds cellular IP, qui sont des points d'accès et des nœuds mobiles implémentant le protocole Cellular IP (voir figure 1.9). Le routeur passerelle assure la liaison entre le réseau cellular IP et le reste de l'Internet. Il filtre, contrôle et propage les paquets en provenance et à destination du réseau cellular IP. Les points d'accès sont connectés ensemble par un réseau filaire et ont une interface sans fil pour communiquer avec les nœuds mobiles. Bien que cette structure soit bien adaptée pour le Fast Handover, des traitements spécifiques pour rendre le handover rapide ont été mis en place tels que le semi-soft handover [41, 40] ou le indirect semi-soft handover [163], qui seront détaillés plus loin.

La gestion de la localisation est très différente de celle de Mobile IP. L'adresse IP n'est plus utilisée pour localiser un équipement mais uniquement pour l'identifier. De ce fait, il n'est plus nécessaire de faire des encapsulations ou des conversions d'adresses. La localisation des nœuds mobiles est contenue dans deux différents caches dans les points d'accès. L'utilisation de deux caches permet de différencier le traitement des nœuds mobiles actifs des inactifs. Un nœud est considéré comme étant actif quand il échange des paquets de données avec au moins un correspondant. Le cache de pagination est utilisé pour localiser les nœuds mobiles inactifs et le cache de routage pour la localisation des nœuds actifs. Ces caches stockent des associations entre l'adresse IP du nœud mobile et l'interface du point d'accès utilisée pour atteindre ce nœud. Ces associations sont construites par le chemin inverse des paquets envoyés par le nœud mobile (aussi bien les paquets de données que ceux de contrôle). Le routage se fait donc de saut-en-saut sur le plus court chemin.

La localisation des nœuds mobiles inactifs est approximative : les points d'accès

sont organisés en aires de pagination et cherchent uniquement à savoir dans quelle aire se situe le nœud mobile. La gestion de la localisation est décrite plus tard. Dans un réseau cellular IP donné, chaque aire de pagination a un identificateur unique. Cet identificateur est annoncé par tous les points d'accès mais tous les points d'accès n'ont pas de cache de pagination.

Dans ce schéma, un nœud mobile n'a pas de point d'attache dédié, il utilise le meilleur à tout moment. Il n'y a donc pas d'authentification entre les points d'accès. Cependant, un contrôle est effectué lors de l'entrée des nœuds mobiles dans le réseau cellular IP et seuls les messages de contrôle peuvent créer de nouvelles entrées dans les caches pour assurer une sécurité.

Le protocole Cellular IP

Trois messages de contrôle sont nécessaires pour le fonctionnement du protocole : *Route Update, Paging-Update et Paging-Teardown.* L'utilisation de ces messages est décrite au fil de ce paragraphe. Le routeur passerelle envoie périodiquement des messages aux points d'accès pour leur indiquer leur port montant (*uplink port*), c'est-à-dire le port utilisé pour atteindre le routeur passerelle (voir figure 1.9). Le protocole est alors dit "plug-and-play" puisqu'aucune configuration préalable n'est requise sur les points d'accès. Tous les autres ports des points d'accès sont des ports descendants (voir figure 1.9). Les points d'accès envoient aussi des messages périodiques contenant entre autres l'identificateur du réseau cellular IP, l'adresse IP du routeur passerelle et un identificateur d'aire de pagination sur leur interface radio.

Quand un nœud mobile entre pour la première fois dans un réseau cellular IP, il envoie une authentification et des informations utilisateur dans un Paging-Update à destination du routeur passerelle. Si le routeur passerelle accepte la demande du nœud mobile, celui-ci doit faire un enregistrement auprès de l'agent mère de Mobile IP.

Lorsque le nœud mobile est inactif, il doit mettre à jour les caches de pagination à chacune de ses entrées dans une nouvelle aire de pagination ou juste avant que son entrée n'expire (cas où le nœud mobile ne s'est pas déplacé). Cette mise à jour est assurée par l'émission d'un Paging-Update. Un Paging-Update détruit aussi les entrées dans les caches de routage. Un nœud mobile peut par ailleurs demander explicitement la destruction de son entrée dans un cache de pagination pour éviter tout conflit. La suppression explicite d'une entrée dans le cache de pagination est faite par l'envoie d'un Paging-Teardown.

Dans son état actif, un nœud mobile envoie et/ou reçoit des paquets de données. Les paquets de données envoyés par le nœud mobile mettent à jour les caches de routage sans que le mobile n'aie besoin d'envoyer d'autres paquets de contrôle. Par contre, quand un nœud mobile ne fait que recevoir des données, il lui faut envoyer périodiquement un Route-Update pour éviter que son entrée dans les caches de routage n'expire. Ceci dit, que le nœud mobile soit en émission ou en réception, il doit envoyer un Route-Update chaque fois qu'il change de point d'accès. Ce paquet crée une nouvelle entrée dans tous les nouveaux points d'accès sur le chemin allant du nœud mobile au routeur passerelle. Les opérations spécifiques au handover sont détaillées dans la sous-section suivante.

Quand un flux de données arrive pour un nœud mobile inactif, le premier paquet est utilisé pour établir un chemin jusqu'au nœud mobile (pagination implicite). Le paquet est transmis suivant les informations contenues dans les caches de pagination. Si jamais un point d'accès reçoit un paquet pour un nœud mobile dont elle n'a aucune entrée (pas de cache de pagination), elle duplique le paquet sur tous ses ports descendants. Quand le nœud mobile reçoit ce premier paquet, il envoie un Route-Update pour créer une entrée dans les caches de routage et devient actif.

Le traitement du handover dans Cellular IP

Le changement de point d'accès, appelé *hard handover*, est automatiquement géré par le protocole. Cependant, des paquets peuvent être perdus pendant le temps que le Route-Update atteigne le point d'accès qui doit réaliser le changement de route. Quand un nœud mobile peut interagir simultanément avec deux points d'accès, il peut faire un *semi-soft handover*. Quand un nœud mobile décide de se déplacer sur un nouveau point d'accès, il envoie un Route-Update avec un champ spécifique à travers le nouveau point d'accès et retourne écouter l'ancien. Quand le Route-Update atteint le premier point d'accès qui avait déjà une entée pour le nœud, une nouvelle entrée est créée (au lieu d'écraser l'ancienne). Les paquets de données sont alors envoyés à l'ancienne et à la nouvelle localisation du nœud mobile. Quand le nœud mobile décide par la suite de se rattacher au nouveau point d'accès, il envoie un Route-Update pour détruire l'entrée avec l'ancien point d'accès.

Pour les nœuds mobiles ne pouvant pas être connectés simultanément à deux points d'accès, la technique de *Indirect semi-soft handover* se rapproche du semi-soft handover. Quand le nœud mobile décide de changer de point d'accès, il envoie un Route-Update avec un champ I à travers son ancien point d'accès avec l'adresse

du nouveau point d'accès dans le champ de l'adresse destination. Ce paquet est transmis au routeur passerelle qui l'envoie au nouveau point d'accès. Sur réception de ce paquet, le nouveau point d'accès envoie un Route-Update avec l'adresse du nœud mobile comme adresse source et on se retrouve alors dans la même situation que dans le semi-soft handover.

1.6 Conclusion

Dans ce chapitre, un ensemble de protocoles dédiés à la gestion de la mobilité des nœuds dans l'Internet Nouvelle Génération a été présenté. Tous ces protocoles se focalisent sur la réduction des effets néfastes (nombre de paquets perdus, temps d'interruption) observés sur les flux que reçoit ou envoie un nœud mobile, ainsi que sur la minimisation du temps de mise à jour de localisation lors de déplacements. Cet ensemble de protocoles peut être divisé en deux grandes parties : la gestion des handovers horizontaux et la gestion des handovers verticaux.

D'une part, nous avons vu que la gestion des handovers horizontaux pouvait se faire à différents niveaux de la pile TCP/IP. Cependant, l'ensemble de ces protocoles repose sur le même principe : un nœud mobile actif doit constamment mettre à jour sa localisation et les communications en cours devront pouvoir perdurer suite à un déplacement, même si une adresse change au cours du temps. La solution la plus populaire est Mobile IPv6 [84], qui propose une extension de la pile IPv6 au sein des nœuds mobiles (et des nœuds correspondants si l'optimisation de routage est utilisée). Lorsque le nœud mobile est connecté dans un réseau visité, un agent mère se charge de relayer les paquets à destination du nœud mobile. La mobilité est cachée aux applications par une manipulation d'en-têtes dans les paquets IPv6. D'autres solutions comme LIN6 [175, 91] proposent de scinder la couche réseau en une partie logique, utilisée par les applications, et une partie physique, qui localise les équipements. Cette solution est d'autant plus intéressante qu'elle favorise l'utilisation d'interfaces multiples : un seul identificateur logique peut être utilisé par plusieurs interfaces. Au niveau de la couche transport, deux familles de protocoles existent. On retrouve les protocoles de gestion bout à bout qui ne requièrent aucune participation d'un élément du réseau à l'opposé de solutions qui revendiquent le découpage des connexions des nœuds mobiles en deux sessions, une sur le réseau sans fil et une autre sur le réseau filaire. Ce type de solutions tend à réduire l'impact des mouvements sur les connexions TCP. Enfin, SIP [161, 162] est une solution de niveau applicatif, au départ définie comme un protocole d'établissement de session, qui permet la mobilité des nœuds avant et pendant un appel.

D'autre part, plusieurs modèles et extensions ont été définis pour prendre en compte la gestion d'interfaces multiples sur un terminal mobile. Certaines solutions reposent sur des entités supplémentaires ajoutées dans le réseau [191, 190], alors que d'autres définissent des nouvelles extensions aux messages de Mobile IPv6 pour permettre une sélection d'interfaces lors des mises à jour de localisation [186, 188, 67, 198]. Comme cité plus haut, LIN6 [106, 173] décrit également comment gérer plusieurs interfaces grâce à la séparation de la couche réseau en deux sous-couches logiques et physiques. Enfin, d'autres auteurs encore [100, 59, 95, 160] se focalisent plutôt sur l'union des deux mondes de communication jusqu'à présent disjoints, à savoir la téléphonie cellulaire et l'Internet sans fil. L'évolution des réseaux vers la 4G passera par un couplage entre les réseaux cellulaires et les WLAN existants, avec un degré d'intégration plus ou moins fort.

Finalement, nous avons étudié les solutions principales d'optimisation de handover, généralement horizontal. La première optimisation concerne la détection de liens. Cette étape est en effet délicate car elle doit à la fois résister au facteur d'échelle et être extrêmement rapide. Dans une certaine mesure, elle va conditionner le temps total d'interruption des communications lors de mouvements entre sous-réseaux. On pourra retenir deux méthodes principales : l'utilisation d'une fréquence très élevée de RA [84] et l'utilisation de déclencheurs de niveau 2 [86], aussi bien au niveau du nœud mobile (émission d'un RS), qu'au niveau du point d'accès [46, 87]. Plus généralement, nous avons vu que le Fast Handover [89, 201], qui permet de préparer la configuration sur le nouveau routeur d'accès préalablement au déplacement, permet de diviser par trois le temps de handover par rapport à Mobile IPv6 [178]. D'autres propositions visent à réduire l'impact d'un handover sur les flux par la mise en place de la duplication du trafic aux différentes localisations potentielles du nœud mobile (Bi-casting [101]), ou à réduire le temps d'enregistrement de Mobile IPv6. Cette dernière catégorie de propositions suggère la division de l'Internet en domaines, à l'intérieur desquels la moblité des nœuds est cachée aux nœuds extérieurs au domaine [42, 168, 38, 33, 52]. Le protocole de gestion de la mobilité des nœuds à l'intérieur des domaines pourra même être différent que le protocole de gestion inter-domaines. On pense par exemple à Cellular IP [41, 183, 40, 163] comme protocole de micro mobilité.

Une première conclusion que nous pouvons tirer de toutes ces propositions est tout d'abord que la gestion de la mobilité dans l'Internet n'est pas encore finalisée. Pour le moment, c'est la réunion de plusieurs optimisations qui permet d'obtenir de bonnes performances, en terme de temps d'interruption et de paquets perdus. En ce qui concerne la gestion de handover horizontal, les solutions de niveau transport et de niveau applicatif ont un avantage important : à partir du moment où ces différents niveaux sont informés qu'un handover est en cours,

ils peuvent s'adapter rapidement (adaptation de la fenêtre TCP, changement de l'encodage au niveau applicatif). Cependant, la détection de handover n'est pas aussi évidente qu'au niveau de la couche réseau. Ces solutions nécessitent une interaction avec les couches inférieures pour être informées des mouvements du nœud mobile. De la même manière, on peut très bien mettre en place un échange entre Mobile IPv6 et les couches supérieures afin de permettre un ajustement des connexions existantes. Par ces réflexions on se rend compte que la clé de la gestion de la mobilité n'est peut-être pas de déterminer dans quelle couche gérer la mobilité, mais plutôt comment mettre au point une interaction optimale entre les différentes couches du modèle TCP/IP. Cet échange d'informations entre les couches du modèle TCP/IP, est au cœur de ce travail de thèse. Nous verrons dans la suite comment réaliser cet échange d'informations, et comment ce dernier permet d'optimiser la gestion de la mobilité.

Une deuxième remarque importante concerne la gestion de la mobilité verticale ; bien que de nombreuses solutions existent, peu d'entre elles considèrent un large étendu de technologies (il est rare de considérer plus de deux interfaces réseau sur le terminal) et surtout peu d'entre elles cherchent à optimiser les différentes opérations. La recherche sur la gestion d'interfaces multiples, ou le "multihoming" plus généralement, en est plutôt au stade d'élaboration de solutions, de première prise en compte du problème. C'est pourquoi, il est important de dépasser ce stade en proposant non seulement une prise en compte générale de la problématique d'un nœud mobile à interfaces multiples, mais également en gardant à l'esprit une recherche d'efficacité.

Afin de comprendre la problématique de la gestion de la mobilité verticale, nous étudierons dans le chapitre suivant trois technologies de communication sans fil. Cette étude nous permettra de réaliser quels sont les enjeux de l'Internet sans fil et quelles informations seront utilisables par les couches supérieures pour optimiser la mobilité IP. La suite du mémoire sera alors consacrée à l'évaluation de Mobile IPv6 et ses principales optimisations, la présentation d'un nouvel outil de simulation, puis à la présentation d'une solution d'optimisation de handovers horizontaux dans les réseaux IEEE 802.11. Enfin nous présenterons la conception et l'évaluation d'une nouvelle architecture pour le terminal qui a pour objectif d'améliorer la connectivité globale d'un nœud mobile équipé d'interfaces multiples.

Chapitre 2

Intérêts et limitations des standards : normes des technologies de communication

2.1 Introduction

Le domaine des radio-communications est en pleine expansion. A l'image de l'essor de la téléphonie mobile sans fil, nous pouvons envisager un fort engouement pour les futures technologies sans fil permettant un accès à l'Internet. Le déploiement d'un réseau sans fil est moins coûteux que la mise en place de câbles et surtout permet une facilité d'utilisation appréciable par l'utilisateur final. Dès lors, il devient aisé de déployer des réseaux publics, par exemple dans des gares ou des cafés, permettant à des utilisateurs de se connecter à l'Internet de manière transparente.

Plusieurs normes de communication sans fil sont en train de voir le jour et le prix de ces équipements devient même très attractif, ce qui contribue au succès de ces technologies. Ce chapitre est consacré à la présentation de trois d'entre elles, certainement les plus abouties et les plus répandues aujourd'hui. Il s'agit des normes IEEE 802.11 [2, 4, 3, 6], de Bluetooth [36, 8] et des systèmes de réseaux cellulaires deuxième et troisième génération [74, 1, 182]. L'intérêt de ce chapitre est de présenter les particularités de chacune des ces technologies, car aucune d'entre elles ne se présente comme la technologie sans fil universelle. En effet, chacune a ses avantages mais aussi ses limitations, et bien que nous nous intéressions particulièrement aux protocoles de niveau 3 (IP), il est important de connaître les

mécanismes et les limites des technologies sous-jacentes utilisées pour émettre et recevoir les paquets. De plus, afin de mettre en place des déclencheurs de niveau 2 [195, 35, 124], il est primordial de connaître comment une technologie fonctionne et de découvrir quelles informations elle peut nous apporter afin d'optimiser les protocoles de gestion de la mobilité. Enfin cette analyse nous permettra de mieux comprendre l'intérêt de l'utilisation d'interfaces multiples ; plus particulièrement nous verrons les gains possibles grâce à l'utilisation simultanée de plusieurs interfaces et pour quelles raisons il est impératif de mettre en place des mécanismes efficaces de redirections de flux entre interfaces (voir chapitre 6).

La suite de ce chapitre est organisée de la façon suivante ; la section suivante est consacrée à la série des réseaux IEEE 802.11, ou Wifi, avec une étude théorique et pratique des débits et temps de handovers observés plus particulièrement dans les réseaux IEEE 802.11b. Ensuite nous verrons comment Bluetooth fonctionne, avec entre autres une illustration des débits possibles et comment la mobilité est gérée. Finalement, nous présenterons les réseaux cellulaires et leur évolution vers la troisième génération, l'UMTS, avant de conclure le chapitre. Nous avons volontairement laissé de côté les liaisons satellite car la mobilité entre points d'accès n'est pas évidente. De plus, les liaisons satellites représentent une technologie différente sur laquelle nous n'avons pas travaillé, et qui offre des liaisons asymétriques à fort délai.

2.2 IEEE 802.11

La série des normes IEEE 802.11 [2] (b [4], a [3] et g [6]), principalement développée aux Etats-Unis, a le même rôle que 802.3 pour l'Ethernet [11], à savoir permettre des communications IP à hauts débits. Les protocoles IEEE 802.11 représentent la technologie la plus utilisée actuellement dans les réseaux locaux sans fil et de nombreux produits sont disponibles sur le marché [76, 98]. La norme définit une interface de propagation radio (pour une interopérabilité entre fournisseurs), une méthode de codage et de modulation et une couche MAC (gestion du médium). Les réseaux locaux sans fil gérés par 802.11 sont destinés à être modulaires et très flexibles. Ils peuvent donc être optimisés pour différents environnements. Dans cette section, nous verrons brièvement un aperçu du protocole MAC, la gestion des déplacements et enfin nous proposerons une étude sur les débits et les temps de handover offerts par IEEE 802.11b.

2.2.1 Topologie

La norme IEEE 802.11 offre deux modes de fonctionnement, à savoir un mode avec infrastructure et un mode *ad hoc*. Dans le mode ad hoc, deux équipements sans fil peuvent communiquer directement entre eux lorsqu'ils sont physiquement dans le rayon de propagation l'un de l'autre. Ce mode de fonctionnement est laissé de côté dans le cadre de notre travail, puisqu'un réseau ad hoc est généralement autonome, et nous nous intéresserons exclusivement aux communications IP dans l'Internet.

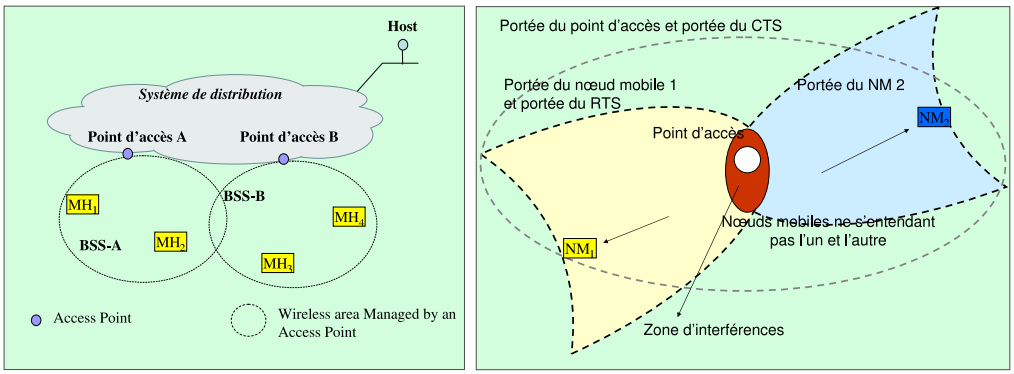
Dans le mode avec infrastructrure, l'entité de base d'un réseau sans fil est la *cellule*, contrôlée par un *point d'accès*. La cellule est la zone dans laquelle la communication sans fil a lieu. Sa taille est déterminée par le rayon de propagation du point d'accès. Un point d'accès est généralement muni de deux interfaces réseaux ; une interface sans fil par laquelle il reçoit toutes les trames échangées dans sa cellule et sur laquelle il retransmet les trames à destination des stations de la cellule, et une autre interface, généralement filaire, utilisée pour communiquer avec d'autres points d'accès ou même utilisée pour accéder à l'Internet. Les stations appartenant à une cellule peuvent se déplacer librement dans la cellule tout en maintenant leurs communications.

Les termes employés dans la norme sont le BSS (pour *Basic Service Set*) qui désigne l'ensemble d'au moins une station sans fil rattachée à un point d'accès ; le terme ESS (pour *Extended Service Set*) est utilisé pour définir une série de BSS contenant chacun au moins un point d'accès et connectés ensemble par un réseau local, eg. Ethernet, et dont les cellules se recouvrent en partie (voir figure 2.1(a)).

2.2.2 Technologie radio et débits offerts

Deux techniques de modulation sont définies dans la norme : l'une à saut de fréquences, FHSS (*Frequency Hopping Spread Spectrum*) et une autre séquentielle, DSSS (*Direct Sequence Spread Spectrum*). Les options du DSSS et du FHSS ont été adaptées pour être conformes aux recommandations du FCC[1] (FCC 15.247) pour des opérations dans la bande des 2,4GHz, qui possède une assignation mondiale pour des opérations non licenciées. Les anciennes générations de carte 802.11b fonctionnaient à des débits de 1 et 2 Mb/s. Cependant, avec l'adoption d'un code à clés complémentaires (Complementary Code Keying), les débits de 5,5 et 11 Mb/s se sont ajoutés. La technologie de modulation alors utilisée pour supporter

[1]FCC - Federal Communications Commission



(a) Structure d'un ESS (Extended Service Set)

(b) Problème des noeuds cachés

FIG. 2.1 – Structure d'une architecture IEEE 802.11

ces débits est DSSS. Les nouvelles versions de la norme IEEE 802.11 offrent des débits supérieurs allant jusqu'à 54 Mb/s pour 802.11a dans la bande des 5GHz et 802.11g dans la bande des 2,4GHz.

2.2.3 Accès au médium

Lorsque plusieurs stations sont rattachées à la même cellule, il est nécessaire de mettre en place une politique d'accès au médium. En effet, toutes les stations d'une cellule ne peuvent pas transmettre simultanément, c'est pourquoi il faut définir un algorithme qui minimisera les collisions et maximisera la bande passante utile disponible pour les données. L'algorithme adopté par la série 802.11 est un algorithme CSMA/CA (*Carrier Sense Multiple Access / Collision Avoidance*). DCF (pour *Distributed Coordination Function*) et PCF (pour *Point Coordinator Functiun*) sont les protocoles de référence qui ont été définis dans la norme. D'autres méthodes comme eDCF ou HCF [5] sont actuellement en cours de normalisation pour proposer la mise en place de qualité de service. Dans la suite, on se propose de décrire DCF, la méthode d'accès au médium de référence sur laquelle toutes les autres méthodes sont basées.

DCF

La méthode d'accès de base est DCF, qui est un protocole d'accès libre, lié au temps et asynchrone. Une station désirant transmettre des données commence à tirer un nombre aléatoire qui lui indique un temps d'attente initial (voir la sous-section suivante). Ce temps d'attente passé, elle sonde le médium. Si le médium est occupé, c'est-à-dire qu'une autre machine est en train d'émettre, alors la station retarde sa transmission. Inversement, si le médium est libre, elle est autorisée à transmettre.

Ce protocole est très efficace si le médium n'est pas trop chargé. Cependant, deux stations peuvent sonder le médium simultanément et donc commencer leur transmission en même temps. Il y aura donc collision et aucune des deux trames de données ne sera transmise correctement. Les situations de collision doivent être identifiées afin que les couches 802.11 puissent retransmettre le paquet de manière autonome, sans recourir à un protocole d'une couche supérieure. Comme une station ne peut pas écouter sa transmission en même temps qu'elle transmet (algorithme *Collision Detection* de 802.3), un système d'acquittement positif est mis en place. Si une station émettrice ne recoit pas d'acquittement après la transmission d'une trame, elle considère qu'une collision a eu lieu sur le médium. Elle devra alors retransmettre la trame.

Un autre problème lié aux caractéristiques radio est apparu lors de la conception d'algorithme de partage de médium. Deux stations rattachées au même point d'accès peuvent ne pas s'entendre mutuellement comme le montre la figure 2.1(b). 802.11 propose alors le mécanisme de RTS/CTS. Ce mécanisme impose à tout émetteur de transmettre une courte trame RTS avant l'émission d'une trame de données. Le destinaire doit alors répondre avec un CTS. Cet échange permet notamment aux stations dans la portée du destinataire de savoir qu'une transmission va avoir lieu. Si une collision se produit, la détection sera plus rapide étant donné que les trames RTS/CTS sont plus courtes que les trames transportant des données.

Algorithme de backoff

Lorsqu'une station désire émettre alors que le médium est occupé, elle doit différer sa transmission. Or, si d'autres stations désiraient émettre pendant l'occupation du médium, elles vont toutes vouloir émettre au même moment, c'est-à-dire une fois que le médium sera libéré, puisqu'elles sonderont un médium libre en même temps. Pour parer ce problème, un algorithme de *backoff exponentiel* est mis en

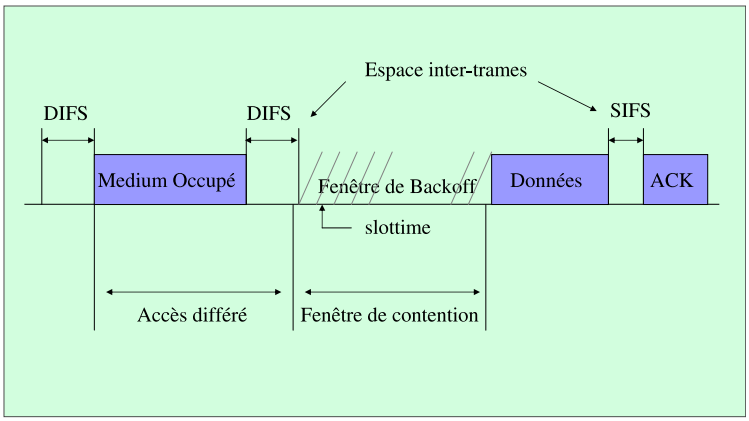


FIG. 2.2 – Algorithme de Backoff - DCF

place : une station tire au hasard un nombre entre 0 et $CW-1$ donné. L'intervalle $[0\,;CW-1]$ est appelé fenêtre de backoff. A la première transmission, $CW-1$ vaut $Cwmin$, défini dans la norme égal à 32. Puis à chaque retransmission, $CW-1$ est doublé jusqu'à $CWmax$ (défini dans la norme égal à 1024).

Une station attendra alors le nombre tiré aléatoirement dans la fenêtre de backoff multiplié par un *slottime* avant d'accéder au médium, toujours en vérifiant qu'aucune autre station n'a accédé au médium avant elle. Le slottime est défini de telle sorte qu'une station sera toujours capable de déterminer si une autre station a accédé au médium au début du slot précédent. Le temps de backoff est décrémenté uniquement quand le médium est libre. La figure 2.2 illustre ce mécanisme.

2.2.4 Déplacement des stations

Lorsqu'une station désire pénétrer dans une cellule existante (BSS), soit après le mode veille ou simplement après un déplacement, elle doit récupérer des informations de synchronisation du point d'accès. Une station peut récupérer ces informations par l'un des deux moyens suivants :

- **Sondage passif** : la station attend simplement de recevoir une trame de signalisation du point d'accès. Cette trame spécifique est envoyée périodiquement par le point d'accès, avec une fréquence généralement configurée à 100 millisecondes.
- **Sondage actif** : la station essaye de localiser le point d'accès en transmettant des trames spéciales appelées *Probe Request*. Les réponses des points

d'accès permettent à la station d'identifier les points d'accès dans sa portée plus rapidement qu'une attente passive.

Processus d'authentification

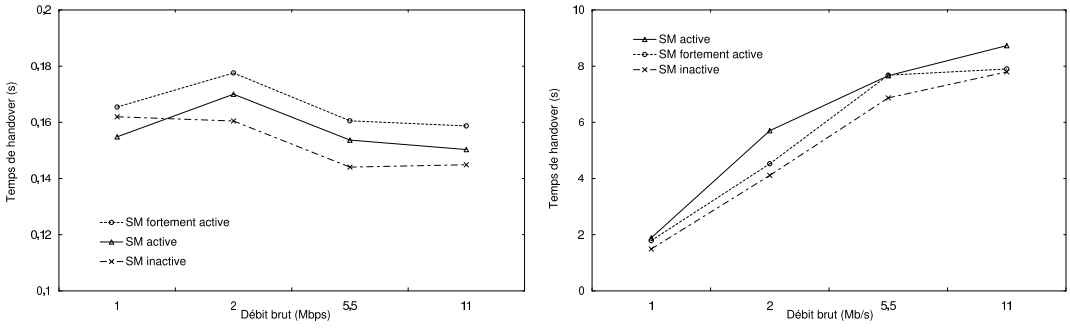
Une fois que la station a localisé et élu un point d'accès, elle passe dans un processus d'authentification. Ce dernier repose sur un échange d'informations entre la station et le point d'accès, où chaque partie prouve la connaissance d'un mot de passe donné. Ce mécanisme étant très peu fiable et non adapté à une authentification globale telle que nécessaire dans le déploiement de Hot Spots commerciaux [76], la norme IEEE 802.11i [7] a défini une authentification de plus haut niveau, appelée *Réseaux à Sécurité Robuste* où chaque partie prouve son identité. Dans cette méthode, l'authentification 802.11 ne doit pas être utilisée : la station doit d'abord s'associer au point d'accès. Cependant, cette association n'est pas complète dans le sens où tout trafic de données est filtré par un port 802.1x [9]. Ce port laisse uniquement passer les paquets d'authentification échangés entre la station, le point d'accès et une tièrce partie, généralement un serveur AAA. Cette méthode d'authentification est décrite dans [195].

Processus d'association

Une fois que la station est authentifiée (authentification de niveau MAC), elle commence le processus d'association. Ceci consiste en l'échange d'informations concernant les capacités des stations et du BSS. Ce processus d'association permet également à l'ensemble des points d'accès constituant l'ESS de connaître la position courante de la station. Une station est capable de transmettre et de recevoir des trames de données uniquement une fois le processus d'association achevé.

Nous avons mesuré le temps nécessaire pour réaliser un handover entre deux point d'accès IEEE 802.11b (opération appelée *Handover de niveau 2*). Les mesures du temps de handover de niveau 2 pour les différents débits de la norme sont données dans les figures 2.3. La figure 2.3(a) représente les temps de handover de niveau 2 pour une station se déplaçant seule alors que la figure 2.3(b) représente le temps de handover pour 5 stations se déplaçant simultanément. Dans chaque cas de figure, la (les) station(s) émette(nt) ou non un trafic de données. Le faible impact du trafic de données sur le temps du handover ressort clairement de ces mesures. Par contre, le fait que 5 stations se déplacent simultanément augmente de manière conséquente le temps de handover : lorsque la station est seule, le temps de handover est approximativement 160 millisecondes, alors que lorsque

plusieurs stations se déplacent simultanément, le temps de handover varie entre 2 et 8 secondes. La mise à jour des firmware des cartes utilisées (Cisco Aironet 350) a corrigé ce problème et le temps de handover est plus stable par rapport au nombre de stations. Les dernières mesures donnent un temps de handover moyen situé entre **200 et 300 millisecondes** quelque soit la configuration.



(a) Temps de handover pour une station seule

(b) Temps de handover pour 5 stations se déplaçant simultanément

FIG. 2.3 – Temps de handover observé dans 802.11b

2.2.5 Sécurité

La sécurité de la transmission de données est un aspect primordial de la réussite d'une technologie. Le comité 802.11 a donc abordé le problème en fournissant un algorithme de cryptage des trames envoyées : le WEP (*Wired Equivalent Privacy*). L'algorithme WEP est un générateur de nombres pseudo-aléatoires initialisés par une clé secrète partagée. Ce générateur de nombres pseudo-aléatoires produit une séquence clé de bits pseudo-aléatoires égale en longueur au paquet le plus large possible et qui est combiné avec le paquet entrant / sortant produisant ainsi le paquet transmis par radio.

La figure 2.4 présente les temps d'interruption lors du basculement d'une station d'un point d'accès à un autre. Ce graphique montre l'impact des différents mécanismes de sécurité proposés pour la sécurisation des liaisons radio sur les temps de déplacement. Les tests réalisés montrent notamment que le cryptage WEP (clé de 128 bits) entraîne une pénalisation dans le temps de handover de 7% alors que l'activation des mécanismes IEEE 802.1X entraînent quant à eux un délai de 11% par rapport à un handover sans aucun niveau de sécurité.

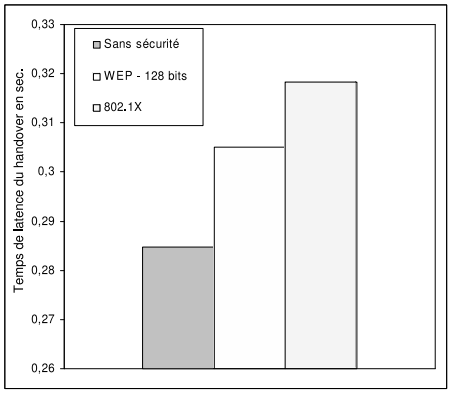


FIG. 2.4 – Impact de la sécurité sur les temps de déplacement

2.2.6 Calcul des débits utiles théoriques dans 802.11b

Comme il a été dit plus haut, 802.11b offre plusieurs débits bruts. On peut donc configurer un point d'accès pour qu'il offre 1, 2, 5,5 ou 11 Mb/s. On peut noter que si l'on autorise tous les débits, le point d'accès s'adaptera au plus faible débit supporté par les stations rattachées. Ces débits bruts servent à faire passer les données, mais aussi toutes les informations de contrôle comme les adresses des équipements ou les acquittements. Les temps inter-trames utilisent aussi une part importante du débit brut.

C'est pourquoi il peut se révéler intéressant de calculer les débits utiles effectivement offerts à une ou plusieurs stations rattachées à un point d'accès. Pour cela, il s'agit d'analyser avec précision le protocole d'accès au médium et l'algorithme de backoff décrits plus haut. D'après l'algorithme, le temps d'accès augmente fortement avec le nombre d'utilisateurs actifs. Une station désireuse d'émettre sonde le médium, attend un temps *DIFS* et transmet son paquet de données si le médium est toujours libre. Si le paquet est reçu correctement, le récepteur envoie un aquittement après avoir attendu un temps minimum inter-trames *SIFS*. Si l'acquittement n'est pas reçu par l'émetteur, il considère qu'il y a eu collision et engage l'algorithme de retransmission (backoff exponentiel). Les calculs présentés dans cette section rejoignent les résultats obtenus dans [77, 39].

Utilisateur unique

Pour un utilisateur unique qui désire envoyer une trame, la transmission prendra le temps suivant :

$$T_{single} = backoff + DIFS + T_{tr} + SIFS + ACK \tag{2.1}$$

$$\text{Où} \begin{cases} T_{tr} & est\ le\ temps\ de\ transmission\ d'une\ trame \\ SIFS & est\ le\ temps\ entre\ une\ trame\ et\ son\ acquittement\ (10\ micro-secondes) \\ ACK & est\ le\ temps\ de\ transmission\ d'un\ acquittement\ (30\ micro-secondes) \\ DIFS & est\ le\ temps\ de\ sondage\ du\ canal\ (50\ micro-secondes) \\ backoff & est\ le\ temps\ total\ d'attente\ de\ l'algorithme\ de\ backoff \end{cases}$$

La variable *backoff* représente le temps cumulé d'attente suite à l'algorithme de Backoff exponentiel. Dans ce calcul, on considère qu'il n'y a pas de perte sur le canal, ie. le temps d'attente moyen tendra vers $((32-1)/2).slottime$, avec le slottime qui vaut 20 micro-secondes. On considère également que chaque fois que la station veut transmettre, le médium est libre. Considérons des trames transportant 1500 octets de données au niveau 2 ($MTU = 1500\ octets$), c'est-à-dire que les en-têtes IP/UDP/TCP et autres sont comprises dans ces 1500 octets de données. IEEE 802.11 ajoute une en-tête 802.11 de 34 octets comprenant entre autre les adresses émettrice et destinatrice et une en-tête PLCP consituée de 72 bits envoyés à 1 Mb/s et de 48 bits envoyés à 2 Mb/s. Le ratio entre le débit brut et le débit utile ($\frac{\text{Temps de transmission du MTU}}{Temps\ de\ transmission\ 802.11}$) et le débit utile correspondant sont donnés dans le tableau 2.1.

	1 Mb/s	2 Mb/s	5,5 Mb/s	11 Mb/s
Ratio	9,939	0,903	0,797	0,673
Débit utile	0,939	1,806	4,384	7,403

TAB. 2.1 – Débits utiles théoriques dans IEEE 802.11b

Plusieurs utilisateurs

Lorsque plusieurs utilisateurs sont actifs à un moment donné sur le même point d'accès, ils doivent se partager la bande passante. D'après l'algorithme d'accès

au médium, chaque fois qu'une station subit une collision, elle doit doubler la borne supérieure de l'intervalle de Backoff CW (jusqu'à $CWmax = 1024$). Or, une station rencontrera d'autant plus de collisions qu'il y a d'utilisateurs et/ou de trafic. Ceci implique que son temps de transmission est dépendant du nombre de stations actives et des flux générés par ces dernières. L'équation suivante représente le temps de transmission d'une trame :

$$T_{multiple} = backoff + (retrans + 1)(DIFS + T_{tr} + SIFS + ACK) \qquad (2.2)$$

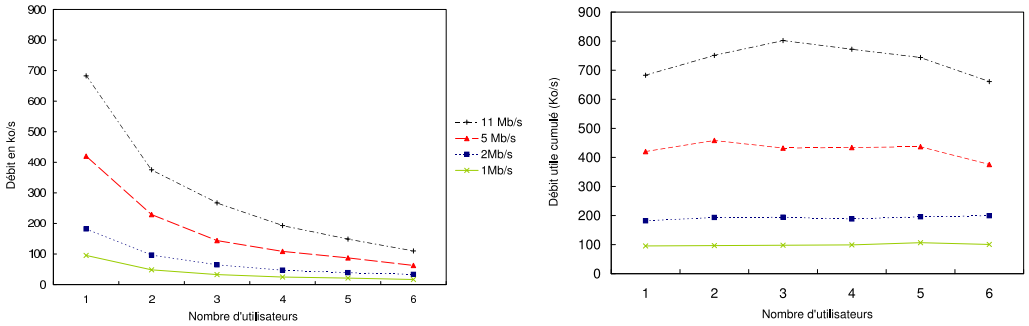
où les variables utilisées sont les mêmes que dans 2.1.

Si une station entre en collision, elle s'en apercevra après le temps de réception de l'acquittement (qu'elle ne recevra pas). La station devra alors tirer un nouveau backoff et retransmettre la trame entièrement (attente de DIFS, temps de transmission, attente SIFS et temps de réception de l'acquittement). Ici la variable $backoff$ désigne le temps mis pour écouler le backoff, c'est-à-dire le nombre aléatoire tiré plus le nombre de slots pendant lesquels le canal n'est pas vide. Ceci est dû au fait qu'une station ne peut décrémenter son backoff seulement lorsque le canal est libre. Ainsi, cette formule dépend fortement du trafic échangé dans la cellule. En cas de collision ou même lorsque le canal est occupé, une transmission peut être fortement différée dans le temps. Une étude par simulation des débits utiles observés pour plusieurs stations se partageant le médium est donnée dans la section 4.4. La section suivante donne les débits mesurés sur notre plate-forme de tests.

2.2.7 Mesures réelles sur différents débits

La mise en place de tests avec mesures réelles est un bon moyen de connaître le débit dont pourra profiter un ensemble de stations rattachées à un point d'accès. Les mesures de débits selon le nombre d'utilisateurs sont présentées dans cette section. Les stations utilisées sont des portables Compaq Evo N150 munis de cartes IEEE 802.11b Cisco Aironet 350. La plate-forme de tests est exclusivement sous IPv6. Le générateur de trafic utilisé est TTCP6 [181].

Les figures 2.5 représentent le débit utile mesuré sur un utilisateur (figure 2.5(a)) et le débit utile global pour le système (figure 2.5(b)) sur une moyenne de 100 mesures du transfert de 10000 paquets de 1590 octets pour chaque débit brut proposé par la norme. Une première remarque concerne les ratio entre le débit utile et le



(a) Impact du nombre d'utilisateurs sur le débit utile dans 802.11b

(b) Impact du nombre d'utilisateurs sur le débit cumulé dans 802.11b

FIG. 2.5 – Débit utile dans les réseaux IEEE 802.11b

débit brut. Pour un utilisateur, le débit observé pour 11 Mb/s est de 682,6 Ko/s, ce qui fait un ratio de 0,496 contre 0,673 théorique. Plus le débit est faible, plus on se rapproche du ratio théorique. Effectivement, à 1 Mb/s, le débit observé est de 95,5 Ko/s ce qui donne un ratio de 0,764 contre 0,939 théorique. L'efficacité est donc supérieure pour un débit brut de 1 Mb/s, comme calculé théoriquement, mais les performances sont bien inférieures que les calculs théoriques. Ceci peut s'expliquer par le fait que l'algorithme d'accès au canal est meilleur pour plusieurs stations que pour une seule. Les courbes de la figure 2.5(b), spécialement celle à 11 Mb/s montrent bien la tendance du débit global du système lorsque le nombre d'utilisateurs augmente : le débit du système devient meilleur pour 2, 3 voire 4 utilisateurs que lorsqu'un seul utilisateur est dans le BSS.

La figure 2.5(a) nous montre que le partage de charge est bien géré : le débit utile disponible pour un utilisateur diminue d'un facteur égal au nombre d'utilisateurs simultanés. Ces premières mesures nous confirment donc que 802.11b est bien adapté au trafic Internet et supporte de manière correcte jusqu'à six utilisateurs simultanés générant un trafic important. Une étude plus poussée par simulation du partage de la bande passante et de l'impact des collisions sur le débit utile est donnée dans la section 4.4.

2.2.8 Conclusion

La norme IEEE 802.11 propose une technologie fonctionnelle et très populaire. Différents mécanismes permettent des optimisations et certains aspects comme la

sécurité ont évolué au cours du temps pour offrir une technologie plus fiable et robuste. De par son prix et sa simplicité d'utilisation, la norme IEEE 802.11 est maintenant devenue la référence dans le domaine des WLAN. Cependant, la relative faible portée des équipements (quelques dizaines de mètres en environnement intérieur), la forte consommation électrique et le manque de qualité de service sont encore des points faibles de la norme. De plus, bien qu'on puisse s'attendre à un important déploiement de cette technologie dans les années à venir, un recouvrement total de l'espace géographique n'est pas envisageable. C'est pourquoi, d'autres technologies, comme Bluetooth et GSM/GPRS/UMTS présentés ci-après, proposent une alternative ou plutôt une continuité de couverture de ces réseaux WLAN.

2.3 Bluetooth

Bluetooth [36, 8] est un protocole de communication sans fil, dont le but premier est de remplacer les câbles qui relient les périphériques entre eux. Bluetooth a été conçu pour être résistant aux parasites, simple à implémenter, sécurisé, consommant peu d'energie, et peu cher à fabriquer. Cette technologie peut être utilisée dans les domaines de la domotique, des télécommunications personnelles et de l'informatique mobile, ainsi que pour accéder à des réseaux informatiques sans fil.

C'est ce dernier point qui rentre dans le cadre de ce travail. Nous allons voir dans cette section les différentes possibilités de raccordement d'un périphérique Bluetooth à un réseau IP, puis nous verrons comment gérer la mobilité entre différents points d'accès tout en essayant de conserver une connectivité IP.

2.3.1 Structure en couches

Couche Radio

Comme la plupart des systèmes réseau, la pile Bluetooth est structurée en cinq couches. Bluetooth opère dans la bande de fréquence ISM (*Industrial, Scientific, Medical*) située dans les 2,4GHz (comme IEEE 802.11b) car elle est libre d'utilisation dans la majorité des pays. La norme définit 79 canaux espacés de 1Mhz chacun, à partir de 2402MHz (jusqu'à 2480MHz donc). Malheureusement, en France et au Japon, au moment de la rédaction des spécifications de la version 1.1, seuls les 23 premiers canaux étaient libres. Le matériel fabriqué spécifique-

ment pour ces deux pays (respectant cette restriction) est donc incompatible avec le reste du parc mondial. La modulation utilisée est une modulation en fréquence. Plusieurs classes émettrices y sont définiess pour contrôler la puissance d'émission, avec le maximum à 100mW, ce qui permet une transmission sur plusieurs mètres.

Couche BaseBand

Les périphériques Bluetooth s'organisent en *piconets*. Un seul équipement est désigné comme maître par piconet, qui peut regrouper jusqu'à 7 périphériques esclaves actifs (qui peuvent émettre sur le lien) et un grand nombre d'esclaves passifs. Un périphérique peut prendre part à plusieurs piconets, en tant que maître ou esclave (mais ne peut être maître que d'un seul piconet) pour former un *scatternet* entre ces piconets.

Dans Bluetooth, les communications se font uniquement entre maître et esclave. Si deux esclaves d'un piconet veulent communiquer, ils doivent créer un second piconet entre eux, à la manière d'un scatternet. Un périphérique peut choisir d'écouter alternativement les différents piconets auxquels il participe mais ne peut en écouter qu'un à la fois.

Pour résister aux interférences provoquées par l'utilisation importante de la bande des 2,4GHz (IEEE 802.11b, fours à micro-ondes...), Bluetooth utilise le saut de fréquence : chaque paquet est envoyé sur un canal différent, et il y a 1600 sauts de fréquence par seconde. On parle de 1600 slots, chacun pouvant contenir un paquet, et durant 625 micro-secondes. La séquence des canaux utilisés lors de ces sauts est connue de tous les périphériques, car elle est dérivée de l'adresse MAC du maître. La phase (position courante dans la séquence) est envoyée par le maître aux esclaves lors de leur connexion, garantissant que tous écoutent le même canal à un instant donné.

A ce niveau de la pile, le débit disponible est de 1 Mb/s. Bluetooth définit aussi des paquets plus longs, pouvant utiliser 3 ou 5 slots. Dans ce cas, les 3 ou 5 slots sont transmis sur le même canal, sans saut, et le paquet suivant est émis sur le canal correspondant à une avancée de 4 (3+1) ou 6 (5+1) dans la séquence des sauts, ce qui évite une désynchronisation avec les périphériques n'ayant pas écouté le réseau à ce moment. Les paquets de données sont appelés DMx (pour *Data Medium rate*) ou DHx (pour *Data High rate*) où x peut valoir 1, 3 ou 5, ce qui indique le nombre de slots utilisés pour la transmission du paquet. La différence entre ces deux types de paquets est principalement la taille du code correcteur : 1/3 des bits des paquets DMx définissent un code correcteur, contre 16 bits seulement

pour un CRC (pas de correction d'erreurs) dans les paquets DHx (les paquets ont une taille comprise entre 600 et 3000 bits). Nous verrons dans la section 2.3.3 les conséquences de cette différence.

Le canal de communication créé par cette suite de slots est découpé en deux : les slots pairs sont réservés aux transmissions du maître, et les slots impairs sont réservés aux réponses des esclaves (la parité est déterminée par la phase de la séquence de saut). Un esclave ne peut transmettre sur le canal que lorsque le maître lui a envoyé un paquet dans le slot précédent.

Deux types de liens sont définis dans cette couche :

- Lien SCO (*Synchronous, Connection Oriented*) : Ce type de lien, orienté connexion, est utilisé pour des communications de type voix à 64 kb/s. L'établissement de ce lien provoque une réservation de slots, afin de s'assurer que le débit peut être soutenu tout en garantissant une faible gigue. C'est une connexion point à point entre le maître et un esclave. Un périphérique peut supporter jusqu'à trois connexions SCO.
- Lien ACL (*Asynchronous, Connection Less*) : Il y a un seul lien ACL par piconet, utilisant les slots qui ne sont pas réservés. Ce lien est de type non connecté, il fait de la commutation de paquets.

L'établissement d'une connexion sécurisée nécessite que les deux parties doivent posséder le même secret, le code PIN (de 8 à 128 bits). Un premier challenge est envoyé, utilisant une clé d'initialisation dérivée du code PIN, d'un nombre aléatoire et de l'adresse du périphérique appelant. Si le challenge est "relevé", cette clé d'initialisation est ensuite utilisée pour échanger la clé de lien, qui est dérivée d'une clé privée de 128 bits contenue dans chaque périphérique (clé générée à la première mise sous tension du périphérique Bluetooth, puis mémorisée en ROM). Cette clé de lien est propre à la connexion entre les deux périphériques et elle sert de secret partagé pour la génération d'une clé qui sera utilisée pour chiffrer les données transmises.

Couche LMP

LMP (protocole de gestion de lien) définit les différents messages qu'envoient les périphériques lorsqu'ils effectuent les actions relatives à l'utilisation du canal logique. Le premier octet de la charge utile comprend un numéro de canal logique sur deux bits qui sert à déterminer si le paquet est de type données ou de type gestion du lien et à identifier les différentes procédures que définit LMP. La majorité de ces actions peuvent être initiées soit par le maître, soit par l'esclave. Les

actions les plus importantes sont l'authentification, la gestion de la clé du lien, le chiffrement, la découverte de fonctionnalités supportées par les voisins, le changement de rôle entre maître et esclave, les requêtes de nom, la déconnexion et la négociation de la puissance.

Couche L2CAP

Un paquet L2CAP rajoute un en-tête comprenant la longueur des données en octets, ainsi qu'un numéro de canal. Les fonctions principales de L2CAP sont (1) le multixage, permettant à plusieurs applications d'utiliser le même lien ACL entre deux périphériques, en utilisant des numéros de canaux L2CAP différents. (2) la segmentation et le réassemblage, permettant de remonter des paquets ayant jusqu'à 64 kb de données alors que la charge utile maximale d'un paquet au niveau BaseBand est de 2745 bits. (3) la qualité de service, permettant aux applications de demander que certains paramètres soient respectés, comme la bande passante ou la gigue. L2CAP est donc un protocole dont le rôle est de fournir une couche réseau complète aux couches supérieures et aux applications.

Profils

Les profils servent à garantir la compatibilité de tous les périphériques Bluetooth en décrivant de manière précise les méthodes à employer pour les différents domaines auxquels est destiné Bluetooth (établissement d'un lien série, d'une connexion réseau, téléphonie, envoi de fichiers et synchronisation de périphériques en utilisant OBEX...). Un profil particulier est le protocole de découverte de services, qui permet à un périphérique de demander à un voisin s'il fournit un service particulier.

2.3.2 Applications

Découverte des voisins

La découverte des voisins se déroule sur 16 fréquences désignées. Elle se déroule en deux parties. Le côté appelant envoie un paquet ID et enregistre les réponses reçues sur chacune des 16 fréquences. Le coté appelé peut choisir de passer en mode *Inquiry Scan* (attente de requête) à intervalles réguliers. Il choisit une fréquence parmi les 16 disponibles, puis lorsqu'il reçoit un paquet ID, il attend pendant une

durée choisie aléatoirement, pour éviter les collisions entre plusieurs réponses, puis envoie un paquet de synchronisation.

Il existe deux types de paquets ID : un ID général, auquel tous les périphériques répondent, et des ID dédiés, correspondant à différents types de périphériques, auxquels seuls les périphériques du même type répondent.

Connexion de deux périphériques Bluetooth

Après la phase de demande de découverte de service, le périphérique appelant, qui sera le maître du piconet qui va être créé, envoie un *Page* au périphérique auquel il souhaite se connecter. Le Paging se déroule sur deux plages de 16 fréquences dédiées. La séquence est la suivante : le maître envoie un paquet ID, qui contient l'adresse MAC de la cible sur chacune des 16 fréquences de la première plage, et s'il n'a pas de réponse, il réémet ce paquet sur la seconde plage. L'esclave passe à intervalles réguliers en mode Page Scan et attend un paquet ID contenant son adresse MAC. Sur réception de ce paquet, il émet un paquet ID contenant sa propre adresse MAC à la fréquence correspondant au slot suivant. Lorsque le maître reçoit ce paquet, il sait quelle fréquence écoutait l'esclave, et il peut en déduire à quelle fréquence il doit lui envoyer le paquet suivant, qui est un paquet FHS contenant ses informations d'horloge. L'esclave utilise ce paquet pour déterminer l'adresse MAC du maître, le code d'accès du piconet qui vient d'être créé, son adresse de membre actif, la séquence de sauts, ainsi que la position courante dans cette séquence. Il envoie ensuite un paquet ID contenant son adresse MAC pour acquitter le paquet FHS. Puis le maître et l'esclave échangent des *POLL* pour terminer le paging. Les périphériques sont maintenant prêts à établir un lien en utilisant LMP.

Connexion d'un périphérique à un piconet existant

Comme nous l'avons vu avant, lorsqu'un périphérique initie une connexion, il devient le maître du piconet ainsi créé. Imaginons un périphérique A qui souhaite se connecter à B, B étant maître d'un piconet. Si A initie une connexion à B, deux piconets seront instanciés : celui de A, dont B est esclave, et celui de B. Cette situation occasionne une perte importante : le piconet de B est indisponible lorsque ce dernier discute sur le piconet de A. Il est alors possible à B d'inverser les rôles avec A et d'éliminer ainsi le partage de ses slots entre les deux piconets. Il n'y aura alors plus qu'un seul piconet. Etant donné que les communications se font entre maître et esclave, et pas entre deux esclaves, on peut imaginer le cas d'un

point d'accès à un réseau et de trois clients : il vaut mieux que le point d'accès soit maître d'un piconet ayant trois esclaves, plutôt que d'avoir trois piconets ayant chacun le point d'accès pour seul esclave et un des clients pour maître : le point d'accès n'aura pas à gérer trois liens ACL, ni à partager son temps entre les trois piconets.

Utilisation d'IP au dessus de Bluetooth

Il existe actuellement deux *Profils* Bluetooth permettant un accès à un réseau IP. RFCOMM est une émulation de port série au-dessus de Bluetooth, avec laquelle il est possible d'établir une liaison PPP [49]. Cette solution n'est pas la plus élégante, du fait des contraintes qu'impose PPP, mais c'est la solution historique car la seule disponible sur la plupart des systèmes anciens, et des systèmes propriétaires. D'un autre côté BNEP est une solution plus récente, qui offre un moyen de faire passer plusieurs protocoles réseaux, dont IPv4 et IPv6 directement sur L2CAP. Malheureusement, dû à son développement tardif, elle n'est pas présente sur tous les systèmes équipés de Bluetooth.

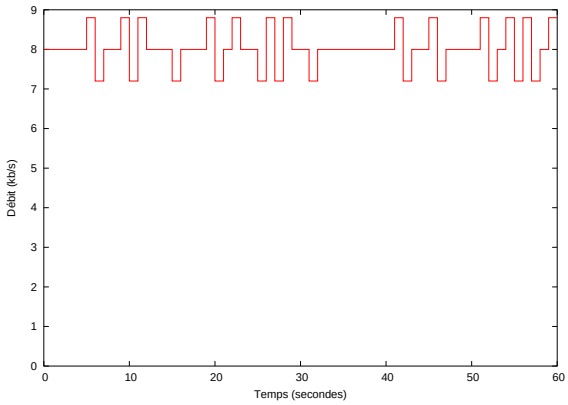
2.3.3 Mesure de qualité des liens RFCOMM et BNEP

Cette section est consacrée à l'évaluation des liens proposés par Bluetooth pour transporter des paquets IPv6. Ces mesures concernent le débit utile observé sur un lien entre un noeud Bluetooth (un IPAQ 3870) et son point d'accès (point d'accès du commerce AXIS pour RFCOMM et point d'accès logiciel sous linux pour BNEP). Le générateur de trafic MGEN [110] qui permet de configurer la distribution d'émission des paquets et leur taille a été utilisé.

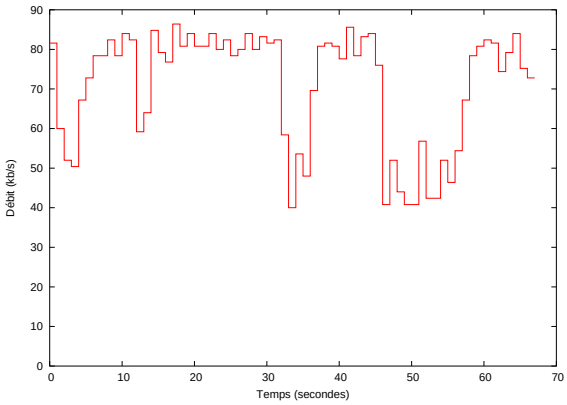
RFCOMM

Dans le premier test, un flux de 10 paquets de 100 octets par seconde est échangé entre le noeud mobile et son correspondant (noeud du réseau filaire). Dans la figure 2.6(a), on observe que la liaison est satisfaisante et fournit bien le débit de 8 kb/s. Lorsque 100 paquets de 100 octets par seconde sont envoyés, le lien a du mal à assurer le débit comme le montre la figure 2.6(b).

La saturation du lien est due aux paquets utilisés, qui sont par défaut DM1 (voir section 2.3.1). Or ces paquets ne permettent qu'un débit maximum de 108 kb/s.

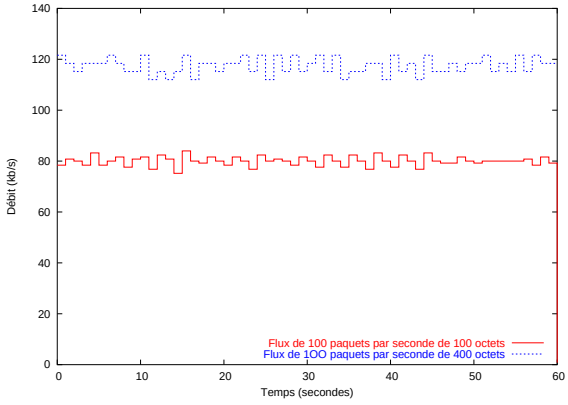


(a) Réception de 10 paquets de 100 octets par seconde

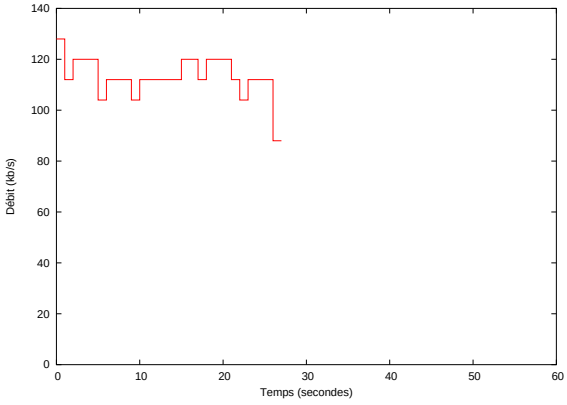


(b) Réception de 100 paquets de 100 octets par seconde

FIG. 2.6 – Réception de flux sur une interface Bluetooth en mode RFCOMM - type de paquet DM1



(a) Réception de 100 paquets de 100/400 octets par seconde



(b) Réception de 100 paquets de 1000 octets par seconde

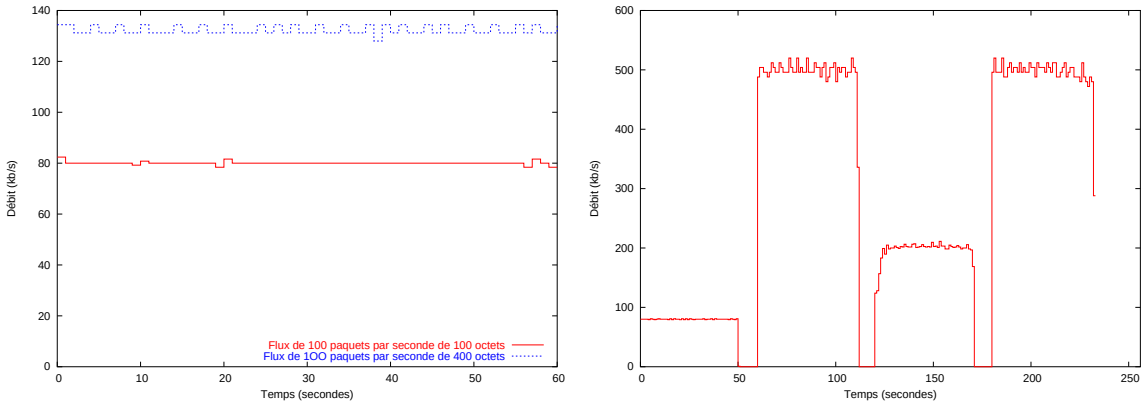
FIG. 2.7 – Réception de flux sur une interface Bluetooth en mode RFCOMM - type de paquet DH5

En forçant le type de paquet à DH5 (qui permet un débit théorique de 723 kb/s en liaison asymétrique), on observe maintenant un débit correct de 80 kb/s comme le montre la courbe du bas de la figure 2.7(a). Lorsque le débit est encore accéléré à 100 paquets de 400 octets par seconde, on s'aperçoit sur la courbe du haut de la

figure 2.7(a) que le lien sature à environ 120 kb/s ; cette vitesse correspond en fait à la vitesse d'horloge de la liaison avec le chipset Bluetooth dans l'iPAQ.

Lorsque le débit augmente encore (100 paquets de 1000 octets) on observe sur la figure 2.7(b) que la connexion PPP se désynchronise et la connexion IP est donc perdue pendant une trentaine de secondes. La liaison RFCOMM, quant à elle, est toujours en place, et il faut uniquement relancer la connexion PPP.

BNEP



(a) Réception de 100 paquets de 100/400 octets par seconde

(b) Réception de différents flux

FIG. 2.8 – Réception de flux sur une interface Bluetooth en mode BNEP

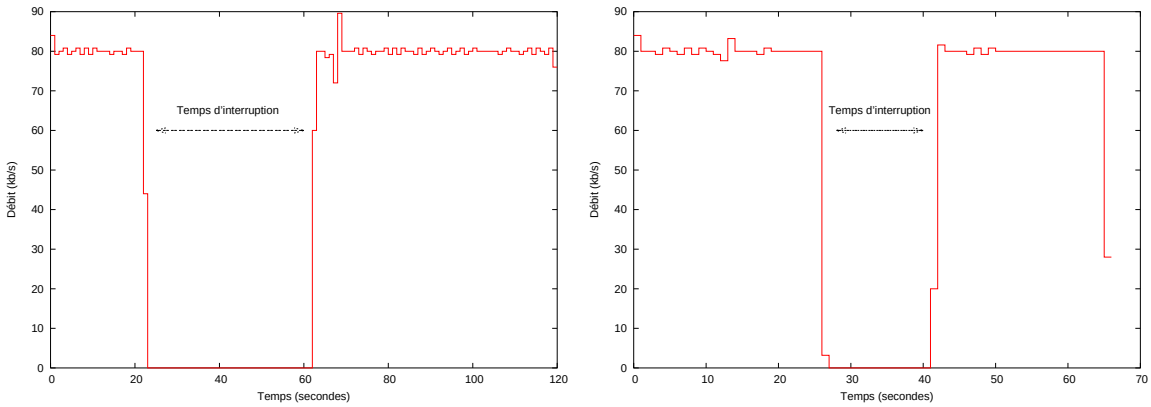
Pour ces tests, seuls les paquets de type DH5 ont été utilisés. La figure 2.8(a) montre le débit observé sur le noeud lorsqu'il reçoit 100 paquets de 100 octets par seconde (courbe du bas). On observe que le débit est respecté et que la courbe est plus lisse qu'avec RFCOMM pour le même débit (voir figure 2.6(b)).

En montant le débit à 100 paquets de 400 octets par seconde et plus, on observe à nouveau que la liaison sature à 130 kb/s, mais la connexion n'est jamais désynchronisée avec BNEP et la connectivité IP reste disponible (courbe du haut de la figure 2.8(a)).

Afin de tester les réelles performances de Bluetooth, la figure 2.8(b) montre différents débits pour un flux échangé directement entre deux périphériques Bluetooth en mode ad hoc. La liaison sature ici à 500 kb/s, ce qui est cohérent, étant

donné le débit théorique d'une liaison DH5, soit 723 kb/s au niveau de la couche 2. Ces premiers tests nous révèlent que la méthode BNEP est de loin préférable pour les connexions IP et que le type de paquet utilisé a une forte influence sur le débit utile. Cependant, les performances globales de Bluetooth au niveau du débit de données restent assez faibles et ne pemettront pas l'échange de tout type de données, comme un flux temps réel de vidéo.

2.3.4 Mobilité avec BNEP



(a) Handover Bluetooth

(b) Déconnexion manuelle

FIG. 2.9 – Réception de flux lors d'un handover

La mobilité d'un noeud se rattachant consécutivement à plusieurs points d'accès n'est pas réellement supportée par Bluetooth. La mise en place de la mobilité consiste en la recherche d'un nouveau point d'accès auquel se reconnecter en cas de déconnexion. Pour cela, il faut avant tout que les points d'accès utilisent un démon de découverte de service, chargé de répondre aux requêtes SDP recherchant un point d'accès muni du service BNEP. Les capacités "hotplug" sur le client sont ensuite nécessaires, pour que la reconnexion puisse se faire sans l'intervention de l'utilisateur.

Pour le premier test, nous avons débranché le point d'accès sur lequel était rattaché le noeud mobile. Ceci provoque donc la perte de connexion sur le noeud mobile, la découverte d'un nouveau point d'accès et la connexion. La figure 2.9(a) retrace les évènements suivants :

– 20 secondes après le dernier paquet reçu sont nécessaires pour déterminer

que la connexion avec le point d'accès courant a expiré.
- 13 secondes supplémentaires sont nécessaires au noeud mobile pour découvrir les périphériques environnants.
- 2 secondes par requête SDP sont ensuite observées pour permettre au client de s'assurer que le périphérique découvert propose le bon service.

Si le périphérique propose le service souhaité, le client tente une connexion à ce dernier, sinon, le client passe au périphérique suivant découvert. La durée pendant laquelle le noeud mobile ne peut pas communiquer au niveau IP est un peu plus longue, car la fonctionnalité point d'accès pour ce noeud mobile prend environ 6 secondes à se stabiliser. Après ce délai, le point d'accès est capable de retransmettre les trames à destination du noeud mobile.

Dans un second test, nous avons forcé la déconnexion du client à son point d'accès courant. On s'affranchit ainsi des 20 secondes nécessaires à la détection de perte de connexion pour directement passer dans la phase de découverte des voisins. On observe un temps de reconnexion de 9 secondes auxquelles s'ajoutent les 6 secondes nécessaires à la stabilisation du pont (voir figure 2.9(b)).

En outre, à chaque handover (c'est-à-dire à chaque déconnexion puis reconnexion), l'interface réseau (généralement appelé *bnep0* dans le système Linux) est détruite et re-créée. Les communications TCP perdent donc leurs sockets, ce qui pose un réel problème pour le maintient des communications. Il serait alors nécessaire de mettre en place une interface virtuelle pour cacher la destruction de l'interface réseau. Les sockets TCP pourraient alors être ouvertes sur cette interface virtuelle constamment présente dans le système.

2.4 Les réseaux cellulaires

Les réseaux cellulaires sont apparus au début des années 1980 par la création d'un groupe de travail appelé Group Spéciale Mobile (GSM), rebaptisé plus tard en *Global System for Mobile Communications*, en charge de la spécification d'un système de normes européennes pour les radio-communications. Les réseaux cellulaires représentent dans une certaine mesure l'abstraction de fils dans la téléphonie fixe. La téléphonie sans fil a connu un véritable engouement en une decennie à peine : alors qu'en 1991 le premier appel entre un abonné fixe et un téléphone cellulaire confirmait la faisabilité du système, aujourd'hui en 2004, les abonnés aux services de la téléphonie cellulaire se comptent par millions. Avec le développement parallèle de l'Internet, les normes évoluent pour supporter le transfert de paquets et à terme offrir un réseau à 2 Mb/s avec l'UMTS [182, 131, 85, 44]. Cette section

présente brièvement les normes de réseaux cellulaires, à savoir GSM, GPRS, et UMTS.

2.4.1 La norme GSM

Les réseaux de type GSM [74, 93, 75] sont des réseaux complètement autonomes. Ils sont interconnectables aux RTCP (Réseaux Terrestres Commutés Publics) et utilisent le format numérique pour la transmission des informations, qu'elles soient de type voix, donnée ou signalisation. Les équipements spécifiques, constituant le squelette matériel d'un réseau GSM, dialoguent entre eux en mettant en oeuvre les mêmes principes que ceux utilisés dans le RNIS (Réseaux Numérique à Intégration de Services), à savoir une architecture en couches (couches 1 à 3 du modèle OSI), utilisation de liaisons sémaphores pour la signalisation et codage MIC (Modulation par Impulsion et Codage).

Un réseau GSM est composé de 3 sous-ensembles. Le sous système radio, appelé *Base Station Sub-system* (BSS) assure et gère les transmissions radios. Le sous système d'acheminement, *Network Sub System* (NSS) comprend l'ensemble des fonctions nécessaires pour les appels et la gestion de la mobilité. Le sous système d'exploitation et de maintenance, *Operation Sub-System* (OSS) qui permet à un opérateur d'exploiter son réseau.

Structure du réseau

La structure du réseau GSM est illustrée sur la figure 2.10. Le poste d'un abonné, aussi appelé *station mobile*, lui permet de se connecter au réseau. Une station mobile est à la fois un poste téléphonique sans fil et un terminal de données qui transmet et reçoit des messages du réseau. La *Base Tranceiver Station* (BTS) est l'équipement terminal du réseau vers les stations mobiles. Une BTS est un groupement d'émetteurs et de récepteurs fixes. Elle échange des messages avec les stations mobiles présentes dans la cellule qu'elle contrôle. La BTS utilise des canaux radio différents selon le type d'informations échangées et selon le sens de l'échange.

L'élément suivant du réseau est le contrôleur de station de base, appelé *Base Station Controler* (BSC). Il dialogue avec une ou plusieurs BTS. Cet équipement est à la fois un concentrateur du trafic issu des stations de base et une passerelle vers le sous système réseau.

Derrière le BSC se trouve le commutateur du réseau GSM appelé *Mobile Switching Centre* (MSC). D'une part il interconnecte un réseau GSM avec le réseau téléphonique public RTCP/RNIS, et d'autre part, il est l'interface des bases de données du réseau GSM avec le sous-système radio. Ces bases de données, outre le fait qu'elles permettent de contrôler les droits d'accès des usagers au réseau, enregistrent la localisation des abonnés. Il y a trois bases de données différentes : la *Visitor Location Register* (VLR), la *Home Location Register* (HLR), et le *Authentication Centre* (AUC).

La base de données VLR stocke des informations se rapportant à des abonnés qui sont en transit. La HLR d'un abonné renferme les informations originales relatives à cet abonné, notamment le profil de son abonnement. Quand cet abonné entre dans le réseau ou quand il demande l'accès à un service, ses privilèges sont contrôlés en interrogeant la HLR propre à cet abonné. Les informations contenues dans la HLR sont donc permanentes alors que celles de la VLR sont temporaires.

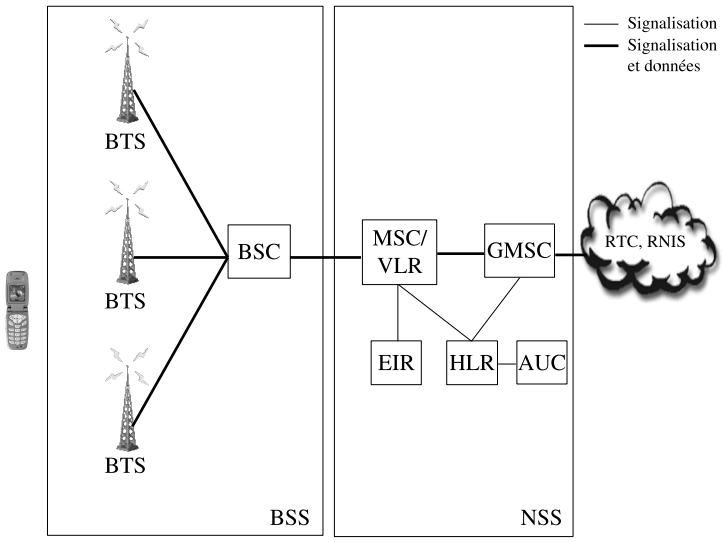


FIG. 2.10 – Architecture du réseau GSM

La mobilité

La gestion de la mobilité des stations mobiles est mise en place par une communication entre la structure fixe du réseau et le mobile. La décision d'effectuer un basculement de fréquence nécessaire au traitement d'un handover reste toutefois à la charge des équipements fixes (MSC et BSC). Cette décision découle des mesures

effectuées par la station mobile des balises envoyées par les BSC environnants, qui sont rapportées au BSC courant. Si plusieurs cellules sont candidates, le MSC décidera de la cellule cible en fonction de la station mobile et de son trafic. Le handover s'effectue avec coupure de la communication (imperceptible pour l'utilisateur). La figure 2.11 illustre un scénario de handover sans changement de VLR.

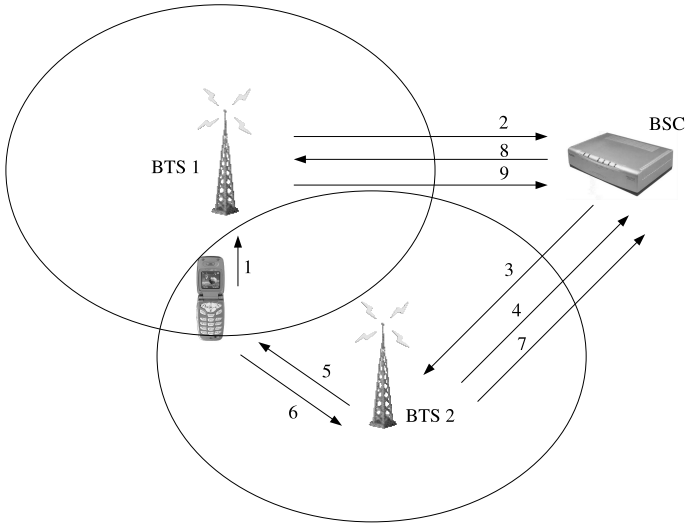


FIG. 2.11 – Scénario de handover sans changement de VLR dans un réseau GSM

GSM et les données

Les réseaux GSM sont basés sur la commutation de circuit, c'est-à-dire qu'un canal de communication est occupé par un utilisateur pendant toute la durée d'une conversation. Ce mode de transmission est tout à fait adapté au transport de la voix qui est continu, mais ne convient pas à la circulation de données qui sont sporadiques. Pour offrir des services de données de meilleure qualité et plus compétitifs, l'ETSI [2] recommande l'intégration des techniques de transmission par paquet dans ses spécifications GSM 2+. C'est alors qu'une nouvelle technologie appelée le GPRS fait son apparition dans les réseaux GSM.

[2]ETSI - Institut Européen de Normalisation des Télécommunications

2.4.2 Les réseaux GPRS

Le GPRS (*General Packet Radio Services*) ne constitue pas à lui tout seul un réseau mobile à part entière, mais est une couche supplémentaire rajoutée à un réseau GSM existant. Conçu pour réutiliser au maximum les infrastructures GSM existantes, le déploiement du GPRS nécessite la mise en place d'une infrastructure réseau basée sur la commutation de paquets et l'introduction de passerelles pour s'adosser aux réseaux GSM existants.

Le GPRS, aussi appelé GSM 2+, repose sur la transmission en mode paquet. Ce principe permet d'affecter à d'autres communications les temps morts d'une première communication (attente d'une réponse à une requête Internet par exemple). La technologie GPRS permet d'accéder aux services Internet avec un débit efficace maximum de 115 kb/s, contre 9,6 kb/s pour le GSM, et ceci grâce à l'utilisation de multiples canaux radio qui sont attribués à chaque utilisateur ou bien partagés entre plusieurs utilisateurs. Les ressources radio sont allouées dynamiquement et la vitesse de transmission varie beaucoup du fait de la souplesse et de l'adaptabilité du mode de transmission par paquet.

Efficace pour les transmissions de données discontinues ou les transmissions fréquentes de petits volumes de données, le GPRS convient également aux transmissions ponctuelles de gros volumes de données. Il ouvre ainsi la voie des applications de bureautique mobile, comme le courier électronique, l'accès à Internet, etc.

Structure d'un réseau GPRS

Les stations de base ne subissent aucune modification, si ce n'est l'adjonction d'un logiciel spécifique. Plus en amont, le contrôleur de stations de base doit être doublé par un contrôleur de paquets PCU (Packet Controler Unit). Vient ensuite la chaîne destinée aux données par paquet, consituée du commutateur ou switch spécifique GPRS appelé *Serving GPRS Support Node* (SGSN), équivalent du MSC, contrôleur qui a pour fonction de vérifier l'enregistrement des abonnés, de les authentifier et d'autoriser les communications, et du module d'accès au monde IP (Internet ou Intranet) appelé *Gateway GPRS Support Node* (GGSN). Une troisième entité, le *Border Gateway* (BG) joue un rôle supplémentaire de sécurité. Elle permet de connecter les réseaux GPRS via un réseau fédérateur et assure les fonctions de sécurité entre ces réseaux. Ces BG ne sont néanmoins pas spécifiées par les recommandations mais elles jouent le rôle d'interface avec les autres PLMN (*Public Land Mobile Network*) permettant ainsi de gérer les niveaux de sécurité

entre les réseaux (entre deux réseaux de deux opérateurs concurrents par exemple). Toutes ces entités sont représentées dans la figure 2.12.

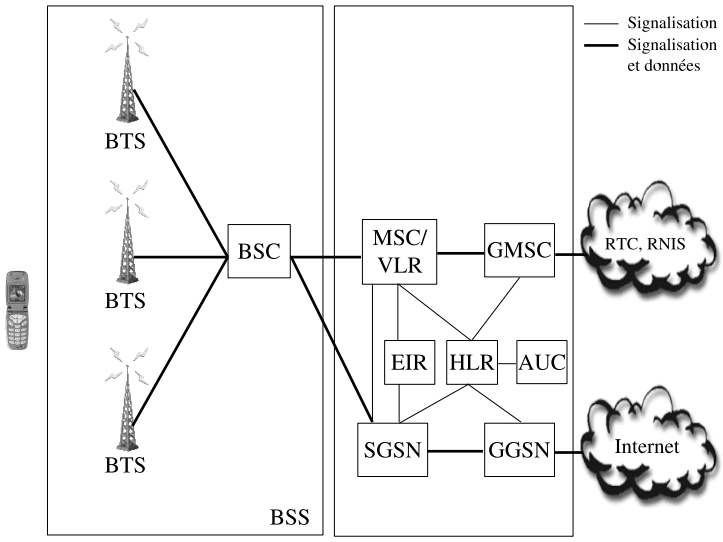


FIG. 2.12 – Architecture du réseau GPRS

Le GGSN permet la communication par paquet avec d'autres réseaux extérieurs au réseau GSM. Il masque au réseau de données les spécificités du GPRS. De plus, il gère la taxation des abonnés du service, et doit supporter le procotole utilisé sur le réseau de données avec lequel il est interconnecté. Les protocoles de données supportés en standard par un GGSN sont IPv4, IPv6, CLNP et X25.

Le SGSN permet de gérer les services offerts à l'utilisateur. Il est l'interface logique entre l'abonné GSM et un réseau de données externe. Ses missions principales sont, d'une part, la gestion des abonnés mobiles actifs, et d'autre part, le relais des paquets de données. Quand un paquet de données arrive d'un *Packet Data Network* (PDN) externe au réseau GSM, le GGSN reçoit ce paquet et le transfert au SGSN qui le retransmet vers la station mobile. Pour les paquets sortant, c'est le SGSN qui les transmet vers le GGSN.

La mise en place d'un réseau GPRS permet donc à un opérateur de proposer de nouveaux services de type données à ses abonnés avec un débit 5 à 10 fois supérieur au débit maximum théorique d'un réseau GSM. Cependant, les réseaux GPRS ne sont qu'une étape vers les réseaux de troisième génération, à savoir les réseaux UMTS.

2.4.3 Les réseaux UMTS

L'UMTS (*Universal Mobile Telecommunications System*) est la version européenne définie par l'ETSI de la troisième génération des services mobiles (3G). L'UMTS devrait délivrer des débits compris entre 384 kb/s et 2 Mb/s. Cette norme fait partie du projet IMT-2000 (*International Mobile Telecommunication System 2000*) défini par l'UIT (Union Internationale des Télécommunications). Celui-ci a pour but de normaliser les systèmes de télécommunications mobiles de troisième génération qui assureront l'accès radioélectrique à l'infrastructure mondiale des télécoms, dans un contexte mondial d'itinérance. Il doit faire intervenir aussi bien les systèmes satellitaires que les moyens terrestres desservant les usagers fixes et mobiles de réseaux publics et privés.

La mise en place d'un réseau UMTS va permettre à un opérateur de compléter son offre existante par l'apport de nouveaux services en mode paquet complétant ainsi les réseaux GSM et GPRS. Le réseau UMTS vient donc se combiner aux réseaux GSM / GPRS existants comme illustré dans la figure 2.13. Ces derniers apportant des fonctionnalités respectives de Voix et de Données, le réseau UMTS apportera, lui, les fonctionnalités multimédia.

Structure d'un réseau UMTS

La mise en place d'un réseau UMTS nécessite l'introduction de nouveaux éléments (voir figure 2.13), comme les Nodes B. Ils sont au réseau UMTS ce que les BTS sont au réseau GSM. Les Nodes B gèrent la couche physique de l'interface radio, régissent le codage du canal, l'entrelacement, l'adaptation du débit et l'étalement. Le RNC (*Radio Network Controller*) est un contrôleur de Node B. Ici encore, le RNC est l'équivalent du BCS dans le réseau GSM. Il contrôle et gère les ressources radio en utilisant le protocole RRC (*Radio Ressource Control*) pour définir les procédures et les communications entre le mobile (par l'intermédiaire des Nodes B) et le réseau. Il s'interface avec le réseau pour les transmissions en mode paquet et en mode circuit. L'ensemble des Nodes B et des RNC constitue l'équivalent de la sous architecture BSS vue précédemment en réseau GSM. En UMTS, on parlera de sous-architecture UTRAN.

Le réseau coeur de l'UMTS s'appuie sur les éléments de base des réseaux GSM et GPRS. Il est en charge de la commutation et du routage des communications (voix et données) vers les réseaux externes. Il se décompose en deux parties : le domaine circuit et le domaine paquet. Le domaine circuit permet de gérer les services temps réel. De telles applications nécessitent un faible délai. L'infrastructure

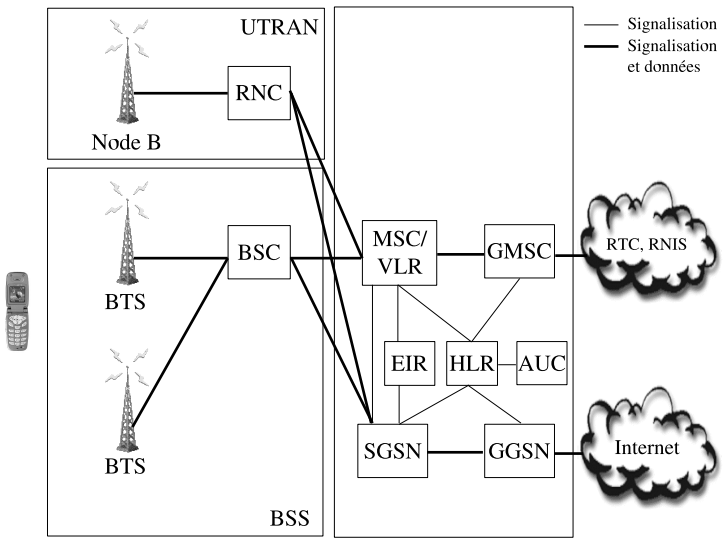


FIG. 2.13 – Architecture du réseau UMTS

s'appuiera alors sur les principaux éléments du réseau GSM : les MSC/VLR et le GMSC, afin d'avoir une connexion directe vers le réseau externe.

Le domaine paquet permet de gérer les services non temps réel. Il s'agit principalement de la navigation sur Internet et de la messagerie électronique. L'infrastructure s'appuiera sur les principaux éléments du réseau GPRS : SGSN et GGSN qui joueront le rôle de commutateurs vers le réseau Internet et vers les autres réseaux publics ou privés de transmission de données.

Migration vers le tout IP

A terme, l'objectif est de migrer le coeur du réseau UMTS vers une solution complète IP à condition d'apporter des solutions aux problèmes du protocole IP en terme de qualité de service (en particulier sur des temps de transfert convenables). Lorsque les applications de type voix pourront être transmises par le biais du protocole IP, les opérateurs migreront vers un réseau unique où les domaines circuit et paquet seront réunis.

Le réseau UMTS est donc complémentaire aux réseaux GSM et GPRS. Le réseau GSM couvre les fonctionnalités nécessaires aux services de type voix dans un mode circuit, le GPRS apporte les premières fonctionnalités à la mise en place

de services de type données dans un mode paquet, et l'UMTS vient compléter ces deux réseaux par une offre de service voix et données complémentaires dans un mode paquet.

2.5 Conclusion

	IEEE 802.11	**Bluetooth**	**Réseaux cellulaires**
Débit théorique	De 1 à 54 Mb/s	1 Mb/s	De 9,6 kb/s à 2 Mb/s
Débit utile	De 40 ko/s à 20 Mb/s	500 kb/s	De 5 kb/s à 30 kb/s[a]
Portée	De 5 à 100 mètres	De 5 à 15 mètres	Quelques kilomètres
Handover	250 ms	30 s	Pas de résultat
Utilisation	Internet multimédia	Réseau personnel	Messagerie (GPRS)

[a]Pour les réseaux GPRS, étant donné que l'UMTS n'a pas encore été déployé.

TAB. 2.2 – Récapitulatifs des technologies présentées

Nous venons d'étudier trois des principales normes de communication sans fil supportant le transport de paquets IP. Cette étude nous a clairement révélé que ces technologies offrent des performances bien distinctes. Le tableau 2.2 retrace les principales caractéristiques de chacune d'entre elles. La série des normes IEEE 802.11 [2, 4, 3, 6] est très populaire, et se voit développée un peu partout dans le monde dans des "Hot Spot" ou pour assurer l'accès Internet à domicile. En Corée, des bornes IEEE 802.11b ont même été déployées dans tout le pays pour fournir une couverture quasi-totale des zones habitées [94]. Ce succès provient non seulement du faible coût d'achat et d'utilisation, mais également de la simplicité du protocole et des bonnes performances en terme de débit et de temps de handover. La version b de la norme proposait déjà un débit brut maximum de 11 Mb/s (mesuré à 5,4 Mb/s utile), et les versions a et g ont porté le débit à 54 Mb/s. Cependant, l'inconvénient principal est la portée des équipements. Le nœud mobile doit être relativement proche de son point d'accès, de quelques dizaines de mètres en intérieur à une centaine de mètres en extérieur.

D'un autre côté, la norme Bluetooth [36, 8] qui semblait si prometteuse lors de sa conception, montre des capacités limitées : faible portée (une dizaine de mètres), faible débit (500 kb/s mesurés), mauvaise performance de handover (autour de 30 secondes) et une qualité de signal très variable. Ces faibles performances proviennent notamment du fait que Bluetooth n'a pas été développé spécifiquement pour les communications IP, mais pour des applications propres avec l'élaboration d'une pile de communication en 5 couches.

La dernière technologie de communication qui a été présentée est l'ensemble des systèmes cellulaires. En effet, les réseaux cellulaires ont tendance à évoluer vers des communications IP pour rejoindre le monde de l'Internet. Ce domaine est en plein développement, avec aujourd'hui la mise à disposition du service GPRS et le développement de l'UMTS. Bien que les débits restent assez faibles, 2 Mb/s théorique au maximum, la zone de couverture et le bon fonctionnement de ces technologies ont fait le succès qu'on leur connaît.

A travers ces analyses, on se rend compte qu'il est peu probable qu'une seule technologie s'impose sur le marché. A contrario, on imagine plutôt une disponibilité alternative de ces technologies, avec des équipements multi-modes. Chaque technologie offre un mode d'utilisation différent, aussi bien au niveau des applications que du modèle de mobilité. La clé de l'optimisation de la connectivité des nœuds mobiles sera alors le degré d'intégration de l'ensemble de ces technologies et les possibilités de répartition de flux sur chacune d'entre elles.

Maintenant que nous avons une connaissance plus approfondie des normes de communications disponibles au niveau 2, nous allons nous intéresser aux performances des protocles de la couche immédiatement supérieure, à savoir la couche IP avec l'évaluation de Mobile IPv6 et ses principales optimisations.

Chapitre 3

Evaluation de Mobile IPv6 et ses optimisations

3.1 Introduction

Jusqu'à présent nous avons présenté les solutions de gestion de la mobilité de niveau 3 (voir chapitre 1) et les principales technologies sans fil permettant une telle mobilité (voir chapitre 2). Mobile IPv6 est le protocole de gestion des handovers choisi comme le protocole au centre de nos développements et propositions. Or nous avons vu dans la section 1.5 qu'un certain nombre d'extensions à Mobile IPv6 ont déjà été proposées en vue de réduire le nombre de paquets perdus et le temps d'interruption, occasionés lors de déplacement d'un noeud mobile dans différents sous-réseaux IPv6. En effet, une évaluation de ces différentes méthodes pourra nous apporter une meilleure compréhension des enjeux liés à la gestion des déplacements. Ce chapitre est donc consacré à l'évaluation aussi bien théorique, que pratique de Mobile IPv6 et ses extensions majeures, à savoir Fast Mobile IPv6 [201] et Mobile IPv6 Hiérarchique [168]. Ces évaluations ont été publiées dans [119, 121].

Les travaux de Costa, Schmitz, Hartenstein et Liebsch [48] tentent de répondre à la question de la combinaison de plusieurs solutions pour une gestion optimale de la mobilité. Mais ces travaux sont basés sur une approche analytique du problème, éloignée des préoccupations en terme de mise à l'échelle, de bande passante, etc. Afin d'évaluer les performances du protocole Mobile IPv6, nous avons procédé en deux étapes. La première étape consiste à réaliser une étude théorique des résultats attendus de Mobile IPv6. Les critères d'évaluation retenus sont les suivants :

- Utilisation ou non des mécanismes de détection de duplication d'adresses (DAD)
- Position sur l'Internet de l'agent mère par rapport aux noeuds mobiles (agent mère dans le même pays, sur le même continent, sur des continents différents)
- Fréquence d'émission des avertissements des routeurs (RA)
- Débit offert sur le lien sans fil

Dans un premier temps, nous avons réalisé une étude théorique des temps de handover selon les différentes optimisations présentées dans la section 1.5. Cette étude nous semblait importante étant donné que le déploiement d'une plate-forme de test à grande échelle (plusieurs déplacements consécutifs, distance / délai variables avec les correspondants) est difficile alors qu'une étude théorique au préalable permet de cibler une certaine catégorie de tests. Ensuite, nous présentons une évaluation de Mobile IPv6 que nous avons pu réaliser sur la plate-forme mise en place pour le déploiement de WLAN IPv6 au sein du campus universitaire de Strasbourg [123, 142].

3.2 Evaluation théorique

Dans cette section, une étude théorique des protocoles Mobile IPv6 [84], Fast Mobile IPv6 [201] et Mobile IPv6 Hiérarchique [168] est présentée. L'objectif de cette étude est d'identifier les points forts de ces méthodes et de rendre compte des impacts de ces dernières sur la continuité des communications des noeuds mobiles.

Cette étude théorique a été réalisée en calculant la taille des différents messages utilisés dans les protocoles, puis en testant les temps d'aller-retour vers différentes positions géographiques dans le monde. Les moyennes des temps observés ainsi que les acronymes que nous utiliserons dans cette partie sont donnés dans le tableau 3.1. Les détails du calcul des tailles des messages et délais dans l'Internet IPv6, ainsi que l'ensemble des résultats peuvent être trouvés dans [116].

Acronyme	**Signification**	**Délai mesuré**
Local	A/R sur un réseau local	10 ms
National	A/R entre deux réseaux d'un même pays	100 ms
Eu-Eu	A/R entre deux villes d'Europe	250 ms
Eu-USA	A/R entre l'Europe et les Etats-Unis	500 ms
Eu-Asie	A/R entre l'Europe et l'Asie	650 ms

TAB. 3.1 – Signification des acronymes utilisés et temps d'aller-retour mesurés

3.2.1 Fast Mobile IPv6

Fast Mobile IPv6 [201] propose une solution de handover rapide par anticipation de mouvements et échange entre l'ancien et le nouveau routeur d'accès du noeud mobile (cf. section 1.5.2). Le fait que le handover est principalement géré localement à la position courante du noeud mobile apparaît clairement dans la figure 3.1 : le temps de latence pour le protocole Fast Mobile IPv6 reste constant quelle que soit le délai entre le noeud mobile et son correspondant. La position du correspondant n'intervient pas dans les calculs puisque l'ancien routeur d'accès est chargé de rediriger les paquets pour le noeud mobile à sa nouvelle localisation. Au contraire, le temps de latence observé pour Mobile IPv6 est fortement dépendant du délai entre le noeud mobile et son correspondant, ce qui augmente les délais de réception des paquets à la nouvelle localisation.

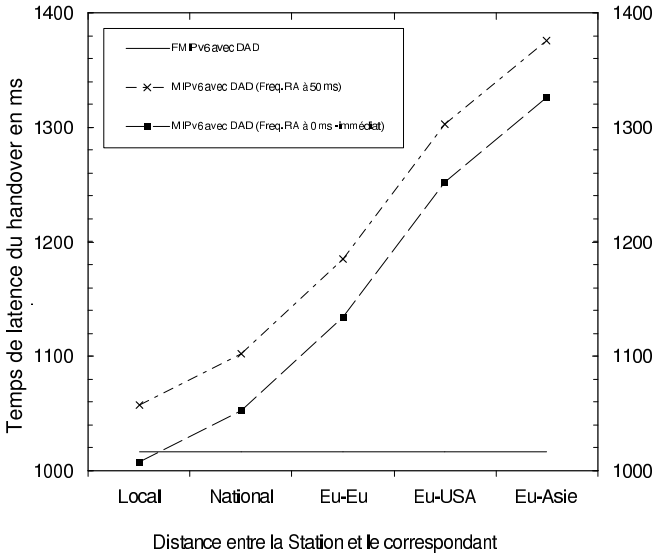


FIG. 3.1 – Comparaison des temps de latence en Mobile IPv6 et Fast Mobile IPv6

Les temps de latence introduits par Mobile IPv6 sont fortement dépendants du moment de détection du changement de sous-réseau. Cette détection passe par la réception d'un nouveau Router Advertisement (RA). C'est seulement après réception de celui-ci que le mobile pourra déclencher les opérations de mise à jour. Dans le cas de Fast Mobile IPv6 (handover contrôlé par le mobile), le noeud mobile détecte qu'il est sur le point de changer de routeur d'accès au travers des messages des couches inférieures (cf. section 6.3.1). Ces messages permettent une détection de mouvement quasi immédiate mais incertaine et un mobile peut, de ce fait, déclencher une procédure de déplacement anticipée.

On pourra noter que lorsque le correspondant d'un noeud mobile est situé localement, le temps pris en compte dans le calcul du temps d'interruption pour la mobilité IPv6 est légèrement meilleur que celui calculé dans Fast Mobile IPv6 (1007,5ms contre 1016,8ms). Ceci est notamment dû aux hypothèses retenues lors de nos tests. En effet, dans le cas de Fast Mobile IPv6, il existe énormément de scénarios possibles. Nous avons considéré que le noeud mobile ne pouvait plus recevoir de trafic à partir du moment où il déclenchait la procédure de « Fast Handover ». Cependant, on peut très bien imaginer que le noeud mobile puisse rester rattaché à son ancien point d'accès et recevoir son trafic le temps que le protocole valide sa nouvelle adresse. Le temps de latence calculé est alors la somme du temps de changement de point d'accès (niveau 2), additionné au temps de réception du message de prise en compte d'un déplacement sur le nouveau routeur d'accès. Dans ce cas (le plus optimiste), la coupure sera très courte et le temps de latence très faible par rapport à celui de Mobile IPv6.

En conclusion, cette première série de tests montre que la mobilité anticipée (Fast Mobile IPv6) offre de meilleurs résultats que la mobilité IPv6 quel que soit le cas de figure considéré (à quelques millisecondes près pour des cas extrêmes). Fast Mobile IPv6 résout le problème de la détection du changement de sous-réseau et accélère l'établissement de la nouvelle adresse temporaire. De plus, même si l'on peut observer une interruption, la majorité des paquets reçus pendant ce laps de temps est transmise de l'ancien routeur d'accès au nouveau qui peut les mettre en tampon et les transmettre au mobile dès sa reconnexion.

3.2.2 Mobile IPv6 Hiérarchique

Nous avons poursuivi nos évaluations de la gestion de la mobilité dans l'Internet Nouvelle Génération en cherchant à déterminer si la mobilité hiérarchique apportait de nouvelles réponses à certains des cas de figure laissés en suspens dans la mobilité IPv6 [128]. La mobilité IPv6 hiérarchique a pour objectif de diminuer la charge et les délais de signalisation engendrés par un déplacement. Le noeud mobile peut faire soit un enregistrement global, lors de l'entrée dans un domaine, soit un enregistrement régional, lors de ses déplacements à l'intérieur d'un même domaine. La figure 3.2 montre les différents temps de latence pour un enregistrement « classique » dans Mobile IPv6 et les deux enregistrements supportés dans Mobile IPv6 Hiérarchique. Les tests ont été effectués pour différents délais entre le noeud mobile et son correspondant. Nous avons retenu différentes fréquences d'émission des RA afin d'évaluer l'impact du bon choix de cette valeur pour la prise en compte de la mobilité des équipements.

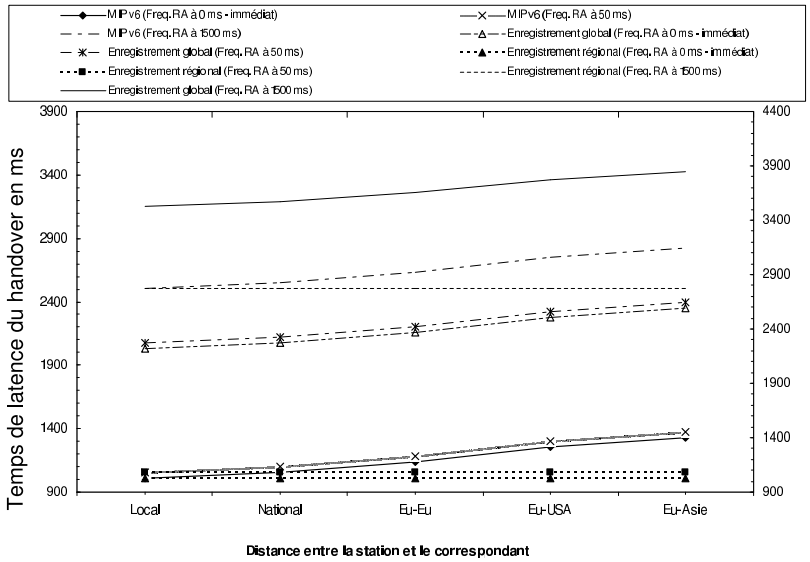


FIG. 3.2 – Comparaison des temps de latence de handover de niveau 3 avec Mobile IPv6 et Mobile IPv6 Hiérarchique

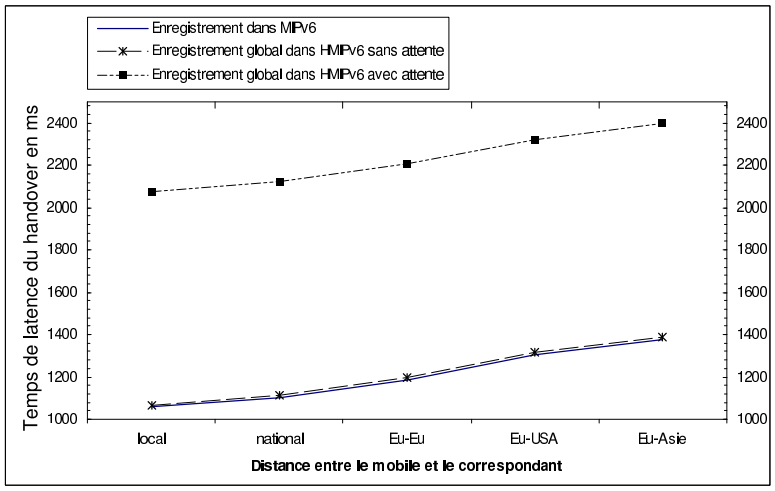


FIG. 3.3 – Comparaison des temps de latence du handover de niveau 3 avec Mobile IPv6 et Mobile IPv6 Hiérarchique en fonction de la position du correspondant

On peut également constater sur la figure 3.2 que, quelle que soit la position du correspondant, le temps de latence avec Mobile IPv6 Hiérarchique reste constant (égal à un enregistrement local) alors que dans le cas de la mobilité IPv6 cet enregistrement varie fortement selon le délai : ce qui souligne d'avantage l'intérêt du

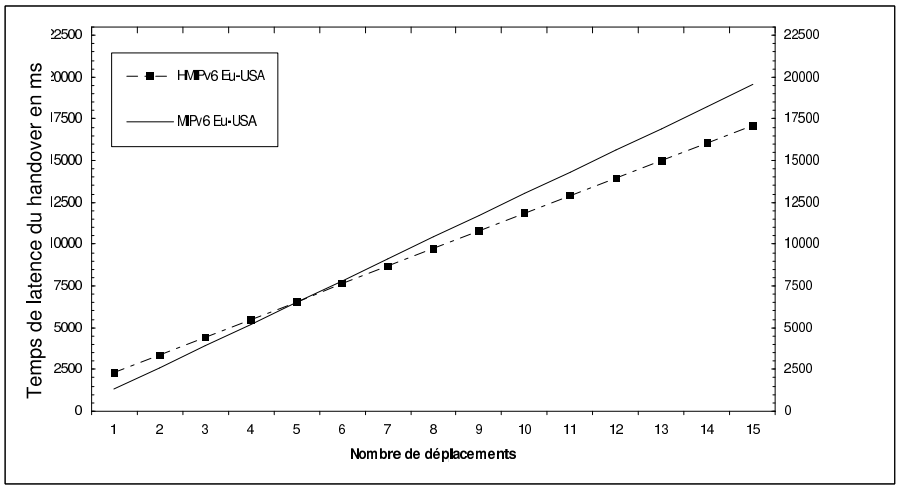


FIG. 3.4 – Comparaison du gain de la mobilité hiérarchique par rapport à la mobilité IPv6 en fonction du nombre de déplacements

protocole Mobile IPv6 Hiérarchique. Par contre, on peut remarquer que l'enregistrement global dépend du délai entre le noeud mobile et son correspondant et qu'il prend plus du double de temps par rapport à Mobile IPv6. Cela découle de l'hypothèse de départ qui considère qu'un noeud mobile doit attendre l'acquittement du point d'ancrage avant d'envoyer un message de localisation à son correspondant. Toutefois, dans les spécifications de la mobilité hiérarchique, cette attente est optionnelle. Aussi, nous avons réalisé de nouvelles évaluations qui permettent de comparer un enregistrement basique avec deux enregistrements globaux dans Mobile IPv6 Hiérarchique (avec et sans attente d'acquittement) (cf. figure 3.3). Nous pouvons constater que lorsqu'un noeud mobile n'attend pas d'acquittement, le temps d'enregistrement global est à peine supérieur à celui de Mobile IPv6. Il reste supérieur puisque le mobile doit tout de même envoyer un message de contrôle supplémentaire (un message à destination du point d'ancrage et un autre à destination de son correspondant). L'attente de l'acquittement engendre donc un coût supplémentaire non négligeable, mais elle permet de s'assurer que l'adresse globale qui sera enregistrée en tant qu'adresse temporaire sur les correspondants et l'agent mère est bien valide. Si le noeud mobile enregistrait une mauvaise adresse, il devrait ensuite réémettre un message pour corriger la mauvaise association et le temps d'enregistrement final serait excessivement long.

Enfin, on remarque dans la figure 3.4, que c'est au bout de cinq déplacements à l'intérieur d'un domaine que le cumul du temps de handover devient inférieur pour la mobilité hiérarchique. Il convient toutefois de se rappeler que dès que le mobile a réalisé un enregistrement global dans le domaine, chaque déplacement

local est plus rapide que pour Mobile IPv6. Finalement, Mobile IPv6 Hiérarchique permet de réduire la latence du déplacement une fois qu'un noeud mobile s'est enregistré dans un domaine. Pour rappel, si ce dernier a plusieurs centaines de correspondants (i.e optimisation de routage), il pourra envoyer un message de localisation à chacun d'entre eux, d'après les spécifications de la mobilité IPv6. Le temps cumulé de l'envoi de ces messages peut être relativement pénalisant.

3.2.3 Comparaison du temps de latence des différents protocoles suivant le débit offert

Il nous a paru intéressant d'évaluer les différences de comportement des protocoles pour différentes valeurs de débit. En effet, jusqu'à présent toutes nos études ont été faites avec comme hypothèse l'utilisation d'un débit garanti de 30 Ko/s entre le mobile et son routeur d'accès. Le choix de cette valeur fut notamment dicté par les résultats des études que nous avons pu mener sur la norme IEEE 802.11b [119, 121, 123]. Cependant, comme nous l'avons énoncé dans la partie précédente, il est fort probable que la mobilité IPv6 s'étende à de nombreuses normes comme GPRS [1] qui offriront potentiellement des débits inférieurs par mobile.

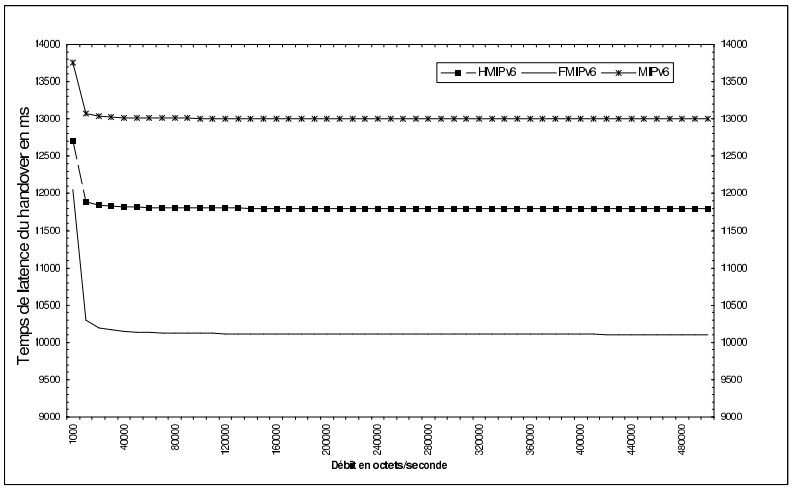


FIG. 3.5 – Comparaison des temps d'interruption des protocoles Mobile IPv6, Mobile IPv6 Hiérarchique, Fast Mobile IPv6 en fonction du débit

La figure 3.5 représente la somme des temps de coupures observés par un mobile suite à dix déplacements consécutifs dans un même domaine. L'évaluation a porté sur les protocoles Mobile IPv6, Mobile IPv6 Hiérarchique et Fast Mobile IPv6

(pour un noeud mobile en Europe et un correspondant aux États-Unis). Dans le cas de la mobilité IPv6 hiérarchique, le premier déplacement correspond à un enregistrement global et neuf enregistrements locaux. Pour la mobilité IPv6 et le protocole de déplacement anticipé, le temps pris pour chaque déplacement est le même. Un point de la courbe est donc la somme de dix temps de latence.

Ces trois courbes sont bien entendu décroissantes puisque le temps de coupure diminue quand le débit augmente. Cependant, à partir d'un débit de 10 Ko/s, la part principale du temps de latence est due, non pas à la limitation de la bande passante, mais au temps de détection de mouvement ainsi qu'au temps de détection de duplication d'adresses (DAD). Ce sont donc ces deux facteurs qui sont restrictifs. La limitation induite par le temps de détection de déplacement est résolue à l'aide du protocole Fast Mobile IPv6, puisqu'un noeud mobile détecte le changement de sous-réseau par les messages de la couche 2. En terme de comparaison, on remarque que Fast Mobile IPv6 offre les meilleurs délais, suivi de Mobile IPv6 Hiérarchique et finalement Mobile IPv6. Dans le cas de Fast Mobile IPv6, un noeud mobile « retrouve » rapidement ses communications grâce au transfert des paquets réalisé temporairement par l'ancien routeur d'accès.

3.2.4 Conclusion

En conclusion, les différentes évaluations réalisées montrent que le temps de détection de changement de sous-réseau dans Mobile IPv6 et Mobile IPv6 Hiérarchique peut atteindre un délai important, jusqu'à 1500ms (une détection de duplication d'adresse (DAD) peut prendre plus de 1000ms). Il est également apparu que Fast Mobile IPv6 offre de meilleurs résultats quelle que soit la localisation des correspondants des noeuds mobiles. Il existe même des cas de figure où la seule coupure que le noeud mobile observera est la coupure de niveau 2. D'un autre côté, Mobile IPv6 hiérarchique offre de meilleurs résultats que la mobilité IPv6 classique une fois qu'un mobile s'est enregistré dans le domaine.

Cette première étude est intéressante dans le sens où elle nous suggère plusieurs interrogations. Tout d'abord, dans le cas de Fast Mobile IPv6, on peut se demander dans quelle mesure un mobile pourra rester rattaché à son ancien point d'accès. Ceci dépendra fortement du modèle de mobilité à prendre en compte. D'autre part, le temps nécessaire pour exécuter la détection de duplication d'adresses est une part importante des temps de latence des trois protocoles. C'est peut-être dans cette voie qu'il faut trouver des solutions. De plus, dans Mobile IPv6 et Mobile IPv6 Hiérarchique, le temps de détection de changement de sous-réseau peut également être pénalisant pour les équipements mobiles. Le Fast Handover résout ce

problème et on pourrait alors envisager de faire des Fast Handover dans un domaine. Par contre, d'autres tests restent encore envisageables, comme l'évaluation des cas d'erreur : que se passe-t-il si une mauvaise anticipation est faite dans Fast Mobile IPv6 ? Ou, dans le cadre de la mobilité IPv6 hiérarchique, quelles sont les performances lorsque le noeud mobile n'attend pas l'acquittement du point d'ancrage ? Ou encore comment évaluer les performances lors d'un déploiement à grande échelle ?

3.3 Evaluation pratique

La seconde étape de notre étude du comportement des spécifications de la mobilité dans l'Internet Nouvelle Génération visait à aller au-delà de l'étude théorique afin de valider nos résultats sur une plate-forme d'expérimentation réelle. Nous avons cherché à dépasser la plate-forme de laboratoire afin d'avoir un réseau sans fil plus réaliste et à plus grande échelle. La mise en place d'un tel réseau s'est concrétisée par un projet réalisé en collaboration avec France Télécom R&D. Ce projet inclut, dans une première phase, la mise à disposition d'un réseau sans fil de type IEEE 802.11b supportant exclusivement IPv6 et Mobile IPv6 à dans une composante de l'université Louis Pasteur (l'UFR de Mathématiques et d'Informatique). Cette expérimentation nous a permis de montrer les limitations pratiques de Mobile IPv6.

3.3.1 Impact des flux applicatifs sur la gestion de la mobilité IPv6

La norme IEEE 802.11b ne propose pas de mécanismes de qualité de service, ni de prioritisation de flux. De ce fait, tout paquet de niveau réseau, que ce soit un paquet de contrôle comme un Binding Update ou un paquet de données est vu de manière identique. Ce constat a automatiquement une répercussion sur les mécanismes de mobilité IPv6. En effet, les différents messages de contrôle, comme la localisation d'un mobile, transitent dans des messages IPv6 « classiques ». Aussi, une question simple se pose : « Quel est l'impact des flux applicatifs sur la gestion de la mobilité IPv6 ». Nous avons cherché à répondre à cette question cruciale qui peut se traduire autrement : « Les spécifications de la mobilité IPv6 sont-elles adaptées aux caractéristiques des réseaux locaux sans fil ? ». La première série de tests que nous avons effectuée consistait à générer un trafic de type MP3 à 128 Kb/sec à destination des équipements sans fil. Les figures 3.6 et 3.7 illustrent l'im-

pact de ce type de trafic sur la prise en compte des déplacements des équipements mobiles au niveau 2 et 3.

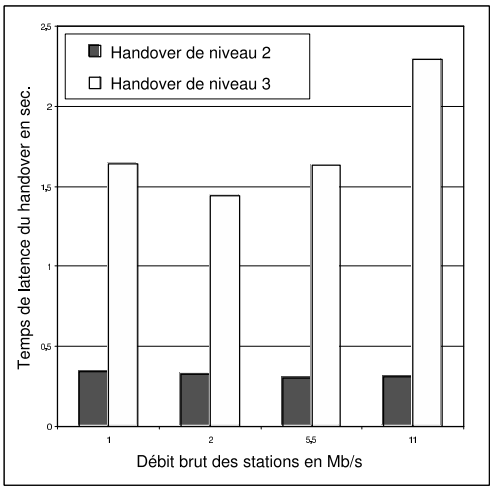


FIG. 3.6 – Impact du trafic MP3 (1 Mobile) sur les temps de déplacements

La figure 3.6 retrace l'impact du trafic lorsqu'il n'y a qu'un seul noeud mobile qui se déplace entre des points d'accès situés sur des sous-réseaux distincts. La figure 3.7 présente le même type d'impact mais avec 6 noeuds mobiles qui se déplacent simultanément. Ces résultats nous dévoilent d'une part que le handover de niveau 2 est relativement stable et dure en moyenne 320 millisecondes, comme illustré dans la section 2.2.4. Par contre, ce qui peut paraître plus surprenant est le temps du handover de niveau 3 qui est relativement « long » (en moyenne 1,653 sec) mais qui peut s'expliquer par la fréquence des Router Advertissement des routeurs IPv6. Ces derniers sont envoyés toutes les secondes, comme le proposaient les spécifications de la mobilité IPv6 (draft 13).

Cette première série de tests pourrait nous inciter à penser que bien que le temps de handover de niveau 3 soit important, il est relativement constant sur des réseaux de type IEEE 802.11b. Mais il convient de rester prudent car un simple calcul nous montre que 6 flux MP3 de 128 Kb/sec ne saturent pas un point d'accès 802.11b à 11 Mb/s. Aussi, nous avons souhaité tester des applications plus « gourmandes » en terme de bande passante. Nous avons donc testé une application IPv6 de vidéo à la demande générant un trafic oscillant entre 2,5 et 3 Mb/sec par client en unicast. Nous avons effectué des tests de mobilité IPv6 de 1 à 4 noeuds mobiles (cf. figure 3.8) pour les différents débits supportés par la norme. Les résultats obtenus sont assez intéressants puisqu'ils montrent que le handover de niveau 2 est toujours stable autour de 300 millisecondes, ce qui n'est pas le

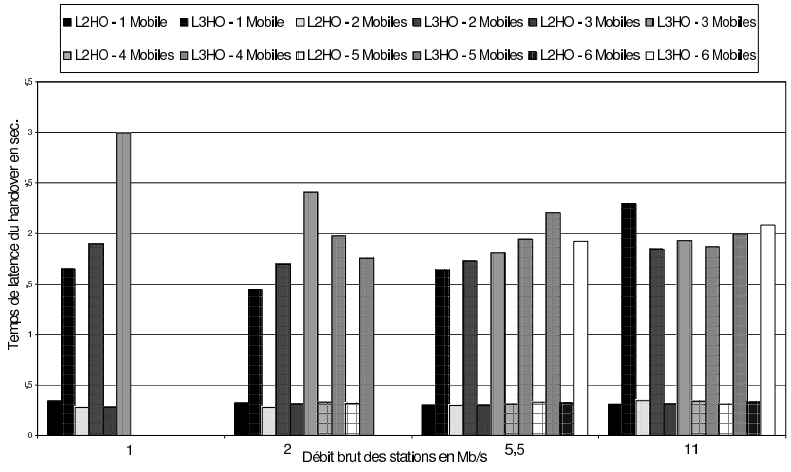


FIG. 3.7 – Impact du trafic MP3 (6 Mobiles) sur les temps de déplacements

cas du handover de niveau 3. Ce dernier peut atteindre une borne supérieure de 8 secondes. Ce résultat a une explication, lorsqu'un trafic sature le réseau sans fil, les pertes de paquets sont nombreuses et cela retarde fortement les enregistrements de niveau 3. De plus, le trafic de contrôle de niveau 802.11 a plus rapidement accès au médium que le trafic de données. En effet même si les trames de contrôle 802.11 sont échangées sur le même canal que les trames de données, elles ont un accès prioritaire par rapport aux données, ce qui n'est pas le cas des trames dites de signalisation IPv6 (*Binding Update* et *Binding Ackowledgement*).

3.4 Conclusions

La conclusion principale de ces expérimentations est la démonstration réussie en grandeur réelle de l'utilisation conjointe des WLAN, d'IPv6 et de Mobile IPv6. En effet, nous avons pu montrer que nous ne sommes plus très loin d'avoir des fonctionnalités complètes et opérationnelles. Bien sûr un certain nombre de problèmes a été identifié, comme les micro coupures lors des handovers soit de niveau 2 soit de niveau 3. Enfin, un des enseignements de cette première étude est la nécessité de modifier certaines applications pour les rendre moins sujettes aux déplacements de l'utilisateur. On peut définir trois types d'applications : les applications faiblement interactives (consultation mail, Web, transfert de fichiers,

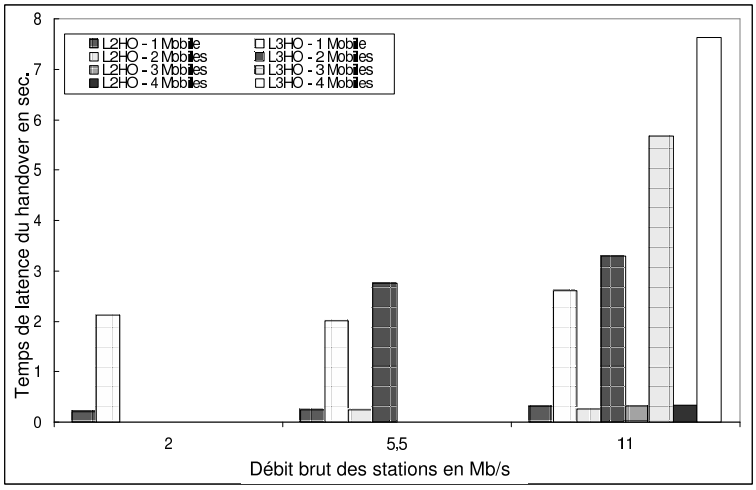


FIG. 3.8 – Mesures des temps de handovers de niveau 2 et 3 avec du trafic MPEG2 sur 4 mobiles

messagerie instantanée), les applications "quasi" temps réel (streaming audio et vidéo), et les applications temps réel (visio et audio conférences). Les applications faiblement interactives ne sont pratiquement pas pénalisées par les déplacements et la mise en place des mécanismes de mobilité de niveau 2 et 3. Les applications temps réel ont besoin de nouveaux protocoles pour minimiser les temps de handover (eg. Bi-casting, Fast Mobile IPv6). Par contre, les applications de streaming pourraient être simplement modifiées pour mieux prendre en compte le concept du terminal Mobile IPv6. En effet, la plupart de ces applications proposent un mécanisme de temporisation, toutefois celui-ci n'est pas suffisamment étendu pour rendre transparents les handovers des terminaux. Il serait donc nécessaire de mieux gérer ces tampons pour masquer ces désagréments aux utilisateurs.

Pour finir, ces différentes expérimentations montrent la nécessité de définir de nouveaux concepts permettant de répondre aux besoins de ce qu'on l'on peut qualifier d'Internet Ambiant. En effet, les technologies sans fil se démocratisent et l'utilisation banalisée des téléphones mobiles cellulaires montre que les utilisateurs veulent de plus en plus communiquer n'importe où, n'importe quand. Aujourd'hui ce concept n'est appliqué qu'aux communications de type voix, mais il devrait rapidement se généraliser à tous les types de communications et notamment aux communications de données.

Cependant, les plate-formes de tests que nous avons utilisées restent restreintes. Pour le moment, les mesures n'ont pu qu'être faites avec un maximum de 6 noeuds mobiles. C'est pourquoi, nous avons pris l'initiative de développer un simulateur

de réseaux sans fil, spécifiquement élaboré pour l'évaluation des communications sans fil et les mesures des handovers. La prochaine section présente ce nouvel outil de simulation appelé *SimulX*. Dans la suite du document, nous exposerons les apports et propositions que nous avons faites pour améliorer les temps de handover, et réduire le temps de coupure ressenti par les applications. En particulier, nous chercherons à améliorer le handover horizontal de niveau 3 spécifiquement dans les réseaux IEEE 802.11b, par une anticipation de mouvements. Cette solution sera évaluée par notre simulateur SimulX. Par la suite, nous proposerons également des solutions optimisées pour le traitement des handovers verticaux, à savoir une architecture complète pour le terminal mobile et les protocoles adéquats de redirection situés aux différents niveaux de la pile TCP/IP.

Chapitre 4

Simulateur de réseaux sans fil : SimulX

4.1 Introduction

La validation et le contrôle des performances sont des étapes obligatoires dans le développement de protocoles. Alors qu'une validation par implémentation sur des ordinateurs est de loin la méthode la plus démonstrative du bon fonctionnement d'un protocole, l'outil de simulation peut quelques fois être préféré, dans deux cas de figure particulièrement. D'une part, des simulations permettent généralement d'obtenir des résultats plus rapidement qu'une implémentation dans un sytème d'exploitation. La simulation pourra alors servir de premier outil de mesure d'un protocole en cours de spécification. L'intérêt est ici évident : les simulations permettront de s'assurer du bon comportement du protocole, d'identifier des situations critiques pour lesquelles aucun mécanisme n'avait été prévu et de déterminer si le nouveau protocole apporte un réel gain pour la solution qu'il se propose de résoudre.

D'autre part, l'outil de simulation peut également servir à vérifier le comportement d'un protocole à très grande échelle. Bien qu'il soit souvent possible d'implémenter un protocole en cours de spécifications (ou même un protocole finalisé), il est souvent difficile de dépasser une certaine taille de plate-forme de tests. Très souvent, les mesures sont limitées à une dizaine de nœuds tout au plus, dans des configurations pré-câblées, qui ne représentent pas forcément l'environnement de fonctionnement du protocole en question. Dans notre cadre particulier de recherche sur la mobilité des nœuds, les quelques articles de recherche [30, 103, 67, 89] que

nous avons vus dans le chapitre 1 confortent ces affirmations : souvent les tests sont réalisés avec un nœud mobile se déplaçant entre deux sous-réseaux IPv6, isolés du reste de l'Internet. Lors de notre évaluation de ces protocoles dans le chapitre 3, nous avons essayé de dépasser ces limites. Cependant, nous n'avons pas pu faire de test avec plus de six nœuds mobiles qui se déplaçaient simultanément. Avec le développement d'un réseau sans fil au sein de l'université Louis Pasteur à Strasbourg [123, 142], et plus récemment avec notre participation à Nautilus [134], nous espérons dépasser les limites d'une plate-forme locale. En particulier, le projet Nautilus a l'ambition de développer un ensemble de protocoles pour la mobilité (Mobile IPv6 [84], NEMO [55], etc.) et de les tester sur une plate-forme internationale regroupant plusieurs sites en France et au Japon. Pour cette étude, 40 utilisateurs ont été munis d'assistants personnels.

Cependant, la collecte d'informations et la mise en place des outils de mesure requièrent énormément d'énergie et de temps. De plus, bien que cette plate-forme de tests soit de dimension plus réaliste, le facteur de mise à l'échelle et la réalisation de scénarii très spécifiques ne sont toujours pas possibles. C'est pourquoi, la simulation demeure un outil incontournable dans le cadre de notre recherche.

Dans cet optique, nous avons entrepris une recherche d'un simulateur capable de reproduire un modèle de l'Internet avec des nœuds mobiles. En particuler, nous recherchions la simulation robuste d'environnements sans fil, avec le respect des spécifications des protocoles de communication simulés. Comme nous l'expliquons dans la suite, nous n'avons pas trouvé l'outil adéquat pour satisfaire nos ambitions, c'est pourquoi nous avons débuté le développement d'un nouveau simulateur nommé *SimulX* qui est présenté dans ce chapitre. Après une brève présentation de NS-2 et ses limites, nous détaillerons les besoins qui ont motivé le développement de SimulX. Ensuite, les fonctionnalités principales seront présentées. Enfin, nous donnerons une étude du débit utile dans les réseaux 802.11 lorsque plusieurs nœuds mobiles se partagent un même point d'accès avant de conclure le chapitre.

4.2 Intérêt du développement d'un nouveau simulateur

Le simulateur NS-2

Afin de tester nos protocoles de gestion de la mobilité, nous avons recherché un outil de simulation non commercial. Le simulateur réseau NS-2 (*Network Si-*

mulator) est un simulateur très réputé dans la communauté de recherches sur les protocoles pour le réseau car il permet de valider un protocole dans une architecture assez complète représentant l'Internet. De plus, ce projet fournit une méthode de visualisation des topologies et du déroulement des simulations grâce à l'outil NAM [62]. NS-2 est un simulateur à événements discrets orienté objet, basé sur le simulateur réseau REAL [155]. Au départ, la version 1.0 de NS a été développée au *Lawrence Berkeley National Laboratory* par le groupe de recherche réseau. Son développement fait maintenant partie du projet VINT [53] à l'université de Californie du Sud, qui a mis en place la version 2.0. Le but de ce projet est la construction d'un simulateur réseau communautaire offrant des outils et des méthodes innovatrices dans un environnement proche de la réalité.

Chaque classe du simulateur est dupliquée en C++ et OTCL. Le langage OTCL est un langage interprété qui ne demande pas de compilation. Il est principalement utilisé pour concaténer des objets, accéder aux objets à partir de l'interpréteur et configurer des simulations (début et arrêt des événements, perte réseau, rassemblement de statistiques). Son utilisation est rapide et assez conviviale. D'un autre côté, C++ est utilisé pour créer les classes de base et pour traiter un grand nombre de données (tel que calcul des tables de routage, mouvement des mobiles).

La mobilité dans NS-2

Dans un premier temps, la mobilité a été introduite dans NS-2 par les chercheurs de l'université Cartegie Mellon de Pittsburgh avec la volonté de simuler des réseaux ad hoc. L'introduction de nouveaux nœuds, de déplacements en 2 dimensions et de protocoles de routage ad hoc ont donc été mis en place. La structure d'un nœud est également implémentée en couches à la manière du modèle TCP/IP, avec dans les couches basses l'implémentation d'une couche liaison, d'une couche MAC et d'une interface réseau qui simule les comportements physiques des ondes radio.

Les limites de ce premier modèle se sont vite faites ressentir et une extension a donc été apportée pour réaliser des simulations hybrides où des nœuds sans fil pouvaient interagir avec des nœuds filaires. Cette extension s'appuie sur un adressage hiérarchique et sur l'introduction d'un nouveau type de nœud représentant les points d'accès. Cependant ces nouveaux types de nœud font également office de routeurs (ou agent de mobilité pour l'Internet IPv4). Ceci amène une première limite du modèle : un handover de niveau 2 (changement de point d'accès) entraînera obligatoirement un handover de niveau 3 (changement de sous-réseau). La première implémentation de Mobile IP (version 4) a été faite par Sun Microsystem

qui a étendu de la même manière que cité précédemment la version de base de NS-2. L'implémentation de Mobile IP consistait en l'ajout de l'entité agent mère et l'encapsulation et la décapsulation des paquets.

L'introduction de la mobilité IPv6 (et par la même occasion des premiers concepts d'IPv6) fut l'objectif du projet MOBIWAN [112]. L'objectif de ce projet était d'intégrer la mobilité des nœuds dans des grandes aires de réseaux IPv6 dans les simulations NS-2. L'implémentation des protocoles Mobile IPv6 et IPv6 est mise en œuvre par un jeu de nouveaux agents et de nouveaux classificateurs NS-2. Toutes les particularités d'IPv6 n'ont pas été mises en œuvre, seules celles nécessaires à Mobile IPv6 ont été implémentées (par exemple le protocole de découverte des voisins [133]).

Les limites du simulateur NS-2

Bien que le simulateur NS-2 soit très complet et propose de nombreuses fonctionnalités, ses limites en ce qui concerne la modélisation des nœuds mobiles sont importantes. Hormis le fait qu'aujoud'hui NS-2 demande un fort investissement dû à sa complexité, ses limites se font ressentir à deux niveaux : d'une part l'implémentation de IEEE 802.11b est incomplète et d'autre part les mécanismes de gestion de la mobilité IPv6 sont restreints.

Tout d'abord la simulation des réseaux IEEE 802.11 n'est que très peu fiable. Plusieurs erreurs dans le protocole d'accès au canal sont fréquemment relevées par les utilisateurs. Des erreurs sur les débits maximum observés ou sur les retransmissions sont régulièrement rapportées dans la liste de diffusion consacrée à NS-2. D'autre part, le handover de niveau 2 dans les réseaux IEEE 802.11b n'est pas implémenté. Un nœud mobile peut recevoir du trafic de plusieurs points d'accès au même moment et le changement de point d'accès est instantané, sans le moindre échange de message. Or, comme décrit dans la section 2.2, un nœud mobile ne peut recevoir de trafic que d'un seul point d'accès à la fois et le mécanisme de handover nécessite plusieurs phases de découverte et d'association par lesquelles le nœud mobile doit passer. Le temps nécessaire au changement de point d'accès est de loin non négligeable (mesuré entre 200 et 300 millisecondes en moyenne dans la section 2.2.4). De plus, comme nous l'avons vu ci-dessus, un handover de niveau 2 entraînera obligatoirement un handover de niveau 3 car un point d'accès représente également un routeur d'accès. Encore une fois ce point là n'est pas représentatif de la réalité puisqu'il est très fréquent que plusieurs points d'accès soient dans un même sous-réseau. De plus, une des limites de l'utilisation des déclencheurs de niveau 2 pour optimiser le handover est le fait de pouvoir déterminer si un change-

ment de point d'accès occasionne ou non un changement de sous-réseau. Comme nous nous intéressons particulièrement à l'utilisation des déclencheurs de niveau 2, les simulations dans NS-2 se révèleraient biaisées.

D'autre part l'implémentation de Mobile IPv6 est très limitée. Les simulations ne peuvent comporter qu'un unique nœud mobile. Cette limitation est extrêmement restrictive puisque l'objectif d'un simulateur est notamment de valider des comportements à grande échelle. De plus, l'implémentation IPv6 disponible n'est que partielle et repose principalement sur des extensions de l'implémentation courante.

Les besoins

Dans le contexte de notre recherche sur la gestion de la mobilité, l'outil de simulation est apparu comme impératif. En effet, tous les tests que nous avons déjà pu faire (voir section 2.2 et chapitre 3) ont été réalisés sur des plate-formes IPv6 de taille restreinte. Or le comportement à grande échelle est nécessaire puisque les nœuds mobiles dans l'Internet se compteront très certainement par millions.

Par ailleurs la multiplication des solutions et optimisations de gestion de la mobilité est telle aujourd'hui qu'il est difficile de comprendre quelle solution est la meilleure. Souvent, les auteurs d'une solution évaluent leur proposition dans un certain environnement prédéfini. Or une comparaison entre ces propositions selon divers paramètres est capitale pour évaluer les avantages et inconvénients de chacune d'entre elles.

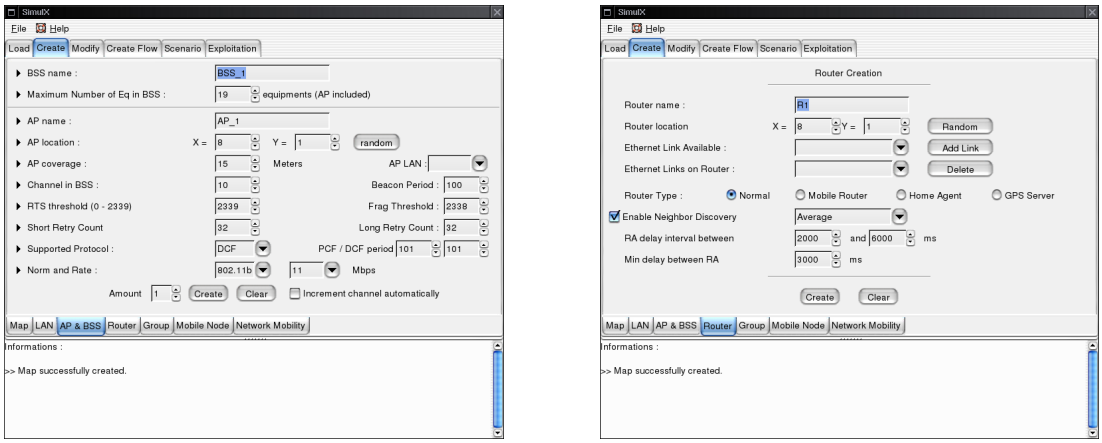
Dans ce contexte, il nous est apparu indispensable de disposer d'un outil robuste de simulation, adapté aux caractéristiques des réseaux sans fil. Les fonctionnalités attendues d'un tel outil sont la modularité afin de pouvoir ajouter une nouvelle technologie ou un nouveau protocole aisément, la rigueur d'implémentation, dans le but d'observer des comportements proches de la réalité et la simplicité d'utilisation dans la configuration et la récupération de résultats. Le simulateur *SimulX* a été développé dans ce sens. Il est présenté dans la suite de ce chapitre.

4.3 Fonctionnalités

L'implémentation du simulateur SimulX a démarré par la simulation de cellules IEEE 802.11b. SimulX a ensuite été étendu pour la simulation de LAN sans fil (ESS) par l'interconnexion des points d'accès par un LAN Ethernet et finalement

par la mise en place de backbone IPv6 avec l'implémentation de routeurs. Chaque composant principal hérite d'une classe abstraite générique qui permet une modularité et la mise en place d'extensions futures. La suite de cette section présente l'interface graphique du simulateur, la configuration et les options de lancement et enfin les résultats générés.

L'interface graphique



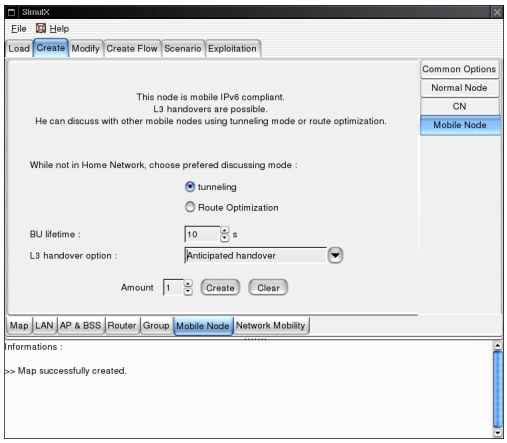
(a) Configuration des points d'accès

(b) Configuration des routeurs

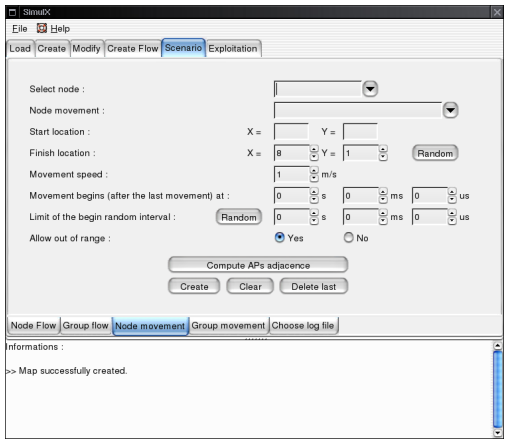
FIG. 4.1 – Configuration des points d'accès et des routeurs dans SimulX

L'interface graphique du simulateur est présentée dans les figures 4.1 et 4.2. On remarquera en particulier certaines listes déroulantes très pratiques pour la simulation de protocoles liés à la mobilité. Par exemple, plusieurs configurations de fréquence d'émission de RA sont proposées lors de la création d'un routeur (figure 4.1(b)), ou alors l'utilisateur peut choisir l'optimisation de handover de niveau 3 lors de la création d'un nœud mobile.

Outre les divers paramètres de configuration, une carte implémentée en OpenGL représente la disposition des nœuds d'une simulation (voir les figures 4.3). La carte est interactive et permet non seulement la visualisation des nœuds, de leur portée et des réseaux, mais également de placer et de déplacer les différents nœuds. Les fonctionnalités de l'outil NAM de NS-2 nous ont également inspiré la lecture à posteriori des simulations : une fois une simulation terminée, il est possible de rejouer la simulation visuellement sur la carte. Comme le montre la figure 4.3(b), l'échange

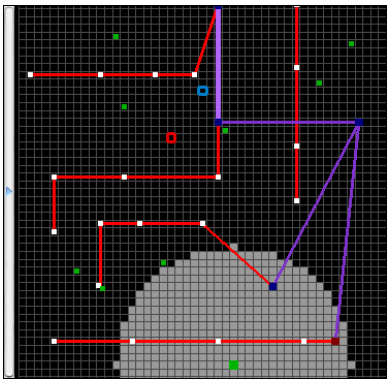


(a) Configuration des nœuds mobiles

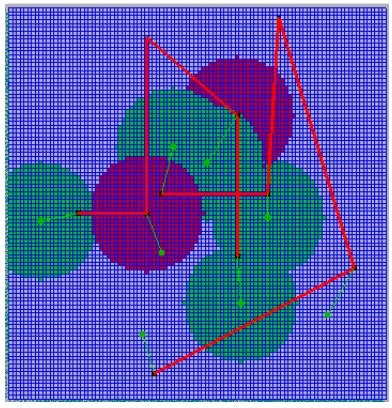


(b) Configuration des déplacements

FIG. 4.2 – Configuration des nœuds mobiles et de leur mouvement dans SimulX



(a) Visualisation d'une configuration



(b) Déroulement d'une simulation

FIG. 4.3 – Configuration des nœuds mobiles et de leur mouvement dans SimulX

des paquets est également représenté. Cette fonctionnalité étend l'utilisation de SimulX à un outil pédagogique d'enseignement pour montrer le fonctionnement des réseaux sans fil.

Configuration

De nombreux paramètres peuvent être configurés et enregistrés dans des fichiers de sauvegarde. Bien que l'objet de cette présentation ne soit pas de détailler l'ensemble du simulateur, trois possibilités de configuration méritent notre attention :

- Le déplacement : comme dans NS-2, il est possible de déterminer un déplacement précis (avec un point d'origine, un point d'arrivée et une vitesse), ou de générer des déplacements aléatoires. Cependant, nous avons étendu les mouvements aléatoires avec une option qui permet de générer des mouvements dont le chemin est couvert par la portée de points d'accès. Ainsi, il est possible de simuler des mouvements aléatoires sans que les nœuds mobiles se retrouvent hors de portée de tout point d'accès. De plus, il est possible de spécifier que chaque mouvement doit comporter au moins un changement de point d'accès. Cette fonctionnalité permet de s'assurer que les nœuds mobiles feront des handovers.
- La notion de groupe : il est possible de créer des groupes de nœuds mobiles afin de faciliter la création de simulations importantes. Ainsi, des modèles de flux ou de déplacement peuvent être attribués à l'ensemble des nœuds mobiles d'un groupe.
- Les différentes configurations adaptées pour la simulation de la mobilité : étant donné que l'objectif premier du développement était la simulation des protocoles de gestion de la mobilité, plusieurs choix quant à la configuration des simulations sont disponibles : configuration du protocole de découverte des voisins (fréquence des RAs), utilisation des déclencheurs de niveau 2, Mobile IPv6 tels que décrit dans le RFC 3775 [84], choix de l'utilisation de l'optimisation de routage, ou la solution de handover anticipé décrite dans le chapitre 5.

Résultats de simulation

Plusieurs fichiers de sortie récapitulent les traces d'une simulation. Le choix des fichiers à générer est disponible en début de simulation. Entre autres, un fichier est consacré au handover de niveau 2 retraçant tous les messages échangés. Un autre fichier récapitule le temps de handover de niveau 3 pour chaque nœud mobile et un autre fichier retrace toutes les statistiques de la simulation : nombre de retransmissions, de pertes, de paquets échangés... Pour chaque flux échangé entre deux stations, le temps d'arrivée de chaque paquet est également enregistré dans un fichier, permettant la création de graphiques.

4.4 Evaluation de la norme IEEE 802.11b

Le premier objectif qui a motivé le développement de ce simulateur a été la simulation des réseaux sans fil IEEE 802.11. Dans le chapitre 2, nous avons testé les performances des réseaux IEEE 802.11b, avec un maximum de 6 stations. Cette étude nous avait montré que le débit utile était très inférieur au débit brut. Plus le débit est élévé, plus le rapport entre le débit brut et le débit utile observé est faible. De plus, nous avions constaté que le débit total du système était meilleur avec plusieurs stations, qu'avec une seule lorsque les équipements étaient configurés à 11 Mb/s. Autrement, nous avons pu observer qu'un point d'accès supporte très bien jusqu'à 6 stations simultanées, qui génèrent intensément du trafic.

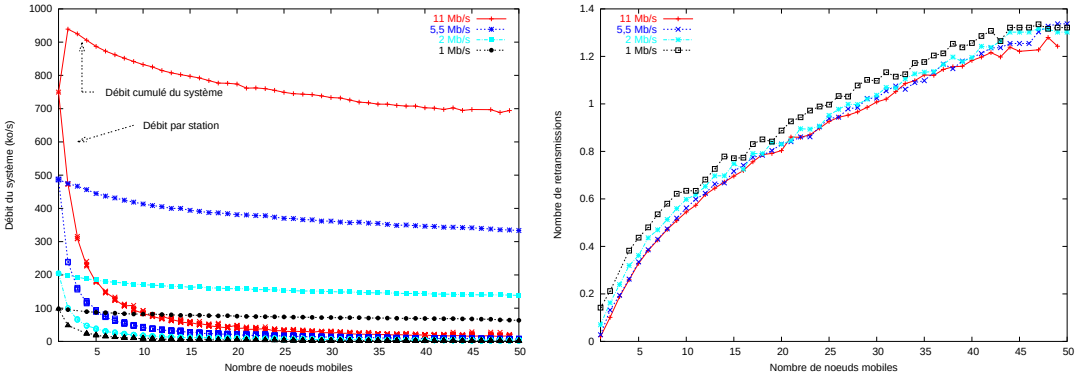
SimulX nous permet de valider ces mesures de performance des réseaux 802.11b par la simulation d'un grand nombre de stations. Nous allons pouvoir déterminer comment est supportée la mise à l'échelle du protocole d'accès au médium, et quels débits utiles peuvent être attendus par les stations IEEE 802.11b rattachées à un BSS.

4.4.1 Hypothèses de tests

Nous avons mis en place 4 séries de 50 tests, une par débit (1, 2, 5,5 et 11 Mb/s). Dans chaque test nous avons incrémenté le nombre de stations, toutes rattachées au même point d'accès (de 1 à 50) et configurées avec le même débit d'émission. Toutes ces stations émettent un flux en continu, c'est-à-dire que dès la fin de transmission d'une trame, une autre est à émettre. Chaque trame contient 1500 octets de données. Chaque simulation a été rejouée 100 fois. Il devra également être noté qu'on ne tient pas compte ici du problème des stations cachées (voir section 2.2.3), puisque toute station du BSS peut entendre la transmission courante de chacune d'entre elles. Les résultats de ces simulations sont donnés dans les sous-sections suivantes.

4.4.2 Résultats

Le graphique 4.4(a) représente non seulement le débit cumulé du système, mais également le débit utile de chaque station pour les quatre valeurs de débit brut disponibles dans la norme (nous rappelons que pour chaque test, toutes les stations sont configurées avec le même débit théorique). Le premier résultat surprenant est le fait que le débit total du système à 11 Mb/s est bien meilleur pour plusieurs



(a) Débit global et débit de chaque stations dans un BSS

(b) Nombre de retransmissions par trame de données envoyée

FIG. 4.4 – Mise à l'échelle du protocole d'accès au médium dans un BSS IEEE 802.11b

stations que lorsqu'une station est seule rattachée au point d'accès : le débit utile d'une station est de 749 ko/s contre 939 ko/s observés dans le BSS lorsque deux stations sont rattachées au point d'accès. Ensuite, plus le nombre de stations augmente, plus le débit dans le BSS baisse. C'est à partir de 25 stations que le débit du BSS repasse en-dessous du débit observé pour une seule station.

Ce résultat est d'autant plus surprenant que le débit du système avec deux stations est même plus grand que le débit utile théorique que nous avions calculé pour une station rattachée à un point d'accès dans la section 2.2.6. En effet, nous avions trouvé 7,403 Mb/s ce qui est équivalent à 925 ko/s contre les 939 ko/s trouvés par simulation pour deux stations rattachées à un point d'accès. L'explication de ce phénomène provient certainement de l'algorithme de backoff (temps d'attente choisi aléatoirement avant chaque transmission). Lorsqu'une station est seule rattachée au point d'accès, elle doit décrémenter son backoff entre chaque transmission effective de trames, comme le montre le haut de la figure 4.5. Or, lorsque deux stations se partagent l'accès au médium, alors qu'une station décrémente son backoff, l'autre station fait de même. Ainsi, après la transmission de la première trame (de la station qui avait le backoff le plus court), l'autre station a moins de temps à attendre que si elle venait de tirer un backoff. Ceci implique que le temps d'attente entre les trames sera à chaque fois inférieur lorsque plusieurs stations veulent émettre que lorsqu'il n'y en a qu'une seule.

Ce phénomène est illustré dans le bas de la figure 4.5 pour deux stations rat-

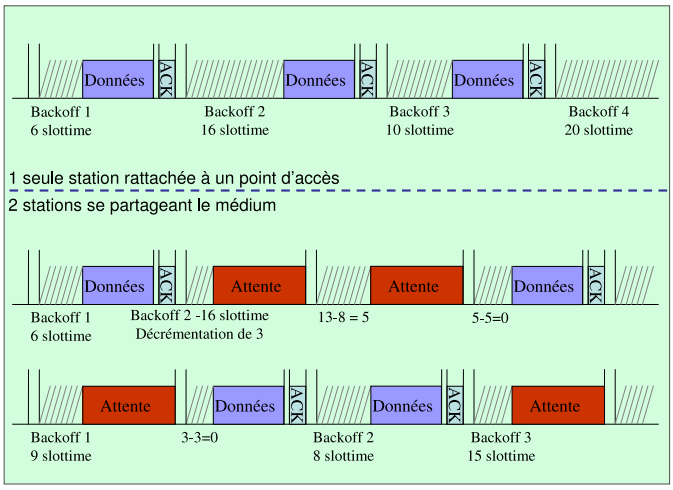


FIG. 4.5 – Différences d'accès au canal entre une et deux stations rattachées à un point d'accès

tachées à un point d'accès. Au départ, les deux stations tirent un backoff de 6 et 9 slottime. Elles décrémentent toutes les deux ce backoff tant que le canal est vide. La station 1 arrive à 0, alors qu'il reste 3 slottime pour la station 2. La station 1 commence alors sa transmission. Une fois cette dernière terminée, elle tire un nouveau backoff, pour sa prochaine transmission, et le décrémente en même temps que la station 2. Or après 3 slottime seulement, le backoff de la station 2 arrive à 0 et la station 2 peut donc émettre. Le temps d'attente entre les trames est donc inférieur quand plusieurs stations veulent émettre en même temps, car elles décrémentent leur backoff toutes en même temps. On voit d'ailleurs sur la figure que pendant le même temps, 3 trames sont transmises lorsque la station est seule rattachée au point d'accès contre 4 trames lorsqu'elles sont deux.

On peut alors se demander pourquoi il n'en est pas de même pour les autres débits bruts. En effet, pour les débits 1, 2 et 5,5 Mb/s, le meilleur débit dans le BSS est atteint lorsqu'une seule station est rattachée au point d'accès, alors que le même algorithme de backoff est employé. Ceci est dû au fait que le temps de transmission des trames est beaucoup plus long pour les bas débits qu'à 11 Mb/s. Le temps d'attente de la décrémentation du backoff a donc une part moins importante sur le temps de transmission total. C'est également pour cette raison que l'efficacité du protocole est meilleure pour les bas débits : plus le temps de transmission effectif d'une trame est long, plus les temps inter-trames et temps d'attente de backoff ont proportionnellement une incidence moindre.

4.4.3 Les retransmissions

Sur le graphique 4.4(b), le nombre de retransmissions par trame de données émise est représenté. On remarque ici que le nombre de retransmissions est très important : pour dix stations rattachées à un point d'accès, une retransmission est nécessaire pour plus d'une trame de données sur deux. A partir de 30 stations, chaque trame (en moyenne), nécessite une retransmission, c'est-à-dire que chaque trame de données doit être envoyée deux fois. C'est ce nombre important de retransmission qui vient diminuer l'effet cité ci-dessus, et qui occasionne une baisse régulière du débit en fonction du nombre de stations.

En outre, on pourra remarquer que le pourcentage de retransmissions est le même pour chaque débit brut.

4.4.4 Le débit de chaque station

Le graphique 4.4(a) montre également le débit moyen par station. Il est important de noter ici que tous les débits moyens observés sur chaque station sont représentés sur la figure, pour former un nuage de points (par exemple, pour 30 stations, il y a 30 valeurs). Ceci nous montre que l'algorithme d'accès au médium est équitable quel que soit le nombre de stations. Chacune d'entre elles finit sa transmission au même moment, sans qu'une station ne prenne la main sur le médium pour l'ensemble de ses transmissions. La bande passante disponible est alors divisée par le nombre de stations rattachées au point d'accès, et ce quel que soit le nombre de stations. On pourra peut-être retenir les valeurs suivantes : lorsque dix stations sont rattachées au point d'accès, chacune peut bénéficier d'une bande passante de 86 ko/s alors que lorsqu'elles sont 20 rattachées au point d'accès, elles peuvent chacune bénéficier de 41 ko/s en moyenne (à 11 Mb/s).

4.4.5 Conclusion

Ces tests nous ont révélé plusieurs caractéristiques de la norme IEEE 802.11b, et plus généralement de la série des normes IEEE 802.11 (b, a, g). Tout d'abord, nous remarquons que l'algorithme d'accès au médium est meilleur pour les bas débits, ce qui se comprend de manière historique. En effet, la norme IEEE 802.11 ne proposait lors de sa création que les débits de 1 et 2 Mb/s, et l'algorithme DCF a été spécifié pour ces vitesses de transmission. Lorsque le débit brut a augmenté avec la spécification de IEEE 802.11b, le même protocole d'accès a été utilisé,

et nous pouvons voir par ces tests sa baisse de performance. Cette baisse sera sensiblement plus importante avec des débits plus hauts, comme ceux disponibles dans 802.11a et 802.11g (jusqu'à 54 Mb/s). C'est peut-être l'efficacité de la couche MAC qui sera un frein à l'augmentation du débit pour cette technologie.

Par ailleurs, ces tests nous révèlent tout de même que la norme IEEE 802.11 s'adapte bien au facteur d'échelle. Malgré un nombre important de retranmissions, 50 stations sont relativement bien supportées et surtout le médium est équitablement partagé entre chaque station pour un débit identique.

4.5 Conclusion

Ce chapitre a brièvement présenté le nouveau simulateur *SimulX* dédié à la simulation de réseaux IPv6 sans fil. Son développement, qui a duré deux ans, est provenu du manque d'outil robuste à la simulation des réseaux IEEE 802.11. NS-2 [143], simulateur reconnu dans la communauté de recherche en réseau, n'a qu'une implémentation partielle de la norme IEEE 802.11 et de Mobile IPv6. Plutôt que d'améliorer le code de ce dernier, nous avons préféré développer notre propre simulateur avec des objectifs bien définis. Une implémentation rigoureuse de la norme IEEE 802.11 fut à la base du simulateur, qui a ensuite évolué vers la simulation de réseaux IPv6 avec l'implémentation de Mobile IPv6 [84] et du protocole de découverte des voisins [133]. SimulX dispose actuellement de plusieurs optimisations à la gestion de la mobilité IPv6, comme celle présentée dans le chapitre suivant sur l'anticipation des handovers.

En plus des protocoles implémentés, nous avons souhaité mettre en place un outil de simulation convivial, qui puisse être étendu dans le futur. C'est pourquoi il a été implémenté en C++, langage objet qui permet une modélisation. C'est aussi la raison pour laquelle nous avons développé une interface graphique. En plus de proposer la configuration de tous les paramètres de simulation (nœuds mobiles, correspondants, points d'accès, routeurs), l'interface graphique propose la visualisation de la carte de simulation en deux dimensions, ainsi que la lecture d'une simulation terminée. En effet, en fin de simulation, les déplacements des nœuds mobiles et l'échange de paquets sont visualisés sur la carte. Par ailleurs, il est évidemment possible de sauvegarder des configurations de tests, et de lancer leur exécution en ligne de commande.

Les premiers résultats que nous avons obtenus avec SimulX, aussi bien ceux présentés dans la section 4.4 que ceux présentés dans la section suivante, sont tout à fait cohérents et confirment le bon fonctionnement du simulateur par rapport

aux évaluations réalisées dans les sections 2.2.6 et 2.2.7. En outre, SimulX est au cœur de nouveaux thèmes de recherche dans notre équipe comme l'optimisation de handover par les techniques de géo-localisation et l'utilisation simultanée de plusieurs interfaces radio. D'autres extensions envisageables concernent la gestion de réseaux mobiles, où les points d'accès et les routeurs se déplacent sur la carte, ou encore l'implémentation de nouvelles technologies de communication comme le GPRS ou Bluetooth.

Chapitre 5

Anticipation des déplacements dans les réseaux IEEE 802.11b

5.1 Introduction

L'évaluation de Mobile IPv6 réalisée dans le chapitre 3 nous a montré que le temps de handover reste encore relativement important. Durant ce temps de latence, les communications des nœuds mobiles sont interrompues. Or ce temps d'interruption est néfaste pour les applications temps réel telles que la vidéo-conférence ou la réception de flux multimédia. C'est pourquoi nous avons cherché une solution qui non seulement minimise le temps de handover, mais également permet au nœud mobile de recevoir ses paquets de données le plus tôt possible à sa nouvelle localisation.

Notre proposition concerne exclusivement les réseaux de type 802.11. Le fait de se limiter aux réseaux 802.11 peut être vu comme une limitation puisque notre solution ne sera pas réalisable avec d'autres technologies, comme Bluetooth ou même d'autres à venir. Mais à contrario, le fait de connaître la technologie sous-jacente va nous permettre de mettre en place des optimisations ciblées et de proposer une solution complète d'amélioration des handovers de niveau 2 et 3. De plus, la norme 802.11 est certainement la solution de WLAN la plus répandue et il semble judicieux de mettre en place des mécanismes spécifiques à cette norme.

Comme nous l'avons vu dans le chapitre 1, un handover de niveau 3 peut généralement se décomposer de la façon suivante : handover de niveau 2, détection de lien, création de la nouvelle adresse temporaire, vérification de l'unicité d'adresse,

puis enregistrement de cette nouvelle adresse temporaire. Nous avions vu que la mise en place d'une solution hiérarchique était fortement recommandée pour réduire le temps d'enregistrement (temps d'aller-retour pour la mise à jour de la localisation). En ce qui concerne la détection de duplication d'adresse [176], il est envisageable de commencer à utiliser la nouvelle adresse temporaire en parallèle de la détection de duplication d'adresse. De toute manière, IPv6 ne prévoit pas de mécanisme en cas de collision, et le nœud mobile, ainsi que la machine qui utilise la même adresse, verront leur accès réseau fortement perturbé sur le lien en question. Cependant, la probabilité de provoquer une duplication d'adresse est très faible (inférieure à un millionième).

Notre solution concerne donc une optimisastion des trois mécanismes restants : le handover de niveau 2, la détection de lien et la création d'adresse. Le handover de niveau 2 sera amélioré par une anticipation de mouvement, la détection de lien ne va plus reposer sur la fréquence des RA, mais plutôt sur l'utilisation des déclencheurs de niveau 2 et la création d'adresse se fera par anticipation. Le *Handover Anticipé* consiste en deux méthodes : une méthode de *Handover Anticipé initié par le nœud mobile* et une méthode de *Handover Anticipé à posteriori*, où l'implication du nœud mobile dans le protocole est limitée. Après la présentation des déclencheurs de niveau 2 que nous allons utiliser, ces deux mécanismes seront décris. Finalement une évaluation du Handover Anticipé dans le simulateur SimulX (voir le chapitre précédent) et une comparaison entre le Handover Anticipé initié par le nœud mobile avec d'autres solutions seront présentées dans la dernière section.

5.2 Utilisation des déclencheurs de niveau 2

Notre méthode de réalisation de handovers anticipés repose principalement sur une collaboration et un échange d'informations entre les couches 2 (niveau MAC) et 3 (niveau IP) du modèle TCP/IP. Afin d'optimiser le handover de niveau 3, le nœud mobile va utiliser l'information présente au niveau 2 pour anticiper ses déplacements. L'information utilisée se présente sous deux formes : la mesure permanente de l'intensité de signal entre le nœud mobile et son point d'accès et l'introduction d'information de niveau 3 dans les messages de contrôle de niveau 2. Ces deux points sont présentés ci-dessous.

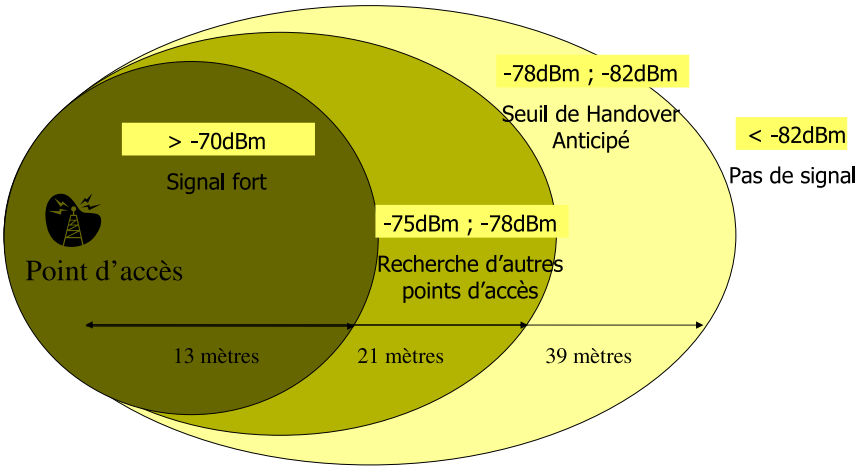


FIG. 5.1 – Mesures d'intensité du signal entre un nœud mobile et son point d'accès

5.2.1 Intensité de signal

Dans les réseaux IEEE 802.11b (voir section 2.2), un nœud mobile surveille constamment l'intensité de signal entre lui-même et son point d'accès. L'intensité de signal mesurée indique au nœud mobile la qualité de sa liaison avec son point d'accès. C'est le seul paramètre que le nœud mobile prend en compte pour provoquer un changement de point d'accès ; lorsque le signal passe en-dessous d'un certain seuil, le nœud mobile déclenche un handover de niveau 2 afin de découvrir un autre point d'accès offrant un meilleur signal.

La figure 5.1 représente les mesures que nous avons faites sur l'intensité de signal par rapport à la distance entre le nœud mobile et son point d'accès, dans un environnement intérieur. D'après ces mesures réalisées avec des cartes Cisco Aironet 350, l'intensité de signal varie entre -30 dBm et -82 dBm lorsque le nœud mobile est rattaché à un point d'accès, la valeur la plus proche de zéro indiquant le meilleur signal. Lorsque le nœud mobile est dans un rayon de 6 mètres autour de son point d'accès, une forte variation de son intensité de signal est observée : tous les deux mètres, l'intensité de signal diminue de 10 dBm jusqu'à atteindre -70 dBm. Ensuite l'intensité de signal est moins volatile et perd environ 1 dBm tous les mètres, jusqu'à atteindre -82 dBm, seuil de désassociation. Au-delà de ce seuil, le nœud mobile ne peut pas rester associé à son point d'accès. D'après nos mesures, cette limite se situe approximativement dans un rayon de 20 mètres autour du point d'accès. Bien entendu cette distance est donnée à titre informatif, car elle dépend fortement de l'infrastructure des bâtiments, de la puissance d'émission des équipements et du débit de ces derniers. Par ailleurs, cette valeur n'est nullement

utilisée dans notre protocole. Le résultat important de ces mesures est que l'intensité de signal entre un nœud mobile et son point d'accès devient relativement fiable et stable lorsque le nœud mobile est à une distance de plus de 6 mètres de son point d'accès.

Nous verrons par la suite comment utiliser cette valeur d'intensité de signal pour déclencher les opérations d'anticipation de mouvement. Un point fort de l'utilisation de cette valeur est que toutes les cartes IEEE 802.11b fournissent déjà un contrôle d'intensité de signal. Il ne sera donc pas nécessaire d'implémenter le contrôle périodique de sa valeur.

5.2.2 Information de niveau 3 dans les messages de niveau 2

Lorsque le nœud mobile entame un processus de handover de niveau 2, il commence par envoyer des *Probe Request* en broadcast (voir section 2.2). Chaque point d'accès qui entend ce message répond alors avec un *Probe Response*. Le deuxième paramètre sur lequel repose notre nouveau mécanisme est l'introduction du préfixe IPv6 utilisé sur le lien du point d'accès émetteur dans les Probe Response. Si plusieurs préfixes IPv6 sont utilisés sur le lien, c'est le préfixe par défaut qui devra être transmis dans les Probe Response. Ainsi, le nœud mobile sera informé au plus tôt du préfixe IPv6 utilisé sur le lien cible. En outre, la découverte du préfixe IPv6 utilisé sur un lien pourrait également contribuer au processus de sélection du point d'accès cible lors de la phase de handover de niveau 2 en émettant une préférence pour les points d'accès situés sur le même sous-réseau que l'ancien point d'accès. Nous avons préféré ajouter ce champ dans le Probe Response plutôt qu'un autre message (comme le Beacon par exemple) car le Probe Response est uniquement envoyé sur soliciation des nœuds mobiles lors des handovers (par opposition à un message périodique). De plus c'est le message qui est déjà utilisé par les stations pour découvrir les capacités des points d'accès. L'étude d'une interaction forte entre les couches 2 et 3 est actuellement en cours dans le groupe de travail [10].

Bien entendu, cette extension requiert des modifications dans les trames de contrôle IEEE 802.11b. La création, l'émission et la réception des trames de contrôle sont implémentées dans le firmware de la carte Cisco Aironet 350 dont nous disposions. Il nous a donc été impossible de modifier les messages sur ce type de matériel. Face à cette constation, nous avons essayé d'introduire de nouvelles trames de gestion, mais leur émission n'était pas assez fiable. Dû au fait de ces difficultés d'implémentation, nous avons opté pour la simulation de notre mécanisme. Bien entendu, les modifications sont tout à fait réalisables mais pour cela il est

nécessaire d'avoir accès aux fonctions adéquates sur le matériel.

Les deux sections suivantes décrivent comment utiliser ces deux paramètres pour mettre en œuvre les deux solutions proposées.

5.3 Anticipation de handover initiée par le nœud mobile

Dans la solution de Handover Anticipé initié par le nœud mobile, c'est le nœud mobile qui a un contrôle total sur ses handovers. Le nœud mobile est en charge de la surveillance de l'intensité de signal, du déclenchement et de l'arrêt de la procédure de handover anticipé, sans aucune assistance du réseau. Ce modèle est bien adapté aux réseaux IEEE 802.11b qui reposent déjà sur une participation active du nœud mobile dans la gestion de ses déplacements. Les sous-sections suivantes détaillent le mécanisme, dont les étapes sont réprésentées dans la figure 5.3.

Initialisation du Handover Anticipé

La procédure de Handover Anticipé est amorcée lorsque l'intensité de signal passe en-dessous d'un certain seuil. D'après les mesures que nous avons faites (voir figure 5.1), et selon une vitesse de déplacement correspondant à la marche à pied, nous avons défini ce seuil à -78 dBm. Lorsque le signal devient moins bon que ce seuil (c'est-à-dire entre -78 dBm et -82 dBm), le nœud mobile déclenche le Handover Anticipé.

La première phase du Handover Anticipé est la découverte des points d'accès environnants. Lorsque le seuil du Handover Anticipé est atteint, le nœud mobile se met en mode de découverte pendant un cours laps de temps et envoie des Probe Request. La réception des Probe Response des points d'accès voisins permet au nœud mobile de connaître l'identité de ces points d'accès (grâce à l'adresse MAC contenu dans les messages) et le préfixe IPv6 utilisé sur le lien. Après cette courte période de découverte, le nœud mobile retourne en mode connecté sur son point d'accès courant. Le temps de découverte doit être court (quelques dizaine de millisecondes) car le nœud mobile ne peut ni envoyer ni recevoir des paquets de données pendant ce laps de temps et il doit pouvoir récupérer son association avec son ancien point d'accès (c'est-à-dire revenir en mode associé avant expiration de son association sur le point d'accès). Si jamais le nœud mobile n'a pas trouvé de point d'accès sur le canal sondé, il lui faudra rechercher un point d'accès sur

un autre canal. Après chaque recherche sur un canal, il est préférable pour le nœud mobile de revenir en mode associé pour récupérer les paquets en attente sur son point d'accès. Cependant dans les simulations qui seront présentées, le nœud mobile recherchera son futur point sans revenir dans le mode associé, même s'il doit changer de canal. Des études supplémentaires sont nécessaires d'une part pour déterminer le temps de sondage approprié et d'autre part le nombre de canaux sondés consécutivement.

A l'issue de cette phase de découverte, le nœud mobile peut avoir découvert un point d'accès cible. Dans ce cas, il lui est nécessaire de stocker l'adresse MAC du point d'accès, le canal et le préfixe IPv6 associé. Si plusieurs points d'accès ont répondu durant la phase d'anticipation, le nœud mobile pourra éventuellement émettre une préférence pour un point d'accès qui est connecté sur le même sous-réseau IPv6, puisque aucune mise à jour de sa localisation ne sera alors nécessaire. Si le point d'accès choisi par le nœud mobile est dans un sous-réseau IPv6 différent, le nœud mobile devra créer une nouvelle adresse temporaire basée sur le nouveau préfixe IPv6. Ensuite, le nœud mobile pourra demander la duplication du trafic (Bi-casting [101]) à la nouvelle adresse qu'il vient de créer en plus de sa position courante. Pour cela, le nœud mobile envoie un Binding Update contenant la nouvelle adresse temporaire dans une option de Bi-casting. Ce message aura pour effet d'ajouter une entrée dans le cache d'association du nœud destinataire plutôt que d'écraser l'ancienne adresse.

Etant donné le trafic supplémentaire que le Bi-casting va engendrer, il est recommandé de mettre un délai assez court sur le temps de duplication du trafic. Ce temps devra être choisi en fonction de la distance entre le nœud mobile et son destinataire, mais un défaut de 10 secondes semble être suffisant pour vérifier que l'anticipation était bonne. Passé ce délai, le nœud destinataire devra cesser de dupliquer le trafic et se contenter de l'envoyer à l'adresse temporaire principale du nœud mobile (dans son "ancien" sous-réseau).

Handover sur le point d'accès cible

Finalement, si le nœud mobile arrive effectivement en bordure de la couverture de son point d'accès courant, il devra commencer un handover de niveau 2. Dans l'éventualité où le nœud mobile a déterminé un point d'accès cible dans la phase de découverte, il pourra directement passer en phase d'authentification avec le point d'accès cible, sans avoir à repasser par la phase de découverte. Cette possibilité offre une optimisation importante sur le temps de handover de niveau 2 comme nous allons le montrer.

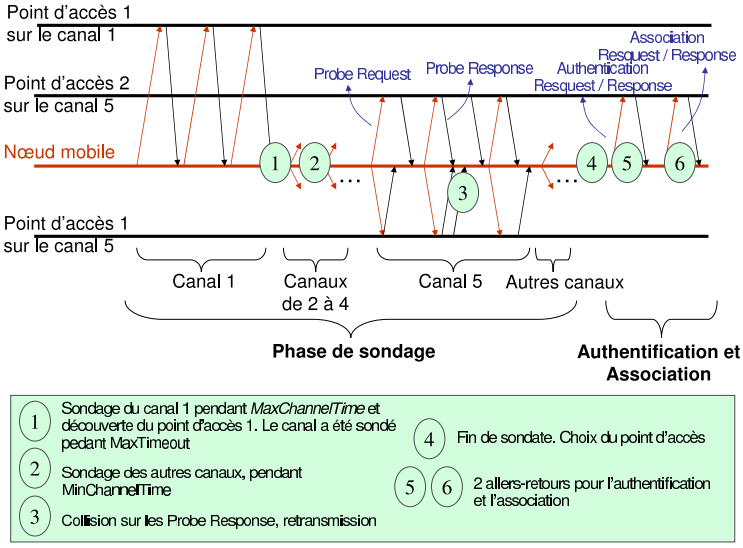


FIG. 5.2 – Différentes étapes du handover de niveau 2

Afin de mieux comprendre les avantages quant à l'omission de la phase de découverte du handover de niveau 2, revenons plus en détail sur ce mécanisme. Comme le montre la figure 5.2, la majeure partie du temps nécessaire pour réaliser un handover de niveau 2 est consacrée à la phase de découverte. La norme introduit deux délais qui sont *MinChannelTime* et *MaxChannelTime*. Le premier délai est le temps minimum de sondage sur un canal. Si le nœud mobile reçoit au moins une réponse d'un point d'accès avant que MinChannelTime n'ait expiré, alors il devra rester MaxChannelTime sur ce canal afin de recevoir les offres de tous les points d'accès présents sur ce canal. Par contre, s'il n'a pas reçu de réponse après avoir attendu MinChannelTime, le nœud mobile devra alors changer de canal. Malheureusement, les spécifications IEEE 802.11 ne vont pas plus loin dans les explications. Les délais MinChannelTime et MaxChannelTime ne sont pas définis. De plus, il n'est pas décrit si le nœud mobile doit continuer à sonder des canaux différents dans l'éventualité où il a déjà trouvé des points d'accès sur un canal. En se référant à l'implémentation Cisco, MinChannelTime vaut 30 millisecondes, MaxChannelTime vaut 200 millisecondes et dès que le nœud mobile a trouvé un point d'accès sur un canal, il ne changera plus de canal. Si on considère une différence de 4 canaux entre deux points d'accès dont les aires de portée se recouvrent partiellement, comme préconisé pour éviter les interférences, la phase de découverte prendra 290 millisecondes à elle toute seule.

C'est pourquoi avec l'anticipation que nous avons mise en place, le nœud mobile peut directement se mettre sur le canal du point d'accès cible et passer dans la

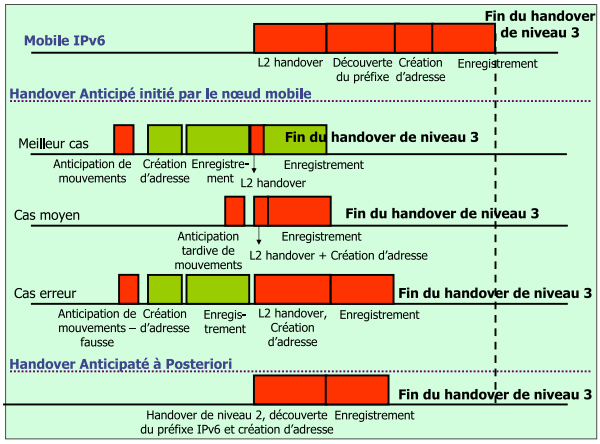


FIG. 5.3 – Echelle de temps dans les différents scénarii de l'Anticipation de Handover initié par le nœud mobile

phase d'authentification. La phase de sondage peut alors être évitée. Ce mécanisme est très efficace puisque si l'anticipation est bonne, le handover de niveau 2 va se résumer à quatre aller-retours entre le point d'accès et le nœud mobile, c'est-à-dire quelques millisecondes à peine. Si jamais l'authentification est mauvaise ou que le point d'accès n'est plus disponible, le nœud mobile s'en rendra compte puisque qu'il ne recevra pas de réponse à ses requêtes d'authentification. Il devra alors réaliser un handover de niveau 2 "classique".

Fin de la procédure

Finalement, une fois que le nœud mobile s'est rattaché à son point d'accès cible, il devra immédiatement envoyé un Binding Update classique pour annuler le Bi-casting. En parallèle, le nœud mobile devra vérifier l'unicité de son adresse sur le nouveau sous-réseau. On pourra noter que dès la fin du handover de niveau 2, le nœud mobile commencera à recevoir les paquets de données sans plus de délai, sous réserve que l'anticipation était bonne.

Cas d'erreur

Cette solution de Handover Anticipé comprend une certaine marge d'erreur dans la mesure où finalement on ne peut pas être sûr de l'anticipation réalisée

par le nœud mobile. Par exemple, le nœud mobile peut changer de direction dans son déplacement, ou le point d'accès cible peut ne plus être disponible (panne ou nombre maximum d'équipement atteint). Dans une telle situation, le nœud mobile s'en rendra compte dans la phase d'authentification puisqu'il ne recevra pas de réponse à ses requêtes. Dans ce cas, il devra alors repasser en phase de découverte et procéder à un handover de niveau 2 "classique". Par contre, comme les Probe Response des points d'accès contiennent le préfixe IPv6 du lien, le nœud mobile pourra découvrir le nouveau préfixe pendant le handover de niveau 2, et donc préparer une nouvelle adresse temporaire si nécessaire. Ainsi, dès le handover de niveau 2 terminé, il pourra envoyer un message de mise à jour de localisation. La figure 5.3 retrace les différentes étapes du protocoles dans le temps, selon les différents cas de figure que nous venons de citer.

Si jamais le nœud mobile ne fait pas de handover de niveau 2 (aller-retour jusqu'au bord de la cellule par exemple), l'entrée provocant le Bi-casting ne sera valide que pendant une certaine durée de vie. Un autre seuil d'intensité de signal peut éventuellement être mis en place qui attesterait que le nœud mobile n'est plus en bordure de couverture de son point d'accès. Après une procédure de Handover Anticipé, si jamais l'intensité de signal devient meilleure que ce seuil (indiquant que le nœud mobile s'est à nouveau rapproché de son point d'accès), le nœud mobile pourrait émettre un Binding Update classique indiquant son adresse temporaire courante comme unique adresse de localisation.

Afin d'éviter la mise en place intempestive du Bi-casting lorsqu'un nœud mobile reste constamment en bordure de couverture de son point d'accès, le nœud mobile n'aura le droit qu'à un nombre limité de tentatives de Handover Anticipé dans un certain laps de temps.

5.4 Anticipation à posteriori

Dans une tout autre optique, nous pouvons envisager une solution de handover optimisé sans faire intervenir le nœud mobile dans la réalisation du protocole. Les nœuds mobiles étant très souvent des équipements miniaturisés fonctionnant de manière autonome sur batterie, une des raisons pour éviter l'implication du nœud mobile dans un protocole peut être l'économie d'energie électrique. Afin de permettre une optimisation, on se propose dans cette méthode d'utiliser uniquement le préfixe IPv6 contenu dans les Probe Request, sans mettre en place une anticipation au préalable.

Dans l'Anticipation à Posteriori (dont l'échelle de temps est représenté dans

la figure 5.3), le protocole de handover de niveau 2 n'est pas modifié : le nœud mobile commence un handover lorsqu'il sort de la portée de son point d'accès. La présence du préfixe IPv6 dans les Probe Response envoyés par les points d'accès vont permettre au nœud mobile de choisir son point d'accès avec un critère supplémentaire et également de préparer une nouvelle adresse temporaire en avance. Ainsi, à la manière du cas d'erreur de la méthode précédente, une fois que le handover de niveau 2 sera terminé, le nœud mobile pourra immédiatement envoyé un Binding Update pour mettre à jour sa localisation. Une fois de plus, la détection de duplication d'adresse pourra être faite en parallèle de la mise à jour.

5.5 Evaluation

Face aux difficultés techniques d'implémentation sur le matériel dont nous disposions et afin d'évaluer ce nouveau mécanisme dans différents scénarii, avec plusieurs nœuds mobiles, nous avons utilisé le simulateur SimulX présenté dans le chapitre 4. L'évaluation se découpe en deux parties. Dans un premier temps, les performances du Handover Anticipé seront présentées. Ces premières mesures concernent surtout la validation du bon fonctionnement du protocole dans des cas d'école. Dans la deuxième partie de l'évaluation, des scénarii plus complexes avec un plus grand nombre de nœuds seront présentés et évalués. Ces scénarii donneront lieu à la comparaison entre différentes optimisations du handover implémentées dans SimulX.

Pour l'ensemble de ces tests, les nœuds mobiles ne mettent à jour leur localisation qu'auprès de leur agent mère (ou point d'ancrage pour un domaine hiérarchique). L'utilisation de l'optimisation de routage aurait introduit des paramètres de test supplémentaires qui ne se révèlent pas utiles pour une première évaluation. Nous avons alors mesuré le temps de handover de niveau 3 comme étant le laps de temps compris entre le détachement du point d'accès courant (début de handover de niveau 2) jusqu'à la réception de l'acquittement (Binding Acknowledgement) de l'agent mère qui indique que la mise à jour de localisation a bien été effectuée. Le temps d'aller-retour entre les nœuds mobiles et l'agent mère est approximativement configuré à 40 ms par défaut (si ce temps est différent, sa valeur sera indiquée le cas échéant). Ce temps d'aller-retour correspond à des déplacements à l'intérieur d'un domaine, puisque comme nous l'avons vu dans le chapitre 3, une solution hiérarchique de gestion de la mobilité permettra toujours des temps d'aller-retours de cet ordre. Chaque valeur donnée dans la suite est la moyenne d'au moins 100 mesures. La variance a toujours appuyé la moyenne des résultats.

5.5.1 Evaluation du Handover Anticipé

La figure 5.4 représente l'échange de messages que nous avons observé pour établir une anticipation de handover lors d'un changement de point d'accès, et par la même occasion de sous-réseau IPv6. Comme décrit ci-dessus, à l'approche de la bordure de la cellule courante, le nœud mobile déclenche le handover anticipé au temps 5 sec. Ensuite le nœud mobile sonde les différents canaux et sélectionne un point d'accès cible. Au temps 5,23 sec. le nœud mobile retourne en mode associé avec son point d'accès et au temps 7,5 sec. il déclenche un handover de niveau 2. Comme il a découvert un point d'accès cible, il passe directement en phase d'authentification, puis d'association au temps 7,5008 sec. Comme il a réussi à se connecter à son point d'accès cible, son anticipation était correcte. Il peut alors envoyer un Binding Update pour annuler le Bicasting (temps 7,5014 sec.). Finalement, le temps d'interruption total aura été de 230 ms pour l'anticipation et 1,4 ms pour faire le handover de niveau 2. Dès la fin du handover de niveau 2, le nœud mobile peut établir des communications IP sur son nouveau point d'attache.

Le graphique 5.5 résume les temps de handovers de niveau 2 et 3 pour trois différentes solutions : Mobile IPv6, Handover Anticipé initié par le nœud mobile, et Handover Anticipé à Posteriori. Il doit être noté que la fréquence des RA dans le cas de Mobile IPv6 était comprise entre 30 et 70 ms, alors que pour les autres cas de figure, cette fréquence était comprise entre 200 et 600 secondes, comme le stipule le RFC 2461 [133], pour économiser de la bande passante. Nous remarquons sur cet histogramme que le temps de handover de niveau 2 avec le Handover Anticipé initié par le nœud mobile est très faible (1,7 ms) par rapport aux deux autres solutions (272 ms). Ce temps est aussi faible grâce à l'anticipation qui a pu être réalisée. Le temps de handover de niveau 3 est également inférieur pour le Handover Anticipé initié par le nœud mobile avec 42 ms contre 231 ms pour le handover Antcipé à Posteriori et 322 ms avec Mobile IPv6. On remarque que le temps de handover avec le Handover Anticipé initié par le nœud mobile est à peu près la somme du handover de niveau 2 plus le temps d'aller-retour avec l'agent mère. Cependant, durant ce temps d'aller-retour, le nœud mobile pourra déjà recevoir des paquets de données, à l'inverse des autres solutions. De manière semblable, avec le Handover Anticipé à Posteriori, le handover de niveau 3 prend 40 ms de plus que le handover de niveau 2. Ceci est dû au fait que la détection de mouvement est faite pendant le handover de niveau 2. Une fois ce dernier terminé, le nœud mobile pourra directement envoyer un Binding Update. Dans le cas de Mobile IPv6, le temps de détection de lien, évalué à environ 50 ms vient s'ajouter au temps total du handover de niveau 3.

A présent intéressons nous à l'impact du handover sur les flux. Le graphique 5.6

```
-----------   5000000: eq12 - commence un SCAN des APs voisins  --------
                        (Handover Anticipe)
5000338: eq12 - Emission d'une Probe Request
5000488: eq1 - Reception d'une Probe Request - On l'ajoute dans Flux[0]
5000538: eq1 - Emission d'une Probe Response
5000736: eq12 - Reception d'une Probe Response
5030000: eq12 - timeoutMinForProbe a expire: aucun AP ne nous a repondu
   -> On passe au canal suivant: 2
5030328: eq12 - Emission d'une Probe Request
5030478: eq2 - Reception d'une Probe Request - On l'ajoute dans Flux[0]
5030528: eq2 - Emission d'une Probe Response
5030726: eq12 - Reception d'une Probe Response
              -> AP2 sur le canal 2 et d'idLan 3 a ete ajoute dans le tableau
5060000: eq12 - timeoutMinForProbe est arrive a expiration
...
5210068: eq12 - Emission d'une Probe Request
5210218: eq2 - Reception d'une Probe Request - On l'ajoute dans Flux[0]
5210268: eq2 - Emission d'une Probe Response
5210466: eq12 - Reception d'une Probe Response
              -> AP2 a envoye une reponse supplementaire (etait deja dans le tableau)
5230000: eq12 - timeoutMaxForProbe est arrive a expiration,
on evalue les APs dans la portee:
L'AP 2 sur le canal 2 a ete selectionne comme cible avec idlan = 3

5230000 ----- FIN DE HDV ANT. au niveau 802.11 --------

Comme on etait en Anticipated Handover,
on privilegie le canal sur lequel
on avait trouve un AP et on passe en phase d'authentification
7500198: eq12 - Emission d'une Authentication Request
7500320: eq2 - Reception d'une Authentication Request
7500528: eq2 - Emission d'une Authentication Response
7500650: eq12 - Reception d'une Authentication Response
7500898: eq12 - Emission d'une Association Request
7501050: eq2 - Reception d'une Association Request
7501358: eq2 - Emission d'une Association Response
7501486: eq12 - Reception d'une Association Response
     ->  Fin de L2 handover, nouvel AP = 2

Bonne anticipation: on envoie le vrai BU pour arreter le bicasting
```

FIG. 5.4 – Fichier de log retracant l'anticipation d'un handover

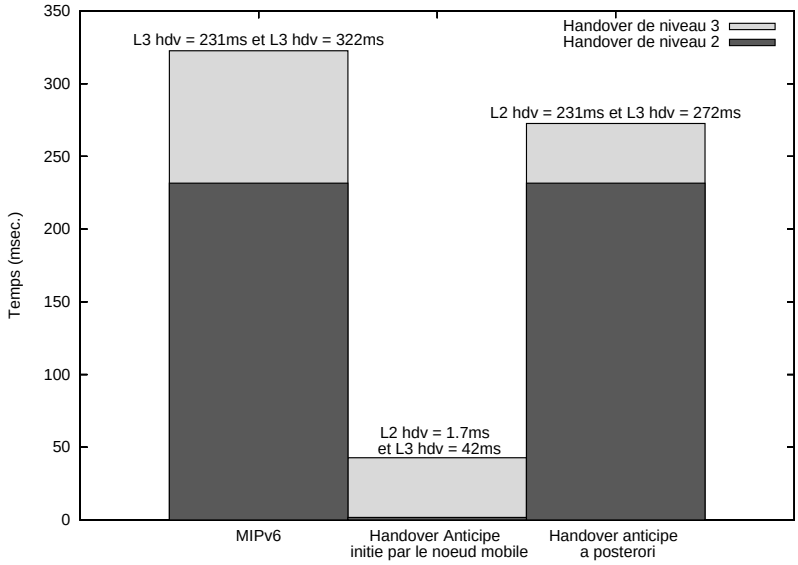


FIG. 5.5 – Temps de handover de niveau 2 et 3

représente la réception de flux sur un nœud mobile qui se déplace. Cinq types de flux ont été utilisés, avec des fréquences d'émission de paquets différentes : 20 ms, 50 ms, 100 ms, 200 ms et 500 ms, pour une taille de paquets de 1280 octets de données (sans les en-têtes). Pour des soucis de lisibilité, seul un handover est représenté sur le graphique et les numéros de séquence des flux de fréquence 100, 200 et 500 ms doivent être rapportés à l'axe des abscisses du haut. Pour les flux de fréquence 20, 50 et 100 ms, on remarque un laps de temps (autour de 100ms) pendant lequel le nœud mobile ne reçoit pas de paquets. Ce dernier correspond à la période de sondage pour mettre en place le Handover Anticipé. Cependant, les paquets ne sont pas perdus, mais stockés par les points d'accès. Pour le flux de fréquence 500 ms, l'anticipation a également lieu lors de l'arrivée d'un paquet de données. Comme l'indique le graphique, la réception est alors décalée dans le temps. Par contre, pour le flux de fréquence 200 ms, aucune perturbation n'est observée sur la réception des paquets de données. L'anticipation s'est faite entre deux réceptions de paquets, (par chance). Par ailleurs, on pourra noter que seul le temps pris pour l'anticipation implique un retard dans la réception des paquets (au maximum 4 à 5 paquets pour le flux de fréquence 20 ms). Les handovers de niveau 2 et 3 ne perturbent aucunement la réception des flux (le handover effectif a lieu à la 34ème seconde).

Finalement, nous avons voulu identifier quelques situations plus réalistes, avec des mouvements moins uniformes. La figure 5.7 illustre quatre scénarii dont les ré-

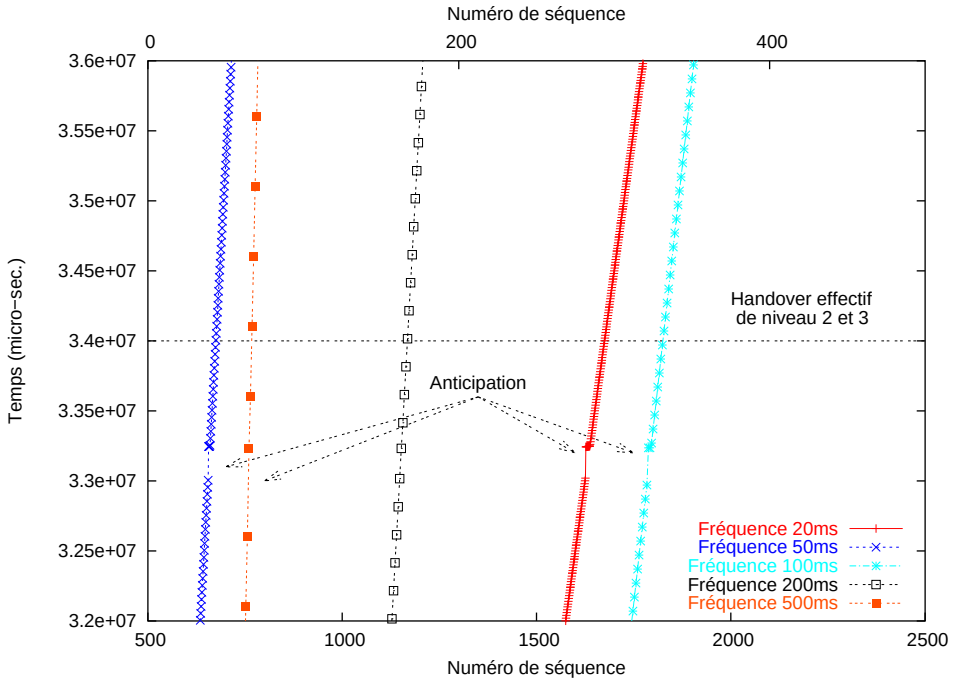


FIG. 5.6 – Réception de flux à fréquence variable en cours de handover

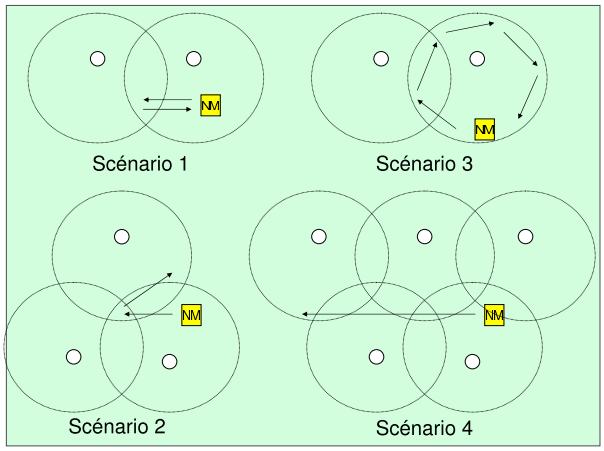


FIG. 5.7 – Scénarii pour le Handover Anticipé

sultats sont représentés dans la figure 5.8. L'histogramme 5.8 indique pour chaque scénario les temps de handover de niveau 2 et 3 (axe de droite), ainsi que le nombre de paquets générés par seconde de simulation (axe de gauche) pour le Handover Anticipé initié par le nœud mobile et Mobile IPv6. Une première constation est

que Mobile IPv6 génère 4 à 50 fois plus de paquets que le Handover Anticipé. Dans les scénarii 1 et 3, le nœud mobile entame des sondages de canaux à la recherche d'un nouveau point d'accès car il se déplace en bordure de sa cellule. Cependant, aucun handover n'est fait. Dans le scénario 2, le nœud mobile fait deux anticipations consécutives, et la deuxième s'avère être bonne, puisqu'il réalise le handover par la suite sur le point d'accès 3. On remarque encore une fois une importante différence de temps en défaveur de Mobile IPv6 pour les handovers de niveau 2 et 3.

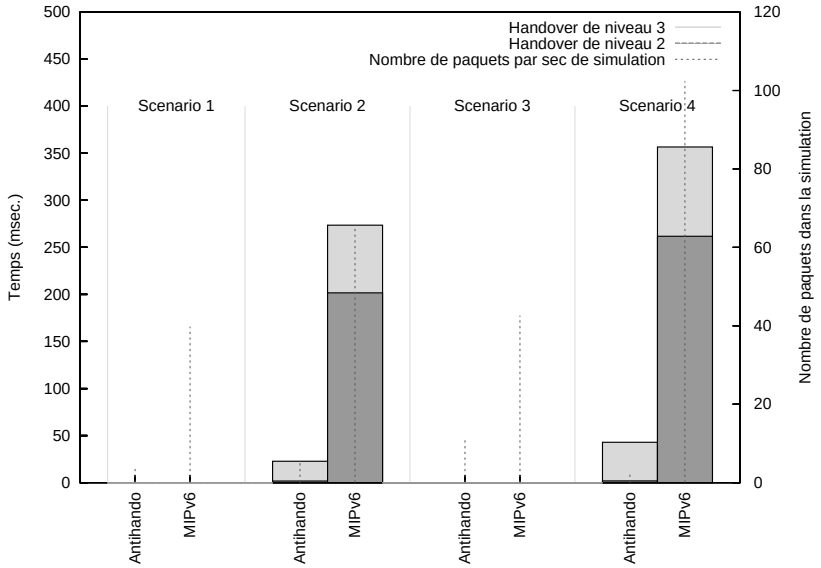


FIG. 5.8 – Résultats des scénarios 1 à 4

Le scénario 4 est encore plus intéressant. A chaque fois que le nœud mobile sort de la portée de son point d'accès, il a le choix entre plusieurs points d'accès. Or, les points d'accès 1 et 2 sont sur les mêmes sous-réseaux. Dans ce scénario, le nœud mobile réalise 4 handovers (niveau 2 et 3) avec Mobile IPv6 en passant par les points d'accès 1, 4, 2, 5 et 3. D'un autre côté, l'anticipation permet au nœud mobile de ne faire que 3 handovers de niveau 2 et 2 handovers de niveau 3, en passant par les points d'accès 1, 2, 5 et 3. Lorsqu'il s'approche du bord de la cellule du point d'accès 1, le nœud mobile découvre les points d'accès 2 et 3. Or le point d'accès 2 étant sur le même sous-réseau que le point d'accès 1 où est rattaché le nœud mobile, ce dernier préfèrera faire un handover de niveau 2 sur le point d'accès 2. Par contre, avec les seules informations disponibles pendant le sondage (i.e le canal, le préfixe IPv6 et l'identificateur du point d'accès), le nœud mobile n'anticipe pas de manière optimale son deuxième déplacement. S'il avait une connaissance de sa direction, et de la position géographique des points d'accès,

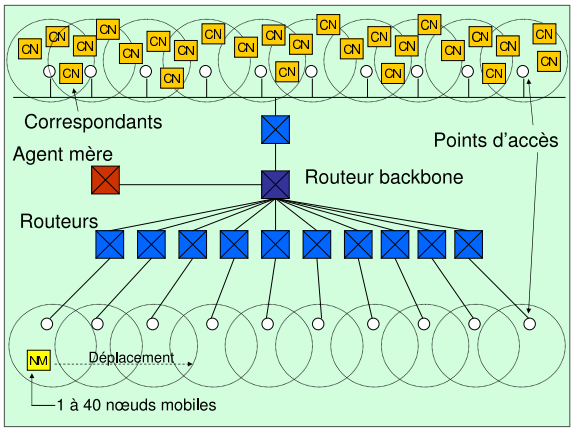


FIG. 5.9 – Représentation des nœuds pour la comparaison des mécanismes de gestion de la mobilité

il pourrait sans doute choisir immédiatement le point d'accès 5 au lieu du point d'accès 3 lorsqu'il sort de la portée du point d'accès 2.

5.5.2 Scénarii plus importants et comparaisons

Afin de réaliser des tests plus élaborés, sur des configurations plus importantes, nous avons également implémenté d'autres méthodes de gestion de handover, à savoir :

- La solution de base, qui utilise le protocole de découverte des voisins, le RFC 2461 [133], avec ses paramètres par défaut. Dans cette solution la réception d'un RA déclenchera la création et l'enregistrement d'une nouvelle adresse temporaire. Les RA sont envoyés à une fréquence comprise entre 200 et 600 secondes sans précision.
- Mobile IPv6 [84], c'est-à-dire la même solution qu'avec le RFC 2461, mais avec les paramètres optimisés pour le support de la mobilité. Les RA sont envoyés dans un intervalle compris entre 30 et 70 millisecondes.
- L'utilisation de déclencheurs de niveau 2 : sur détection de changement de lien, les nœuds mobiles envoient un RS. La réception de ce RS sur le routeur d'accès provoque l'émission d'un RA qui permet aux nœuds mobiles de découvrir le lien sur demande explicite.

La carte des simulations présentées dans cette section est illustrée dans la figure 5.9. Les nœuds mobiles partent tous de la même position et font un dépla-

Numéro de test	Nb de nœuds mobiles	Réception de flux	Nb de correspondants
Test 1	1	NON	0
Test 2	1	OUI	1
Test 3	10	NON	0
Test 4	10	OUI	10
Test 5	20	NON	0
Test 6	20	OUI	20
Test 7	30	NON	0
Test 8	30	OUI	30
Test 9	40	NON	0
Test 10	40	OUI	40

TAB. 5.1 – Configuration des tests du scénario 1

cement qui engendrera 10 handovers consécutifs. Chaque point d'accès est relié à un routeur, qui est lui-même relié à un routeur principal. A ce routeur principal sont reliés l'agent mère et les différents points d'accès auxquels sont rattachés les nœuds correspondants des mobiles.

Scenario 1

Dans un premier temps, nous avons mis en place 10 tests, en faisant varier le nombre de nœuds mobiles (de 1 à 40) et le nombre de flux applicatif (0 ou 1 par nœud mobile). Le tableau 5.1 indique les paramètres pour chaque numéro de test. Les nœuds mobiles débutent leur déplacement aléatoirement dans la première seconde de simulation. Dans les scénarii avec les flux (1 test sur 2), les nœuds correspondants commencent également la transmission du flux à un temps tiré aléatoirement dans la première seconde de simulation. Les paquets du flux considéré sont de 1280 octets, et émis à une fréquence de un toutes les 20 ms, comme indiqué par le codec G711 [70]. Nous avons choisi un tel flux, car il représente bien un type de flux multimédia temps réel, et car l'impact des handovers y sera certainement plus important que sur des flux moins gourmands en terme de bande passante.

Comme nous allons le voir, la configuration par défaut du RFC 2461 révèle de mauvaises performances, à tel point que dans certains cas les nœuds mobiles ne détectent pas le changement de lien, opération nécessaire pour effectuer des handovers de niveau 3. La fréquence des RA étant trop réduite, les nœuds mobiles ne reçoient jamais d'indication comme quoi le sous-réseau d'attachement a changé.

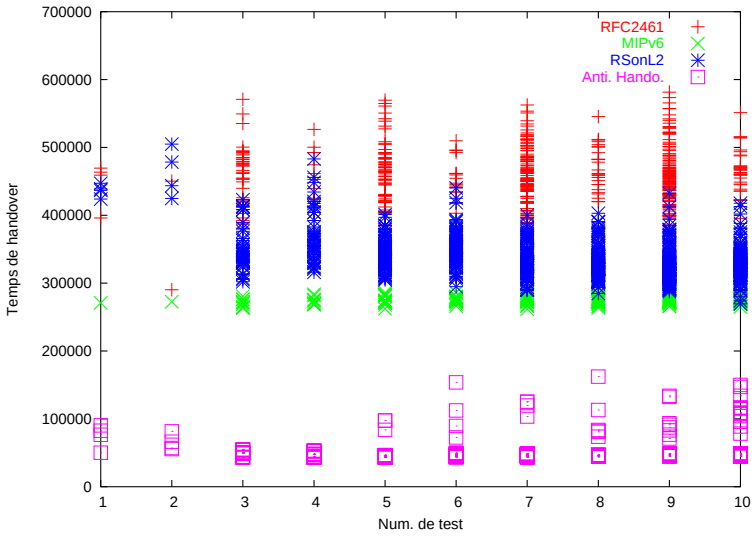
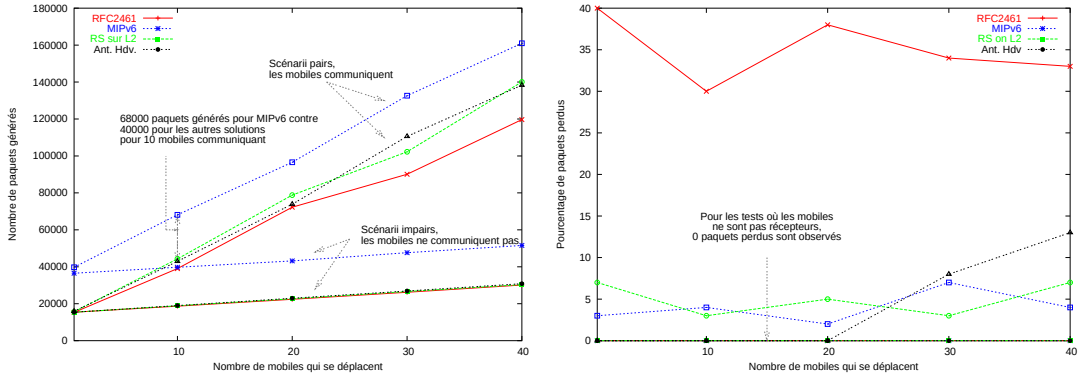


FIG. 5.10 – Temps de handover pour chaque noeud mobile

Ceci a pour conséquence qu'aucun paquet n'arrive à destination du nœud mobile. Afin de répondre à ce problème, nous avons fait varier la fréquence des RA dans les intervalles $[20s; 60s]$, $[2s; 6s]$ et enfin $[200ms; 600ms]$. Nous avons constaté que c'est uniquement avec l'intervalle $[200ms; 600ms]$ que le temps de handover de niveau 3 était du même ordre de grandeur que le temps obtenu avec les autres solutions. C'est pourquoi nous avons choisi cette fréquence de RA pour la solution RFC 2461 pour les mesures représentées sur le graphique 5.10.

La figure 5.10 représente la répartition des temps de handover mesurés sur chaque nœud mobile, pour chaque numéro de tests. On remarque très clairement que la solution de Handover Anticipé est meilleure quel que soit le nombre de nœuds mobiles dans la simulation. En effet, le handover de niveau 3 est mesuré entre 50 et 100 ms contre presque 300 ms pour Mobile IPv6, entre 300 et 400 ms avec l'émission du RS sur déclencheur de niveau 2 et plus de 400 ms pour le RFC 2461. Pour toutes les solutions, le temps de handover de niveau 3 est à peu près stable. Cependant on peut voir une légère tendance à la hausse pour la solution de Handover Anticipé : à partir du test 6 (20 nœuds mobiles communiquants), les temps de handover sont plus répartis dans le temps. Par contre, pour les trois autres solutions, on observe plutôt l'effet inverse : plus le nombre de nœuds mobiles augmente, plus les temps de handover de chacun sont similaires, avec une tendance à la baisse. Cette observation est d'autant plus flagrante dans le cas de l'émission du RS, où tous les handovers après le test 6 sont en-dessous de 400 ms. Ces observations seront complétées dans la suite, notamment par l'analyse des

graphiques de la figure 5.13.



(a) Nombre de paquets générés

(b) Pourcentage de paquets perdus

FIG. 5.11 – Statistiques de la simulation

Mais l'objectif de ces simulations était également d'observer les impacts de la gestion de la mobilité sur d'autres paramètres, pas uniquement sur le temps de handover. Les graphiques de la figure 5.11 montrent le nombre de paquets générés pendant toute la durée de la simulation (graphique 5.11(a)) et le pourcentage de paquets perdus (graphique 5.11(b)). Mobile IPv6 engendre près de 30.000 paquets supplémentaires par rapport aux autres méthodes, soit 15 à 100% de paquets supplémentaires, selon le nombre de nœuds mobiles. On peut même remarquer que la signalisation générée par Mobile IPv6 pour gérer le mouvement de 10 nœuds mobiles sur toute la carte est aussi importante que la signalisation générée par la réception du flux respectant le codec G711 sur ces mêmes 10 nœuds mobiles, lorsque leur mobilité est gérée par l'une des trois autres méthodes. Bien entendu cette comparaison vaut uniquement dans notre contexte, c'est-à-dire sur 10 déplacements pour une durée de 230 secondes. Mais ce résultat permet d'imaginer la charge supplémentaire induite par la fréquence élevée d'émission des RA introduite par Mobile IPv6. A partir du test 8 (30 nœuds mobiles), une différence apparaît également entre les trois autres méthodes. L'augmentation du nombre de paquets générés dans le cas du Handover Anticipé est due au Bi-casting demandé par les nœuds mobiles. Effectivement, à chaque anticipation, l'agent mère va dupliquer les paquets à destination du nœud mobile.

Le graphique 5.11(b) nous montre le pourcentage de paquets perdus pour chacune des solutions. Bien entendu, lorsque les nœuds ne communiquent pas (un scénario sur deux), le pourcentage de paquets perdus est très proche de 0 (seules

quelques trames de contrôle sont perdues). Par contre, lorsque les nœuds mobiles communiquent, on observe d'importantes différences entre les méthodes utilisées. D'un côté, on peut s'apercevoir une nouvelle fois que les valeurs par défaut du RFC 2461 sans mécanisme supplémentaire ne sont pas adaptées à une gestion de la mobilité, puisqu'on observe près de 40% de perte dans chaque test. La solution qui préconise l'émission d'un RS sur déclencheur de niveau 2 et Mobile IPv6 sont relativement stables avec une moyenne de 5% et 3% de perte respectivement. La solution de Handover Anticipé est très bonne jusqu'au test 8. Pour 30 et 40 nœuds mobiles communiquants, on observe 8 et 13% de perte. Ces pertes plus importantes concernent uniquement les trames utilisées pour réaliser l'anticipation ; lorsque beaucoup de nœuds mobiles se déplacent simultanément, ils arriveront à l'extrémité de l'aire de couverture du point d'accès courant dans le même temps. Ils débuteront alors le sondage des canaux voisins, qui demande une émission importante de messages, rapidement, puisque pendant ce temps là les noeuds mobiles ne pourront pas échanger de paquets de données. Comme nous le verrons plus loin, de nombreuses retransmissions seront nécessaires. Ces collisions seront tellement importantes qu'elle pourront même empêcher certains nœuds mobiles de réaliser leur anticipation.

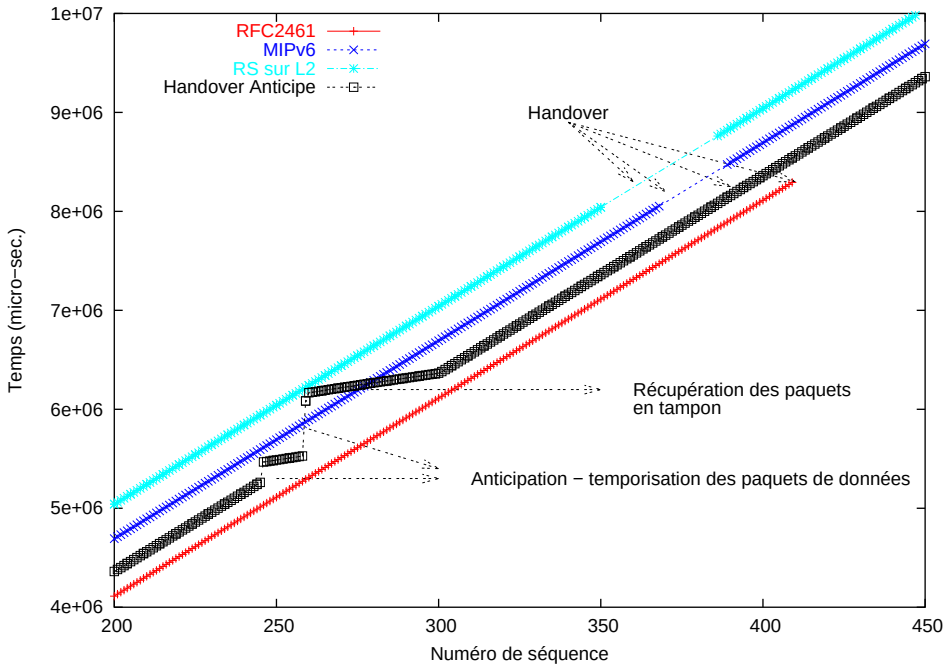


FIG. 5.12 – Impact d'un handover sur la réception d'un flux de fréquence 20ms

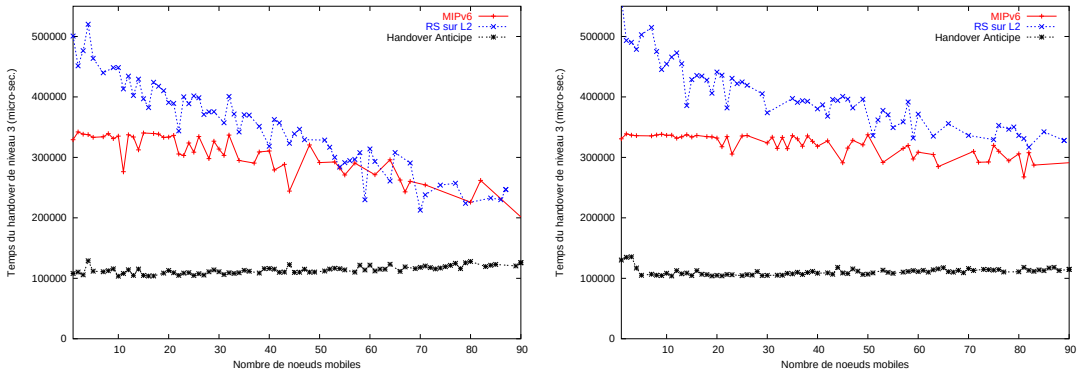
Finalement la figure 5.12 montre la réception du flux respectant le codec G711

sur un des nœuds mobiles qui se déplace. Seule la partie correspondante au premier handover est représentée ici en guise d'illustration. Tout d'abord, on remarque sur la courbe du bas qu'avec les valeurs par défaut du RFC 2461 et sans aucun autre mécanisme, le nœud mobile ne reçoît plus le flux après le handover. La courbe suivante montre la réception du flux avec l'utilisation de Mobile IPv6. Comme on peut le voir, un temps d'interruption est clairement visible, pendant lequel 29 paquets sont perdus. L'avant-dernière courbe représente la réception du flux lorsque le nœud mobile anticipe son déplacement. On voit apparaître deux irrégularités entre 5 et 6 secondes. C'est le temps nécessaire au nœud mobile pour sonder les points d'accès voisins. Lorsqu'il retourne en mode connecté (autour de 6 s.), il récupère les paquets de données qui avaient été mis en tampon sur le point d'accès (environ 10 paquets). Ensuite le nœud mobile effectue ses handovers de niveau 2 et 3 au temps 8,3 secondes, sans qu'aucune perturbation ne soit observée sur la réception du flux : comme on l'a déjà vu, le handover de niveau 2 dure à peine plus d'une milliseconde et grâce au Bi-casting mis en place lors de l'anticipation, le nœud mobile reçoit les paquets de données dans le nouveau sous-réseau dès la fin du handover de niveau 2. La dernière courbe représente l'impact du handover de niveau 2 et 3 lorsque le nœud mobile envoie un RS après le handover de niveau 2. 36 paquets sont perdus pendant le temps des deux handovers.

Cette première série de tests a éclairci plusieurs points. Tout d'abord, on peut constater que la solution de Handover Anticipé est très performante et permet une réelle amélioration des handovers de niveau 2 et 3. Cependant, le temps de sondage reste une partie critique du protocole et un compromis est nécessaire entre le temps pris pour sonder les différents canaux, et la qualité de découverte des points d'accès avoisinants. D'autre part, nous avons clairement mis en évidence que le RFC 2461 ne peut suffir à la gestion des mouvements des nœuds dans l'Internet. En prenant les valeurs par défaut, les nœuds mobiles mettent plus de temps à identifier un handover. L'utilisation de déclencheurs de niveau 2 semble être un bon compromis entre le RFC 2461 qui ne permet pas en fin de compte aux nœuds de se déplacer en maintenant leur communication, et Mobile IPv6, qui bien qu'un peu meilleur, requiert tout de même l'innondation du réseau par les RA. Au-delà de ces brèves conclusions, nous avons également vu que le Handover Anticipé laissait percevoir une légère baisse de performances lorsque le nombre d'utilisateurs croît, contrairement à Mobile IPv6 ou à l'utilisation de déclencheurs de niveau 2. Nous avons donc mis une deuxième série de tests en place qui est présentée ci-dessous.

Scénario 2

Afin d'identifier la robustesse des différentes méthodes quant au nombre de nœuds mobiles se dépl açant simultanément, nous avons lancé une deuxième batterie de tests. Nous avons alors utilisé la même topologie du réseau que représentée dans la figure 5.9, mais sans utiliser les nœuds correspondants. Nous avons fait se déplacer de 1 à 90 nœuds mobiles simultanément, afin d'observer les comportements des différents protocoles.



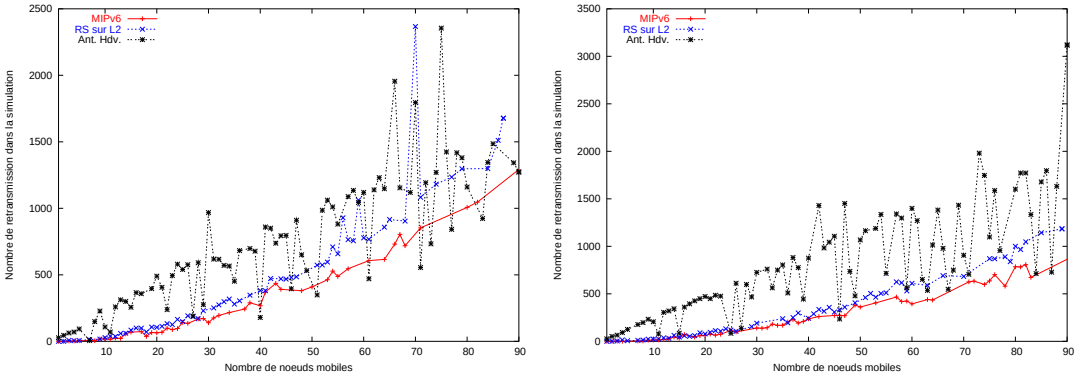
(a) Moyenne du temps de handover pour un déplacement dans la même seconde

(b) Moyenne du temps de handover pour un déplacement dans les deux secondes

FIG. 5.13 – Temps de handover en fonction du nombre de nœuds mobiles

Le graphique 5.13(a) montre le temps de handover moyen sur dix simulations pour des nœuds mobiles qui partent de la même position, à un temps choisi aléatoirement entre 0 et 1 seconde. Le graphique 5.13(b) représente également la moyenne des temps de handover lorsque le temps de début du déplacement est choisi aléatoirement entre 0 et 2 secondes. Ce que nous supposions dans la série de tests précédente apparaît clairement sur ces deux graphiques : plus le nombre de noeuds mobiles augmente, plus le temps de handover diminue avec Mobile IPv6 et le déclencheur de niveau 2, alors qu'il augmente légèrement avec le Handover Anticipé. La plus forte décroissance du temps de handover est observée avec l'utilisation des déclencheurs de niveau 2, qui passe de 450 ms avec moins de dix nœuds mobiles se déplaçant simultanément à 250 ms pour plus de 80 nœuds mobiles. A partir de 50 nœuds mobiles, le temps de handover avec utilisation des déclencheurs de niveau 2 devient même meilleur qu'avec Mobile IPv6. Ceci est dû au fait que lorsqu'un nœud mobile change de point d'accès et envoie un RS, le routeur d'accès attend un temps aléatoire entre 0 et 500 ms avant d'y répondre. Or, pendant le délai introduit

par le routeur d'accès avant d'émettre la réponse, plusieurs autres nœuds mobiles pourront s'associer avec le même point d'accès. Ils profiteront alors tous du RA demandé par le premier arrivé dans la cellule. Le fait qu'un plus grand nombre de nœud mobile profite du RA demandé par un autre nœud, réduira leur temps de handover de niveau 3 et fera alors baisser la moyenne. Par contre, le premier nœud qui est arrivé sur le point d'accès ne verra pas son temps de handover décroître.



(a) Nombre de retransmission pour un déplacement dans la même seconde

(b) Nombre de retransmission pour un déplacement dans les deux secondes

FIG. 5.14 – Nombre de rentransmission de trames en fonction du nombre de nœuds mobiles qui se déplacent simultanément

Afin de comprendre pourquoi le temps de handover de niveau 3 a tendance à légèrement augmenter avec le Handover Anticipé, analysons les graphiques de la figure 5.14 représentant le nombre de retransmissions totales observé dans chaque simulation. Il apparaît clairement ici que plus il y a de nœuds mobiles se déplaçant "simultanément", plus il y a de retransmissions. Effectivement, ils sortiront tous pratiquement en même temps de la cellule de leur point d'accès courant et entameront les opérations nécessaires au handover. Dans le cas du Handover Anticipé, c'est à ce moment précis qu'un trafic supplémentaire est généré. Cependant, il risque d'être rare de voir plus de 50 nœuds mobiles se déplacer dans la même seconde à partir du même point d'accès.

5.6 Conclusion

Nous venons de voir dans ce chapitre une optimisation de handover, aussi bien au niveau 2 qu'au niveau 3, dans les réseaux IP de type WLAN IEEE 802.11b. Après une analyse en profondeur de la norme IEEE 802.11b dans la section 2.2, nous avons proposé dans ce chapitre d'adapter les mécanismes disponibles dans la norme pour permettre au nœud mobile d'établir une anticipation de mouvement. Comme nous l'avons vu, cette anticipation, lorsqu'elle est exacte, réduit le temps de handover de niveau 2 à l'ordre de la milliseconde et permet la réception des paquets de données à la nouvelle localisation dès la fin du handover de niveau 2.

L'amélioration proposée est née de ce constat : dans le processus de handover de niveau 2 de 802.11, la majeure partie du temps est consacrée au sondage des points d'accès environnants. Dans la solution que nous avons appelée Handover Anticipé initié par le nœud mobile, ce sondage est réalisé avant la déconnexion effective du nœud mobile de son point d'accès courant, éventuellement en plusieurs fois (car pendant le sondage, le nœud mobile ne peut pas recevoir de paquets de données). Par ailleurs, pour permettre également d'anticiper les mouvements de niveau 3, nous proposons d'introduire le préfixe IPv6 dans les trames de Probe Response. Ainsi, lors de la phase de sondage anticipé, le nœud mobile pourra découvrir non seulement un point d'accès cible, mais également le préfixe IPv6 associé au nouveau lien.

Cette anticipation faite alors que le nœud mobile est toujours connecté à son point d'accès, lui permet de demander la duplication du trafic à la localisation cible. Ainsi, après le mouvement effectif sur le point d'accès cible (qui nécessite uniquement les phases d'authentification et d'association), le nœud mobile recevra ses paquets de données sans plus de délai. Dans le cas d'une mauvaise anticipation, le nœud mobile s'aperçoit que le point d'accès cible n'est pas accessible lorsque celui-ci ne répond pas à sa tentative d'authentification. Il entame alors une phase de sondage pour découvrir d'autres points d'accès. Le préfixe IPv6 contenu dans les trames de Probe Response permet au nœud mobile d'identifier le lien cible, et, en cas d'association, d'immédiatement envoyer une mise à jour de localisation.

Une erreur d'anticipation dans le cas précédent revient à une deuxième proposition que nous avons imaginée. Lorsqu'un protocole destiné à la gestion de la mobilité des nœuds est défini, il est recommandé de minimiser le trafic et les opérations supplémentaires nécessaires au protocole. C'est pourquoi dans la solution de Handover Anticipé à Posteriori, le nœud mobile réalise un handover tel que décrit dans la norme 802.11, mais découvre le préfixe IPv6 pendant ce dernier. Cette découverte lui permet de mettre à jour sa localisation dès la fin du handover

de niveau 2.

L'évaluation des handovers avec anticiation dans SimulX nous a révélé que l'ensemble du handover de niveau 3 dans le cas du Handover Anticipé initié par le nœud mobile n'avait *aucun* impact sur la réception de flux (pourtant de fréquence 20 ms). Cependant, un certain retard dans la réception des paquets se produit lors du sondage par anticipation (de l'ordre de 10 paquets). Cependant, le temps de sondage peut être modulé ; plus il sera court, plus le temps d'interruption des communications sera court, mais plus l'anticipation risquera d'être incomplète. Les derniers tests ont également montré que notre solution résistait de manière correcte à la mise à l'échelle, jusqu'à 50 nœuds mobiles qui se déplacent dans la même seconde vers le même point d'accès.

Au-delà de ce protocole d'anticipation, il apparaît aujourd'hui évident que de nombreuses optimisations sont possibles pour la gestion de la mobilité de niveau 3 dans les réseaux 802.11. Bien que la norme IEEE 802.11 connaisse un important succès, des améliorations et optimisations sont constamment proposées pour faire évoluer la norme sous tel ou tel aspect. Alors que du côté de la qualité de service l'élaboration de protocoles tels que eDCF [5] permettent de mettre en œuvre une priorité sur les flux, le groupe IEEE 802.21 [10] a pris en charge l'optimisation de handover. Les objectifs de ce groupe sont non seulement d'optimiser le handover de niveau 2, lorsque le nœud mobile change de point d'accès, mais également de fournir une information fiable et utile à la couche IP afin de mettre en œuvre d'autres optimisations pour le handover de niveau 3.

D'après l'expérience que nous avons pu acquérir sur la norme IEEE 802.11 et la mobilité IP, il nous semble primordial de permettre une interaction forte entre les couches 2 et 3 du modèle TCP/IP. Le processus de handover de niveau 3 peut bénéficier de multiples avantages grâce à une connaissance précise des états du système de niveau 2. Cette interaction ne va pas à l'encontre des objectifs premiers de la séparation en couches, qui est de permettre une interopérabilité et une abstraction des différents niveaux, mais s'oriente plutôt vers une extension des échanges d'informations.

Plus généralement, si le nœud mobile est muni d'interfaces multiples, la connaissance des états des protocoles de niveau 2 permet également une sélection plus fine des interfaces. Comme nous l'avons vu dans le chapitre 2, de nombreuses technologies de communication existent et toutes offrent des possibilités très différentes. Il est donc intéressant de pouvoir utiliser chaque interface au moment opportun, en fonction de leur disponibilité et de leurs caractéristiques théoriques. En plus de ces critères, si une interaction forte existe entre les couches 2, qui gèrent l'accès au médium et la mobilité entre points d'accès, et le niveau 3, qui contrôle la

répartition des flux sur les interfaces, alors une adaptation en temps réel, efficace et rapide sera possible. Le chapitre suivant traite de la mise en place d'échanges forts entre les différentes couches du modèle TCP/IP pour établir une sélection d'interfaces en connaissance de l'ensemble des critères de choix disponibles sur le terminal et pour optimiser les opérations de mobilité, comme la redirection de flux entre interfaces.

Chapitre 6

Gestion d'interfaces multiples : l'architecture MIMA

6.1 Introduction

Comme nous l'avons déjà dit, les équipements mobiles seront de plus en plus nombreux dans les années à venir. Certaines études prédisent même que le pourcentage d'équipements informatiques portables pourvus de moyens de communication sans fil sera de 90% en 2007 [71]. D'autres études annoncent que plus de 150 millions d'équipements réseaux seront équipés de puces WIFI avant la fin de l'années 2007 [14]. L'arrivée de tels équipements s'accompagnera d'une volonté d'être connecté où que l'on soit et par n'importe quel moyen [34].

Par ailleurs, notre étude présentée dans le chapitre 2 nous a révélé qu'aucune technologie de communication sans fil ne semble pouvoir s'imposer et devenir la technologie universelle des communications pour l'Internet. Chacune peut être considérée comme la meilleure étant donnée une certaine caractéristique, mais une limitation forte empêche une exclusivité d'utilisation : les réseaux cellulaires ne permettent qu'un débit de données faible pour un coût d'utilisation important, les réseaux IEEE 802.11 restent limités dans la portée, pour ne citer que deux d'entre elles. Ces considérations nous laissent donc penser que la clé des réseaux Nouvelle Génération sera l'intégration et l'utilisation optimisée de plusieurs technologies de communication.

La gestion des interfaces multiples sur un équipement mobile n'est pas clairement prise en compte par les différentes propositions de gestion de la mobi-

lité [21, 32, 188]. De ce fait, soit les différentes implémentations ne la gèrent pas, soit elle la gèrent chacune d'une manière différente, ce qui empêche toute hypothèse sur le comportement de l'équipement mobile.

A partir de ces constatations, il devient impératif d'étudier comment plusieurs interfaces réseau peuvent être intégrées dans le modèle existant de la structure d'un nœud mobile. Il s'agit ici de proposer une architecture complète pour le terminal qui favorisera l'intégration de plusieurs interfaces à tous les niveaux. Cette étude porte donc sur un modèle en couches afin de solutionner la prise en compte des différents moyens de communication, aussi bien du point de vue de l'utilisateur pour un contrôle du comportement du terminal, que dans les couches basses du système pour permettre une grande réactivité de l'ensemble du système à l'environnement courant. Nous avons donc choisi d'étendre les couches du modèle TCP/IP. L'ajout de nouveaux modules va permettre une plus forte interaction entre les différentes couches et une auto-adaptation selon de multiples critères : interfaces réseau disponibles, capacité des interfaces, préférences utilisateurs, profils d'utilisation, besoins des applications, etc. De plus, la modularité de l'architecture permettra une facilité d'extension, par l'ajout ultérieur de nouveaux modules ou la modification de certains d'entre eux.

Ces nouveaux modules peuvent être répartis en deux groupes. Un premier ensemble de modules est défini dans l'espace utilisateur du système. Ces modules concernent principalement l'adaptation des applications, et la gestion de profils, tels que profil d'accès réseau, ou préférence de l'utilisateur par rapport à l'utilisation des interfaces. A ce niveau, la réactivité du système est relativement lente (de l'ordre de la seconde). Seules les décisions globales sont prises par rapport aux choix d'utilisation des interfaces et comportement du terminal. Différents profils permettent également une plus grande souplesse lors de l'initialisation des interfaces, en permettant une configuration automatique. Les modules constituant cette partie du système interagissent à travers un module de communication entre cet espace utilisateur et le noyau du système.

Le deuxième grand groupe de modules est défini dans le noyau du système. Ces modules sont en charge d'appliquer les règles de plus haut niveau en fonction des disponibilités des interfaces réseau. Ces modules vont donc contrôler en permanence l'état des interfaces, et réagiront automatiquement et rapidement aux différents changements de connectivité. Ces modules permettent une adaptation rapide (de l'ordre de la milliseconde) et un contrôle total de la mobilité.

Enfin, notre approche pour les équipements mobile multi-interfaces ne se limite pas à la définition d'une nouvelle architecture de communication. Nous avons également défini les comportements que devrait suivre un équipement pourvu de

plusieurs moyens de communication. Nous regroupons ces comportements sous le terme : contrôle de la mobilité. Le contrôle de la mobilité porte non seulement sur la redirection entre interfaces afin de permettre une utilisation alternative des interfaces multiples, mais également sur les possibilités d'utilisation simultanée de plusieurs technologies de communication. En effet, une utilisation simultanée de plusieurs interfaces pourrait fortement améliorer la connectivité d'un nœud, en lui permettant de répartir différents flux sur ses interfaces. Ainsi nous avons défini un ensemble de comportements et les mécanismes de réalisation qui y sont associés, permettant à un nœud mobile de s'adapter de manière optimisée à son environnement réseau. Plus particulièrement, nous chercherons à mettre en œuvre l'ensemble des optimisations dont nous avons déjà parlé, à savoir l'utilisation des déclencheurs de niveau 2. De la sorte, notre architecture permettra d'optimiser aussi bien les handovers horizontaux que les handovers verticaux.

Ce chapitre est consacré à la présentation et l'évaluation de cette nouvelle architecture appelée *Multiple Interfaces Management Architecture* (MIMA). La section suivante introduit les objectifs qui ont motivé le développement d'une telle architecture. Ensuite, nous décrirons chaque module constituant MIMA, avant de voir les différents algorithmes de redirection proposés. Nous suggèrerons ensuite d'autres fonctionnalités qui peuvent être mises en place grâce à cette nouvelle architecture. Finalement, nous présenterons les performances de différentes redirections que nous avons obtenues par l'implémentation de l'architecture dans un système Linux avant de conclure ce chapitre.

6.2 Les objectifs

Un nœud mobile peut être pourvu de plusieurs technologies d'accès sans fil. Il peut notamment être amené à utiliser soit simultanément, soit alternativement l'ensemble de ses interfaces. Toutefois, suivant la technologie sans fil utilisée, il est fréquent de se retrouver hors de portée d'un point d'attachement. Afin d'éviter la rupture des communications via une interface en cours de déconnexion (ou déconnectée), un nœud mobile cherchera à rediriger les dites communications sur une autre interface active. Cependant, le support d'interfaces multiples sur un terminal mobile pose un certain nombre de problèmes ou d'interrogations :

- Quand et comment rediriger des flux de données entre les interfaces ?
- Doit-on rediriger tous les flux d'une interface donnée ou seulement certains ?
- Que se passe-t-il si deux technologies d'accès sont sur le même lien IPv6 ?
- Quelle politique de sélection d'interface est retenue (en fonction de la bande

passante, du coût) ?
- Comment utiliser plusieurs interfaces pour éviter totalement ou du moins minimiser les interruptions de communication ?
- Est-il envisageable de tirer parti des spécificités de chaque technologie sans fil pour augmenter le temps global de disponibilité d'un terminal ?

En réalité, dans le cadre des handovers verticaux de nombreuses questions surgissent pour lesquelles bien peu de réponses sont à ce jour apportées. Les spécifications de Mobile IPv6 [84] ne proposent actuellement aucun éclaircissement sur le sujet et les implémentations de la mobilité existantes soit ne gèrent pas les handovers verticaux, soit les gèrent de manière très approximative. Ceci est d'autant plus problématique que d'une implémentation à l'autre, les comportements ne sont pas les mêmes, ce qui annihile toute possibilité d'offrir de nouvelles fonctionnalités dont pourraient tirer parti les applications grâce à ces capacités de communications multiples. Nos travaux sur les terminaux mobiles multi-interfaces proposent donc de trouver de nouvelles réponses à ces questions.

6.2.1 Un exemple concret

Afin de mieux comprendre nos ambitions, voici un exemple concret de scénario qui a motivé le développement d'une nouvelle architecture de gestion d'interfaces multiples. Quand un employé de bureau démarre son ordinateur portable, un premier ensemble d'interfaces doit être démarré. Etant dans son environnement de travail, l'utilisateur a configuré son terminal pour que les interfaces Ethernet et 802.11b soient mises en route. Les informations stockées dans différents profils d'accès seront utilisées pour obtenir une connexion automatique de ces interfaces. Une application de vidéo-conférence est alors démarrée et le système va établir une liste d'interface préférée pour cette application, selon des critères comme les préférences de l'utilisateur et la bande passante des interfaces. Dans ce cas de figure, admettons que l'interface Ethernet soit préférée aussi bien pour la vidéo que pour l'audio. De plus, une règle dans le terminal peut indiquer que si l'applications vidéo-conférence est lancée, il faut préparer l'interface GPRS pour assurer une continuité de la communication en cas de déconnexion des interfaces utilisées. Une connexion GPRS est alors établie sur le terminal.

Quand l'employé quitte son bureau, il débranche son interface Ethernet. Afin de poursuivre l'application de vidéo-conférence, la liste d'interfaces préférées est consultée. Faisons l'hypothèse que cette dernière indique qu'en cas d'indisponibilité de l'interface Ethernet, c'est l'interface 802.11b qui doit être utilisée pour la vidéo et l'interface GPRS pour l'audio. Si le système peut être informé de ce

changement d'interfaces, et de ce fait du changement des capacités, l'application de vidéo-conférence pourra s'adapter en diminuant la bande passante nécessaire (vidéo plus compressée par exemple). Lorsque l'employé sort de l'immeuble, il perdra sa connexion 802.11b avec les points d'accès à l'intérieur de son bureau. La seule interface qui sera encore disponible à ce moment là sera l'interface GPRS. Suite à une évaluation des capacités des interfaces disponibles, le système pourra conclure qu'il n'est pas nécessaire de rediriger le flux vidéo sur l'interface GPRS, celle-ci n'offrant pas d'assez bonnes performances pour supporter le flux. Seul le flux audio sera reçu sur le terminal.

Cet exemple simple fait apparaître de nombreux problèmes : comment préconfigurer un terminal pour qu'il s'auto-configure en fonction de son environnement, comment prendre en compte des préférences utilisateurs, comment définir et appliquer des comportements par défaut, comment rediriger les flux entre les interfaces ou encore comment adapter les applications en fonction des capacités du terminal à un instant donné. Nous verrons dans la suite comment mettre en place un tel système pour la gestion d'interfaces multiples.

6.2.2 Les hypothèses

Afin de mieux comprendre les tenants et les aboutissants de la gestion d'interfaces multiples sur un terminal, une analyse des différents cas de figure est nécessaire. En effet, différents comportements seront à mettre en place selon le nombre de points d'attachement du nœud mobile, leur connexion à l'Internet et la configuration du nœud mobile. L'hypothèse forte de départ est l'utilisation de Mobile IPv6 [84]. Une des raisons principales est que Mobile IPv6 est la solution de base de la gestion de la mobilité des nœuds dans l'Internet. Ce choix nous permettra de profiter du développement massif de ce protocole dans l'Internet pour permettre une forte interopérabilité de nos mécanismes. Bien qu'on ne s'interesse pas exclusivement à des nœuds mobiles puisque nous cherchons à contrôler l'utilisation d'interfaces multiples, les mécanismes de Mobile IPv6 pourront aisément être utilisés pour atteindre nos objectifs.

Au niveau de la terminologie, nous considérons qu'une interface est disponible sur un terminal pour des communications lorsque cette interface fournit une connectivité de niveau IP. Une telle connectivité est fournie lorsque l'interface a accès à un médium de communication et a terminé sa configuration. D'une part, l'accès au médium pour une interface sans fil est établi lorsque l'interface est en mode connecté avec son point d'accès et que l'équipement a été authentifié. Pour une interface filaire, l'accès au médium est généralement garanti dès lors qu'un

câble est branché dans le terminal. D'autre part, la configuration consiste en l'attribution d'une adresse IPv6 unique et la connaissance d'un routeur d'accès par défaut lorsque le terminal est directement connecté à l'Internet. Lorsque le nœud mobile doit utiliser une connexion point à point [49], la configuration est considérée acquise lorsque le tunnel nécessaire pour le transfert des données a été mis en place (eg. cas pour une interface GPRS).

L'adressage est l'élément clé de la gestion des interfaces multiples. C'est grâce à la sélection d'adresses et à leur manipulation que des redirections entre interfaces pourront être mises en place. Plus particulièrement, lorsqu'un nœud mobile a plusieurs interfaces, il est possible qu'il ait également plusieurs adresses principales (ou adresses mères). Le fait d'avoir plusieurs adresses mères peut se révéler être une solution de gestion d'interfaces multiples, mais il n'est pas certain que tout nœud mobile puisse décider du nombre de ses adresses mères, puisque celles-ci sont généralement distribuées par l'entité en charge du réseau principal. Ainsi, nous préférons prendre le nombre d'adresses mères comme un paramètre dans la gestion d'interfaces multiples.

La topologie du réseau auquel le nœud mobile est rattaché est également un paramètre primordial. Le préfixe IPv6 et le routeur par défaut sont les seuls identificateurs de lien utilisables pour le moment. Ces paramètres étant capitaux dans la gestion d'interfaces multiples, il sera nécessaire de connaître cette information à tout moment afin d'optimiser la redirection entre interfaces. Nous verrons par la suite comment acquérir et utiliser cette information.

6.2.3 Le contrôle des redirections

L'objectif du développement d'une architecture pour le terminal est double. D'une part il s'agit de prendre en compte les contraintes liées aux capacités de chacune des interfaces, les préférences utilisateurs et les différents profils de comportement du terminal souhaité et d'autre part de mettre en place des mécanismes permettant la mise en œuvre d'utilisation d'interfaces multiples et de redirections entre interfaces. Plus particulièrement, les redirections de flux entre interfaces sur un terminal mobile pourront intervenir dans les situations suivantes :

- Une interface en cours d'utilisation devient indisponible suite à un déplacement hors portée du point d'accès courant par exemple. Le nœud mobile pourra alors tirer avantage du fait de posséder plusieurs interfaces et voudra probablement rediriger l'ensemble ou du moins un sous-ensemble des flux utilisant l'interface indisponible sur une autre interface.

- Une interface devient disponible. Dans ce cas, les préférences sur les interface peuvent indiquer que cette interface est préférée pour un (ou plusieurs) flux utilisant actuellement une autre interface. Il sera alors éventuellement nécessaire de rediriger un ensemble de flux sur cette nouvelle interface.
- Lorsque le nœud mobile utilise une interface sans fil, il peut se déplacer tout en communiquant. Cependant, un déplacement peut provoquer un handover si le nœud mobile se déplace hors portée de son point d'accès courant. Or un handover implique toujours un temps de latence pendant lequel le nœud mobile ne peut pas recevoir ni envoyer de flux par l'interface. Le fait de posséder plusieurs interfaces permet au nœud mobile de rediriger temporairement un ensemble de flux d'une interface en cours de procédure de handover sur une autre interface disponible du système. Une fois que le handover est terminé, l'ensemble des flux peut être à nouveau redirigé sur l'interface initiale. Si cette redirection peut se faire en avance du handover, aucune perte de paquets ne sera observée.
- Les conditions réseaux / capacités des interfaces changent. Le nœud mobile peut constater une baisse dans les performances d'une interface. Dans cette situation, il peut s'avérer qu'une autre interface que celle utilisée pour un ensemble de flux devienne "meilleure" du point de vue des préférences du système. Dans ce cas là, il sera alors judicieux de rediriger cet ensemble sur l'interface de plus forte capacité. Un exemple concret de cette situation est une interface 802.11a : le débit brut peut chuter rapidement à mesure que le nœud mobile s'éloigne de son point d'accès. Passé un certain seuil, cette interface offrira de moins bonnes performances en terme de débit de données qu'une interface 802.11b.

6.2.4 La granularité de la mobilité

Maintenant que nous avons vu dans quelles circonstances des redirections entre interfaces réseau peuvent avoir lieu, il est intéressant d'étudier la granularité des redirections, ou plus génériquement de la mobilité. Il ne s'agit pas ici de décrire les mécanismes de redirection, mais d'étudier les concepts liés aux redirections sur un terminal multi-interfaces. Un nœud mobile aura très certainement plusieurs correspondants et plusieurs communications à un instant donné. Or, une redirection pourra ne concerner qu'un sous-ensemble de correspondants ou de flux, selon des règles de préférences de répartition de flux sur les interfaces. Ainsi, comme nous l'avons décrit dans [126, 127], un certain degré de raffinement peut être mis en place pour sélectionner uniquement un sous-ensemble de flux à rediriger. La granularité des répartitions / redirections des flux sur les différentes interfaces d'un

nœud mobile peut être classifiée de la manière suivante :

1. Utilisation alternative des interfaces. Dans cette situation, le contrôle de la mobilité verticale est très restreint : une simple classification des interfaces est nécessaire pour sélectionner quelle interface utiliser. Lorsque celle-ci devient indisponible ou que les préférences changent, l'ensemble des flux seront redirigés sur une autre interface.
2. Utilisation simultanée de plusieurs interfaces selon le correspondant. Des préférences par correspondant pourront être mises en place afin d'utiliser différentes interfaces avec des correspondants différents. A ce niveau de granularité, les préférences pourront même être définies selon le type du flux, et appliquées uniquement si le nœud mobile n'utilise pas déjà une autre interface pour le même correspondant. Par exemple, le nœud mobile pourra préférer l'interface Ethernet pour la réception de vidéo à la demande et l'interface 802.11b pour le transfert de fichier. Si les flux sont établis avec des correspondants différents, le nœud mobile pourra respecter ses préférences. Autrement, si les deux flux sont échangés avec le même correspondant, l'une des deux préférences ne pourra être respectée. Il faudra alors déterminer quelle règle est prioritaire. Ce niveau de granularité est très intéressant puisqu'il ne requiert aucune modification du protocole Mobile IPv6 : il suffira d'enregistrer avec chaque correspondant l'adresse temporaire allouée à l'interface préférée pour le type de flux.
3. Utilisation simultanée de plusieurs interfaces selon le type de flux. Cette granularité rejoint la précédente en terme de préférence entre flux et interfaces : une liste d'interfaces préférées pourra être choisie en fonction du flux. Mais dans cette situation, l'interface préférée sélectionnée pourra être utilisée même s'il s'agit du même correspondant avec lequel le nœud mobile a déjà des communications en cours sur d'autres interfaces. Par contre, cette solution nécessite le transport de nouvelles options dans les messages de mise à jour afin d'identifier pour quel(s) flux une adresse temporaire est mise à jour. Alors que les documents [167, 90] mettent en avant le traitement des nouvelles options par les nœuds correspondants, nous avons proposé une solution centralisée sur l'agent mère dans [122]. Cette proposition est détaillée plus longuement dans la section 6.5.2.
4. Utilisation simultanée de plusieurs interfaces pour un même flux. Cette fonctionnalité permet une utilisation très évoluée d'interfaces multiples sur un terminal : le nœud mobile pourra partager la charge d'un flux sur plusieurs interfaces. Cependant, ce mécanisme ne convient pas pour tous les types de flux, étant donné qu'il va introduire des délais différents sur les différents chemins que prendront les paquets de données. Pour mettre en place cette

méthode, les correspondants devront être modifiés pour pouvoir enregistrer plusieurs adresses temporaires pour le même flux, ainsi que pour mémoriser la proportion de paquets devant être envoyée à chaque localisation. Les détails de cette mise en œuvre sont décrits dans [126].

Bien entendu, le raffinement du contrôle de la répartition et des redirections de communications entraîne une complexité grandissante de la mise en œuvre. Alors que pour une utilisation alternative Mobile IPv6 est suffisant (sous réserve d'une utilisation spécifique du protocole), de nombreuses modifications sont nécessaires pour mettre en place des redirections plus complexes telles que l'utilisation simultanée de plusieurs interfaces pour un même flux. Un choix est donc à faire quant à la complexité du protocole et les possibilités de ce dernier. Dans notre implémentation, nous avons commencé par vérifier le bon comportement des deux premiers points, à savoir l'utilisation alternative et une mobilité que nous avons appelée *mobilité par correspondant.*

Afin d'atteindre ces objectifs, il est non seulement nécessaire de mettre en œuvre des mécanismes permettant les répartitions et redirections citées ci-dessus, mais également d'offrir à l'utilisateur une maîtrise accrue du comportement général de son terminal. De plus, l'adaptation des applications nécessite la remontée d'informations concernant les capacités courantes du réseau. C'est pourquoi, nous proposons une nouvelle architecture complète pour le terminal qui augmente le modèle de la couche TCP/IP par l'ajout de différents modules à différents niveaux. Ces nouveaux modules permettent l'optimisation de la gestion de la mobilité, une interaction entre les différents niveaux du modèle TCP/IP et une adaptabilité de tout le système en temps réel. La section suivante décrit cette nouvelle architecture pour les nœuds munis d'interfaces multiples.

6.3 L'architecture MIMA

A partir des expérimentations vues dans les précédents chapitres et des constatations décrites ci-dessus, il apparaît important de définir une nouvelle architecture pour le nœud mobile offrant à la fois une flexibilité et une extensibilité pour l'ensemble des normes de communication existantes et futures mais également des spécificités permettant de palier aux limites des propositions actuelles. La nouvelle architecture est composée de plusieurs modules que nous regroupons sous le terme de MIMA pour *Multiple Interfaces Management Architecture.* Une illustration simplifiée de cette architecture est donnée dans la figure 6.1.

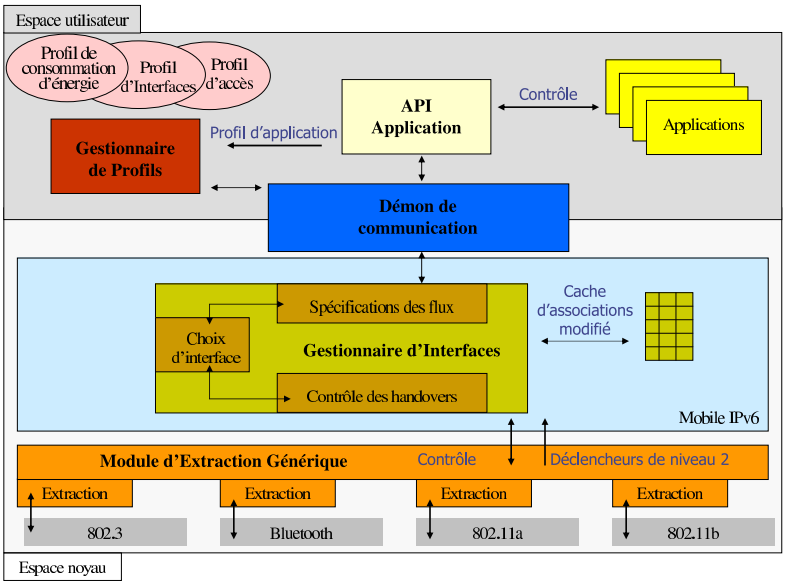


FIG. 6.1 – Les différents modules constituant MIMA

Afin de respecter nos objectifs de base, nous avons découpé notre architecture en modules. Chaque module a des tâches bien identifiées et suffisamment segmentées afin de rester le plus extensible possible. La séparation en modules apporte un avantage double : d'une part cela permet une intégration simplifiée dans le modèle existant TCP/IP, et d'autre part il laisse la porte ouverte pour des extensions futures par l'ajout de nouvelles briques. Les sous-sections suivantes présentent chacun des modules constituant MIMA.

6.3.1 Le module d'extraction

Le rôle du module d'Extraction est de surveiller les états des différentes interfaces réseau. Il se situe dans le noyau du système, entre la couche 2 et la couche 3 du modèle TCP/IP comme le représente la figure 6.1. Afin de cacher les spécificités des différentes technologies sous-jacentes, et pour proposer une interface générique aux couches supérieures, le module d'Extraction est lui-même scindé en deux parties : une partie générique et une partie spécifique à chaque technologie d'accès (IEEE 802.11, Bluetooth, Ethernet, etc.). Le fait de séparer la partie générique de la partie spécifique permet notamment l'intégration future de nouvelles technologies de communication ; seule la partie spécifique sera à implémenter pour bénéficier d'une nouvelle interface réseau.

Ce module a donc à charge de remonter les informations relatives aux technologies sous-jacentes, et de détecter les différents événements sur les interfaces, comme le début ou la fin d'un déplacement, ou les changements de qualité de lien avec un point d'accès. Pour remplir ce rôle, il utilise des messages connus sous le nom de *déclencheurs de niveau 2* [86]. Nous avons proposé une définition complète et générique des déclencheurs de niveau 2 potentiellement disponibles sur un terminal dans les publications [124, 35, 195]. Notamment, la spécification [195] est considérée comme un thème majeur du groupe de travail DNA [56] à l'IETF. Ces déclencheurs de niveau 2 permettent aux couches supérieures de connaître les réseaux disponibles et leurs capacités, ce qui permet une plus vive réactivité, comme nous l'avons montré dans [145, 138] dans le cas de handover entre réseaux GPRS et 802.11b. Toutes les informations des couches inférieures peuvent être utilisées à différentes fins, comme la détection rapide de lien (voir chapitre 5), ou comme critère de sélection d'interfaces réseau. Nous considérons deux types de déclencheurs : ceux discutés de manière générique à l'IETF [86] et une nouvelle famille que nous avons introduite afin de rediriger des flux entre plusieurs interfaces. Les déclencheurs de la première famille sont basés sur des trames de contrôle de niveaux 2, comme l'établissement d'une connexion avec un nouveau point d'attachement ou la perte de cette connexion. Nous introduisons également une nouvelle catégorie de déclencheurs qui sont basés sur les caractéristiques physiques des nœuds mobiles et des points d'attachement dans le voisinage d'un nœud mobile [124, 35]. Le tableau 6.1 retrace certains des déclencheurs retenus, et quelques exemples d'application.

La seconde partie du module d'extraction est composée d'un module générique pour l'ensemble des interfaces de l'équipement. Les informations reçues par les différents modules spécifiques à chaque technologie de communication sont converties en informations génériques et transmises au module appelé *Gestionnaire d'Interfaces* (voir sous-section suivante). L'extensibilité fut la raison principale de la scission du module d'Extraction en deux parties. En effet, lors de l'apparition d'une nouvelle norme de communication, les mêmes messages seront remontés aux couches supérieures même si la nouvelle technologie a un tout autre mode de fonctionnement.

Identification du réseau		
ID de réseau	Identification du réseau courant	ISP, SSID
Médium	Type de support physique	Ethernet
ID logique	Identificateur logique de l'interface	eth0, wifi0
ID physique	Identificateur physique de l'interface	Adresse MAC
ID du point d'attache	Adresse physique du point d'accès	Adresse MAC
Adresse(s) IP	Liste d'adresses allouées à l'interface	Adresse(s) IP
Etats des interfaces		
Statut de l'interface	Présence d'une interface	vrai ou faux
Statut du lien	Disponibilité d'une connexion réseau	vrai ou faux
Suspension de l'interface	Interface en mode veille	vrai ou faux
Début de Handover	Le mobile débute un handover	vrai ou faux
Fin de Handover	Le mobile termine un handover	vrai ou faux
Etats du lien		
Niveau de bruit	Niveau de bruit dans les paquets reçus	dBm
Erreur sur réception	Trames erronées à la réception	Entier
Erreur sur transmission	Erreur de transmission	Entier
Qualité du signal	Niveau d'intensité de signal	dBm
RSSI normalisé	RSSI normalisé	$[0; 100]$
Charge du point d'accès	Nombre de mobiles rattachés	Entier
Charge en réception	Nombre de bits reçus	Entier
Charge en transmission	Nombre de bits envoyés	Entier
Débit courant	Débit brut théorique courant	Mb/s
Débits sup. par l'AP	Débits configurés sur le point d'accès	Mb/s
Débits sup. par le NM	Débits configurés sur le mobile	Mb/s
Fréquence des RA	Fréquence d'émission des RA	Secondes
Fréquence des Beacon	Fréquence d'émission des *Beacon*	Secondes
Fragmentation	Seuil de Fragmentation de niveau 2	bits
Economie d'énergie	Mode opératoire d'économie d'énergie	Veille, active
Puissance de réception	Puissance de réception	mW
Puissance de trans.	Puissance de transmission	mW
Securité		
Capacité du logiciel	Sécurité logicielle de l'interface	Nom
Authentification	Authentification courante	Protocole

TAB. 6.1 – Implémentation des messages génériques pour des interfaces hétérogènes

6.3.2 Le gestionnaire d'interfaces

Le module nommé *Gestionnaire d'Interfaces* est la pièce centrale de notre architecture. C'est notamment lui qui prendra la décision du basculement des communications entre les interfaces. Cette décision est prise en fonction de nombreux critères. Ceux-ci sont à la fois donnés par l'utilisateur de l'équipement ou l'administrateur du terminal mais aussi par différents événements réseaux et physiques survenus sur les différentes interfaces de communication. Le Gestionnaire d'Interfaces réévalue périodiquement l'association entre les flux et les interfaces. Cette réévaluation est effectuée à l'aide des informations fournies par le module d'Extraction sur les caractéristiques courantes des interfaces (débit, nombre de clients sur un point d'accès, etc.). Cette même évaluation est réalisée dès la réception d'événements majeurs comme la perte de connectivité, le lancement ou l'arrêt d'un flux de données, etc.

Afin de maintenir un ensemble d'informations sur les interfaces, le Gestionnaire d'Interfaces utilise une liste chaînée ; chaque structure de cette liste contient toute l'information qui servira à sélectionner une interface pour un flux et à effectuer les redirections entre interfaces. Ces informations concernent la disponibilité de l'interface (présence ou non dans le terminal, rattachée à un point d'accès...), les paramètres IPv6 et les capacités courantes offertes par l'interface. Selon la granularité de la mobilité mise en place (voir section 6.2.4), et les informations courantes sur les interfaces, différentes méthodes de redirection sont utilisées. Ces méthodes de redirection sont détaillées dans la section 6.4.3.

La redirection entre interfaces est effectuée grâce à l'enregistrement sur les correspondants de l'adresse temporaire allouée à l'interface sélectionnée. En outre, le Gestionnaire d'Interfaces doit s'assurer qu'une adresse temporaire principale est enregistrée en permanence sur l'agent mère du nœud mobile. Cette adresse temporaire principale doit être l'adresse globale utilisée par l'interface par défaut du terminal, i.e sur laquelle seront reçus les nouveaux flux. Ensuite la sélection d'interface se traduit par la sélection d'adresse temporaire par le Gestionnaire d'Interfaces.

Les préférences sur les interfaces sont maintenues par différentes listes dans le Gestionnaire d'Interfaces. Un flux est identifié par le quintuplet des adresses source et destination, des numéros de port source et destination et du numéro de protocole. Une préférence est une liste d'interfaces, classées dans l'ordre d'utilisation souhaitée (i.e si la première interface de la liste n'est pas disponible, l'interface suivante est utilisée pour le flux en question). Une telle préférence peut être définie pour un flux précis où les cinq champs du quintuplet sont renseignés, ou

alors uniquement pour un sous-ensemble du quintuplet, comme seul un numéro de port. Une règle de préférence où seul le numéro de port est rempli s'appliquera à toutes les communications du nœud mobile utilisant ce numéro de port, quelque soit le correspondant. La possibilité de pouvoir uniquement renseigner une partie du quintuplet permet de mettre en place des règles avant la réception ou l'émission du flux en question. La règle de préférence appliquée pour un flux sera celle qui renseigne le plus de paramètres du flux. La règle par défaut du terminal sera celle où tous les champs du quintuplet ne sont pas renseignés.

Si le Gestionnaire d'Interfaces met en place une mobilité par correspondant (voir 6.2.4), des conflits pourront éventuellement être observés : si une règle de préférence préconise l'utilisation de l'interface Ethernet pour le port 25 et l'interface Bluetooth pour le port 22, et si des flux utilisant ces numéros de port sont échangés avec le même correspondant, le Gestionnaire d'Interfaces devra choisir quelle règle respecter. Une préférence globale sur les flux indique alors au Gestionnaire d'Interfaces quelle règle appliquer. Cette préférence globale est simplement un classement des règles sur les flux, pour identifier quelle règle doit être considérée en cas de conflit.

6.3.3 Le Démon de Communication

Le Démon de Communication n'introduit pas spécifiquement de nouvelles fonctionnalités à MIMA, mais a été développé dans un souci de simplification d'implémentation. Effectivement, ce module est chargé des échanges entre les modules de l'espace noyau du système et les modules de l'espace utilisateur. Par cette interface de communication, chaque partie du système peut être développée relativement indépendamment. Les différentes requêtes peuvent être temporisées ou simulées en phase de développement, ce qui permet une importante aisance au niveau du codage. Le développement de cette architecture faisant partie du projet RNRT Cyberté intégrant plusieurs équipes de recherche en France[1], le Démon de Communication s'est révélé plus qu'utile lors de la mise en commun des implémentations des différents modules composant l'architecture.

D'autre part, le Démon de Communication permet une extension intéressante de MIMA. Comme nous l'avons vu, notre architecture est orientée "contrôle par le terminal", c'est-à-dire que les préférences, et décisions sur les comportements sont uniquement prises sur le terminal. Or, il peut s'avérer intéressant de mettre en place un système de contrôle par le réseau. On imagine aisément qu'une entité

[1]Les équipes de recherche impliquées dans le projet RNRT Cyberté sont Cisco, ENST Bretagne, France Télécom, IRISA et LSIIT

du réseau veuille contrôler le comportement de chaque terminal. Le Démon de Communication permet ce type d'extension, puisqu'il utilise le language XML. Ce format de message permet la réception et le traitement de requêtes par le réseau. A partir du moment où le Démon de Communication peut être contrôlé par le réseau, une entité spécifique pourra alors contrôler un ensemble de nœuds mobiles.

6.3.4 Le gestionnaire de profils

Comme nous l'avons déjà vu, le rôle de cette architecture est d'unifier la prise en compte des différents paramètres de préférences des flux par rapport aux interfaces, les mécanismes de collecte d'informations et la mise en œuvre des opérations de redirections. Au niveau utilisateur, nous avons choisi d'utiliser différents profils pour récolter les différents paramètres qui seront les critères de sélection pour le sous-système. Ces profils sont tous gérés par un Gestionnaire de Profils (voir figure 6.1), qui réunira les différents critères fournis pour les profils individuels afin d'élaborer une pré-sélection d'interfaces. Cette pré-sélection consiste en l'élaboration d'une liste d'interfaces préférées, qui devra être respectée par le sous-système, i.e le Gestionnaire d'Interfaces.

Les profils sont stockés dans une base de données et contiennent l'information clé sur les différents composants qu'ils représentent. Plus généralement, les profils servent à atteindre les buts suivants :

- Automatiser la sélection d'interfaces et l'accès réseau en maintenant l'information nécessaire dans la base de données des profils.
- Assister le Gestionnaire d'Interfaces pour le choix de la meilleure option d'accès, en tenant compte des besoins des applications et des préférences utilisateur.
- Informer les applications adaptatives à propos des capacités courantes des interfaces et du réseau.
- Rester compatible avec des applications non spécifiques, qui n'ont pas été modifiées pour prendre en compte les états du réseau.

Deux types de profils existent dans la base de données des profils. Les profils génériques décrivent quel type d'information est contenu dans un profil et de quelle manière les profils peuvent être stockés. Ce sont des méta-profils, dans le sens où ils représentent des modèles de profils. Quatre profils génériques différents ont été définis, comme représentés dans la figure 6.1 :

- Profil de préférences et de ressources : il spécifie comment les ressources du

système peuvent être utilisées.
- Profil de description des flux : il contient les paramètres de qualité de service associés aux flux.
- Profil d'accès réseau : il définit l'information nécessaire pour permettre de se connecter au réseau. Cette information concerne aussi bien les paramètres de niveau 2 que ceux de niveau 3.
- Profil d'interface réseau : il stocke les spécifications théoriques de chaque interface réseau du système, ainsi que les statistiques collectées au cours du temps sur les interfaces actives.

Ces profils génériques peuvent alors être instanciés par un administrateur, des utilisateurs ou des applications, pour créer un profil spécifique. Ces profils spécifiques vont ensuite être utilisés par le Gestionnaire de Profils pour établir un ordre de préférence sur les interfaces, en respectant l'ensemble des demandes et besoins du système. Cette liste d'interfaces préférées sera alors transmise au noyau du système pour réaliser la répartition des flux sur les interfaces, dans la limite de la faisabilité en fonction de l'environnement du terminal mobile.

6.3.5 L'adaptation des applications

Toute la conception de cette architecture repose sur le concept d'adaptation du terminal mobile à son environnement. Comme on a pu le voir ci-dessus, cette adaptation passe par la mise en place d'un système de collecte d'informations à tous les niveaux du système et par des mécanismes de changement d'interfaces en temps réel. Pour compléter l'ensemble de ces fonctionnalités, l'adaptation des applications vient concrétiser l'ensemble des mécanismes proposés. Effectivement, lorsque la disponibilité des interfaces change, ou que les performances des interfaces se dégradent ou s'améliorent, une adaptation des applications permet à l'utilisateur final de continuer à recevoir son flux de manière correcte, à une dimension ou à une cadence différente. Prenons l'exemple d'une application de réception d'un flux vidéo. Si le nœud mobile dispose dans un premier temps d'une interface Ethernet, il pourra demander à recevoir des images de grande taille et utiliser un codage nécessitant un haut débit. Par contre, si l'interface Ethernet est débranchée et si le terminal mobile redirige toutes ses communications sur une interface 802.11b, les performances réseau vont décroître. En réponse à cette dégradation, la vidéo s'adapte par l'utilisation d'un codec vidéo différent et/ou par la réduction de la taille des images. Ainsi, la vidéo sera de plus faible qualité (par exemple de dimension plus faible) mais toujours fluide. De plus fortes conséquences sont encore envisageables lorsque le terminal mobile passe d'une interface 802.11 ou Ethernet à une interface GPRS ou plus généralement à une interface vers un réseau de troi-

sième génération (cf. section 2.4). L'adaptation des applications devient dans ce cas incontournable.

Ainsi, une adaptation des applications a été avancée dans ce projet. La réalisation de l'adaptation d'une vidéo à la demande a été mise en place (Darwin [51] pour le côté serveur et mp4player [129] pour le client) ainsi que l'adaptation d'un logiciel de vidéo-conférence (Vic [184] et Rat [154]). Ces applications dédiées nous ont notamment permis d'obtenir un résultat visuel de notre architecture et aussi de montrer le gain en terme de faisabilité et de performance apporté par le développement de cette architecture. Alors qu'avec des applications classiques qui ne s'adaptent pas, l'application devient inutilisable lors de changement(s) dans le statut des interfaces, MIMA permet de poursuivre les communications en cours, en s'auto-adaptant en temps réel aux différentes conditions environnementales du terminal. Ces deux applications multimédia illustrent donc parfaitement les objectifs que nous nous étions fixés, à savoir tirer au mieux parti d'interfaces multiples sur un terminal mobile.

6.4 Algorithmes et mécanisme du Gestionnaire d'Interfaces

Comme nous l'avons vu dans la section précédente, le Gestionnaire d'Interfaces est en charge de la prise de décision finale de la répartition des flux sur les interfaces. L'algorithme mis en place pour réaliser cette répartition est relativement complexe, dans le sens où le nombre de paramètres à prendre en compte est important. Bien qu'il ait à disposition une liste d'interfaces préférées, éventuellement différente selon le flux, la décision finale de répartition devra être prise en fonction de l'état courant des flux, des interfaces et des fonctionnalités présentes sur les nœuds correspondants.

6.4.1 Régularité des évaluations

Afin de rendre l'ensemble de notre système réactif aux différents paramètres d'entrée, comme la disponibilité d'une interface ou le changement de capacité d'une interface, le Gestionnaire d'Interfaces est amené à contrôler régulièrement la répartition des flux sur les interfaces. D'une part, un contrôle périodique est nécessaire pour valider la répartition courante et le statut courant des interfaces. D'autre part, un contrôle sur évènement est nécessaire afin de réagir immédiatement à un

changement ou à une modification des critères de choix d'interfaces. Par exemple, lorsqu'une interface en cours d'utilisation devient indisponible (eg. suite à un déplacement d'un nœud mobile hors de la portée de son point d'accès), il faut rediriger au plus vite les flux en cours sur une autre interface dans un délai minimum. Finalement, le Gestionnaire d'Interfaces contrôlera régulièrement la répartition des flux et réagira sur évènements spécifiques. Ces évènements et les opérations engendrées par ceux-ci sont décrits dans la sous-section suivante.

6.4.2 Les événements déclencheurs de redirection

Afin d'être le plus réactif possible, le module d'Extraction d'une part et le Gestionnaire de Profils via le Démon de communication d'autre part émettent des messages au Gestionnaire d'Interfaces lors de changements spécifiques. Ces changements peuvent être la perte d'une interface, un changement dans les préférences utilisateur ou encore un nouveau flux qui démarre. L'ensemble des fonctionnalités associées à ces événements, qui ont été implémentées dans notre maquette, est décrit ci-dessous.

Bien entendu, avant chaque redirection, le Gestionnaire d'Interfaces vérifie l'adresse du correspondant afin de ne pas écraser une association déjà existante sur le correspondant, si jamais d'autres flux étaient échangés avec ce dernier (mobilité par correspondant, comme expliqué dans la section 6.2.4).

Insertion d'une interface

Dans un terminal mobile comme un ordinateur portable ou un assistant personnel, il est très fréquent d'avoir des cartes de type PCMCIA qui peuvent s'insérer et s'enlever physiquement de la machine, alors que le système est en marche. Lorsqu'une nouvelle interface réseau est insérée dans le terminal mobile, le module d'Extraction envoie immédiatement un message au Gestionnaire d'Interfaces pour signaler cet événement. Le Gestionnaire d'Interfaces déclenche alors deux actions. La première est de vérifier si cette interface est préférée par rapport à celle(s) actuellement utilisée(s) pour les différents flux. La seconde est l'émission d'une requête au Gestionnaire de Profils afin d'obtenir le profil d'accès associé à cette interface. Cette requête permettra au sytème de configurer les différents paramètres associés à cette interface.

Une fois que l'interface a établi une connexion réseau (généralement une connexion avec un point d'accès s'il s'agit d'une interface sans fil), le Gestionnaire d'Interfaces

déclenche une découverte de lien IPv6 plus ou moins rapide selon la préférence de l'interface. Si cette nouvelle interface doit être utilisée à la place d'une (ou plusieurs) autre(s), le Gestionnaire d'Interfaces provoque l'émission d'un Router Solicitation (RS) afin de découvrir au plus vite le routeur et préfixe par défaut du lien. Nous faisons également l'hypothèse que le nœud mobile peut faire une détection de duplication d'adresse sur son adresse lien local [176] parallèlement à la découverte du préfixe. Autrement, si l'interface ne doit pas être utilisée dans l'immédiat, le Gestionnaire d'Interfaces peut mettre l'interface en veille pour économiser la consommation électrique du système (voir section 6.5.1) ou laisser l'interface s'autoconfigurer d'elle-même sans essayer de rendre le processus plus rapide. Ceci évite d'envoyer des messages supplémentaires, puisque dans ce cas on pourra attendre la réception d'un Router Advertisement (RA), dont l'émission est périodique.

Par la suite, si l'interface était préférée à une autre, une fois qu'elle a acquis une configuration IPv6 (adresse globale unique, préfixe IPv6, routeur par défaut), le Gestionnaire d'Interfaces effectuera les redirections nécessaires. Le détail des mécanismes de redirection est donné dans la sous-section 6.4.3.

Retrait d'une interface

De manière symétrique, une interface réseau peut être physiquement retirée de la machine alors que le système est allumé, et ceci même si l'interface est utilisée. Le module d'Extraction permet encore une fois de prendre en compte dynamiquement ce changement en envoyant un message au Gestionnaire d'Interfaces. Ce dernier vérifie si l'interface en question est utilisée par des flux. Si ce n'est pas le cas, le Gestionnaire d'Interfaces supprime simplement la structure de données associée à cette interface et retransmet l'information dans les couches supérieures afin de maintenir le système à jour.

Par contre, si l'interface retirée est en cours d'utilisation, le Gestionnaire d'Interfaces doit déterminer comment rediriger les flux sur d'autres interfaces. Si d'autres interfaces ne sont pas disponibles, les flux seront interrompus. Si d'autres interfaces sont disponibles, le Gestionnaire d'Interfaces identifie le flux le plus prioritaire et essaie de rediriger ce dernier sur l'interface préférée suivante de la liste de préférence.

Début d'un nouveau flux

Un nouveau flux démarre soit lorsque le nœud mobile initialise une nouvelle communication, soit lorsque le nœud mobile reçoit un nouveau flux transféré par son agent mère sur son adresse temporaire principale. Dans les deux cas de figure, le nœud mobile pourra éventuellement enregistrer une adresse temporaire différente de l'adresse temporaire principale auprès du correspondant en question en vue d'utiliser une interface différente avec ce même correspondant.

Lorsqu'un nouveau flux démarre, le Gestionnaire d'Interfaces doit choisir l'interface réseau de sortie la plus adéquate. Il consultera alors la liste des interfaces préférées pour ce flux et identifiera une interface cible (la première interface de la liste qui est disponible dans le système). Cependant, il est possible que l'interface sélectionnée ne puisse pas être utilisée pour ce flux : si ce flux est établi avec le même correspondant qu'un autre flux du nœud mobile et que cet autre flux utilise une interface différente que celle sélectionnée, il faudra choisir une seule et même interface pour ces deux flux. La priorité sur les flux indiquera alors quelle préférence respecter. Il se pourra alors que le flux précédant soit déplacé sur une autre interface.

De plus, le démarrage d'un nouveau flux peut changer les performances des interfaces, ce qui peut également amener un changement radical de la répartition des flux sur les interfaces. Par exemple, si le nouveau flux vient s'ajouter sur une interface déjà fortement utilisée, l'interface peut devenir saturée, c'est-à-dire qu'elle ne pourra pas supporter l'ensemble des flux. Afin de résoudre ce problème, le Gestionnaire d'Interfaces vérifiera également la charge des interfaces lors du démarrage d'un nouveau flux. Si jamais l'ajout de ce flux sur une interface vient saturer cette dernière, le flux le moins prioritaire sera redirigé sur une autre interface. Le Gestionnaire d'Interfaces recommence cette opération tant que l'interface reste saturée.

Handover horizontal

La caractéristique forte d'une interface sans fil est d'autoriser le mouvement en cours de communication. Or comme nous l'avons déjà dit, un mouvement physique peut se traduire par un handover entre deux points d'accès. Comme décrit dans le chapitre 2, le processus de handover peut être plus ou moins long, variant de quelques centaines de millisecondes pour 802.11 à une trentaine de secondes pour Bluetooth. Si jamais l'interface sur laquelle un handover se produit est utilisée par des flux et que d'autres interfaces sont disponibles pour le système, on peut alors

imaginer rediriger temporairement ces flux sur d'autres interfaces.

Lors du début de handover horizontal sur une interface, le module d'Extraction tient le Gestionnaire d'Interfaces au courant du processus en cours. Si jamais l'interface en question est utilisée par des flux, le Gestionnaire d'Intefaces pourra rediriger ce flux sur une autre interface. La décision d'effectuer la redirection temporaire dépendera du type de l'interface et de l'importance des flux. Si jamais la redirection est souhaitée, le Gestionnaire d'Interfaces sélectionnera une interface cible et effectuera la redirection comme indiqué dans la sous-section suivante. Ensuite, en fin de handover de niveau 2 (information transmise par le module d'Extraction), le Gestionnaire d'Interfaces déclenche une détection de lien rapide. Une fois que l'interface offre à nouveau une connectivité IPv6, le Gestionnaire d'Interfaces pourra rétablir la même répartition des flux que précédemment. Les modules du niveau utilisateur ne sont pas tenus informés de ces opérations qui sont décidées de manière autonome et qui doivent rester extrêmement rapides.

Fin de handover de niveau 2

Plus généralement, lorsqu'un handover horizontal de niveau 2 se termine, le module d'Extraction qui détecte cet événement transmet l'information au Gestionnaire d'Interfaces. Si l'interface en question est en cours d'utilisation, il est important que le handover de niveau 3 se termine dans les plus brefs délais, puisque pendant un processus de handover de niveau 3, le nœud mobile ne peut ni envoyer ni recevoir des paquets de données sur cette interface. Afin de terminer le handover de niveau 3 au plus vite, le Gestionnaire d'Interfaces déclenchera alors une détection de lien en provocant l'émission d'un RS. La réception du RA en réponse indiquera si effectivement le lien IPv6 a changé par le préfixe contenu dans le message. Si jamais le préfixe a changé, une nouvelle adresse temporaire globale est créée par le nœud mobile et mise à jour auprès des destinataires concernés (présents dans la liste des Binding Update envoyés). Comme nous le verrons dans la section consacrée à l'évaluation (6.6), l'utilisation d'un tel déclencheur de niveau 2 permet d'atteindre de bonnes performances.

Changement de performances

Le dernier déclencheur que nous avons mis en place est le changement de performances observé sur une interface. En plus du coût financier d'utilisation d'une interface, les performances d'une interface comme le débit offert, seront souvent un paramètre primordial pour établir la liste des interfaces préférées sur le système.

Or, certaines interfaces proposent plusieurs débits bruts, comme 802.11a [3] qui permet des transmissions de 1 à 54 Mb/s. Cependant, ce n'est généralement pas le nœud mobile qui choisit le débit brut, car ce dernier est sélectionné en fonction de la distance entre le nœud mobile et son point d'accès. C'est donc une valeur imposée par l'environnement. Ainsi, alors qu'il semble évident que 802.11a sera généralement préféré à 802.11b lorsque le débit brut pourra être de 54 Mb/s, si le débit est de 6 Mb/s théorique à un instant donné, il serait certainement préférable d'utiliser 802.11b.

A partir de cette constatation, nous avons mis en place un algorithme de surveillance des capacités offertes par les interfaces. Cet algorithme se matérialise pour le moment par la surveillance du débit brut offert par une interface : si celui-ci passe en-dessous d'un certain seuil, le Gestionnaire d'Interfaces, toujours informé par le module d'Extraction, pourra réordonner la liste des interfaces préférées. Ce réordonnancement pourra déclencher des redirections de flux. Les mécanismes de redirection sont présentés dans la sous-section suivante.

6.4.3 Mécanismes de gestion de la mobilité

Actuellement, la plupart des travaux de recherche sur les communications issues d'un terminal multi-interfaces ne proposent que des solutions pour la gestion des flux de données lorsqu'un terminal passe d'un sous-réseau à un autre [190, 192, 193, 198]. Nos travaux découpent les redirections en deux familles, indépendamment du fait qu'il puisse s'agir d'un handover vertical ou horizontal. Ces deux familles se distinguent non seulement par la granularité qu'on désire mettre en œuvre, mais également par l'information de préfixe IPv6 maintenue au cours du temps par le Gestionnaire d'Interfaces. L'un d'entre eux utilise le protocole Mobile IPv6 [84] étendu pour supporter le multi-interfaces [126]. Le deuxième mécanisme, notamment présenté dans [127], est plus optimal encore et complètement indépendant de Mobile IPv6. Par contre, comme nous allons le voir, il ne permet pas une redirection différente pour chaque flux de l'interface.

Afin de simplifier les explications de ces deux mécanismes, nous proposons d'utiliser la terminologie suivante : nous dénommerons l'interface à partir de laquelle les flux sont redirigés l'*interface source* et l'interface sur laquelle seront redirigés les flux, l'*interface cible*.

Redirection locale

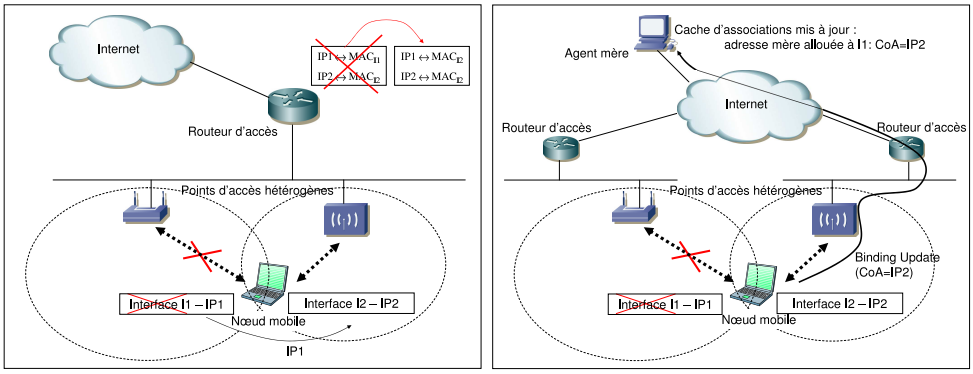
Le mécanisme de redirection locale consiste à rediriger tous les flux entre deux interfaces connectées au même sous-réseau IPv6. En effet, il est tout à fait envisageable qu'un terminal multi-interfaces dispose de deux interfaces dans le même sous-réseau. Prenons l'exemple d'un utilisateur qui dispose d'un équipement pourvu d'une interface filaire (type Ethernet) et d'une interface sans fil (type IEEE 802.11b). Il est concevable que le point d'accès se trouve connecté au réseau filaire interconnectant déjà la prise Ethernet de l'équipement mobile. Aussi, l'utilisateur peut être amené à débrancher son équipement pour changer de bureau tout en restant connecté au travers du réseau sans fil. Si le mobile est pourvu des extensions pour le support de la mobilité (i.e. Mobile IPv6), ceci entraînera automatiquement l'envoi de messages de localisation à son agent mère et potentiellement à l'ensemble de ses correspondants. Pourtant, du point de vue de la localisation, l'équipement mobile sera toujours situé dans le même réseau.

Le mécanisme de redirection locale consiste à associer simplement l'adresse IPv6 de l'interface source à l'interface cible sur laquelle les flux seront redirigés (voir figure 6.2(a)). Cette allocation est réalisée grâce à un message d'invalidation de l'ensemble des caches des nœuds voisins associant l'adresse IPv6 à l'adresse physique (MAC) de la carte. De cette manière l'adresse IPv6 de l'interface source sera associée à l'adresse physique de l'interface cible. Ce mécanisme local évite un grand nombre d'envois de messages de contrôle inutiles. Bien sûr, un certain nombre de traitements plus évolués sont nécessaires sur l'équipement mobile pour tenir compte des différents cas de figure pouvant survenir après la mise en place de ce procédé (par exemple : reconnexion de l'interface source, déconnexion de l'interface cible, handover de l'interface sans fil). L'ensemble des détails de la maintenance et de l'évolution protocolaire suite à une redirection locale est décrit dans [120, 128]. Les figures 6.5 et 6.6 montrent de manière évidente l'apport de la redirection locale et le gain de temps sur l'interruption des communications par rapport à Mobile IPv6. Les résultats sont significatifs puisque le temps d'interruption est divisé par un facteur supérieur à 70 (voir section 6.6).

Redirection globale : MMI

La redirection globale est une utilisation particulière et une amélioration de la gestion des handovers proposée par Mobile IPv6. Actuellement, les spécifications de Mobile IPv6 ne proposent pas la gestion des interfaces multiples de manière explicite. Aussi, les différentes implémentations de Mobile IPv6 proposent une certaine prise en compte d'interfaces multiples, mais la compatibilité est souvent

inexistante et les fonctionnalités incomplètes. C'est pourquoi nous avons proposé un mécanisme standard de redirection entre plusieurs interfaces du même terminal [126, 127].



(a) Mécanisme de redirection locale (b) Mécanisme de redirection globale

FIG. 6.2 – Mécanismes de redirection entre interfaces hétérogènes

La figure 6.2(b) illustre le cas de figure où un nœud mobile dispose de deux interfaces, chacune connectée à un sous-réseau IPv6 distinct. Si un nœud mobile peut déterminer que l'une de ses interfaces (I_2) va bientôt être déconnectée (exemple : signal de plus en plus faible), il peut anticiper cette déconnexion en enregistrant auprès de son agent mère (et de ses correspondants) sa nouvelle localisation dans un autre réseau (au travers de l'interface I1). Par ce biais, il diminuera fortement le temps d'interruption des communications en cours. De même, l'architecture définie permet également de réagir à posteriori d'une perte d'interface, par la remontée de déclencheur de niveau 2, du module d'Extraction au Gestionnaire d'Interfaces, indiquant qu'une interface n'est plus disponible. Notre extension appelée MMI pour *Mobile IPv6 for Multiple Interfaces* permet à un équipement mobile de réaliser les procédures de mise à jour de sa localisation (association entre son adresse IPv6 principale et son adresse IPv6 temporaire) non plus en émettant les messages de contrôle impérativement depuis la même interface mais depuis l'une de ses interfaces courantes. En effet, un nœud mobile pourvu de plusieurs interfaces peut se retrouver dans la nécessité de rediriger une partie, voire l'ensemble des flux d'une interface à une autre, comme nous l'avons montré dans la sous-section précédente.

6.5 Autres fonctionnalités

Notre architecture et les mécanismes associés permettant de gérer de manière optimale plusieurs interfaces sur un terminal mobile ont été présentés dans les sections précédentes. Bien que nous ayons essayé d'inclure un maximum de fonctionnalités, l'étendue des possibilités de comportement du terminal est immense et beaucoup de travail peut encore être fait dans ce sens. D'ailleurs, la conception de l'architecture a volontairement été modulaire afin de permettre de telles extensions. Dans la suite, nous mettons en avant quelques fonctionnalités supplémentaires que nous pensons intéressantes du point de vue de la gestion d'interfaces multiples. Ces fonctionnalités pourront, à terme, être également intégrées à la maquette du prototype déjà développé.

6.5.1 Economie d'énergie

Bien que notre architecture fut conçue pour intégrer des interfaces filaires et des mécanismes de gestion d'interfaces multiples indépendants d'une mobilité physique du terminal, le nœud mobile qui a guidé le développement de cette architecture est un équipement miniaturisé de type assistant personnel, sous-entendu fortement mobile. Effectivement, le scénario type est un utilisateur qui se déplace avec son équipement miniaturisé et qui accède simultanément et consécutivement à différentes technologies de communication. Or, lorsque l'utilisateur se déplace, il utilise généralement son terminal sans pouvoir le brancher sur une prise électrique. La batterie interne de l'équipement permet alors une autonomie qui reste limitée dans le temps. La capacité des batteries est en effet souvent très restreinte, avec moins d'une heure d'utilisation pour le Compaq IPAQ 3870 par exemple. L'utilisation d'une interface réseau génère une consommation non négligeable d'energie électrique. De plus, chaque interface a un comportement de consommation électrique différent [96, 97]. C'est pourquoi, l'intégration de politiques d'utilisation particulières en fonction de la consommation électrique du terminal et des interfaces apparaît nécessaire. Lorsque le terminal est branché sur secteur, on pourra se permettre de laisser allumées toutes les interfaces, même si celles-ci ne sont pas utilisées pour des communications. Laisser une interface constamment allumée et connectée permet d'initialiser une communication ou une redirection plus rapidement, étant donné qu'il ne sera pas nécessaire de passer par une phase de configuration. Par contre, laisser une interface non utilisée par les flux de données allumée, lorsque le terminal fonctionne sur ses batteries propres, n'a pas de sens. Nous pouvons alors imaginer deux modèles d'utilisation du terminal, selon qu'il soit ou non branché sur secteur.

6.5.2 Filtrage des flux

Une autre extension possible concerne le filtrage des flux sur un même correspondant, ce qui correspond au cas 3 de la granularité de la mobilité introduite en section 6.2.4. Cette notion de filtrage est d'autant plus bénéfique si elle peut être faite sur l'agent mère du nœud mobile [122]. Imaginons un terminal mobile muni de deux interfaces, une inteface 802.11b et une interface GPRS. Lorsque le nœud mobile sort d'un WLAN et perd sa connexion 802.11b, il veut certainement rediriger uniquement un sous-ensemble des flux qu'il échange sur son interface 802.11b, étant donné que l'interface GPRS ne supporte pas le même débit (moins de 100 Kb/s contre 11 Mb/s). Alors plutôt que de saturer l'interface GPRS, si le nœud mobile peut placer des règles de filtrage sur son agent mère indiquant quels flux rediriger et quels flux ne pas rediriger permettrait d'éviter de saturer inutilement l'interface GPRS. Cependant, la mise en place de filtrage nécessite des extensions supplémentaires dans les Binding Update [68, 90, 167, 122]. Les extensions que nous avons définies dans [122] permettent à un nœud mobile d'interdir ou d'autoriser la retransmission par l'agent mère d'un flux, identifié par un ou plusieurs champs du quintuplet[2].

6.5.3 Les micro flux

L'utilisation de plusieurs interfaces peut également être une opportunité pour optimiser la répartition des flux au sein d'un même terminal. Nous proposons notamment la séparation des flux d'une même communication sur l'ensemble des interfaces disponibles à un instant donné au sein d'un terminal pourvu de MIMA [120, 126]. Cette répartition peut s'effectuer de deux manières distinctes :

- Gestion par micro flux : le Gestionnaire d'Interfaces détermine quel flux applicatif (entrant et/ou sortant) peut être scindé en micro flux afin de répartir ces derniers sur les différentes interfaces. Le cas d'une application de vidéo à la demande se prête naturellement à ce cas de figure, où le Gestionnaire d'Interfaces peut décider que l'acheminement des paquets de "type voix" passent par une interface et les paquets de "type vidéo" passent par une autre.
- La répartition de charge par pourcentage peut également être utilisée afin de permettre à un certain pourcentage de paquets de transiter par une interface et le reste par une ou plusieurs autres.

[2]Le quintuplet identifiant une communication est consituée des adresses source et destination, des numéros de port source et destination et du numéro de protocole.

6.5.4 Gestion forte de la mobilité

L'utilisation du module d'Extraction permettant d'interagir avec les couches basses d'un terminal permet également d'envisager d'autres nouvelles fonctionnalités comme forcer un déplacement. En effet, de nombreuses technologies de communication sans fil implémentent des comportements types qui ne sont pas forcément les plus intéressants. On peut notamment citer deux cas de figure liés à la norme IEEE 802.11b, l'un est appelé le mode agressif et l'autre peut être qualifié de mode « lâche ». Pour certaines cartes IEEE 802.11b, il est impératif de disposer dans la majorité des cas de la meilleure qualité de signal. Ce comportement a pour conséquence de déclencher des déconnexions/connexions intempestives inutiles. Au contraire, certaines autres cartes privilégient la connexion courante et préfèrent rester connectées à un point d'accès même si la qualité de la communication entraîne de nombreuses pertes de paquets. Il nous paraît toutefois intéressant de disposer de comportements plus intelligents basés sur plusieurs déclencheurs comme par exemple une combinaison de la qualité de la liaison radio, le débit offert, le nombre de clients sur un point d'accès, le nombre de paquets perdus, etc. La combinaison d'un Gestionnaire d'Interfaces et d'un module d'Extraction nous permet d'envisager ce type de comportement.

6.6 Evaluation

Afin de démontrer l'intérêt d'une architecture telle que MIMA, il convient d'en évaluer les performances. Nous avons donc implémenté un ensemble de fonctionnalités de notre approche afin de déterminer les gains obtenus dans différents scénarii (décrits de manière générique dans [125]). Les détails techniques de la plate-forme de tests sont présentés dans la sous-section suivante. Ensuite, à chaque test est consacré une sous-section, à savoir : optimisation des handovers intra-technologie, optimisation des handovers verticaux pour une utilisation alternative de plusieurs interfaces, anticipation de la perte de lien, démarrage d'un nouveau flux, redirection temporaire et redirection suite à une baisse de performance d'une interface. Ces tests représentent l'évaluation des redirections et leur impact sur les flux, suite à la prise en compte des déclecheurs étudiés en section 6.4.2.

6.6.1 Hypothèse de tests

L'architecture présentée dans les sections précédentes a été implémentée dans le système Linux, noyau 2.4.19. Nous avons choisi ce système d'exploitation car un des objectifs du projet RNRT Cyberté [135] était de développer ces différentes fonctionnalités aussi bien sur un ordinateur portable que sur un assistant personnel, de type IPAQ. Comme il est possible d'installer Linux sur un IPAQ, Linux s'est imposé de lui-même. Nous avons choisi le noyau 2.4.19 car il nous fallait utiliser une implémentation de Mobile IPv6, dont la dernière version avait été développée pour ce noyau. MIPL [111] (version 0.9.4) a donc été l'implémentation de Mobile IPv6 sur laquelle nous avons basé le développement du Gestionnaire d'Interfaces. L'implémentation non modifiée a été utilisée pour comparer les performances entre Mobile IPv6 "standard" et les mécanismes proposés.

La partie noyau de MIMA a été implémentée en langage C. Le module d'Extraction a consisté en l'ajout d'une interface de communication dans chaque driver des cartes réseau afin de remonter les événements et valeurs des différents paramètres relevés sur la carte. Les cartes réseau utilisées étaient les suivantes :

- Cartes Ethernet : Realtek 8139 et Intel eepro100 (driver pour les cartes intégrées et les cartes en PCMCIA)
- Carte IEEE 802.11b : Cisco Aironet 350
- Carte IEEE 802.11a : Cisco Aironet 1200
- Carte Bluetooth : 3COM 3CRWB6096-EU, version 2.
- Liaison GPRS : Modification du driver PPP pour la liaison GPRS via une liaison série avec un téléphone portable.

Le démon de communication traite des requêtes au format XML. Dans la partie utilisateur, le Gestionnaire de Profils et l'adaptateur d'applications exécutent des scripts RUBY [158]. Tous ces modules de l'espace utilisateur ont été développés par nos partenaires du projet Cyberté.

Les nœuds mobiles utilisés pour les tests étaient des COMPAQ EVO N150 (Pentium III 800 MHz, 128 Mo de RAM). Trois sous-réseaux, en plus du réseau mère, étaient à notre disposition, comme représenté dans la figure 6.3. Comme on peut le voir sur la figure, les différentes technologies sont réparties dans les différents sous-réseaux. Alors que les points d'accès 802.11 sont des points d'accès du commerce (Aironet 350 et 1200), le point d'accès Bluetooth est un point d'accès logiciel, ie. un ordinateur avec le module BNEP et proposant le service point d'accès (voir section 2.3). Bien que notre plate-forme soit exclusivement en IPv6, nous avons dû utiliser un tunnel IPv4 pour le GPRS. L'IPv6 n'étant pas fourni en natif

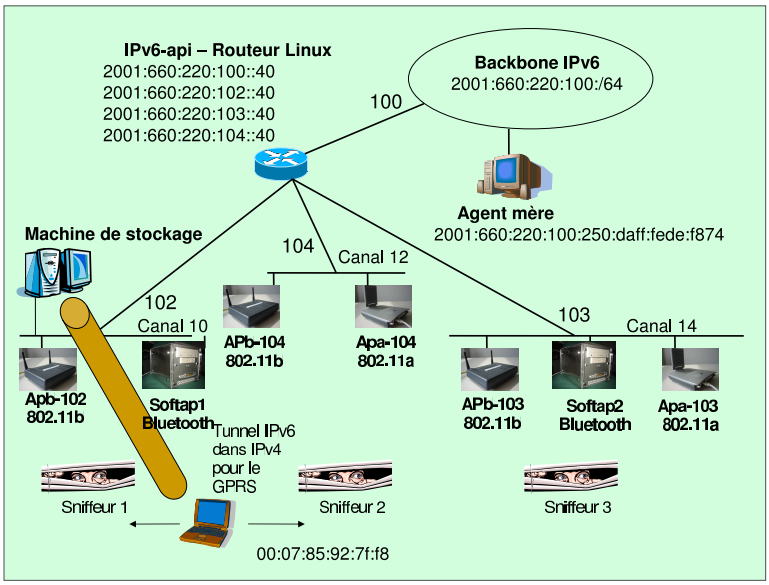


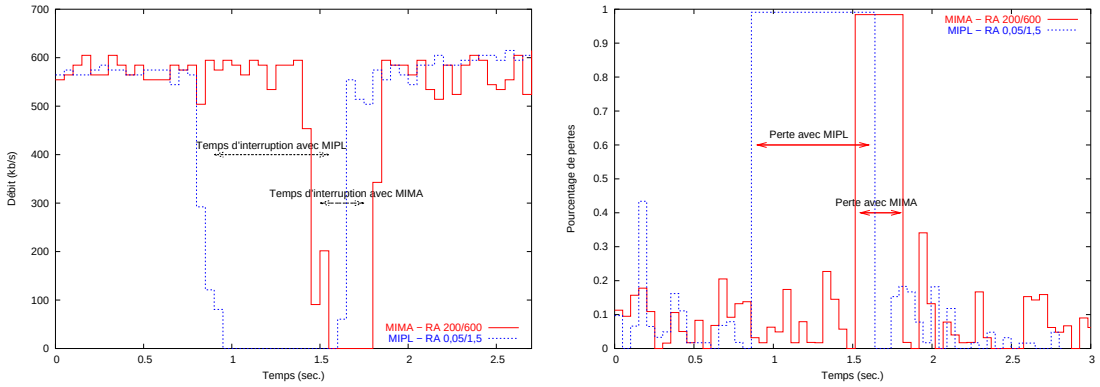
FIG. 6.3 – Plate-forme générique de tests

sur le réseau GPRS, nous avons mis en place un tunnel IPv6 dans IPv4 avec l'une de nos machines munie des deux versions du protocole Internet. Dans chaque sous-réseau nous avons également mis des machines spéciales chargées d'écouter tous le trafic échangé. Ces "sniffeurs" avaient leur carte réseau en mode dit *promiscuous*, mode qui permet uniquement de recevoir des paquets sans interagir sur le réseau. Leur rôle était de relever les temps d'émission des différents messages, grâce au programme TCPDUMP [172].

Afin de constater l'impact de nos mécanismes et de notre architecture sur les flux, nous avons utilisé le générateur de trafic MGEN [110]. Ce générateur permet de définir la taille et la fréquence d'émission des paquets et fonctionne de la manière suivante : un script est lancé sur le nœud émetteur, qui indique les paramètres du flux et le destinataire. Le récepteur écrit dans un fichier le temps et le numéro de séquence de chaque paquet reçu (au niveau du socket), ce qui permet de connaître les déséquencements et pertes qui se sont produites. La plupart des tests ont été réalisés avec un flux de 800 kb/s avec des paquets de 500 octets. Lorsqu'un autre flux sera utilisé, des précisions seront apportées. Dans tous nos tests, le nœud mobile est uniquement récepteur (il ne sera jamais la source du trafic), car nous avons voulu constater les impacts sur la réception d'un flux, de type audio / vidéo multimédia. Tous les graphiques ont une échelle de temps en abscisse débutant à 0, qui ne correspond pas forcément au début du test ; afin de représenter uniquement la partie intéressante du graphique, nous avons fait débuter les graphiques au

moment opportun. Chaque graphique est la moyenne de 100 mesures.

6.6.2 Déplacements intra-technologie



(a) Débit en réception

(b) Pourcentage de paquets perdus

FIG. 6.4 – Impact d'un handover horizontal sur la réception d'un flux

Nous avons tout d'abord cherché à déterminer si les mécanismes proposés pour l'optimisation des déplacements horizontaux ont apporté de réelles avancées. Les figures 6.4(a) et 6.4(b) représentent les temps de latence dûs aux basculements d'un mobile équipé d'une interface IEEE 802.11b. Ces basculements ont été réalisés à l'aide d'un nœud mobile disposant soit d'une implémentation standard de la mobilité IPv6, soit d'une implémentation de MIMA (avec la même version de la mobilité). Comme nous pouvons le constater, l'utilisation des modules d'Extraction et du Gestionnaire d'Interfaces permet de diminuer fortement les temps d'interruption. En effet, le module d'Extraction informe immédiatement le Gestionnaire d'Interfaces après un changement de point d'accès (information de niveau 2). Ce qui permet au Gestionnaire d'Interfaces de déclencher la procédure d'acquisition d'une nouvelle adresse temporaire 8 ms après le rattachement au nouveau point d'accès. Sur un réseau optimisé pour la gestion de la mobilité IPv6, le temps de latence pour un mobile est de 732 ms alors qu'il est divisé par deux (364 ms) avec un mobile utilisant MIMA.

6.6.3 Handover vertical

Nous avons également souhaité déterminer l'impact de notre architecture sur l'utilisation alternative de plusieurs interfaces. Il sera bien entendu probable qu'un utilisateur d'un terminal multi-interfaces préfère une seule et même interface pour l'ensemble des flux qu'il échange à un moment donné. Cette préférence pour une seule interface peut être observée dans plusieurs cas de figure, comme un souci d'économie d'énergie (les autres interfaces peuvent être éteintes), ou le niveau de granularité de la mobilité mis en œuvre. Cependant, si jamais le nœud mobile perd la connexion au réseau sur son interface préférée, par un déplacement hors de la portée de son point d'accès courant par exemple, il est plus que souhaitable de pouvoir rediriger dans les plus brefs délais l'ensemble des flux sur une autre interface du système. De manière symétrique, si l'interface préférée redevient disponible et retrouve une connectivité IPv6, il sera nécessaire de repasser l'ensemble des flux sur cette interface.

Les trois sous-sections suivantes présentent deux illustrations de ce comportement : tout d'abord nous étudierons le mécanisme de redirection, puis son impact sur les flux reçus par le nœud mobile pour les interfaces Ethernet et 802.11b. La dernière section présente le même scénario pour les interfaces 802.11b et GPRS.

Mécanisme de redirection entre Ethernet et 802.11b

L'utilisation alternative des interfaces Ethernet et 802.11b sera très certainement un phénomène courant d'un utilisateur dans son entreprise. Un comportement typique d'un utilisateur sera notamment de basculer ses communications entre ces deux interfaces. En effet, de plus en plus de portables sont équipés en standard de ces deux moyens de communication. On peut donc très facilement imaginer qu'un utilisateur émettra une préférence pour le réseau filaire, mais que lorsqu'il souhaitera se déplacer, il cherchera à maintenir les communications en cours en les basculant sur l'interface sans fil. Inversement, à l'issue de l'un de ses déplacements, il pourra revenir à son bureau et procéder au basculement inverse. Nous nous sommes donc intéressés aux mécanismes et leurs performances en mesurant les temps d'interruption des communications dans ces deux cas de figure.

La figure 6.5 représente les temps et délais des différentes opérations lors du passage de l'interface filaire vers l'interface sans fil. Ce graphique permet notamment de comparer la solution Mobile IPv6 de MIPL, avec nos deux solutions de redirection (voir section 6.4.3). En effet, les deux modes de communication (filaire et sans fil) peuvent être localisés dans le même sous-réseau, ce qui se traduit par

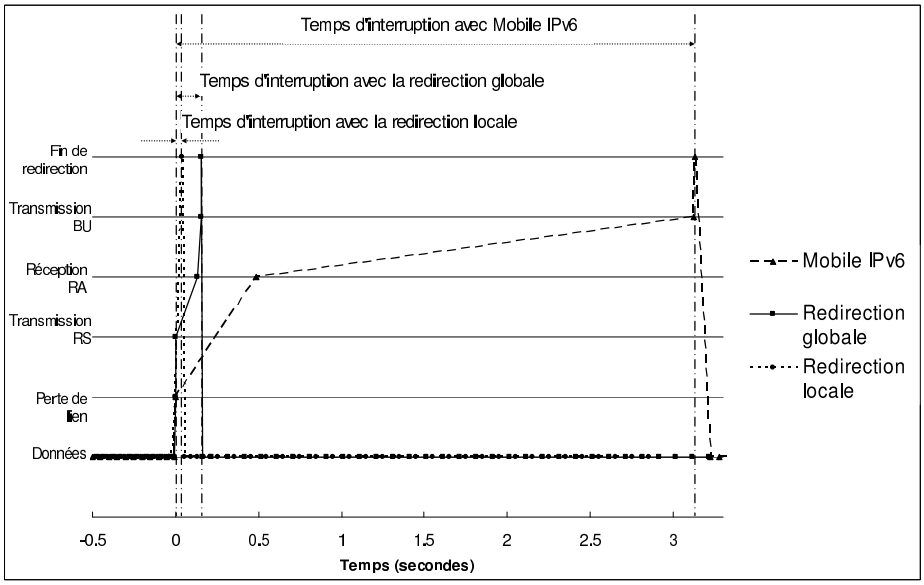


FIG. 6.5 – Différentes étapes d'un handover vertical d'une interface Ethernet vers une interface 802.11b

l'utilisation de la redirection locale. A l'inverse, si les deux technologies d'accès sont situées dans des sous-réseaux différents nous aurons recours à une redirection globale. Ces deux cas de figure ne sont pas présents dans les spécifications de Mobile IPv6, qui les traite au travers d'une redirection globale basique.

Les résultats montrent que dans le cas d'une utilisation de la redirection locale, MIMA permet un handover 76 fois plus rapide. En effet, la redirection locale entraîne une interruption de 41 ms dans les communications alors que l'interruption dure 3129 ms avec Mobile IPv6. Notre seconde solution de redirection globale est entre ces deux temps, mesurée à 491 ms. Au niveau protocolaire, il apparaît clairement sur cette figure que le temps pris pour la découverte du lien (faite sur réception du RA) est nettement supérieure dans Mobile IPv6 : 500ms contre moins de 200ms avec notre méthode de redirection globale, grâce à l'utilisation d'informations de niveau 2 relayées par le module d'Extraction. Une différence encore plus importante concerne le temps d'émission du Binding Update : alors que son émission est quasi-instantanée avec MIMA, elle prend presque 3 s avec Mobile IPv6. Cette différence est due à l'implémentation de la mobilité utilisée qui positionne automatiquement une préférence plus forte sur l'interface filaire par rapport à l'interface sans fil. De ce fait, le Binding Update n'est envoyé qu'à expiration de la configuration IPv6 positionnée sur l'interface filaire. Ce comportement montre bien que Mobile IPv6 ne tient pas du tout compte d'informations de plus bas

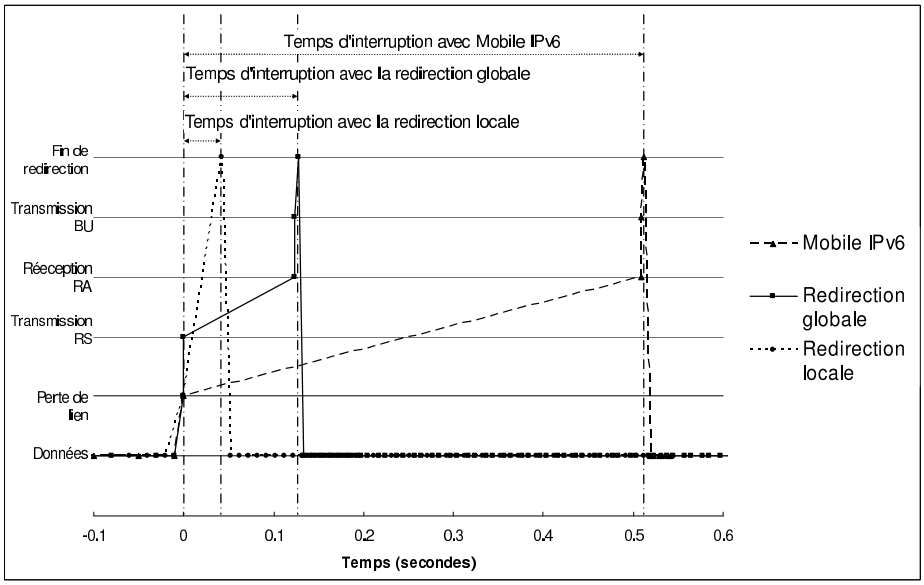
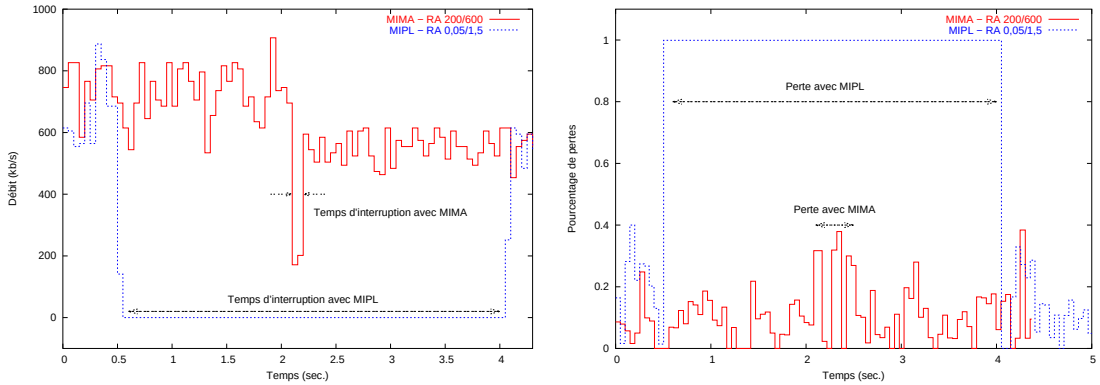


FIG. 6.6 – Différentes étapes d'un handover vertical d'une interface 802.11b vers une interface Ethernet

niveau telle que la déconnexion du câble Ethernet.

Nous avons reproduit les mêmes tests lors d'un basculement inverse : passage de l'interface sans fil à l'interface filaire, qui est représenté dans la figure 6.6. Dans ce scénario, le temps d'interruption dû à l'utilisation de la redirection locale reste identique (41 ms) alors que Mobile IPv6 entraîne une latence de 512 ms à comparer aux 127 ms de la redirection globale de MIMA. Etant donné que l'interface filaire est préférée par rapport à l'interface sans fil dans l'implémentation de Mobile IPv6, dès que l'interface filaire est prête à être utilisée (configuration IPv6 acquise), le Binding Update est envoyé. Le temps de réception du RA, qui informe du lien sur lequel le nœud mobile est positionné, joue alors le plus grand rôle : les 400ms de différence entre la redirection globale de MIMA et Mobile IPv6 sont uniquement dûs à ce temps de détection de lien. Malgré la dissymétrie observée dans ces deux derniers scénarii, l'utilisation de MIMA permet des gains notables dans la diminution du temps de handover vertical. Dans la suite, nous étudierons l'impact de handovers verticaux sur la réception d'un flux.



(a) Débit en réception

(b) Pourcentage de paquets perdus

FIG. 6.7 – Impact d'un handover vertical sur la réception d'un flux entre Ethernet et 802.11b, suite à une perte de lien

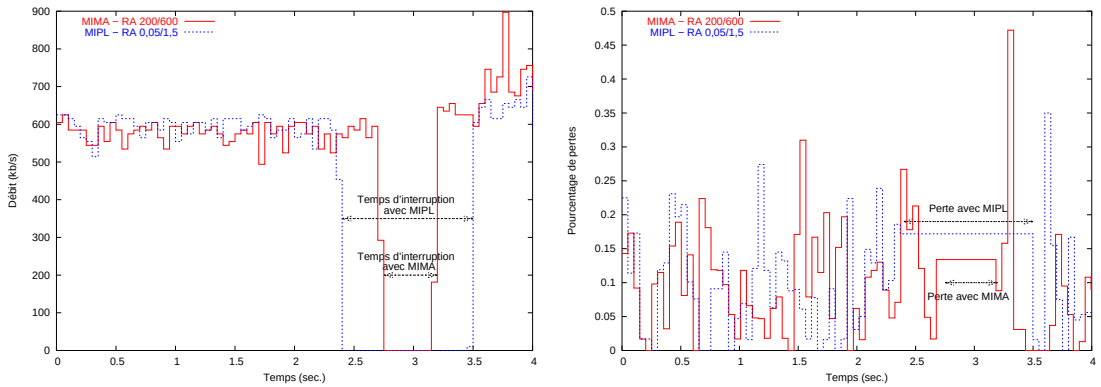
Impact sur un flux de la redirection entre Ethernet et 802.11b

Après avoir vu les différentes opérations nécessaires au basculement entre deux interfaces, nous proposons d'étudier ici quel en est l'impact sur les flux. Deux types de redirection peuvent être identifiés. Une redirection transparente (sans perte de paquet) peut théoriquement être observée lorsque l'interface source (origine de la redirection) reste disponible pendant les opérations de redirection. Ainsi, les paquets sont continuellement reçus sur l'interface source pendant les opérations de redirection. Le deuxième type de redirection est une redirection qu'on peut qualifier de réactive : suite à la perte d'une interface, la redirection est déclenchée. Dans ce cas de figure, le temps de redirection devra être minimiser et entraînera très certainement une perte de paquets de données.

Les figures 6.7(a) et 6.7(b) montrent le débit en réception et la perte de paquets correspondante lors d'une redirection entre une interface Ethernet et 802.11b. Ce cas de figure représente le cas critique, lorsque le câble Ethernet est débranché du terminal. Aussi bien les performances obtenues avec Mobile IPv6 que par les mécanismes de MIMA y sont représentés. La courbe en pointillés (bleus) représentant la réception d'un flux avec l'utilisation de Mobile IPv6, illustre bien les résultats étudiés plus haut : pendant plus de 3 s le nœud mobile ne reçoit pas de paquet de données, délai nécessaire à Mobile IPv6 pour rediriger le flux sur l'interface 802.11b moins préférée. Par contre, MIMA permet de prendre en compte le déclencheur de niveau 2 indiquant le retrait du câble Ethernet ce qui engendre immédiatement la

redirection. Comme le montre la courbe pleine (rouge), une courte perturbation est observée par le nœud mobile au niveau de la réception au temps 2 s, mais peu de paquets ont été perdus du fait de ce handover vertical. Ceci montre bien l'efficacité de la prise en compte d'informations de niveau 2.

Les figures 6.8(a) et 6.8(b) montrent l'impact d'une redirection suite à la réinsertion du câble Ethernet alors que le nœud mobile utilisait l'interface 802.11b. De manière très surprenante, alors qu'on s'attendait à ne pas voir d'implication sur la qualité de réception du flux puisque l'interface 802.11b est toujours disponible pendant la redirection, une baisse conséquente du débit est observée dans les deux situations. Mobile IPv6 laisse apparaître un trou en réception d'un peu plus d'une seconde contre une demi seconde avec MIMA. Ce comportement est reflété de manière moins flagrante sur les courbes du graphique 6.8(b) concernant le taux de perte : pendant la durée de redirection, le taux de perte est constant autour de 20% pour Mobile IPv6 et 15% avec MIMA. Cet impact est peut-être dû à une mise en tampon effectuée par le point d'accès du nœud mobile et qui provoquerait donc un problème de synchronisation au passage du flux sur l'interface filaire.



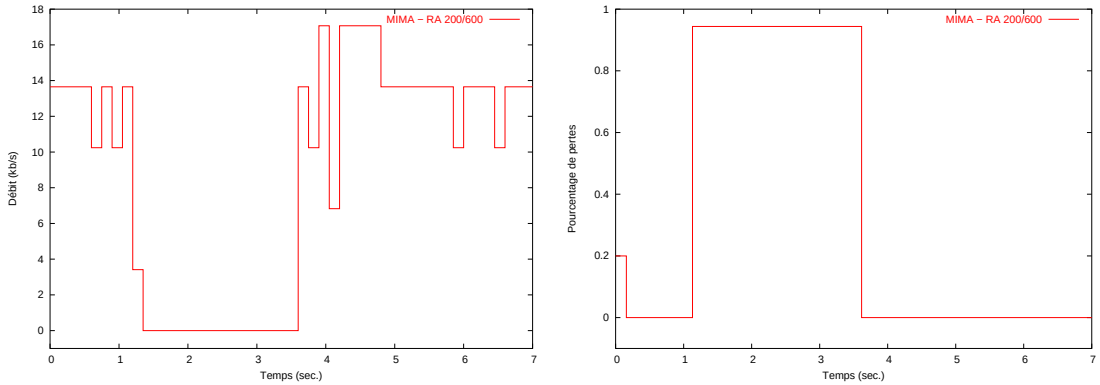
(a) Débit en réception

(b) Pourcentage de paquets perdus

FIG. 6.8 – Impact d'un handover vertical sur la réception d'un flux entre 802.11b et Ethernet, suite à une détection de lien

Impact sur un flux de la redirection entre 802.11b et GPRS

Comme notre architecture a été conçue pour intégrer différentes interfaces réseau, nous désirions valider les comportements de redirection observés ci-dessus sur d'autres technologies. C'est pourquoi nous avons choisi les interfaces 802.11b



(a) Débit en réception

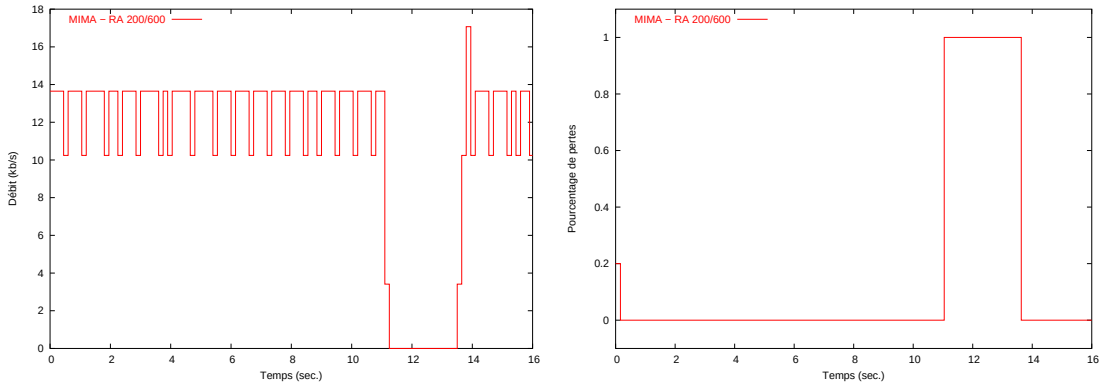
(b) Pourcentage de paquets perdus

FIG. 6.9 – Impact d'un handover vertical sur la réception d'un flux entre 802.11b et GPRS, suite à une perte de lien

et GPRS. Il est effectivement très probable de constater une utilisation alternative d'une interface 802.11b dans des "hot spot" et d'une interface de type 3ème génération comme GPRS. L'utilisation conjointe de ces interfaces est connue sous le nom de réseau de 4ème génération. Nous pensons donc qu'il sera fréquent qu'un utilisateur ayant accès aux technologies simultanément n'utilisera que l'interface 802.11b, pour des raisons de coût et de performance. Cependant, la portée des réseaux 802.11b étant limitée, si l'utilisateur se déplace, il désirera certainement rediriger une partie de ses flux sur l'interface GPRS. Inversement, quand il entrera à nouveau dans un "hot spot" 802.11b, l'utilisateur voudra certainement rediriger ses flux sur l'interface 802.11b.

Les graphiques 6.9(a) et 6.9(b) représentent les implications d'un handover vertical réactif suite à la perte de connexion avec l'interface 802.11b avec MIMA. Une perte pendant 2 s. est observée alors que dans le cas du passage entre Ethernet et 802.11b, très peu de perturbations avaient été observées (voir figure 6.7). Ce temps important est dû à la mauvaise qualité de la liaison GPRS : les émissions et réceptions de paquets dans GPRS se font par vagues de plusieurs paquets (*burst* en anglais) : un ensemble de paquets devant circuler sont mis en tampon et envoyés en raffale à un moment donné.

Les figures 6.10(a) et 6.10(b) représentent les implications d'un handover vertical lorqu'un nœud mobile redirige son flux sur une interface 802.11b qui devient disponible. Encore une fois une perturbation d'environ 2 s peut être observée. Pour le moment nous n'expliquons pas ce délai exagérément important, étant donné que



(a) Débit en réception

(b) Pourcentage de paquets perdus

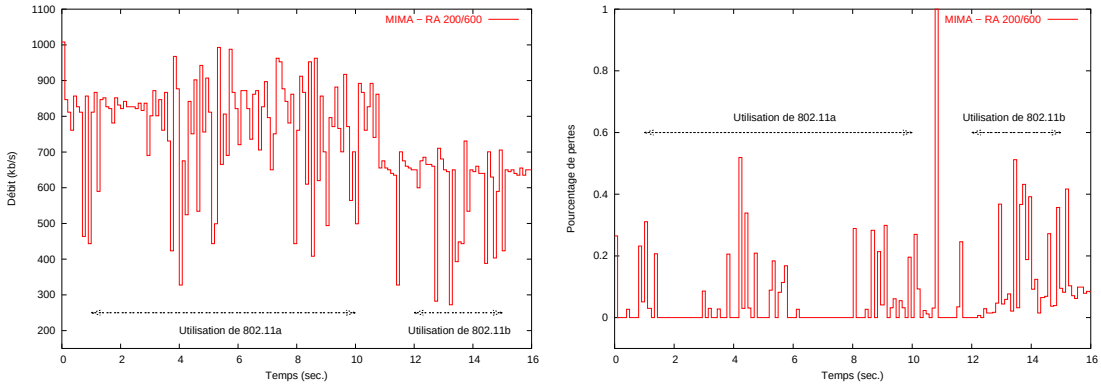
FIG. 6.10 – Impact d'un handover vertical sur la réception d'un flux entre GPRS et 802.11b, suite à une détection de lien

le nœud mobile devrait continuer à recevoir le flux par son interface GPRS. Cette observation est d'autant plus intrigante, car dans tous les tests que nous avons pu faire entre GPRS et d'autres technologies, le passage de GPRS à une autre technologie est toujours ralenti.

6.6.4 Anticipation de perte de lien

Les graphiques de la figure 6.11 représentent une redirection de flux entre une interface 802.11a et une interface 802.11b sur anticipation de perte de lien. Grâce à la surveillance de l'intensité de signal de la liaison entre le nœud mobile et le point d'accès, le nœud mobile anticipe la perte de connexion et redirige son flux sur une autre interface disponible, ie. 802.11b. Comme on peut le voir sur la variation du débit (figure 6.11(a)) et la variation du nombre de paquets perdus (figure 6.11(b)), l'anticipation de la perte de connexion permet une redirection transparente du flux entre les deux interfaces. Aucune baisse de performance n'est observée sur ce test.

Cependant, dans les réseaux 802.11, bien qu'il soit possible d'anticiper une perte de connexion, il n'est pas possible de distinguer la différence entre un handover imminent et une perte définitive de connexion. Le même déclencheur de niveau 2 est capté par le module d'Extraction, à savoir une baisse conséquente de l'intensité de signal.

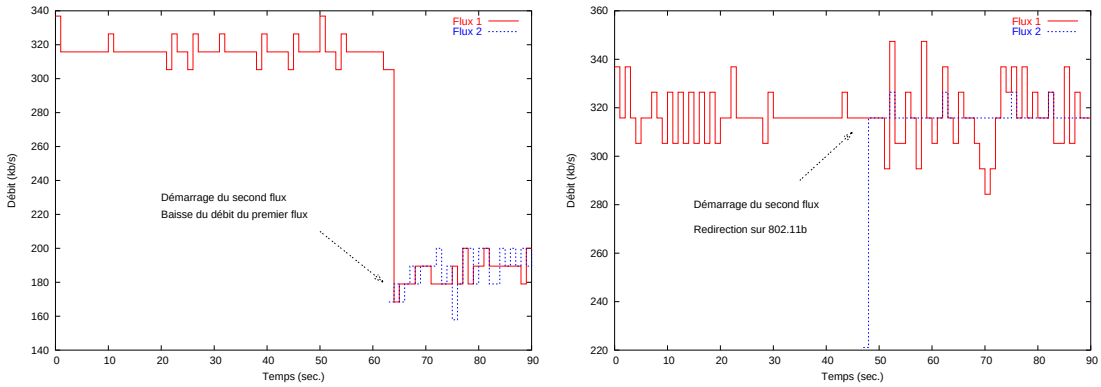


(a) Débit en réception

(b) Pourcentage de paquets perdus

FIG. 6.11 – Impact d'un handover vertical par anticipation sur la réception d'un flux entre 802.11a et 802.11b

6.6.5 Démarrage d'un nouveau flux



(a) Débit en réception sans redirection

(b) Débit en réception avec redirection

FIG. 6.12 – Impact du démarrage d'un second flux sur une interface Bluetooth

L'initialisation d'un nouveau flux sur une interface déjà en cours d'utilisation peut avoir un impact néfaste sur les performances des autres flux échangés sur cette interface. Les graphiques de la figure 6.12 représentent le début de réception d'un deuxième flux sur une interface Bluetooth alors qu'un flux existant était déjà en cours de réception sur cette même interface. Le graphique de la figure 6.12(a)

montre clairement la dégradation en réception du premier flux lorsque le deuxième flux démarre : le débit de réception du premier flux chute brusquement en passant de 320 kb/s à 180 kb/s. Les deux flux sont alors reçus à cadence égale sur le nœud mobile.

MIMA apporte la possibilé de détecter ce phénomène et de proposer une solution, de manière automatique. Lors de la réception du deuxième flux, le Gestionnaire d'Interfaces détectera la baisse de performance sur l'interface et déclenchera une redirection sur 802.11b. Comme on peut le voir sur le graphique 6.12(b), la redirection est fluide et on observe une régularité dans le débit de réception. Ce graphique montre pleinement le gain obtenu grâce à l'utilisation simultanée d'interfaces multiples.

6.6.6 Redirection temporaire pendant un handover

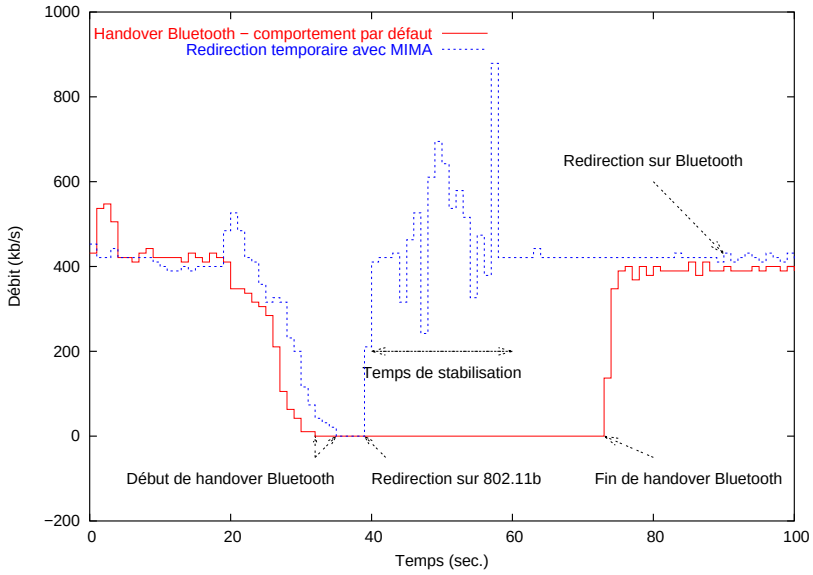


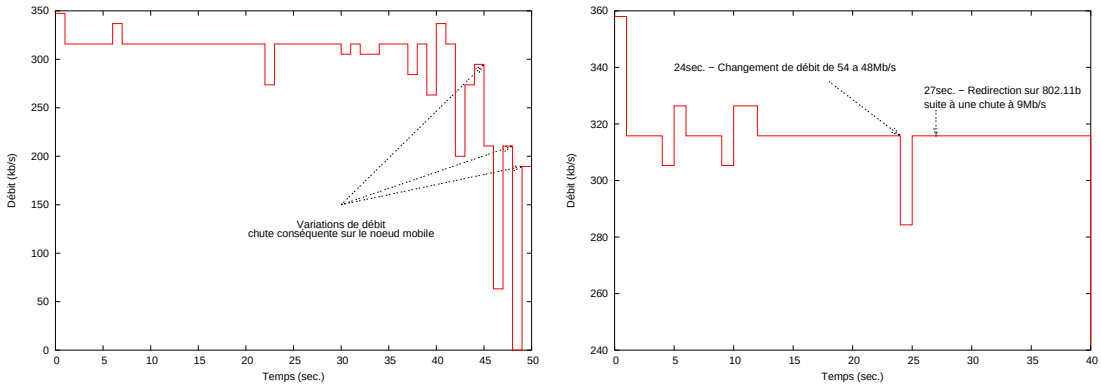
FIG. 6.13 – Effet d'un handover Bluetooth

Comme nous l'avons vu dans la section 2.3, un handover entre deux points d'accès Bluetooth dure une trentaine de secondes. Durant ce laps de temps, le nœud mobile ne pourra pas recevoir de paquet de données sur son interface Bluetooth, ce qui pourra potentiellement dégrader les applications du terminal. Or, la mobilité de l'équipement sera certainement une caractéristique forte du terminal, ce qui implique que des handovers Bluetooth pourront être relativement fréquents. Comme

l'objectif principal de notre architecture est l'utilisation simultanée de plusieurs interfaces, il est impératif de pouvoir gérer les handovers sur cette technologie.

Afin de réduire la perte de paquets sur une interface Bluetooth en cours de handover, nous proposons de rediriger temporairement l'ensemble des flux sur une autre interface. La figure 6.13 représente le débit de réception d'un flux sur le terminal mobile alors qu'il procède à un handover Bluetooth. Deux cas de figure sont représentés dans ce graphique : d'une part le comportement par défaut, où le handover Bluetooth provoque la perte de tous les paquets pendant une trentaine de secondes. D'autre part, nous pouvons observer qu'une redirection temporaire sur une interface 802.11b permet de continuer à recevoir les paquets de données pendant la majeure partie du temps du handover Bluetooth. Par contre, environ deux secondes de délai sont nécessaires afin d'être assuré qu'il s'agit bien d'un handover. La volatilité des indicateurs de qualité de liaison dans la technologie Bluetooth est telle que ce délai de deux secondes a dû être mis en place pour éviter des redirections intempestives, alors qu'aucun handover n'a lieu. Plus tard sur le graphique (autour de 90 s), on voit que le flux est redirigé sur l'interface Bluetooth, une fois qu'on est bien assuré que le handover est terminé. Cette redirection est entièrement transparente, puisque les deux interfaces sont déjà configurées au moment de la redirection. Pendant le temps nécessaire à la redirection effective, le flux est toujours perçu sur l'interface 802.11b.

6.6.7 Baisse de performance sur une interface



(a) Débit en réception sans redirection

(b) Débit en réception avec redirection

FIG. 6.14 – Impact du changement de débit brut sur un point d'accès 802.11a

Comme nous l'avons déjà mentionné, MIMA permet une surveillance permanente de l'état et plus précisément des capacités courantes des interfaces en cours d'utilisation. Effectivement, surtout dans le cas des interfaces sans fil, les capacités des interfaces sont relativement volatiles et des différences importantes en terme de performance peuvent être observées au cours du temps. Le graphique 6.14(a) en est une illustration par la représentation de la réception d'un flux sur une interface 802.11a alors que le nœud mobile s'éloigne de son point d'accès. La conséquence de cet éloignement est la baisse du débit brut utilisé entre le point d'accès et le nœud mobile. Or, comme on peut le voir sur la figure 6.14(a), à partir de 40 secondes le débit de réception sur le nœud mobile décroît de manière importante. Cette baisse des performances est due non seulement à des changements consécutifs de débit, mais également au fait qu'après un changement le débit maximum disponible est inférieur.

Or, comme nous l'avons vu dans la section 6.3, le module d'Extraction surveille en permanence aussi bien les capacités théoriques que pratiques des interfaces. Lorsque le débit brut par défaut utilisé sur une interface change, le module d'Extraction remonte cette information au Gestionnaire d'Interfaces. Pour évaluer l'importance de ce mécanisme, nous avons mis à disposition du terminal mobile deux interfaces, une interface 802.11a et une autre 802.11b. Nous avons également mis la politique suivante sur le nœud mobile : l'interface 802.11a est préférée tant que son débit brut reste supérieur à celui offert par l'interface 802.11b. Il apparaît très clairement sur le graphique 6.14(b) qu'une redirection sur l'interface 802.11b est bénéfique pour la qualité de connexion globale du nœud mobile, pour le même déplacement que précédemment : une première chute de débit sur 802.11a provoque une légère perturbation dans la réception du flux, mais n'incite par le Gestionnaire d'Interfaces à déclencher une redirection entre les deux interfaces. Par contre, à la 27ième seconde, le débit brut de 802.11a passe à 6 Mb/s, ce qui provoque une redirection immédiate sur 802.11b. Comme on le voit très bien sur la figure 6.14(b), cette redirection est quasi totalement transparente en terme de paquets perdus.

6.7 Conclusion

Les nouvelles technologies de communication sans fil ont chacune leurs spécificités et se destinent potentiellement à des usages différents. D'autre part, les équipements mobiles (portables, assistants personnels, etc.) sont de plus en plus souvent pré-équipés de plusieurs de ces technologies. Parallèlement, la nouvelle version du protocole IP offre des atouts pour une meilleure intégration de la mobilité. Toutefois, cette intégration a des limites que nous cherchons à supprimer.

Nos propositions ont pour objectif de diminuer ces limites et d'offrir un terminal le plus ubiquitaire possible. Les premières expérimentations réalisées montrent que les gains apportés par l'architecture appelée MIMA sont significatifs et dans certains cas offrent une totale transparence dans le changement d'interface. Il convient toutefois d'envisager des optimisations supplémentaires.

Conclusion

L'introduction de nouveaux terminaux miniaturisés dans le monde informatique, équipés de plusieurs interfaces de communication, a fait apparaître un nouveau défi. Les nouvelles fonctionnalités de ces terminaux offrent de nouveaux modèles d'utilisation pour lesquels il est nécessaire de définir de nouveaux protocoles de gestion. Ces nouveaux équipements, qui représentent la convergence de la téléphonie mobile et de l'Internet Nouvelle Génération [109], permettront une connexion permanente à l'Internet, quel que soit le lieu et à tout instant. La diversité d'interfaces de communication offre aux utilisateurs un choix d'utilisation de l'interface la plus adéquate par rapport à l'application demandée.

Cependant, la gestion de tels équipements mobiles n'est actuellement que très peu prise en compte par les standards. Nous avons cité les méthodes les plus reconnues qui se proposent de gérer la mobilité des nœuds dans l'Internet, aux différents niveaux du modèle TCP/IP. La gestion de la mobilité peut être faite dans la couche réseau, grâce à des protocoles comme Mobile IPv6 ou LIN6. Ces protocoles utilisent une adresse logique servant aux applications, invariable selon la position de l'équipement, et une (ou plusieurs) adresse(s) de localisation. Des protocoles comme I-TCP ou SCTP proposent une gestion dans la couche transport, avec soit une séparation d'une connexion TCP en deux sessions, soit une manipulation de plusieurs adresses pour une même session. Enfin, SIP est une solution du niveau application, qui permet également de rendre transparents les mouvements des nœuds mobiles dans l'Internet. Nous avons choisi de travailler avec Mobile IPv6, car ce protocole définit une gestion totalement générique de la mobilité, que ce soit pour des connexions UDP, TCP ou autre, et pour n'importe quelle application.

Ensuite, nous nous sommes intéressés à la gestion de la mobilité verticale, c'est-à-dire comment prendre en compte la présence d'interfaces multiples au sein d'un même terminal. Deux approches peuvent grossièrement être distinguées. Certaines propositions s'appuient sur le déploiement d'une architecture dans le réseau afin

de favoriser l'utilisation de technologies différentes, alors que d'autres suggèrent des extensions à Mobile IPv6 pour réaliser du filtrage de flux. Le filtrage de flux consiste à répartir chaque flux indépendamment sur les interfaces pour permettre une utilisation simultanée des interfaces réseau.

Efin, nous avons étudié les possibilités d'optimisations de la gestion de la mobilité. Ces optimisations concernent le temps d'enregistrement de la mise à jour de la nouvelle localisation (à travers des modèles hiérarchiques), la rapidité de configuration suite à un déplacement (détection rapide de lien et Fast Mobile IPv6) et enfin la minimisation des pertes de paquets (Bi-casting). L'idée qui ressort de toutes ces méthodes est qu'il existe un réel manque d'une solution complète, intégrant à la fois la gestion des handovers horizontaux, la gestion des handovers verticaux et l'utilisation simultanée d'interfaces multiples, en ayant toujours à l'esprit un souci d'efficacité.

Dans les chapitres 2 et 3, nous avons évalué les performances des déplacements d'un nœud mobile dans les environnements sans fil. De cette étude, il est apparu qu'aucune technologie ne peut s'imposer comme la technologie universelle des communications IP. L'utilisation d'interfaces multiples, aussi bien de manière alternative que simultanée semble donc être impérative pour réunir satisfaction de l'utilisateur final, connexion permanente et optimisation des performances. D'autre part, la décomposition de l'Internet en domaines favorisant la mise en place de solutions hiérarchiques se révèle très bénéfique pour minimiser le temps d'interruption occasioné lors de déplacements entre sous-réseaux.

A partir de notre étude sur les protocoles de handover, aussi bien au niveau 2 dans les réseaux IEEE 802.11b (changement de point d'accès), qu'au niveau 3 (changement de sous-réseau), nous avons proposé une optimisation appelée *Handover Anticipé* (voir chapitre 5). Cette optimisation a pour objectif de réduire le temps de latence dû au handover de niveau 2, et également de permettre la réception des paquets de données à la nouvelle localisation le plus rapidement possible. Nous avons montré, par simulation dans SimulX [3], que le temps de handover était fortement réduit grâce à une anticipation de mouvements, et grâce à l'utilisation de déclencheurs de niveau 2. En effet, en introduisant des informations de niveau 3 dans la couche 2, le nœud mobile peut accélérer le processus de handover de niveau 3 en envoyant des mises à jour avant ses déplacements effectifs.

Maintenant que nous avons pu identifier les points critiques de la gestion de la mobilité, nous nous sommes intéressés à la prise en compte d'interfaces multiples au sein d'un terminal mobile IPv6. L'objectif de cette recherche était non seulement

[3]SimulX est un nouvel outil de simulation présenté dans le chapitre 4

la possibilité d'utilisation à la fois alternative et simultanée des interfaces, mais également la description de différents scenarii de comportement, accompagnée de leur gestion. L'architecture, que nous avons appelée MIMA, a été définie à cet effet (voir chapitre 6). La conception de MIMA a suivi une approche en couches afin de conserver notre volonté de prise en compte global de tous les paramètres disponibles sur le système. L'architecture consiste en plusieurs modules, qui sont généralement des extensions aux couches du modèle TCP/IP. Les principales caractéristiques de cette architecture sont les suivantes :

- La définition d'une abstraction des technologies de communication, pour fournir une information sur les états des interfaces réseau, quelles que soient leurs spécificités.
- Les opérations de répartition et de redirection de flux entre interfaces en temps réel par rapport aux changements de connectivité. Différents niveaux de granularité de la répartition des flux sur les interfaces pourront être mis en place.
- La prise en compte de différents profils définissant les règles d'utilisation, comme les préférences utilisateurs ou les paramètres de connexion.
- L'adaptation des applications, en fonction des conditions et capacités du réseau.

Une implémentation de MIMA nous a révélé des gains importants sur le nœud mobile, tant au niveau des temps d'interruption lors de redirections entre interfaces, que sur la connectivité globale du nœud. La conception de cette architecture étant modulaire, des extensions futures sont possibles, comme des politiques de gestion d'économie d'énergie, la mise en place de nouveaux profils et/ou comportements, ou l'intégration de nouvelles technologies de communication (eg. l'UMTS).

Perspectives

Le développement de MIMA (voir chapitre 6) a notamment amené la définition d'une abstraction des technologies de communication pour les couches supérieures. La standardisation d'une telle abstraction est un point important pour le développement futur des mécanismes de gestion de la mobilité de niveau 3, car la prise en compte des événements de niveau 2 permet une optimisation évidente, comme nous l'avons montré tout au long de ce rapport. En effet, beaucoup de mécanismes de gestion de la mobilité proposent d'utiliser des déclencheurs de niveau 2, mais sans standardisation, toutes ces propositions manqueront de rigueur, de compatibilité et leur extension à des technologies futures n'est pas assurée. C'est pourquoi,

nous avons commencé une telle standardisation [195, 35, 124].

Cette architecture a été développée dans l'optique d'un contrôle par le terminal, dans le sens où les nouvelles fonctionnalités et mécanismes mis en place sont implémentés au niveau du terminal. Une des extensions possibles de ce travail est d'étendre l'architecture à un modèle de contrôle par le réseau, qui offrirait une vision complémentaire et une granularité plus fine de la gestion de la mobilité. L'approche "contrôle par le réseau" peut se révéler fort intéressante, du fait des connaissances que le réseau peut avoir. Par exemple, un opérateur pourrait répartir des terminaux mobiles sur un ensemble de points d'accès sans fil pour faire de la répartition de charge. Le contrôle par le réseau permettrait également de placer des règles de préférences sur les terminaux selon des critères plus globaux que ceux uniquement détenus par le terminal, comme la carte des différents points d'accès sans fil pour choisir au mieux son point d'attachement selon sa localisation.

En outre, d'autres fonctionnalités et mécanismes sont encore possibles dans la gestion d'interfaces multiples. La prise en compte de modèles d'économie d'énergie semble plus que nécessaire puisque l'équipement cible généralement considéré est un assistant électronique fonctionnant sur batterie. Chaque technologie de communication génère une consommation électrique différente et plusieurs politiques d'utilisation peuvent être mises en place par rapport à ce facteur. De plus, une granularité plus fine de la mobilité est possible et nécessite de plus amples recherches. L'utilisation de plusieurs interfaces pour un même flux peut être efficace dans une certaine mesure, mais les différents délais qui peuvent être introduits par l'utilisation de chemins distincts dans l'Internet requiert une attention particulière.

Par ailleurs, l'outil de simulation s'avère incontournable dans la recherche de protocoles réseau. SimulX a été spécifiquement développé pour la simulation des réseaux sans fil IPv6 et la validation de protocoles de gestion de la mobilité. Ce simulateur est une base sur laquelle d'autres technologies et mécanismes peuvent être développés. La maintenance et la gestion de sa pérennisation est donc un des mes objectifs dans la suite de ma carrière.

En outre, il devient aujourd'hui très intéressant d'étudier la problématique d'interfaces multiples et plus généralement de "*multihoming*" dans le contexte des réseaux mobiles. Un réseau mobile est un réseau capable de changer de point d'attachement à l'Internet, comme une voiture fournissant un accès Internet à ses passagers. J'ai commencé à étudier cette problématique [177, 114], mais d'importantes recherches restent en suspens : un réseau mobile introduit des nouvelles caractéristiques en terme de bande passante, et de délai de communication qui sont autant d'informations à prendre en compte au niveau du choix des interfaces à utiliser. MIMA pourrait être étendue dans ce sens, afin d'intégrer des nouveaux

mécanismes et préférences spécifiques aux réseaux mobiles.

En particulier, lorsqu'un routeur mobile se déplace, il peut se rattacher à un autre réseau mobile. Le réseau ainsi créé est alors appelé réseau mobile agrégé. Plusieurs problèmes peuvent alors se poser :

- Comment supporter la mise à l'échelle du support de réseaux et routeurs mobiles dans l'Internet Nouvelle Génération ?
- Comment construire l'arbre des routeurs mobiles dans un réseau mobile agrégé ?
- Comment un nœud mobile peut choisir son routeur par défaut ?
- Comment les nœuds mobiles à l'intérieur du réseau mobile (agrégé) peuvent communiquer avec les nœuds extérieurs ?
- Comment les nœuds mobiles à l'intérieur du réseau mobile (agrégé) peuvent communiquer avec d'autres nœuds du même réseau ?

Toutes ces questions sont encore sans réponse à ce jour et de plus amples recherches sont à faire dans ce domaine.

Bibliographie

[1] Third Generation Partnership Project (3GPP), `http://www.3gpp.org/`.

[2] *IEEE Std 802.11, 1999 Edition (R2003) (ISO/IEC 8802-11 : 1999) IEEE Standard for Information Technology - Telecommunications and Information Exchange between Systems - Local and Metropolitan Area Network - Specific Requirements - Part 11 : Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, 1999.

[3] *IEEE Std. 802.11a-1999, Local and Metropolitan Area networks - Specific Requirements-part11 : Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications : Higher-Speed Physical Layer in the 5 Ghz Band*, Septembre 1999.

[4] *IEEE Std. 802.11b-1999, Local and Metropolitan Area networks - Specific Requirements- part11 : Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications : Higher-Speed Physical Layer Extension in the 2.4 Ghz Band*, Septembre 1999.

[5] *IEEE Std. 802.11e, Draft Amendment to Standard [for] for Information Technology– Telecommunications and Information Exchange Between Systems– LAN/MAN Specific Requirements– Part 11 Wireless Medium Access Control (MAC) and Physical Layer (PHY) specifications : Medium Access Control (MAC) Quality of Service (QoS) Enhancements.*

[6] *IEEE Std. 802.11a-2003, Local and Metropolitan Area networks - Specific Requirements-part11 : Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 4 : Further Higher Data Rate Extension in the 2.4 Ghz Band*, 2003.

[7] *IEEE Std. 802.11i/D4.0, Draft Amendment to Standard for Telecommunications and Information Exchange Between Systems - LAN/MAN Specific Requirements Part 11 : Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications : Medium Access Control (MAC) Security Enhancements*, Mai 2003.

[8] *IEEE 802.15.1(tm)-2002, IEEE Standard for Information technology– Telecommunications and information exchange between systems– Local and metropolitan area networks– Specific requirements Part 15.1 : Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Wireless Personal Area Networks (WPANs(TM)).*

[9] *IEEE Std. 802.1X-2001, IEEE Standards for Local and Metropolitan Area Networks - Port-Based Network Access Control*, Juin 2001.

[10] IEEE 802.21 Working Group, `http://www.ieee802.org/21/`.

[11] *IEEE Std. 802.3-2002, IEEE Standard for Information technology– Telecommunications and information exchange between systems– Local and metropolitan area networks–Specific requirements– Part 3 : Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications*, 2002.

[12] *IEEE Std. 802.5-1998, IEEE Standard for Information technology– Telecommunications and information exchange between systems– Local and metropolitan area networks–Specific requirements– Part 5 : Token Ring Access Method and Physical Layer Specification*, 1998.

[13] Authentication, Authorization and Accounting (aaa), Internet Engineering Task Force Working Group, `http://ietf.org/html.charters/aaa-charter.html`.

[14] Allied Business Intelligence - WLAN Forecast and Analysis, 2001-2006, `http://www.abiresearch.com`.

[15] Kalle Ahmavaara, Henry Haverinen, and Roman Pichna. Interworking Architecture Between 3GPP and WLAN Systems. *IEEE Communications Magazine*, 41 :74–81, Novembre 2003.

[16] O.B. Akan and I.F. Akyildiz. ATL : Andaptive Transport Layer for Next Generation Wireless Terminals. *IEEE JSAC*, 2004.

[17] Ian Akyildiz, Yucel Altunbasak, Faramarz Fekri, and Raghupathy Sivakumar. AdaptNet : An Adaptive Protocol Suite for the Next-Generation Wireless Internet . *IEEE Communications Magazine*, 42 :128–135, Mars 2004.

[18] I.F. Akylidiz and M.C. Vuran. A-MAC : Adaptuve Medium Access Control for Next Generation Wireless Terminals. *BWN Lab Technical Report, School of ECE, GA Institute of Technology*, Novembre 2003.

[19] K. Almeroth. The Evolution of Multicast : From the MBone to Inter-Domain Multicast to Internet2 Deployment. *IEEE Network*, 14(1) :10–20, Jan-Feb 2000.

[20] F. André, J-M. Bonnin, B. Deniau, K. Guillouard, N. Montavont, T. Noel, and L. Suciu. Optimized Support of Multiple Wireless Interfaces within an

IPv6 End-terminal. In *Smart Objects Conference (SOC'2003)*, Grenoble, France, 15-17 Mai 2003.

[21] M. Annoni, R. Hancock, T. Paila, E. Scarrone, R. Tonjes, L. DellUomo, D. Wisely, and R. Mort. . In *12th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC 2001)*, San Diego, USA, Septembre 2001.

[22] J. Arkko, V. Devarapalli, and F. Dupont. Using ipsec to protect mobile ipv6 signaling between mobile nodes and home agents, work in progress, internet engineering task force draft-ietf-mobileip-mipv6-ha-ipsec-06.txt, Juin 2003.

[23] D. Atkins, W. Stallings, and P. Zimmermann. PGP Message Exchange Formats, Internet Engineering Task Force Request for Comments (RFC) 1991, Août 1996.

[24] I. Aydin, Woojin Seok, and Chien-Chung Shen. Cellular SCTP : a transport-layer approach to internet mobility. In *Proceedings of the 12th International Conference on Computer Communications and Networks (ICCCN)*, pages 285– 290, Dallas, USA, Octobre 2003.

[25] R. Bagrodia, W.W. Chu, L. Kleinrock, and C. Popek. Vision, issues and architecture for Nomadic computing and communications. *IEEE Personal Communications*, 2 :14–27, Décembre 1995.

[26] Mary G. Baker. Changing Communication Environments in Mosquitonet. In *IEEE Workshop on Mobile Computing Systems and Applications*, Décembre 1994.

[27] Mary G. Baker, Xinhua Zhao, Stuart Cheshire, and Jonathan Stone. Supporting Mobility in MosquitoNet. In *Dans les proceedings de USENIX Winter Technical Conference*, Janvier 1996.

[28] A. Bakre. I-TCP : indirect TCP for mobile hosts. In *Proceedings of the 15th International Conference on Distributed Computing Systems (ICDCS'95)*, page 136, Vancouver, Canada, Mai-Juin 1995.

[29] A. Bakre. I-TCP : indirect TCP for mobile hosts. In *Proceedings of the 15th International Conference on Distributed Computing Systems (ICDCS'95)*, page 136, Vancouver, Canada, Mai-Juin 1995.

[30] A. Bakre and B.R. Badrinath. Implementation and performance evaluation of Indirect TCP/IP. *IEEE Transactions on Computers, special issue on Mobile Computing*, 46 :260 – 278, Mars 1997.

[31] Ajay Bakre and B.R. Badrinath. Handoff and system support for indirect TCP/IP. In *In Second USENIX Symposium on Mobile and Location-Independent Computing Proceedings*, Ann Arbor, Michigin, Avril 1995.

[32] Luca Becchetti, Francesco Delli Priscoli, Tiziano Inzerilli, Petri Mahonen, and Luis Munoz. Enhancing IP Service Provision over Heterogeneous Wireless Networks : A Path toward 4G. *IEEE Communications Magazine*, 39, Août 2001.

[33] Farouk Belghoul, Yan Moret, and Christian Bonnet. IP-Based Handover Management over Heterogeneous Wireless Networks. In *28th Annual IEEE Conference on Local Computer Networks (LCN)*, Bonn, Germany, Octobre 2003.

[34] P. Bertin, A. Kadelka, J. Rapp, A. Lappetelainen, B. Wegmann, and H. Li. Concepts for IP-based Radio Interface in the BRAIN Framework. In *11th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC 2000)*, pages 437–444, Londres, UK, Septembre 2000.

[35] P. Bertin, N. Montavont, and T. Noël. Parameters for Link Hints, Work in Progress, Internet Engineering Task Force draft-bertin-hints-params-00.txt, Août 2003.

[36] Bluetooth specification, `http://www.bleutooth.org/spec`.

[37] R. Bush, A. Durand, B. Fink, O. Gudmundsson, and T. Hain. Representing Internet Protocol version 6 (IPv6) Addresses in the Domain Name System (DNS), Internet Engineering Task Force Request For Comments (RFC) 3363, Août 2002.

[38] Ramon Caceres and Venkata Padmanabhan. Fast and Scalable Wireless Handoffs in support of mobile Internet audio. *Journal de l'ACM, Mobile Networks and Applications (MONET)*, pages 351–363, décembre 1998.

[39] F. Cali, M. Conti, and E. Gregori. IEEE 802.11 Wireless LAN : Capacity Analysis and Protocol Enhancement. In *Proceedings of IEEE The Conference on Computer Communications (INFOCOM'98)*, Mars 1998.

[40] A. Campbell, J. Gomez, S. Kim, A. Valko, C-Y. Wan, and Z. Turanyi. Design, Implementation, and Evaluation of Cellular IP. *IEEE Personal Communications*, 7 :42–49, Août 2000.

[41] A. Campbell, J. Gomez, C-Y. Wan, S. Kim, Z. Turanyi, and A. Valko. Cellular IP, Internet Engineering Task Force draft-ietf-mobileip-cellularip-00.txt, Décembre 1999.

[42] Claude Castelluccia. Toward a hierarchical mobile IPv6. In *In Eighth IFIP Conference on High Performance Networking (HPN'98)*, Vienne, Autriche, Septembre 1998.

[43] Claude Castellucia. HMIPv6 : a hierarchical mobile IPv6 proposal. *ACM Mobile Computing and Communication Review (MC2R)*, Avril 2000.

[44] J.P. Castro. *The UMTS Network and Radio Access Technology : Air Interface Techniques for Future Mobile Systems.* John Wiley & Sons, Inc., 2001.

[45] Stuart Cheshire and Mary Baker. Experiences with a wireless network in MosquitoNet. In *IEEE Micro*, Février 1996.

[46] JinHyeock Choi and DongYun Shin. Fast router discovery with AP notification, Work in Progress, Internet Engineering Task Force draft-jinchoi-l2trigger-fastrd-01.txt, 2002.

[47] G. Cizault. *IPv6 - 3ème édition.* Editions O'Reilly, 2002.

[48] Xavier Costa, Ralf Schmitz, Hannes Hartenstein, and Marco Liebsch. A MIPv6, FMIPv6 and HMIPv6 Handover Latency Study : Analytical Approach. In *11th IST Mobile and Wireless Telecommunications Summit 2002*, Juin 2002.

[49] Perkins D. The Point-to-Point Protocol for the Transmission of Multi-Protocol of Datagrams Over Point-to-Point Links, Internet Engineering Task Force Request For Comments (RFC) 1171, Juillet 1990.

[50] Greg Daley, Brett Pentland, and Richard Nelson. Movement Detection Optimizations in Mobile IPv6. In *The 11th IEEE International Conference on Networks (ICON)*, Sydney, Australie, Septembre 2003.

[51] Darwin Streaming Server, `http://developer.apple.com/darwin/projects/streaming`.

[52] Subir Das, Anthony McAuley, Ashutosh Dutta, Archan Misra, Kaushik Chakraborty, and Sajal K. Das. IDMP : An Intradomain Mobility Management Protocol for Next-Generation Wireless Networks. *IEEE Wireless Communications, special issue on Mobile and Wireless Internet*, 9 :38–45, Juin 2002.

[53] Université de Californie du Sud ISI. Virtual InterNetwork Testbed (VINT) : methods ans system, 1996.

[54] S. Deering and R. Hinden. Internet Protocol, Version 6 (IPv6), Internet Engineering Task Force Request For Comments (RFC) 1883, Décembre 1995.

[55] Vijay Devarapalli, Ryuji Wakikawa, Alexandru Petrescu, and Pascal Thubert. Network mobility (nemo) basic support protocol, work in progress, internet engineering task force draft-ietf-nemo-basic-support-03.txt, Juin 2004.

[56] Detecting Network Attachment (DNA), Internet Engineering Task Force Working Group, `http://ietf.org/html.charters/dna-charter.html`.

[57] S. Donovan and J. Rosenberg. Session Timers in the Session Initiation Protocol (SIP), Work in Progress, Internet Engineering Task Force draft-ietf-sip-session-timer-14, Février 2004.

[58] R. Droms. Dynamic Host Configuration Protocol, Internet Engineering Task Force Request for Comments (RFC) 2131, Mars 1997.

[59] Adrian Duda and Cormac J. Sreenan. Challenges for Quality of Service in Next Generation Mobile Networks. In *Information Technology and Telecommunications (IT&T03)*, Ireland, Octobre 2003.

[60] W.W. Erdman. Wireless Communications : A decade of progress. *IEEE Communications Magazine*, 31 :48–51, Décembre 1993.

[61] T. Ernst, N. Montavont, R. Wakikawa, E. Paik, C. Ng, K. Kuladinithi, and T. Noël. Goals and Benefits of Multihoming, Work in Progress, Internet Engineering Task Force draft-multihoming-generic-goals-and-benefits-01.txt, Février 2004.

[62] D. Estrin, M. Handley, J. Heidemann, S. McCanne, Y. Xu, and H. Yu. Network Visualization with the VINT Network Animator NAM, Mars 1999.

[63] R. Stewart et al. Stream Control Transmission Protocol (SCTP) Dynamic Address Reconfiguration Internet Engineering Task Force, draft-ietf-tsvwg-addip-sctp-08.txt, Novembre 2003.

[64] Trevor Blackwell et al. Secure Short-Cut Routing for Mobile IP. In *Summer USENIX*, Juin 1994.

[65] N. A. Fikouras and C. Gorg. Performance Comparison of Hinted and Advertisement Based Movement Detection Methods for Mobile IP Hand-offs. In *Proceedings of the European Wireless 2000*, Dresden, Germany, Septembre 2000.

[66] N. A. Fikouras, A. J. Konsgen, and C. Gorg. Accelerating Mobile IP Hand-offs through Link-layer Information. In *Proceedings of the International Multiconference on Measurement, Modelling, and Evaluation of Computer-Communication Systems (MMB)*, Aachen, Germany, September 2001.

[67] N. A. Fikouras, A. Udugama, C. Gorg, W. Zirwas, and J. M. Eichinger. Experimental Evaluation of Load Balancing for Mobile Internet Real-Time Communications. In *Proceedings of the Sixth International Symposium on Wireless Personal Multimedia Communications (WPMC)*, Yokosuka, Kanagawa, Japan, Octobre 2003.

[68] N. A. Fikouras, A. Udugama, K. Kuladinithi, C. Gorg, and W. Zirwas. Filters for Mobile IP Bindings (NOMAD), Work in Progress, Internet Engineering Task Force draft-nomad-mobileip-filters-05.txt, Ocrobre 2003.

[69] S. Fu and M. Atiquzzaman. SCTP : State of the Art in Research, Products, and Technical Challenges. *IEEE Communications Magazine*, 42 :64–76, Avril 2004.

[70] ITU-T Recommendation G.711, Pulse Code Modulation (PCM) of Voice Frequencies, 1972.

[71] The Gartner Group - Wireless LAN Equipment Market, `http://www.gartner.com`.

[72] R. Gilligan, S. Thomson, J. Bound, and W. Stevens. Basic Socket Interface Extensions for IPv6, Internet Engineering Task Force Request For Comments (RFC) 2133, Avril 1997.

[73] P. Grossetete, J. Bound, and T. Hain. Rapport de la north american ipv6 task force, September 2003.

[74] GSM Standards, `http://www.mobilein.com/gsm_standards.htm`.

[75] GSM World, `http://www.gsmworld.com/index1.html`.

[76] Paul S. Henry and Hui Luo. WiFi : What's Next ? *IEEE Communications Magazine*, pages 66–72, Décembre 2002.

[77] Martin Heusse, Franck Rousseau, Gilles Berger-Sabbatel, and Andrzej Duda. Performance Anomaly of 802.11b. In *Proceedings of IEEE The Conference on Computer Communications (INFOCOM'03)*, Avril 2003.

[78] R. Hinden. RFC-2450 - Proposed TLA and NLA Assignment Rules, December 1998.

[79] Host Identity Protocol (HIP), Internet Engineering Task Force Working Group, `http://ietf.org/html.charters/hip-charter.html`.

[80] H.-Y. Hsieh and R. Sivakumar. A Receiver-centric Transmission Protocol for Mobile Hosts with Heterogeneous Wireless Interfaces. In *Proc. of ACM MOBICOM*, San Diego, CA, Septembre 2003.

[81] Masugi Inoue, Khaled Mahmud, Mikio Hasegawa, Homare Murakami, and Gang Wu. MIRAI : A Solution to seamless Integration of Heterogeneous Wireless Networks. In *International Conference on Telecommunications (ICT02)*, Pekin, Chine, Juin 2002.

[82] M. Ishiyama, M. Kunishi, and F. Teraoka. An Analysis of Mobility Handling in LIN6. In *In Proc. of The Fourth International Symposium on Wireless Personal Multimedia Communications*, Aalborg, Denmark, Septembre 2001.

[83] M. Ishiyama, M. Kunishi, K. Uehara, H. Esaki, and F. Teraoka. LINA : A New Approach to Mobility Support in Wide Area Networks. *IEICE Transactions on Communication*, E84-B :2076–2086, Août 2001.

[84] D. Jonhson, C. Perkins, and J. Arkko. Mobility support in ipv6, internet engineering task force request for comments (rfc) 3775, Juin 2004.

[85] Heikki Kaaranen, Siamäk Naghian, Lauri Laitinen, Ari Ahtiainen, and Valtteri Niemi. *UMTS Networks : Architecture, Mobility and Services*. Hardcover, 2001.

[86] J. Kempf and et. Al. Supporting optimized handover for ip mobility - requirements for underlying systems, work in progress, Internet Engineering Task Force, draft-manyfolks-l2-mobilereq-02.txt, June 2002.

[87] James Kempf, Mohamed M. Khalil, and Brett Pentland. IPv6 Fast Router Advertisement, Work in Progress, Internet Engineering Task Force draft-mkhalil-ipv6-fastra-01.txt, 2002.

[88] Seok J. Koh, Mee Jeong Lee, Maximilian Riegel, Mary Li Ma, and Michael Tuexen. Mobile SCTP for Transport Layer Mobility, Internet Engineering Task Force, draft-sjkoh-sctp-mobility-03.txt, Février 2004.

[89] Rajeev Koodli and Charles E. Perkins. Fast Handovers and Context Transferts in Mobile Networks. *ACM Computer Communication Review*, 31, Octobre 2001.

[90] K. Kuladinithi, N. A. Fikouras, and C. Goerg. Filters for Mobile IPv6 Bindings (NOMADv6), Work in Progress, Internet Engineering Task Force draft-nomadv6-mobileip-filters-02.txt, Mai 2004.

[91] M. Kunishi, M. Ishiyama, K. Uehara, H. Esaki, and F. Teraoka. LIN6 : A New Approach to Mobility Support in IPv6. In *In Proc. of The Third International Symposium on Wireless Personal Multimedia Communications*, Bangkok, Thailand, Novembre 2000.

[92] Ted Taekyoung Kwon, Mario Gerla, Sajal Das, and Subir Das. Mobility Management for VoIP Service : MIP vs SIP. *IEEE Wireless Communications*, 9 :66–75, Octobre 2002.

[93] X. Lagrange, Philippe Godelewski, and Sami Tabbane. *Réseaux GSM : des principes à la norme, Cinquième édition.* Hermes, 2000.

[94] Yong-Kyung Lee and Dongmyun Lee. Broadband Access in Korea : Experience and Future Perspective. *IEEE Communications Magazine, topics in Emerging Technologies*, 41 :30–36, Décembre 2003.

[95] Jun Li, Stephen B. Weinstein, Junbiao Zhang, and Nan Tu. Public Access Mobility LAN : Extending the Wireless Internet Into the LAN Environment. *IEEE Wireless Communications, special issue on Mobile and Wireless Internet*, 9 :22–30, Juin 2002.

[96] Jean Lorchat and Thomas Noel. Aggrégation de trames et économie d'énergie dans les réseaux à infrastructure à la norme ieee 802.11. In *CFIP*, 2003.

[97] Jean Lorchat and Thomas Noel. Power performance comparison of heterogeneous wireless network interfaces. In *IEEE Vehicular Technology Conference Fall*, 2003.

[98] C.S. Loredo and S.W. deGrimaldo. Wireless LANs : Global Trends in the Workplace and Public Domain. *The strategies Group*, 2002.

[99] J. Loughney, M. Nakhjiri, C. Perkins, and R. Koodli. Context Transfer Protocol, Work in Progress, Internet Engineering Task Force draft-ietf-seamoby-ctp-09.txt, Avril 2004.

[100] Petri Mahonen, Janne Riihijarvi, Marina Petrova, and Zach Shelby. Hop-by-Hop Toward Future Mobile Broadband IP . *IEEE Communications Magazine*, 42 :138–146, Mars 2004.

[101] Karim El Malki and Hesham Soliman. Simultaneous bindings for mobile ipv6 fast handovers, work in progress, internet engineering task force draft-elmalki-mobileip-bicasting-v6-05.txt, Octobre 2003.

[102] D. Maltz and P. Bhagwat. Tcp splicing for application layer proxy performance, Mars 1998.

[103] David A. Maltz and Pravin Bhagwat. MSOCKS : An architecture for transport layer mobility. In *Proceedings of IEEE The Conference on Computer Communications (INFOCOM'98)*, pages 1037–1045, 1998.

[104] J. Manner, M. Kojo, T. Suihko, P. Eardley, D. Wisely, R. Hancock, and N. Georganopoulos. Mobility Related Terminology, Work in Progress, Internet Engineering Task Force draft-ietf-seamoby-mobility-terminology-06.txt, Février 2004.

[105] D. Marples. Naming and accessing internet appliances using extensions to the session initiation protocol. In *in Proc. of SIP 2000 Conference and Exhibition*, Paris, France, Mai 2000.

[106] Arifumi Matsumoto, Kenji Fujikawa, Yasuo Okabe, Masataka Ohta, Fumio Teraoka, Mitsunobu Kunishi, and Masahiro Ishiyama. Multihoming Support based on Mobile Node Protocol LIN6. In *The 2003 International Symposium on Applications and the Internet (SAINT)*, Orlando, Florida, USA, Janvier 2003.

[107] P. McCann. Mobile IPv6 Fast Handovers for 802.11 Networks, Work in Progress, Internet Engineering Task Force draft-ietf-mipshop-80211fh-00.txt, Février 2004.

[108] K.S. Meier-Hellstern, E. Alonso, and D.R. O'Neil. The Use of GSM to Support High Density Personal Communication. In *Record of the IEEE International Conference on Telecommunications*, Juin 1992.

[109] David G. Messerschmitt. The Convergence of Telecommunications and Computing : What Are The Implications Today ? *Proceedings of the IEEE*, 84 :1168–1188, Août 1996.

[110] The Multi-Generator Toolset, MGEN, `http://mgen.pf.itd.nrl.navy.mil/`.

[111] Mobile IPv6 Implementation for Linux (MIPL), `www.mipl.mediapoli.com/`.

[112] MobiWan : NS-2 extensions to study mobility in Wide-Area IPv6 Networks, `http://www.inrialpes.fr/planete/pub/mobiwan/`.

[113] Moby dick project, `http://wwwhome.cs.utwente.nl/havinga/mobydick.html`.

[114] Nicolas Montavont, Thierry Ernst, and Thomas Noël. Multihoming in Nested Mobile Networks. In *IEEE Computer Society Press in the proceedings of the 2004 International Symposium on Applications and the Internet - Workshops (SAINT 2004 workshops)*, Tokyo, Japon, 25-30 Janvier 2004.

[115] Nicolas Montavont, Thierry Ernst, Ryuji Wakikawa, and Thomas Noël. Problem Statement for Mutlihomed MN, Work in Progress, Internet Engineering Task Force draft-montavont-mobileip-multihoming-pb-statement.txt, Octobre 2003.

[116] Nicolas Montavont and Thomas Noël. La mobilité dans les réseaux IP, livrable LSIIT pour le CRE France Télécom R&D, Novembre 2001.

[117] Nicolas Montavont and Thomas Noël. Anticipation des Handovers dans les Réseaux sans Fils. In *16em congrès DNAC : de nouvelles architectures pour les communications, la Génération WIFI et l'Internet Ambiant*, Paris, France, 2-4 Décembre 2002.

[118] Nicolas Montavont and Thomas Noël. Fast Handover Protocol over IEEE 802.11b WLANs. In *IEEE International Symposium on advances in Wireless Communications (ISWC'2002)*, Victoria, Canada, Septembre 2002.

[119] Nicolas Montavont and Thomas Noël. Handover Management for Mobile Nodes in IPv6 Networks. *IEEE Communications Magazine*, 40(8) :38– 43, Août 2002.

[120] Nicolas Montavont and Thomas Noël. A New Middleware for IPv6 Mobile Computing. In *in proceedings of the International Conference on Networks (ICON 2003)*, Sydney, Australie, Septembre 2003.

[121] Nicolas Montavont and Thomas Noël. Analysis and Evaluation of Mobile IPv6 Handovers over Wireless LAN. *Journal de l'ACM, Mobile Networking and Applications (MONET), special issue on Mobile Networking through IPv6 or IPv4*, 8(6) :643 – 653, Novembre 2003.

[122] Nicolas Montavont and Thomas Noël. Home Agent Filtering for Mobile IPv6, Work in Progress, Internet Engineering Task Force draft-montavont-mobileip-ha-filtering-v6-00.txt, Juillet 2003.

[123] Nicolas Montavont and Thomas Noël. Mobile IPv6 and WLANs : Campus Scale Experimentation and Evaluation. In *Workshop on Promoting WLANs and Virtual Campus - Making Broadband Access a Priority*, Lisbonne, Portugal, 14 Février 2003.

[124] Nicolas Montavont, Thomas Noël, and Philippe Bertin. Parameters Abstraction to Optimize Mobility Control. In *in proceedings of the 6th International Conference on Advance Communication Technology (ICACT 2004)*, Phoenix Park, Corée, Février 2004.

[125] Nicolas Montavont, Thomas Noël, and Mohamed Kassi-Lahlou. Multiple Interfaces Management, livrable SP1 LSIIT Cyberté, Avril 2002.

[126] Nicolas Montavont, Thomas Noël, and Mohamed Kassi-Lahlou. MIPv6 for Multiple Interfaces (MMI), Work in Progress, Internet Engineering Task Force draft-montavont-mobileip-mmi-01.txt, Octobre 2003.

[127] Nicolas Montavont, Thomas Noël, and Mohamed Kassi-Lahlou. Description and Evaluation of Mobile IPv6 for Multiple Interfaces. In *in proceedings of the IEEE Wireless Communication and Networking Conference (WCNC 2004)*, Atlanta, Georgie, USA, 21-25 Mars 2004.

[128] Nicolas Montavont, Thomas Noël, Jean-Marc Muller, Nicolas Dichtel, and Mohamed Kassi-Lahlou. MIMA specification, version 2.0, livrable SP1 LSIIT Cyberté, Avril 2003.

[129] MPEG4IP - Open Streaming for Video and Audio, `http://www.mpeg4ip.sourceforge.net`.

[130] Multi6, Internet Engineering Task Force Working Group, `http://www.ietf.org/html.charters/multi6wg-charter.html`.

[131] F. Muratore. *UMTS Mobile Communications for the Future.* F. Muratore, 2001.

[132] N. Nakajima, A. Dutta, S. Das, and H. Schulzrinne. Handoff Delay Analysis and Measurement for SIP based mobility in IPv6. In *Proceedings of he IEEE International Conference on Communications (ICC'03)*, pages 1085 – 1089, Anchorage, Alaska, USA, Mai 2003.

[133] T. Narten, E. Nordmark, and W. Simpson. Neighbor Discovery for IP Version 6, Internet Engineering Task Force Request For Comments (RFC) 2461, Décembre 1998.

[134] Nautilus 6 Project, `http://www.nautilus6.org/`.

[135] Projet RNRT Cyberté, `http://cyberte.u-strasbg.fr/`.

[136] R. Nelson, G. Daley, and N. Moore. Implementation of Hierarchical Mobile IPv6 for Linux. In *In the proceedings of The Sixth International Symposium on Communications Interworking (IFIP Interworking 2002)*, Santa Barbara, CA USA, Octobre 2002.

[137] Pekka Nikander, Jukka Ylitalo, and Jorma Wall. Integrating Security, Mobility, and Multi-Homing in a HIP Way. In *in Proc. Network and Distributed Systems Security Symposium*, pages 87–89, San Diego, CA, Février 2003.

[138] Eric Njedjou, Franck Lebeugle, and Nicolas Montavont. Link Triggers Assisted Optimizations For Mobile IPv4/v6 Vertical Handovers. In *13th IST Mobile and Wireless Communications Summit 2004*, Lyon, France, Juin 2004.

[139] Thomas Noel. M-LAR : A New Protocol for Communications with Mobile Hosts. In *10th IEEE International Conference on Telecommunications*, Papeete, Polynésie française, Février 2003.

[140] Thomas Noel, Dominique Grad, Jean-Jacques Pansiot, and A. Alloui. LAR : Un protocole de routage intégrant la mobilité. *Calculateurs Parallèles "Routage dans les réseaux"*, 11 :33–57, 1999.

[141] Novell. Advanced NetWare V2.1 Internetwork Packet Exchange Protocol (IPX) with Asynchronous Event Scheduler (AES), Octobre 1986.

[142] Thomas Noël, Nicolas Montavont, and Philippe Bertin. Mobile IPv6 et WLAN : Expérimentation et évaluation à l'échelle d'un campus. In *16em congrès DNAC : de nouvelles architectures pour les communications, la Génération WIFI et l'Internet Ambiant*, Paris, France, 2-4 Décembre 2002.

[143] The Network Simulator - ns-2, `http://www.isi.edu/nsnam/ns/`.

[144] Data Processing Open System Interconnection, Basic Reference Model, ISO IS 7498, 1984.

[145] S. Daniel Park, Eric Njedjou, and Nicolas Montavont. L2 Triggers Optimized Mobile IPv6 Vertical Handover : the 802.11/GPRS Example, Work in Progress, Internet Engineering Task Force draft-daniel-mip6-optimized-vertical-handover-00.txt, Janvier 2004.

[146] Charles Perkins. IP Encapsulation within IP, Work in Progress, Internet Engineering Task Force Request For Comments (RFC) 2003, 1996.

[147] Charles Perkins and David B. Johnson. Mobility Support in IPv6. In *ACM/IEEE International Conference on Mobile Computing and Networking*, 1996.

[148] Charles Perkins and David B. Johnson. Route Optimization in Mobile IP, Work in Progress, Internet Engineering Task Force draft-ietf-mobileip-optim-09.txt, February 2000.

[149] Charles Perkins, Andrew Myles, and David B. Johnson. The Internet Mobile Host Protocol (IMHP). In *INET'94*, Juin 1994.

[150] J. Postel. User Datagram Protocol, Internet Engineering Task Force Request for Comments (RFC) 768, Août 1980.

[151] J. Postel. Internet Protocol, Internet Engineering Task Force Request For Comments (RFC) 791, Septembre 1981.

[152] J. Postel. Transmission Control Protocol, Internet Engineering Task Force Request for Comments (RFC) 793, Septembre 1981.

[153] R. Ramjee, T. La Porta, S. Thuel, K. Varadhan, and L. Salgarelli. IP Micro-mobility support using HAWAII, Internet Engineering Task Force draft-ietf-mobileip-hawaii-00.txt, Juin 1999.

[154] Robust Audio Tool (RAT), `http://www-mice.cs.ucl.ac.uk/multimedia/software/rat/`.

[155] REAL Simulator, `http://www.cs.cornell.edu/skeshav/real/overview.html`.

[156] J. Romkey. A nonstandard for transmission of IP datagrams over serial lines : SLIP, Internet Engineering Task Force Request For Comments (RFC) 1055, Juin 1988.

[157] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, and E. Schooler. SIP : Session Initiation Protocol, Internet Engineering Task Force Request For Comments (RFC) 2543, Juin 2002.

[158] Scripts RUBY, `http://www.ruby-lang.org/en/`.

[159] L. Sahasrabuddhe and B. Mukherjee. Multicast routing algorithms and protocols : a tutorial. *IEEE Network*, 14(1) :90–102, Jan-Feb 2000.

[160] Apostolis K. Salkintzis, Chad Fors, and Rajesj Pazhyannur. WLAN - GPRS Integration for Next Generation Mobile Data Networks. *IEEE Wireless Communications*, 9 :112–124, Octobre 2002.

[161] H. Schulzrinne and J. Rosenberg. The session initiation protocol : Internet-centric signaling. *IEEE Communications Magazine*, 38, Octobre 2000.

[162] Henning Schulzrinne and Elin Wedlund. Application-Layer Mobility Using SIP. *Mobile Computing and Communications Review*, 1, Mars 2001.

[163] Z. Shelby, D. Gatzounas, A. Campbell, and C-Y. Wan. Cellular IPv6, Internet Engineering Task Force draft-shelby-seamoby-cellularipv6-00.txt, Novembre 2000.

[164] K. Shima. Route Optimization hint option, Work in Progress, Internet Engineering Task Force draft-shima-mip6-rohints-00, Octobre 2003.

[165] Prasan De Silva and Harsha Sirisena. A Mobility Management Protocol for IP-based Cellular Networks. *IEEE Wireless Communications, special issue on Mobile and Wireless Internet*, 9 :31–37, Juin 2002.

[166] Implémentation de sip, `http://www.sipcenter.com/testarea/testdevelopersnews.html`.

[167] H. Soliman, K. ElMalki, and C. Castelluccia. Per-flow movement in mipv6, work in progress, Internet Engineering Task Force, draft-soliman-mobileip-flow-move-03.txt, June 2003.

[168] Hesham Soliman, Claude Castelluccia, Karim El Malki, and Ludovic Bellier. Hierarchical Mobile IPv6 mobility management (HMIPv6), Work in Progress, Internet Engineering Task Force draft-ietf-mipshop-hmipv6-01.txt, Février 2004.

[169] Mark Stemm and Randy H. Katz. Vertical Handoffs in Wireless Overlay Networks. *Journal de l'ACM, Mobile Networks and Applications (MONET)*, 3, 1998.

[170] R. Stewart, Q. Xie, K. Morneault, C. Sharp, and H. Schwarzbauer and. Stream Control Transmission Protocol (SCTP), Internet Engineering Task Force Request For Comments (RFC) 2960, Octobre 2000.

[171] A. Tanenbaum. *Réseaux (4ème édition)*. Pearson Education, 2003.

[172] Outil de capture de trafic, TCPDUMP, `http://www.tcpdump.org/`.

[173] Fumio Teraoka, Masahiro Ishiyam, and Mitsunobu Kunishi. LIN6 : A Solution to Multihoming and Mobility in IPv6, Internet Engineering Task Force, draft-teraoka-multi6-lin6-00.txt, Décembre 2003.

[174] Fumio Teraoka and Mario Tokoro. Host Migration Transparency in IP Networks : The VIP Approach. *Computer Communications Rev., ACM*, Janvier 1993.

[175] Fumio Teraoka, Keisuke Uehara, Hideki Sunahara, and Jun Murai. VIP : A Protocol Providing Host Mobility. *Communications of the ACM*, Août 1995.

[176] S. Thomson and T. Narten. IPv6 Stateless Address Autoconfiguration, Internet Engineering Task Force Request For Comments (RFC) 2462, Décembre 1998.

[177] Pascal Thubert and Nicolas Montavont. Nested Nemo Tree Discovery, Work in Progress, Internet Engineering Task Force draft-thubert-tree-discovery-00.txt, Mai 2004.

[178] Marc Torrent-Moreno, Xavier Perez-Costa, and Sebastien Sallent-Ribes. A Performance Study of Fast Handovers for Mobile IPv6. In *28th Annual IEEE International Conference on Local Computer Networks*, Bonn/Konigswinter, Germany, Octobre 2003.

[179] Jean Tourrilhes and Casey Carter. P-Handoff : a protocol for fine graines peer-to-peer vertical handoff. In *The 13th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC) 2002 - HP external report HPL-2002-258*, Lisbonne, Portugal, Septembre 2002.

[180] Transport Area, Internet Engineering Task Force Working Group, `http://www.ietf.org/html.charters/tsvwg-charter.html`.

[181] Test TCP, `http://www.netcordia.com/tools/tools/TTCP/ttcp.html`.

[182] UMTS forum, `http://www.umts-forum.org`.

[183] A. Valko. Cellular IP : A New Approach to Internet Host Mobility. *ACM SIGCOMM Computer Communication*, 29 :50 – 65, Janvier 1999.

[184] VIdeo Conference tool, `http://www-mice.cs.ucl.ac.uk/multimedia/software/vic/`.

[185] C. Vogt, R. Bless, M. Doll, and T. K'fner. Early Binding Updates for Mobile IPv6, Work in Progress, Internet Engineering Task Force draft-vogt-mip6-early-binding-updates-00, Février 2004.

[186] R. Wakikawa, K. Uehara, and T. Ernst. Multiple careof address Registration on Mobile IPv6, Work in Progress, Internet Engineering Task Force draft-wakikawa-mobileip-multiplecoa-01.txt, Juin 2003.

[187] Ryuji Wakikawa, Vijay Devarapalli, and Pascal Thubert. Inter Home Agents Protocol (HAHA), Work in Progress, Internet Engineering Task Force draft-wakikawa-mip6-nemo-haha-01.txt, February 2004.

[188] Ryuji Wakikawa, Susumu Koshiba, Keisuke Uehara, and Jun Murai. Multiple Network Interfaces Support by Policy-Based Routing on Mobile IPv6. In *Proc. of the 2002 International Conference on Wireless Networks (ICWN)*, Las Vegas NV, USA, Juillet 2002.

[189] Elin Wedlund and Henning Schulzrinne. Mobility support using SIP. In *Proceedings of the 2nd ACM international workshop on Wireless mobile multimedia*, pages 76 – 82, Seattle, Washington, United States, 1999.

[190] Gang Wu, Paul J.M. Havinga, and Mitsuhiko Mizuno. Wireless Internet over Heterogeneous Wireless Networks. In *IEEE GLOBECOM, IEEE Computer Society ISBN 0-7803-7208-5*, pages 1759–1765, San Antonio, Novembre 2001.

[191] Gang Wu, Mitsuhiko Mizuno, and Paul J.M. Havinga. MIRAI Architecture for Heterogeneous Network. *IEEE Communications Magazine*, pages 126–134, Février 2002.

[192] Jon Chiung-Shien Wu, Chieh-Wen Cheng, Nen-Fu Huang, and Gin-Kou Ma. Intelligent Handoff for Mobile Wireless Internet. *Journal de l'ACM, Mobile Networks and Applications (MONET)*, 6 :67–79, Janvier 2001.

[193] Lin Xu, Toni Paila, Wolfgang Hansmann, and Matthias Frank. IPv6 based Infrastructure for Wireless IP in Multi-Radio Environments with Quality of Service Support. In *26th Annual IEEE International Conference on Local Computer Networks (LCN 2001)*, Bonn/Konigswinter, Germany, Novembre 2001.

[194] Y. Imai Y. Ezaki. Mobile IPv6 handoff by Explicit Multicast, Work in Progress, Internet Engineering Task Force draft-ezaki-smoothhandoff-xcast6-00.txt, Juin 2002.

[195] A. Yegin, E. Njedjou, S. Veerepalli, N. Montavont, and T. Noël. Link-Layer Hints for Detecting Network Attachments, Work in Progress, Internet Engineering Task Force draft-yegin-dna-l2-hints-01.txt, Août 2004.

[196] C. Zaccone, Y. T'Joens, and B Sales. Address reuse in the Internet, adjourning or suspending the adoption of IP next generation ? In *Proceedings of Icon'00*, September 2000. Singapore.

[197] Xinhua Zhao, Claude Castelluccia, and Mary Baker. Flexible network support for mobility. In *Fourth ACM International Conference on Mobile Computing and Networking (MOBICOM'98) ISBN 1-58113-035-X*, pages 145–156, Dallas, Texas, USA, Octobre 1998.

[198] Xinhua Zhao, Claude Castelluccia, and Mary Baker. Flexible network support for mobility. *Journal de l'ACM Kluwer, Mobile Networks and Applications (MONET), Special Issue on Management of Mobility in Distributed Systems*, 6 :137–149, Mars/Avril 2001.

[199] C. Perkins (éditeur). IP Mobility Support for IPv4, Internet Engineering Task Force Request For Comments (RFC) 3220, Janvier 2002.

[200] James Kempf (éditeur), O. Levkowetz, P. Calhoun, G. Kenward, H. Syed, J. Manner, M. Nakhjiri, G. Krishnamurthi, R. Koodli, K. Atwald, M. Thomas, M. Horan, and P. Neumiller. Problem Description : Reasons For Performing Context Transfers Between Nodes in an IP Access Network, Request for Comments (RFC) 3374, Informational, Internet Engineering Task Force, Septembre 2002.

[201] R. Koodli (éditeur), G. Tsirtsis, A. Yegin, C. Perkins, G. Dommety, K. El-Malki, and M. Khalil. Fast Handovers for Mobile IPv6, Work in Progress, Internet Engineering Task Force draft-ietf-mipshop-fast-mipv6-01.txt, Janvier 2004.

Liste des publications

Journaux internationaux

- **Analysis and Evaluation of Mobile IPv6 Handovers over Wireless LAN**, N. Montavont, T. Noel, Mobile Networking and Applications (MONET), special issue on Mobile Networking through IPv6 or IPv4, Volume 8, Numéro 6, p643-653, Novembre 2003.

- **Handover Management for Mobile Nodes in IPv6 Networks**, N. Montavont, T. Noel, IEEE Communication Magazine, volume 40, numéro 8, p38-43, Août 2002.

Conférences internationales

- **Link Triggers Assisted Optimizations For Mobile IPv4/v6 Vertical Handovers**, Eric Njedjou, Franck Lebeugle and Nicolas Montavont, 13th IST Mobile and Wireless Communications Summit 2004, Lyon, France, Juin 2004.

- **Description and Evaluation of Mobile IPv6 for Multiple Interfaces**, N. Montavont, M. Kassi-Lahlou, T. Noel, IEEE Wireless Communication and Networking Conference (WCNC 2004), Atlanta, Georgie, USA, 21-25 Mars 2004.

- **Parameters Abstraction to Optimize Mobility Control**, N. Montavont, T. Noel and P. Bertin, The 6th International Conference on Advance Communication Technology (ICACT 2004) Phoenix Park, Corée, 9-11 Février 2004.

- **Multihoming in Nested Mobile Networks**, N. Montavont, T. Ernst, T. Noel, IEEE Computer Society Press in the proceedings of the 2004 International Symposium on Applications and the Internet - Workshops (SAINT 2004 workshops), Tokyo, Japon, 26-30 Janvier 2004.

- **A New Middleware for IPv6 Mobile Computing**, N. Montavont, T. Noel, International Conference on Networks (ICON 2003), Sydney, Australie, 28 Septembre - 1er Octobre, 2003.

- **Optimized Support of Multiple Wireless Interfaces within an IPv6 End-terminal**, F. André, J-M. Bonnin, B. Deniau, K. Guillouard, N. Montavont, T. Noel, L. Suciu, Smart Objects Conference (SOC'2003), Grenoble, France, 15-17 Mai, 2003.

- **Mobile IPv6 and WLANs : Campus Scale Experimentation and Evaluation**, Nicolas Montavont, T. Noel, Workshop on Promoting WLANs and Virtual Campus - Making Broadband Access a Priority, Lisbonne, Portugal, 14 Février, 2003.

- **Fast Handover Protocol over IEEE 802.11b WLANs**, N. Montavont, T. Noel, IEEE International Symposium on advances in Wireless Communications (ISWC'2002), Victoria, Canada, 23 Septembre 2002.

- **Handoffs Management in Multiple Access Technologies**, N. Montavont, T. Noel, International Conference on Computer and Information Science (ICIS'01), Orlando, USA, 3-5 Octobre, 2001.

Conferences nationales

- **Anticipation des Handovers dans les Réseaux sans Fil**, N. Montavont, T. Noel, 16em congrès DNAC : de nouvelles architectures pour les communications, la Génération WIFI et l'Internet Ambiant, Paris, 2-4 Décembre 2002.

- **Mobile IPv6 et WLAN : Expérimentation et évaluation à l'échelle d'un campus**, T. Noel, N. Montavont, P. Bertin, 16em congrès DNAC : de nouvelles architectures pour les communications, la Génération WIFI et l'Internet Ambiant, Paris, 2-4 Décembre 2002.

Internet-Drafts à l'IETF

- **Detecting Network Attachment in IPv6 - Best Current Practices**, S. Narayanan, G. Daley and N. Montavont, draft-narayanan-dna-bcp-00.txt, DNA working group, Juin 2004.

- **Nested Nemo Tree Discovery**, Pascal Thubert, Nicolas Montavont, draft-thubert-tree-discovery-00.txt, NEMO working group, Mai 2004.

- **L2 Triggers Optimized Mobile IPv6 Vertical Handover : the 802.11/GPRS Example**, S. Daniel Park, Eric Njedjou, Nicolas Montavont, presenté à l'IETF 59 en Corée, MOBOPS working group, Janvier 2004.

- **Goals and Benefits of Multihoming**, Thierry Ernst, Nicolas Montavont, Ryuji Wakikawa, Eun Kyoung Paik, Chan-Wah Ng, Koojaana Kuladinithi, Thomas Noel, draft-multihoming-generic-goals-and-benefits-00, présenté en Corée à l'IETF 59, Février 2004.

- **Problem Statement for Mutlihomed MN**, Nicolas Montavont, Thierry Ernst, Ryuji Wakikawa, Thomas Noel, draft-montavont-mip6-multihoming-pb-statement-00.txt, MIP6 working group, October 2003, presenté à l'IETF 58 à Minneapolis, Octobre 2003.

- **MIPv6 for Multiple Interfaces**, N. Montavont, T. Noel, M. Kassi-Lahlou, draft-montavont-mobileip-mmi-01.txt, Octobre 2003.

- **Home Agent Filtering for Mobile IPv6**, N. Montavont, T. Noel, draft-montavont-mobileip-ha-filtering-v6-00.txt, MIP6 Working Group, Juillet 2003.

- **Link-Layer Hints for Detecting Network Attachments**, A. yegin, E. Njedjou, S. Veerepalli, N. Montavont, T. Noel, draft-yegin-dna-l2-hints-01.txt, DNA Working Group, Février 2004.

- **Parameters for Link Hints**, P. Bertin, N. Montavont, T. Noel, draft-bertin-hints-params-00.txt, Internet Engineering Task Force Draft, Août 2003.

Rapports de contrats

- **Wide Report**, Nautlilus report 1, Janvier 2003.

- **Multiple Interfaces Management Architecture Specification**, Nicolas Montavont, Thomas Noel, Jean-Marc Muller, Nicolas Dichtel, livrable RNRT projet Cyberté, Avril 2003.

- **Basic Scenarios of Multiple Interfaces Mobile Nodes**, Nicolas Montavont, Thomas Noel, Mohammed Kassi-Lahlou, livrable France télécom, Avril 2002.

- **Étude de la mobilité IP et analyse des réseaux IEEE 802.11b**, Ni-

colas Montavont, livrable France Télécom, Novembre 2001.

Glossaire

- **3GPP** : Third Generation Partnership Project. Organisme en charge de la standardisation des protocoles liés à la téléphonie mobile de 3ème génération.

- **Backoff** ou **Procédure de backoff** : Algorithme d'attente exponentiel pour l'accès au médium dans les réseaux IEEE 802.11.

- **BNEP** : Bluetooth Network Encapsulation Protocol. Protocole utilisé par les profils Bluetooth définissant un format de paquet pour encapsuler des protocoles réseau tels que IP.

- **BSS** : Basic Service Set. Ensemble de stations IEEE 802.11 contrôlées par une seule fonction de coordination.

- **CRC** : Cyclic Redundancy Cheksum. Somme sur les bits de données permettant de contrôler l'intégrité du paquet.

- **CSMA** : Carrier Sense Multiple Access. Méthode d'accès au lien utilisant la détection de porteuse pour attribuer le médium partagé.

- **DAD** : Duplicate Address Detection. Processus de découverte des adresses IPv6 dupliquées sur un lien local.

- **DCF** : Distributed Coordination Function. Algorithme non centralisé d'accès au médium dans les normes IEEE 802.11.

- **DNS** : Domain Name System. Système distribué sur l'Internet permettant de faire l'association entre les noms de machine et leur(s) adresse(s) IP.

- **DIFS** : Distributed Interframe Space. Intervalle de temps (50 micro-secondes) séparant la fin d'une émission et le début de la fenêtre de contention dans l'algorithme d'accès au canal de IEEE 802.11 (DCF).

- **DSSS** : Direct Sequence Spread Spectrum. Mode de découpage d'une bande de fréquence permettant de répartir le signal sur un ensemble de fréquences voisines.

- **FHSS** : Frequency Hop Spread Spectrum. Mode de découpage d'une bande de fréquence créant des canaux à partir d'un motif cyclique de sauts duquel on déduit la fréquence utilisée à l'instant voulu.

- **Fast Handover** : Fast Handovers for Mobile IPv6 (FMIPv6). Protocole d'optimisation des déplacements des équipements supportant Mobile IPv6.

- **Handover** : Processus consistant à changer de point d'attachement sur un réseau, appelé également *Handoff*. Plus particulièrement, le handover de niveau 2 est le changement de point d'accès et le handover de niveau 3 est le changement de sous-réseau IPv6.

- **Mobile IPv6 Hiérarchique** : Hierarchical Mobile IPv6 (HMIPv6). Protocole de gestion hiérarchisée de la mobilité IPv6.

- **IEEE** : Institute of Electronics and Eletricity Engineers. Société américaine responsable du développement des spécifications des normes réseau.

- **IETF** : Internet Engineering Task Force. Organisme de standardisation des protocoles pour l'Internet.

- **Internet-draft** : Spécifications préliminaires d'un protocole à l'IETF, stade préalable au RFC.

- **IP** : Internet Protocol. Protocole utilisé sur l'Internet pour véhiculer des paquets.

- **IPv6** : Internet Protocol version 6. Successeur du protocole IP, également appelé protocole Nouvelle Génération.

- **LAN** : Local Area Network. Acronyme regroupant l'ensemble des composants définissant un réseau local.

- **LIN6** : Location Independent Network Architecture for IPv6. Protocole de gestion de la mobilité et du multihoming des nœuds proposant la séparation

de la couche réseau en une sous-couche logique et une sous-couche de localisation.

- **MIPv6** : Protocole de gestion des stations mobiles dans l'Internet Nouvelle Génération.

- **MPEG** : Motion Picture Expert Group. Organisme de standardisation de protocoles liés à la compression audio (MP3) et vidéo (MPEG4).

- **Multihoming** : Possibilité pour un nœud de choisir entre plusieurs adresses pour échanger des données avec des correspondants. Un nœud peut avoir plusieurs adresses simultanément soit parce qu'il est équipé de plusieurs interfaces réseau, soit parce qu'il reçoit plusieurs préfixes sur le(s) lien(s) au(x)quel(s) il est connecté.

- **NEMO** : Network Mobility. Groupe de travail de l'IETF sur les réseaux mobiles.

- **Pagination** : Processus de localisation des nœuds inactifs.

- **Paquet DHx** : Data High Rate packet. Paquet utilisé dans la couche BaseBand de la pile Bluetooth, prennant 1, 3 ou 5 slots. Uniquement 16 bits sont utilisés pour le CRC dans chaque paquet.

- **Paquet DMx** : Data Medium Rate packet. Paquet utilisé dans la couche BaseBand de la pile Bluetooth, prennant 1, 3 ou 5 slots. Un tiers des bits est consacré au contrôle et à la détection d'erreurs.

- **PCF** : Point Coordinator Function. Algorithme centralisé d'accès au médium dans les normes IEEE 802.11.

- **OSI** : Open System Interconnexion. Schéma en couches permettant le développement de protocoles fonctionnant en milieu hétérogène.

- **RFC** : Request for Comments. Spécifications référencant l'état de définition d'un protocole de l'IETF.

- **RFCOMM** : Emulation de port série au-dessus du protocole L2CAP de la pile Bluetooth, permettant d'établir une liaison PPP.

– **SIFS** : Short Interframe Space. Intervalle de temps (10 micro-secondes) minimum séparant deux trames consécutives (eg. temps entre une trame de son acquittement).

– **SIP** : Session Initiation Protocol. Protocole de niveau applicatif dédié à la création, à la modification et à la terminaison de sessions multimédia entre plusieurs utilisateurs.

– **WLAN** : Wireless LAN. Acronyme définissant les normes régissant les réseaux locaux sans fil.