REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université des Sciences et de la Technologie Houari Boumedienne

Faculté d'Electronique et d'Informatique
Département d'Informatique

Thèse de Magistère

Impact de la mobilité sur le protocole TCP

Présentée par : Bouziane Nabila

Proposée et dirigée par : Pr. N.Badache
Mr. D.Tandjaoui

Président du jury :
Pr Ahmed Nacer (USTHB , Faculté d'Electronique et d'Informatique,
Département d'Informatique)

Membre du jury :
Pr Larabi (USTHB , Faculté d'Electronique et d'Informatique,
Département d'Informatique)
Pr Azzoune (USTHB ,Faculté d'Electronique et d'Informatique,
Département d'Informatique)

Année 2004

Résumé

Supporter la mobilité sur Internet réside dans la capacité de permettre aux mobiles de se déplacer entre plusieurs cellules, sous réseaux et domaines tout en leur garantissent une bonne qualité de services. Cependant l'intégration des réseaux sans fil dans Internet pose un nouveau challenge. La principale raison est que le protocole TCP / IP a été conçu pour des réseaux fixes.

Les mobiles nécessitent un changement dans le protocole de routage de telle sorte que les paquets soient délivrés à la bonne destination. Mobile IP (MIP) est une approche qui a été proposée pour résoudre ce problème. Il permet aux hôtes mobiles de se déplacer d'un sous-réseau IP à un autre; il résout le problème de la "macro-mobilité". Il est moins bien adapté à la "micro-mobilité", lorsque les mobiles effectuent des handoffs fréquents dans une même zone géographique. C'est donc pour cela qu'un certain nombre de protocoles de micro-mobilité visant à améliorer les performances de MIP ont été proposés. Ces nouveaux protocoles perturbent le comportement des applications à temps réel et surtout les applications basées sur le protocole TCP. Dans cette thèse nous étudierons les protocoles Cellular IP et Hawaii et nous évaluerons leurs impacts sur le protocole TCP.

Mots clés : Mobile IP, Mobilité, Handoff , TCP, Cellular IP, Hawaii.

Remerciements

Ce mémoire est l'aboutissement de longues heures de travail portant sur un sujet passionnant du domaine de la mobilité. Le monde très vaste des télécommunications ne peut se prétendre d'être entièrement connu, mais ce travail m'a permis d'en découvrir quelques facettes.

Je voudrais remercier tout particulièrement mes promoteurs, les Professeurs N.Badache et D.Tandjaoui, de m'avoir supervisé et encouragé tout au long de ces deux années de recherche dans la poursuite quotidienne de mon travail. Leurs expériences et leurs connaissances m'ont apporté une aide précieuse dans la réalisation de ce travail.

Je dédie cette thèse à ma famille. J’y puise un soutien, une confiance et un amour indéfectibles que j’espère rendre pareillement et transmettre à mon tour.

Merci enfin à ceux que je n'ai pas cités, mais qui se reconnaîtront, pour les échanges fructifiants lors des longues conversations que nous avons pu tenir.

Table des matières

Introduction **1**

Chapitre 1 Les environnements mobiles **3**
1.1 Introduction 3
1.2 Les environnements mobiles 3
1.3 Caractéristiques des unités mobiles 5
1.4 L'utilisation des ondes radio dans la communication sans fil 5
1.5 La fiabilité de la communication sans fil 5
1.6 La communication cellulaire 6
1.7 Quelques éléments de l'infrastructure sans fil 6
1.8 Conclusion 6

Chapitre 2 Le protocole TCP **8**
2.1 Introduction 8
2.2 Fonctionnalités de TCP 8
2.3 Interfaces 9
2.4 Fonctionnement du protocole TCP 10
2.4.1 Eléments constitutifs du réseau 10
2.4.2 Modèle de fonctionnement 10
2.4.3 Format de l'en-tête 10
2.4.4 Etablissement et libération de connexion TCP 11
2.4.5 Echange des données TCP 12
2.5 Gestion des temporisations 14
2.6 Gestion de la fenêtre de transmission 14
2.7 Contrôle de trafic et de congestion 15
2.7.1 Algorithmes du démarrage lent et d'évitement de congestion 15
2.7.2 Algorithmes de retransmission et de recouvrement rapide 17
2.8 Les différentes versions de TCP 19
2.9 Conclusion 21

Chapitre 3 La macro mobilité **22**
3.1 Introduction 22
3.2 Problème de la mobilité IP 22
3.3 Principaux enjeux de la mobilité dans IP 23
3.4 Mobile IP 23
3.5 Quelques Définitions 24
3.6 Fonctionnement du protocole Mobile IP 24
3.6.1 Principe de base 24
3.6.2 Adresse temporaire 25
3.6.3 Encapsulation IP dans IP 26
3.6.4 Découverte d'agent (Agent discovery) 26
3.6.5 Détection de mouvement (Handoffs) 27
3.6.6 Enregistrement 27

3.6.7 Routage....29
3.6.8 Extensions....31
3.7 Limites de Mobile IP....31
3.8 Conclusion....31

Chapitre 4 La micro mobilité....32
4.1 Introduction....32
4.2 Courte description des protocoles de micro mobilité....32
4.3 Coexistence de plusieurs solutions dans l'Internet....33
4.4 Cellular IP....34
4.4.1 L'architecture d'un réseau d'accès sans fil (Wireless access network)....35
4.4.2 Routage....35
4.4.3 La passerelle Cellular IP....38
4.4.4 Handoff....39
4.4.5 Pagination....41
4.4.6 Paramètres du protocole....43
4.4.7 Sécurité....44
4.5 HAWAII....45
4.5.1 Architecture du réseau....45
4.5.2 Routage....46
4.5.3 Handoff....47
4.5.4 Pagination....49
4.5.5 Détails protocolaires....49
4.5.6 Sécurité....53
4.6 Tableau récapitulatif des différentes caractéristiques des deux protocoles étudiés....54
4.7 Conclusion....55

Chapitre 5 Etudes réalisées....56
5.1 Introduction....56
5.2 Comparaison des protocoles de micro mobilité....56
5.3 Performance de TCP et UDP sous les protocoles de micro mobilité....58
5.4 Conclusion....60

Chapitre 6 Simulation....61
6.1 Introduction....61
6.2 Scénario de simulation....61
6.3 Comportement de TCP sous Cellular IP....63
6.3.1 Débit TCP....63
6.3.2 Fenêtre de congestion....64
6.3.3 Vitesse de l'hôte mobile....64
6.3.4 Taille de la zone de chevauchement....68
6.3.5 Variantes de TCP....70

6.4 Comportement de TCP sous Hawaii 71
6.4.1 Débit TCP 71
6.4.2 Fenêtre de congestion 72
6.4.3 Vitesse de l'hôte mobile 72
6.4.4 Variantes de TCP 76
6.5 Comparaison 76
6.5.1 Vitesse de l'hôte mobile 76
6.5.2 Nombre de handoffs 78
6.5.3 Variantes TCP 80
6.6 Conclusion 81

Conclusion 83

Annexes
Annexe 1 : Les différentes techniques d'évaluation des performances 87
Annexe 2 : Formats des paquets 92
Annexe 3 : MIPv4 et MIPv6 96
Annexe 4 : Présentation du simulateur réseau NS-2 100

Références 108
Table des figures 112
Table des tableaux 114

Introduction

Le marché des terminaux portables (téléphones, ordinateurs portables, assistants personnels, etc.) est en forte croissance. Avec les réseaux sans fil, l'utilisateur peut accéder à des informations sans devoir chercher un emplacement de branchement. Comparé avec l'ancien environnement (l'environnement statique), ce nouvel environnement appelé environnement mobile permet aux unités de calcul, une libre mobilité et ne pose aucune restriction sur la localisation des usagers.

Les environnements mobiles reposent sur une infrastructure de réseaux sans fil. Ce type de réseaux présente de nombreux avantages sur les réseaux câblés classiques, en termes de productivité, simplicité et coût. Ils assurent aux utilisateurs d'un réseau un accès aux informations en temps réel. Ce service est accessible quelle que soit leur localisation à l'intérieur d'une certaine zone de couverture.

Les réseaux mobiles sans fil peuvent être classés en deux catégories : les réseaux avec infrastructure qui utilisent généralement le modèle de la communication cellulaire, et les réseaux sans infrastructure ou réseaux ad hoc. Plusieurs systèmes utilisent déjà le modèle cellulaire et connaissent une très forte expansion (les réseaux GSM par exemple), mais requièrent une infrastructure logistique et matérielle fixe.

L'intégration de ce nouvel environnement mobile aux réseaux traditionnels et surtout Internet pose un nouveau challenge. En effet, l'ensemble des protocoles TCP / IP d'Internet a été, à l'origine, conçu sans aucune considération pour la mobilité des unités de calcul. Sa grande rigidité face aux environnements mobiles réside dans le service de routage offert par la couche IP "*Internet Protocol"* qui permet un routage des données dans un environnement statique.

Afin de supporter la mobilité, le protocole IP a été amélioré pour donner naissance au protocole Mobile IP [PER96a]. Mobile IP gère la mobilité des utilisateurs lorsqu'ils se déplacent d'un réseau à un autre. Sa simplicité et sa souplesse l'ont rendu plus déployable par rapport à d'autres protocoles. Cependant, Mobile IP présente un inconvénient majeur dans les environnements où les handoffs sont fréquents. Pour remédier à cet inconvénient, la mobilité a été divisée en deux environnements : macro et micro-mobilité. La macro-mobilité gère la mobilité à grande échelle, le protocole Mobile IP est le mieux adapté pour ce type de mobilité. La micro-mobilité gère la mobilité au sein d'un domaine et tout mouvement d'un nœud mobile est caché à l'agent mère du réseau mère. Plusieurs, protocoles de micro-mobilité ont été proposés parmi lesquels Hawaii et cellular IP sont les plus standardisé par l'IETF(*Internet Engeenering Task Force*).

Ces nouveaux protocoles résolvent le problème des handoffs fréquents. Cependant, ils perturbent énormément les applications des nœuds mobiles et surtout les applications TCP [NAG84]. Pour cela, la conception efficace de tels réseaux de communication ne peut pas être accomplie sans une bonne connaissance de leurs performances. Différentes techniques d'évaluation des performances existent, elles peuvent être classées en trois grandes catégories:

l’obtention de mesures directes sur le réseau (ou sur un prototype), les techniques analytiques et numériques et la simulation.

Deux types d'études ont été réalisés concernant la micro mobilité : Certaines comparent uniquement les différents protocoles de micro mobilité et d'autres évaluent les performances des protocoles TCP et UDP sous les protocoles de micro mobilité. Dans [REI01] et [BON01] une comparaison de quatre protocoles existants de micro-mobilité dans IP est détaillée. Cette comparaison est réalisée dans un cadre général permettant, non pas de s’attarder sur les défauts et qualités de tel ou tel protocole, mais de tirer des conclusions générales pour l’évaluation des propriétés de ceux-ci. Dans [GHA01], les performances de UDP et TCP ont été évaluées selon les schémas de handoff de Cellular IP.

Dans cette thèse, nous analyserons et évaluerons des performances du protocole TCP (*Transmission Control Protocol*) dans les environnements sans fil, en particulier les réseaux avec infrastructure. Plus précisément, nous simulerons et comparerons les performances de TCP sous les différents protocoles de micro-mobilité à savoir Hawaii et cellular IP. Ces évaluations et comparaisons des performances, nous permettront par exemple de comparer les différents protocoles TCP en fonction des applications et des services offerts; d’anticiper et de corriger des éventuels problèmes de performance avant qu’il ne soit trop tard et d’optimiser les ressources du réseau au fur et à mesure que les caractéristiques du trafic des utilisateurs changent.

Cette thèse est composée de six chapitres : dans le premier chapitre, on présente les environnements mobiles et les principaux concepts liés à ces derniers. Dans le deuxième chapitre, on détaille le fonctionnement du protocole TCP afin de pouvoir étudier son comportement sous les environnements mobiles. Dans le troisième chapitre, on présente "Mobile IP" l'un des protocoles de macro mobilité le plus utilisé, son fonctionnement, ses versions et ses limites. Le quatrième chapitre est consacré à la présentation des différents protocoles de micro mobilité existants dans le contexte des environnements mobiles. On décrit deux de ces protocoles à savoir Cellular IP et Hawaii. Le cinquième chapitre présente les différentes études réalisées pour étudier le comportement du protocole TCP sous les différents protocoles de micro-mobilité. Enfin, le dernier chapitre est consacré à l'évaluation et la comparaison des performances du protocole TCP sous les protocoles Cellular IP et Hawaii en utilisant une des techniques d'évaluation des performances : la simulation.

Chapitre 1 Les environnements mobiles

1.1 Introduction

L'évolution rapide de la technologie dans le domaine de la communication sans fil, a permis à des usagers munis d'unités de calcul portables d'accéder à l'information indépendamment des facteurs : temps et lieu. Ces unités, qui communiquent à travers leurs interfaces sans fil, peuvent être de diverses configurations : avec ou sans disque, des capacités de sauvegarde et de traitement plus ou moins modestes et alimentés par des sources d'énergie autonomes (batteries). Elles sont équipées d'interfaces de communication sans fil pour l'accès aux réseaux d'information.

L'environnement de calcul résultant, appelé environnement mobile ou nomade, n'astreint plus l'usager à une localisation fixe, mais lui permet une libre mobilité tout en assurant sa connexion avec le réseau.

Les environnements mobiles permettent une grande flexibilité d'emploi. En particulier, ils permettent la mise en réseau des sites dont le câblage serait trop onéreux dans leur totalité, voire même impossible, (par exemple en présence d'une composante mobile).

L'environnement mobile offre beaucoup d'avantages par rapport à l'environnement habituel. Cependant de nouveaux problèmes surgissent, causés par les nouvelles caractéristiques de cet environnement. Il est caractérisé par de fréquentes déconnexions, des limitations significatives du débit de transfert de l'information et des sources d'énergie, des restrictions sur les ressources utilisées (capacité disque et mémoire, vitesse du processeur) et des changements fréquents de localisation. De nouvelles solutions doivent être trouvées pour s'adapter aux limitations qui existent.

La mobilité et la portabilité offertes par les environnements mobiles, permettront le développement de nouvelles classes d'applications : services d'informations avec accès à diverses bases de données en tout lieu et tout temps (pages jaunes, distribution, spectacles, etc.) et des applications dites verticales relevant de domaines spécifiques : compagnies de location, localisation d'employés dans une entreprise, etc.

La messagerie électronique connaîtra un développement spectaculaire, les usagers munis de communicateurs pourront envoyer et recevoir des messages de n'importe où et les nouvelles électroniques leurs seront délivrées en fonction de leurs profils respectifs [IMI92]. La permanence de la connexion des usagers aux réseaux d'information, indépendamment de leurs positions géographiques contribuera au développement des applications coopératives [DAV92, DAV93, IMI94].

Ce chapitre a pour but de présenter l'environnement mobile, et les principaux concepts liés à ce nouvel environnement. Il introduit la technologie de communication sans fil utilisée par les réseaux mobiles. Quelques principales notions nécessaires à la compréhension de ces systèmes seront présentées. Le chapitre est essentiellement inspiré de [BAD98].

1.2 Les environnements mobiles

Un environnement mobile est un système composé de sites mobiles. Il permet à ses utilisateurs d'accéder à l'information indépendamment de leurs positions géographiques. Les réseaux mobiles ou sans fil peuvent être classés en deux classes : les réseaux avec infrastructure et les réseaux sans infrastructure.

Le modèle de système intégrant des sites mobiles et qui a tendance à se généraliser, est composé de deux ensembles d'entités distinctes : les «sites fixes» d'un réseau de communication

filaire classique (*wired network*), et les « sites mobiles » (*wireless network*) [IMI 94]. Certains sites fixes, appelés stations support mobile (*Mobile Support Station*) ou station de base (BS) sont munis d'une interface de communication sans fil pour la communication directe avec les sites ou unités mobiles (UM) (Figure 1.1), localisés dans une zone géographique limitée, appelée cellule.

A chaque BS correspond une cellule à partir de laquelle des unités mobiles peuvent émettre et recevoir des messages. Alors que les sites fixes sont interconnectés entre eux à travers un réseau de communication filaire, généralement fiable et d'un débit élevé, les liaisons sans fil ont une bande passante limitée qui réduit sévèrement le volume des informations échangées [DUC92].

Une unité mobile ne peut être, à un instant donné, directement connectée qu'à une seule station de base. Elle peut communiquer avec les autres sites à travers la station à laquelle elle est directement rattachée. L'autonomie réduite de sa source d'énergie lui occasionne de fréquentes déconnexions du réseau, sa reconnexion peut alors se faire dans un environnement nouveau, voire dans une nouvelle localisation.

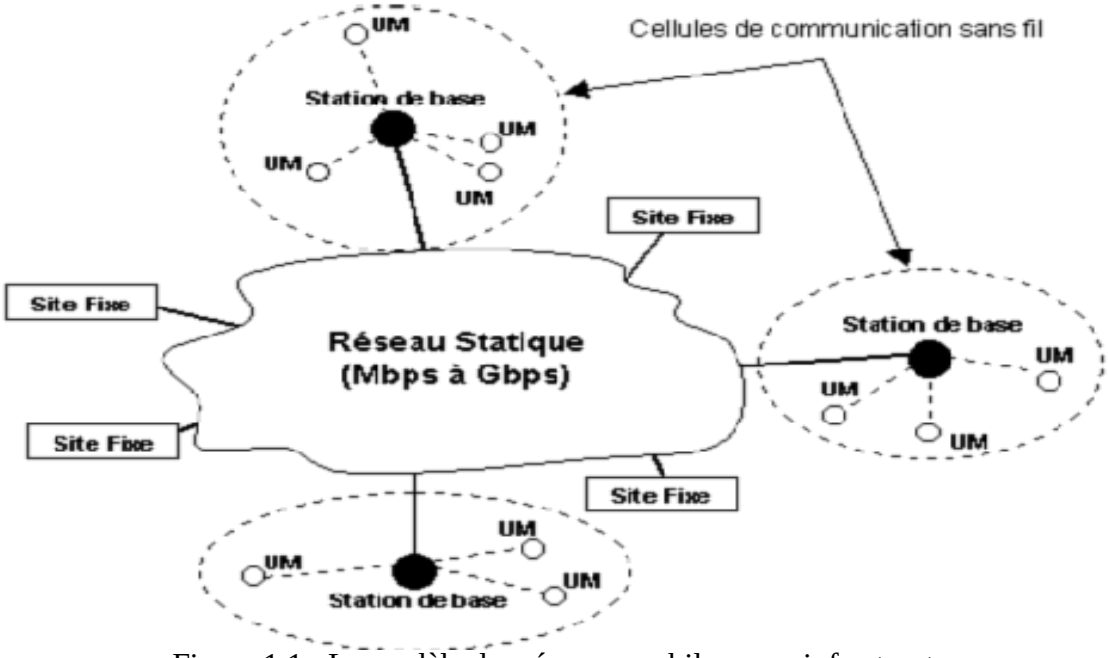


Figure 1.1 : Le modèle des réseaux mobiles avec infrastructure

Le modèle de réseau sans infrastructure préexistante ne comporte pas l'entité "site fixe", tous les sites du réseau sont mobiles et se communiquent d'une manière directe en utilisant leurs interfaces de communication sans fil (Figure 1.2).

L'absence d'infrastructure ou de stations de base, oblige les unités mobiles à se comporter comme des routeurs qui participent à la découverte et la maintenance des chemins pour les autres hôtes du réseau.

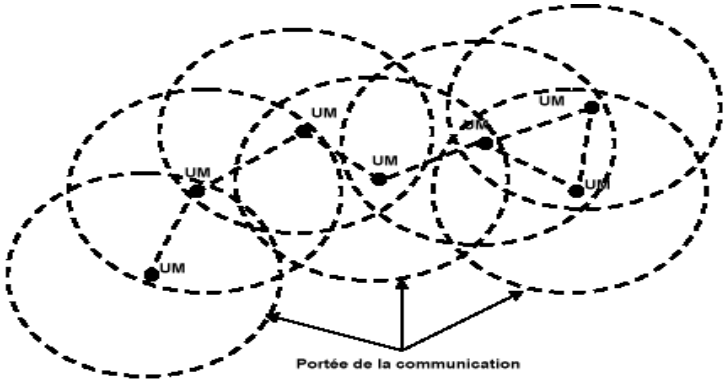


Figure 1.2 : Le modèle des réseaux mobiles sans infrastructure

1.3 Caractéristiques des unités mobiles

Les réseaux informationnels futurs dits PCN (*Personal Communication Network*) intégreront une large variété de services (voix, données, multimédia) offerts aux usagers indépendamment de leur position géographique.

L'architecture générale de ces réseaux, sera construite autour des infrastructures déjà existantes telles que : les réseaux téléphoniques cellulaires, reliés au réseau téléphonique public, les réseaux locaux traditionnels tels que Ethernet, étendus à la communication sans fil, et reliés à des réseaux plus étendus de type LAN, WAN, Internet, et enfin les architectures orientées vers des services spécialisés fournis par diffusion sur des portions d'ondes radio en modulation de fréquence ou par satellite à des usagers munis de terminaux spéciaux.

La même unité mobile peut, en principe, interagir avec les trois types d'infrastructures à différents moments, par exemple, en se déplaçant de l'intérieur d'un bâtiment où elle interagit avec un réseau local pourvu d'une interface de communication sans fil, à l'extérieur du bâtiment où elle interagit avec le réseau téléphonique cellulaire.

Il est à prévoir que l'émergence d'un marché massif du calcul dit mobile, que la plupart des auteurs situent autour de la fin de cette décennie, sera basée essentiellement sur des applications orientées vers des services d'information et de messagerie, et verra le développement de diverses configurations d'unités mobiles plus au moins évoluées[FOR94].

Les configurations existantes sont diverses tels que les ordinateurs de poche (*palmtops*), avec une fréquences d'horloge qui oscille entre 16 et 400 Mhz, une RAM de 2 Moctets à 64 Moctets et une ROM de 4 Moctets à 16 Moctets, sont généralement comparable à celle d'un ordinateur personnel de bureau avec une capacité mémoire de 2 à 8 Moctets et une fréquence d'horloge de 15 à 20 Mhz.

1.4 L'utilisation des ondes radio dans la communication sans fil

La transmission radio utilisée dans la communication sans fil des unités mobiles est basée sur le principe que l'accélération d'un électron crée un champ électromagnétique qui à son tour accélère d'autres électrons et ainsi de suite. Il est alors possible de provoquer le déplacement électromagnétique. Plus le nombre d'électrons déplacés est important, plus le signal est fort et plus sera grande sa portée, avec une vitesse proche de celle de la lumière[WAY93].

Un déplacement coordonné d'électrons peut alors servir pour le transfert d'informations et constitue la base de la communication sans fil. L'approche standard de la transmission radio est le déplacement des électrons à une fréquence donnée. Des techniques de modulation et de multiplexage permettent d'adapter les signaux transmis à la bande passante du support de communication et de rentabiliser son utilisation.

Des signaux sur la même fréquence s'interfèrent et s'altèrent mutuellement. Pour y remédier, le spectre de fréquence est divisé en plusieurs parties (bandes de fréquence), chaque partie est dédiée à une utilisation spécifique. La taille limitée du spectre de fréquence impose donc le regroupement d'usagers dans des bandes étroites comme dans le cas de la radio cellulaire. Au lieu d'allouer à chaque appel la totalité de la fréquence, la technologie cellulaire limite la puissance du signal au minimum nécessaire, ce qui réduit les limites des interférences à une région de taille réduites autour de la station d'émission.

Deux stations d'émission / réception situées dans des régions différentes suffisamment éloignées l'une de l'autre, peuvent utiliser la même fréquence sans risque d'interférence.

1.5 La fiabilité de la communication sans fil

La communication sans fil est moins fiable que la communication dans les réseaux filaires. La propagation du signal subit des perturbations (erreurs de transfert, microcoupure,timeout) dues à l'environnement, qui altère l'information transférée. Il

s'ensuit alors un accroissement du délai de transit de messages à cause de l'augmentation du nombre de retransmissions. La connexion peut aussi être rompue ou altérée par la mobilité des sites.

1.6 La communication cellulaire

La configuration standard d'un système de communication cellulaire est un maillage (*grid*) de cellules hexagonales (Figure 1.3) [HIL95]. Initialement, une région peut être couverte uniquement par une seule cellule. Quand la compétition devient importante pour l'allocation des canaux, la cellule est généralement divisée en sept cellules plus petites, dont le rayon est égal à un tiers du rayon de la cellule de départ. Cette subdivision peut être répétée et l'on parle alors de systèmes micro cellulaires. Les cellules adjacentes dans le maillage doivent utiliser des fréquences différentes, contrairement à celles qui sont situées sur les côtés opposés du maillage et qui peuvent utiliser la même fréquence sans risque d'interférence.

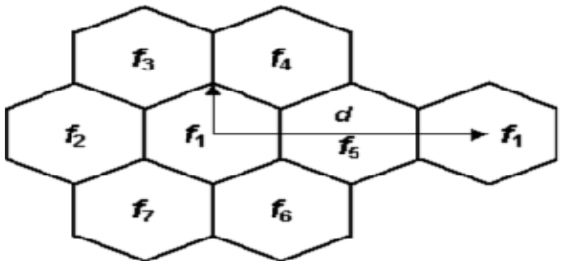


Figure 1.3 : Le principe de réutilisation de fréquence

1.7 Quelques éléments de l'infrastructure sans fil

Les réseaux informationnels de demain dits PCN (*Personal Communication Network*) intégreront une large variété de services (voix, données, multimédia … etc.) offerts aux usagers indépendamment de leur position géographique. L'architecture générale de ces réseaux, bien qu'encore en débat, sera construite autour des infrastructures déjà existantes telles que :

- Les réseaux téléphoniques cellulaires (à l'avenir micro cellulaires) reliés au réseau téléphonique public.
- Les réseaux locaux traditionnels tels que Ethernet, étendus à la communication sans fil, et reliés à des réseaux plus étendus de type LAN, WAN, Internet,…etc.
- Les architectures orientées vers des services spécialisés fournis par diffusion sur des portions d'ondes radio en modulation de fréquence ou par des satellites à des usagers munis de terminaux spéciaux [PIT 93, IMI 94].

La même unité mobile peut, en principe, interagir avec les trois types d'infrastructures à différents moments, par exemple, en se déplaçant de l'intérieur d'un bâtiment où elle interagit avec un réseau local pourvu d'une interface de communication sans fil, à l'extérieur du bâtiment où elle interagit avec le réseau téléphonique cellulaire.

1.8 Conclusion

Ce chapitre a été axé sur le concept des environnements mobiles et leurs utilisations. L'évolution rapide qu'a connu la technologie sans fil récemment, a permis l'apparition de nouveaux systèmes de communication qui offrent plus d'avantages par rapport aux systèmes classiques. Les nouveaux systèmes n'astreignent plus l'usager à une localisation fixe, mais lui permettent une libre mobilité.

La compréhension parfaite de la communication utilisée dans le nouvel environnement, nécessite la compréhension des notions de base de la technologie sans fil comme l'utilisation des ondes radio, la notion de bande passante, la réutilisation des fréquences, le portée d'une unité

mobile … etc. Le but de ce chapitre a été de donner un aperçu général sur cette technologie qui ne cesse de croître.

Chapitre 2 Le protocole TCP

2.1 Introduction

Le développement du protocole TCP a commencé dès le début du projet ARPAnet. Le réseau ARPAnet ne garantissait pas un transfert fiable des données. Ceci signifie que le réseau ne garantissait pas la livraison des paquets aux machines destinatrices. En cas de congestion, le réseau peut jeter des paquets sans informer ni l'émetteur ni le récepteur de ce problème. A cela vient s'ajouter le fait que les routes utilisées dans l'Internet peuvent changer de façon dynamique, d'où la possibilité de dé-séquencement des paquets. Or le réseau Internet a été conçu pour être utilisé par des applications militaires où la fiabilité est primordiale. D'où la nécessité de détection des pertes et la retransmission des paquets perdus.

Le protocole TCP (*Transmission Control Protocol*), défini dans le RFC 793, offre un service de transfert bidirectionnel, fiable, avec contrôle d'erreurs et contrôle de flux. Il a été conçu pour s'implanter dans un ensemble de protocoles multicouches, supportant le fonctionnement des réseaux hétérogènes. TCP suppose uniquement que les couches de communication qui lui sont inférieures lui procurent un service de transmission de paquet simple, dont la qualité n'est pas garantie.

TCP s'intègre dans une architecture multicouches de protocoles, juste au-dessus du protocole Internet IP. Ce dernier permet à TCP l'envoi et la réception de segments de longueur variable, encapsulés dans un paquet Internet appelé aussi "datagramme". Le datagramme Internet dispose des mécanismes permettant l'adressage d'un service TCP source et un destinataire, quelles que soient leurs positions dans le réseau. Le protocole IP s'occupe aussi de la fragmentation et du réassemblage des paquets TCP lors de la traversée de réseaux de plus faibles caractéristiques. Cependant, il ne garantit ni le séquencement ni la livraison des paquets.

2.2 Fonctionnalités de TCP

TCP est conçu pour fournir un service de transmission de données fiables entre deux machines raccordées sur un réseau. Pour pouvoir assurer ce service même au-dessus d'une couche de protocole moins fiable, les fonctionnalités suivantes sont nécessaires: Transfert de données de base, Correction d'erreur, Contrôle de flux, Multiplexage, Gestion de connexions.

▪ **Transfert de données de base :** Le transfert de données de base est la capacité de TCP à transférer un flux continu d'octets dans chaque direction. Les octets sont envoyés entre des processus d'application s'exécutant sur des systèmes distants utilisant TCP. Les processus d'application regroupent ensuite dans un segment de message un ensemble d'octets devant être envoyés / reçus. Les segments de message peuvent être de taille arbitraire. Les messages doivent être envoyés sous forme de datagramme IP, dont la taille est limitée par la taille de l'unité de transmission maximale (MTU) d'une interface de réseau. Néanmoins, au niveau de TCP, il n'existe pas de restriction sur la taille des messages, car la conversion des segments de messages en datagrammes IP ne concernent que la couche IP.

▪ **Contrôle d'erreur :** TCP doit considérer et traiter les cas de données perdues, erronées, dupliquées ou arrivées dans le désordre à l'autre bout de la liaison Internet. Ceci est réalisé par l'insertion d'un numéro de séquence (le numéro de séquence du premier octet des données) et par l'obligation d'émission d'un "accusé de réception" (ACK) par le destinataire TCP.

Si l'accusé de réception n'est pas reçu au bout d'un temps prédéfini, le paquet sera réémis. Côté récepteur, les numéros de séquence sont utilisés pour reconstituer dans le bon ordre le flux original, et éliminer les paquets dupliqués. L'élimination des erreurs physiques de transmission se fait par encodage d'un checksum à l'émission, recalcul de ce checksum par le destinataire, et élimination des paquets pour lesquels les deux valeurs ne correspondent pas. Tant que TCP fonctionne correctement, et que le réseau Internet n'est pas saturé, aucune faute de transmission ne devrait transparaître dans la communication.

- **Contrôle de flux** : Les machines qui émettent et reçoivent les segments de données TCP ne le font pas toutes au même rythme en raison de la diversité des unités centrales et des bandes passantes. Il peut donc arriver que l'émetteur envoie ses données beaucoup plus rapidement que le récepteur ne peut les gérer. TCP implémente donc un mécanisme de contrôle du flux des données. Ce contrôle est obtenu en retournant une information de "fenêtre" avec chaque accusé de réception indiquant la capacité de réception instantanée en termes de numéros de séquence. Ce paramètre noté "*window*" indique le nombre d'octets que l'émetteur peut envoyer avant une autorisation d'émettre ultérieure.

- **Multiplexage** : Pour permettre à plusieurs processus d'une même machine de communiquer simultanément via TCP, le protocole définit un ensemble d'adresses et de ports pour la machine. Un "*socket*" est défini par l'association des adresses Internet source, destinataire, ainsi que les deux adresses de port à chaque bout. Une connexion nécessite la mise en place de deux *sockets*. Un *socket* peut être utilisé par plusieurs connexions distinctes. L'affectation des ports aux processus est établie par chaque ordinateur. Cependant, certains numéros de ports sont réservés pour des services caractérisés et souvent utilisés.

- **Connexions :** Les mécanismes de fiabilisation et de contrôle de flux décrits ci-dessus imposent à TCP l'initialisation et la maintenance de certaines informations pour chaque communication. La combinaison de ces informations, dont les *sockets*, les fenêtres, et les numéros de séquence formeront ce qu'on appelle une connexion. Chaque connexion est identifiée de manière unique par sa paire de *sockets*, définissant chacun des deux sens de la communication. Lorsque deux processus désirent communiquer, leurs TCP respectifs doivent tout d'abord négocier et établir une connexion. Pour identifier une connexion, il faut les données correspondant aux deux extrémités. Il est donc possible de connecter plusieurs processus à la même extrémité distante. TCP peut prendre en charge simultanément plusieurs connexions multiplexées sur la même interface réseau. Il peut transmettre les données sur une connexion dans un sens ou dans l'autre. C'est une connexion full-duplex. Lorsque la communication s'achève, elle sera fermée, en libérant ses ressources à d'autres usages.

2.3 Interfaces

TCP s'interface avec un processus utilisateur ou applicatif et un protocole de niveau inférieur du type *Internet Protocol*.

- Interface TCP vers couche supérieure : L'interface avec les applicatifs consiste en un ensemble de commandes comme le ferait une application à un système d'exploitation pour la manipulation de fichiers. On trouvera des commandes OPEN ou CLOSE pour l'ouverture et la fermeture d'une communication, SEND ou RECEIVE pour l'émission ou la réception de données ou STATUS pour connaître l'état d'une communication.
- Interface TCP vers couche inférieure : TCP effectue des appels vers une couche inférieure du protocole de communication, chargée de "router" les segments. Le réseau Internet s'appuie sur le protocole Internet (IP). L'interface TCP / couche inférieure (IP) dispose de commandes pour

recevoir et émettre des paquets vers des modules TCP où qu'ils soient sur le réseau. Ces appels ont des paramètres tels qu'adresses, type de service, priorité, sécurité, et autres informations de contrôle. Tout protocole sur lequel s'appuiera TCP doit pouvoir fournir l'adresse source, et destination.

2.4 Fonctionnement du protocole TCP

2.4.1 Eléments constitutifs du réseau

L'environnement réseau est constitué de machines raccordées sur des réseaux, eux-mêmes interconnectés par l'intermédiaire de routeurs. Ces réseaux peuvent être des réseaux locaux ou réseaux étendus. Les éléments actifs qui consomment et produisent des paquets sont appelés des processus. Un certain nombre de niveaux de protocoles réseau, au niveau des machines ou des routeurs, permettent d'établir une chaîne complète de communication entre les processus actifs de n'importe quelle machine.

Le terme paquet désigne les données d'une transaction unitaire entre un processus et le réseau. Les "hôtes" sont des ordinateurs raccordés au réseau, et, du point de vue de ce dernier, sont les sources et destinations des paquets. Les processus sont identifiés comme les éléments actifs dans ces machines. Les terminaux, fichiers, et autres périphériques d'entrée/sortie peuvent être identifiés comme "éléments communicants" par l'intermédiaire d'un processus. De ce fait, toute communication reste identifiée comme un échange inter-processus.

2.4.2 Modèle de fonctionnement

Les processus transmettent les données en faisant appel à TCP et en passant des tampons de données comme arguments. TCP met en forme les données de ces tampons dans des segments afin de les transférer au protocole Internet qui a son tour les acheminera vers le TCP distant. Celui-ci reçoit les segments, les copie dans un tampon temporaire, et en avise l'émetteur. Le protocole TCP inclut les informations nécessaires à la "reconstruction" en bon ordre des données originales.

Le modèle d'une communication Internet fait qu'il existe pour chaque TCP actif un module de protocole Internet chargé de l'acheminement de données. Ce module Internet "encapsule" à son tour les paquets TCP sous la forme de paquets Internet et les transmet au module Internet distant via des "routeurs". Pour transmettre le paquet ainsi constitué à travers un réseau local, une troisième couche de protocole, spécifique au réseau, est ajoutée. Les paquets peuvent alors subir un grand nombre de transformations, fragmentations ou autres opérations pendant leur acheminement au module Internet distant.

A l'arrivée dans un routeur, le paquet Internet est "désossé", puis ses informations sont examinées pour savoir vers quel réseau le paquet doit être acheminé. Un nouveau paquet Internet est constitué, selon les spécifications du segment du réseau désigné, puis transmis sur ce réseau.

Un routeur peut procéder à une segmentation plus fine des paquets, si le réseau en sortie n'a pas les performances suffisantes pour véhiculer le paquet d'origine. Pour réaliser ceci, le routeur exécute une nouvelle segmentation en "fragments". Ces mêmes fragments peuvent à leur tour être redécoupés en chemin par un autre routeur. Le format de paquets fragmentés est standardisé de sorte que le réassemblage soit toujours possible, étape par étape, jusqu'à restitution des données originales. Le module Internet d'arrivée extrait le datagramme du niveau supérieur et le passe à la couche TCP.

2.4.3 Format de l'en-tête

Les paquets TCP sont envoyés sous forme de datagrammes Internet. L'en-tête IP transmet un certain nombre de paramètres, tels que les adresses Internet source et destinataire. L'en-tête TCP est placé à la suite, contenant les informations spécifiques au protocole TCP. Cette division

permet l'utilisation de protocoles autres que TCP, au-dessus de la couche IP.(Figure 2.1). Le format des paquets TCP est représenté dans l'annexe 2.

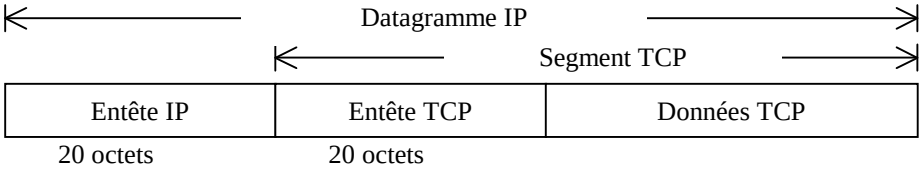


Figure 2.1 : Encapsulation de données TCP dans un datagramme IP

2.4.4 Etablissement et libération de connexion TCP

Trois phases sont nécessaires pour le transfert de données en utilisant le protocole TCP : ouverture de la connexion, transfert de données et fermeture de la connexion. Dans cette section on introduira les mécanismes d'établissement et de libération de connexion TCP. On appellera *client* l'entité TCP qui demande l'ouverture d'une connexion et *serveur* l'entité TCP qui répond par une acceptation ou un refus de la connexion. De la même façon on appellera *application client* un processus applicatif qui demande au fournisseur du service transport « TCP » d'ouvrir une connexion active et *application serveur* un processus applicatif qui demande l'ouverture d'une connexion passive.

Suite à une demande d'ouverture de *connexion active*, TCP commence la phase d'établissement de connexion avec le serveur. Une demande d'ouverture de *connexion passive* signifie que l'application serveur demande l'acceptation des connexions entrantes. Avant chaque ouverture de connexion le processus serveur doit demander l'ouverture d'une connexion passive. Quand une connexion est ouverte, le processus serveur reçoit une indication l'informant de l'ouverture de la connexion. Lors de la demande d'ouverture d'une connexion active, le processus client doit spécifier à la couche TCP l'adresse IP ainsi que le numéro de port identifiant de façon unique le processus serveur. Le client envoie un segment TCP de type SYN (voir Annexe 2) avec un numéro de séquence égal au numéro de séquence initial (ISN : *Initial Sequence Number*). L'ISN est un compteur sur 32 bits qui doit être incrémenté toutes les 4 microsecondes (L'intérêt de l'ISN est de prévenir contre le risque de considérer un segment retardé dans le réseau et appartenant à une connexion antérieure comme un segment d'une connexion en cours ayant les mêmes numéros de ports que la connexion antérieure).

Le serveur envoie un segment SYN en réponse à la demande de connexion. Ce segment SYN contient un acquittement égal à l'ISN du client + 1. A la réception du segment SYN, le client envoie, à son tour, un acquittement du segment SYN du serveur en acquittant l'ISN du serveur + 1 et informe le processus client de l'ouverture réussie de la connexion.

De la même façon, en recevant l'acquittement du segment SYN, le serveur informe le processus serveur de l'établissement de la connexion. La Figure 2.2 montre un exemple d'établissement de connexion réussie. Plus de détails sur l'établissement de connexions TCP peuvent se trouver dans [POS81].

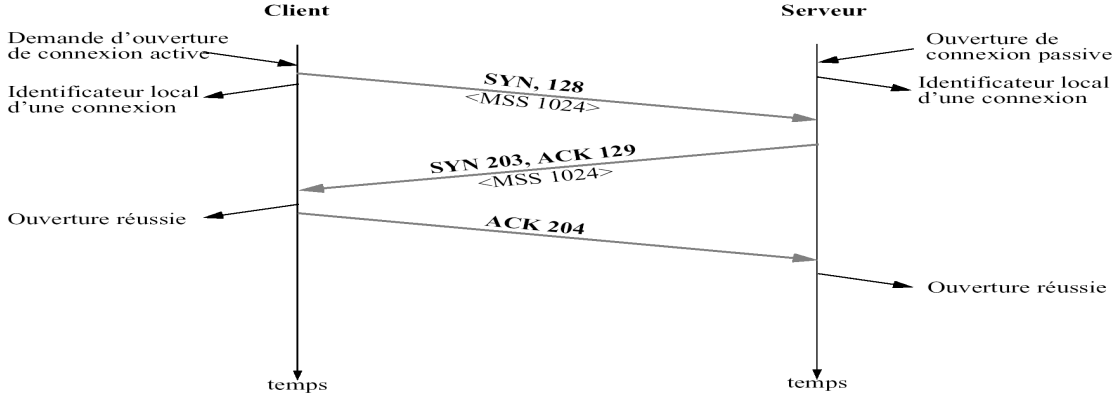


Figure 2.2 : Exemple d'établissement de connexion TCP réussie

A la différence de l'étape d'établissement de connexion TCP qui se fait en trois phases ("*three-way handshake*"), la fermeture de la connexion TCP se fait en quatre (Figure 2.3).

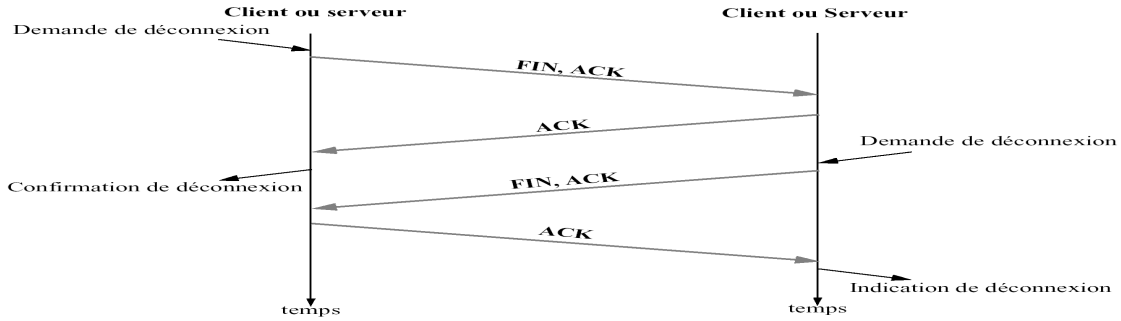


Figure 2.3 : Exemple de déconnexion réussie

Comme une connexion TCP est une connexion bidirectionnelle, le processus client et serveur doivent demander la fermeture de la connexion de façon individuelle. Quand un des deux processus, client ou serveur, demande la fermeture de la connexion, le processus homologue n'ayant pas encore demandé la fermeture de la connexion, peut continuer à envoyer des données.

Suite à une demande de fermeture de connexion, TCP envoie un segment de type FIN et une confirmation de déconnexion est transmise au processus ayant demandé la fermeture de la connexion. Le segment FIN est acquitté par l'entité TCP homologue. Cette dernière rentre dans un état dit semi-fermé ("*half-close*") et peut continuer à envoyer normalement des données. Après avoir fini cette procédure d'envoi, un segment FIN est envoyé. La connexion est fermée à la réception de l'acquittement du deuxième segment FIN. Une indication de déconnexion est alors transmise au second processus.

2.4.5 Echange des données TCP

L'une des particularités du protocole TCP est l'utilisation des numéros d'octets et non pas des numéros de paquets pour gérer les acquittements et les retransmissions. Le champ *séquence* indique la position du premier octet du paquet dans le flux de données alors que le champ *acquittement* indique le prochain numéro de l'octet attendu par le récepteur(voir annexe 2). TCP effectue le transfert d'un flux continu de données dans les deux directions en regroupant à chaque fois un certain nombre d'octets pour former des segments. En général, TCP décide quand il doit bloquer et envoyer des données en utilisant un contrôle de flux connu sous l'appellation *fenêtre glissante.* Ce contrôle de flux permet d'envoyer plusieurs paquets avant de se bloquer en attente d'acquittements ("*Stop and Wait*"). Ceci permet d'améliorer le débit puisque l'émetteur n'est pas obligé d'attendre la réception d'acquittement après chaque envoi de paquet de données. Parfois les applications (utilisant le service TCP) doivent être sûres que toutes les données qu'elles ont soumises ont été transmises. Pour ce faire, l'interface de programmation offre le moyen aux applications de positionner le bit PSH dans un paquet de données.

TCP fournit les moyens au récepteur d'asservir la quantité de données envoyées par l'émetteur. Ceci est réalisé par le biais du champ fenêtre qui indique un intervalle des numéros de séquences qui peuvent être acceptés par le récepteur.

Afin d'assurer la fiabilité, TCP effectue une retransmission des paquets perdus. Pour détecter ces pertes, un temporisateur est armé à chaque envoi d'un paquet. A la réception d'un segment qui acquitte la réception du paquet envoyé le temporisateur est désactivé. Par contre si l'émetteur ne reçoit aucun acquittement avant l'expiration du temporisateur, le paquet en question est retransmis. La Figure 2.4 représente un exemple d'échange de données utilisant TCP. Sur le diagramme les champs : séquence, acquittement, fenêtre, drapeaux, taille du segment et options TCP sont représentés. Les segments 1, 2, 3 correspondent à l'établissement

de la connexion TCP en 3 phases. La connexion est établie à l'initiative de la machine A qui annonce, dans le champ options (voir Annexe2), un MSS "*Maximum Segment Size*" de 1460 et une fenêtre de 4096 octets (segment 1).

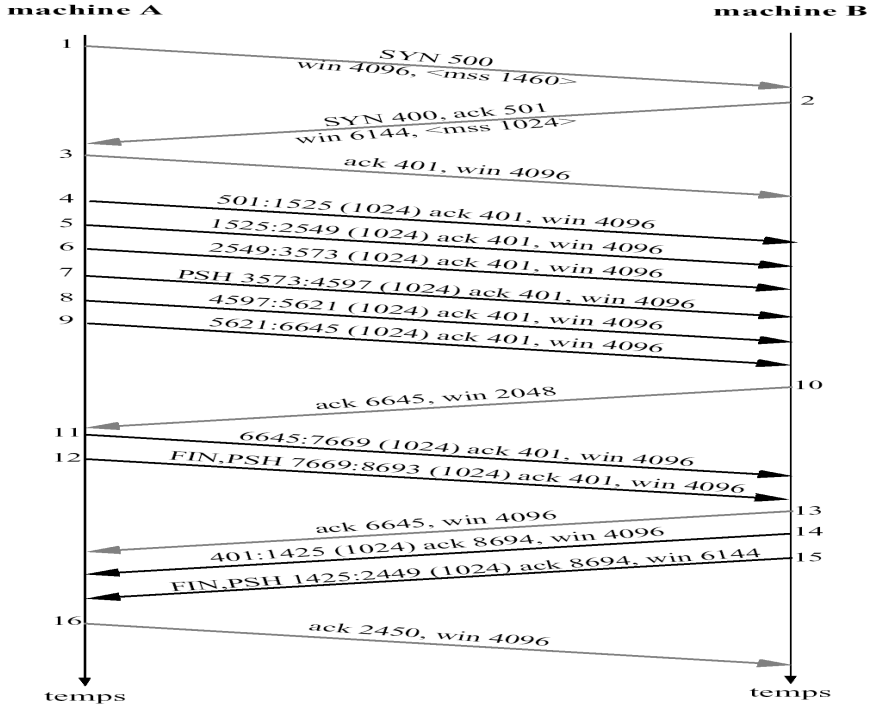


Figure 2.4 : Exemple d'échange de données

La machine B accepte la connexion en envoyant le segment 2 qui annonce un MSS de 1024 octets et une fenêtre de 6144. L'échange de données se fait en utilisant des segments de taille maximale égale à 1024 octets. Ensuite la machine A envoie 6 segments de taille 1024 octets (segment 4, 5, 6, 7, 8, 9) avant de se bloquer en attente d'un acquittement, à cause du mécanisme de contrôle de flux imposé par le récepteur. En effet, la fenêtre qui a été annoncée par la machine B est de 6144 ce qui autorise l'envoie simultané d'un maximum de 6 segments de taille 1024 octets. Peu de temps après, la machine B acquitte les 6 segments (i.e. 4, 5, 6, 7, 8, 9) en envoyant le segment 10.

TCP utilise le mécanisme d'*acquittement cumulatif* qui permet d'acquitter plusieurs segments de données par un seul acquittement. La machine A peut maintenant envoyer des données. Cette dernière envoie les segments 11 et 12 en activant le drapeau FIN du segment 12. Ceci signifie que la machine A n'a plus de données à envoyer, mais peut continuer à recevoir des données.

La machine B envoie le segment 13 ayant comme acquittement le même que le segment 10 mais annonçant une fenêtre plus grande (4096 octets). La machine B envoie les segments 14 et 15 et demande à son tour la fermeture de la connexion en activant le drapeau FIN du segment 15. Finalement la machine A acquitte les deux segments de données 14 et 15 dans le segment 16. La connexion est alors fermée.

La Figure 2.5 représente un exemple de transfert de données avec retransmission. Après l'établissement de la connexion, la machine A envoie les segments 1, 2 et 3 et arme un temporisateur. Le paquet 1 est perdu dans le réseau. A la réception de chacun des paquets 2 et 3, la machine B envoie un acquittement avec un numéro d'acquittement égal au numéro de séquence du segment 1. L'utilité de ces acquittements est, d'une part, d'informer l'émetteur (machine A) que le segment 1 (séquence 501) n'est toujours pas reçu, et d'autre part d'annoncer la nouvelle taille de fenêtre. A l'expiration du temporisateur, la machine A retransmet le segment ayant le numéro de séquence 501. La machine B délivre les 3 segments de données et acquitte

simultanément les 3 paquets de données par le segment 7. On note que la fenêtre annoncée dans le segment 7 est de 6144.

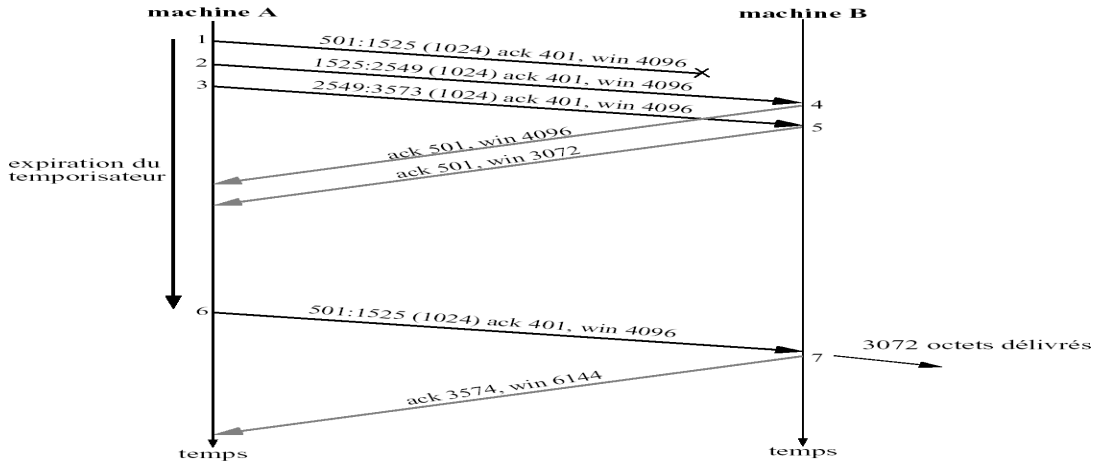


Figure 2.5 : Exemple de retransmission de paquet perdu

2.5 Gestion des temporisations

Un des paramètres qui affecte les performances de TCP est le temporisateur de retransmission « timeout ». Si ce paramètre est sous-estimé TCP peut retransmettre inutilement des segments déjà reçus et si le paramètre est surestimé une perte sera détectée tardivement résultant en un temps d'inactivité de TCP assez important. Comme le protocole TCP n'a aucune connaissance des caractéristiques du réseau physique et de la charge de ce dernier, il doit ajuster dynamiquement son temporisateur de retransmission. Le RFC 793, spécifie une méthode d'estimation du timeout.

L'idée consiste à calculer une moyenne, du type EWMA ("*Exponential Weighted Moving Average*"), des RTT (temps d'aller retour) mesurés et de recalculer une nouvelle valeur du timeout à chaque changement de cette moyenne. Pour ce faire, chaque connexion TCP définit les deux variables suivantes :

• *SampleRTT* : un échantillon du RTT. A chaque émission de segment, TCP enregistre l'instant de son émission. A la réception de l'acquittement correspondant, *SampleRTT* est calculé (différence entre l'instant de réception de l'acquittement et l'instant d'émission du segment).

• *EstimatedRTT* : moyenne pondérée des *SampleRTT*. La valeur du timeout est alors donnée par l'expression suivante :Timeout = 2 * *EstimatedRTT*.

En 1987, Karn et Partridge ont introduit deux améliorations sur la gestion des temporisateurs [KAR87]:

• Le backoff exponentiel : Consiste à initialiser la valeur du timeout au double de sa valeur précédente à chaque fois qu'un paquet est retransmis.

• Estimation du RTT : Le SampleRTT ne doit pas être estimé en cas de retransmission. En effet quand un segment est retransmis et tout de suite après un acquittement est reçu, il est impossible de déterminer si cet acquittement peut être associé à la première ou à la seconde transmission du segment. D'où la nécessité de ne pas estimer le RTT pendant les retransmissions.

Toutes les implémentations actuelles de TCP (Tahoe, Reno, NewReno et SACK) incluent le backoff exponentiel proposé par Karn et Partridge et l'estimation du timeout proposée par Jacobson.

2.6 Gestion de la fenêtre de transmission

La fenêtre transmise dans chaque segment indique la plage de numéros de séquence que l'émetteur de la fenêtre (celui qui reçoit les données) est prêt à accepter. La taille de cette fenêtre est en relation avec la taille disponible des tampons de données associés à cette connexion.

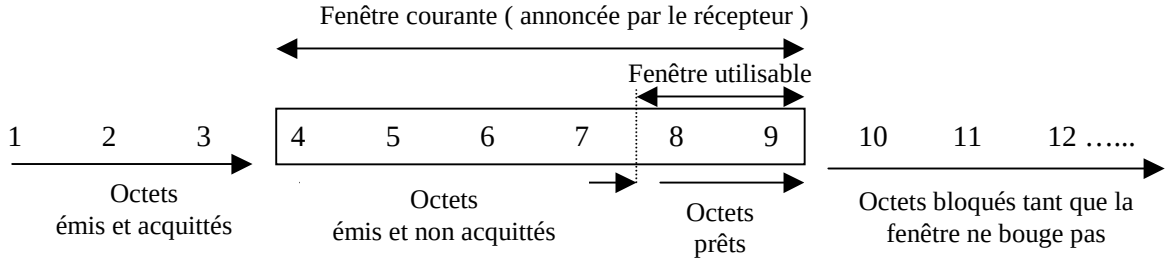


Figure 2.6 : Visualisation de la fenêtre glissante de TCP

La Figure 2.6 numérote les octets de 1 à 12. La fenêtre annoncée par le récepteur est appelée fenêtre annoncée ou fenêtre offerte (*offered window*) et couvre les octets 4 à 9, signifiant que le récepteur a acquitté tous les octets jusqu'à et incluant le numéro 3, et a annoncé une taille de fenêtre de 6. L'émetteur calcule sa fenêtre utilisable, égale à la quantité de données qu'elle peut envoyer immédiatement. Au cours du temps cette fenêtre glissante se déplace vers la droite, à mesure que le récepteur acquitte des données. Le mouvement relatif des deux extrémités de la fenêtre accroît ou décroît la taille de la fenêtre.

Une grande taille de fenêtre encourage l'émission. Si le nombre de données reçues est supérieur à ce que la fenêtre indique, les données hors fenêtre seront rejetées. Cette déperdition entraîne un grand nombre de retransmissions et surcharge inutilement le réseau.

L'usage d'une petite taille de fenêtre morcelle le débit en ajoutant un certain retard supplémentaire au "temps de boucle", mais en limitant la surcharge du réseau due aux retransmissions. La gestion de la largeur de fenêtre a une influence importante sur les performances de la communication.

2.7 Contrôle de trafic et de congestion

La congestion peut se produire lorsque les données arrivent sur un gros tuyau (un LAN rapide) et sont transmises sur un tuyau plus petit (un WAN plus lent). La congestion peut également se produire lorsque de multiples flux d'entrée parviennent à un routeur dont la capacité de sortie est inférieure à la somme des entrées.

Dans cette section on introduit les algorithmes mis en oeuvre pour prévenir et contrôler la congestion dans les réseaux TCP/IP. Le protocole TCP, responsable du contrôle de trafic, est devenu stable au début des années 80 et mettait en oeuvre un algorithme de contrôle de flux basé sur l'utilisation du champ fenêtre du protocole.

Ce contrôle de flux permettait d'éviter la perte des paquets au niveau du récepteur. Cependant TCP n'utilisait aucun mécanisme de contrôle de trafic permettant de prévenir les congestions dans le réseau. L'Internet souffrait d'un phénomène de congestion connu sous l'appellation *congestion collapse* [NAG84]. Ce phénomène est le résultat de transmission des paquets, au rythme des fenêtres de contrôle de flux. Ceci donne lieu à un état de congestion et à des retransmissions très souvent inutiles, suite à l'expiration du temporisateur de retransmission. En 1988 Van Jacobson a proposé un ensemble d'algorithmes de contrôle de trafic pour TCP permettant de prévenir les congestions [JAC88].

La version de TCP qui intègre l'ensemble des modifications proposées est connue sous l'appellation TCP *Tahoe* et constitue sans aucun doute l'une des évolutions majeures dans le domaine du contrôle de trafic du monde Internet.

2.7.1 Algorithmes du démarrage lent et d'évitement de congestion

Dans ce paragraphe on introduit le principe de la fenêtre de congestion (introduit par Van Jacobson) [JAC88]. L'algorithme de contrôle de trafic est du type *basé sur fenêtre* "*window based*" et utilise une *notification "feedback" implicite* de l'état de congestion dans le réseau (la

source ajuste son trafic en fonction de changements de l'état du réseau : perte de paquet). Pour ce faire TCP maintient une nouvelle variable : fenêtre de congestion "*CongestionWindow*" appelée *cwnd*, qui indique la taille de la fenêtre de congestion.

La fenêtre de congestion représente la quantité maximale de données en transit dans le réseau. TCP augmente progressivement (au rythme des RTT) la valeur de cette fenêtre jusqu'à la détection d'une perte. A ce point, la source réduit la taille de la fenêtre de congestion et recommence l'augmentation progressivement. Cet aspect dynamique de la fenêtre est connu dans la littérature par le terme : AIMD ("*Additive Increase/Multiplicative Decrease*"). La Figure 2.7 montre l'évolution de la fenêtre de congestion TCP en fonction du temps d'aller retour : RTT ("*Round Trip Time*). La croissance de la fenêtre de congestion se fait en deux phases : démarrage lent "*Slow-start"* et évitement de congestion "*congestion avoidance"* [JAC88], [STE97].

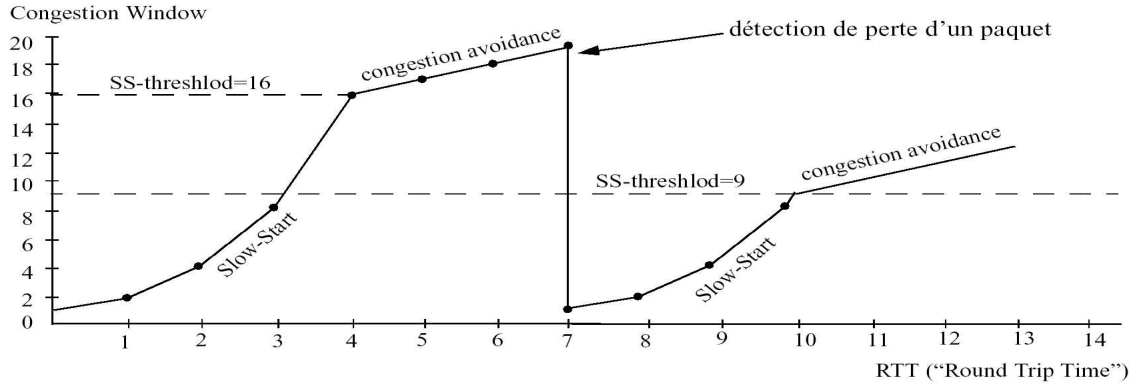


Figure 2.7 : Evolution de la fenêtre de congestion de TCP Tahoe

- **Démarrage lent "Slow Start"**

Après l'établissement de la connexion ou après l'expiration d'un temporisateur de retransmission, l'émetteur fixe la taille de la fenêtre de congestion à 1 segment (MSS). A chaque réception d'acquittement, il augmente la taille de la fenêtre de congestion par 1 MSS.

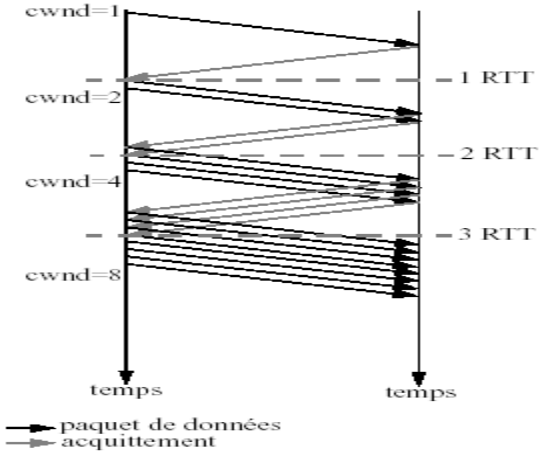


Figure 2.8 : Evolution de la fenêtre de congestion pendant la phase du démarrage lent

Cet algorithme continue jusqu'à ce que la fenêtre de congestion atteigne un seuil (*SS-threshold*). Ceci résulte en un accroissement exponentiel de la fenêtre de congestion : si chaque paquet est acquitté, la fenêtre de congestion double de taille après chaque RTT.

A ce stade les limites d'Internet peuvent être atteintes, et un routeur intermédiaire commencera à rejeter les paquets. Ceci prévient l'émetteur que la fenêtre de congestion est devenue trop large. Le terme démarrage lent peut induire une confusion puisque la fenêtre croit de façon exponentielle et permet de remplir, assez rapidement, le lien.

- **Evitement de congestion "*Congestion avoidance*"**

Après la phase du démarrage lent, qui prend fin au franchissement du seuil *SS-threshold*, la fenêtre évolue de façon linéaire : pour chaque acquittement reçu la taille de la fenêtre augmente de MSS * MSS / *CongestionWindow*. Ceci revient à incrémenter la fenêtre par un MSS à chaque fois qu'une fenêtre de congestion entière est acquittée et résulte en un accroissement linéaire.

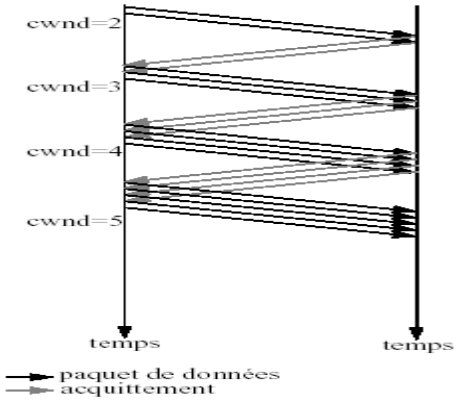


Figure 2.9 : Evolution de la fenêtre de congestion pendant la phase d'évitement de congestion

Dans cette phase la croissance est linéaire plutôt qu'exponentielle pour éviter de revenir trop rapidement dans une phase de congestion. Cette phase continue jusqu'à la détection de perte de paquet (la détection de perte peut se faire de deux manières : soit par l'expiration de temporisateur, soit à la réception d'acquittements dupliqués. A ce point, TCP met à jour la valeur du seuil, *SS-threshold*, à la moitié de la fenêtre de congestion (Figure 2.7).

Quand la fenêtre de congestion atteint son seuil maximum, donné par la taille maximale de la fenêtre du récepteur, sans qu'il y ait des pertes, la fenêtre de congestion ne croit plus et reste constante jusqu'à la détection de la prochaine perte.

2.7.2 Algorithmes de retransmission et de recouvrement rapide

Dans ce paragraphe, deux autres algorithmes mis en oeuvre dans TCP sont décrits : retransmission rapide et recouvrement rapide.

- **Retransmission rapide**

Les algorithmes décrits ci-dessus constituent la brique de base pour le contrôle de trafic dans TCP. Ces algorithmes réagissent aux pertes de paquets, considérés comme un *feedback* implicite, pour ajuster la taille de la fenêtre de congestion et par conséquent le trafic de l'émetteur. Dans tout système de contrôle il est très important de recevoir une indication au plus vite afin de refléter au maximum l'état du système et réagir rapidement aux changements.

La détection de pertes de paquets TCP se faisait en utilisant le temporisateur de retransmission. Or, à cause de contraintes matérielles et logicielles, plusieurs systèmes d'exploitation n'offrent qu'une très grande granularité pour les retransmissions (de l'ordre de 300 à 500 ms). Si le RTT est relativement faible par rapport au temporisateur de retransmissions (souvent le cas pour les réseaux à haut débit), toute perte de paquet engendre la dégradation de la performance de TCP. Ceci est essentiellement dû au temps d'inactivité élevé dû à l'attente de l'expiration du temporisateur de retransmission.

Pour avoir une indication plus rapide de la perte de paquets, Jacobson a proposé l'utilisation d'acquittements dupliqués [JAC88]. L'idée des acquittements dupliqués est très simple : à chaque fois qu'un segment est reçu par le récepteur, ce dernier doit répondre par l'envoi d'un acquittement même si d'autres acquittements avec le même numéro d'acquittement ont été envoyés. En d'autres termes si le numéro de séquence du segment reçu n'est pas celui attendu, TCP ne peut pas l'acquitter du fait qu'un ou plusieurs autres segments sont attendus. TCP envoie alors le même acquittement envoyé auparavant : d'où l'appellation acquittement

dupliqué. A la réception d'un certain nombre d'acquittements dupliqués, l'émetteur TCP devine qu'il y a eu une perte de segments. Au lieu d'attendre l'expiration du temporisateur de retransmission, ce dernier retransmet le paquet perdu : d'où l'appellation retransmission rapide ("*fast retransmit*"). Comme il y a un risque de déséquencement dans l'Internet et pour éviter de retransmettre inutilement des paquets, Jacobson a proposé de retransmettre le paquet estimé perdu après la réception de trois acquittements dupliqués.

Après la phase de retransmission rapide, TCP initialise le seuil *SS-threshold* à la moitié de la valeur de la fenêtre de congestion courante et rentre dans la phase du démarrage lent en initialisant la fenêtre à 1 segment de façon similaire à une retransmission suite à l'expiration d'un timeout. La Figure 2.10 présente un exemple de retransmission rapide. La machine A envoie 5 segments de 1024 octets (1 à 5) de suite.

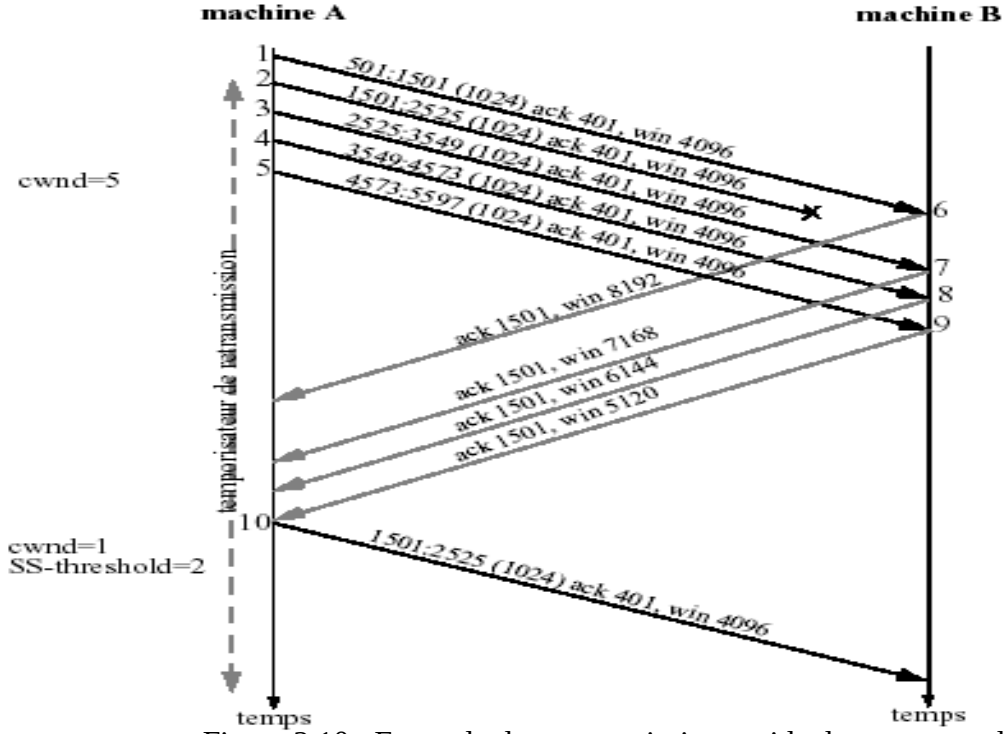


Figure 2.10 : Exemple de retransmission rapide de paquet perdu

A la réception du segment 1 la machine B envoie un acquittement ayant comme numéro d'acquittement 1501 : informant la machine A que le prochain numéro de séquence attendu est 1501. Le paquet 2 (ayant comme n° de séquence 1501) est détruit dans le réseau. A la réception des segments 3, 4 et 5 la machine B les stocke dans une file de re-séquencement et génère pour chaque segment un acquittement toujours avec le numéro d'acquittement 1501. A la réception, par la machine A, de 4 acquittements (6 à 9) ayant comme numéro d'acquittement 1501, cette dernière retransmet le segment 9 perdu (n° de séquence 1501) sans attendre l'expiration du temporisateur de retransmission.

- **Recouvrement rapide**

L'algorithme de recouvrement rapide est une optimisation de la retransmission rapide. Ce dernier, suite à une détection de perte de paquets initialise la fenêtre de congestion à 1 et rentre dans la phase démarrage lent souvent susceptible de baisser la performance de TCP. Le recouvrement rapide est un algorithme qui s'exécute juste après la retransmission rapide du paquet perdu et peut être résumé en deux phases :

• initialiser la fenêtre de congestion ainsi que le *SS-threshold* à la moitié de la fenêtre de congestion (mais pas moins de 2 segments) à la réception du 3 ème acquittement dupliqué.
(ssthresh = cwnd /2 , cwnd = ssthresh)

▪ Incrémenter cwnd de 3 MSS (nombre de segments qui ont été transmis sur le réseau et bufferisés par le récepteur): cwnd = ssthresh + 3 * MSS

• pour chaque acquittement dupliqué reçu envoyer un nouveau segment (ceci permet de maintenir constant le nombre de paquets dans le réseau en envoyant un nouveau paquet chaque fois qu'un acquittement est reçu), car étant donné que les acquittements dupliqués ne permettent pas d'acquitter les paquets envoyés et qui se trouvent dans le buffer réception, il faut augmenter cwnd pour éviter que l'émetteur arrive à une situation où il ne peut plus transmettre de paquets et donc il y aurait une chute de débit)

• à la réception d'un acquittement du segment retransmis, réduire cwnd à la valeur de ssthresh (précédemment calculée). Quitter la phase de recouvrement rapide et rentrer dans la phase d'évitement de congestion.

2.8 Les différentes versions de TCP

▪ **TCP Tahoe** (1988) est la version standard,

- Démarrage lent + évitement de congestion + Retransmission rapide.
- Si expiration de RTO lancer la procédure du *go-back-N* avec démarrage lent (*go-back-N* est une technique de retransmission : après la détection d'une perte, l'émetteur retransmet non seulement le segment perdu mais aussi des segments qui le succèdent).
- Retransmission rapide après trois DUP ACK et *go-back-N* avec démarrage lent.

▪ **TCP Reno** est une amélioration de TCP proposée, aussi, par Jacobson en 1990. La seule amélioration apportée par rapport à TCP Tahoe est l'ajout de l'algorithme de recouvrement rapide. Cet algorithme permet, plutôt que de rentrer dans la phase démarrage lent dans laquelle TCP perd considérablement ses performances suite à une perte de paquet, de transmettre des nouveaux paquets juste après la phase de recouvrement rapide ensuite passer directement à la phase d'évitement de congestion après avoir reçu un acquittement du paquet perdu.

- Ajout du recouvrement rapide au Tahoe.
- Si expiration de RTO *go-back-N* avec démarrage lent.
- Si entré dans la retransmission rapide ne pas revenir au démarrage lent mais entrer dans le recouvrement rapide cwnd=cwnd/2 +3*MSS . Chaque DUP ACK reçu provoque la transmission de nouvelles données.
- La sortie du recouvrement rapide se fait lors de la réception d'un ACK pour de nouvelles données que celles émises par la retransmission rapide(recovery ACK) et cwnd=cwnd/2.

▪ **TCP Vegas** : La version de TCP Vegas change le procédé par lequel elle fait varier les tailles des fenêtres par rapport aux autres versions de TCP. Son principe est d'évaluer la taille des buffers en entrée des routeurs et d'en observer leur évolution sur base d'informations calculées à partir des mesures de RTT. Un algorithme fait varier la fenêtre à partir d'une comparaison du taux de transmission attendu par rapport au taux de transmission en cours. Ce n'est donc plus un système régulé uniquement par la réception des accusés de réception.
Vegas calcule le taux de transmission attendu par : Taille de la fenêtre en cours / RTT où le RTT est calculé sur base d'un segment envoyé dans le réseau sans qu'il y ait congestion. Pour obtenir le taux actuel de transmission, il faut compter le nombre d'informations envoyées sur le réseau entre le temps pour lequel un paquet est émis et le temps pour lequel son accusé de réception est reçu. Vegas ajuste son taux de transmission pour le garder inférieur au taux de transmission attendu. Ce taux de transmission, Vegas l'ajustera en proportion avec le délai de propagation et avec le délai dû aux files d'attente.

- **TCP NewReno** inclut des modifications mineures par rapport à TCP Reno. La modification consiste à éviter des timeouts quand plusieurs segments TCP, appartenant à la même fenêtre de congestion, sont perdus. Le changement concerne le comportement de l'émetteur pendant la phase de recouvrement rapide quand un acquittement partiel est reçu. Un acquittement partiel est défini comme étant un acquittement qui acquitte une partie mais pas la totalité des segments déjà transmis au moment du passage à la phase de recouvrement rapide.

Dans Reno, les acquittements partiels permettent de sortir de la phase de recouvrement rapide et de rentrer dans la phase d'évitement de congestion. Dans NewReno, les acquittements partiels ne permettent pas de sortir de la phase de recouvrement rapide. Les acquittements partiels constituent une indication de la perte du segment succédant le paquet acquitté (par l'acquittement partiel). Par conséquent, chaque acquittement partiel déclenche la retransmission d'un paquet perdu. Ainsi si plusieurs segments appartenant à une seule fenêtre sont perdus, NewReno permet la retransmission d'un paquet par RTT et évite les timeouts. La phase de recouvrement rapide continue jusqu'à ce que tous les paquets transmis avant de rentrer dans cette phase soient acquittés.

L'algorithme peut être résumé de la façon suivante:

1.Quand le 3 ème ACK dupliqué est reçu et l'émetteur n'est pas déjà entrain d'exécuter la procédure du recouvrement rapide faire: ssthresh = cwnd /2 enregistrer le numéro de séquence le plus élevé qui a été transmis dans une variable "*recover*".

2. Retransmettre le segment perdu et affecter à cwnd la valeur de ssthresh plus 3*MSS (nombre de segments qui ont été transmis sur le réseau et bufferisés par le récepteur): cwnd = ssthresh + 3 * MSS

3.Pour chaque acquittement dupliqué reçu incrémenter cwnd de 1 MSS et envoyer un nouveau segment.

4. À la réception d'un acquittement de nouvelles données (cet ACK peut être une réponse à la retransmission de l'étape 2, ou à une retransmission plus récente), si cet ACK acquitte le segment de numéro de séquence *recover* alors il est considéré comme l'acquittement de tout les segments envoyés depuis la transmission du paquet perdu et jusqu'à la réception du 3 ème ACK dupliqué. Si l'ACK reçu est de valeur inférieur à *recover* (il est appelé acquittement partiel), c'est une indication de perte de paquet, alors on retransmet le premier segment non acquitté sans attendre trois acquittements dupliqués et on ne réduit pas ssthresh.

5.La phase de recouvrement rapide continue jusqu'à ce que tout les paquets transmis avant de rentrer dans cette phase soient acquittés

- **TCP SACK :** TCP Tahoe et TCP Reno utilisent une technique de retransmission connue dans la littérature par l'appellation *Go-back-n*. Dans *Go-back-n* après la détection d'une perte, l'émetteur retransmet non seulement le segment perdu mais aussi des segments qui le succèdent. L'inconvénient d'une telle stratégie est que les paquets retransmis peuvent être déjà reçus par le récepteur. Pour remédier à ce problème il est possible d'utiliser les acquittements sélectifs. Le principe des acquittements sélectifs est simple et consiste à reporter dans les acquittements les blocs contigus de segments reçus par le récepteur. L'émetteur retransmet alors les segments perdus.

L'en-tête du protocole TCP ne prévoit pas l'utilisation d'acquittements sélectifs. Pour permettre des acquittements sélectifs, il a fallu définir une nouvelle option TCP permettant d'informer l'émetteur des blocs contigus de données en possession du récepteur.

L'utilisation des acquittements sélectifs améliore considérablement les performances quand le réseau présente un taux de pertes, au niveau physique, élevé (exp. liaison satellite, réseaux sans fil...). L'utilisation de l'option SACK est négociée lors de l'établissement de la connexion. Ceci permet de garder une compatibilité avec les versions antérieures de TCP.

Nom	Caractéristiques
Tahoe	Démarrage lent + Evitement de congestion + Retransmission rapide
Reno	Tahoe + Recouvrement rapide
NewReno	Extension de l'algorithme de retransmission rapide de Reno (acquittements partiels)
Vegas	Estimation de la bande passante (utilisation des RTT) : variation de la fenêtre de transmission à partir d'une comparaison du taux de transmission attendu par rapport au taux de transmission en cours
Sack	Utilisation des acquittements sélectifs pour l'algorithme de retransmission rapide

Tableau 2.1 : Récapulatif des caractéristiques des différentes versions de TCP

2.9 Conclusion

TCP procure un service fiable, orienté connexion, à flux de données, au niveau de la couche transport. TCP assure donc un transfert de données fiable d'extrémité à une autre, fournit un contrôle de flux, ainsi qu'un contrôle d'erreur. Il s'interface avec un processus applicatif et un protocole de niveau inférieur du type Internet Protocol. Ce dernier fournit à TCP les adresses source et destination des correspondants, ainsi que d'autres informations. Face à une congestion détectée par l'expiration du timeout ou de la réception d'acquittements dupliqués, TCP déclenche les différents algorithmes d'évitement et de contrôle de congestion.

La question posée est : comment réagira TCP face à la mobilité des unités ? sachant que les adresses des nœuds mobiles ne sont plus statiques et que des pertes de paquets (non dues à la congestion mais dues aux caractéristiques des environnements mobiles) sont fréquentes. Ces problèmes seront analysés dans les chapitres suivants.

Chapitre 3 La macro mobilité

3.1 Introduction

L’objectif de ce chapitre est de présenter l'une des solutions proposées par l'IETF pour résoudre le problème de la mobilité dans les réseaux IP : Mobile IP. Dans ce chapitre, après une brève introduction aux problèmes liés à la mobilité dans les environnements IP, on décrit le protocole Mobile IP, qui est actuellement le plus largement utilisé dans l’Internet pour gérer la mobilité. Ce protocole est adapté aux problèmes de macro mobilité (c'est à dire entre différents domaines de mobilité).

3.2 Problème de la mobilité IP

Le protocole IP identifie le point d'accès d'un nœud sur l'Internet de manière unique grâce à son adresse IP. Celle-ci se décompose en deux parties :

- le préfixe qui détermine le sous-réseau sur lequel la machine se trouve,
- l’identifiant de la machine sur son sous-réseau.

L’Internet est un réseau de trop grande taille pour que chaque routeur puisse mémoriser une route vers toutes les machines qui y sont attachées. En fait, les routeurs ne stockent que des entrées correspondant à des sous-réseaux, considérant que des datagrammes destinés à des machines ayant le même préfixe seront tous routés de manière identique.

La mobilité introduit un nouveau problème de routage : les mobiles se déplacent d’un sous-réseau IP vers un autre sous-réseau IP, mais ces derniers ont des préfixes différents. Par conséquent, un nœud doit être situé sur le réseau indiqué par son adresse IP afin de pouvoir recevoir les paquets qui lui sont destinés. Pour qu'un nœud puisse changer de point d'accès sans perdre la possibilité de communiquer, deux mécanismes peuvent être employés :

- le nœud doit changer d'adresse IP à chaque fois qu'il change de point d'accès,
- des chemins spécifiques à l'hôte doivent être propagés dans presque toute la structure de routage de l'Internet.

Ces deux alternatives sont souvent inacceptables. La première ne permet pas à un nœud de conserver des connexions au niveau de la couche transport ou des couches supérieures lorsqu'il change de position. La seconde pose des problèmes de passage à l'échelle.

Un nouveau mécanisme flexible est nécessaire afin de s'adapter à la mobilité des nœuds sur Internet. Le protocole Mobile IP permet aux nœuds de changer de point d'accès à l'Internet sans changer d'adresse IP.

Les spécifications minimales de la solution recherchée sont les suivantes :

- le déplacement d'un mobile ne doit pas provoquer de coupure des connexions ouvertes,
- l’opération doit être simple à mettre en œuvre et d'un coût raisonnable,
- l’accès aux ressources doit être transparent,
- la solution doit être compatible avec le protocole IP et en particulier avec les algorithmes de routage. Le support de la mobilité ne doit pas nécessiter la modification de tous les routeurs.

De nombreux protocoles et architectures ont été proposés pour gérer la mobilité. Par exemple, Muse-IP dans [TER89], le réseau virtuel de Sunshine et Postel [SUN80], le réseau logique dans [DEE95], des solutions hybrides dans [ION91], VIP proposé dans [TER91] et le protocole Mobile IP de l’IETF.

3.3 Principaux enjeux de la mobilité dans IP

Dans ce contexte, on définit succinctement quelques enjeux importants pour la gestion de la mobilité. Ces enjeux bien connus conduiront à proposer une comparaison sur des bases bien assises au niveau des concepts. Ils sont :

– La dualité micro-mobilité/macro-mobilité : On sépare généralement la gestion de la mobilité en macro et micro-mobilité. Il s'agit en fait de gérer les déplacements des mobiles entre les domaines pour la macro-mobilité et à l'intérieur d'un domaine donné pour la micro-mobilité. Cette division est aujourd'hui largement admise et les deux types de mobilité sont gérés indépendamment. On utilise généralement Mobile IP comme protocole de gestion de la macro-mobilité, la micro-mobilité faisant encore l'objet de nombreuses recherches.

– La gestion du handoff : Le handoff concerne l'adaptation du réseau aux changements de point d'attachement d'un mobile au cours de son mouvement. Ceci peut être vu aussi bien au niveau macro-mobilité qu'au niveau micro-mobilité. La gestion du handoff est évidemment le point essentiel dans la gestion de la mobilité, un nœud mobile pouvant provoquer de nombreux handoffs durant une connexion. Ce sera le critère déterminant pour juger une proposition. [REI01]

– Le support de la connectivité passive (la pagination) : La connectivité passive est issue d'un problème bien connu en téléphonie mobile : l'économie des batteries des stations mobiles. Emettre coûte cher en terme de consommation d'énergie et les batteries des appareils portables sont de faible capacité. Typiquement, les hôtes 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. Etre "toujours connecté" de cette manière aboutit à être accessible à tout instant et de pouvoir accéder aux ressources Internet. Les utilisateurs mobiles connectés à l'Internet sans fil s'attendront à un service semblable. Malheureusement, le maintien de l'information de localisation par les mobiles pour rester continuellement accessible, exigeraient des mises à jour de localisations fréquentes, ce qui consommerait de la bande passante précieuse et l'énergie de la batterie. Ce surplus de signalisation et cette consommation électrique d'hôte mobile peut être réduite par l'introduction de la pagination. Le réseau doit alors être partitionné en différentes zones géographiques : les aires de pagination (*paging areas*).

En temps normal, le nœud mobile renseigne sa position à chaque fois qu'il change de station de base. Il s'avère impératif d'éviter que les hôtes mobiles inoccupés n'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 paginer des hôtes mobiles inactifs. Des hôtes mobiles inactifs ne doivent pas enregistrer leur déplacement dans la même aire de pagination pour ne signaler que leur changement d'aire de pagination. [MON01]

3.4 Mobile IP

L'ensemble des protocoles TCP / IP d'Internet a été, à l'origine, conçu sans aucune considération de la mobilité des unités. Sa grande résistance face aux environnements mobiles réside dans le service de routage offert par la couche Internet Protocol qui permet un routage des données dans un environnement statique (des unités ayant des adresses statiques).

Afin de supporter la mobilité, le protocole IP a été amélioré pour donner naissance au protocole Mobile IP développé par l'IETF (*Internet Engineering Task Force*). Ce dernier assure une mobilité transparente des utilisateurs dans l'Internet. Ce protocole garantit le routage correct des paquets d'un nœud mobile lorsque celui-ci change de point d'attachement sur l'Internet.

L'objectif de cette section est de présenter l'une des solutions proposées par l'IETF pour résoudre le problème de la mobilité dans les environnements IP : Mobile IP [PER96a]. Il est

actuellement le plus largement utilisé dans l'Internet pour gérer la mobilité. Ce protocole est adapté aux problèmes de macro mobilité (c'est à dire entre différents domaines de mobilité).

3.5 Quelques Définitions

▪ Nœud mobile (*Mobile Node* MN ou hôte mobile MH) : c'est un hôte ou routeur qui change de point d'accès d'un réseau (ou sous-réseau) à un autre. Comme toute machine fixe, un mobile appartient initialement à un réseau sur Internet. Ce réseau, appelé réseau mère (*Home Network*), affecte son adresse IP au mobile. Un nœud mobile peut changer de position sans changer d'adresse IP.

▪ *Home Agent* (HA) : c'est le routeur sur le réseau mère d'un mobile, qui envoie les datagrammes dans un tunnel pour les remettre au mobile lorsqu'il visite un autre réseau. Le *home agent* met à jour les informations concernant la position du mobile.

▪ *Foreign Agent* (FA) : c'est le routeur sur un réseau visité par le nœud mobile, qui fournit des services de routage au mobile lorsqu'il est enregistré auprès de lui.

▪ Adresse permanente / temporaire (COA) : Un nœud mobile possède une adresse IP permanente (*Home Address*) sur son réseau mère *(Home Network*). Lorsqu'il visite un autre réseau (*Foreign Network*), une adresse temporaire (*care-of address*) est affectée au mobile. Cette adresse reflète le point d'accès du mobile. En général, le mobile utilise son adresse mère comme adresse source dans tous les datagrammes IP qu'il envoie.

▪ Nœud Correspondant (CN ou CH) : Machine (mobile ou non) qui dialogue avec un mobile.

▪ Agent de mobilité « *Mobility Agent* » : *home agent* ou le f*oreign agent.*

3.6 Fonctionnement du protocole Mobile IP

3.6.1 Principe de base

Un nœud mobile se déplace de réseau en réseau et s'attache à des bases (Figure 3.1). Une base implémente des fonctionnalités de niveau 2 du modèle OSI. Elle assure une connectivité de niveau liaison de données et permet l'échange d'informations avec un mobile par un canal sans fil. Quand un correspondant veut envoyer des données à un mobile, il crée un paquet IP ordinaire. L'adresse IP source est celle du correspondant, l'adresse IP destination, celle du mobile sur son réseau mère. Les paquets arrivent donc sur le réseau mère du mobile où ils sont interceptés par le *home agent*. Celui-ci transmet les paquets vers la position courante du mobile.

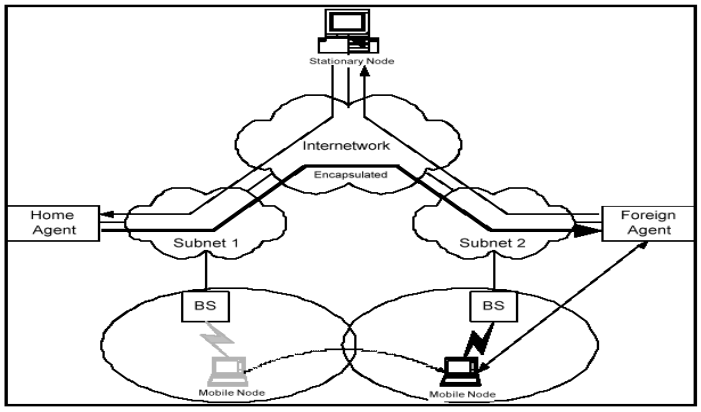


Figure 3.1 : Exemple de deux sous – réseaux avec le support de mobilité

Les services suivants sont définis pour Mobile IP :

▪ Découverte d'agent (*Agent Discovery*) : Les *foreign agents* et les *home agents* doivent avertir les nœuds mobiles de leur disponibilité d'offrir un service. Un nœud mobile peut envoyer des messages de sollicitation afin de prospecter la présence d'agent.

▪ Enregistrement : Quand un nœud mobile est loin de son réseau mère, il enregistre son adresse temporaire «*care-of-address*» au niveau de son *home agent*. Le nœud mobile enregistre son adresse temporaire soit directement avec son *home agent* ou à travers le *foreign agent* qui envoie l'enregistrement au *home agent*, cela dépend de la méthode d'attachement du nœud mobile.

Les étapes suivantes expliquent les opérations principales qu'effectue le protocole Mobile IP :

- Les agents de mobilité avertissent de leurs présences les nœuds mobiles via des messages d'annonciation. Un nœud mobile peut solliciter un agent de mobilité à travers un message de sollicitation.
- Un nœud mobile reçoit des messages d'annonciation et détermine s'ils proviennent de son réseau mère ou d'un réseau visité.
- Quand le nœud mobile détecte qu'il est dans son réseau mère, il n'utilise pas les services de la mobilité.
- Quand un nœud mobile détecte qu'il a changé de localisation vers un autre réseau, il obtient une *care-of-address* dans son *foreign network*. L'adresse temporaire peut être déterminer soit à partir du message d'annonciation du *foreign agent* (*foreign agent care-of-address*) ou par un mécanisme externe d'affectation d'adresses tel que DHCP (*Dynamic Host Configuration Protocol*).
- Quand le nœud mobile est en dehors de son *home network*, il doit enregistrer sa nouvelle adresse temporaire au niveau de son *home agent* en échangeant avec ce dernier des messages de demande d'enregistrement « *Registration Request messages* » et des messages de réponse d'enregistrement « *Registration Reply messages* » (possible via un *foreign agent*).
- Les datagrammes envoyés à l'adresse mère du nœud mobile sont interceptés par son *home agent*, ensuite envoyés par le *home agent* à l'adresse temporaire du nœud via un tunnel, arrivés à la fin du tunnel (*foreign agent* ou le nœud mobile lui-même), ils sont délivrés au nœud mobile.
- Dans la direction inverse, les datagrammes envoyés par le nœud mobile sont généralement délivrés à leur destination en utilisant le mécanisme de routage standard, ils ne passent pas nécessairement par le *home agent*.

3.6.2 Adresse temporaire

Lorsqu'un mobile quitte son réseau mère, Mobile IP utilise le tunnelage « *tunneling* » pour cacher l'adresse mère du mobile aux routeurs situés entre le réseau mère et le mobile (Figure 3.2). La fin du tunnel correspond à l'adresse temporaire du mobile. Cette adresse temporaire doit être une adresse à laquelle les datagrammes peuvent être remis par des mécanismes de routage IP classiques. A l'adresse temporaire, le datagramme d'origine est enlevé du tunnel et remis au nœud mobile.

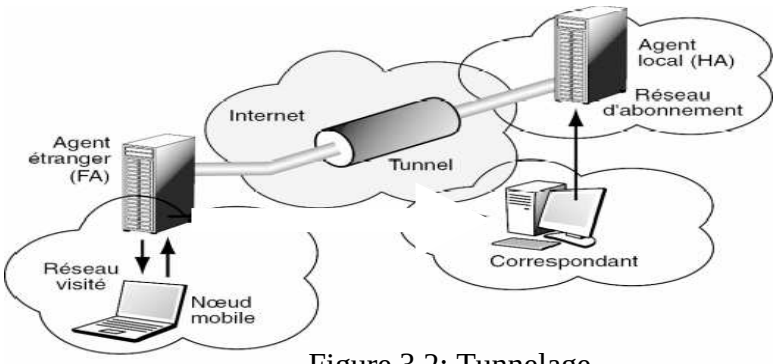


Figure 3.2: Tunnelage

L'adresse temporaire peut être obtenue de deux manières différentes :

- Une "f*oreign agent care-of address*" est une adresse temporaire fournie par un f*oreign agent* grâce aux messages d'annonciation. Dans ce cas, l'adresse temporaire est l'adresse IP du f*oreign*

agent. C'est donc le f*oreign agent* qui est à l'extrémité du tunnel, lorsqu'il reçoit les datagrammes tunnelés, il les décapsule et remet le datagramme d'origine au mobile. Ce mode d'acquisition d'adresse temporaire est préférable car il permet à de nombreux nœuds mobiles de partager une même adresse temporaire.

- Une "*co- located care-of address*" est une adresse IP locale, acquise par des moyens externes, que le mobile associe à l'une de ses interfaces réseau. Cette adresse peut être acquise dynamiquement par des mécanismes tels que DHCP, ou peut être possédée par le mobile comme adresse à utiliser dans certains réseaux visités. Lorsque le nœud mobile utilise une *co- located care-of address*, il se trouve lui-même à l'extrémité du tunnel et décapsule les datagrammes tunnelés jusqu'à lui.

3.6.3 Encapsulation IP dans IP

Le *home agent* ne fait qu'encapsuler les paquets, c'est à dire qu'il ne les modifie pas de telle façon qu'il n'a pas besoin de recalculer les sommes de contrôle (*checksum*) des couches supérieures. Il se contente donc d'ajouter un nouvel en-tête IP devant l'en-tête du datagramme reçu [PER96b], comme le montre la Figure 3.3. L'entête IP contient en réalité plus d'informations que ce qui est représenté sur la figure ci-dessous. Ainsi les datagrammes sont facilement redirigés et routés dans l'Internet. L'adresse IP vers laquelle le *home agent* fait suivre les paquets destinés au mobile est la *care-of address*.

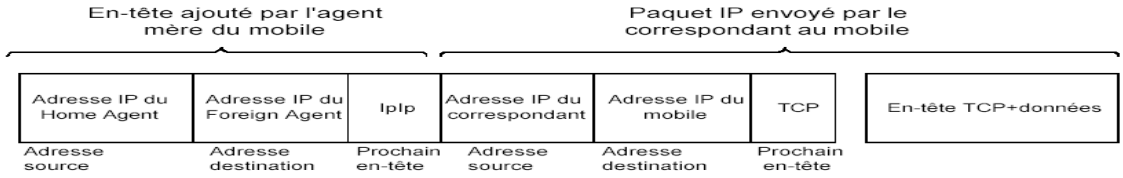


Figure 3.3 : Encapsulation IP dans IP

Mobile IP définit un ensemble de messages de contrôle, envoyés avec UDP en utilisant le numéro de port 434. On s'intéresse à deux types de messages : demande d'enregistrement (*Registration Request*) et réponse d'enregistrement (*Registration Reply*). Pour la découverte des agents, Mobile IP utilise les messages existants d'annonciation de routeur « *Router Advertisement* » et de sollicitation de routeur « *Router Solicitation* » définis pour la découverte de routeurs.

3.6.4 Découverte d'agent (*Agent discovery*)

La découverte d'agent est la méthode utilisée par un nœud mobile pour déterminer s'il est actuellement connecté à son réseau mère ou un réseau visité (*foreign network*). Cette méthode permet aussi au nœud mobile de déterminer l'adresse temporaire offerte par chaque *foreign agent* dans le réseau visité.

Pour la découverte des agents, Mobile IP utilise deux types de messages :

- **Message d'annonciation** « *Agent Advertisement* message »: un message d'annonciation est formé en ajoutant une extension (*Mobility Agent Advertisement Extension*) dans un message ICMP d'annonciation de router (« *Internet Control Message Protocol » Router Advertisement message*). Ces messages sont envoyés par un agent de mobilité aux nœuds mobiles afin qu'ils déterminent leurs points courants d'attachement à Internet.
- **Message de sollicitation** « *Agent Solicitation message»* : un message de sollicitation est identique à "*ICMP Router Solicitation message"* excepté que son champ TTL (*Time to live* ou durée de vie du datagramme) doit être à 1 (en secondes). Il est envoyé par le nœud mobile pour solliciter un agent de mobilité. Aussitôt qu'un message d'annonciation est reçu, les sollicitations s'arrêtent.

3.6.5 Détection de mouvement (Handoffs)

Trois mécanismes sont fournis aux nœuds mobiles pour détecter leurs mouvements d'un sous réseau à un autre. Quand un nœud mobile détecte son mouvement, il doit enregistrer son adresse temporaire au niveau du nouveau *foreign agent*.

- **Lazy Cell Switching (LCS) :** Cette méthode utilise la durée de temps (*lifetime*) avant l'expiration d'un message d'annonciation comme une indication d'un mouvement du nœud mobile. Ce dernier doit enregistrer le *lifetime* reçu dans chaque message d'annonciation, jusqu'à ce que ce *lifetime* expire. Si entre temps le nœud mobile reçoit des messages d'annonciation d'autres agents de mobilité, alors il choisit de ne pas effectuer un handoff et ignore tout nouvel agent découvert. Après expiration du *lifetime* (quand le nœud mobile manque trois messages d'annonciation successifs indiquant qu'il a quitté le réseau), le nœud mobile supposera qu'il a perdu contact avec cet agent, et ainsi il peut tenter un enregistrement avec d'autres agents ou envoyer des messages de sollicitation pour découvrir d'autres agents.

  L'avantage de cette méthode est quelle évite les handoffs entre les agents de mobilité dans un même sous-réseau. En supposant que le *broadcast* des messages d'annonciation atteint tous les nœuds mobiles d'un même sous-réseau, alors il est improbable qu'un nœud mobile perde contact avec son agent courant durant *l'advertisement lifetime*. Cette méthode peut être appliquée à des scénarios où le nœud mobile est dans l'aire de chevauchement de plusieurs sous-réseaux. Si la technologie des réseaux sans fil le permet, le nœud mobile peut recevoir plusieurs messages d'annonciation différents. En absence de la capacité de la technologie sans fil de participer à différents sous réseaux, cet algorithme entraîne un délai d'attente non nécessaire.

- **Pattern Matching (PM) :** Cette méthode a une fonctionnalité similaire à celle de LCS avec une seule différence. Le nœud mobile compare les préfixes des sous-réseaux des agents de mobilité afin de déterminer les nouveaux agents et éliminer les agents d'un même sous-réseau. Normalement, un message d'annonciation ne contient pas d'information concernant la taille du préfixe du sous réseau de l'agent. Par conséquent, pour qu'un nœud mobile utilise ce schéma de détection de mouvement, tous les agents doivent inclure une extension (*prefix-length*) contenant la longueur du préfixe de l'adresse de l'agent dans leurs messages.

  Cette méthode est utile quand le nœud mobile peut recevoir des messages d'annonciation de plusieurs agents localisés soit dans le même sous réseau ou dans différents sous réseaux. En supposant, qu'il n'existe qu'un seul agent dans un sous réseau, et que la technologie d'un nœud ne lui permet pas de participer à de multiples sous réseaux simultanément, alors cette méthode est sans effet. Les méthodes LCS et PM tendent à avoir un même comportement sous les conditions précédentes.

- **Eager Cell Switching (ECS) :** Cette méthode fonctionne d'une manière contraire à celle de LCS. Elle suppose qu'un nœud mobile tend à changer sa direction de mouvement très lentement. Alors s'il bouge dans une certaine direction, il est peu probable qu'il s'arrête et fait marche arrière. Il est approprié donc aux nœuds de changer de cellules (handoffs) immédiatement dés qu'ils rencontrent un nouveau message d'annonciation. Cette méthode tend à réduire l'intervalle de détection de mouvement en comparaison avec la méthode LCS et ainsi gère rapidement les handoffs.

3.6.6 Enregistrement

Des mécanismes de communication entre les mobiles et leurs f*oreign agent*s sont nécessaires. En particulier, il faut qu'un mobile puisse apprendre l'adresse IP des agents relais. Le mobile doit ensuite s'enregistrer auprès d'un f*oreign agent* et obtenir son accord pour qu'il

relaye les paquets. Puis, le *home agent* doit être informé de l'adresse IP du nouveau f*oreign agent*, qui devient l'adresse temporaire du mobile (COA).

On peut également transmettre cette adresse à l'ancien f*oreign agent* (sur le réseau que le mobile a quitté) pour que les paquets en transit au moment du déplacement du mobile arrivent bien à destination. En effet, il se peut que le *home agent* continue à envoyer des paquets lors de la procédure de changement de f*oreign agent*.

L'enregistrement dans Mobile IP procure un mécanisme flexible aux nœuds mobiles pour communiquer leurs informations d'atteignabilité courantes à leurs *home agent*s. C'est une méthode par laquelle les nœuds mobiles :

- Demandent de nouveaux services quand ils visitent un *foreign network*,
- Informent leurs *home agent*s de leurs adresses temporaires,
- Renouvellent un enregistrement qui va expirer, et / ou,
- Dé-enregistrent quand ils retournent au réseau mère.

Mobile IP définit deux procédures d'enregistrement, l'une à travers le *foreign agent* qui transmet l'enregistrement au *home agent* du nœud mobile, et l'autre directement avec le *home agent* du nœud. Les règles suivantes déterminent quelle procédure utilisée dans des circonstances particulières :

- Si le nœud mobile a enregistré une *care-of-address* du *foreign agent* alors il doit s'enregistrer via ce *foreign agent*,
- Si le nœud mobile a utilisé une *co-located care-of-address*, alors il doit effectuer son enregistrement directement avec son *home agent*,
- Si le nœud mobile est retourné à son réseau mère et il s'est dé-enregistré avec son *home agent*, alors il doit s'enregistrer directement avec son *home agent*.

Les deux procédures impliquent l'échange des messages de demande d'enregistrement (*Registration Request message*) et de réponse d'enregistrement (*Registration Reply message*). La réception par un mobile de l'un de ces messages lui permet de savoir s'il est ou non sur son réseau mère et de démarrer la procédure d'enregistrement.

▪ **Enregistrement via un *foreign agent* :** Lorsque l'enregistrement se fait via un *foreign agent*, la procédure d'enregistrement se décompose en quatre messages [NOE98] :

- demande d'enregistrement au f*oreign agent* (*Registration Request*),
- traitement de la demande par le f*oreign agent*, puis transmission de la demande au HA,
- notification d'acceptation ou de refus de prise en charge du mobile du HA au FA (*Registration Reply*),
- traitement de la notification par le f*oreign agent* et transmission de l'information au mobile.

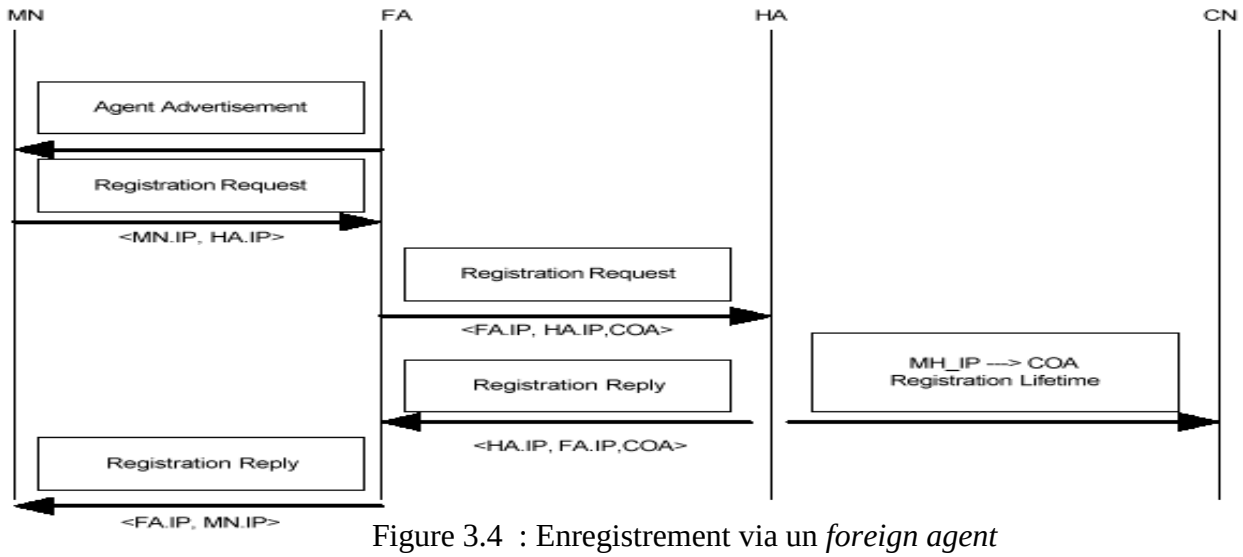


Figure 3.4 : Enregistrement via un *foreign agent*

La procédure d'enregistrement est accompagnée d'une authentification afin d'assurer que les paquets destinés au mobile ne sont pas détournés par une autre machine. Lorsque le mobile est situé sur son réseau mère, il n'utilise pas les services de la mobilité.

L'une des extensions des messages "*ICMP Router Discovery"*, appelée "*Mobility Agent Advertisement Extension*", permet d'utiliser un champ durée de vie (TTL), présent dans les messages "*ICMP Router Advertisement*" envoyés régulièrement par un f*oreign agent*. Quand un mobile se connecte sur un réseau, il enregistre la durée de vie associée à son f*oreign agent*. Lorsque cette valeur expire, le mobile suppose qu'il n'est plus géré par ce f*oreign agent*. Par contre, si le mobile reçoit un message d'annonciation d'un autre f*oreign agent*, il s'enregistre auprès de celui-ci.

- **Enregistrement direct :** Quand le nœud mobile s'enregistre directement avec son *home agent* la procédure d'enregistrement requiert uniquement l'échange des deux messages suivants [PER96a]: Le nœud mobile envoie un message de demande d'enregistrement au *home agent*, Le *home agent* envoie à son tour un message de réponse au nœud, tout en précisant l'acceptation ou le refus de la requête.

3.6.7 Routage

Lorsqu'un correspondant veut dialoguer avec un mobile (sens descendant Figure 3.5), il ne connaît pas la position courante de ce dernier et envoie les datagrammes à l'adresse mère du mobile. Ceux-ci sont interceptés par le *home agent* et envoyés dans un tunnel du *home agent* vers l'adresse temporaire du mobile. A la sortie du tunnel (au niveau soit du f*oreign agent*, soit du mobile lui-même), les datagrammes sont transmis au mobile. Si le correspondant est mobile, il doit également passer par son propre f*oreign agent*.

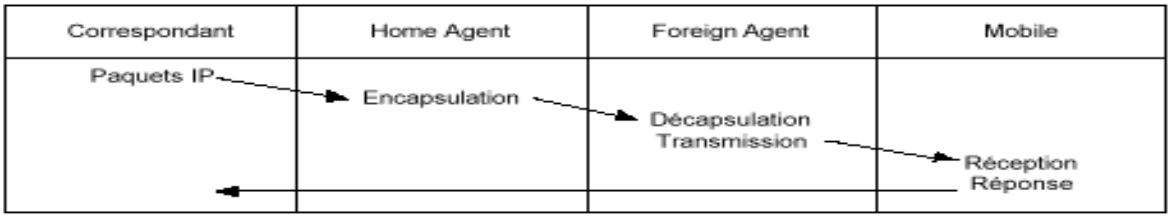


Figure 3.5 : Dialogue avec un mobile

Dans l'autre sens (sens montant), les datagrammes envoyés par le nœud mobile sont généralement remis aux destinataires en utilisant des mécanismes de routage IP standards, en passant par le *home agent* (mode bidirectionnel) ou non (mode standard) [PER96a].

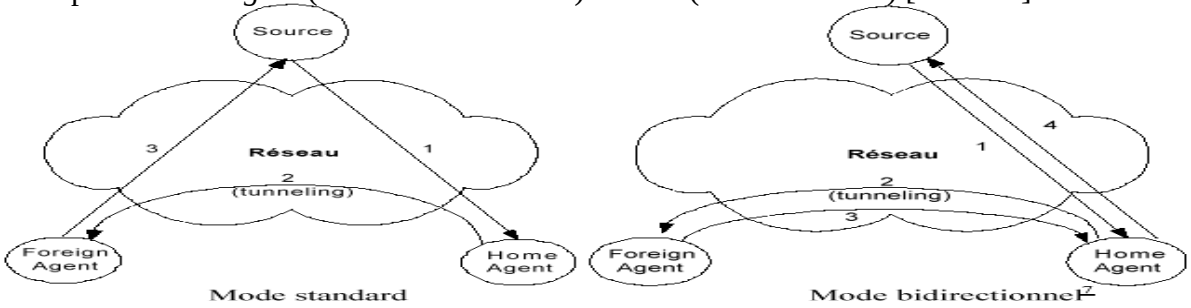


Figure 3.6 : Routage des données

Le mode bidirectionnel est très peu utilisé (Figure 3.6). Il peut servir lorsqu'un mobile reçoit des paquets *multicast*, si son *home agent* est un routeur *multicast*. Le mobile encapsule les messages vers son *home agent* et ce dernier envoie les messages *multicast* pour le mobile dans le tunnel. Pour des paquets tunnelés au HA, l'adresse source dans l'en-tête IP doit être l'adresse mère du mobile.

La station émettrice (E) envoie un datagramme destiné au mobile (Figure 3.7). E utilise l'adresse mère du mobile sans se préoccuper de la mobilité.

Cas 1 : Le mobile est actuellement dans son réseau mère, son HA ne procède à aucune action particulière, le paquet est acheminé normalement jusqu'à lui.

Cas 2 : Le mobile s'est déplacé et ne dépend plus de son réseau mère. Le HA intercepte le paquet et l'encapsule dans un nouveau datagramme IP en mettant la dernière adresse temporaire connue du mobile. Il s'agit ici de l'adresse du FA et non d'une adresse attribuée dynamiquement. Le paquet arrive au FA qui le décapsule et envoie le datagramme initial sur son réseau.

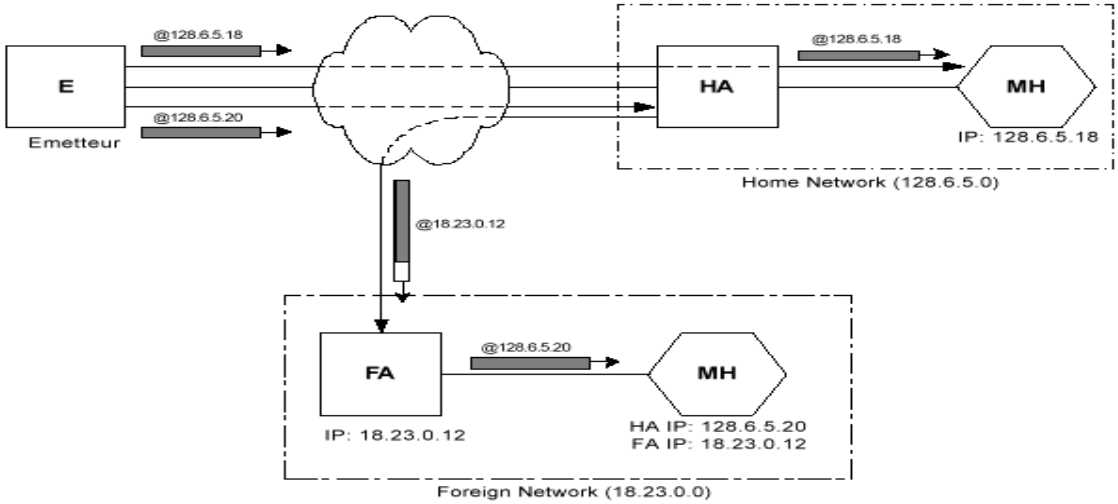


Figure 3.7 : Exemple de routage

▪ **Routage triangulaire :** Du point de vue du routage, la principale faiblesse de Mobile IP est le routage triangulaire (c'est le routage utilisé par défaut). Il devient particulièrement peu performant lorsque le mobile en déplacement et son correspondant sont sur le même réseau IP. En effet, lorsqu'un mobile hors de son sous-réseau mère cherche à joindre une machine sur le même réseau visité, les paquets doivent néanmoins transiter par le sous-réseau mère du mobile. Ce problème n'apparaît pas dans le sens mobile vers correspondant car Mobile IP définit un routage direct.

▪ **Routage optimisé :** Afin d'obtenir un routage performant (C'est à dire qui évite le routage triangulaire) dans le sens correspondant vers mobile, [PER96b] propose d'ajouter des mécanismes supplémentaires sur les correspondants d'un mobile et de modifier la pile de protocoles TCP/IP. Mais cette solution supprime la transparence du protocole par rapport aux utilisateurs, c'est à dire qu'un correspondant doit à présent savoir s'il dialogue avec une machine fixe ou mobile. Dans cette proposition, seul le premier paquet envoyé à un mobile passe par son home agent, après réception de ce paquet, le home agent indique au correspondant que la machine qu'il cherche à joindre est un mobile et fournit l'adresse par laquelle il peut être joint (son adresse temporaire).

Un mobile peut avoir plusieurs correspondants susceptibles d'utiliser des routes non optimales (c'est à dire qui passent par le *home agent*). Le *home agent* mémorise donc la liste des correspondants de chacun des mobiles qu'il gère pour informer les correspondants des mouvements des mobiles.

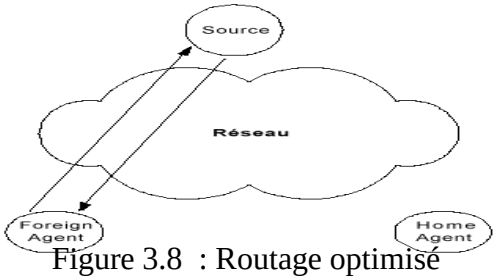


Figure 3.8 : Routage optimisé

3.6.8 Extensions

Mobile IP définit un mécanisme d'extension général pour permettre aux messages de contrôle Mobile IP et aux messages de découverte de routeurs ICMP de transporter des informations supplémentaires.

Certaines extensions n'apparaissent que dans les messages des contrôle Mobile IP : Authentification mobile/*home agent,* Authentification mobile/f*oreign agent et* Authentification f*oreign agent* / *home agent.*

Les extensions présentes uniquement dans les messages de recherche de routeur ICMP sont : bourrage d'un octet, annonce d'agent de mobilité (*Mobility agent advertisement*) et longueurs des préfixes.

3.7 Limites de Mobile IP

Mobile IP a été conçu pour permettre aux hôtes mobiles de se déplacer d'un sous-réseau IP à un autre; il résout le problème de la "macro-mobilité". Il est moins bien adapté à la "micro-mobilité", lorsque les mobiles effectuent des handoffs fréquents dans une même zone géographique. Dans la version standard de Mobile IP, la nouvelle localisation d'un mobile est toujours signalée à son *home agent*. Ce dernier est ainsi averti de tous les déplacements des mobiles qu'il gère. Ces opérations génèrent une grande quantité de signalisation. De plus, les pertes de paquets pendant les handoffs peuvent être importantes puisque la procédure d'enregistrement est longue, en particulier si le *home agent* se trouve à l'autre bout du monde. La durée d'un handoff peut atteindre plusieurs secondes dans l'Internet actuel.

D'autre part, pour des mobiles bénéficiant d'une certaine qualité de service, l'acquisition d'une nouvelle *care-of address* pour chaque handoff implique de remettre en place des réservations entre le *home agent* et le nouveau f*oreign agent*. Cependant, une grande partie du chemin entre le correspondant et le mobile risque d'être identique avant et après le handoff (si le mobile ne change pas de domaine).

3.8 Conclusion

Ce chapitre a présenté le protocole Mobile IP qui supporte la mobilité des utilisateurs de l'Internet. Mobile IP est conçu pour permettre aux nœuds de se déplacer d'un sous-réseau IP à un autre; il résoud le problème de la macro mobilité. Cependant, il est moins bien adapté à la "micro- mobilité", lorsque les mobiles effectuent des handoffs fréquents dans une même zone géographique. L'accès sans fil risque de devenir de plus en plus fréquent dans les prochaines années. Les utilisateurs mobiles demanderont évidemment des niveaux de qualité de service identiques à ceux offerts aux utilisateurs de postes fixes. D'un autre côté, le protocole utilisé pour gérer la mobilité IP (MIP) fait cruellement ressentir ses limites dans un tel environnement. Une telle vision présente donc un certain nombre de défis techniques pour la mobilité IP en terme de performance et de mise à l'échelle. C'est donc pour cela qu'un certain nombre de protocoles de micro mobilité visant à améliorer les performances de MIP ont été présentés dans le groupe de travail de l'IETF.

Chapitre 4 La micro mobilité

4.1 Introduction

La micro mobilité représente la gestion de la mobilité locale. Elle permet à des mobiles de s'enregistrer localement à l'intérieur du réseau qu'ils visitent. Cet enregistrement local réduit le délai de signalisation, ce qui peut améliorer les performances des handoffs.

Plusieurs solutions pour gérer la micro mobilité ont été proposées. Toutes reposent sur le principe suivant : le réseau visité par un mobile se charge des déplacements locaux; le *home agent* n'est donc pas prévenu de tous les changements de localisation du mobile qu'il gère. Ces solutions n'ont nullement l'ambition de se substituer à Mobile IP; ils complètent Mobile IP. Celui-ci gère la macro mobilité alors que ces nouveaux protocoles ne s'occupent que de la micro mobilité. Parmi les protocoles de la micro mobilité on cite Cellular IP[CAM99a], Hawaii[RAM00], TeleMIP [DAS00]et EMA [ONE00], [COR00]. Deux de ces protocoles à savoir Cellular IP et HAWAII seront présentés en détail.

4.2 Courte description des protocoles de micro mobilité

Cette section présente de façon succincte les quatre protocoles de micro-mobilité. Ces quatre propositions s'appuient sur Mobile IP pour gérer la mobilité inter-domaine.

Cellular IP

Cellular IP est un protocole de micro-mobilité reposant sur Mobile IP pour la gestion de la mobilité entre domaines. Cellular IP ambitionne de remplacer IP dans le réseau d'accès. Dans Cellular IP, l'acheminement des paquets est effectué sur base de routes spécifiques établies et mises à jour par le mobile pendant son séjour dans le réseau. Ces routes sont matérialisées dans les stations qui les jalonnent par la présence de *routing cache*. Ces caches contiennent, pour chaque route vers une COA, l'interface par laquelle les paquets qui lui sont destinés doivent être routés. Les routes sont établies par la transmission de proche en proche de paquets spécifiques. Elles sont rafraîchies régulièrement et mises à jour au moment du handoff. Le handoff est géré via deux mécanismes spécifiques : hard handoff et semi-soft handoff. Ce dernier permet de garantir que la perte de paquets due au handoff sera minimale. Un réseau Cellular IP est relié au reste du monde via une passerelle qui centralise toutes les communications, entrantes et sortantes. Il inonde régulièrement le réseau avec un paquet *beacon* qui permet à toutes les stations de savoir par laquelle de leur interface elles peuvent atteindre la passerelle. Cellular IP présente également de façon native un support de la connectivité passive.

HAWAII

Contrairement à Cellular IP, HAWAII ne remplace pas IP mais s'appuie sur lui dans son fonctionnement. Chaque station du réseau doit donc pouvoir fournir les services d'un routeur IP classique plus certaines fonctionnalités de gestion de la mobilité. La gestion de la mobilité se fait de façon très similaire à Cellular IP : chaque station maintient un cache de routage qui lui permet de déterminer le traitement à appliquer aux paquets qu'elle reçoit. Le handoff est traité suivant deux mécanismes. Comme Cellular IP, HAWAII présente un support natif de la connectivité passive. Dans HAWAII, les stations faisant partie d'une *paging area* sont toutes membres du même groupe IP *multicast*. Ceci permet de distribuer une requête de *paging* à toutes les stations d'un même groupe en l'adressant à ce groupe.

TeleMIP

TeleMIP est un protocole bien adapté au cas des réseaux CDMA. TeleMIP définit un nouvel agent de mobilité appelé TeleMIP *Mobility Agent* (TMA). Un réseau TeleMIP est composé d'un ensemble de sous-réseaux (composés d'une machine centrale agissant comme FA local et d'un ensemble de stations de base qui lui sont connectées) et d'un série de machines faisant fonction de TMA. Chaque FA est connecté à au moins un TMA du réseau global. Lorsqu'un mobile se connecte au réseau, il obtient deux adresses temporaires du FA local : l'une est enregistrée auprès d'un TMA et restera valide tant que le MN restera dans le domaine (cela lui servira de COA), l'autre est strictement locale et ne sert à identifier le mobile que tant qu'il reste dans le sous-réseau. Lorsqu'un paquet arrive dans le réseau pour un mobile, il porte la COA du mobile comme adresse destination. Il est intercepté par le TMA concerné qui consulte sa table de correspondance entre adresse locale et COA. Ceci lui permet alors de faire suivre le paquet jusqu'au réseau local où se trouve le mobile à ce moment, le FA local n'ayant plus qu'à délivrer le paquet au MN via la bonne station de base.

EMA

EMA vise à définir un système générique de gestion de la mobilité dans les domaines d'accès mobile. Dans ce système, les auteurs de EMA discutent la possibilité d'utiliser TORA, un protocole de gestion de réseaux ad-hoc, pour le cas des réseaux mobiles mais ne restreignent pas leur approche à ce seul cas particulier. Les réseaux ad-hoc constituent en effet une forme extrême de réseau mobile. Dans ce type de réseau, on ne suppose l'existence d'aucune architecture fixe : les stations sont toutes équivalentes et se déplacent les une par rapport aux autres. Des protocoles très particuliers doivent alors gérer les interactions entre celles-ci pour assurer l'acheminement des paquets dans une topologie sans cesse mouvante. EMA ne fait aucune supposition sur une technique d'accès radio particulière et propose une gestion du handoff transparente pour le protocole de routage tournant au-dessus. Au moment de la connexion au réseau, un MN obtient une adresse du sous-réseau dans lequel il se trouve, permettant ainsi un routage par préfixe tant qu'il y reste dans ce sous-réseau. Lorsqu'un mobile sort du sous-réseau auquel il s'est d'abord connecté, EMA prévoit l'injection de routes spécifiques pour atteindre celui-ci.

4.3 Coexistence de plusieurs solutions dans l'Internet

Les solutions de micro mobilité décrites ci-dessus supposent que Mobile IP sait gérer le problème de la macro mobilité, mais toutes définissent une solution pour gérer la micro mobilité à l'intérieur des réseaux. Dans le pire des cas, ceci peut mener à une situation où les mobiles ne peuvent pas se rendre dans un réseau qui utilise une solution de micro mobilité inconnue.
Il existe deux manières de traiter ce problème:
- définir un unique protocole de micro mobilité qui sera utilisé partout dans l'Internet,
- définir comment plusieurs solutions pour la micro mobilité peuvent coexister dans l'Internet.

La première solution n'est évidemment pas adaptée : elle nécessite de nombreux compromis et il semble qu'il ne peut pas y avoir de solution optimale pour tous les types de réseaux. Par exemple, un réseau sans fil peut utiliser une solution de micro mobilité différente d'un réseau qui ne contient que des liens filaires. Il est indispensable de s'assurer que plusieurs solutions pour gérer la micro mobilité peuvent coexister dans l'Internet de manière à ce que cela soit transparent pour les correspondants et les *home agent* [CAS99].

Les solutions présentées dans ce chapitre fonctionnent dans un environnement IPv4. Mobile IPv6 étant un peu différent de Mobile IPv4, les problèmes et les solutions de micro mobilité varient légèrement (voir annexe 3). Toutefois, les principes de base sont les mêmes et

la plupart des idées présentées pour Mobile IPv4 sont également valides pour Mobile IPv6. Toutes les propositions décrites précédemment peuvent être modifiées pour fonctionner dans un environnement IPv6.

4.4 Cellular IP

Cellular IP est un protocole de la micro-mobilité reposant sur le protocole Mobile IP pour la gestion de la mobilité entre domaines (Figure 4.1). Il s'agit d'une extension de Mobile IP et non d'une solution de remplacement. Il ambitionne de remplacer IP dans les réseaux d'accès sans fil. L'une des différences principales par rapport aux autres solutions de micro mobilité est que dans Cellular IP, la gestion de la localisation des hôtes oisifs (inactifs) n'est pas la même que celle des machines qui envoient ou reçoivent activement des données (actifs).

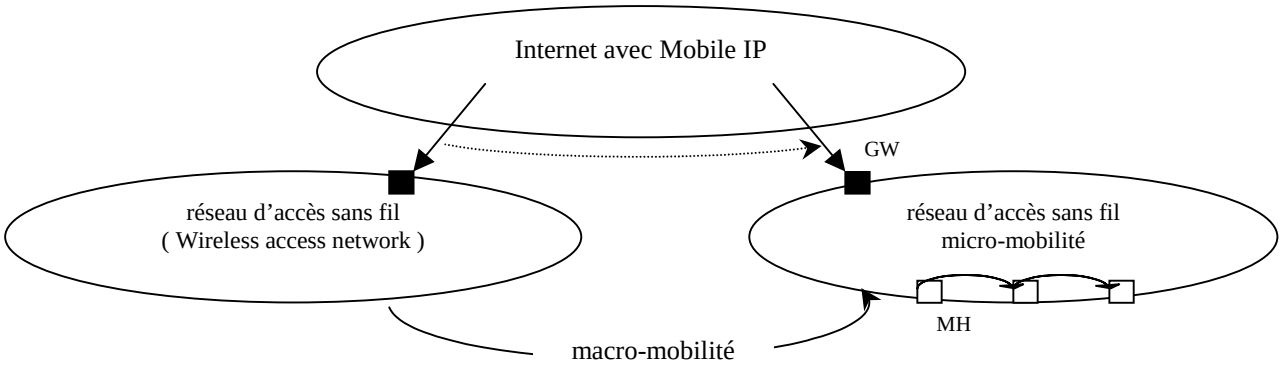


Figure 4.1 : Réseau d'accès sans fil et Mobile IP

Un réseau Cellular IP est relié au reste du monde via une passerelle (*gateway*, GW) qui centralise toutes les communications, entrantes et sortantes. Elle inonde régulièrement le réseau avec un paquet *beacon* qui permet à toutes les stations de savoir par quelle interface elles peuvent atteindre la passerelle. Les hôtes mobiles attachés au réseau utilisent l'adresse IP de la passerelle comme étant leurs adresses temporaires (*care-of address*). Cellular IP présente également de façon native un support de la connectivité passive.

Dans Cellular IP, l'acheminement des paquets est effectué sur base de routes spécifiques établies et mises à jour par le mobile pendant son séjour dans le réseau. Ces routes sont matérialisées dans les stations qui les jalonnent par la présence de *routing cache*. Ces caches contiennent, pour chaque route vers une COA, l'interface par laquelle les paquets qui lui sont destinés doivent être routés. Les routes sont établies par la transmission de proche en proche de paquets spécifiques. Elles sont rafraîchies régulièrement et mises à jour au moment du handoff. Le handoff est géré via deux mécanismes spécifiques : hard handoff et semi-soft handoff.

Dans Cellular IP, les paquets envoyés par l'hôte mobile sont routés de proche en proche en direction de la passerelle. Les stations de bases mémorisent le chemin emprunté par les paquets à leur passage. Afin de router des paquets adressés à l'hôte mobile, le chemin emprunté par le plus récent paquet appartenant au même nœud est inversé.

Quand l'hôte mobile n'a aucune donnée à transmettre, alors il envoie des paquets IP vides à la passerelle afin de maintenir l'itinéraire de la GW vers l'hôte mobile. En suivant le principe de la connexion passive, les hôtes mobiles qui ne reçoivent pas de paquets durant une certaine période autorisent la suppression de leurs itinéraires. Afin de router efficacement les paquets aux hôtes mobiles, Cellular IP utilise le mécanisme *paging* de la téléphonie cellulaire.

La localisation des mobiles Cellular IP est gérée par deux caches au niveau des stations de base : une cache de routage (*routing cache*) dédiée aux mobiles actifs (en train d'envoyer et de recevoir des paquets), et une autre cache (*paging cache*) pour la localisation des mobiles inactifs.

Ces caches contiennent les correspondances (*mappings*) entre l'adresse IP de l'hôte mobile et l'interface de sortie de la station de base à emprunter pour atteindre ce nœud.

4.4.1 L'architecture d'un réseau d'accès sans fil (*Wireless access network*)

Un réseau d'accès sans fil consiste principalement en stations de base interconnectées par des liens filaires (Figre 4.2).

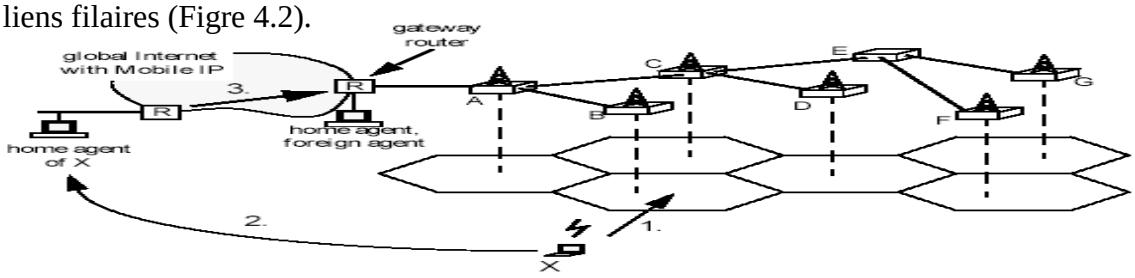


Figure 4.2 : Architecture d'un réseau sans fil

La Figure 4.2 est composée :

– d'une passerelle (*gateway router*), qui assure la liaison entre le réseau sans fil et le reste de l'Internet. Elle filtre, contrôle et propage les paquets en provenance et à destination du réseau.

– de stations de base (*base stations*, BS ou nœud) servant de point d'accès aux mobiles et assurant les fonctions relatives à la mobilité comme la gestion d'emplacement (location management), ainsi que les fonctions d'acheminement des paquets IP dans le réseau sans fil.

– d'hôtes mobiles (*mobile host* ou MH).

Mis à part les stations de base, le réseau peut contenir des nœuds n'ayant pas de dispositifs radio mais servent de concentrateurs de trafic. Dans la Figure 4.2, tous les nœuds ont des dispositifs radio excepté le nœud E. On suppose que dans l'Internet, Mobile IP gère la mobilité des hôtes mobiles. Une fois que l'hôte mobile X entre dans le réseau sans fil (étape 1), il enregistre son adresse temporaire (COA) auprès de son *home agent* (étape 2), qui lui expédiera les paquets qui lui sont adressés (étape 3). Aussi longtemps que l'hôte mobile est relié au même réseau sans fil, sa mobilité est transparente à son *home agent*.

4.4.2 Routage

La passerelle du réseau Cellular IP diffuse périodiquement un paquet balise (*beacon*) qui inonde le réseau d'accès. Les stations de base enregistrent le nœud voisin duquel elles ont reçu cette balise et l'utilisent pour router des paquets vers la passerelle. Tous les paquets transmis par les hôtes mobiles, quelle que soit leur adresse de destination, sont d'abord routés vers la passerelle selon ces mêmes routes par les stations de base, et par la suite vers l'Internet.

Quand un paquet de données engendré par un hôte mobile arrive à une station de base, la cache de routage locale stocke l'adresse IP de l'hôte source ainsi que le nœud voisin duquel est arrivé le paquet : (@IP, BSi, timeout)(Figure 4.3). Cet état de correspondance reste valide pendant un temps spécifique appelé *route-timeout*. Si cet état n'est pas rafraîchi avant l'expiration du *route-timeout* il sera alors supprimé. Les paquets de données sont utilisés pour maintenir et rafraîchir ces correspondances : tant que l'hôte mobile envoie régulièrement des paquets de données, les stations de base se trouvant le long du chemin entre le point de rattachement actuel du mobile et la passerelle maintiendront des correspondances valides dans leur cache de routage; les paquets adressés au mobile seront routés donc de proche en proche en utilisant cette cache de routage.[CAM98]

Un hôte mobile peut parfois vouloir maintenir les correspondances dans les caches de routage même s'il n'est pas en train de transmettre régulièrement des paquets de données. Un exemple typique de cette situation est lorsqu'un hôte mobile reçoit un flux UDP de paquets sur la voie descendante (*downlink*) mais n'a aucune donnée à transmettre sur la voie montante (*uplink*). Afin que les correspondances n'expirent pas, l'hôte mobile transmet alors sur la voie montante des paquets de signalisation Cellular IP "*route-update-packets"* adressés à la passerelle à des

intervalles réguliers (*route-update-time*) rafraîchissant ainsi les états des correspondances dans les caches de routage.

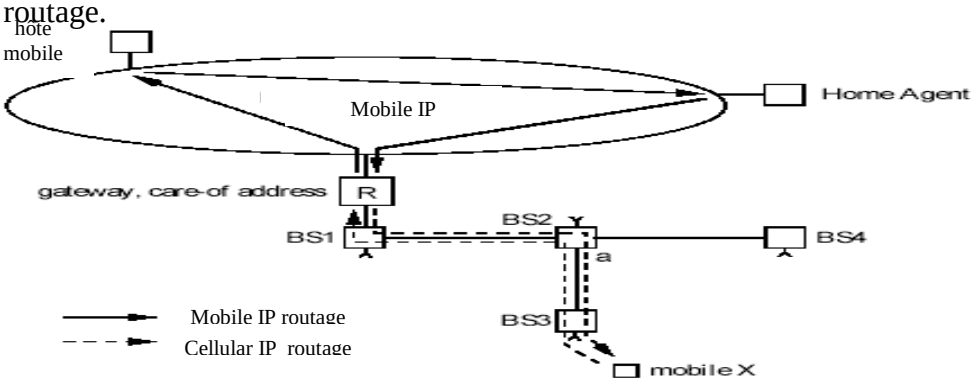


Figure 4.3 : Routage

Dans la direction *uplink* (vers la passerelle), les paquets sont routés dans le réseau Cellular IP de proche en proche. Le voisin auquel un nœud expédiera un paquet adressé à la passerelle est désigné sous le nom du voisin *uplink* du nœud. Les voisins d'un nœud autres que le voisin *uplink* s'appellent les voisins *downlink* (Figure 4.4).

Les nœuds Cellular IP doivent implémenter l'algorithme décrit dans cette section. Ils n'ont pas besoin de la capacité du routage IP standard. Cette section décrit la procédure de routage des nœuds Cellular IP autres que la passerelle. Les fonctions supplémentaires exigées seulement dans la passerelle Cellular IP sont décrites dans la section 4.4.3.

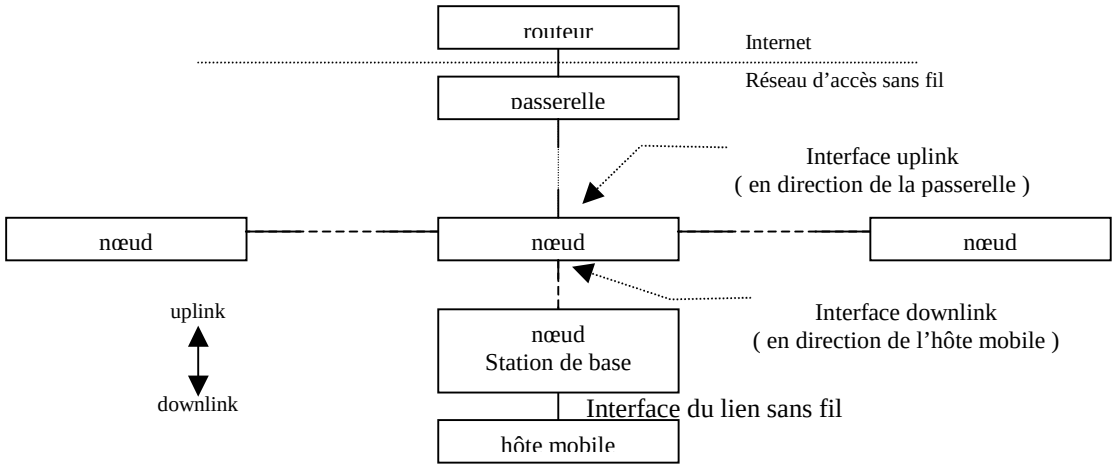


Figure 4.4 : Topologie du réseau d'accès sans fil

▪ **Routage uplink :** Un paquet arrivant à un nœud d'un de ses voisins *downlink* est supposé envoyé par l'hôte mobile. Le paquet est d'abord utilisé pour mettre à jour les *Routing* et *Paging caches* du nœud, ensuite il est expédié au voisin *uplink* du nœud. Pour mettre à jour les caches, le nœud lit le type du paquet, le numéro de port et l'adresse IP de la source. Les paquets p*aging-update* mettent à jour seulement les *paging caches*. Les paquets r*oute-update* mettent à jour les *routing* et *paging caches*. Les paquets de données rafraîchissent seulement les états des deux caches, mais ne les changent pas[VAL99a]. Les deux types de caches se composent de :
{IP-address, interface, MAC address, expiration time, timestamp} 5-tuples, appelés correspondences (*mappings*).

L'adresse IP est l'adresse de l'hôte mobile. L'interface et l'adresse MAC dénotent le voisin *downlink* (vers l'hôte mobile). Le champ *timestamp* contient l'estampille du paquet de contrôle qui a établi l'itinéraire.

Quand un paquet de données arrive d'un voisin *downlink*, l'entrée correspondante à l'adresse IP source est d'abord recherchée dans la *routing cache*. Si le paquet de données vient

du même voisin indiqué par l'entrée de la routing cache, il est alors envoyé dans la même direction où l'hôte mobile a été pour la dernière fois vu. Dans ce cas la correspondance est seulement rafraîchie: le temps d'expiration est remplacé par : temps actuel + *route-timeout*. Si le nœud a une *paging cache*, alors le temps d'expiration de la correspondance dans la *paging cache* est remplacé par : le temps actuel + *paging-timeout*. Ensuite le paquet est expédié au voisin *uplink*. Si le paquet de données arrive d'un voisin différent de celui de la correspondance ou qu'aucune correspondance n'existe pour l'adresse IP, alors le paquet est supprimé.

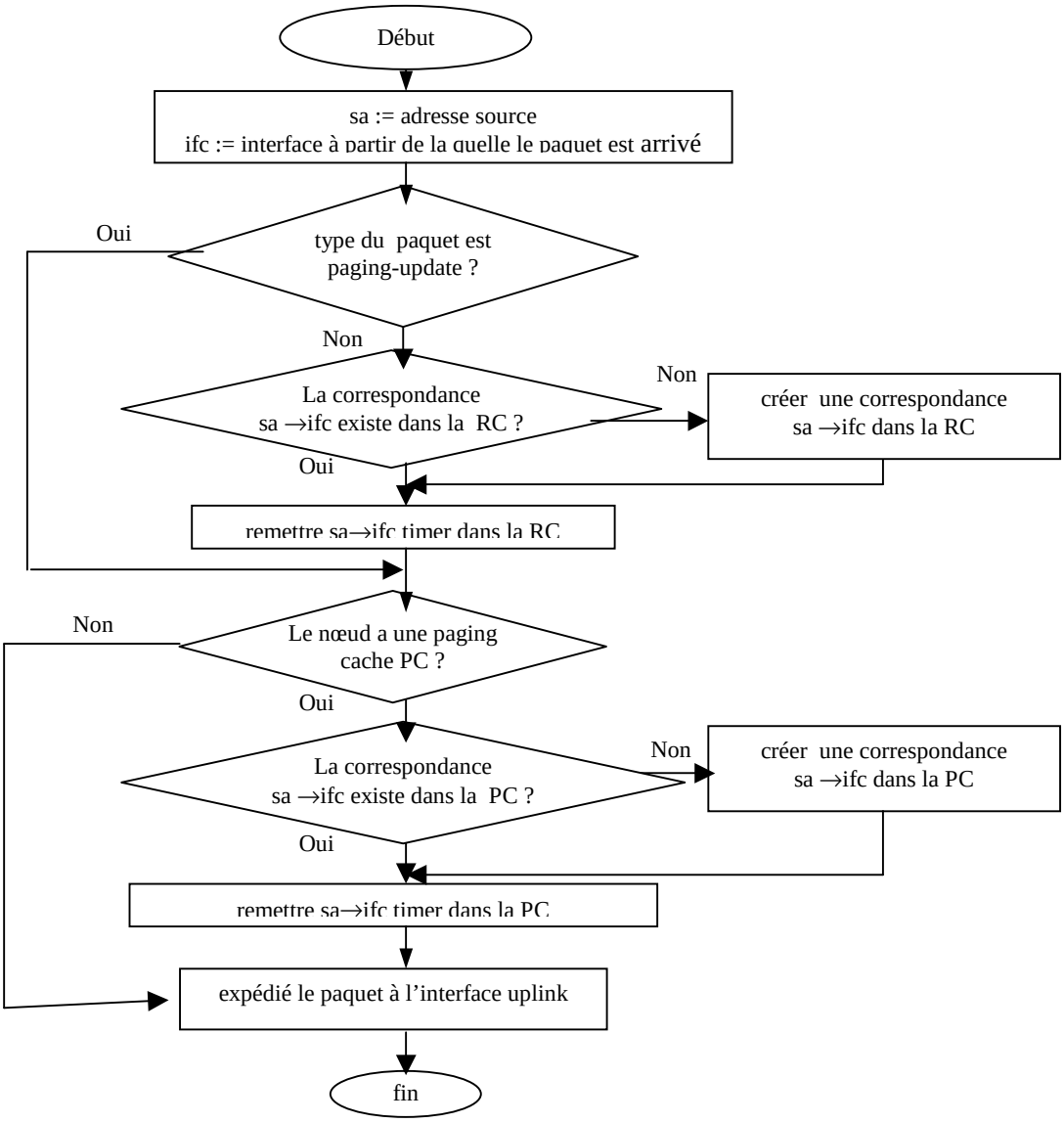


Figure 4.5 : Algorithme de routage *uplink*

Quand un paquet de contrôle arrive d'un voisin *downlink* alors l'authentification doit d'abord être validée. Les paquets avec une authentification incorrecte doivent être supprimés et ce cas devrait être enregistré comme tentative potentielle de trifouillage.
Pour les paquets valides le nœud crée le 5-tuple suivant:
{l'adresse IP de la source du paquet récemment arrivé, l'interface par laquelle le paquet est arrivé, l'adresse MAC de la source du paquet arrivé, temps actuel + *route-timeout*, l'estampille du *update packet* arrivé }

Cette correspondance est utilisée pour mettre à jour la routing cache, si le paquet entrant est un paquet *route-update*. Si une correspondance valide de l'adresse IP de la source existe déjà, alors elle est remplacée par le nouveau 5-tuple si l'estampille est plus récente que l'ancienne, autrement le paquet est supprimé. Si aucune correspondance n'existe pour l'adresse IP source

alors la correspondance est ajoutée à la *routing cache*. La *paging cache* est mise à jour de la même manière, mais en utilisant le *paging-timeout* au lieu du *route-timeout*. Si le nœud n'a pas de *Paging Cache* alors seulement la *routing cache* est mise à jour. Si le paquet entrant est un *paging-update*, alors seulement la *paging cache* est mise à jour.

Si le paquet est un paquet *paging-teardown* et l'information d'authentification est valide, alors les correspondances de l'hôte mobile avec une estampille plus récente que celle du paquet sont supprimées des *paging cache* et *routing cache*. Après les modifications des caches, le paquet de contrôle est expédié au voisin *uplink*.

▪ **Routage downlink :** Un paquet arrivant à un nœud du voisin *uplink* est supposé adressé à l'hôte mobile. Le nœud vérifie d'abord si l'adresse IP de destination a une correspondance valide dans sa *routing cache*. Si une telle correspondance existe, le paquet est expédié au voisin *downlink* désigné par la correspondance.

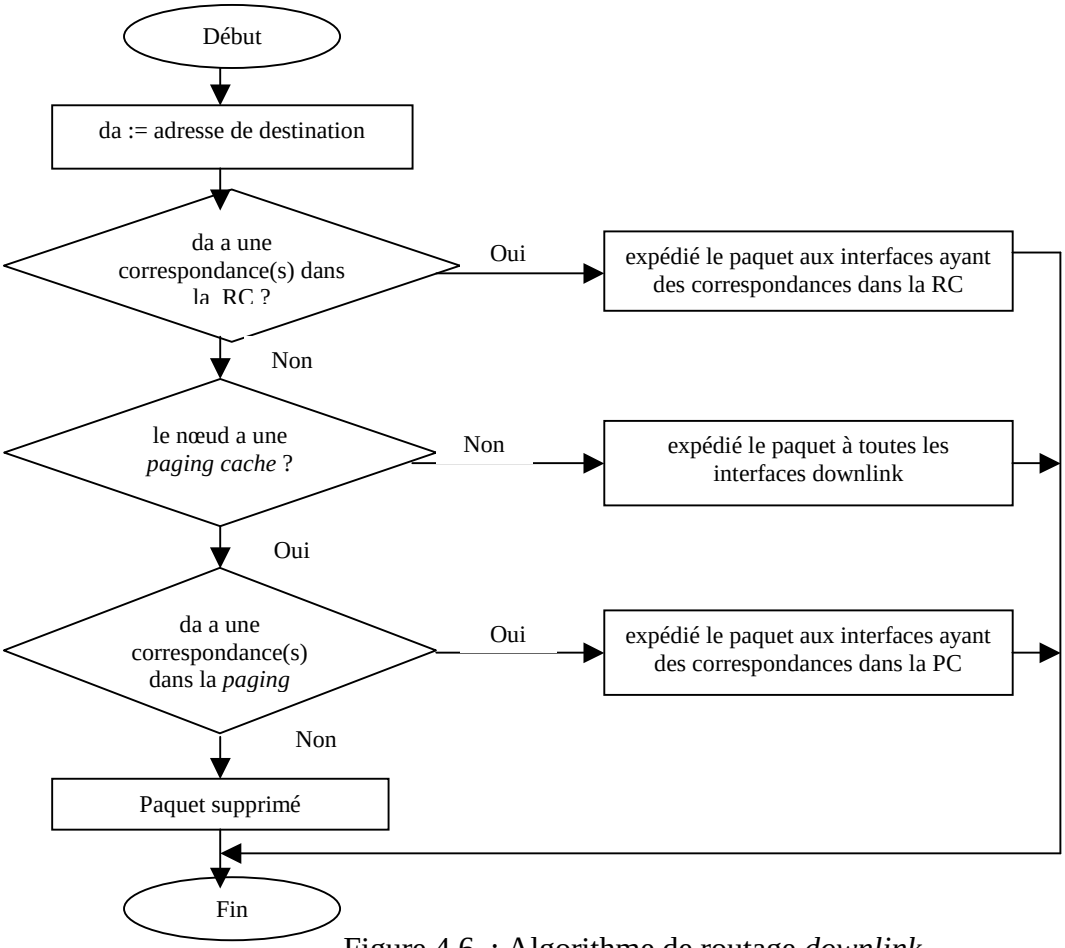


Figure 4.6 : Algorithme de routage *downlink*

Si la *routing cache* ne contient aucune correspondance pour l'adresse IP de destination et le nœud n'a pas de *paging cache*, alors le paquet est diffusé sur toutes les interfaces du nœud excepté l'interface du voisin *uplink*. Si le nœud a une *paging cache* et il existe une correspondance pour l'adresse IP de destination, alors le paquet est expédié au voisin désigné par cette correspondance. Si le nœud a une *paging cache*, mais il n'y a aucune correspondance pour l'adresse IP de destination, alors le paquet est supprimé[VAL99a].

4.4.3 La passerelle Cellular IP

La passerelle Cellular IP peut être logiquement divisée en trois modules : un nœud Cellular IP standard (*regular Cellular IP node*), un filtre de paquet (*Gateway Packet Filter*) et un contrôleur (*Gateway Controlle*) (Figure 4.7). Les paquets *uplink* mettent à jour la *routing cache*

et/ou la *paging cache* du nœud Cellular IP et sont expédiés vers le filtre de la passerelle. Ce filtre relève l'adresse IP de destination. Si cette adresse est l'adresse de la passerelle, le paquet est expédié au contrôleur de la passerelle. La plupart de ces paquets sont des paquets de contrôle avec le champ control information vide et sont immédiatement supprimés. Si le paquet comporte le champ control information non vide, par exemple une requête d'enregistrement, il est interprété et traité par le contrôleur de la passerelle.

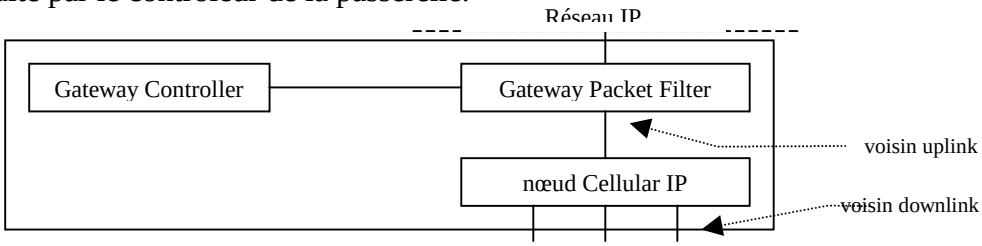


Figure 4.7 : Une vue schématique d'une passerelle Cellular IP

Si l'adresse de destination n'est pas l'adresse de la passerelle, le paquet est expédié vers l'Internet. (Ceci signifie qu'un paquet envoyé par un hôte mobile à un autre hôte mobile dans le même réseau Cellular IP passe par le *home agent* de destination. Cependant, ce n'est pas le cas si l'optimisation de la route est utilisée. Pour fonctionner efficacement même sans optimisation de mobile IP, le filtre de paquets de la passerelle peut également contrôler si l'adresse de destination d'un paquet *uplink* a une correspondance valide dans l'une des caches de la passerelle. Si la correspondance existe, le paquet "est retourné en arrière" et est traité comme un paquet *downlink*.)

Les paquets arrivant de l'Internet (utilisant Mobile IP) et s'adressant aux hôtes mobiles sont décapsulés et expédiés au *Cellular IP node block*. Les paquets arrivés n'utilisant pas Mobile IP sont supposés envoyés aux hôtes mobiles dont le *home network* est le réseau Cellular IP. Si aucun enregistrement du mobile ne prouve qu'il est parti, ces paquets sont expédiés au *Cellular IP node block* sans changement.

Le *CellularIP node block* traite ces paquets selon les correspondances existantes dans les *routing cache* et *paging cache* en exécutant l'algorithme de routage Cellular IP (section 4.4.2). Il est facultatif, que les nœuds Cellular IP aient une *paging cache* configurée ou pas. Cependant, il est recommandé qu'au moins le nœud Cellular IP de la passerelle ait une *routing cache* configurée. Ceci assure que les paquets adressés aux hôtes mobiles qui sont actuellement non reliés au réseau Cellular IP n'y accèdent pas, mais sont immédiatement rejetés par la passerelle si aucune correspondance de l'adresse de destination n'est trouvée dans l'une des caches. (Il peut être avantageux de générer également un message ICMP dans ce cas-ci et de l'envoyer de nouveau à l'adresse de la source du paquet.)

4.4.4 Handoff

Les hôtes mobiles sont constamment à l'écoute des balises transmises par les stations de base, et déclenchent le handoff basé sur des mesures de la puissance des signaux reçus des différentes balises. Cellular IP supporte deux types de handoff, le hard et le semi-soft handoff.

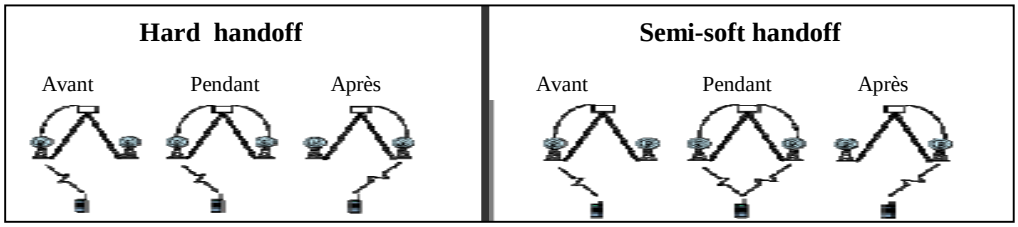


Figure 4.8 : Une simple comparaison des différents schémas de handoff de Cellular IP

Le hard handoff est une approche simple qui, au prix d'un certain taux de perte de paquets, minimise le trafic de signalisation. Le semi-soft handoff exploite le fait que les hôtes mobiles peuvent recevoir des paquets simultanément de la nouvelle et de l'ancienne station de base durant le handoff, minimisant ainsi la perte de paquets.

▪ **Hard handoff :** Dans le hard handoff, l'hôte mobile s'accorde sur la nouvelle station de base et envoie un message *route-update*. Ce message crée des correspondances dans les caches de routage des stations de base sur le chemin menant vers la passerelle, configurant ainsi la route descendante vers le nouvel emplacement du mobile. La latence associée à ce type de handoff est le temps qui s'écoule entre le début du handoff et l'arrivée du premier paquet par la nouvelle route, ce temps est égal au temps d'aller-retour entre l'hôte mobile et le nœud représentant le point de croisement entre l'ancienne et la nouvelle station de base (Figure 4.9), qui, dans le pire des cas, est la passerelle. Durant cet intervalle de temps, les paquets sur la voie descendante peuvent être perdus parce qu'ils suivent toujours l'ancien chemin. Une fois que le message *route-update* a créé une nouvelle correspondance au niveau du nœud de croisement vers la nouvelle station de base, il n'y aura plus de paquets transmis le long de l'ancien chemin.

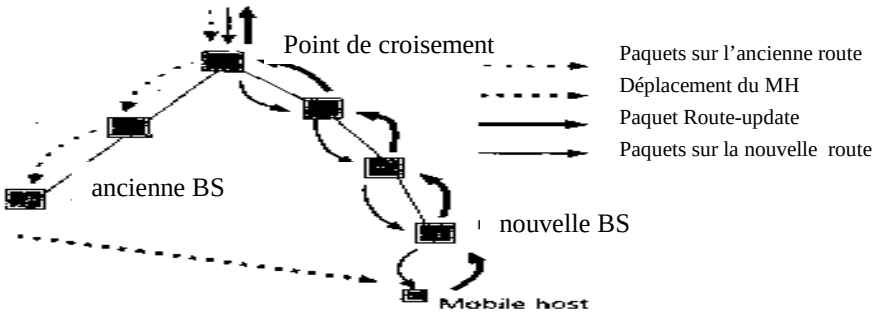


Figure 4.9 : Point de croisement entre l'ancienne et la nouvelle BS

▪ **Semi-soft handoff :** Le semi-soft handoff exploite la notion que quelques hôtes mobiles peuvent simultanément recevoir des paquets des nouvelles et anciennes stations de base pendant le handoff. Pendant le semi-soft handoff un hôte mobile peut être en contact avec l'ancienne et la nouvelle station de base et reçoit des paquets d'elles. Les paquets destinés à l'hôte mobile sont envoyés aux deux stations de base, ainsi quand l'hôte mobile se déplace par la suite à son nouvel emplacement il peut continuer à recevoir des paquets sans interruption.

Pour initialiser un semi-soft handoff, l'hôte mobile transmet un paquet *route-update* à la nouvelle station de base et continue à écouter l'ancienne station de base. Le flag S contenu dans ce paquet indique le semi-soft handoff, comme représenté sur la Figure 4.10. Ce paquet a pour but d'établir les nouvelles correspondances entre le routeur de croisement et la nouvelle station de base. Durant cette phase d'établissement de la nouvelle route, l'hôte mobile est toujours connecté à son ancienne station de base. Les paquets semi-soft *route-update* créent de nouvelles entrées dans les *routing* et *paging caches* similairement aux paquets *route update* standard. Quand le paquet semi-soft *route-update* atteint le routeur de croisement des entrées sont ajoutées aux caches au lieu de remplacer les anciennes. Le semi-soft handoff réduit ainsi la latence du handoff en mettant à jour les correspondances associées à la nouvelle station de base avant que le handoff ne se déroule effectivement.

Les paquets envoyés à l'hôte mobile sont envoyés aux deux voisins *downlink*. Quand l'hôte mobile se déplace, les paquets seront déjà envoyés à la nouvelle station de base et le handoff peut être exécuté avec une perte minimale de paquets. Après un délai dit *semi-soft delay*, l'hôte mobile effectue le handoff. Ce délai garantit qu'une fois l'hôte mobile s'accorde sur la nouvelle station de base, les paquets sur la voie descendante lui seront délivrés à la fois à travers l'ancienne et la nouvelle route. L'hôte mobile envoie un paquet *route-update* à la nouvelle station

de base avec le bit de S mis à zéro. Ce paquet supprimera toutes les entrées dans la *routing cache* exceptées celles correspondant à la nouvelle station de base. Le semi-soft handoff est alors terminé. Si le chemin menant à la nouvelle station de base est plus long que celui menant à l'ancienne station de base ou si l'hôte mobile met un temps significatif pour s'accorder avec la nouvelle station de base, alors quelques paquets peuvent ne pas atteindre l'hôte mobile. Pour surmonter ce problème, les paquets envoyés à la nouvelle station de base peuvent être différés pendant le semi-soft handoff. De cette façon quelques paquets peuvent être délivrés deux fois à l'hôte mobile, mais dans beaucoup de cas cela résulte en une meilleure performance malgré la perte de quelques paquets.

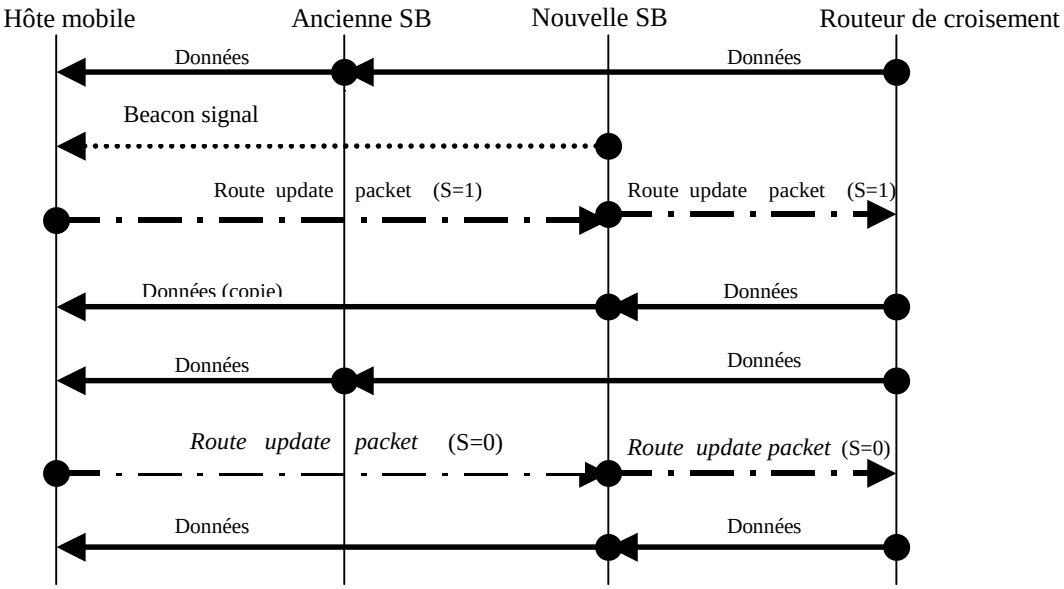


Figure 4.10 : Le trafic de signalisation correspondant au semi-soft handoff

4.4.5 Pagination

Les systèmes cellulaires utilisent la notion de connexion passive pour réduire la consommation d'énergie des hôtes mobiles inactifs. La connectivité passive est issue d'un problème bien connu en téléphonie mobile : l'économie des batteries des stations mobiles. Emettre coûte cher en terme de consommation d'énergie et les batteries des appareils portables sont de faible capacité. Pour augmenter la durée d'utilisation d'une batterie, on cherche à réduire les émissions des stations mobiles au strict minimum (essentiellement pendant la durée de la communication) [VAL99a].

Une solution adoptée dans le monde de la téléphonie mobile est d'accepter de ne pas connaître à tout moment la localisation exacte du MN mais seulement la zone dans laquelle il se trouve. Pour pouvoir le contacter, il faut donc un mécanisme dédié : le *paging*. Le réseau doit alors être partitionné en différentes zones géographiques : les *paging areas*. En temps normal, l'hôte mobile renseigne sa position à chaque fois qu'il change de station de base. Le mobile sera dit passif lorsqu'il ne renseigne sa position que lorsqu'il change de *paging area*. Les stations de base sont donc groupées en régions appelées *paging areas* (Figure 4.11).

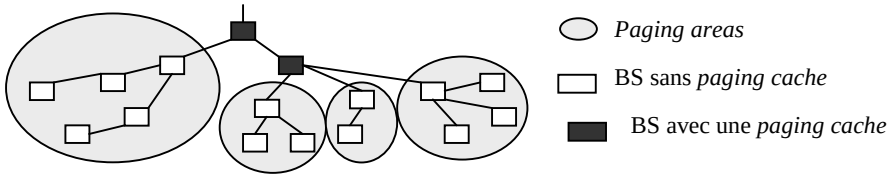


Figure 4.11: *Paging areas*

Les hôtes mobiles inactifs génèrent périodiquement des paquets de contrôle, appelés *paging-update*. Ces derniers sont envoyés à la station de base la plus proche à un intervalle de temps spécifique appelé *paging-update-time*. Ils traversent ensuite le réseau de proche en proche en direction de la passerelle. Le nœud équipé de *paging cache* contrôle les paquets *paging-update* et met à jour la *paging cache* si nécessaire. La passerelle abandonne ensuite les paquets *paging-update* et ainsi les opérations spécifiques au réseau Cellular IP sont isolées de l'Internet.

Dans la Figure 4.12, l'hôte mobile X est dans la cellule du nœud G. Les paquets *paging-update* générés par l'hôte mobile se déplacent vers la passerelle à travers les nœuds G, E, C et A. Dans cet exemple, les nœuds A et E contiennent des PCs, mais pas le nœud C. Ainsi le nœud C expédie les paquets *paging-update* à la passerelle sans enregistrer localement les informations concernant l'hôte mobile X. Le nœud A note que les paquets envoyés par X arrivent via le port de C, tant disque E note que ces paquets arrivent via le port de G.

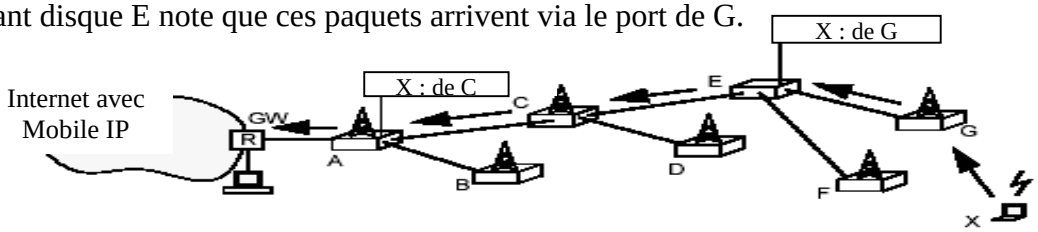


Figure 4.12: Création des correspondances dans les PCs par les paquets *paging-update*

Même quand l'hôte mobile inactif se déplace, il continue à envoyer des paquets *paging-update* à la plus proche station de base, forçant ainsi la mise à jour des correspondances des *paging caches*. Les correspondances non mises à jour seront supprimées après l'expiration d'un timer spécifique «*paging-timeout*». Si, par exemple, l'hôte mobile X se déplace vers la cellule de F, ces paquets *paging-update* seront envoyés à F (Figure 4.13).

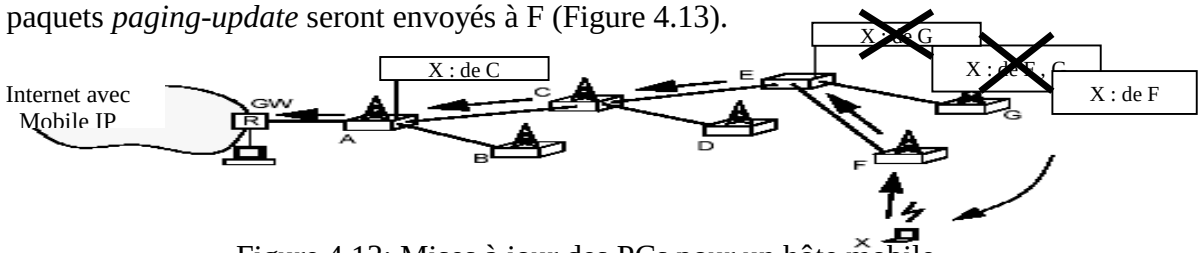


Figure 4.13: Mises à jour des PCs pour un hôte mobile

A la différence du nœud A qui n'effectue pas de mise à jour de sa PC, le nœud E crée une nouvelle correspondance pour l'hôte mobile X après la suppression de l'ancienne correspondance (expiration du timer). Durant un court moment les deux correspondances coexistent garantissant ainsi que l'hôte mobile reste toujours atteignable durant sa migration[VAL99a].

Quand un paquet IP arrive à la passerelle, adressé à l'hôte mobile pour lequel il n'existe aucune information de routage, les *paging caches* sont utilisées pour localiser l'hôte mobile. La passerelle met en file d'attente les paquets IP qui arrivent et génère un paquet de contrôle, appelé *paging packet* contenant l'identificateur de l'hôte mobile recherché. Ce dernier est routé dans le réseau en utilisant les *paging caches* en empruntant simplement la route inverse prise par le plus récent paquet *paging-update*. Si tous les nœuds ont des *paging caches*, alors une route de proche en proche complète est disponible pour aboutir à l'hôte mobile. Si quelques nœuds n'ont pas de *paging cache*, alors ils diffusent le *paging packet* sur tous leurs ports de sortie.

Dans la Figure 4.14, A vérifie sa cache afin de router les *paging packets* et découvre que les paquets *paging-update* de X sont arrivés récemment via le port du nœud C. Ainsi A envoie le *paging packet* à C, qui ne trouve aucune information concernant l'hôte mobile X, envoie à son tour le paquet vers deux directions (E , D). Le *paging packet* envoyé à D est abandonné car D est conscient que l'hôte mobile X ne se trouve pas dans sa cellule. Celui envoyé à E lui permet

de chercher dans sa cache une information concernant le nœud X et découvre que X a envoyé des paquets à travers F, ainsi E transmet le paquet à F et donc à X.

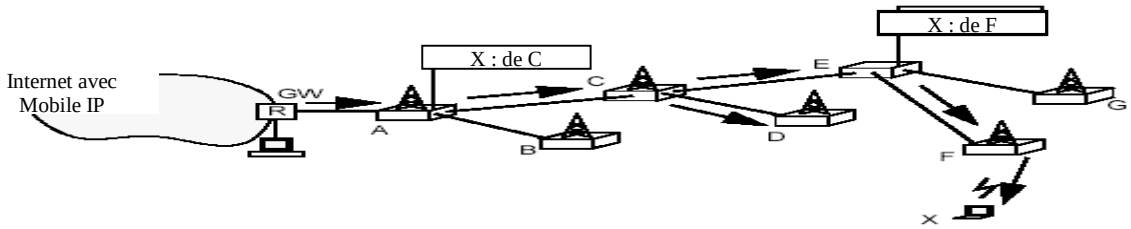


Figure 4.14: *Paging packet* est routé vers l'hôte mobile en utilisant les PCs

Dès la réception du *paging packet*, le mobile crée un paquet de contrôle appelé *route-update* et l'envoie à sa station de base F. Similairement aux paquets *paging-update*, les paquets *route-update* sont envoyés à la passerelle et ils créent des correspondances du mobile dans les *routing caches* sur leurs chemins. Quand les paquets *route-update* atteignent la passerelle, tous les RCs sur leurs chemins sont configurées et les paquets IP mis en attente dans la passerelle peuvent être délivrés à l'hôte mobile. Ce processus de recherche retarde la livraison du premier paquet IP mais une fois le chemin tracé, les paquets suivants utilisent ce chemin sans itérer la recherche. Le Tableau 4.1 clarifie la dualité qui existe entre les *paging caches* et les *routing caches*.

	Paging caches	***Routing caches***
rafraîchit par	tous les paquets *uplink* (données, *paging-update*, *route-update*)	données et paquet *route-update*
mises à jour par	tous les paquets de mise à jour (*paging-update*, *route-update*)	paquet *route-update*
mises à jour quand	déplacement vers une nouvelle *paging-area*, ou après expiration du *paging-update-time*	déplacement vers une nouvelle cellule, ou après expiration du *route-update-time*
portée	MHs actifs et passifs	MHs actifs
objectif	router les paquets *downlink* s'il n'existe pas de correspondances dans les *routing-caches*	router les paquets *downlink*

Tableau 4.1 : Résumé des opérations des *paging caches* et *routing caches*

4.4.6 Paramètres du protocole

Quelques précisions sont définies dans [CAM99] concernant Cellular IP. Par exemple, la fréquence typique d'apparition des messages *route-update* est de 3sec, et la valeur du route-timeout est de 9sec, ce qui montre bien une gestion assez rapide de la mobilité (Tableau 4.2). Bien sûr, ces valeurs peuvent être configurées au besoin. Les formats des messages utilisés par Cellular IP sont présentés dans l'annexe 3.

Timer	**Signification**	**Valeur**
route-update-time	Intervalle de temps maximal d'arrivée des paquets de mise à jour de la *routing cache*	3 sec
route-timeout	Validité des correspondances de la *routing cache*	9 sec
paging-update-time	Intervalle de temps maximal d'arrivée des paquets de mise à jour de la *paging cache*	3 min
paging-timeout	Validité des correspondances de la *paging cache*	9 min

Tableau 4.2 : Les valeurs typiques des différents timers

4.4.7 Sécurité

Cellular IP a été conçu pour supporter un handoff sécurisé et transparent. Les systèmes mobiles sont exposés à un certain nombre de problèmes de sécurité qui n'existent pas dans les systèmes fixes. Dans un réseau fixe, le préfixe d'un sous-réseau est habituellement configuré manuellement, et l'emplacement du sous-réseau est communiqué entre les routeurs qui utilisent des protocoles sécurisés. Ceci rend la personnification des utilisateurs difficile dans les systèmes fixes. D'autre part, les hôtes mobiles doivent mettre à jour leur emplacement tout en se déplaçant. Ces messages de localisation rendent la personnification possible à moins qu'ils soient correctement sécurisés. Les réseaux d'accès sans fil rencontrent des problèmes de sécurité parce que les paquets peuvent être trifouillés au-dessus de l'interface sans fil. Cellular IP doit faire face à l'usurpation d'identité et aux attaques de trifouillage. [AND00]

Cellular IP aborde ces problèmes de sécurité. D'abord, seulement les paquets authentifiés peuvent établir ou changer les correspondances des caches dans le réseau Cellular IP. En authentifiant les messages de contrôle (*route update packet* et *paging update packet*), les utilisateurs malveillants sont empêchés de capturer le trafic destiné aux hôtes mobiles. Dans les réseaux Cellular IP, seuls les paquets de contrôle sont authentifiés. Dans ce cas, les paquets de données ne sont pas authentifiés, ce qui seraient coûteux en terme de performance de transport. Les messages de contrôle établissent et mettent à jour les correspondances des caches. En revanche, les paquets de données peuvent uniquement rafraîchir ces correspondances. Les hôtes mobiles actifs transmettent des paquets *route-update* pendant le handoff pour créer une nouvelle chaîne de correspondances dans les caches qui dirigera les nouveaux paquets de données vers le nouveau point d'attachement.

L'hôte mobile peut quitter le réseau Cellular IP à tout moment sans avertissement. Les correspondances associées à l'hôte mobile seront effacées après un expiration d'un timer. Alternativement, pour optimiser l'exécution, l'hôte mobile peut envoyer à un paquet *paging-teardown* pour effacer les correspondances de l'hôte mobile des caches.

L'hôte mobile peut identifier les réseaux Cellular IP en utilisant l'identificateur du réseau Cellular IP contenu dans les signaux « *beacon* » envoyés par les stations de base. Ce signal contient également l'adresse IP de la passerelle. Pour raison de sécurité, l'authentification et d'autres informations relatives à l'utilisateur doivent être fournies par l'hôte mobile, quand il entre pour la première fois en contact avec un réseau Cellular IP. Ces informations seront insérées dans l'entête du premier paquet *paging-update* et peuvent être répétées dans les paquets *paging-update* suivants.

Dès la réception du premier paquet *paging-update*, la passerelle exécute un contrôle d'admission. La réponse de la passerelle est envoyée à l'hôte mobile dans le paquet(s) IP standard. Si la demande était acceptée, la réponse peut également porter la configuration exigée pour des paramètres du protocole. Après la réussite de l'authentification, l'hôte mobile peut envoyer un message d'enregistrement Mobile IP à son *home agent*, indiquant l'adresse IP de la passerelle en tant que son adresse temporaire (*care-of-address*).

Chaque réseau Cellular IP a donc un identificateur secret d'une longueur arbitraire connu par les nœuds Cellular IP. L'identificateur du réseau est caché aux hôtes mobiles et aux nœuds qui se trouvent à l'extérieur du réseau Cellular IP. Dès l'achèvement de l'enregistrement initial de l'hôte mobile, la passerelle doit authentifier et peut-être autoriser l'hôte mobile. Ainsi elle acquiert l'identificateur public dé l'hôte mobile, le crypte et l'envoie à l'hôte. De cette façon l'hôte mobile et le réseau Cellular IP partagent un secret commun.

4.5 HAWAII

Handoff-Aware Wireless Access Internet Infrastructure (HAWAII) a été conçu pour gérer la micro-mobilité à l'intérieur d'un réseau visité. Le principe de HAWAII est de définir au sein des réseaux un domaine HAWAII, c'est à dire un ensemble de machines interconnectées utilisant ce protocole. L'hôte mobile continue à utiliser Mobile IP et peut ainsi se déplacer librement sans se préoccuper du réseau qui va l'accueillir. Il utilise le protocole Mobile IP standard avec NAI (*Network Access identifier*)[CAL99], l'optimisation du routage et les extensions C*hallenge/ Response*. La génération et le traitement des messages d'enregistrement Mobile IP sont séparés en deux parties entre l'hôte mobile et la station de base et entre la station de base et le *home agent* [RAM00].

Le mobile conserve la même adresse IP pendant son séjour dans le domaine HAWAII. Les routeurs utilisent des soft-states pour maintenir les routes depuis le *Domain Root Router* (DRR, routeur d'entrée du réseau) vers le mobile. L'intérêt de HAWAII est de limiter la signalisation lors des handoffs : lorsqu'un hôte mobile procède à un handoff au sein du même domaine, le *home agent* n'en est pas averti.

Contrairement à Cellular IP, HAWAII ne remplace pas IP mais s'appuie sur lui dans son fonctionnement. Chaque station du réseau doit donc pouvoir fournir les services d'un routeur IP classique plus certaines fonctionnalités de gestion de la mobilité. La gestion de la mobilité se fait de façon très similaire à Cellular IP : chaque station maintient une cache de routage qui lui permet de déterminer le traitement à appliquer aux paquets qu'elle reçoit. Le handoff est traité suivant deux mécanismes. Comme Cellular IP, HAWAII présente un support de la pagination. Dans HAWAII, les stations de bases faisant partie d'une zone de pagination sont toutes membres du même groupe IP *multicast*. Ceci permet de distribuer une requête de pagination à toutes les stations d'un même groupe en l'adressant à ce groupe.

4.5.1 Architecture du réseau

L'architecture d'un réseau HAWAII est basée sur une hiérarchie de domaines comme représenté sur la Figure 4.15. La passerelle dans chaque domaine s'appelle *domain root router* (DRR). Un domaine HAWAII comporte plusieurs routeurs et stations de base implémentant le protocole HAWAI, aussi bien que les hôtes mobiles.

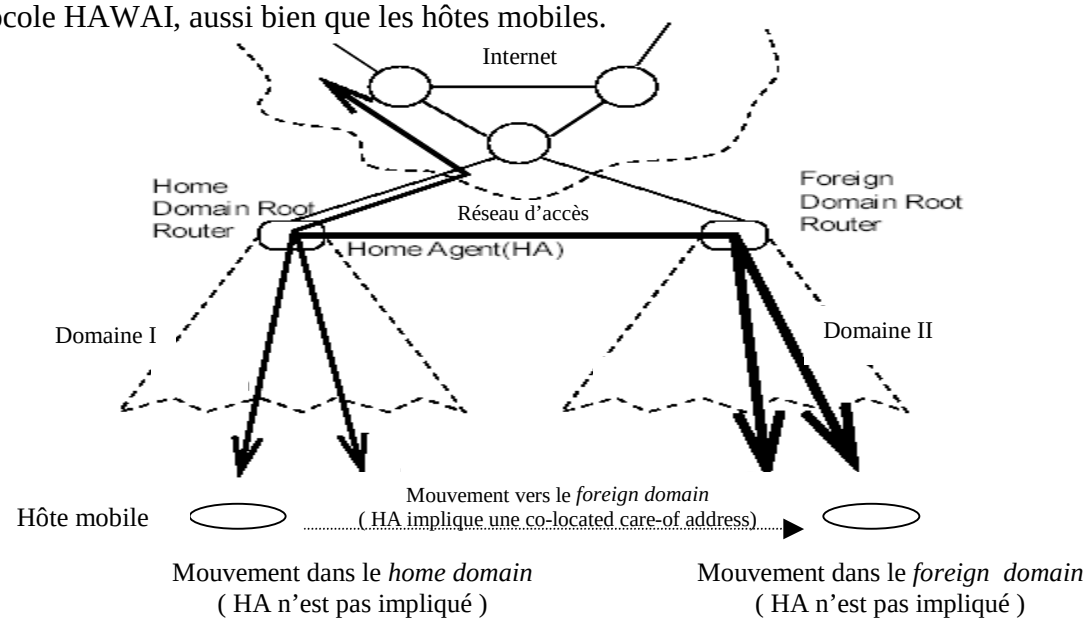


Figure 4.15 : Hiérarchie utilisant des domaines

Un hôte mobile dans un environnement HAWAII exécute le protocole Mobile IP standard avec NAI (*Network Access identifier*)[CAL99], l'optimisation du routage et les extensions

Challenge / Response. HAWAII n'effectue pas de grandes modifications au protocole Mobile IP s'exécutant sur un hôte mobile pour empêcher les hôtes mobiles d'avoir deux piles de protocoles implémentées. La génération et le traitement des messages d'enregistrement Mobile IP sont séparés en deux parties : entre l'hôte mobile et la station de base et entre la station de base et le *home agent* de l'hôte mobile. Puisque quelques messages Mobile IP sont translatés en messages HAWAII et sont traités localement dans le domaine HAWAII, alors il y aura moins de messages de mises à jour transmis au *home agent* comparé au protocole standard Mobile IP.

Chaque hôte mobile a une adresse IP et un domaine mère (*home domain*). Tout en se déplaçant dans son *home domain*, l'hôte mobile maintient son adresse IP. Les paquets destinés à l'hôte mobile atteignent le *domain root router* en se basant sur l'adresse du sous-réseau du domaine, puis sont expédiés à l'hôte mobile sur des routes établies dynamiquement. Le *home agent* n'est pas impliqué dans le transfert de données, ayant pour résultat une augmentation de fiabilité et un routage efficace. [RAM00]

Quand l'hôte mobile se déplace vers un domaine étranger (*foreign domain*), on retourne aux mécanismes traditionnels de Mobile IP. Si le *foreign domain* est également basé sur HAWAII, alors une adresse spéciale (*a co-located care-of address*) est assignée à l'hôte mobile par son *foreign domain*. Les paquets sont tunnelés vers la *care-of address* par un *home agent*. En se déplaçant dans le *foreign domain*, l'hôte mobile maintient sa *care-of address* inchangée (ainsi, le *home agent* n'est pas impliqué dans ces mouvements), et la connectivité est maintenue à l'aide des voies d'accès établies dynamiquement dans le *foreign domain*.

4.5.2 Routage

Il y a trois types de messages HAWAII d'établissement de chemin (*path setup messages*) : de mise sous tension "*path setup power-up message*", de mise à jour "*path setup update message*"et de rafraîchissement "*path setup refresh message*". Dès la mise sous tension l'hôte mobile envoie un message de demande d'enregistrement Mobile IP (*Mobile IP registration request message*) à la station de base correspondante. La station de base envoie alors un message "*path setup power-up message*" au *DRR*. Ce message ajoute des entrées correspondant à l'hôte mobile dans les tables de routages de tous les routeurs sur son chemin au *DRR*. Ce dernier envoie un accusé de réception du message reçu à la station de base qui informe finalement l'hôte mobile avec une réponse Mobile IP d'enregistrement (*a Mobile IP registration reply*).

Si un routeur constate de multiples routes menant au DRR, il peut utiliser n'importe laquelle d'entre elles, mais il doit toujours utiliser la même route pour un hôte mobile spécifique. Les routeurs utilisent des états-soft pour maintenir les routes depuis le DRR vers l'hôte mobile. La notion de "état-soft" (*soft-state*) se rapporte à l'état établi dans les routeurs qui doit être périodiquement rafraîchi par des messages de rafraîchissement (*path setup refresh messages*) qui sont envoyés indépendamment par chaque nœud du réseau et qui peuvent être agrégés; autrement, cet état sera supprimé automatiquement quand un timer lié à cet état expire.

Les routeurs, par lesquels le message *path setup* lié à un hôte mobile n'est pas passé, n'ont aucune connaissance de la localisation de ce mobile. Toutefois, un routeur qui reçoit un paquet destiné à un hôte mobile inconnu, par exemple d'un autre hôte mobile dans le domaine, utilise une interface préconfigurée par défaut qui mène au DRR. Ce paquet sera expédié dans cette direction jusqu'à ce qu'il arrive à un routeur connaissant une route menant à l'hôte mobile. Dans le pire des cas ce sera le DRR.

A Chaque hôte mobile est assigné un domaine mère (*home domain*) aussi bien qu'une adresse IP unique. Tout en se déplaçant dans le domaine mère l'hôte mobile maintient son adresse IP qui peut être assigné statiquement ou dynamiquement. Quand les paquets de données destinés à un hôte mobile spécifique arrivent au DRR, ils sont expédiés sur des routes établies dynamiquement à l'hôte mobile. Ainsi le *home agent* n'est pas impliqué. En se déplaçant vers un

domaine étranger (*foreign domain*), le *home agent* de l'hôte mobile est impliqué en utilisant les mécanismes classiques de Mobile IP. Si le *foreign domain* est également basé sur HAWAII, alors une adresse spéciale (*a co-located care-of address* «CCOA») est assignée à l'hôte mobile par son *foreign domain*. Ainsi le *home agent* tunnele les paquets à l'hôte mobile via la CCOA. En se déplaçant dans le *foreign domain*, l'hôte mobile maintient la même CCOA inchangée. Ainsi, le *home agent* n'est pas impliqué dans ces mouvements, et la connectivité est maintenue à l'aide des routes établies dynamiquement dans le *foreign domain*.

4.5.3 Handoff

Les stations de base sont supposées implémenter la fonctionnalité de routage IP. Le chemin du *domain root router* aux différents hôtes mobiles forme une topologie d'arbre virtuelle. Pour les sous-sections suivantes, un routeur de croisement est défini comme étant le routeur le plus proche de l'hôte mobile qui est à l'intersection de deux voies d'accès, une entre le DRR et l'ancienne station de base, et la seconde entre l'ancienne station de base et la nouvelle station de base. Deux schémas d'établissement de chemin utilisés pour rétablir l'état du chemin quand l'hôte mobile se déplace d'une station de base à une autre dans le même domaine seront décrits.

Considérant un hôte mobile mis sous tension dans un domaine, ceci implique que des entrées de la table de routage concernant l'hôte mobile ont été établies au niveau du *domain root router* et au niveau de tout routeur intermédiaire en direction de l'hôte mobile. Les deux schémas d'établissement de chemin considérés sont basés sur la façon dont les paquets sont délivrés à l'hôte mobile pendant un handoff : dans le premier type, les paquets sont expédiés de l'ancienne station de base vers la nouvelle, tandis que dans le second, ils sont détournés au niveau du routeur de croisement.

Les deux variantes de schémas d'établissement de chemin sont motivées par deux types de réseaux sans fil. Le schéma *Forwarding* est utilisé pour des réseaux où l'hôte mobile peut émettre / recevoir d'une seule station de base comme dans le cas d'un réseau TDMA (*Time Division Multiple Access*). Le schéma *Non-Forwarding* est utilisé pour des réseaux où l'hôte mobile peut émettre / recevoir de deux stations de base ou plus simultanément pour une courte durée, comme dans le cas d'un réseau CDMA (*Code Division Multiple Access*).

- **Schéma *Forwarding***

Dans ce schéma, les paquets sont d'abord expédiés de l'ancienne station de base à la nouvelle station de base avant d'être détournés au niveau du routeur de croisement (Figure 4.16). Les entrées des tables de routage sont montrées à côté des routeurs. Le numéro du message responsable d'établir le chemin est ajouté à l'entête de l'entrée (le numéro de message zéro indique une entrée préexistante). Les lettres dénotent les différentes interfaces.

Dans le cas d'un handoff, indiqué par une différence entre le NAI de la station de base actuelle et le NAI de l'ancienne station, une requête d'enregistrement Mobile-IP est d'abord envoyée par l'hôte mobile à la nouvelle station de base. Le message contient l'adresse de l'ancienne station de base comme étant une partie de la notification de l'ancien *foreign agent* (*Previous Foreign agent Notification Extension* « PFANE »)[PER97]. La nouvelle station de base envoie alors une mise à jour d'établissement de chemin « *path setup update message* » à l'ancienne station de base. L'ancienne station de base recherche dans sa table de routage la nouvelle station de base et détermine l'interface, interface A, ainsi que le prochain routeur à visiter, routeur 1. Elle ajoute alors une entrée dans sa table {l'adresse IP de l'hôte mobile, l'interface de sotie A }. Elle envoie ensuite le message au routeur 1 (message 3). Le routeur 1 effectue des actions semblables et transmet le message au routeur 0. Le routeur 0, le routeur de croisement dans ce cas-ci, ajoute une entrée dans sa table de routage. Par conséquent, les nouveaux paquets sont détournés au niveau du routeur de croisement. Il expédie alors le message vers la nouvelle station de base. Par la suite le message atteint la nouvelle station de base

(message 6). La nouvelle station de base change son entrée dans sa table de routage et envoie une réponse d'enregistrement Mobile-IP à l'hôte mobile (message 7).

Seules les nouvelles et les anciennes stations de base, et les routeurs reliés à elles, sont impliqués dans le traitement du message d'établissement de chemin. En outre, seuls les routeurs sur le chemin entre la nouvelle station de base et le DRR recevront périodiquement des messages de rafraîchissement. Par conséquent, les entrées des tables de routage du le routeur 1 et de l'ancienne station de base (qui ne sont pas le long de ce chemin) seront supprimées après l'expiration d'un timer spécifique, alors que les entrées des tables de routages des routeurs 0 et 2, et la nouvelle station de base seront rafraîchies.

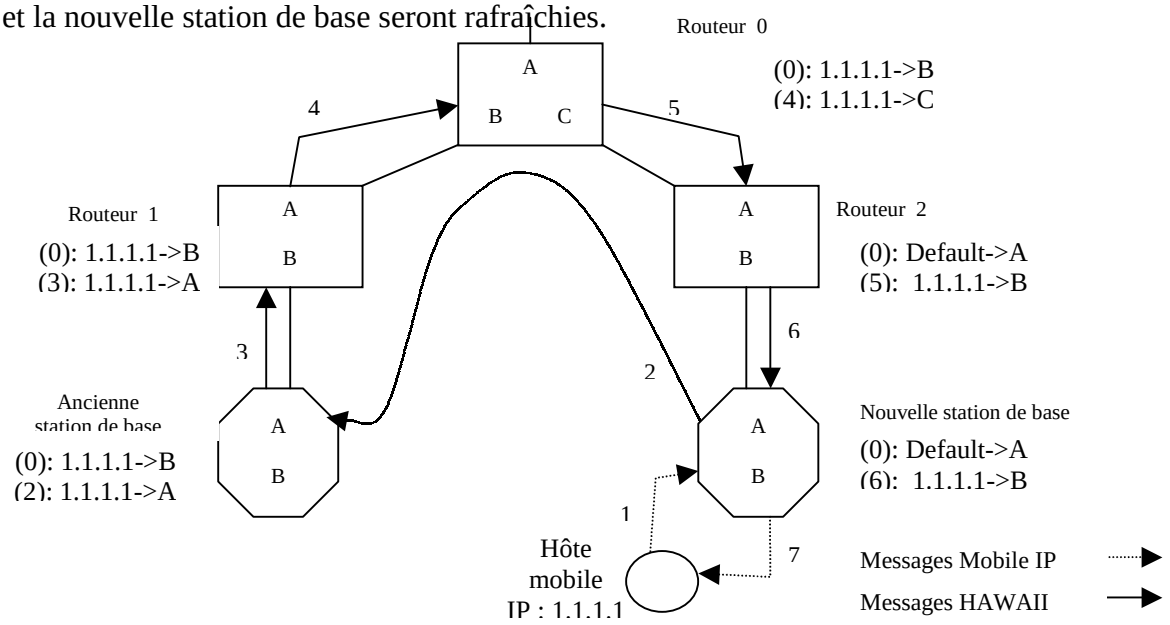


Figure 4.16 : Schéma *Forwarding*

- **Schéma *Non-Forwarding***

Dans ce schéma, les paquets de données sont détournés au niveau du routeur de croisement vers la nouvelle station de base à partir de l'instant où le message de mise à jour de la route «*path setup update message*» passe pour la première fois par le routeur de croisement, ayant pour résultat aucune expédition des paquets de l'ancienne station de base (Figure 4.17).

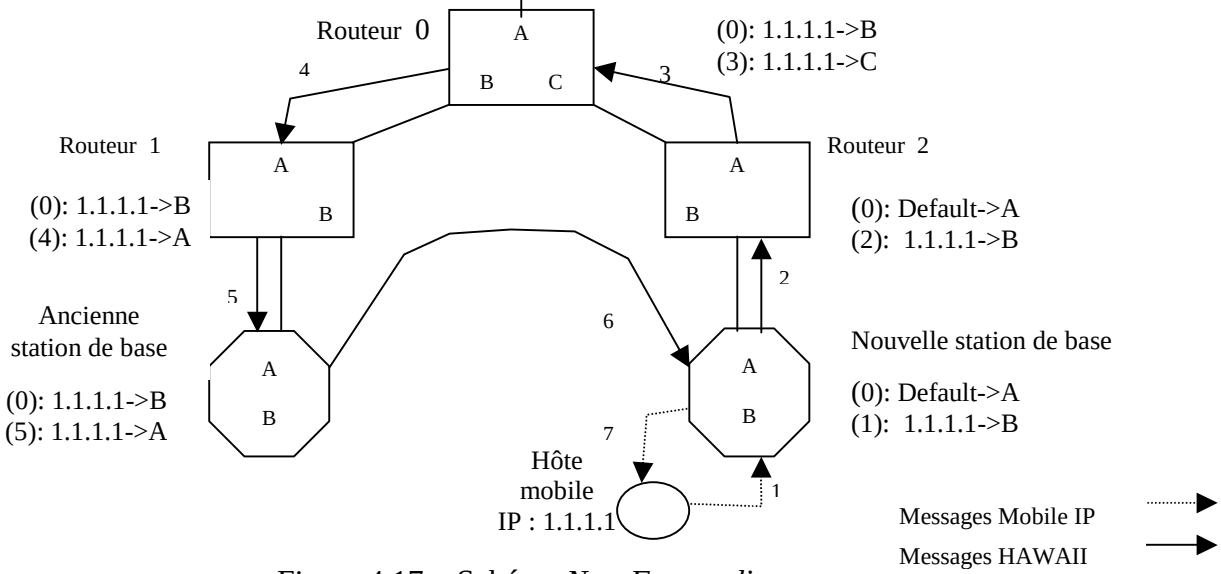


Figure 4.17 : Schéma *Non-Forwarding*

Quand la nouvelle station de base reçoit un message d'enregistrement Mobile-IP avec le champ PFANE, elle ajoute une entrée dans sa table de routage pour l'adresse IP de l'hôte mobile

avec l'interface de sortie qui correspond à l'interface sur laquelle elle a reçu ce message. Elle recherche alors dans la table de routage l'adresse de l'ancienne station de base (identifiée en utilisant le champ PFANE dans le message d'enregistrement) et détermine le prochain routeur, le routeur 2. La nouvelle station de base envoie alors le message d'établissement de chemin au routeur 2 (message 2). Ce routeur exécute des actions semblables et envoie le message au routeur 0. Au niveau du routeur 0 (dans ce cas c'est le routeur de croisement) des entrées sont ajoutées. Par conséquent, les nouveaux paquets sont détournés au niveau du routeur de croisement directement à l'hôte mobile. Par la suite le message atteint l'ancienne station de base(message 5). L'ancienne station de base change son entrée dans sa table de routage et envoie un accusé de réception du message d'établissement de chemin à la nouvelle station de base qui envoie à son tour une réponse d'enregistrement Mobile-IP à l'hôte mobile (message 7).

4.5.4 Pagination

Informer le réseau de chaque handoff sans se soucier, que l'hôte mobile soit en activité ou pas consomme beaucoup d'énergie de la batterie. Puisque dans un environnement mobile ceci n'est pas souhaitable, l'hôte mobile peut basculé d'un état actif à un état inactif dans lequel il ne doit pas informer le réseau sur chaque handoff. Dans ce cas-ci le réseau ne doit pas garder l'information exacte de la localisation de ces hôtes mobiles, mais seulement des informations sur la localisation approximative. Typiquement un domaine inclut un couple de zones de pagination (*paging areas*) chacune regroupant plusieurs stations de base. HAWAII n'exige pas une définition spécifique d'une zone de pagination. Il définit des zones hiérarchiques aussi bien que des zones fixes ou même des zones de pagination personnalisées.

Les hôtes mobiles inactifs doivent seulement informer le réseau des déplacements d'une zone de pagination à une autre et pas des handoffs entre les stations de base. Quand un paquet arrive à un hôte mobile inactif, l'hôte mobile bascule à l'état actif immédiatement. L'utilisation du support de pagination de HAWAII nécessite la présence de la fonctionnalité de pagination de la couche *link-layer* sur le lien sans fil qui signifie que l'hôte mobile peut identifier sa zone de pagination et détecter des demandes de pagination. Une solution typique pour identifier la zone de pagination est l'envoi périodique des signaux « *beacon* » par les stations de base incluant l'identité de la zone de pagination, ainsi un hôte mobile en écoute peut facilement détecter un changement.

Le réseau doit maintenir l'information de pagination pour chaque hôte mobile et doit délivrer les requêtes de pagination de ces hôtes aux stations de base. Pour réaliser ceci, les demandes de pagination doivent être fournir à chaque station de base dans la zone de pagination en utilisant un message *unicast* pour chacune. Puisque ce serait un gaspillage de la largeur de la bande passante, HAWAII se base sur le protocole de routage IP *multicast*. A chaque zone de pagination est assignée une adresse *multicast* de groupe. Toutes les stations de base faisant partie d'une zone de pagination sont toutes membres du même groupe IP *multicast*. Ceci permet de distribuer une requête de pagination à toutes les stations d'un même groupe en l'adressant à ce groupe.

4.5.5 Détails protocolaires

Dans cette section, les détails protocolaires des schémas d'établissement de chemin (*HAWAII path setup schemes*) sont décrits. Le traitement au niveau de l'hôte mobile est présenté et ainsi que le traitement au niveau des stations de base et des routeurs sont présentés[RAM00]. Les formats des messages de mise à jour d'établissement de chemin (*path setup update message*) et les messages de rafraîchissement (*path setup refresh message*) sont décrits dans l'annexe 3.

- **Traitement effectué par l'hôte mobile :**

L'hôte mobile envoie des messages d'enregistrement Mobile IP avec diverses extensions, puisqu'il se rend seulement compte du protocole Mobile IP et pas de HAWAII. Afin d'assurer une mobilité transparente entre les domaines HAWAII et Mobile IP des informations doivent être présentes au niveau des hôtes mobiles et des stations de base[RAM00].

Sachant que HAWAII divise le réseau d'accès en domaines, des identificateurs d'accès de réseau (NAI, *Network Access Identifiers*) [CAL99] sont utilisés pour identifier les différents domaines HAWAII. En outre, chaque hôte mobile est associé à un domaine mère et le Mobile IP *home agent* est impliqué seulement quand l'hôte mobile visite un domaine étranger. Cependant, même quand l'hôte mobile se déplace dans le domaine mère, il doit envoyer des Mobile IP enregistrements à la station de base pour chaque handoff de sorte que les entrées dans les tables de routage concernant l'hôte mobile soient rétablies localement. Ceci est accompli comme suit.

- **Algorithme 1 : Traitement effectué par l'hôte mobile**

```
1. Si les NAIs de l'ancienne BS et de la nouvelle BS sont identiques /*mouvement intra-domaine */
    si les NAIs de l'hôte mobile et de la nouvelle BS sont identiques
     /* HAWAII home domain */
1.1.1  Envoyer une requête Mobile IP d'enregistrement à la nouvelle BS en utilisant la COA
         annoncée.
      sinon  /* HAWAII foreign domain */
1.1.2  Envoyer une requête Mobile IP d'enregistrement à la nouvelle BS en utilisant la CCOA
           précédente.
      finsi
   sinon  /* mouvement inter-domaine */
1.1  Acquérir une CCOA en utilisant DHCP
1.2  Envoyer une requête Mobile IP d'enregistrement à la nouvelle BS en utilisant une nouvelle
     CCOA.
   Finsi
```

Chaque hôte mobile est configuré (statiquement ou dynamiquement) avec un NAI (*Network Access identifier*). Chaque fois que l'hôte mobile détecte un changement de station de base il doit émettre un message Mobile IP d'enregistrement à la nouvelle station de base. Ces enregistrements sont utilisés pour déclencher les schémas d'établissement de chemin à l'intérieur du domaine.

Quand l'hôte mobile est dans un domaine étranger HAWAII, il doit acquérir une *co-located care-of address*. Le NAI annoncé par la station de base et celui de l'hôte mobile sont comparés pour distinguer si l'hôte mobile est dans son domaine mère HAWAII ou s'il est dans un domaine étranger HAWAII. Si les deux identificateurs de réseau sont identiques, l'hôte mobile doit s'enregistrer auprès de la station de base en utilisant la COA (*non co-located*); autrement l'hôte mobile doit s'enregistrer auprès de la station de base en utilisant la *co-located COA* (CCOA).

Que l'hôte mobile utilise une *co-located* COA ou pas, la requête d'enregistrement doit inclure une extension de notification du *foreign agent* précédent (*Previous-Foreign agent Notification extension* « PFANE ») à moins que ce soit le premier enregistrement après la mise sous tension. La requête d'enregistrement doit également inclure toutes les extensions obligatoires définies dans [PER96a] (*the mobile-foreign authentication extension, the mobile-challenge-response extension* [PER99] et NAI *extension*).

En outre, l'hôte mobile doit être préparé pour recevoir des réponses d'enregistrement générées par la station de base sans la participation du HA, ainsi l'extension *mobile-home authentication* n'est pas incluse. Néanmoins, de telles réponses d'enregistrement doivent inclure une extension *mobile-foreign authentication* valide.

- **Traitements effectués par les stations de base et les routeurs**

Les traitements des messages HAWAII effectués par les stations de base et les routeurs sont détaillés dans cette section. Alors que les routeurs traitent seulement les messages HAWAII, les stations de base ont la responsabilité supplémentaire de mettre en application la fonctionnalité de Mobile-IP *foreign agent* (sans la fonction de décapsulation) et de produire des messages HAWAII pour effectuer des traitements dans le domaine.

Les stations de base émettent périodiquement des messages "*agent advertisement*", et répondent aux messages "*agent-solicitation*". Les messages *agent advertisement* doivent inclure l'extension *foreign-agent-challenge* [PER99] et le NAI du domaine administratif auquel la station de base appartient. La station de base authentifie la requête de l'hôte mobile en utilisant les mécanismes définis dans [PER99] en plus des mécanismes d'authentification de Mobile IP définis dans [PER96a].

Dès la réception d'une requête d'enregistrement avec une authentification *mobile-foreign* valide et une authentification *challenge-response* valide, la station de base doit vérifier si le NAI de l'hôte mobile présent dans la requête d'enregistrement s'apparie avec le NAI du domaine auquel elle appartient. Si cela est vérifié, la station de base rejette toute requête d'enregistrement ayant une *co-located* COA. Si la requête est valide, la station de base génère des messages HAWAII de mise sous tension ou des messages de mise à jour de handoff en fonction de la présence ou non du champ PFANE. La station de base est également responsable de l'enregistrement avec le *home agent* quand cela est nécessaire (voir algorithme 2 et 4).

Les algorithmes de traitement des messages de mise sous tension (*power up update messages*) sont détaillés dans les algorithmes 2 et 3. Chaque nœud ajoute une entrée concernant l'hôte mobile dans sa table de routage et transmet le message au prochain routeur le long de la route par défaut menant au DRR. Quand le message atteint le DRR, un accusé de réception est envoyé à la station de base qui envoie une réponse d'enregistrement à l'hôte mobile.

- **Algorithme 2: Traitement des mises sous tension au niveau des stations de base**

1. Dès la réception d'un message d'enregistrement d'un nouvel hôte mobile sur l'interface A
   (Le PFANE n'est pas présent puisque c'est le premier enregistrement)
2. si le NAI de l'hôte mobile s'apparie avec le NAI du domaine
   / * ce domaine est le *home domain* de l'hôte mobile * /
2.1 authentification du message: si échec, s'interrompre avec une réponse négative
2.2 ajouter/mettre à jour l'entrée { MH IP ADDRESS → Interface A}, mettre à jour le timer
2.3 envoyer des messages *HAWAII Power up update* au voisin ascendant le long d'une des routes par défaut
   sinon / * ce domaine est le *foreign domain* de l'hôte mobile */
2.4 envoyer le message au *home agent* de l'hôte mobile
2.5 si l'enregistrement est reçu par le *home agent*, exécuter 2.2
   finsi
3. si l'ACK HAWAII est reçu, envoyer une réponse d'accord d'enregistrement avec une extension *mobile-foreign authentication*

- **Algorithme 3 : Traitement des mises sous tension au niveau des routeurs**

1. Dès la réception d'un message *Power up update* de l'hôte mobile sur l'interface A
   le message contient MH IP ADDRESS, METRIC/*distance au MH*/, TIMESTAMP
2. ajouter/mettre à jour l'entrée {MH IP ADDRESS → Interface A}, mettre à jour le timer
3. si je suis le *Domain Root Router*
3.1 envoyer un accusé de réception à la station de base
   sinon
3.2 mettre à jour METRIC et envoyer la mise à jour au voisin ascendant le long d'une des routes par défaut
   finsi

Les algorithmes de traitement des messages de mise à jour pendant les handoffs avec un schéma *Forwarding* et avec le schéma *Non-Forwarding* sont détaillés dans l'algorithme 4(a) et l'algorithme 4(b) respectivement. Le traitement d'un message de mise à jour au niveau d'un routeur est assez simple: dès la réception du message, l'entrée concernant l'hôte mobile dans la table de routage est modifiée et le message de mise à jour est envoyé à la nouvelle ou l'ancienne station de base en fonction du schéma utilisé (*Forwarding* ou *Non-Forwarding*).

- **Algorithme 4(a) : Traitement des messages de mise à jour pendant les handoffs *Forwarding***

```
1. si un message d'enregistrement avec l'extension PFANE est reçu,
      si le NAI de l'hôte mobile s'apparie avec le NAI du domaine /* intra-domaine */
1.1   envoyer un message de mise à jour à l'ancienne station de base.
      sinon   /* inter-domaine */
1.2   envoyer une requête d'enregistrement au home agent
1.3   si l'enregistrement est accepté par le home agent, exécuter 1.1
      finsi
    finsi
2. Dès la réception d'un message de mise à jour sur l'interface A
      Message contient MH IP ADDRESS, OLD BS ADDRESS, TIMESTAMP
3. si NEW BS ADDRESS s'apparie avec une des adresses des interfaces locales alors
3.1 soit B l'interface locale
  sinon
3.2 rechercher dans la table de routage la NEW BS ADDRESS et déterminer le prochain
      routeur et l'interface de sortie l'interface B
    finsi
4. si TIMESTAMP est plus récent ou METRIC est plus petit pour le même TIMESTAMP alors
      ajouter/mettre à jour l'entrée {MH IP ADDRESS -> Interface B}, mettre à jour le timer
  finsi
5. si NEW BS ADDRESS s'apparie avec une des adresses des interfaces locales alors
5.1 mettre à jour Mobile-IP lifetime et générer une réponse d'enregistrement au MH
    sinon
5.2 mettre à jour METRIC et envoyer message au prochain routeur dans l'étape 3.2
  fins
```

- **Algorithme 4(b) : Traitement des messages de mise à jour pendant les handoffs *Non-Forwarding***

```
1.  si un message d'enregistrement avec l'extension PFANE est reçu,
      si le NAI de l'hôte mobile s'apparie avec le NAI du domaine /* intra-domaine */
1.1     obtenir MH IP ADDRESS, OLD BS ADDRESS, TIMESTAMP
      sinon  /* inter-domain */
1.2     envoyer une requête d'enregistrement au home agent
1.3     si l'enregistrement est accepté par le home agent, exécuter 1.1
      finsi
       aller à l'étape 3.
   finsi
2. Dès la réception d'un message de mise à jour d'un voisin sur l'interface A
    message contient MH IP ADDRESS, OLD BS ADDRESS, TIMESTAMP
3. si TIMESTAMP est plus récent ou METRIC est plus petit pour le même TIMESTAMP alors
    ajouter/mettre à jour l'entrée {MH IP ADDRESS -> Interface A}, mettre à jour le timer
  finsi
4. si OLD BS ADDRESS s'apparie avec une des adresses des interfaces locales alors
4.1 générer un accusé de réception pour la NEW BS
```

```
   sinon
4.2  rechercher dans la table de routage le prochain routeur pour OLD BS ADDRESS
        mettre à jour METRIC et envoyer/générer message au prochain routeur
   finsi
5. si ACK  HAWAII est reçu alors
      mettre à jour Mobile-IP lifetime et générer une réponse d'enregistrement au MH
   finsi
```

Les messages de rafraîchissement des états-soft sont envoyés indépendamment par chacun des nœuds de proche en proche. L'hôte mobile envoie périodiquement des renouvellements de message d'enregistrement Mobile-IP à la station de base. La station de base est responsable de maintenir la demande d'enregistrement du mobile avec son *home agent*, en générant des demandes d'enregistrement au nom de l'hôte mobile. De telles substitutions de demandes [CAL98] ne contiennent pas une extension valide *mobile-home authentication*, mais doivent contenir une extension *foreign-home authentication* valide. L'algorithme de traitement d'un message de rafraîchissement est l'algorithme 5. Les stations de base et les routeurs envoient périodiquement des messages HAWAII de rafraîchissement à leurs routeurs ascendants (en se basant sur leurs routes par défaut menant au DRR). Quand le message de rafraîchissement est reçu, le temporisateur correspondant à l'entrée rafraîchie est mis à jour.

➢ **Algorithme 5 : Traitement du message de rafraîchissement (les deux schémas)**

```
1. Dès la réception d'un message de rafraîchissement d'un voisin sur l'interface A
      le message contient plusieurs tuples de {MH IP ADDRESS, TIMESTAMP}
2. pour chaque tuple faire
      si une entrée existe pour  MH IP ADDRESS
        si TIMESTAMP est supérieur ou égal au timestamp de l'entrée
          si l'entrée a déjà une interface : Interface A
2.1           réinitialiser le  timer de l'entrée
          sinon si METRIC n'est pas plus grand
            /*échec de changement d'interface, pas de propagation */
2.2           mettre à jour l'entrée {MH IP ADDRESS → Interface A}, mettre à jour le timer
          finsi
        finsi
      sinon     /*il n' existe aucune entrée pour  MH IP ADDRESS  propagation */
2.3    ajouter l'entrée {MH IP ADDRESS → Interface A}, mettre à jour le timer
2.4    envoyer la mise à jour en empruntant la route par défaut
      finsi
3.  envoyer périodiquement des rafraîchissements pour toutes les entrées
4.  quand la route par défaut change
      envoyer périodiquement des rafraîchissements pour toutes les entrées
```

4.5.6 Sécurité

Concernant la sécurité du protocole, il y a deux questions importantes. La première est l'authentification de l'utilisateur par l'intermédiaire du serveur DHCP pendant l'affectation d'une *co-located care-of address* qui se produit pendant une mise sous tension et pendant des déplacements inter-domaines; et la seconde est la sécurité et l'authentification des messages des protocoles Mobile IP et HAWAII. Des mécanismes tels que le protocole RADIUS [RIG97] pourraient être employés pour exécuter l'authentification au niveau du serveur DHCP.

Les hôtes mobiles doivent vérifier les réponses d'enregistrement produites par les *foreign agent*s, sans l'intervention du *home agent*; aussi, les *home agent*s doivent vérifier les

enregistrements produits par les *foreign agent*s, sans l'intervention de l'hôte mobile. Ceci présume l'existence d'une infrastructure de vérification et de gestion de clés de sessions (key-management), pour distribuer des clés de sessions provisoires à l'hôte mobile, aux *foreign agent*s et au *home agent*. En plus, la même infrastructure permet de vérifier qu'un ensemble particulier de stations de base est autorisé par le HA pour servir ses hôtes mobiles.

L'authentification des messages du protocole HAWAII n'est pas un problème difficile puisque ces messages sont générés et traités seulement par des nœuds dans un seul domaine administratif, leur authentification est facile à aborder par exemple en utilisant un champ de mot de passe[RAM99].

4.6 Tableau récapitulatif des différentes caractéristiques des deux protocoles étudiés

Protocole	**Cellular IP**		**HAWAII**	
Type de handoff	**Hard**	**Semisoft**	**Forwarding**	**Non-Forwarding**
Modèle du réseau	Réseau Cellular IP		Domaine Hawaii	
Passerelle au réseau	*Gateway* Cellular IP		*Domain root router(DRR)*	
Couche radio	TDMA	CDMA	TDMA	CDMA
Type des stations	Switch évolué avec la fonction de paging (*Routing cache "RC":* MN actif et/ou *paging cache "PC" : MN inactif*)		Routeur IP avec la fonctionnalités de mobilité (Table de routage)	
Tailles des tables (pire des cas)	Au niveau de la passerelle, une entrée pour chaque MN actuellement connecté au réseau		Similaire à Mobile IP	
Mise à jour et rafraîchissement des caches de routage	– Mise à jour par les paquets r*oute-update envoyés* par le MN à la passerelle. –Rafraîchissement par des paquets de données (*uplink*) ou les paquets *route-update.*		– Mise à jour par les messages "*Path setup power-up messages*" envoyés de la station de base au DRR ou par les messages "*Path setup update messages"* envoyés de la nouvelle BS à l'ancienne BS. – Rafraîchissement par des messages spécifiques.	
Initiateur de handoff	MN		MN	
Durant le handoff	Les hôtes mobiles peuvent recevoir des paquets d'une seule BS à la fois.	Les hôtes mobiles peuvent recevoir des paquets simultanément de la nouvelle et de l'ancienne BS.	Les paquets sont d'abord expédiés de l'ancienne BS à la nouvelle BS avant d'être détournés au niveau du routeur de croisement.	Les paquets sont détournés au niveau du routeur de croisement vers la nouvelle BS .

Support de Paging	– Les correspondances de Paging sont implantées dans les paging caches. –Les paging caches définissent des paging areas. –Les paquets Route-update and paging-update mettent à jour les entrées des paging caches. –Les paquets destinés à un MN inactif sont routés grace aux paging caches. –Diffusés à tous les voisins si un nœud ne possède pas de paging caches.	- Les correspondances de paging sont repérées dans les tables de routage. –Dynamiquement créees sur le chemin menant du MN au DRR. –Ces entrée sont mises à jour par les messages "path setup messages" and "paging update messages". –Les paging areas sont définies par des adresses de groupes "Multicast". –La requête de paging est initiée, ensuite diffusée dans la paging area.
Messages MIP	Non	oui

4.7 Conclusion

Ce chapitre a présenté quatre solutions pour gérer la micro mobilité (Cellular IP et HAWAII sont étudiés en détail). Toutes reposent sur le principe suivant : le réseau visité par un mobile se charge des déplacements locaux; le *home agent* n'est donc pas prévenu de tous les changements de localisation du mobile qu'il gère.

Ces protocoles de micro-mobilité sont conçus pour des environnements où les nœuds mobiles changent leur point d'attache à l'Internet si fréquemment que le *tunneling* de MIP s'avère insuffisant : surcharge de la signalisation, perte de paquets, livraison des données aux applications retardées. Ces retards sont directement liés au temps d'aller-retour des messages d'enregistrement. Les protocoles de micro-mobilité aspirent à gérer les déplacements locaux (à l'intérieur de domaines) des nœuds mobiles sans interagir avec MIP, c'est-à-dire cacher au reste de l'Internet les mouvements des nœuds mobiles à l'intérieur d'un domaine. Cela a l'avantage de réduire le retard et la perte des paquets pendant un déplacement et élimine l'enregistrement entre un nœud mobile et son agent mère qui peut être éloigné.

Par ailleurs, la signalisation engendrée par la gestion de la mobilité s'accroît avec le nombre d'utilisateurs. Dans les réseaux cellulaires, l'enregistrement et les techniques de pagination sont employés pour réduire au maximum la signalisation et optimiser les performances de la gestion de la mobilité. Une caractéristique importante des protocoles de micro-mobilité 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 impérative puisqu'il permet de réduire la charge des réseaux.

Chapitre 5 Etudes réalisées

5.1 Introduction

Deux types d'études ont été réalisés concernant la micro mobilité : Certaines comparent uniquement les différents protocoles de micro mobilité et d'autres évaluent les performances des protocoles TCP et UDP sous les protocoles de micro mobilité. Dans ce chapitre, on étale une étude réalisée par les auteurs de [REI01] et [BON01] qui compare les protocoles de la micro-mobilité. Ensuite on présentera les résultats de simulation d'une autre étude qui évalue les performances de TCP et UDP sous les protocoles de la micro mobilité.

5.2 Comparaison des protocoles de micro mobilité

Dans [REI01] et [BON01] les auteurs présentent une comparaison détaillée de quatre protocoles existants de micro-mobilité dans IP. Cette comparaison est réalisée dans un cadre général permettant, non pas de s'attarder sur les défauts et qualités de tel ou tel protocole, mais de tirer des conclusions d'ensemble pour l'évaluation des propriétés de ceux-ci. Dans cette optique, ils tirent parti d'un choix de protocoles aux propriétés variées pour couvrir un large spectre de possibilités. Ils examinent et comparent Cellular IP, HAWAII, TeleMIP et EMA. Dans ce qui suit, on résume les résultats de leur comparaison en se limitant uniquement aux protocoles Cellular IP et HAWAII. Les différents protocoles sont comparés effectivement en utilisant différents critères:

- **Handoff** : Trois caractéristiques essentielles du mécanisme de handoff sont examinées :

– Temps de latence : le temps nécessaire pour que le réseau soit stabilisé après un handoff,
– La perte de paquet : le nombre de paquets potentiellement perdus à cause du handoff,
– La signalisation : le nombre de messages de signalisation qui doivent circuler dans le réseau pour gérer chaque handoff.

Pour cette comparaison, ils utilisent un modèle de réseau simplifié (Figure 5.1). Ils supposent que n_{pass} est le nombre moyen de sauts entre un MN et la passerelle du réseau d'accès. Le temps pour parcourir cette distance est de t_{pass} msec. De la même façon, n_{anc} représente le nombre moyen de sauts entre le MN et la station de base avec laquelle il était connecté avant le handoff. Ils supposent également que cette station peut être jointe en un temps moyen de t_{anc} msec. Enfin, t_{crois} représente le temps moyen pour atteindre le routeur de croisement. Ils considèrent qu'en moyenne $t_{pass} \geq t_{anc} \geq t_{crois}$. Enfin, τ est le taux d'émission moyen d'un MN durant le handoff.

➢ Dans Cellular IP, la mise à jour des tables de routage se fait via l'émission par le MN d'un paquet spécifique qui est retransmis de proche en proche jusqu'à la passerelle. Ce dernier doit alors envoyer un acquittement et le handoff ne se termine qu'au moment où le MN le reçoit. Le temps de latence de ce mécanisme est donc de $2t_{pass}$ avec n_{pass} mises à jour dans les différentes machines du réseau. Cellular IP propose deux types de handoff différents. Avec le semi-soft handoff, ils considèrent que les pertes de paquets seront nulles. D'autre part, le hard handoff génère τt_{crois} pertes de paquets en moyenne. En effet, c'est seulement à partir du moment où le paquet d'*update* atteint le routeur de croisement que le routage est de nouveau correctement réalisé.

➢ Le mécanisme de gestion du handoff dans HAWAII est constitué d'un échange de message entre les deux stations de base concernées. La latence du mécanisme sera donc $2t_{anc}$. Par contre, les pertes de paquets seront différentes suivant le schéma de handoff utilisé. Dans le cas

du schéma *forwarding*, les pertes seront de τt_{anc} car il faut que l'ancienne station de base soit atteinte par le paquet de mise à jour pour que le routage se fasse à nouveau correctement. Dans le cas du schéma *non-forwarding*, les pertes seront de τt_{crois} pour les mêmes raisons que dans le cas de du hard handoff de Cellular IP.

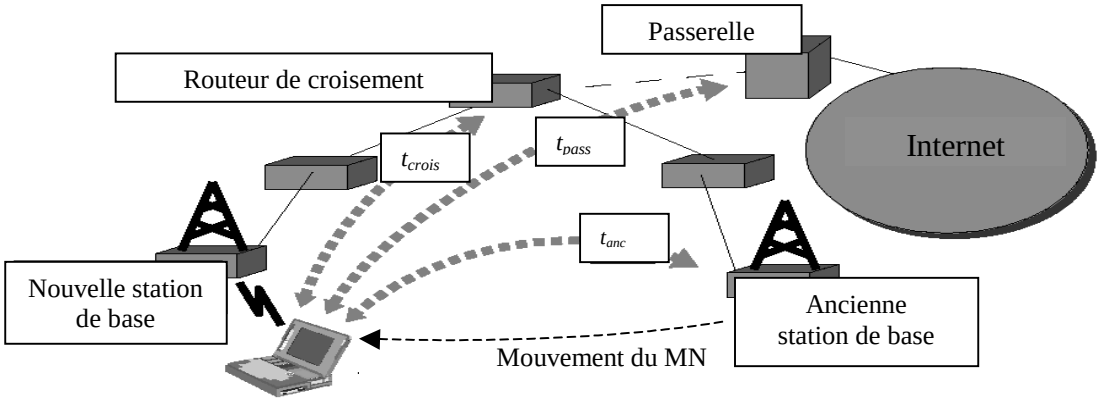


Figure 5.1 : Un modèle de réseau pour la comparaison des paramètres du handoff

Le Tableau 5.1 résume cette discussion. Selon les auteurs, il apparaît clairement qu'un compromis doit être fait et qu'aucune des propositions n'est totalement satisfaisante, réunissant latence minimale, pertes nulles et nombre d'*updates* réduit. Dans cette optique, HAWAII peut sembler offrir un bon compromis.

Protocole	**Type de handoff**	**Latence**	**Pertes**	**Nombre d'updates**
Cellular IP	Semi-soft handoff	$2t_{pass}$	0	n_1
	Hard handoff	$2t_{pass}$	τt_{crois}	n_1
HAWAII	Schéma forwarding	$2t_{anc}$	τt_{anc}	n_2
	Schéma non-forwarding	$2t_{anc}$	τt_{crois}	n_2

Tableau 5.1 : Comparaison des paramètres de gestion du handoff

- **Pagination** : La différence principale entre Cellular IP et HAWAII est située au niveau de l'algorithme de pagination :

- Dans Cellular IP, la passerelle effectue un *paging* dès qu'un paquet arrive pour un MN en mode passif. Le processus de pagination est pris en charge par la passerelle et un ensemble de machines dédiées gardant en mémoire l'information nécessaire. Ces machines ont donc toute la charge du *paging* et sont définies statiquement.

- Par contre, HAWAII définit un algorithme de répartition dynamique de la charge du *paging* sur les machines du réseau, en tendant à repousser cette charge vers les stations de base. La machine qui effectue un *paging* est choisie dynamiquement sur la base de la charge effective des stations du réseau.

- **Support du trafic interne au réseau d'accès** : La plus grande part des communications GSM aujourd'hui est constituée d'échanges de données entre utilisateurs du même réseau :

- Avec Cellular IP, tout le trafic venant du MN doit passer par la passerelle, même si le destinataire se trouve dans le même réseau. Ce système, loin d'être optimal du point de vue du routage, impose un surplus inutile de charge de travail à la passerelle et aux stations environnantes.

- HAWAII est un protocole dont le niveau de fonctionnement est au-dessus de IP, on peut donc supposer que le trafic destiné à l'intérieur du réseau sera routé directement.

- **Niveau de fonctionnement des stations** : Les réseaux GSM actuels fonctionnent avec des millions d'utilisateurs connectés simultanément et on peut considérer que les futurs réseaux mobiles devront supporter au moins une charge équivalente à celle-là. Ceci est à mettre en relation avec les problèmes de gestion des tables de routage de plus en plus grandes dans l'Internet aujourd'hui : gérer dynamiquement une table de quelques centaines de milliers d'entrées devient un véritable problème. Dans Cellular IP et HAWAII, la passerelle est une machine extrêmement chargée. Elle doit maintenir en permanence une table contenant une entrée par mobile effectivement connecté au réseau. La gestion de cette table est particulièrement difficile.

- HAWAII assume pour sa part que le réseau est composé de routeurs IP classiques capables de gérer la mobilité. Les stations doivent donc non seulement effectuer les fonctions de routage classiques mais aussi assurer leur rôle dans HAWAII; elles sont donc particulièrement chargées.
- Pour Cellular IP, les stations du réseau doivent assurer les fonctions de «*switchs* évolués », ce qui les rendent plus simples et plus légères. Cependant, les mécanismes évolués de Cellular IP comme le semi-soft handoff ou le *paging*, augmentent sensiblement les fonctionnalités devant être supportées par les stations, les rendant dans les faits plus proches de vrais routeurs que de *switchs*.

- **Robustesse et flexibilité**

Du point de vue de la flexibilité et de la robustesse, Cellular IP et HAWAII présentent un défaut important qui est leur structure en arbre hiérarchique. Dans cette configuration, la machine assumant les fonctions de la passerelle est la plus chargée du réseau puisque tous les messages de mise à jour (routage ou *paging*) lui sont retransmis. De plus, la structure d'arbre signifie que plus une station est proche de la passerelle, plus elle doit supporter une charge importante. Cette augmentation de charge est due au traitement des messages de contrôle et du routage ainsi qu'au maintien en mémoire de tables de plus en plus grandes, la passerelle devant traiter tous les messages de mise à jour et devant maintenir une entrée pour tous les mobiles présents dans le réseau. Un tel système n'est évidemment pas très flexible. De plus, il est fort probable que les futurs réseaux d'accès mobile ne présenteront pas qu'une seule passerelle, ce qui n'est pas prévu dans ces deux protocoles.

Ce type d'architecture ne favorise pas non plus la robustesse du réseau. En effet, tout le système repose sur quelques machines dédiées assurant les fonctions principales, le crash de l'une d'entre elles est donc particulièrement difficile à récupérer. Dans cette optique, HAWAII est mieux placé que Cellular IP puisque toutes les routes ne passent pas par la passerelle et que le *paging* est distribué dans l'ensemble du réseau. Le coût de cette robustesse accrue étant bien sûr une plus grande charge de la mémoire.

5.3 Performance de TCP et UDP sous les protocoles de micro mobilité

Dans [GHA01], Les performances de UDP et TCP sont examinés selon les schémas de handoff en utilisant les codes sources de CIMS "*The Columbia IP micro mobility Suite*" [CIM01] implémentés dans NS-2. La plate-forme utilisée pour la simulation est montrée dans la Figure 5.2. L'hôte mobile exécute des handoffs consécutifs entre B1 et B4 et vice versa. (Pour UDP : l'hôte mobile reçoit des paquets UDP de 512 bytes avec un débit 500Kbps, Pour TCP : le TCP Newreno est utilisé)

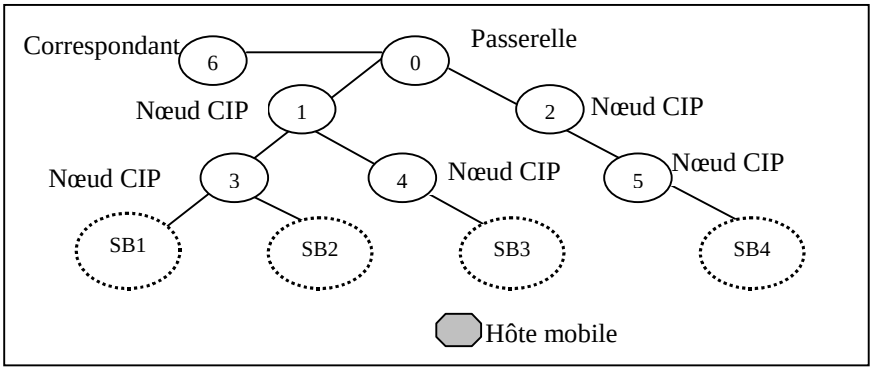


Figure 5.2 : La plate-forme Cellular IP

• Performances de TCP et UDP sous Cellular IP

- Plus l'hôte mobile se déplace rapidement, moins il passe de temps dans la région de chevauchement donc la perte de paquets augmente. Ce résultat peut être expliqué par la comparaison des délais des différents handoffs au temps que l'hôte mobile passe dans la région de chevauchement. Dans le semi-soft handoff, l'augmentation de perte de paquets est la plus faible parce que l'hôte mobile peut simultanément recevoir des paquets des nouvelles et anciennes stations de base une fois que le paquet *routing update* arrive au routeur de croisement. Cependant contrairement au schéma semi-soft, l'hôte mobile utilisant un hard peut seulement envoyer et recevoir des paquets d'une station de base à la fois ainsi il ne peut pas recevoir tous les paquets envoyés à lui et perd plus de paquets.
- La performance de TCP se dégrade à mesure que la fréquence du handoff augmente, cela est dû aux pertes de paquets. TCP a moins de temps pour se remettre de ces pertes quand la fréquence du handoff augmente (quand la fréquence du handoff atteint un handoff toutes les 3 secondes). Ceci force TCP à opérer au-dessous de son point opérationnel optimal ce qui engendre une dégradation considérable de ses performances. Le semi-soft handoff réduit la perte des paquets et améliore considérablement le débit du transport par rapport au hard handoff.
- Plus la distance du handoff est grande, plus la perte de paquet augmente dans les différents schémas de handoff de Cellular IP. Si le délai du handoff est plus court que le temps que l'hôte mobile dépense dans la région de chevauchement alors le semi-soft handoff a la meilleure performance pour le protocole Cellular IP.
- A des débits élevés, il y a une probabilité importante d'une perte accrue de paquets quand le handoff se produit; en d'autres termes la perte de paquets augmente à des débits élevés.

• Comparaison de Cellular IP et HAWAII

- La différence majeure entre les deux approches est en rapport avec les procédures de handoff et de rafraîchissement des chemins (c.-à-d. gestion des caches). HAWAII implique un routage dynamique, seulement quelques nœuds sélectionnés implémentent les caches. En comparaison avec Cellular IP, HAWAII utilise seulement une fréquence de rafraîchissement des caches, et les paquets de données ne sont pas impliqués dans la gestion des caches.
- Les performances du *semi-soft* handoff de Cellular IP et du *non-forwarding* handoff de Hawaii sont approximativement les mêmes en termes de maximum numéro de séquence reçu et ceci en utilisant un même scénario de simulation pour Cellular IP et Hawaii (le nœud mobile se déplace à une vitesse de 20 mètres/sec et effectue 3 handoffs durant 30 secondes). Par contre, les performances du *hard* handoff de Cellular IP sont meilleures que celles du *forwarding* handoff de Hawaii.
- En augmentant le nombre de handoffs les schémas de handoff de Hawaii deviennent plus performants que les schémas de handoff de Cellular IP.

- Dans un réseau Cellular IP, le nombre de sauts de la source à la destination est constant puisque tous les paquets *routing-update* doivent atteindre la passerelle. Cependant dans HAWAII, les nœuds supérieurs au routeur de croisement ne sont pas impliqués quand le handoff se produit. Par conséquent HAWAII est plus fiable et comparé à Cellular IP, a moins de paquets de contrôle au niveau des nœuds supérieurs au routeur de croisement comparé à Cellular IP.

5.4 Conclusion

Au terme de la comparaison des protocoles de micro mobilité, les auteurs ont constaté que chacun des protocoles examinés présente des avantages et des faiblesses vis-à-vis des points importants de la gestion de la micro-mobilité sous IP. Ils ont pu en tirer des conclusions plus générales vis-à-vis des protocoles de micro-mobilité.

- Tout d'abord, la gestion du handoff est et restera le problème majeur à résoudre, c'est l'essence même de la mobilité. Cette gestion doit être rapide, efficace et limitée aux seules stations qui sont concernées. On doit veiller aussi à réduire la quantité de messages de contrôle générés.
- La connectivité passive est une importante fonctionnalité des réseaux mobiles. Un support efficace de celle-ci est nécessaire à toute proposition. La réutilisation des concepts cellulaires de *paging area* semble une solution à la fois efficace et adaptable à IP.
- Dans une vue globale, la robustesse et l'adaptabilité sont nécessaires pour tout protocole réseau. La trop grande simplicité de certains mécanismes peut laisser entrevoir des problèmes car l'évaluation de ces points doit toujours être faite en ayant à l'esprit la charge future des réseaux mobiles qui sera très importante si on en croit les estimations (plusieurs millions d'utilisateurs par réseau) et les limitations des machines qui serviront ces réseaux.
- Enfin, la gestion du trafic à l'intérieur du réseau mobile doit être efficace. Non seulement ce type de communication entre utilisateurs sera vraisemblablement très important dans l'avenir mais le trafic de contrôle fait aussi partie de ce trafic. Si la bande passante sur la voie radio est une ressource partagée, chère et à économiser, les réseaux d'accès sans fil sont souvent construits sur la base de nombreuses liaisons louées dont l'utilisation doit être optimisée.

A ces conclusions, il faut ajouter ce qui suit :

- Il est difficile de définir la manière dont l'intégration technologique se fera dans les futurs réseaux mobiles. En effet, les services de la couche radio sont encore mal définis (notamment au niveau de la qualité de service) et une spécification d'un protocole du niveau de IP ne peut s'envisager sans une idée claire des services apportés par les couches inférieures.

Enfin, ils soulignent le manque flagrant de données numériques et de simulations réalistes dans le domaine de la micro-mobilité sous IP. Il est clair qu'une recherche doit être menée dans le sens d'une évaluation *quantitative* des propositions qui sont faites. Cette évaluation peut se faire via des simulations pertinentes, à partir des données des réseaux existants dans une optique d'anticipation. Cette recherche est encore à faire.

De plus les performances de UDP et TCP ont été examinées selon les schémas de handoff de Cellular IP dans [GHA01]. Une comparaison des performances de TCP sous le protocole de micro-mobilité Cellular IP et sous Hawaii a été réalisée selon certains critères tels que le nombre de handoff du nœud mobile ou le maximum numéro de séquence reçu par le nœud mobile.

Chapitre 6 Simulation

6.1 Introduction

TCP est le protocole de transport le plus utilisé. Son mécanisme de contrôle de flux est basé sur le timeout et l'ajustement de la taille de la fenêtre. Ce dernier est très sollicité dans les réseaux filaires ayant des BER (bit error rates) de l'ordre de 10^8. Cependant, dans le cas des liens sans fil ayant des BER de l'ordre de 10^3, TCP ne peut pas fonctionner efficacement à cause des pertes de paquets engendrées par un BER élevé et non pas à cause du problème de congestion qui survient dans les réseaux filaires.

Le rendement de TCP baisse dans les réseaux sans fil parce que TCP ne distingue pas les pertes de paquets causées par des taux d'erreurs importants et celles provoquées par une congestion. Dans les deux cas, après un timeout TCP diminue la taille de la fenêtre ce qui engendre un faible débit dans le réseau.

Dans ce chapitre, on simulera le comportement du protocole TCP quand des handoffs se produisent dans des réseaux d'accès Cellular IP ainsi que dans des réseaux d'accès Hawaii. On comparera par la suite les résultats des simulations des schémas de handoffs de Cellular IP avec ceux de Hawaii.

6.2 Scénario de simulation

Afin de comparer les différents schémas de handoffs, on a utilisé les scripts de CIMS [CIM01] implémentés dans le simulateur NS2 (voir Annexe 4). Les topologies utilisées pour la simulation des deux protocoles sont identiques (Figure 6.1 et Figure 6.2). Les modèles de simulation des deux protocoles Cellular IP et Hawaii sont basés sur les descriptions présentées au chapitre 4.

Le réseau comprend des connexions filaires en liaisons duplex modelées à 10Mbps avec un délai de 2ms. On a appliqué le protocole 802.11 pour la liaison sans fil. Les résultats des simulations sont obtenus en utilisant un unique hôte mobile qui se déplace de la station de base 1 vers la station de base 4 et inversement en supposant que le réseau mère de l'hôte mobile (MH) est le réseau de simulation. L'hôte correspondant (CH) exploite une application ftp (500kb/s) à partir de la première seconde du début de la simulation. Ainsi le trafic TCP est dirigé CH vers le MH. Les paquets TCP ont une taille par défaut de 1000 bytes.

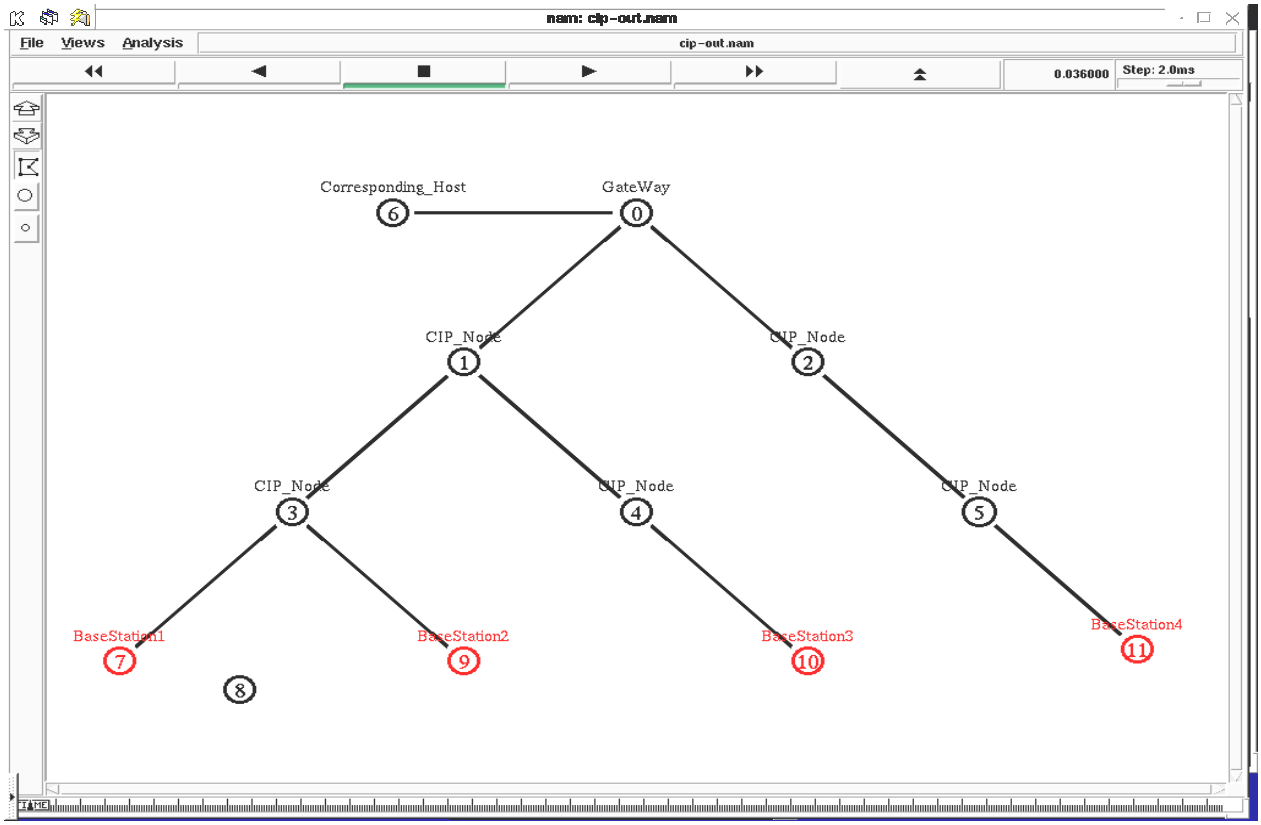


Figure 6.1 : La plate-forme Cellular IP

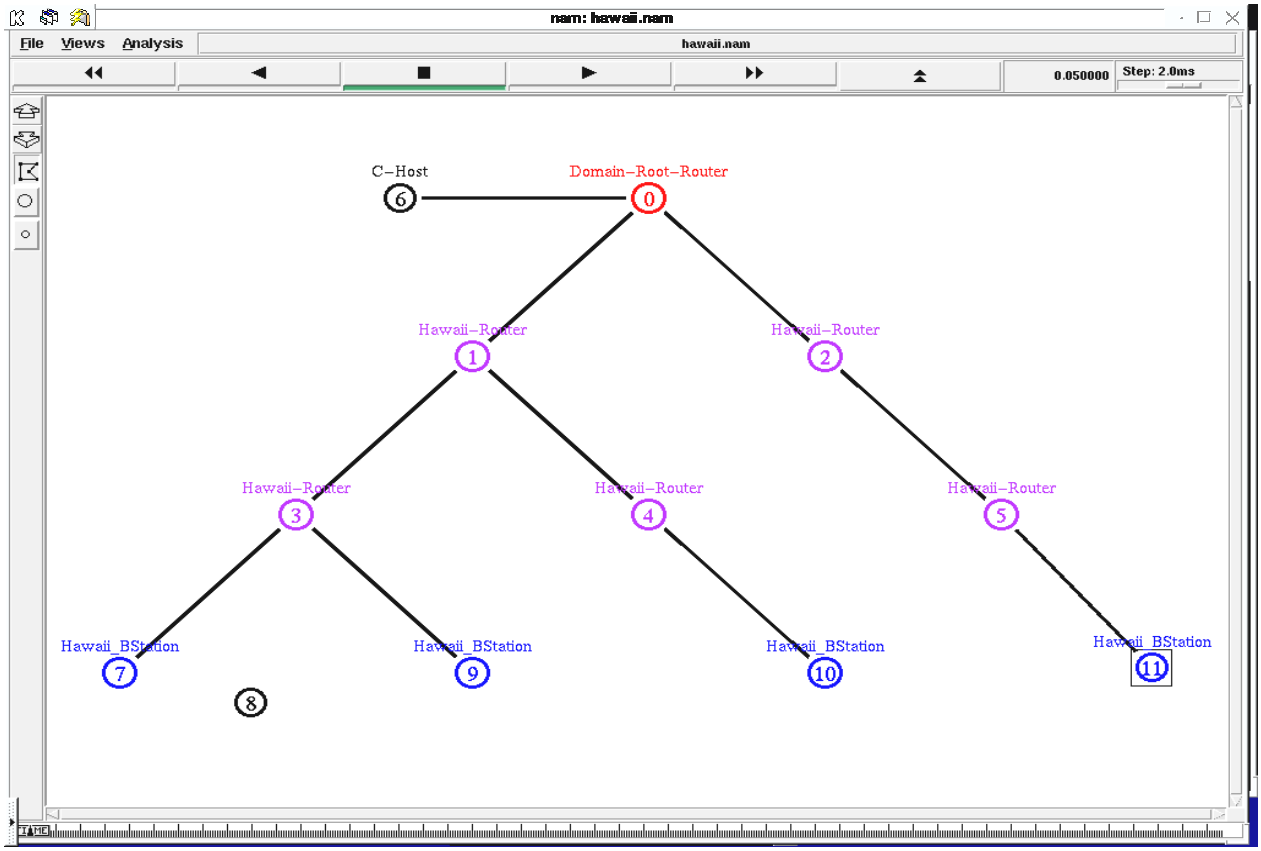


Figure 6.2 : La plate-forme Hawaii

6.3 Comportement de TCP sous Cellular IP

On étudiera dans ce qui suit le comportement de TCP Tahoe sous les deux schémas de handoffs de Cellular IP selon certains critères tels que la vitesse de l'hôte mobile, la taille de la zone de chevauchement. On comparera également le comportement des différentes variantes de TCP (voir chapitre 2) à savoir TCP Tahoe, TCP Reno, TCP Newreno, TCP Sack, TCP Vegas sous les deux schémas de handoffs de Cellular IP.

6.3.1 Débit TCP

La Figure 6.3 montre le comportement de TCP Tahoe en calculant le débit reçu au niveau de l'hôte mobile toutes les secondes sous les deux schémas de handoffs de Cellular IP (Hard et semisoft) où l'hôte mobile effectue 6 handoffs pendant une période de simulation de 60 secondes en ayant une vitesse constante de 20 unités par seconde.

On remarque que le débit de TCP baisse considérablement de 6 fois pour le Hard handoff. Cette baisse est due au 6 handoffs qu'effectue l'hôte mobile pendant 60 secondes en effectuant un aller / retour de la station de base 1 à la station de base 4. Par contre cette baisse est moins importante sous le semisoft handoff.

(1er handoff s'effectue dans l'intervalle de temps [5-10 sec]
2ème handoff s'effectue dans l'intervalle de temps [15-20 sec]
3 ^{ème} handoff s'effectue dans l'intervalle de temps [25-30 sec]
4 ^{ème} handoff s'effectue dans l'intervalle de temps [35-40 sec]
5 ^{ème} handoff s'effectue dans l'intervalle de temps [45-50 sec]
6 ^{ème} handoff s'effectue dans l'intervalle de temps [55-60 sec])

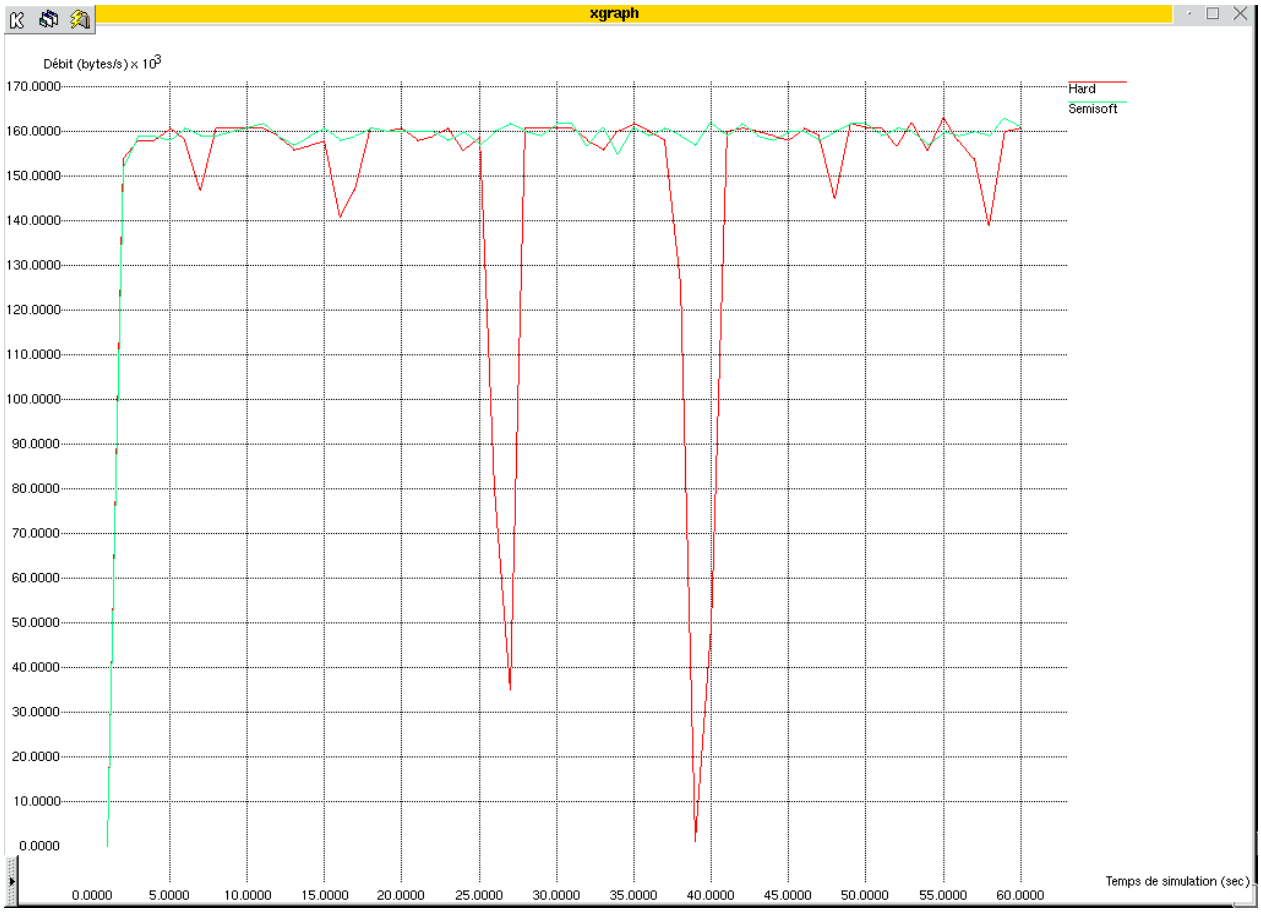


Figure 6.3 : Débit TCP « Tahoe » pour les deux schémas de handoffs de Cellular IP

6.3.2 Fenêtre de congestion

On remarque sur la figure ci-dessous que la fenêtre de congestion, initialisée à 1, croît initialement de façon exponentielle (ce qui représente la phase du démarrage lent où la valeur de cette dernière double après chaque RTT si chaque paquet est acquitté). Par la suite, elle croît linéairement à partir d'un certain seuil (ce qui représente la phase d'évitement de congestion) jusqu'à la détection de perte de paquet (la détection de perte peut se faire de deux manières : soit par l'expiration de temporisateur, soit à la réception d'acquittements dupliqués). Après la détection d'une perte, cette dernière est réinitialisée à 1 et les algorithmes du démarrage lent et d'évitement de congestion sont ré-exécutés. On remarque que la courbe du semisoft handoff chûte moins de fois que celle du hard handoff. On constate donc mois de perte de paquets avec le semisoft handoff. On remarque également que toutes les chûtes de la courbe du hard handoff sont dans les intervalles de temps où les handoffs s'effectuent.

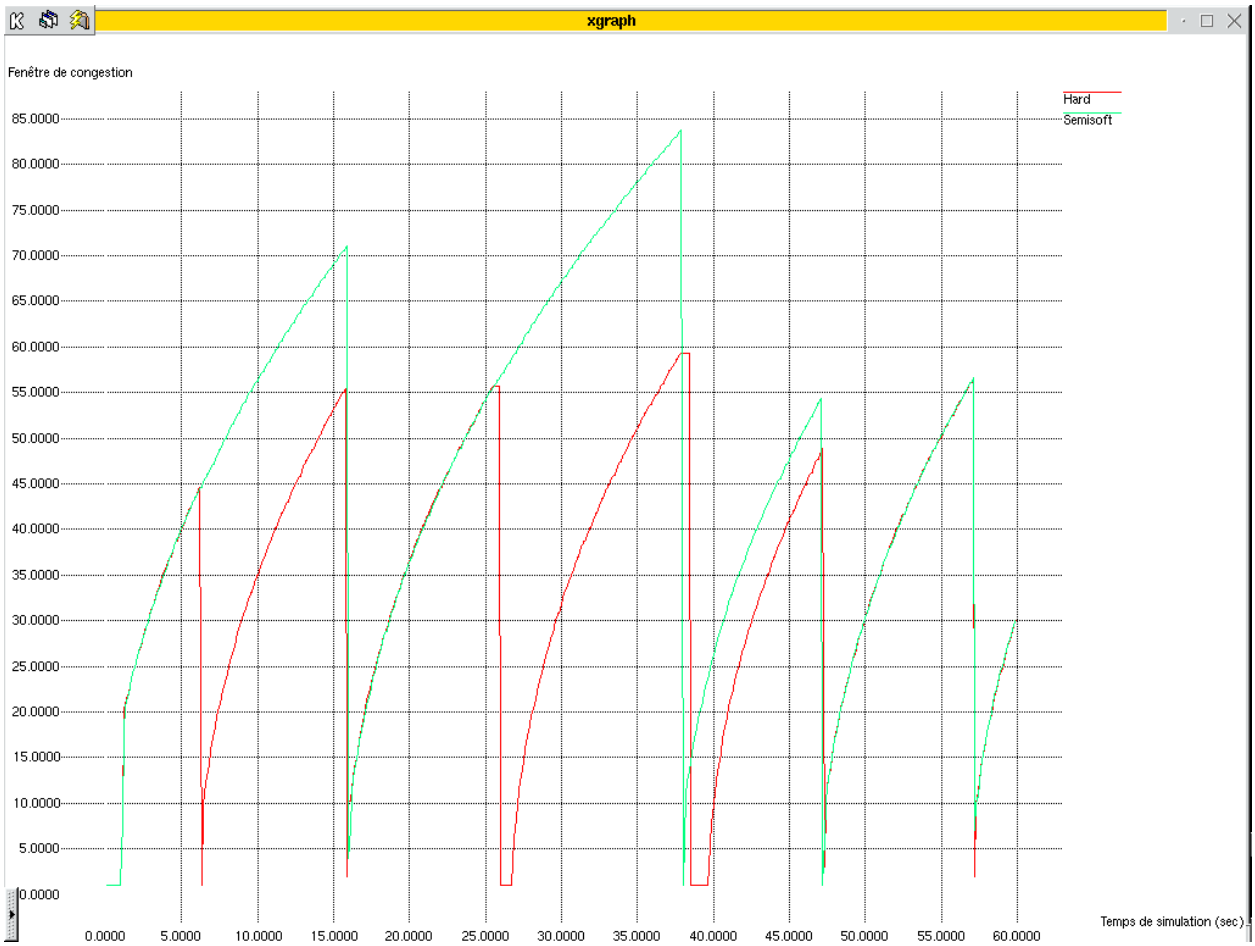


Figure 6.4 : Comportement de la fenêtre de congestion pour les deux schémas de handoff de Cellular IP

6.3.3 Vitesse de l'hôte mobile

En augmentant la vitesse de l'hôte mobile on peut constater sur les figures 6.5 et 6.6 comment la vitesse affecte la perte de paquets pendant le handoff et c'est ce qui réduit TCP au-dessous de son point optimal. A partir des Figures 6.5 et 6.6, on constate que les numéros de séquence des paquets reçus par l'hôte mobile diminuent en augmentant la vitesse du mobile dans le hard handoff et le semisoft handoff respectivement. Cependant, on remarque une anomalie au niveau du hard handoff où la courbe correspondante à la vitesse 90 unité/sec est supérieure à celle de 80 unités/sec. Hormis, ce cas précis on peut généraliser et affirmer qu'en augmentant la vitesse du mobile les numéros de séquence reçus par l'hôte mobile diminuent.

Figure 6.5 : Numéros de séquence sous le hard handoff pour différentes vitesses du MH

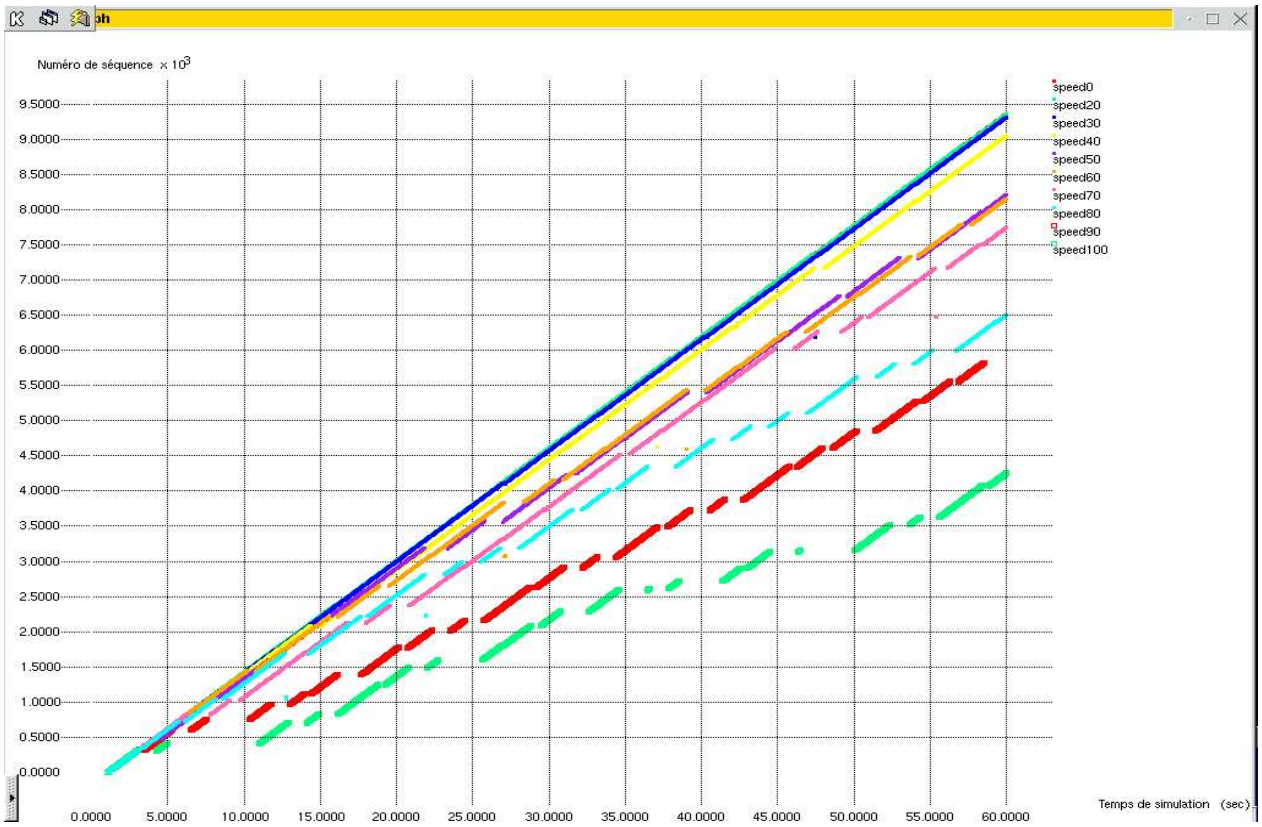


Figure 6.6 : Numéros de séquence sous le semisoft handoff pour différentes vitesses du MH

De la Figure 6.7 on constate que plus la vitesse de l'hôte augmente plus le nombre de hard handoffs augmente et moins le numéro de séquence maximum envoyé par le correspondant augmente. On remarque une chute considérable du numéro de séquence maximum à la vitesse 80 unité/sec et pourtant ce numéro devrait être supérieur à celui constaté à la vitesse 90 unité/sec.

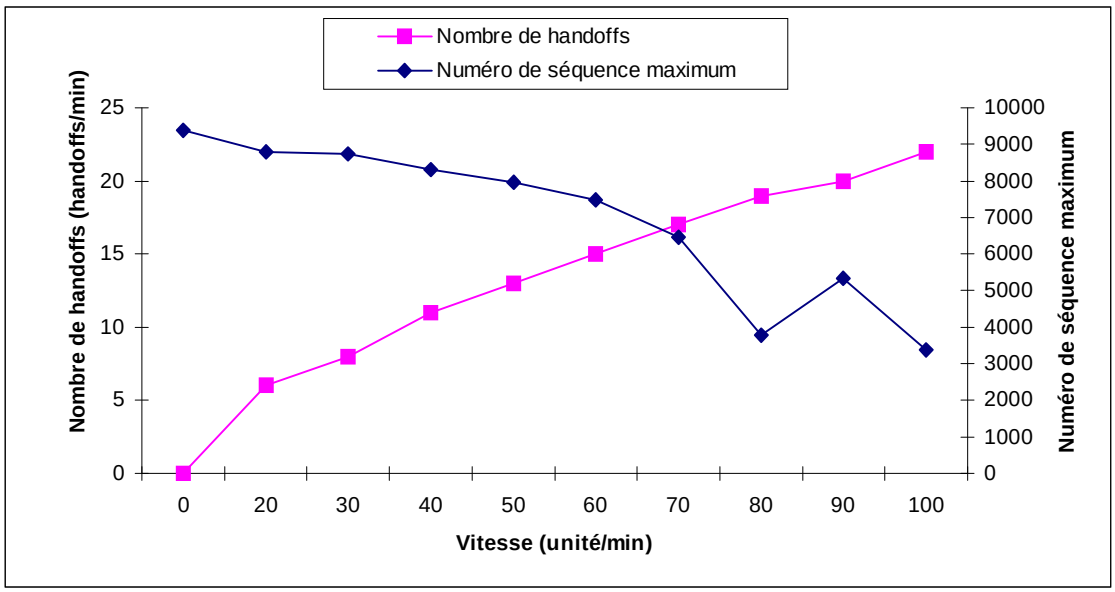


Figure 6.7 : Numéro de séquence maximum et le nombre de hard handoffs en fonction de la vitesse de HM

De la Figure 6.8, on constate également que plus la vitesse de l'hôte augmente plus le nombre de semisoft handoffs augmente et plus le numéro de séquence maximum envoyé par le correspondant diminue.

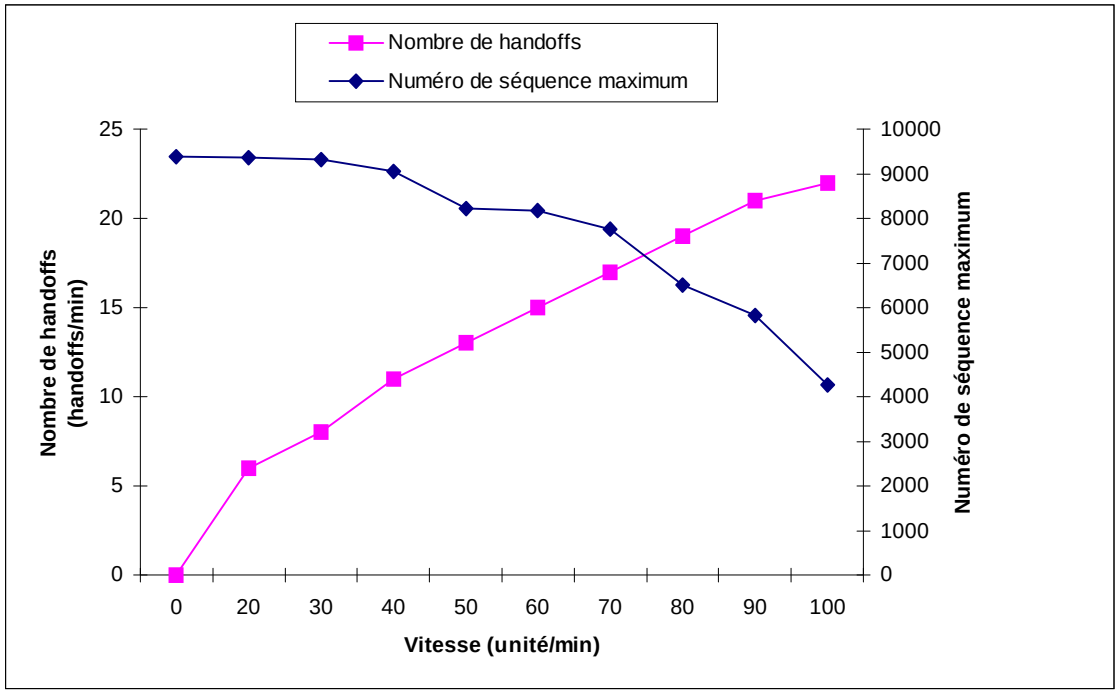


Figure 6.8 : Numéro de séquence maximum et le nombre de semisoft handoffs en fonction de la vitesse de HM

La Figure 6.9 compare le numéro de séquence maximum envoyé par le correspondant en fonction de la vitesse de l'hôte mobile des deux schémas de handoffs.

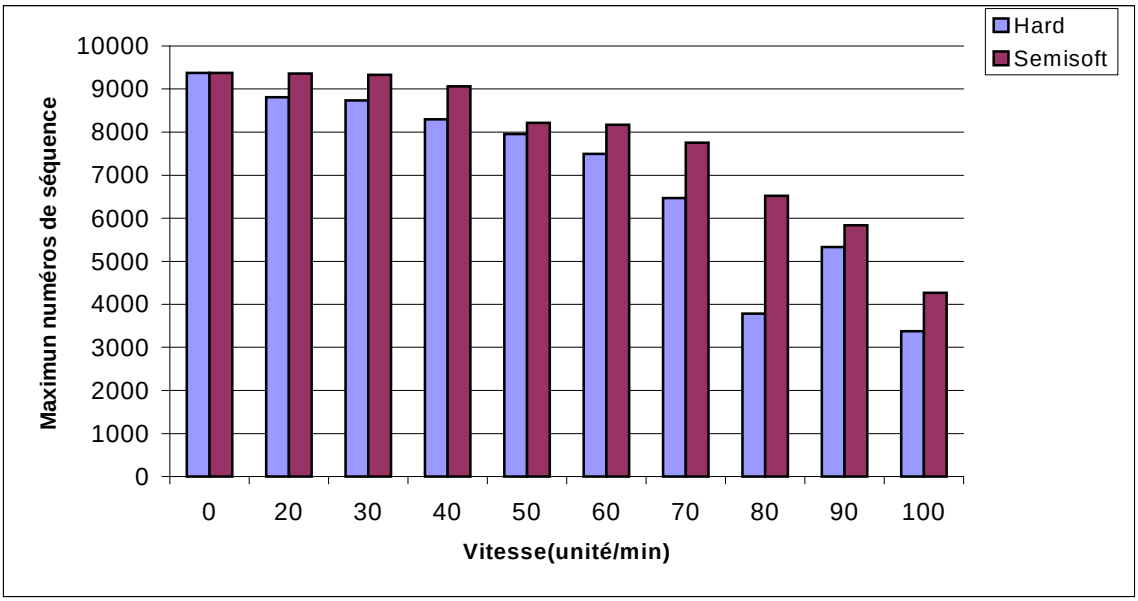


Figure 6.9 : Numéro de séquence maximum en fonction de la vitesse de HM pour les deux schémas de handoffs Cellular IP

On constate que les numéros de séquence maximums sous le semisoft handoff pour les différentes vitesses sont supérieurs à ceux sous le hard handoff. Ces résultats peuvent être expliquer comme suit : dans le hard handoff, les hôtes mobiles ne peuvent pas recevoir des paquets simultanément de la nouvelle et de l'ancienne station de base durant le handoff ce qui minimise le trafic de signalisation au prix d'un certain taux de perte de paquets, par contre le semisoft handoff exploite le fait que les hôtes mobiles peuvent recevoir des paquets simultanément de la nouvelle et de l'ancienne station de base durant le handoff, minimisant ainsi la perte de paquets et fournissant ainsi des performances améliorées pour le protocole TCP.

Comme le montre la Figure 6.10, quand le nombre de handoffs par minute augmente le débit TCP reçu au niveau de l'hôte mobile diminue. La courbe du semisoft handoff est plus élevée que celle du hard handoff. Ceci peut être expliqué par le fait que l'hôte mobile sous le semisoft handoff reçoit simultané des paquets de plus d'une station de base, ainsi il reçoit plus de paquets.

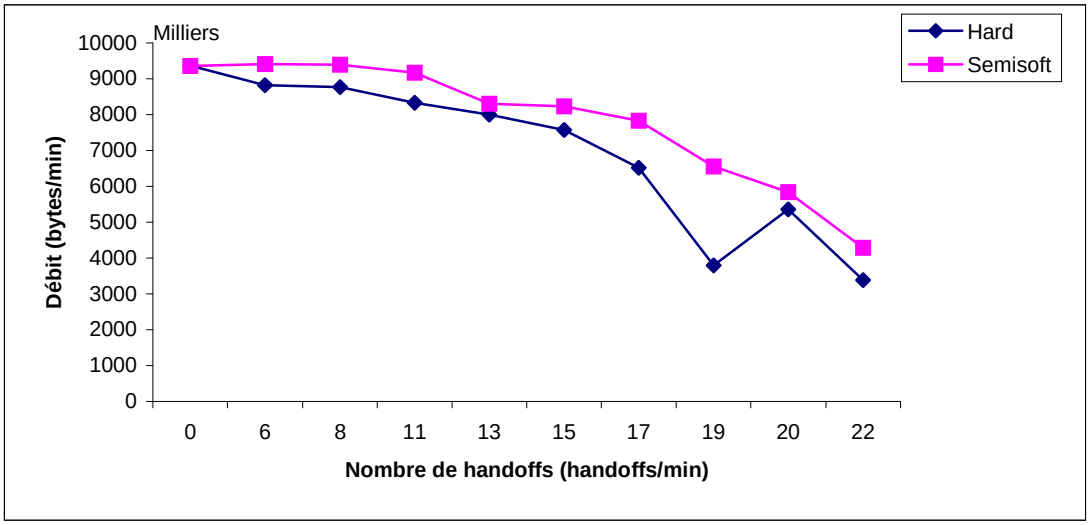


Figure 6.10 : Débit TCP en fonction de la vitesse de HM pour les deux schémas de handoffs Cellular IP

6.3.4 Taille de la zone de chevauchement

Une zone de chevauchement est la zone d'intersection de deux zones de couverture de deux stations de base. Les résultats de simulation concernant la zone de chevauchement sont obtenus avec une vitesse du mobile constante 20 unité/sec. On constate de la Figure 6.11 que tant que le nombre de hard handoffs est à 6 handoffs par minute le numéro de séquence maximum envoyé par l'hôte mobile reste stable et monotone même si la taille de la zone de chevauchement augmente. A partir de la taille 50 unités, une chute importante du numéro de séquence maximum est constatée à cause du nombre de handoffs qui augmente.

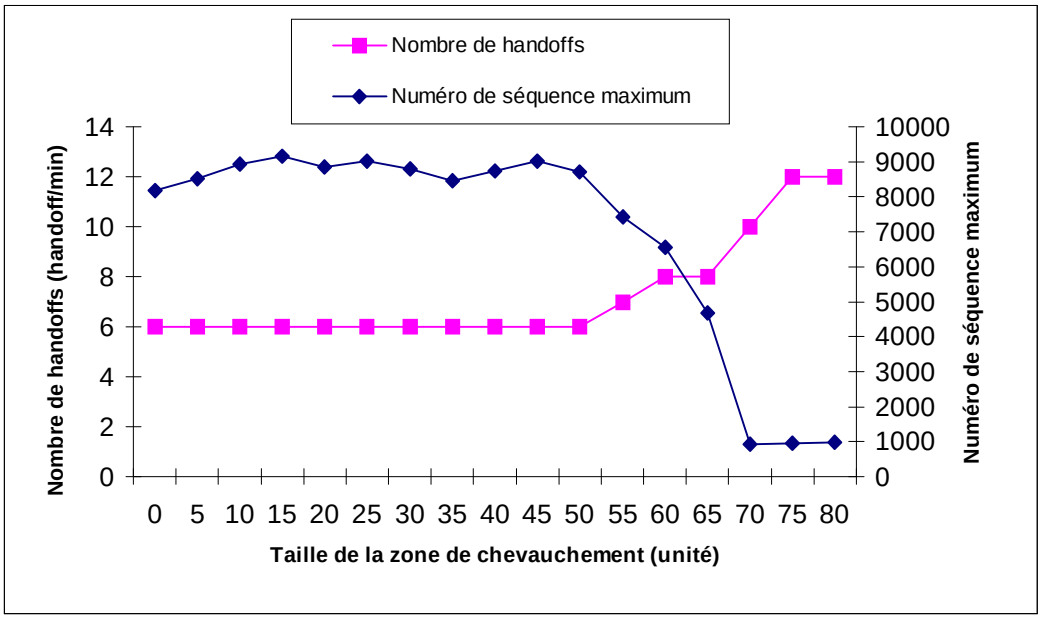


Figure 6.11 : Numéro de séquence maximum et le nombre de hard handoffs en fonction de la zone de chevauchement

Pour le semisoft handoff, on constate sur la Figure 6.12 que le numéro de séquence maximum envoyé par l'hôte mobile augmente en élargissant la zone de chevauchement de 0 à 15 unités, par contre entre 20 et 50 unités le numéro de séquence maximum reste constant et tout ceci en ayant un nombre de handoffs constants (6 handoffs / min). A partir de la taille 55 unités de la zone chevauchement le nombre de handoffs par minute augmente ce qui engendre une baisse progressive du numéro de séquence maximum.

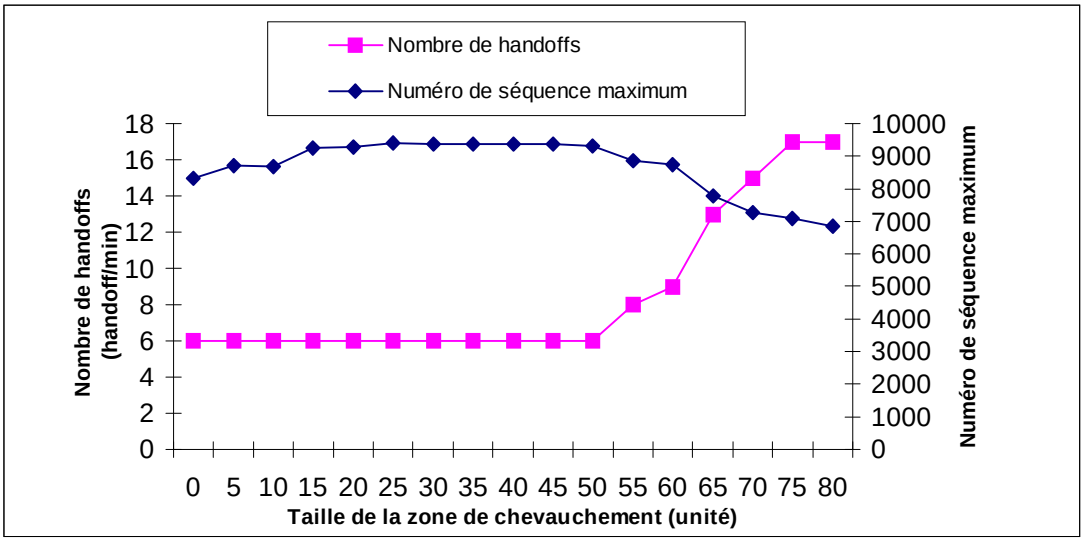


Figure 6.12 : Numéro de séquence maximum et le nombre de semisoft handoffs en fonction de la taille e la zone de chevauchement

La Figure 6.13 compare le numéro de séquence maximum envoyé par l'hôte mobile sous les deux schémas de handoffs de Cellular IP en augmentant la taille de la zone de chevauchement des zones de couverture pendant une minute de simulation.

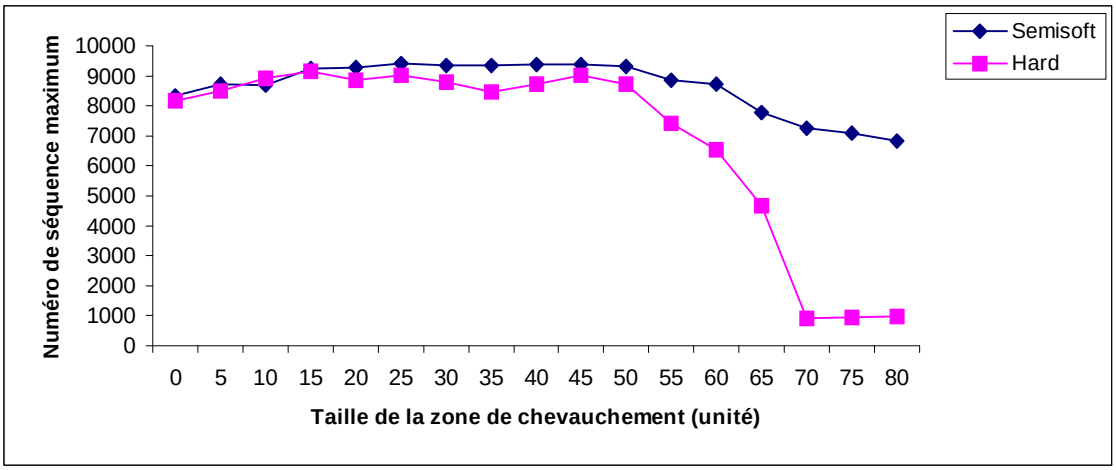


Figure 6.13 : Numéro de séquence maximum en fonction de la taille de la zone de chevauchement pour les deux schémas de handoffs Cellular IP

On constate sur la figure précédente que la courbe concernant le semisoft est légèrement supérieure à celle du hard handoff entre la taille 0 et 50 unités. En augmentant la taille de la zone de chevauchement au-delà de 50 unités, on constate que la courbe concernant le semisoft est sensiblement supérieure à celle du hard handoff. Une fois que le nombre de handoffs augmente à partir de la taille 55 unités la courbe du semisoft handoff chute progressivement par contre celle du hard handoff subit une chute agressive.

On étudie dans ce qui suit le comportement de UDP en élargissant la taille de la zone de chevauchement est ceci en calculant le nombre de paquets perdus pendant une minute de simulation.

On constate sur la Figure 6.14 que le nombre de paquets perdus sous le hard handoff diminue considérablement de 299 jusqu'à 28 paquets entre la taille 0 et 15 unités ensuite il reste constant (5 – 7 paquets perdus) entre 20 et 55 unités et enfin au-delà de 55 unités il augmente. On peut résumer ceci de la sorte : le nombre de paquets perdus diminue en augmentant la taille de la zone de chevauchement tant que le nombre de handoffs par minute reste constant, mais une fois que ce dernier augmente le nombre de paquets perdus augmente de façon exponentielle.

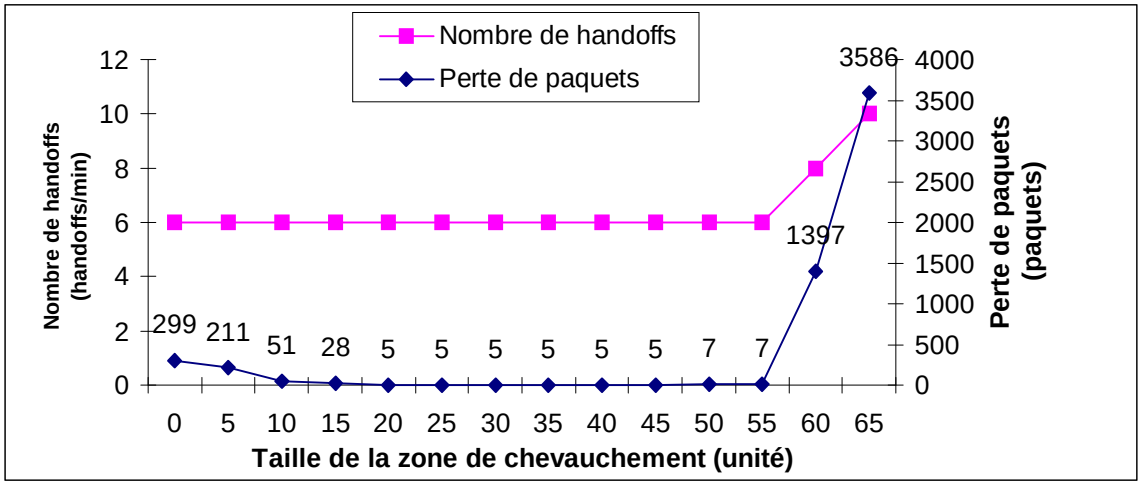


Figure 6.14 : Nombre de paquets perdus et le nombre de hard handoffs en fonction de la taille de la zone de chevauchement

Pour le semisoft handoff on constate sur la Figure 6.15 que le nombre de paquets perdus diminue considérablement de 300 jusqu'à 0 paquet entre la taille 0 et 55 unités ensuite au-delà de 55 unités il augmente. On peut résumer ceci de la sorte : le nombre de paquets perdus diminue en augmentant la taille de la zone de chevauchement tant que le nombre de handoffs par minute reste constant, mais une fois que ce dernier augmente le nombre de paquets perdus augmente de façon moins importante que sous le hard handoffs à savoir: 159 paquets perdus sous le semisoft handoff et 3586 paquets pour le hard handoff en ayant une taille de la zone de chevauchement de 65 unités.

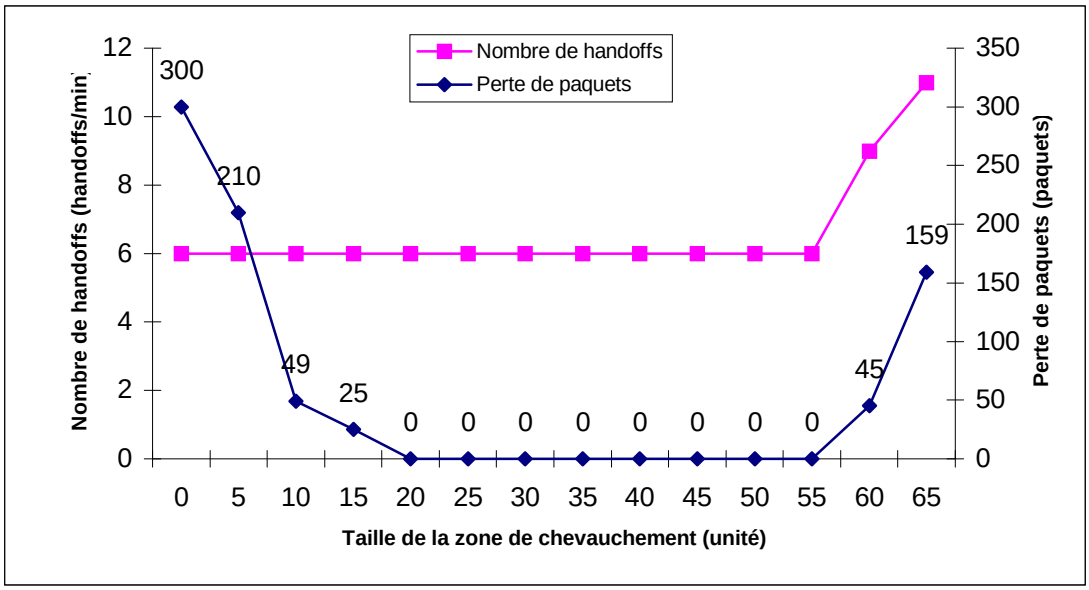


Figure 6.15 : Nombre de paquets perdus et le nombre de semisoft handoffs en fonction de la taille de la zone de chevauchement

6.3.5 Variantes de TCP

La Figure 6.16 est une comparaison des débits TCP calculés au niveau de l'hôte mobile pour différentes implémentations de TCP (présentées au chapitre 2) en appliquant les deux schémas de handoff du protocole Cellular IP.

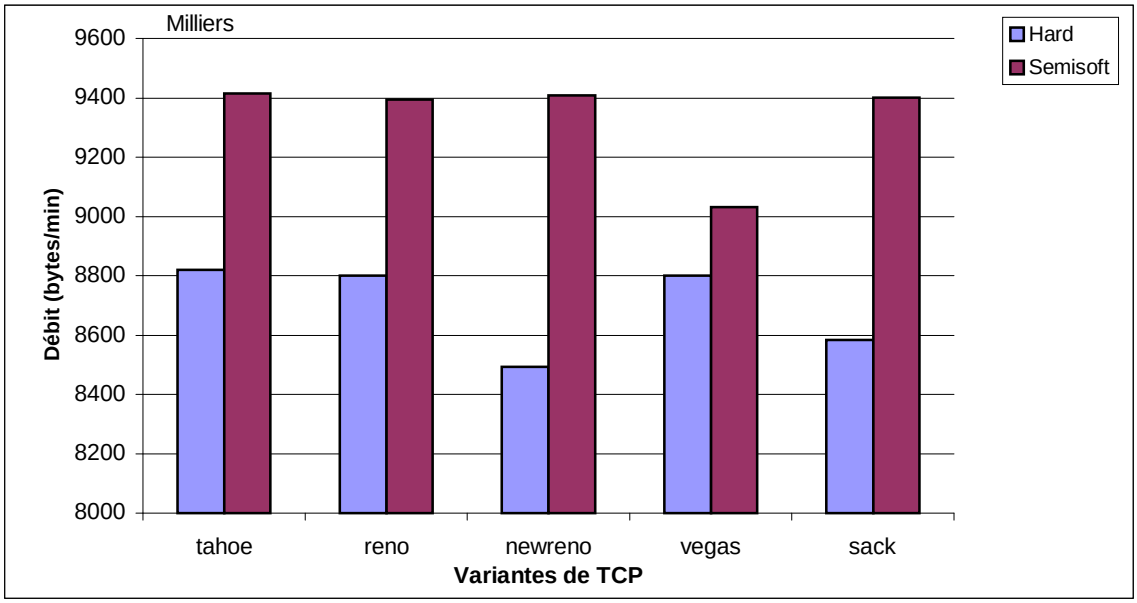


Figure 6.16 : Débit pour différentes implémentations de TCP en appliquant les deux schémas de handoff du protocole Cellular IP

On constate selon l'histogramme du semisoft handoff que pour les quatre variantes de TCP (Tahoe, Reno, Newreno, Sack) le débit de TCP est identique sauf pour le TCP Vegas où le débit est moins important. Pour le hard handoff, on constate que les débit TCP reçus pour trois variantes de TCP (Tahoe, Reno, Vegas) sont égaux et sont supérieurs à ceux du TCP Newreno et TCP Sack. Cependant, quelque soit la variante TCP le débit reçu sous le semisoft handoff est supérieur à celui du hard handoff.

6.4 Comportement de TCP sous Hawaii

On étudiera dans ce qui suit le comportement de TCP Tahoe sous les deux schémas de handoffs de Hawaii selon la variation de la vitesse de l'hôte mobile. On comparera également le comportement des différentes variantes de TCP à savoir TCP Tahoe, TCP Reno, TCP Newreno, TCP Sack, TCP Vegas sous les deux schémas de handoffs de Hawaii.

6.4.1 Débit TCP

La Figure 6.17 montre le comportement de TCP Tahoe en calculant le débit reçu au niveau de l'hôte mobile toutes les secondes sous les deux schémas de handoffs de Hawaii (*Forwarding* et *Non-Forwarding*) où l'hôte mobile effectue 6 handoffs pendant une période de simulation de 60 secondes en ayant une vitesse constante de 20 unités par seconde.

On remarque qu'il n'y pas une nette différence entre la courbe du schéma *Forwarding* et celle du *Non-Forwarding*. Cependant, on constate aucune chute brusque du débit TCP.

Figure 6.17 : Débit TCP (Tahoe) pour les deux schémas de handoffs de Hawaii

6.4.2 Fenêtre de congestion

On remarque sur la figure 6.18 que la fenêtre de congestion, initialisée à 1, croît initialement de façon exponentielle (ce qui représente la phase du démarrage lent où la valeur de cette dernière double après chaque RTT si chaque paquet est acquitté). Par la suite, elle croît linéairement à partir d'un certain seuil (ce qui représente la phase d'évitement de congestion) jusqu'à la détection de perte de paquet (la détection de perte peut se faire de deux manières : soit par l'expiration de temporisateur, soit à la réception d'acquittements dupliqués). Après la détection d'une perte, cette dernière est réinitialisée à 1 et les algorithmes du démarrage lent et d'évitement de congestion sont ré-exécutés. On remarque également que les courbes qui représentent respectivement le comportement de la fenêtre de congestion durant le *forwarding* handoff et le *non forwarding* handoff sont identiques.

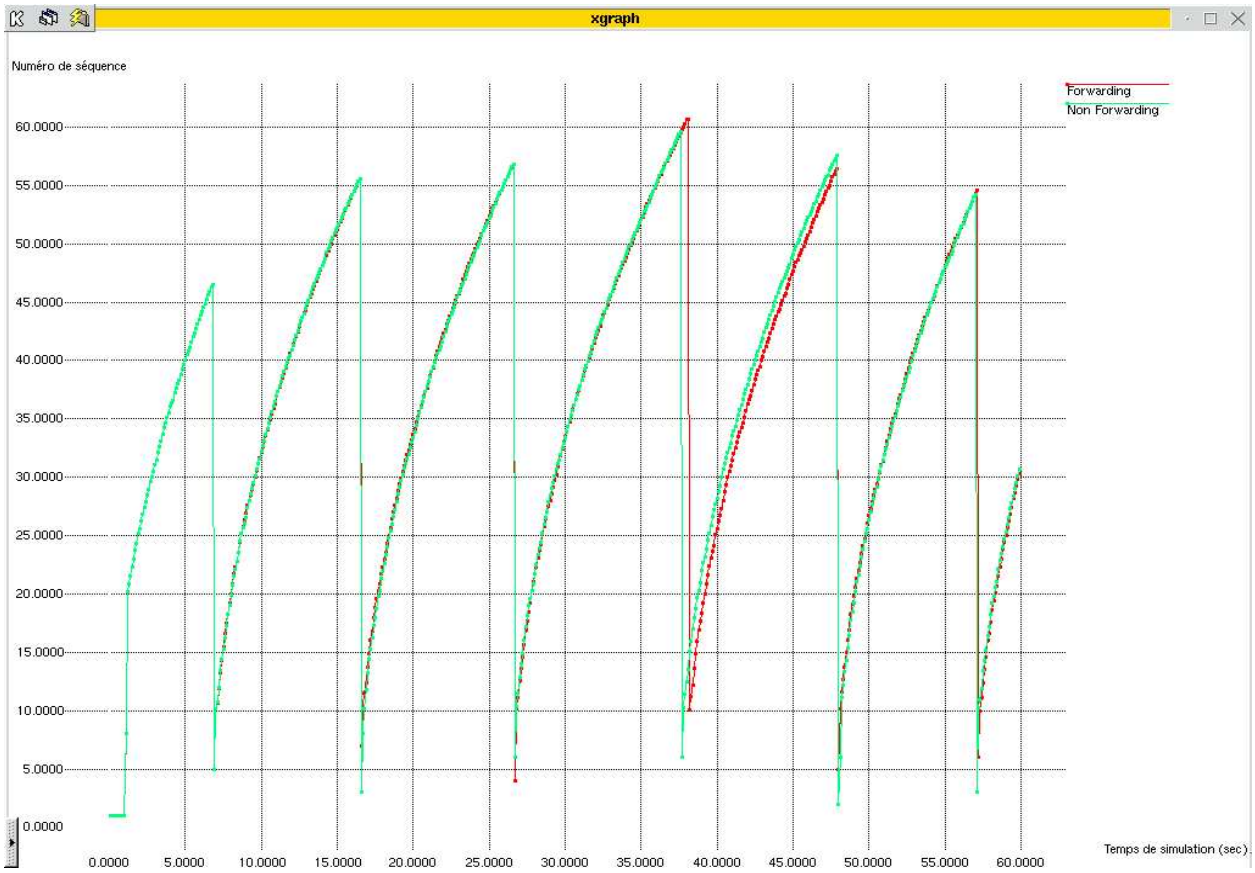


Figure 6.18 : Comportement de la fenêtre de congestion pour les deux schémas de handoff de Hawaii

6.4.3 Vitesse de l'hôte mobile

On constate sur les Figures 6.19 et 6.20 que les numéros de séquence des paquets reçus par l'hôte mobile diminuent en augmentant la vitesse du mobile quelque soit le schéma de handoff.

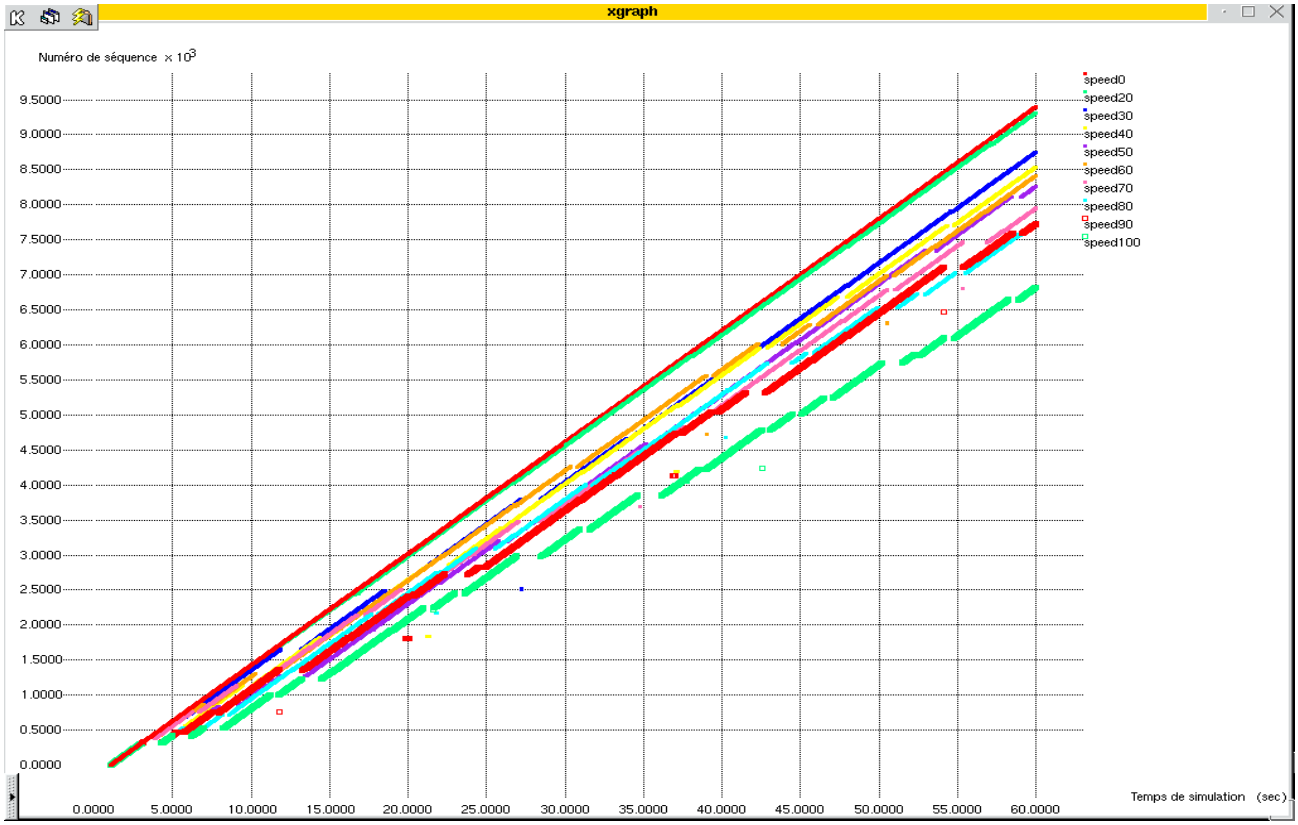


Figure 6.19 : Numéros de séquence sous le *Forwarding* handoff pour différentes vitesses du MH

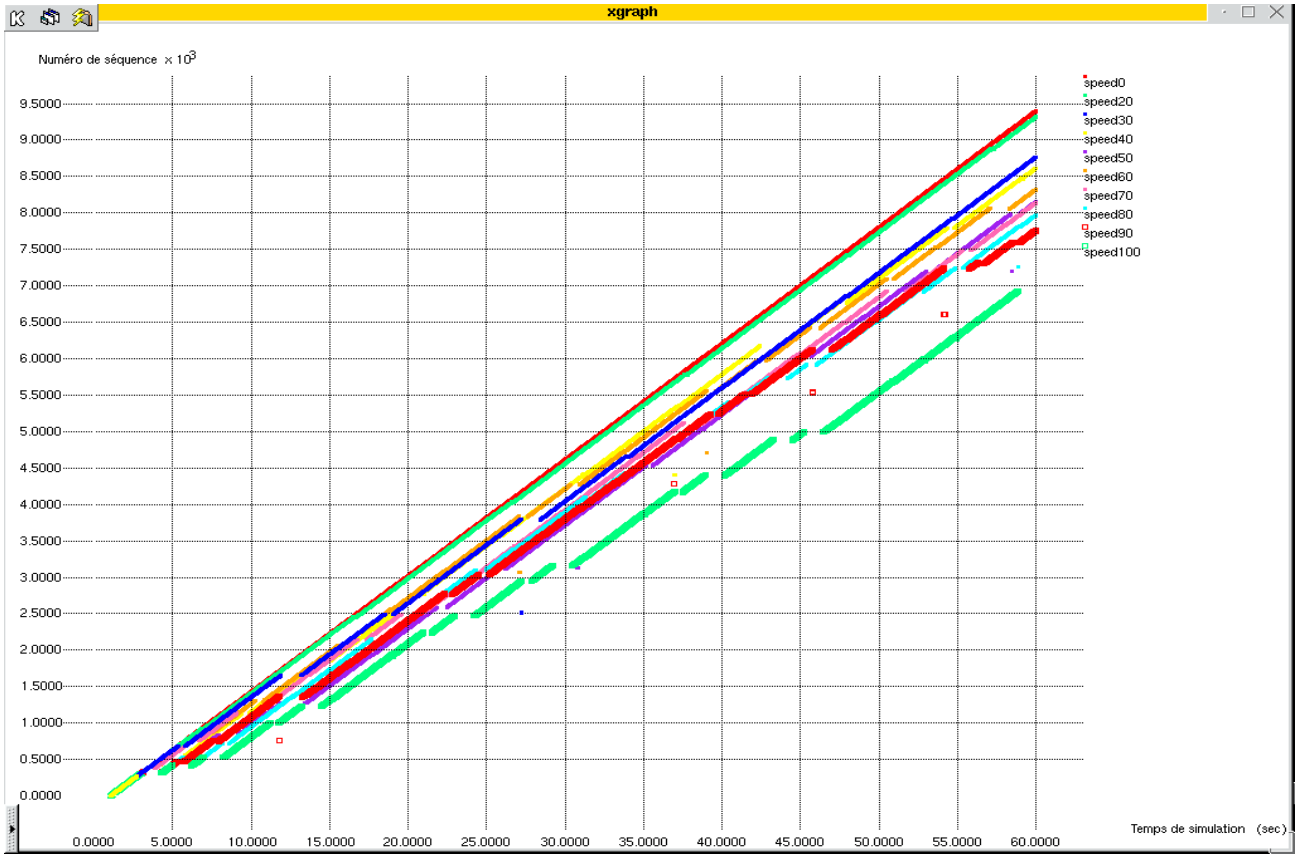


Figure 6.20 : Numéros de séquence sous le *Non-Forwarding* handoff pour différentes vitesses du MH

Des Figures 6.21 et 6.22 on constate que plus la vitesse de l'hôte augmente plus le nombre de handoffs augmente et moins le maximum numéro de séquence envoyé par le correspondant augmente pour les deux schémas de handoff Hawaii.

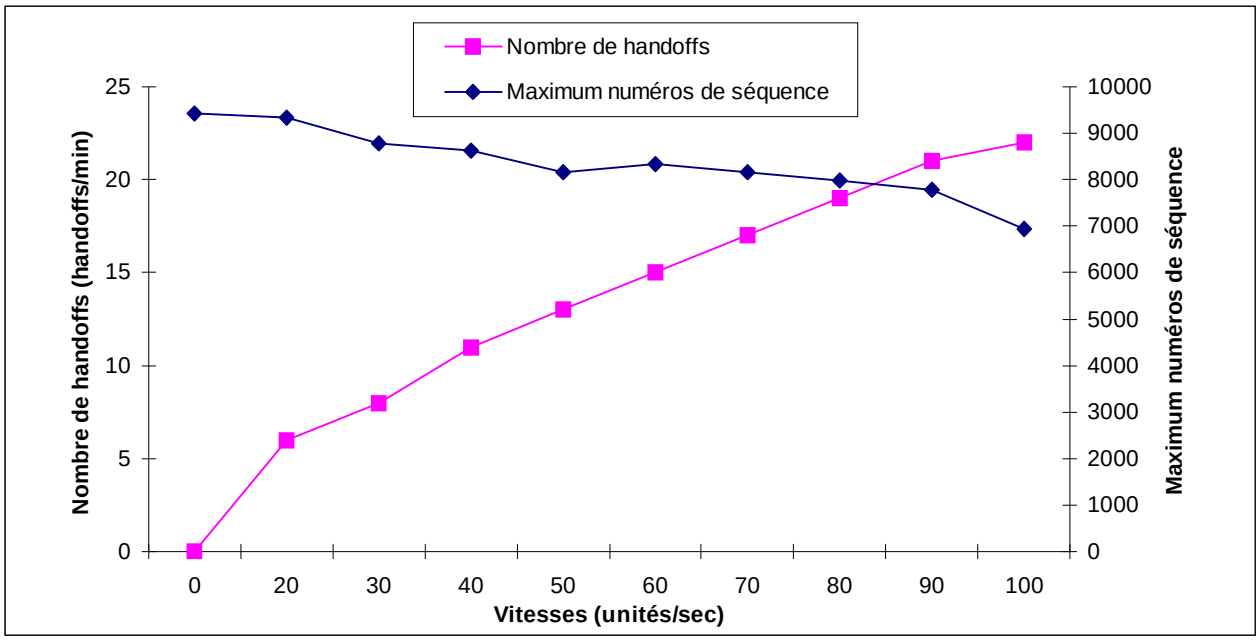


Figure 6.21 : Numéro de séquence maximum et le nombre de handoffs en fonction de la vitesse de HM sous le *Non-Forwarding* handoff

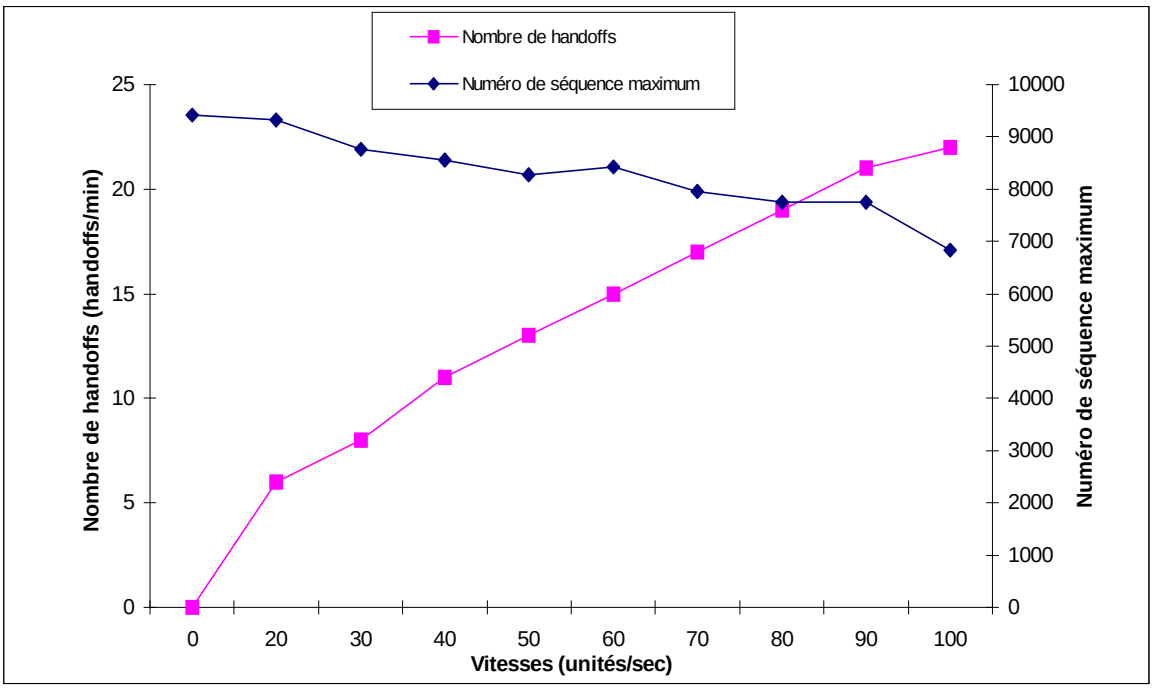


Figure 6.22 : Numéro de séquence maximum et le nombre de handoffs en fonction de la vitesse de HM sous le *Forwarding* handoff

La Figure 6.23 compare le numéro de séquence maximum envoyé par le correspondant en fonction de la vitesse de l'hôte mobile des deux schémas de handoffs de Hawaii. On constate que les deux courbes présentes sur la figure ci-dessus sont pratiquement superposées sauf dans l'intervalle de vitesse [60-90unités/sec] où la courbe correspondante au *non-forwarding* handoff prend légèrement le dessus sur celle du *forwarding* handoff. On constate également que plus la vitesse de l'hôte mobile augmente plus le numéro de séquence maximum décroît quelque soit le schémas de handoff.

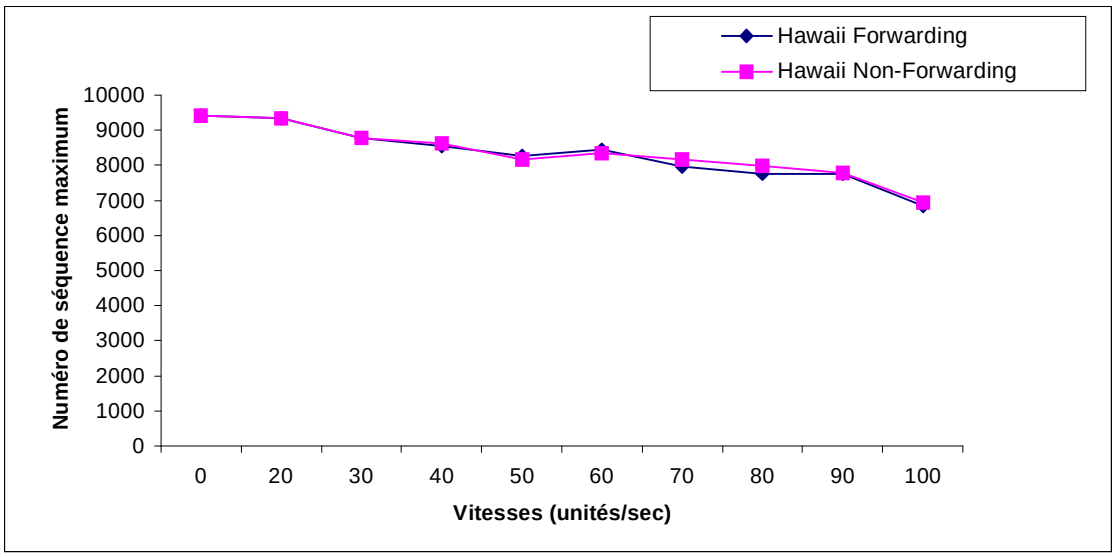


Figure 6.23 : Numéro de séquence maximum en fonction de la vitesse de HM pour les deux schémas de handoff Hawaii

Comme le montre la Figure 6.24, quand le nombre de handoffs par minute augmente le débit TCP reçu au niveau de l'hôte mobile diminue pour les deux schémas de handoff.

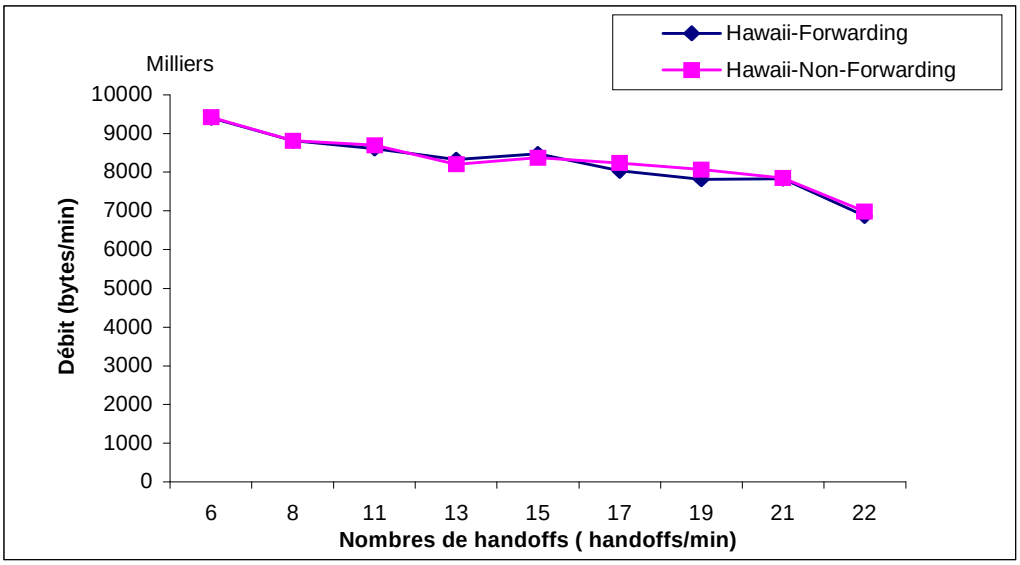


Figure 6.24 : Débit TCP en fonction de la vitesse de HM pour les deux schémas de handoffs Hawaii

On constate également que les deux courbes présentes sur la figure ci-dessus sont pratiquement superposées sauf dans l'intervalle [17-21handoffs/min] où la courbe correspondante au *non-forwarding* handoff prend légèrement le dessus sur celle du *forwarding* handoff.

6.4.4 Variantes de TCP

La Figure 6.25 est une comparaison des débits TCP calculés au niveau de l'hôte mobile pour différentes implémentations de TCP (présentées au chapitre 2) en appliquant les deux schémas de handoff du protocole Hawaii.

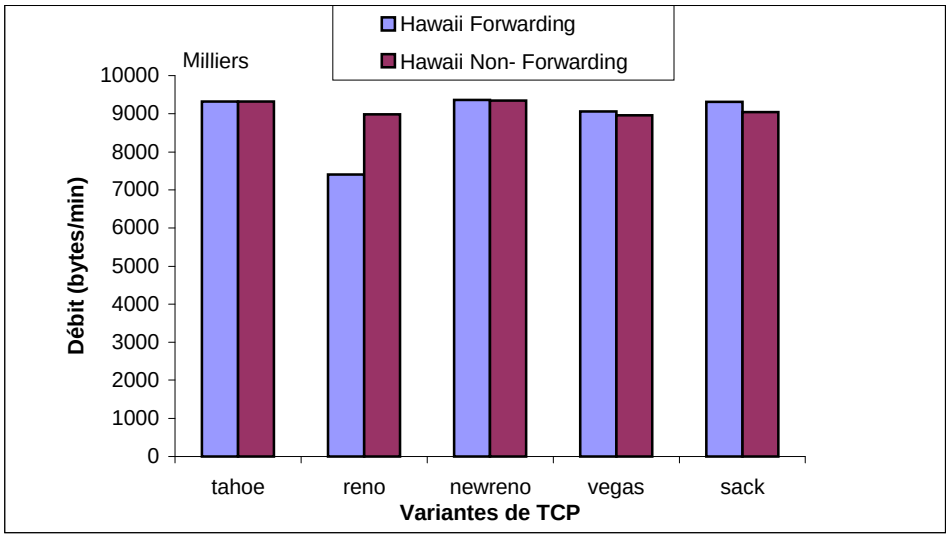


Figure 6.25 : Débit pour différentes implémentations de TCP en appliquant les deux schémas de handoff du protocole Hawaii

On constate selon l'histogramme du *forwarding* handoff que pour les quatre variantes de TCP (Tahoe, Vegas, Newreno, Sack) le débit TCP est presque égal sauf pour le TCP Reno où le débit est moins important. Pour le *non-forwarding* handoff, on constate que tous les débits TCP se rapprochent, il n'y a pas une différence importante. On remarque également que les débits reçus pour le TCP Tahoe et Newreno sont égaux quelque soit le schéma de handoff. Par contre les débits reçus pour le TCP Vegas et TCP Sack sous le *forwarding* schéma sont légèrement supérieurs à ceux reçus sous le schéma *non-forwarding*. Cependent, les débits reçus pour le TCP Reno sous le schéma *non-forwarding* sont considérablement supérieurs à ceux reçus sous le schéma *non-forwarding*.

6.5 Comparaison

Sachant que le *forwarding* handoff et le *hard* handoff sont utilisés pour des réseaux où l'hôte mobile peut émettre / recevoir d'une seule station de base comme dans le cas d'un réseau TDMA (*Time Division Multiple Access*) et que le *non-forwarding* et le *semisoft* handoff sont utilisés pour des réseaux où l'hôte mobile peut émettre / recevoir de deux stations de base ou plus simultanément pour une courte durée, comme dans le cas d'un réseau CDMA (*Code Division Multiple Access*), on compare donc les deux protocoles Cellular IP et Hawaii en comparant le *semisoft* handoff avec le *non-forwarding* handoff, et le *hard* handoff avec le *forwarding* handoff. On maintient la même topologie du réseau pour les différents schémas de handoff.

6.5.1 Vitesse de l'hôte mobile

On constate de la figure 6.26 les points suivants :

- Plus la vitesse de l'hôte mobile augmente plus le maximum numéro de séquence sous le forwarding handoff est supérieur au maximum numéro de séquence sous le hard handoff. (on

prend par exemple à la vitesse 100 unités/sec le maximum numéros de séquence sous le hard handoff est de 3382 et sous le hawaii forwarding s'est pratiquement le double 6837).

- Le semisoft handoff prend le dessus sur le non-forwarding handoff entre les vitesse (20-50unités/sec). Au-delà de la vitesse 50unités/sec c'est l'inverse. Plus on augmente la vitesse de l'hôte mobile plus le maximum numéro de séquence sous le non-forwarding handoff est supérieur à celui sous le semisoft handoff.

(exemple : à la vitesse 30 unités/sec : le maximum numéros de séquence sous le semisoft handoff est de 9325 et sous le hawaii non-forwarding est de 8777. Par contre à vitesse 100 unités/sec : le maximum numéros de séquence sous le semisoft handoff est de 4269 et sous le hawaii non-forwarding est de 6942.)

- De façon plus générale, plus on augmente la vitesse du mobile plus les schémas de handoff de Hawaii sont plus performants que les schémas de handoff de Cellular IP.

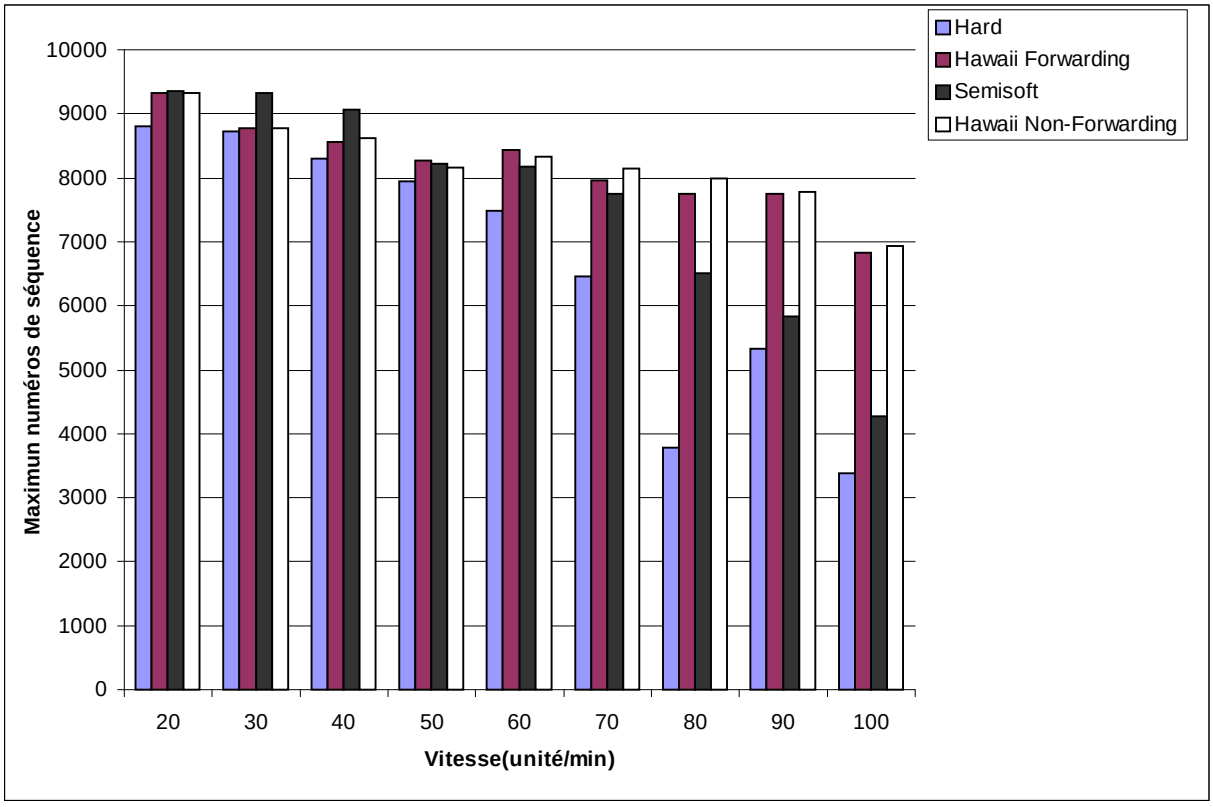


Figure 6.26 : Comparaison de l'effet de la vitesse sur les schémas de handoff de Cellular IP et de Hawaii

Le tableau 6.1 récapitule les performances des différents schémas de handoffs (à savoir le hard handoff et le semisoft handoff de Cellular IP ainsi que le forwarding et le non forwarding handoff de Hawaii) par rapport aux vitesses de l'hôte mobile selon les numéros de séquence maximum.

	Vitesse	20	30	40	50	60	70	80	90	100
Hard	Max n° seq	8804	8735	8303	7955	7489	6468	3785	5332	3382
	Performance	Assez B	Assez B	Assez B	Moyen	Moyen	Passable	Médiocre	Médiocre	Médiocre
Forwarding	Max n° seq	9331	8770	8556	8276	8432	7962	7744	7747	6837
	Performance	Bien	Assez B	Assez B	Assez B	Assez B	Moyen	Moyen	Moyen	Passable
Semisoft	Max n° seq	9361	9325	9061	8221	8171	7755	6515	5831	4269
	Performance	Bien	Bien	Bien	Assez B	Assez B	Moyen	Passable	Médiocre	Médiocre
Non Forwarding	Max n° seq	9330	8777	8625	8168	8338	8153	7991	7780	6942
	Performance	Bien	Assez B	Assez B	Assez B	Assez B	Assez B	Moyen	Moyen	Passable

Tableau 6.1 : Performances des différents schémas de handoffs par rapport aux vitesses de l'hôte mobile selon les numéros de séquence maximum

Légende (Max n° seq)
Bien : supérieur à 9000
Assez bien : [8000 - 9000 [
Moyen : [7000 - 8000 [
Passable : [6000 - 7000 [
Médiocre : inférieur à 6000

La Figure 6.27 présente une comparaison de l'effet du nombre de handoffs sur les schémas de handoff de Cellular IP et de Hawaii. On constate de cette figure les points suivants :

- Plus le nombre de handoffs augmente plus le débit TCP reçu par l'hôte mobile sous le forwarding handoff est supérieur au débit TCP reçu par l'hôte mobile sous le hard handoff. (Par exemple quand le nombre de handoff atteind 19 handoffs /min le débit sous le hawaii forwarding handoff est pratiquement le double de celui sous le hard handoff).

- Le semisoft handoff prend le dessus sur le non-forwarding handoff dans l'intervalle (8–13 handoffs/min). Au-delà de 13 handoffs/min c'est l'inverse. Plus le nombre de handoffs par minute augmente plus le débit TCP sous le non-forwarding handoff est supérieur à celui sous le semisoft handoff.

- De façon plus générale, plus le nombre de handoffs augmente plus les schémas de handoff de Hawaii sont plus performants que les schémas de handoff de Cellular IP.

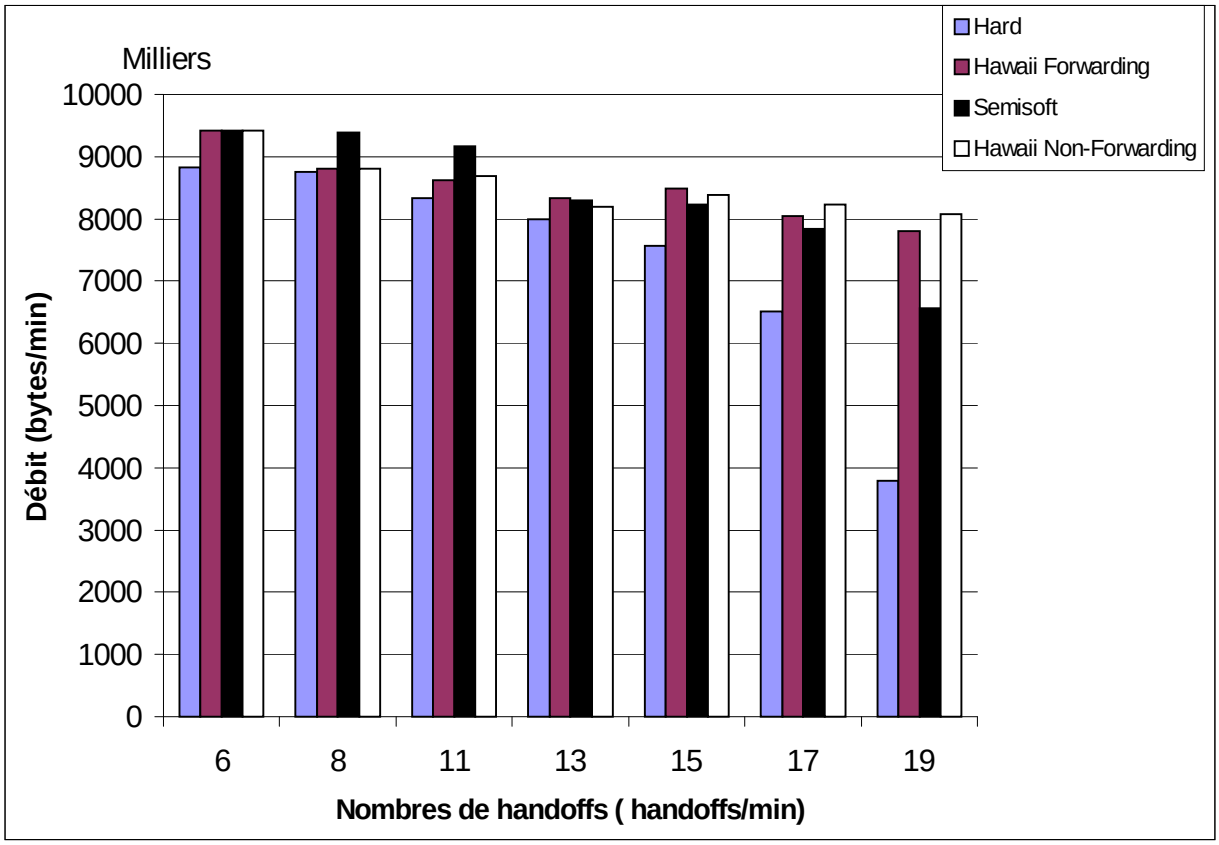


Figure 6.27 : Comparaison de l'effet du nombre de handoffs sur les schémas de handoff de Cellular IP et de Hawaii

Le tableau 6.2 récapitule les performances des différents schémas de handoffs par rapport aux nombre de handoffs par minute effectués par l'hôte mobile selon les débits reçus par ce dernier.

	Nombre de handoffs	6	8	11	13	15	17	19
Hard	Débit	8819000	8767000	8326000	7996000	7573000	6515000	3791000
	Performance	Assez B	Assez B	Assez B	Moyen	Moyen	Passable	Médiocre
Forwarding	Débit	9415000	8817000	8619000	8329000	8485000	8044000	7813000
	Performance	Bien	Assez B	Assez B	Assez B	Assez B	Assez B	Moyen
Semisoft	Débit	9415000	9389000	9174000	8303000	8236000	7834000	6558000
	Performance	Bien	Bien	Bien	Assez B	Assez B	Moyen	Passable
Non Forwarding	Débit	9418000	8816000	8693000	8199000	8381000	8237000	8074000
	Performance	Bien	Assez B	Assez B	Assez B	Assez B	Assez B	Assez B

Tableau 6.2 : Comparaison des performances des différentes variantes TCP par rapport au nombre de handoffs selon les débits reçus par l'hôte mobile

Légende : Débit (bytes/min)

Bien : supérieur à 9000000
Assez bien : [8000000 - 9000000 [
Moyen : [7000000 - 8000000 [
Passable : [6000000 - 7000000 [
Médiocre : inférieur à 6000000

6.5.3 Variantes TCP

La Figure 6.28 présente une comparaison des débits des différentes implémentations de TCP sous différents schémas de handoff de Cellular IP et Hawaii. La vitesse de l'hôte mobile est constante (20unités/sec).

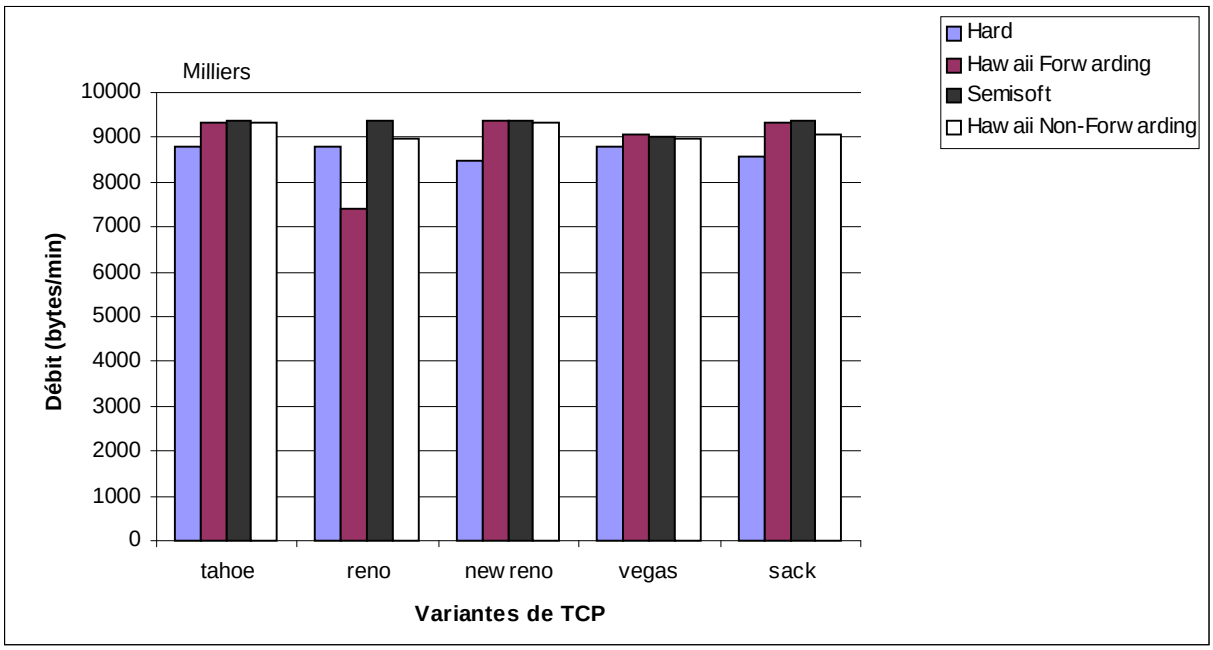


Figure 6.28 : Débit pour différentes implémentations de TCP sous les différents schémas de handoff des protocoles Cellular IP et Hawaii

Le tableau 6.3 résume les performances des différentes variantes TCP étudiées au chapitre 2 par rapport aux différents schémas de handoffs selon les débits reçus par l'hôte mobile.

	Type de handoffs	Hard	Forwarding	Semisoft	Non Forwarding
Tahoe	Débit	8798000	9318000	9357000	9321000
	Performance	Assez bien	Bien	Bien	Bien
Reno	Débit	8776000	7402000	9384000	8988000
	Performance	Assez bien	Moyen	Bien	Assez bien
Newreno	Débit	8475000	9362000	9368000	9342000
	Performance	Assez bien	Bien	Bien	Bien
Vegas	Débit	8793000	9060000	9030000	8964000
	Performance	Assez bien	Bien	Bien	Assez bien
Sack	Débit	8562000	9312000	9395000	9047000
	Performance	Assez bien	Bien	Bien	Bien

Tableau 6.3 : Comparaison des performances des différentes variantes TCP par rapport aux differents schémas de handoffs selon les débits reçus par l'hôte mobile

Légende : Débit (bytes/min)
Bien : supérieur à 9000000
Assez bien : [8000000 - 9000000 [
Moyen : [7000000 - 8000000 [

6.6 Conclusion

Dans ce chapitre, on a simulé le comportement de TCP Tahoe sous le deux schémas de handoff de Cellular IP ainsi que sous les deux schémas de handoff de Hawaii selon certains critères par exemple la vitesse de l'hôte mobile, la taille de la zone de chevauchement. On a également évalué les performances des differentes variantes de TCP sous les differents schémas de handoff. Au travers de ces évaluations, on a pu dégager les remarques suivantes :

- Pour Cellular IP :

- Les performances de TCP sous le Semisoft handoff sont meilleures que celles sous le hard handoff avec une vitesse fixe de l'hôte mobile (20unités/sec).

- Plus le nombre de handoffs par minute augmente, plus les performances de TCP sous les deux schémas de handoffs se dégradent. Ce pendant les performances sous le semisoft restent toujours supérieures à celles sous hard handoff.

- En augmentant la taille de la zone de chevauchement, les performances de TCP sous le Semisoft handoff sont sensiblement supérieures à celle du hard handoff jusqu'à un certain seuil de la taille de la zone de chevauchement (50 unités). Au-delà de ce seuil le nombre de handoffs par minute augmente et les performances de TCP sous le semisoft handoff se dégradent progressivement par contre celles sous le hard handoff subissent une chute agressive.

- En augmentant la vitesse du mobile les performances de TCP se dégradent quelque soit le schémas de handoff (Hard handoff ou Semisoft handoff).

- Quelle que soit la vitesse de l'hôte du mobile, les performances de TCP sous le Semisoft handoff restent meilleures (même si elles se dégradent) que celles sous le hard handoff.

- Quelle que soit la variante de TCP utilisée, les performances sous le semisoft handoff sont supérieures à celles sous le hard handoff.

- Pour Hawaii :

- Les performances de TCP sous les deux schémas de handoff de Hawaii sont identiques avec une vitesse fixe de l'hôte mobile (20unités/sec).

- Plus le nombre de handoffs par minute augmente, plus les performances de TCP sous les deux schémas de handoffs se dégradent. Cependant les performances sous les deux schémas de handoff restent identiques.

- En augmentant la vitesse du mobile les performances de TCP se dégradent quel que soit le schéma de handoff. Cependant les performances sous les deux schémas de handoff restent identiques.

- Quelle que soit la variante de TCP utilisée, les performances de TCP sous les deux schémas de handoff sont semblables.

Par la suite, on a comparé les performances de TCP sous les schémas de handoff de Cellular IP avec celles sous les schémas de handoff de Hawaii. On a dressé quelques tableaux qui récapitulent cette comparaison. On a constaté que :

- Plus le nombre de handoffs augmente plus les schémas de handoff de Hawaii sont plus performants que les schémas de handoff de Cellular IP.

- Plus on augmente la vitesse du mobile plus les schémas de handoff de Hawaii sont plus performants que les schémas de handoff de Cellular IP.

- Sous le semisoft handoff, les performances des différentes variantes de TCP restent globalement stables et supérieures à celles sous les autres schémas de handoff (avec une vitesse fixe de l'hôte mobile 20unités/sec).

On constate également que les résultats de simulations des performances de TCP sous Cellular IP et Hawaii réalisées dans [GHA01], sités ci-dessous, conforment les résultats de simulations (concernant les mêmes critères de comparaison) réalisées dans ce chapitre :

- Pour Cellular IP :

- Vitesse de l'hôte mobile : Plus l'hôte mobile se déplace rapidement, moins il passe de temps dans la région de chevauchement donc la perte de paquets augmente. Dans le semi-soft handoff, l'augmentation de perte de paquets est la plus faible parce que l'hôte mobile peut simultanément recevoir des paquets des nouvelles et anciennes stations de base. Cependant contrairement au schéma semi-soft, l'hôte mobile utilisant un hard peut seulement envoyer et recevoir des paquets d'une station de base à la fois ainsi il ne peut pas recevoir tous les paquets envoyés à lui et perd plus de paquets.
- Nombre de handoff : La performance de TCP se dégrade à mesure que la fréquence du handoff augmente. Le semi-soft handoff réduit la perte des paquets et améliore considérablement le débit du transport par rapport au hard handoff.

- Comparaison de Cellular IP et Hawaii :

- En augmentant le nombre de handoffs les schémas de handoff de Hawaii deviennent plus performants que les schémas de handoff de Cellular IP.
- Les performances du *semi-soft* handoff de Cellular IP et du *non-forwarding* handoff de Hawaii sont approximativement les mêmes en termes de maximum numéro de séquence reçu et ceci en utilisant un même scénario de simulation pour Cellular IP et Hawaii (le nœud mobile se déplace à une vitesse de 20 mètres/sec).

Cependant, contrairement au résultat de simulation réalisée dans [GHA01], les performances du *forwarding* handoff de Hawaii sont meilleures que celles du *hard* handoff de Cellular IP et ceci en utilisant un même scénario de simulation pour Cellular IP et Hawaii (le nœud mobile se déplace à une vitesse de 20 mètres/sec).

Conclusion

Depuis quelques années, les nombreuses évolutions réalisées dans les domaines des terminaux portables et des réseaux sans-fil suscitent un intérêt croissant pour l'informatique mobile. L'utilisation de ces nouveaux environnements introduit de nouvelles problématiques et créé de nouveaux besoins.

Ces environnements présentent, en effet, une hétérogénéité importante et une grande variabilité aussi bien au niveau des moyens d'exécution que des moyens de communication. Les ressources offertes par un terminal portable sont, en général, bien moins importantes que celles que l'on peut trouver dans une station fixe. De plus, la disponibilité de ces ressources n'est pas figée et peut varier en fonction d'ajout ou de suppression de périphériques ou de limitations imposées par des politiques d'économie de la batterie.

Les réseaux de communications sans fil utilisés par ces terminaux présentent également des différences notoires par rapport aux réseaux filaires. La mobilité et les performances espérées dépendent grandement de la structure du réseau et notamment de l'infrastructure déployée et de la portée de communication. De plus, les performances observées sur le lien sans fil sont soumises à d'importantes variations occasionnées par l'environnement proche comme les interférences et déconnexions dues à des changements de cellules ou de réseau.

L'intégration de ces nouveaux environnements aux réseaux traditionnels et surtout Internet pose un nouveau challenge. En effet, l'ensemble des protocoles TCP / IP d'Internet a été, à l'origine, conçu sans aucune considération pour la mobilité des unités de calcul. Sa grande rigidité face aux environnements mobiles réside dans le service de routage offert par la couche IP "*Internet Protocol"* qui permet un routage des données dans un environnement statique.

Afin de supporter la mobilité, le protocole IP a été amélioré pour donner naissance au protocole Mobile IP. L'IETF a donc standardisé le protocole Mobile IP pour intégrer le support de services de mobilité dans les réseaux IP étendus. Mobile IP permet un routage transparent des paquets IP destinés aux nœuds mobiles et permet d'assurer par conséquent la continuité des communications en cours. Il s'applique parfaitement à des besoins de gestion de mobilité peu contraignants en termes de délais de mise à jour, typiquement pour des services dits de "macro mobilité" qui assurent la continuité de service pour un terminal se déplaçant d'un réseau d'accès à un autre.

Cependant, si Mobile IP est utilisé pour gérer la "micro mobilité", c'est à dire la mobilité localisée dans un même réseau d'accès et surtout lorsque ce réseau d'accès est distant du réseau principal du terminal, les délais de diffusion de la nouvelle localisation ainsi que le trafic de contrôle généré pour joindre le réseau principal sont pénalisants. En effet, ils entraînent des interruptions de services de plusieurs secondes, des pertes de paquets de données correspondants aux connexions en cours et des échanges de signalisations qui chargent inutilement le réseau.

Ainsi, plusieurs protocoles de micro mobilité ont été développés (Cellular IP, Hawaii, Ema, TeleMIP …etc). Ces derniers sont conçus pour des environnements où les nœuds mobiles changent leurs points d'attachement à l'Internet si fréquemment que le routage de MIP s'avère insuffisant : surcharge de la signalisation, perte de paquets, livraison des données aux

applications retardées. Ces retards sont directement liés au temps d'aller-retour des messages d'enregistrement. Les protocoles de micro-mobilité aspirent à gérer les déplacements locaux (à l'intérieur des domaines) des nœuds mobiles sans interagir avec MIP, c'est-à-dire cacher au reste de l'Internet les mouvements des nœuds mobiles à l'intérieur d'un domaine. Cela a l'avantage de réduire le retard et la perte des paquets pendant un déplacement et élimine l'enregistrement entre un nœud mobile et son agent mère qui peut être éloigné.

Cellular IP est un protocole de micro-mobilité reposant sur Mobile IP pour la gestion de la mobilité entre domaines. Cellular IP ambitionne de remplacer IP dans le domaine. Il gère les mouvements à l'intérieur d'un domaine et les cache au reste de l'Internet. L'information de routage est totalement distribuée dans tous les nœuds du domaine. Le handoff est traité suivant deux mécanismes : le hard et le semisoft handoff. Cellular IP est prévu pour fonctionner avec un grand nombre de mobile et gérer les handoffs de manière optimisée. Cellular IP interagit avec MIP pour ce qui est de la mobilité entre domaines. Contrairement à Cellular IP, HAWAII ne remplace pas IP mais s'appuie sur lui dans son fonctionnement. Chaque station du réseau doit donc pouvoir fournir les services d'un routeur IP classique plus certaines fonctionnalités de gestion de la mobilité. La gestion de la mobilité se fait de façon très similaire à Cellular IP. Le handoff est traité suivant deux mécanismes : le forwarding et le non-forwarding handoff. Comme Cellular IP, HAWAII présente un support natif de la connectivité passive.

Ces nouveaux protocoles de micro mobilité perturbent le comportement des applications à temps réel et surtout les applications basées sur le protocole TCP. Face aux pertes et aux erreurs et coupures de transmission dues aux caractéristiques des réseaux sans fil, le protocole « TCP », qui assure les transferts fiables de données réagit de la même façon que dans les environnements filiaux, il diminue la taille de la fenêtre de transmission avant de retransmettre les paquets puis initialise les mécanismes d'évitement et de contrôle de congestion et réinitialise le timer de retransmission. Du fait que la perte des paquets n'est pas due à la congestion, ces mesures s'avèrent non utiles et engendrent une réduction non nécessaire de l'utilisation de la bande passante des liens ce qui dégrade les performances de TCP dans les réseaux sans fil.

L'objectif de cette thèse était donc d'évaluer les performances du protocole TCP dans les réseaux de communication sans fil et plus précisément sous les différents schémas de handoff des deux protocoles de micro mobilité à savoir Cellular IP et Hawaii. Pour ce faire, on a étudié le fonctionnement du protocole TCP et plus spécialement les procédures de contrôle et d'évitement de congestion, le protocole de macro mobilité Mobile IP ainsi que les deux protocoles de micro mobilité Cellular IP et Hawaii.

Durant cette étude, on a souligné les problèmes de la mobilité et leurs influences sur les performances du protocole TCP. Ces problèmes peuvent se résumer dans les points suivants : les problèmes dus aux caractéristiques des environnements sans fil telles que les pertes et erreurs de transmission, les bandes passantes limitées, les interruptions des communications suite à l'atténuation des signaux …etc, ainsi que le phénomène de passation ou de changement de point d'attachement. On a également présenté les études réalisées concernant la micro mobilité Certaines comparent uniquement les différents protocoles de micro mobilité et d'autres évaluent les performances des protocoles TCP et UDP sous le protocole de micro mobilité Cellular IP.

Suite à cette étape, on a étudié le simulateur de réseau « NS-2 » le plus utilisé dans la communauté de recherche. Il implémente en C++ et OTcl tout un ensemble de classes pour définir les concepts de nœuds, liaisons et agents. La structure d'un nœud est fortement inspirée par le modèle en couche OSI. La mobilité a été introduite dans NS-2 dans sa version 2. Plusieurs extensions ont été ajoutées notamment Mobile IP, Cellular IP, Hawaii…etc.

Par la suite, on est passé à une phase d'expérimentation dans laquelle on a testé et évalué en utilisant le simulateur NS-2 les performances du protocole TCP sous les différents schémas de handoffs des deux protocoles de micro mobilité.

Au terme de ces simulations, on a dégagé quelques conclusions que l'on a résumées dans des tableaux. A travers ces tableaux comparatifs, on a constaté que les différents schémas de handoff des protocoles de micro mobilité utilisent des méthodes et techniques différentes pour résoudre le problème de performances de TCP dans les réseaux sans fil. Chaque méthode présente des avantages et des inconvénients selon des critères de comparaisons.

On a constaté également que les protocoles de micro-mobilité offrent des solutions performantes pour le support de la mobilité localisée. Ils présentent l'avantage de résoudre les principales limites de Mobile IP sans complexifier le mécanisme de gestion de la mobilité. Cependant, des améliorations sont encore possibles surtout en ce qui concerne la phase de décision de l'exécution d'un handoff, la procédure de détection du mouvement et le temps d'exécution du handoff.

En effet, dans ces solutions de mobilité, la phase de prise de décision de l'exécution du handoff est exclue du protocole. Un nœud mobile subit simplement le handoff décidé au niveau liaison. Par exemple, à chaque fois que la qualité de la liaison radio avec son point d'accès courant se dégrade, le mobile change de point d'accès. Au niveau IP, le nœud mobile doit attendre la réception d'une annonce de routeur pour détecter son mouvement et entamer la mise à jour de sa localisation. Il y a donc un intervalle de temps non négligeable entre le déplacement physique du nœud mobile (couches 1 et 2 du modèle OSI) et la détection du mouvement IP (couche 3), ce qui peut accroître le temps d'exécution du handoff.

La conclusion de cette étude, est que la conception d'une solution d'amélioration des performances de TCP dans les réseaux sans fil doit tenir compte de tous les facteurs et limitations physiques imposés par l'environnement sans fil ainsi que la faisabilité d'implémentation de la solution dans l'internet.

Dans la continuité du travail présenté, on pourrait tout d'abord évaluer les performances du protocole TCP sous d'autres protocoles de micro mobilité par exemple TeleMIP et EMA afin de dégager une comparaison plus globale des performances de TCP sous ces différents protocoles. Il serait également intéressant d'étudier le comportement du protocole TCP dans les réseaux ad-hoc. Finalement, on pourrait étudier et évaluer les différents protocoles de sécurité dans le cas des réseaux sans fil.

Annexes

Annexe 1 : Les différentes techniques d'évaluation des performances 87
1.1 Mesures directes 87
1.2 Les techniques d'analyse mathématique 87
1.3 La simulation 88
1.4 Quelques outils de simulation spécialisés dans l'analyse des réseaux 89

Annexe 2 : Formats des paquets 92
2.1 Formats des paquets TCP 92
2.2 Formats des paquets Cellular IP 93
2.3 Formats des messages Hawaii 94

Annexe 3 : MIPv4 et MIPv6 96
3.1 MIPv4 96
3.1.1 Découverte des agents de mobilité 96
3.1.2 Enregistrement auprès de l'agent mère 96
3.1.3 Communication 97
3.2 MIPv6 98
3.2.1 Fonctionnalités requises 98
3.2.2 Découverte des routeurs d'accès 98
3.2.3 Enregistrement 99
3.3 Comparaison de Mobile IPv4 avec Mobile IPv6 99

Annexe 4 : Présentation du simulateur réseau NS-2 100
4.1 Introduction 100
4.2 Architecture et implémentation 100
4.3 Point de vue de l'utilisateur 104
4.4 Statistiques et visualisation 105
4.5 Extension pour la mobilité 107

Annexe 1 : Les différentes techniques d'évaluation des performances

Les différentes techniques d'évaluation des performances sont schématisées dans la figure 29. Elles peuvent être classées en trois grandes catégories : l'obtention de mesures directes sur le réseau (ou sur un prototype), les techniques analytiques et numériques et la simulation [JAI91, JAI86].

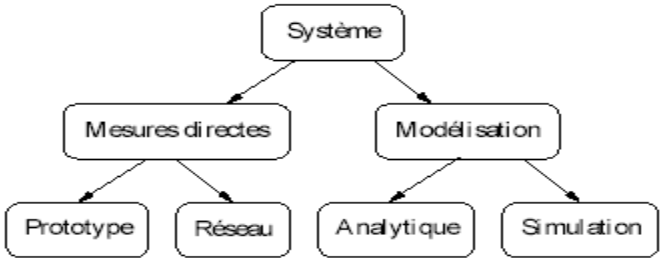


Figure 29 : Techniques d'évaluation des performances de réseau

On décrit maintenant ces différentes méthodes d'évaluation.

1.1 Mesures directes

Les mesures directes de performances peuvent être obtenues à partir de réseaux déjà opérationnels ou de prototypes de systèmes en cours de développement. Cette technique consiste à extraire des informations du réseau au travers de moniteurs ou analyseurs qui scrutent le trafic y véhiculé ou au travers les perturbations observées sur des flots de contrôle introduits dans le réseau.

Les mesures directes conforment la base des autres techniques d'évaluation des performances car elles fournissent les données avec lesquelles les modèles du système peuvent être conçus et validés. C'est la seule technique qui peut offrir « l'image réelle » de l'état d'un système réel en tenant compte de toutes les caractéristiques de celui-ci. Elle est très utilisée dans une gestion préventive : elle permet, par exemple, d'estimer avec précision la qualité de service offerte par le réseau et de disposer d'une base sur laquelle de probables extensions du réseau peuvent être projetées. Néanmoins, cette méthode a plusieurs inconvénients. D'une part, le trafic du réseau est extrêmement variable et imprévisible. Par conséquent, les échantillons du trafic obtenus à un moment donné ne permettent pas toujours de prévoir le comportement du réseau dans d'autres conditions. Par exemple, si le réseau utilise des mécanismes de contrôle de congestion le trafic observé quand le réseau est congestionné est à priori très différent du trafic que serait généré par les sources ayant eu plus de ressources. On n'est donc pas en mesure de planifier de nouveaux réseaux et services du réseau. D'autre part, cette technique est l'option la plus coûteuse : selon le système et selon le niveau de détail désiré, les mesures directes peuvent nécessiter une connaissance approfondie de l'environnement, du personnel spécialisé et une instrumentation délicate et lourde à mettre en œuvre. Enfin, on ne peut pas recourir à cette approche si le réseau n'existe que sur les planches à dessin, ou, autre situation, pour analyser un réseau existant dans les conditions d'opération (de charge, par exemple) prévues dans un futur proche mais différentes de celles d'aujourd'hui.

1.2 Les techniques d'analyse mathématique

L'évaluation analytique des performances consiste à représenter le système par un modèle mathématique (par exemple, un système d'équations) qui peut être résolu directement ou au travers des méthodes numériques. Il existe plusieurs formalismes et méthodes souvent utilisés pour évaluer les performances de réseau ; on peut citer, entre autres : les méthodes de graphes,

les réseaux de Petri, les grammaires formels, les modèles fluides et les files d'attente (et les réseaux de files d'attente). Par exemple, avec cette dernière méthode, un nœud de réseau peut être représenté par une file d'attente avec une certaine discipline de service. On suppose typiquement que le temps entre arrivées de paquets suit une loi de distribution définie par un ensemble de paramètres. On se donne également une distribution paramètrée pour la longueur des paquets. La théorie des files d'attente permet d'obtenir des mesures de performance telles que la moyenne du temps de service, la probabilité de perte et l'occupation moyenne du nœud comme des fonctions simples de ces paramètres des lois, au prix d'hypothèses précises sur ces dernières ainsi que sur les mécanismes dans les files. L'avantage principal de ces techniques est qu'elles sont très efficaces, surtout lorsque la solution analytique existe. En plus, elles permettent d'explorer un modèle de réseau que l'on contrôle parfaitement. Ceci entraîne en général une meilleure connaissance de la dynamique du système. Par exemple, certains auteurs considèrent qu'une partie non négligeable de l'énorme succès qu'ont connu les réseaux de commutation de paquets est due à l'existence de modèles mathématiques de performance de ces réseaux. Toutefois, la modélisation analytique implique un niveau d'abstraction considérable. Ces abstractions peuvent parfois masquer des phénomènes ayant une incidence majeure sur les mesures de performance évaluées. Aussi, cette technique nécessite une bonne connaissance du système et de fortes compétences en modélisation afin de maintenir la capacité à traiter les modèles analytiquement. Malheureusement, la complexité croissante des réseaux de communication rend cette tâche impossible. On ne peut pas utiliser ces méthodes sans faire beaucoup de suppositions simplificatrices qui peuvent compromettre la qualité des résultats. Ceci restreint, donc, leur champ d'application.

1.3 La simulation

Face à la complexité des réseaux de communication, la simulation a été et reste toujours l'outil privilégié pour évaluer les performances de réseau et pour étudier le comportement des protocoles de réseau. C'est, de loin, la méthode de choix en milieu industriel. Ici, les différents composants de réseau (source, lien, nœud, protocole, . . .), sont modélisés en termes d'instructions qui sont interprétées par une entité de coordination, le simulateur.

La simulation est la plus flexible, elle permet, en principe, d'étudier les modèles de réseau avec n'importe quel niveau de détail. De plus, on peut facilement modifier les conditions de fonctionnement du réseau et comparer les mesures d'intérêt d'un scénario à l'autre. Pour un non spécialiste, la simulation a plus de crédibilité que les modèles analytiques car elle est « plus proche » du système réel et nécessite, en principe, moins de simplifications et quasiment pas d'hypothèse spécifique. Or, la souplesse offerte par la simulation se paye cher : les simulations sont souvent gourmandes en ressources passives (par exemple, mémoire vive) et en temps de calcul lorsque les modèles utilisés sont un tant soit peu réalistes ou bien si l'on souhaite obtenir des résultats avec un niveau de précision élevé. Par exemple, au cours du processus de simulation, on peut estimer les mesures de performance d'intérêt, mais à la différence des techniques analytiques, les observations de la simulation sont toujours entourées d'incertitude. Cette incertitude provient du fait que les résultats de la simulation ne sont qu'un échantillon d'un nombre potentiellement infini d'observations. Une manière d'améliorer la précision des résultats de la simulation, consiste à répéter l'expérience plusieurs fois, donc à employer plus de temps. Il est donc d'un grand intérêt de déceler des techniques visant à accélérer le processus de simulation tout en préservant les multiples avantages de cette méthode.

1.4 Quelques outils de simulation spécialisés dans l'analyse des réseaux

Il y a une très grande variété d'outils de simulation modélisant des composants de réseaux de communication et cette offre ne cesse de croître. On présente dans ce qui suit quelques exemples:

BONeS Designer. BONeS Designer (Block Oriented Network Simulator) est un langage de simulation et un environnement de simulation qui contient plusieurs facilités permettant la modélisation graphique et la simulation d'un réseau de communication[LAW94]. Les briques de base pour construire un modèle sont des structures de données et des blocs fonctionnels. Par exemple, on utilise les structures de données pour représenter un paquet et on spécifie un diagramme de blocs interconnectés pour décrire le parcours du paquet à travers le réseau. Ce logiciel propose une bibliothèque significative de blocs fonctionnels pour représenter des composants de réseau, des générateurs de trafic, etc. De plus, il permet d'ajouter de nouveaux types de nœuds, de liens ou de protocoles non fournis au départ. Les résultats de la simulation peuvent être directement liés aux blocs de représentation graphique ou ils peuvent être triés pour représenter les résultats sous forme des tableaux. BONeS Designer était produit et commercialisé par Cadence Design Systems, Inc. Récemment, cette compagnie a décidé d'arrêter le développement du simulateur.

COMNET III est un progiciel commercial de simulation qui permet d'analyser tout type de réseau[COM??]. Il est produit et distribué par CACI Products Company. Il possède une interface graphique sophistiquée à travers laquelle on dessine la topologie du réseau en sélectionnant et en glissant des icônes d'objets de réseau. Moyennant des fenêtres de dialogue, on peut initialiser les paramètres des objets. L'outil peut créer automatiquement la topologie du réseau à partir d'informations collectées par d'autres outils de gestion (par exemple, à partir de moniteurs et d'analyseurs de réseau). L'interface graphique permet aussi de visualiser une animation en « temps-réel » du déroulement de la simulation. De plus, elle permet de réaliser des simulations interactives : on peut initier une simulation, observer les résultats produits au fur et à mesure de l'exécution, suspendre la simulation pour modifier certains paramètres et continuer la simulation afin d'observer immédiatement l'effet des modifications. Les principaux blocs d'objets sont des nœuds (e.g., équipements terminaux, routeurs, commutateurs), des liens de communication, des protocoles et des générateurs de trafic pour modéliser des applications de réseau. Bien que la bibliothèque des modèles fournis soit très complète, il existe une version qui permet de modifier les objets proposés et d'ajouter de nouveaux objets.

INSANE Développé par Bruce MAH à l'Université de California, Berkeley, INSANE (Internet Simulated ATM Networking Environement) est un simulateur conçu pour tester divers algorithmes d'IP au-dessus d'ATM [MAH??]. Il fournit des modèles des protocoles IP, TCP et UDP. Pour la génération de trafic, il est fourni avec des traces de trafic dérivées de mesures empiriques ainsi que de divers modules modélisant les applications principales d'Internet (Telnet, FTP, WWW, . . .). Le simulateur événementiel est écrit en C++ mais les configurations des modèles sont spécifiées en TCL. INSANE a été conçu pour simuler des réseaux de taille considérable (de l'ordre de 1 000 nœuds). Les résultats sont traités « off-line » après la simulation. Si INSANE est séquentiel, il possède une interface graphique écrite en TCL/Tk pour superviser l'exécution d'un ensemble de simulations en parallèle de répliques indépendantes.

NetMaker NetMaker MainStation est un ensemble d'outils pour modéliser, concevoir et analyser des réseaux hétérogènes (i.e., ayant des équipements de divers fournisseurs et de diverses technologies) [NET??]. Produit par Make Systems, Inc., il est axé sur le marché des

fournisseurs des services de réseau. À présent, il peut modéliser des réseaux IP, Frame Relay et ATM. Planner est l'outil d'évaluation des performances de NetMaker. Il possède une riche bibliothèque de nœuds de réseau qui prend en compte des spécificités des fournisseurs d'équipement (e.g., Cisco, Lucent, Bay Networks). Planner utilise une approche hybride : la création des tables de routage, l'établissement de trajectoires et la dynamique des protocoles de routage sont simulés avec une approche orientée événement, tandis que l'analyse des performances est faite au travers d'outils analytiques. Les modèles de réseau sont basés sur trois types d'objets : des nœuds, des liens et des demandes. Ces dernières sont des profils d'application paramétrables (e.g., taille des paquets, longueur moyenne d'une rafale de paquets, débit d'émission) dont le but est de modéliser les applications de réseau.

NIST Développé à l'institut NIST (National Institute of Standards and Technology) aux États-Unis, il s'agit d'un simulateur événementiel de réseaux ATM/HFC écrit en langage C[NIS??]. Le simulateur a été développé avec deux objectifs principaux : 1) l'évaluation des performances de réseau en fonction du dimensionnement des équipements (liens, nœuds) ; 2) l'analyse des protocoles liés à la technologie ATM. NIST possède une interface graphique à travers laquelle on peut créer des topologies de réseau, assigner des valeurs aux paramètres des composants, visualiser l'évolution de la simulation et analyser les résultats de la simulation. Les composants offerts sont des équipements terminaux, des liens, des commutateurs et des sources de trafic. Les routes sont point à point et statiques, prédéfinis au moment où la topologie est définie.

OPNET. OPNET (Optimized Network Engineering Tools) est un produit commercial très connu développé par MIL3, Inc[OPN??]. Il est composé d'un ensemble d'outils orientés vers la modélisation et l'évaluation des performances des réseaux de communication et des systèmes distribués. OPNET suit une approche hiérarchique de trois niveaux pour modéliser un système : i) le niveau processus est le niveau le plus fin. On y spécifie le comportement des protocoles, des ressources, des applications et des politiques de gestion des files d'attente à travers des automates à état fini. Les états et transitions décrivent l'évolution du processus en réponse aux événements reçus. À chaque état, on peut associer une routine écrite en Proto-C, une extension du langage C. ii) Au niveau nœud, on définit l'architecture des nœuds de réseau à travers des diagrammes de flux de données en ajoutant aux processus de niveau inférieur d'autres modules (processeurs, files d'attente, générateurs de trafic, etc.). iii) Au niveau réseau on détermine la topologie du réseau composé des nœuds et de divers types de liens de communication (liens point à point, de bus, liens sans fils, etc.). La topologie est normalement spécifiée à travers une interface graphique de haute qualité. Une fois le modèle spécifié, il doit être compilé de manière à produire un simulateur événementiel du système. Pendant l'exécution du simulateur, des données importantes sont collectées. Une fois la simulation terminée, les résultats peuvent être synthétisés et éventuellement montrés graphiquement. En plus des éditeurs graphiques de haute qualité, OPNET possède une bibliothèque très complète de composants et de protocoles de réseau (ATM, TCP/IP, Frame Relay, FDDI, réseaux satellitales, . . .).

QNAP (Queueing Network Analysis Package) est un langage de description et un outil d'analyse de réseaux de files d'attente[MOD??]. Il est issu de la recherche menée à l'INRIA dans les années 70 et 80, principalement, et maintenant il est distribué et maintenu par la société Simulog. Le système à évaluer est modélisé par un réseau de stations. Une station est composée d'une file d'attente, d'un ou de multiples serveurs attachés à cette file, d'une discipline de service et d'un mécanisme de routage pour véhiculer les clients à travers le réseau. QNAP dispose de plusieurs modules de résolution pour analyser le réseau de files d'attente.Un nombre limité de configurations simples peut être résolu analytiquement par des solveurs exacts ou approchés.

Pour des configurations plus complexes, il possède un solveur Markovien. Si cela ne suffit pas, par exemple, à cause d'une explosion d'états du système, QNAP analyse le réseau au travers d'une simulation à événements discrets.

GloMoSim (*Global Mobile Information System Simulator*) est un simulateur réseau modulaire pour la simulation et l'évaluation des performances des réseaux sans fil (locaux ou étendus) [GLO??]. Il utilise la simulation parallèle à événements discrets fournie par PARSEC (*Parallel Simulation Environment for Complex System*) développé par PCL (*Parallel Computing Laboratory*) de l'université UCLA (*University of California at Los Angles*). Parsec permet de bien séparer la description du modèle de la simulation de sa nature d'exécution (parallèle ou séquentielle). Pour réaliser une simulation, GloMoSim utilise une approche simple. En effet si on suppose que chaque nœud dans la simulation est une entité à part, alors on remarque non seulement une dégradation mais aussi une limitation des performances de la simulation car l'initialisation d'un millier d'entités peut influer sur le temps d'exécution et sur l'utilisation de l'espace mémoire. Pour palier à ce problème, GloMoSim fait recours à une approche d'agrégation de nœuds où une seule entité peut simuler divers nœuds réseaux à la fois. Chaque nœud a sa propre structure de données et son code doit s'exécuter sans interférence ni violation d'accès aux structures de données des autres nœuds. Chaque nœud à son propre état et l'ensemble des états complets de tous les nœuds est maintenu par l'entité. Cette dernière approche permet d'augmenter le nombre de nœuds dans une simulation sans augmenter le nombre d'entité. Une entité n'est autre qu'une zone géographique où chaque nœud est déterminé par ses coordonnées dans cette zone. GloMosim a en plus une structure de couches proche de celle du modèle OSI. En effet, chaque entité intègre les diverses couches réseaux qui peuvent être simulées sous forme de fonction. Au début de la simulation une fonction d'initialisation est appelée pour chaque couche de chaque nœud. A la réception d'un message, une couche exécute la tâche demandée. A la fin de la simulation, une nouvelle fonction est lancée pour mettre fin à la simulation et collecter les statistiques désirées. Pour faciliter la communication entre les diverses couches réseaux, un ensemble d'API (*Application Programming Interface*) est spécifié.

NS est un simulateur très répandu orienté vers la recherche sur les réseaux TCP/IP. (voir annexe 4). *Network Simulator* est un simulateur réseau de LBNL (*Lawrence Berkely National Laboratory*). Il est écrit en C++ et utilise OTcl (*Object Oriented Tool Command Language*) comme interface de commande et de configuration. NS permet de simuler des larges réseaux filaires ou sans fil. Le modèle sans fil est essentiellement composé d'un objet MobileNode qui permet de simuler divers types de réseaux sans fil avec la plupart des couches du modèle OSI. Le simulteur GloMosim est un simulateur spécifique pour les réseaux mobiles et qui est facile à utiliser. Il se base sur une approche de simulation parallèle à événements discrets lui offrant une rapidité d'exécution. Par contre NS est un simulateur réseau mixte pour tout type de réseaux filaires ou autres. Il utilise une structure orientée objet et utilise un langage de configuration portable(OTcl)lui permettant le changement rapide des configurations de simulation. Tous les modèles de simulations de GloMosim peuvent être utilisés par NS.

Annexe 2 : Formats des paquets

2.1 Formats des paquets TCP

Les différents champs du paquet TCP sont :

• Numéros de ports : le *port source* et le *port destination* permettent, sur 16 bits, d'identifier les applications, client et serveur, s'exécutant sur les deux machines. En effet une adresse IP permet d'identifier une seule machine sur l'Internet. Or sur cette machine il peut y avoir plusieurs applications qui utilisent le service TCP/ IP. Le numéro de port vient alors s'ajouter à l'adresse IP pour identifier de façon unique une application tournant sur une machine spécifique.

▪ Numéro de séquence : indique, sur 32 bits, le numéro de séquence du premier octet des données.

• Acquittement : contient, sur 32 bits, le numéro de séquence du prochain octet attendu. Il est à noter que le numéro de séquence fait référence au flot de données dans le même sens que les données, tandis que l'acquittement fait référence au flot de sens contraire.

• Déplacement : donne la taille, en mots de 32 bits, de l'en-tête du paquet TCP.

• Réservé : les bits *réservés* sont prévus pour un usage futur.

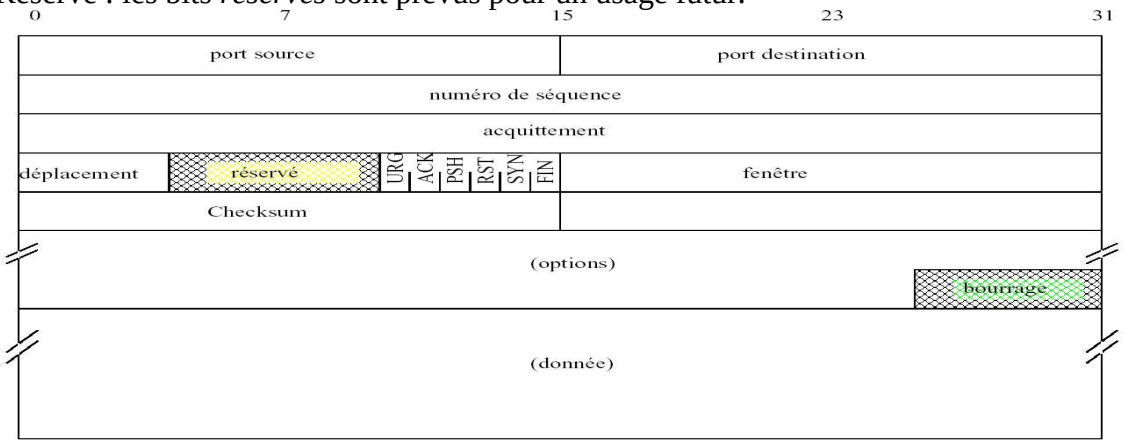


Figure 30 : Format des paquets TCP

• Drapeaux : l'entête TCP contient 6 flags de taille un bit chacun. Ces drapeaux permettent de définir des types de messages ainsi que la validité de certains champs. Les différents drapeaux ainsi que leur signification s'ils sont à 1 est comme suit :

URG : le pointeur de données urgentes est valide.
ACK : le champ acquittement est valide.
PSH : ce segment impose de délivrer toutes les données en attente à l'application.
RST : demande de fermeture de la connexion à cause d'une erreur irrécupérable.
SYN : ouverture de connexion.
FIN : fermeture de la connexion.

• Fenêtre : indique, sur 16 bits, à l'autre extrémité la quantité maximale de données que peut envoyer l'émetteur avant d'être obligé d'attendre des acquittements. Ce champ permet de freiner l'émission des paquets si le récepteur ne dispose pas d'espace mémoire. En effet, pour chaque instance de communication, TCP réserve un espace mémoire pour stocker les paquets avant que l'application locale ne les consomme. Il se trouve que les applications ne consomment pas les données au rythme de leur arrivée. Pour éviter de perdre inutilement des données, le récepteur contrôle le flux de données de l'émetteur en utilisant le champ fenêtre.

• Checksum : codé sur 16 bits, est calculé non seulement pour l'en-tête du paquet comme c'est le cas pour IP, mais aussi pour la partie donnée du paquet. Le checksum TCP protège donc contre

les erreurs qui peuvent toucher tout le paquet TCP. Le checksum est calculé de façon similaire à celui de l'entête IP.

• Pointeur message urgent : indique, sur 16 bits, la position du dernier octet d'un message urgent.

• Options : permet d'échanger des données optionnelles pendant la phase d'établissement de la connexion ou pendant la phase de transfert des données. L'option la plus couramment utilisée lors de l'établissement d'une connexion est la taille maximale d'un segment1 (MSS : Maximum Segment Size. D'autres options ont été définies dans le RFC 1323 : les estampilles de temps permettant de calculer le RTT et dans le RFC 2018 permettant les retransmissions sélectives en cas de pertes.

▪ bourrage (*Padding*) (longueur variable) : Les octets de bourrage terminent l'en-tête TCP : de sorte que le nombre d'octets de celle – ci soit toujours multiple de 4 (32 bits), de sorte que l'offset de données marquée dans l'en-tête corresponde bien au début des données applicatives.

2.2 Formats des paquets Cellular IP

➢ **Structure du signal beacon**

Les éléments d'information transportés dans les balises beacon incluent :

- Les paramètres de niveau 2 relatifs à la station de base,
- Un identificateur du réseau cellular IP,
- L'adresse IP de la passerelle,
- L'identificateur de la paging area pour la localisation des mobiles inactifs.

▪ **Paquet de données :** Cellular IP transmet les paquets IP standards sans modification, segmentation, encapsulation ou tunneling.

▪ **Structure du route-update packet :** Le paquet route-update est un paquet ICMP dans lequel :

– L'adresse IP source est celle du hôte mobile émetteur,

– L'adresse IP destination est celle de la passerelle,

– Le Type indique 'un paquet Cellular IP de contrôle' et le Code indique 'route-update'.

Le reste du paquet transporte des informations d'authentification et de contrôle encodées sous le format suivant (Figure 31).

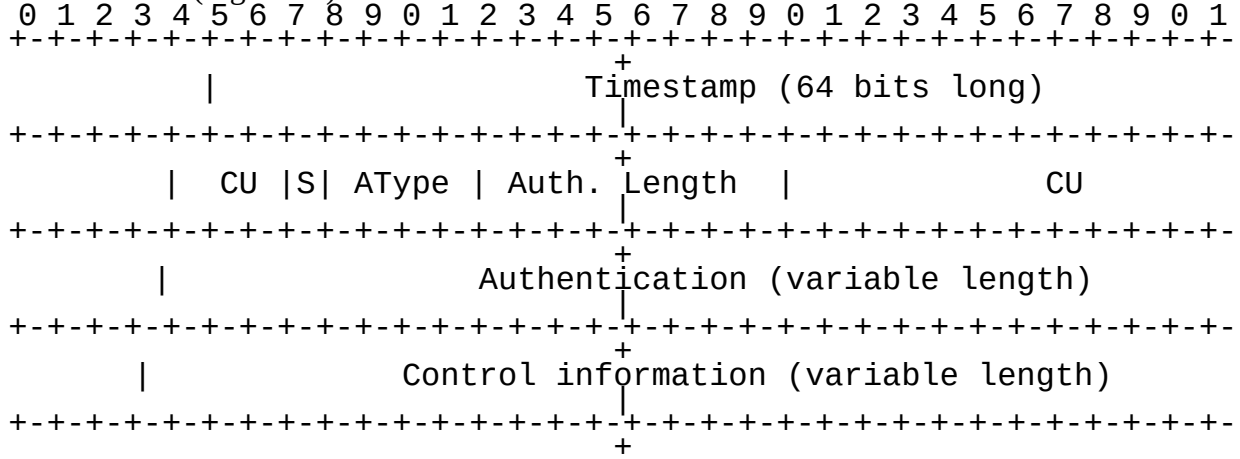


Figure 31: Structure du message route-update

Timestamp contient une estampille utilisée pour déterminer l'ordre dans lequel les update-packets sont envoyés. Ce champ est mis en format selon le protocole Network Time Protocol [MIL92].

CU non utilisé (Currently Unused), mis à zéro.

S flag mis à 1 pour indiquer le semi-soft handoff. Par défaut sa valeur est 0. Tout nœud Cellular IP ne supportant pas les semi-soft handoffs doivent ignorer ce bit.

Atype dénote la méthode d'authentification utilisée. La méthode d'authentification par défaut est décrite dans [MET95]. Toutes les méthodes d'authentification utilisent le champ timestamp.

Auth. Length dénote la longueur de l'information d'authentification (bytes).

Authentication contient l'information d'authentification.
Control information est encodé selon le format Type-Length-Value :

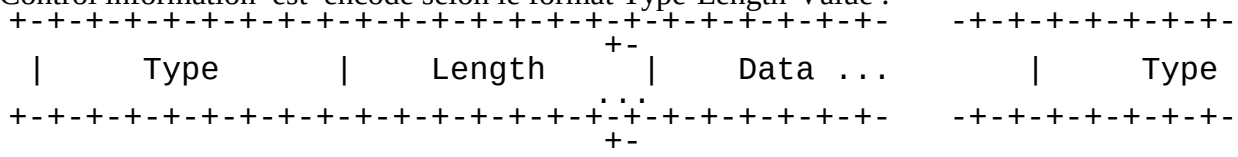


Figure 32: Structure du champ control information

Type indique le type particulier de l'information de contrôle.
Length indique la longueur (en bytes) du champ suivant « Data ».
Data Ce champ est de longueur variable. La signification, format et la longueur de champ sont déterminés par les champs Type et Length.

- **Structure du paging-update packet :** Le paquet paging-update est un paquet ICMP dans lequel : - L'adresse IP source est celle du hôte mobile émetteur,
  - L'adresse IP destination est celle de la passerelle,
  - Le Type indique 'un paquet Cellular IP de contrôle' et le Code 'paging- update'.

Le reste du paquet transporte des informations d'authentification et de contrôle sous le même format que celui du route-update packet. Le champ S flag est mis à 0 pour les paging-update packets.

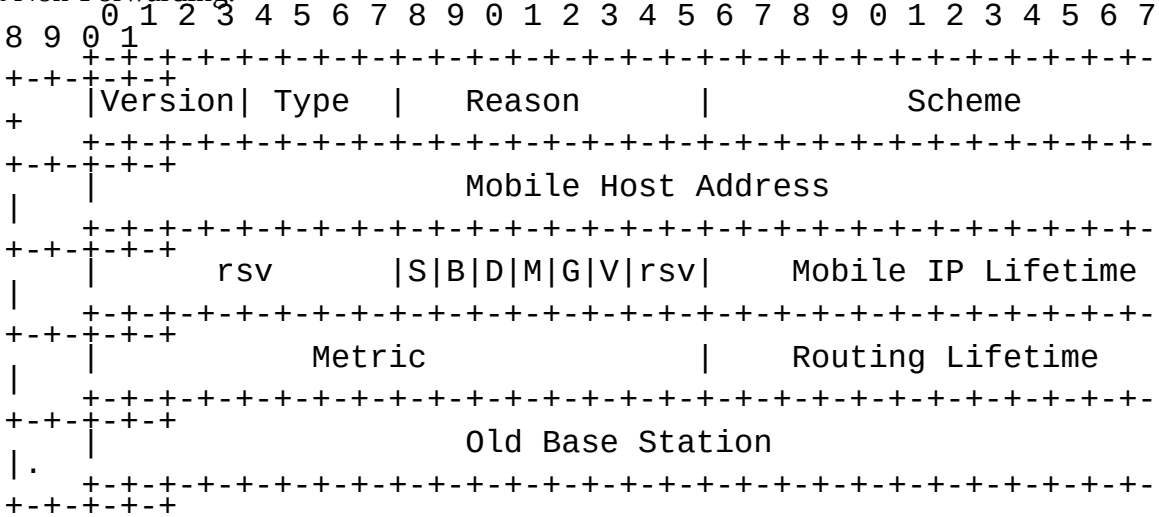
- **Structure du paging-teardown packet :** Le paquet paging-teardown est un paquet ICMP dans lequel : - L'adresse IP source est celle du hôte mobile émetteur,
  - L'adresse IP destination est celle de la passerelle,
  - Le Type indique 'un paquet Cellular IP de contrôle' et le Code indique 'paging-teardown'.

Le reste du paquet transporte des informations d'authentification et de contrôle sous le même format que le route-update packet. Le champ S flag est mis à 0 pour les paging-teardown packets.

2.3 Formats des messages Hawaii

Dans cette section, les formats des messages HAWAII envoyés entre les stations de base et les routeurs dans un domaine sont présentés. Le format des messages envoyés entre les hôtes mobile et les stations de base suivent le format des messages Mobiles IP avec des extensions.

Le format d'un message "path setup update" envoyé par des stations de base et des routeurs est montré ci-dessous. Le message est envoyé en utilisant le protocole UDP à un port réservé. Les messages "Power up updates" (type 1) sont envoyés par la station de base actuelle. Les messages de mise à jour du Handoff « Handoff updates » (type 2) sont envoyés à l'ancienne station de base dans le cas du schéma Forwarding, et à la nouvelle station de base dans le cas du schéma Non-Forwarding.



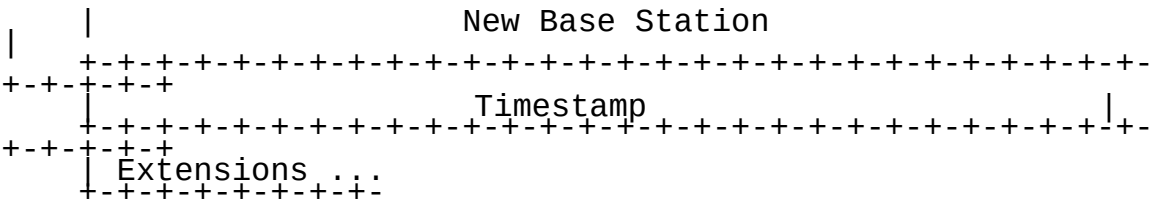


Figure 33 : Format de « path setup update message »

Version	1
Type	1 (Power up update), 2 (handoff update), 3 (acknowledgement)
Reason	utilisé seulement pour le Type 3 : 0 accepté, 1 message mal formaté, 2 authentification échouée, 3 schéma non supporté , 4 ressource non disponible
Scheme	1 (Forwarding), 2 (Non-Forwarding)
Mobile host Address	Home address dans le Home domain, COA dans le Foreign domain
rsv	réservé, envoyé en tant que 0
S,B,D,M,G,V	flags de l'enregistrement Mobile IP
Mobile IP lifetime	champ ‘Lifetime’ dans l’enregistrement Mobile IP
Metric	distance à l’hôte mobile en sauts
Routing Lifetime	valeur du temporisateur de l’état-soft
Old Base Station	adresse IP de l’ancienne BS pour le Type 2, 0.0.0.0 pour Type 1
New Base Station	adresse IP de la nouvelle BS pour le Type 2, BS courante pour Type 1
Timestamp	estampille comme dans le Network Time Protocol [MIL92]
Extensions	champ d’authentification

Le format du message de rafraîchissement est montré ci-dessous. Le message peut contenir de multiples entrées en tant qu'élément d'un agrégat de rafraîchissements. Il est envoyé par des stations de base et des routeurs à leur routeur ascendant. Cependant, la taille du message ne doit pas excéder 4kb.

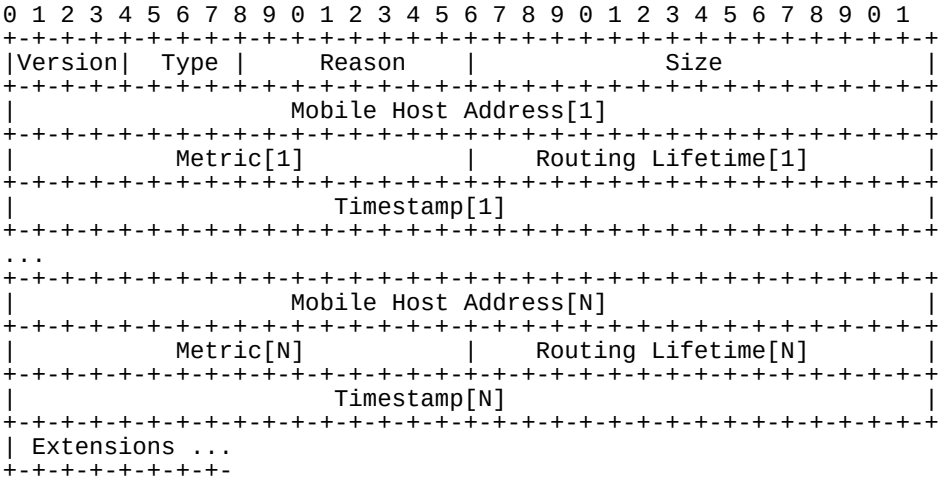


Figure 34 : Format du «path setup refresh message»

Version	1
Type	4 (rafraîchissement)
Reason	0 (normal), 1 (déclenché en raison de panne d’un lien/hôte)
Size	nombre d'entrées d’un hôte mobile
Mobile host Address	adresse de l’hôte mobile
Metric	distance à l’hôte mobile en sauts
Routing Lifetime	valeur du temporisateur de l’état-soft
Timestamp	estampille de l’entrée correspondante à l’hôte mobile
Extensions	champ d’authentification

Annexe 3 : MIPv4 et MIPv6

3.1 MIPv4

Mobile IPv4 [PER96a] définit trois entités fonctionnelles : le noeud mobile et deux agents de mobilité, l'agent mère et l'agent étranger. Le noeud mobile est configuré avec une adresse IP permanente, appelée adresse mère et appartenant à son réseau mère. L'agent mère est un routeur spécifique du réseau mère, il enregistre la localisation du mobile lorsque celuici est en visite dans un réseau externe. L'agent étranger est un routeur du réseau externe (réseau "visité") dans lequel le noeud mobile est localisé à un moment donné. Il est généralement utilisé par le noeud mobile pour obtenir l'adresse IP temporaire correspondant à sa nouvelle localisation, adresse qu'il enregistre auprès de son agent mère. La version de base de Mobile IPv4 impose que les paquets adressés aux nœuds mobiles soient routés à travers l'agent mère. Ils suivent donc des chemins qui peuvent être assez longs et non optimaux. L'extension nommée Optimisation de routage remédie à ce problème en permettant à un nœud correspondant de maintenir un cache d'associations qui pointe continuellement sur l'adresse temporaire courante du nœud mobile. Il peut alors router directement les paquets sans passer par l'agent mère. Ces caches d'associations sont créés et mis à jour par des nouveaux types de messages.

3.1.1 Découverte des agents de mobilité

Une caractéristique propre au mobile est de pouvoir se déplacer en cours d'une communication. Pour cela, un nœud mobile doit pouvoir détecter ses déplacements, c'est-à-dire détecter le changement de sous-réseau, ce qui nécessite l'obtention d'une nouvelle adresse temporaire. Le protocole de découverte des agents met en place un échange de messages permettant cette détection : les agents de mobilité envoient périodiquement des messages annonçant leur disponibilité sur le lien par l'émission de messages *Agent Advertisement* contenant l'information nécessaire pour l'identification du sous-réseau. Cette information peut être le préfixe réseau par exemple. Par ailleurs, un nœud mobile ne désirant pas attendre un tel message peut explicitement en demander un par l'émission d'un *Agent Solicitation* (cas où l'agent tombe en panne par exemple). Ces messages sont authentifiés et sont envoyés en broadcast ou multicast.

3.1.2 Enregistrement auprès de l'agent mère

Lorsque le nœud mobile détecte qu'il a changé de sous-réseau (à travers les messages explicités ci-dessus), il doit acquérir une nouvelle adresse temporaire et s'enregistrer auprès de son agent mère et de l'agent visité du réseau visité. L'acquisition de cette nouvelle adresse se fait grâce au protocole DHCP (Dynamic Host Configuration Protocol) [DRO97] (Figure 35).

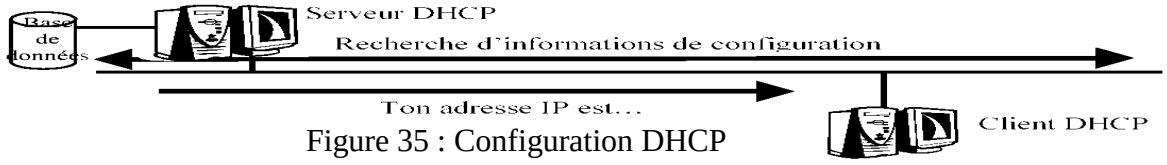


Figure 35 : Configuration DHCP

Une fois que le nœud mobile a une adresse temporaire valide, il émet un message *Registration Request* (étape 1 dans la Figure 36) en indiquant la correspondance entre son adresse principale et son adresse temporaire et éventuellement d'autres options. Ce message passe par l'agent visité qui le transmet à l'agent mère du mobile s'il accepte les requêtes du nœud mobile.

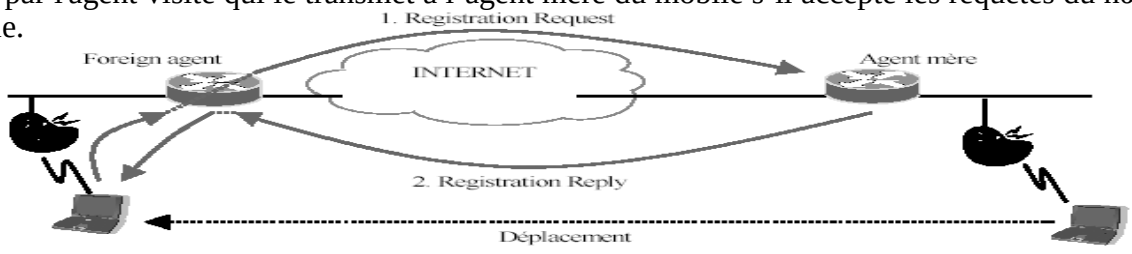


Figure 36 : Enregistrement v4

L'agent mère doit acquitter le *Registration Request* pour bien confirmer la réception (message UDP) et pour informer le nœud mobile de l'acceptation ou du refus de la requête par un *Registration Reply* (étape 2). A la réception du *Registration Request*, aussi bien l'agent mère que l'agent visité mettent à jour leur cache d'association pour ce nœud mobile.

Ensuite, tant que le nœud mobile reste dans le même sous-réseau étranger, il doit uniquement envoyer un *Registration Request* à intervalle régulier pour éviter que son entrée dans le cache d'association des agents de mobilité n'expire. Par contre, à chaque nouveau déplacement dans un autre sous-réseau étranger, il devra reprendre les mêmes opérations que celles décrites ci-dessus.

Si le nœud mobile retourne dans son sous-réseau mère, il doit se dés-enregistrer auprès de son agent mère. Il envoie alors un *De-Registration Request* (étape 1 dans la Figure 37) jusqu'à ce qu'il reçoive un *De-Regitration Reply* (étape 2) qui spécifie que l'agent mère a bien reçu le message et qu'il a supprimé l'entrée pour ce nœud mobile.

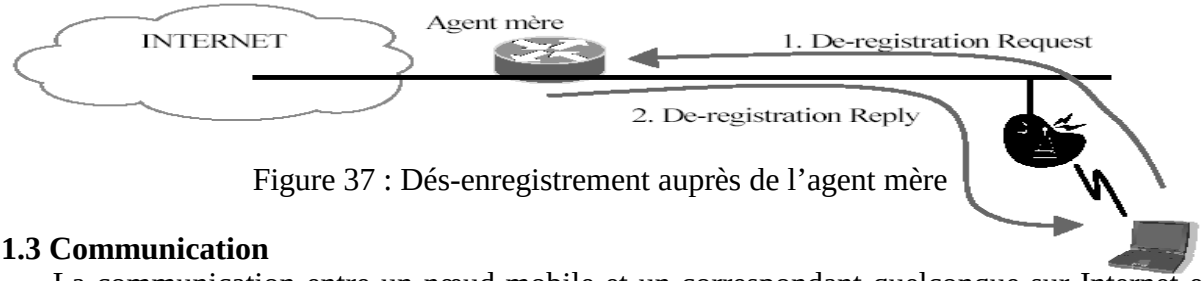


Figure 37 : Dés-enregistrement auprès de l'agent mère

3.1.3 Communication

La communication entre un nœud mobile et un correspondant quelconque sur Internet est très spécifique et requiert plusieurs mécanismes des agents de mobilité. Comme un nœud correspondant d'un nœud mobile ne connaît que l'adresse principale du nœud mobile, les paquets à destination du nœud mobile sont toujours envoyés dans le sous réseau mère du nœud mobile. Si le nœud mobile ne s'est pas déplacé, les paquets lui seront « livrés » de la même manière qu'un nœud fixe, c'est-à-dire sans opérations supplémentaires. Par contre, si le nœud mobile est dans un sous-réseau visité, son agent mère devra capturer tous les paquets destinés au nœud mobile et les lui transmettre à son adresse temporaire, grâce à son cache d'association (comme illustre la Figure 38).

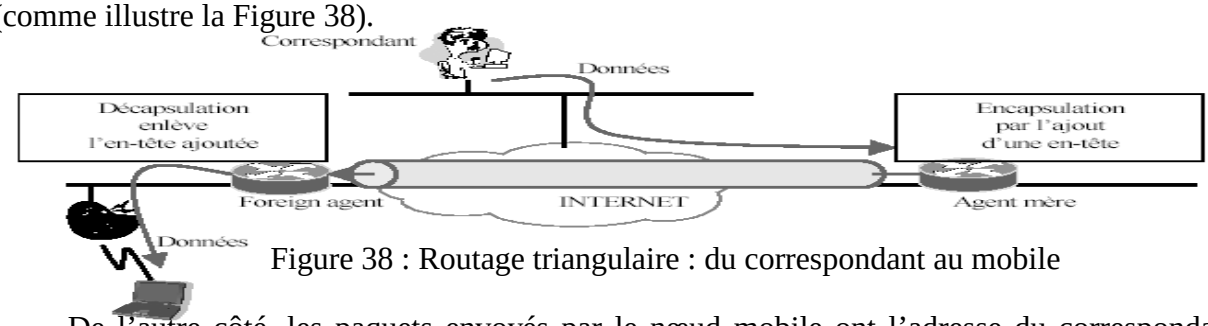


Figure 38 : Routage triangulaire : du correspondant au mobile

De l'autre côté, les paquets envoyés par le nœud mobile ont l'adresse du correspondant comme adresse destination et l'adresse principale du mobile comme adresse source. Ceci présente une entorse au modèle de l'Internet puisque l'adresse source des paquets envoyés par le nœud mobile ne correspond pas au préfixe du sous réseau visité. Les paquets devront alors obligatoirement passer par l'agent visité pour éviter qu'ils ne soient détruits. Par contre, une fois que les paquets ont été routés hors du sous-réseau visité, ils vont directement du nœud mobile au correspondant sans passer par le réseau mère. (Figure 39).

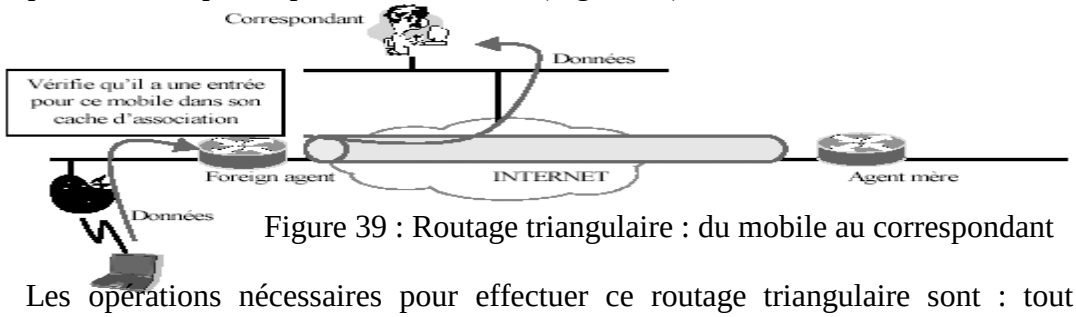


Figure 39 : Routage triangulaire : du mobile au correspondant

Les opérations nécessaires pour effectuer ce routage triangulaire sont : tout d'abord, lorsque le nœud mobile se déplace dans un sous-réseau visité, il doit en informer son agent mère à travers un *Registration Request*. A la réception de ce message, si l'agent mère accepte la

requête, en plus de créer ou de mettre à jour l'entrée pour ce nœud mobile, il envoie une requête ARP sur le réseau principal afin de faire correspondre l'adresse IP du mobile avec son adresse MAC. Ainsi il peut intercepter les paquets à destination du mobile. Ensuite, l'agent mère doit faire suivre ces paquets à la position courante du mobile. Pour cela, il encapsule chaque paquet en ajoutant un en-tête de destination rempli avec l'adresse temporaire courante du mobile comme adresse destination et avec son adresse comme adresse source avant de les tunnellés à l'agent visité. Enfin, chaque paquet est décapsulé par l'agent visité (suppression de l'en-tête) et délivré au nœud mobile.

3.2 MIPv6

Une nouvelle version du protocole IP est en train d'émerger depuis quelques années : il s'agit de la version 6 du protocole IP. Ce protocole inclut entre autres la mobilité en standard. L'objectif de MIPv6 est d'offrir une communication directe (routage optimisé) entre un nœud mobile et ses correspondants (élimination du routage triangulaire) et d'éviter les ruptures des communications pendant les déplacements. Bien que MIPv6 reprenne des mécanismes de MIPv4, de nombreuses fonctionnalités supplémentaires ont été mises en place.

3.2.1 Fonctionnalités requises

Dans MIPv6, l'agent visité décrit dans MIPv4 n'existe plus. Par contre, l'agent mère est encore un routeur d'accès du sous-réseau principal du nœud mobile. Son rôle est le même que dans le cas de MIPv4, à savoir capturer les paquets à destination du mobile et les lui tunneller à sa localisation courante.

Par contre, les correspondants doivent mettre en œuvre certains mécanismes supplémentaires : tout d'abord, ils doivent disposer d'un cache d'association tout comme l'agent mère; dans ce cache sera stockée la correspondance entre l'adresse principale d'un nœud mobile avec lequel il a une communication et son adresse temporaire courante. Il devra donc être capable de traiter des messages de registration envoyés par un nœud mobile. De plus, il devra être capable d'effectuer le routage directement vers le mobile. Ceci constitue un apport important dans le fonctionnement de la mobilité puisque les paquets des correspondants n'auront pas à passer par le réseau mère systématiquement. Mais toutes ces fonctionnalités supplémentaires ne sont faites qu'au niveau de la couche IP; l'adresse identifiant la communication au niveau applicatif sera toujours l'adresse principale du nœud mobile, la couche IP sachant l'adresse temporaire source (ou destination selon qu'on se situe sur le nœud mobile ou le correspondant).

D'un autre côté, un nœud mobile doit toujours conserver la liste des correspondants auxquels il envoie un message de registration (pour les mises à jour éventuelles) et doit être capable de décapsuler lui-même les paquets qui lui sont transmis ; au niveau application, un nœud mobile utilise toujours son adresse principale, c'est pourquoi la couche IP doit pouvoir décapsuler l'en-tête indiquant l'adresse temporaire. Cette opération était exécutée par le agent visité dans MIPv4.

3.2.2 Découverte des routeurs d'accès

Le protocole de découverte des voisins [NAR98] offert par IPv6 joue un rôle important dans MIPv6. 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 *Router Advertisment* contenant la liste des préfixes sur le lien. Un nœud mobile peut éventuellement en demander un explicitement, à travers un *RouterSolicitation*.

Les routeurs d'accès offrant des fonctionnalités pour la mobilité émettent des *Router Advertisement* quelque peu modifié (pour avertir les mobiles de leur capacité). En outre, l'information contenue dans ces *Router Advertisement* permet aux noeuds mobiles de créer une adresse temporaire (auto configuration offerte par IPv6). Ensuite il leur faudra vérifier l'unicité de celle-ci grâce au protocole de détection de duplication d'adresse [THO98]. La découverte des voisins ainsi que la découverte de l'adresse de niveau 2 d'un équipement voisin s'avère aussi très utile dans la mobilité, notamment pour effectuer des registrations plus rapides. L'utilisation des

ces données sera détaillée plus tard car le protocole MIPv6 ne prend pas encore en compte ces données.

3.2.3 Enregistrement

De la même manière que dans MIPv4, lorsqu'un nœud mobile se déplace hors de son sous-réseau mère, il doit en informer son agent mère. Le nœud mobile signale la correspondance entre son adresse principale et son adresse temporaire courante dans un message *Binding Update*. Ce message peut éventuellement être envoyé en «*piggy-backing*»(le fait de mettre des informations de contrôle dans des paquets de données). En réponse à une telle requête, l'agent mère envoie un *Binding Ackowledgement* pour indiquer s'il peut répondre à la requête du mobile. Pour le moment, tout se passe comme dans MIPv4. Cependant, le mobile a par la suite la possibilité d'informer ses correspondants de sa position courante; Lorsqu'il reçoit un paquet d'un correspondant, il détermine si le paquet a transigé par le réseau mère en regardant si le paquet contient un *routing header* ou s'il a été tunnelé par l'agent mère (encapsulation). S'il ne contient pas de *routing header*, le nœud mobile en déduit que le correspondant émetteur n'a pas d'entrée dans son cache d'association pour lui. Il peut alors lui envoyer un *Binding Update* pour qu'il lui envoie les paquets directement, sans plus passer par son sous-réseau mère (Figure 40).

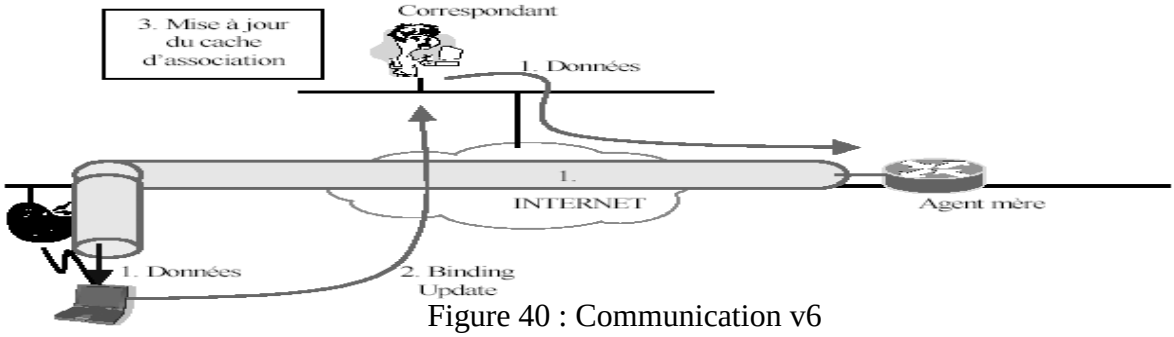


Figure 40 : Communication v6

Le fait que les correspondants aient la possibilité d'envoyer les paquets directement au mobile offre une meilleure résistance au facteur d'échelle et fiabilité. La communication entre nœuds mobiles et correspondants engendre moins de charge sur le réseau et est plus rapide. Comme l'agent mère est peu sollicité pour la retransmission des paquets, il y a beaucoup moins de risque de congestion au niveau de l'agent mère et une panne de l'agent mère aura un effet moindre. Un nœud mobile peut détenir plus d'une adresse temporaire à un instant donné. Celle enregistrée auprès de l'agent mère (une seule) est dite principale. L'utilisation de plusieurs adresses temporaires peut être utile pour améliorer les performances lors d'un déplacement lorsque par exemple deux cellules de points d'accès se recouvrent fortement ; le nœud mobile peut alors acquérir une nouvelle adresse temporaire tout en utilisant son ancienne le temps de l'opération.

3.3 Comparaison de Mobile IPv4 avec Mobile IPv6

La conception de Mobile IPv6 s'est basée sur les expériences acquises du développement de Mobile IPv4 et sur les nouvelles opportunités offertes par le protocole IPv6 , telles que le nombre plus important d'adresses et les mécanismes d'auto configuration.

L'utilisation des options destination d'IPv6, qui fournissent des informations au noeud destinataire final, permet aux informations de contrôle de Mobile IPv6 d'être transportées dans l'entête des paquets IP contrairement à Mobile IPv4 où un paquet UDP spécifique doit être utilisé pour chaque type de message de contrôle. L'optimisation de routage est intégrée dans le protocole Mobile IPv6 puisqu'elle est assurée, comme l'enregistrement avec l'agent mère, par des messages de mise à jour d'associations. Le protocole IPv6 permet aux noeuds mobiles exécutant Mobile IP de communiquer à travers des routeurs filtrants en utilisant l'adresse temporaire comme adresse source. L'adresse mère est indiquée dans une option de destination, appelée option adresse mère, du paquet IPv6. Enfin, Mobile IPv6 ne requiert pas le déploiement d'agents étrangers. Les noeuds mobiles utilisent les mécanismes d'auto configuration IPv6 fonctionnant dans tous réseau IPv6 visité.

Annexe 4 : Présentation du simulateur réseau NS-2

4.1 Introduction

NS-2, Network Simulator, est aujourd'hui le simulateur de réseau probablement le plus utilisé par la communauté scientifique des réseaux. Il s'agit d'un simulateur à événements discrets, fruit de la collaboration entre l'université de Berkeley, USC(*University of Southern California*) et Xerox PARC dans le cadre du projet VINT (*Virtual Inter Network Testbed*). Ce projet est soutenu par le DARPA (*Defense Advanced Projects Agency*). NS est un outil de recherche très utile pour le design et la compréhension des protocoles. Il sert aussi bien dans l'étude des protocoles de routage qu'à l'étude des réseaux mobiles ou les communications par satellites.

NS-2 permet à l'utilisateur de définir un réseau et de simuler les communications entre les nœuds. Le simulateur utilise le langage orienté objet OTCL(*Object Tools Command Language*) dérivé de TCL pour la description des conditions de simulation sous forme de script.

Dans le script l'utilisateur fournit la topologie du réseau, les caractéristiques des liens physiques, les protocoles utilisés, le type de trafic généré par les sources, les événements, etc. Si le script écrit en OTCL permet une utilisation (édition, modification des simulations) facilitée du simulateur, les routines sont elles écrites en C++ pour avoir une plus grande puissance de calculs. Un grand nombre de classes sont prédéfinies et mettent en oeuvre plusieurs types de protocoles, des files d'attentes, de sources et algorithmes de routage. Le résultat d'une simulation est un fichier texte contenant tous les événements de la simulation. Un traitement ultérieur (A l'aide de l'utilitaire AWK par exemple, très pratique pour le traitement de fichiers textes de données structurées) de ce fichier permet d'en soustraire l'information souhaitée.

Par ailleurs, le simulateur permet la création d'un fichier d'animation (d'extension .tr), permettant de visualiser la simulation sur l'interface graphique NAM(*Network AniMator*). Ce visualisateur fournit une représentation du graphe du réseau sur laquelle on peut voir les paquets circuler, suivre le niveau des files d'attente et observer le débit courant des liaisons.

L'utilisation de ces outils et l'exploitation des résultats requiert certaines compétences:

- Connaissance du C++ et maîtrise des concepts de la programmation orientée objet pour l'édition des routines qui composent le simulateur.
- Connaissance du langage script OTCL pour la description des simulations et le pré traitement des données.
- Maîtrise de l'outil indispensable à l'extraction des données des fichiers,AWK.
- Capacité à faire des calculs et à produire des graphes à partir des fichiers produits.

Pour prendre contact avec NS, le tutorial de Marc Greis (http://www.isi.edu/nsnam/ns/tutorial/index.html) est idéal. Cependant il n'entre pas du tout dans les détails, l'utilisateur est donc rapidement amené à consulter le "NS Manual" (http://www.isi.edu/nsnam/ns/.) . Ce volumineux manuel n'est rien d'autre qu'un parcours rapide du code de NS. L'utilisateur est donc quasi obligé de se plonger dans le code s'il désire comprendre le fonctionnement du simulateur. Il en ressort que cette grosse lacune au niveau de la documentation rend l'apprentissage de ce simulateur pénible et fastidieux. Un effort important devrait être fait dans cette direction pour donner au simulateur une meilleure accessibilité et ainsi encourager son utilisation, faisant de lui un outil puissant et répandu de simulation.

Les concepteurs de NS ne manquent pas de mettre en garde les utilisateurs sur l'aspect non achevé de NS. Il ne s'agit pas d'un produit fini, de nouveaux bogues sont souvent trouvés. Le logiciel est par essence en constante évolution de par la nature des réseaux et protocoles qui sont en perpétuelle évolution.

4.2 Architecture et implémentation

L'architecture réseau de NS-2 est fortement basée sur le modèle des couches OSI. Il s'agit de la décomposition de la pile réseau en couches. La dernière version de NS-2 est sortie le 18 janvier 2004 (ns-2.27). Les sources sont disponibles sur le site ISI (*Information Sciences Institute*) dans la section *nsnam* [http://www.isi.edu/nsnam/ns]. Les sources se présentent sous deux formes : l'une dite « tout en un » qui contient le code NS-2 et d'autres composants utilisés (comme OTcl, NAM…), soit par morceaux, c'est-à-dire qu'on peut choisir uniquement les composants dont on a besoin. Le package comprend aussi des exemples de script ainsi que des modèles de mouvement pour les nœuds mobiles ou de génération de

trafic. NS-2 est un simulateur à évènements à temps discrets orienté objet. Il est développé en C++ et en OTcl (extension objet du langage interprété Tcl). Le paquage inclus une hiérarchie de classe compilée d'objets écrits en C++ et une hiérarchie de classe interprétée d'objets écrits en OTcl. Ces deux hiérarchies sont étroitement liées; quand l'utilisateur crée un nouvel objet par l'interpréteur OTcl, un objet correspondant appelé objet reflet est aussi crée dans la hiérarchie compilée. On dit que ces objets sont des «objets fendus». Bien entendu, les objets peuvent être accédés aussi bien en OTcl qu'en C++ grâce à la mise en place de procédures d'appel entre OTcl et C++. 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 évènements, perte réseau, rassemblement de statistiques). 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…).

➢ **Moteur du simulateur :** Le moteur de simulation est extensible, configurable et programmable. La mise en oeuvre actuelle est un chaînage d'évènements (un seul événement est exécuté à un instant donné). Il ne supporte pas l'exécution partielle d'événements, ni la préemption. Les événements sont décrits par des renvois et une fonction manipulatrice. Le type du planificateur d'événements utilisé pour conduire la simulation peut être choisie parmi quatre actuellement disponibles : une liste liée simple (par défaut), le tas, la file d'attente de calendrier et un type spécial appelé temps réel. Chaque méthode est implémentée avec une structure de données différente :

- Le planificateur « liste liée simple » fournit une liste d'événements tenue dans l'ordre chronologique, du plus premier au dernier. Pour insérer ou supprimer une entrée, il faut alors balayer toute la liste pour trouver l'entrée appropriée. L'entrée en tête est toujours exécutée en première. Les entrées avec le même temps simulé sont extraites selon leur ordre dans la liste.
- Le code du planificateur de tas est emprunté au simulateur MARS-2.0. Cette implémentation est plus efficace que le planificateur de liste liée quand le nombre d'événements est grand.
- Dans la mise en oeuvre du planificateur de file d'attente de calendrier, les événements avec le même "mois/jour" de différentes "années" sont enregistrés dans un même "jour".
- Le planificateur en temps réel essaie de synchroniser l'exécution d'événements en temps réel. Il est actuellement mis en oeuvre comme une sous-classe du planificateur de liste. La capacité temps réel dans ns est toujours en développement, mais elle est employée pour introduire un réseau simulé dans une topologie réelle pour expérimenter de manière simple des topologies réseau, la traversée de trafic… Les travaux réalisés jusqu'à présent fonctionnent pour des taux de trafic de données relativement lents. Or le simulateur doit être capable de suivre le taux d'arrivée des paquets du réseau réel. Cette synchronisation entre le réseau simulé et le réseau réel n'est actuellement pas mise en œuvre.

➢ **Implémentation du simulateur :** Au plus bas niveau, il y a six classes qui définissent l'ensemble de la structure du programme et fournissent les méthodes élémentaires. Il s'agit des classes *Tcl*, *TclObject*, *TclClass*, *TclCommand*, *EmbeddedTcl*, *InstVar*. Elles définissent entre autres les méthodes utilisées par C++ pour accéder à l'interpréteur, la hiérarchie, les principales commandes de haut niveau et les méthodes pour accéder aux variables C++ et OTcl.

La simulation est configurée, contrôlée et gérée à l'aide des interfaces fournies par la classe OTcl *Simulator*. Cette classe fournit des procédures pour créer et gérer la topologie, initialiser le format des paquets et choisir le planificateur d'évènements. Elle stocke intérieurement des références à chaque élément de la topologie. Un script devra donc toujours commencer par l'instanciation d'une variable de cette classe. L'utilisateur crée ensuite la topologie à travers OTcl en utilisant les classes *node* et *link*, composants essentiels de la topologie. Ces éléments sont décrits dans la sous section suivante.

➢ **Composants de la topologie :** La topologie NS-2 est essentiellement composée de nœuds et de liens. La définition des nœuds se fait dans un premier temps à travers l'instance de *Simulator* puis à travers l'instance de la classe *Node*. La fonction d'un nœud est de recevoir des paquets, les examiner et les mapper à ses interfaces sortantes appropriées. Cette classe est composée d'un classificateur et de méthodes pour configurer un nœud. Les méthodes proposées sont des fonctions de contrôle, de gestion d'adresse et de port, de gestion d'agents et de repérage des voisins. Le classificateur est la partie du nœud qui traite chaque segment des paquets reçus. Il en existe donc plusieurs, chacun étant spécifique au champ examiné (ex : le classificateur d'adresse est utilisé pour supporter la propagation des paquets). Les liens constituent la deuxième partie de la topologie. Les liens entre les nœuds sont définis dans la classe *Link* et *SimpleLink* plus précisément lorsqu'il s'agit de relier deux nœuds. Plusieurs types de liaisons sont

supportés, comme le point à point, le broadcast ou les liaisons sans fil pour la mobilité. Une liaison est définie par une séquence de connecteurs (classe *Connector*) qui sont rangés dans une file d'attente. Ces connecteurs font suivre les paquets qui leur sont envoyés dans une seule direction ; le paquet est alors délivré au voisin cible ou il est détruit. A présent, on décrit les structures mises en place autour de la topologie pour faire interagir les nœuds entre eux.

➢ **La gestion des files d'attente :** La gestion des files d'attente et la simulation des délais sur les liens sont implémentés dans les classes *Queue* et *LinkDelay* respectivement. Les files d'attente actuellement disponible dans NS sont : FIFO, RED buffer management, CBQ (priorité et circulaire), plusieurs variantes de file d'attente juste (Fair Queue). Pour simuler un quelconque délai dans la réception ou l'émission d'un paquet, la file d'attente correspondante est simplement bloquée.

➢ **Les agents :** A un niveau supérieur, on retrouve les agents (*classe Agent*) qui jouent un rôle important dans les simulations. Les utilisateurs créent de nouvelles sources ou récepteurs à partir de la classe *Agent*. Ils font partie intégrante d'un nœud et sont les points terminaux vis à vis des paquets couche réseau ; leur rôle est de générer et réceptionner des paquets suivant les protocoles de transport (TCP,UDP..). La génération de trafic dans NS peut se faire de deux manières différentes et est décrit dans la classe *Application*. Il est possible de générer des paquets par un générateur de trafic (classe *TrafficGenerator*) ou par la simulation d'applications existantes (classe prenant le nom de l'application), toutes ces classes étant dérivées dans la classe *Application*. Les générateurs de trafic peuvent être de quatre types :

- Classe *Exponential* : génère un trafic ON/OFF(un noeud qui génère du trafic ON/OFF est un nœud qui alternativement émet des paquets et stoppe son émission.) à intervalle de temps régulier.
- Classe *Pareto* : génère un trafic ON/OFF à intervalle de temps aléatoire.
- Classe *CBQ* : débit de bit constant.
- Classe *Trace* : permet de lire la génération de trafic dans un fichier.

➢ **Le routage :** Lors d'une simulation, l'utilisateur doit spécifier un protocole de routage, c'est-à-dire indiquer au simulateur comment construire les routes entre les nœuds. NS offre trois types de routage dans un réseau filaire : Routage statique (routage utilisé par défaut. Il est exécuté au début de la simulation une fois pour toutes), routage session (routage identique au routage statique mais ré-exécute l'algorithme à chaque changement de topologie), routage dynamique (une valeur est assignée à chaque route et un tableau stocke toutes les routes les plus courtes). Il est possible d'indiquer le protocole de routage uniquement à un sous-ensemble des nœuds constituant la topologie (par défaut il s'applique à la totalité des nœuds).

NS permet de provoquer des erreurs dans les simulations pour tester la robustesse des protocoles. Pour cela, il existe un modèle d'erreur implémenté dans la classe *ErrorModel* qui simule les erreurs au niveau liaison par l'envoie des paquets à des agents destructeurs. L'unité d'erreur peut être spécifiée en terme de paquets, bits ou temps. Maintenant que les modules de base ont été présentés, on peut voir comment spécifier des caractéristiques au niveau des nœuds pour étoffer les simulations (comment simuler les réseaux locaux et les nœuds mobiles).

➢ **Les réseaux locaux (LAN) :** Pour permettre des simulations à plus grande échelle, NS permet d'utiliser la notion de LAN. Cette nouvelle entité a été introduite en tant que nouveau type de nœud. Les composants d'un LAN sont la couche liaison, la couche MAC et la couche physique.

Au niveau de la couche physique, l'objet *Channel* simule le médium partagé et supporte des mécanismes d'accès au médium des objets *Mac* (phase d'émission). L'objet *Classifier/Mac* est responsable de la livraison et de la réplication des paquets pour des objets *Mac* (phase réception). Les détections de collisions se font au niveau de la couche MAC où sont implémentés les protocoles d'accès au médium (CSMA…). La couche liaison est composée de deux objets : *Queue* qui simule l'interface de file d'attente et *LinkLayer* qui implémente un protocole de couche de données.

➢ **La mobilité dans NS :** Dans un premier temps, la mobilité a été introduite dans NS-2 par les chercheurs de l'université Cartegie Mellon24 de Pittsburgh (CMU) dans la volonté de simuler des réseaux ad hoc. L'apport de la mobilité passe par l'ajout d'un nouveau type de nœuds définis dans la classe *MobileNode*, qui ne sont pas connectés entre eux. Les caractéristiques de la mobilité telles que le mouvement des nœuds, les mises à jour de localisation ou les limites de la topologie sont implémentées

en C++. Par contre, les composants réseaux comme le nœud mobile lui-même (classificateur, couche liaison…) sont implémentés en OTcl.

Comme l'objectif était de simuler des réseaux entièrement mobiles, il a fallu mettre en place des protocoles de routage. Actuellement, il y a quatre protocoles de routage mis en œuvre dans NS-2 :

- DSDV : (Séquence de destination à vecteur de distance,***Dynamic Destination-Sequenced Distance Vector Routing Protocol***)[PER94] : les tables de routage contiennent le nombre de sauts pour atteindre la destination, et un numéro de séquence associé au nœud destination, qui sert à détecter les routes qui ne sont plus à jour. Les tables de routage sont diffusées intégralement ou incrémentalement.
- DSR (Routage par source dynamique,Dynamic Source Routing Protocol) [JOH00], repose sur des mécanismes de découverte et maintien de route. Si un destinataire est dans le cache du nœud source, la route connue est utilisée, sinon on déclenche une découverte de route. Les paquets de découverte de route contiennent les adresses source et destination ainsi qu'un identifiant qui sert aux nœuds intermédiaires pour savoir s'ils ont déjà traité ce paquet. Il existe deux types de paquets de maintenance de route : Route Erronnée (pour effacer les routes qui ne sont plus valides) et Acquittement.
-TORA (Algorithme de routage ordonné temporaire, Temporally Ordered Routing Algorithm) [PAR97]. TORA cherche à limiter la quantité de signalisation dans le réseau. Par exemple, lorsqu'un changement se passe dans le réseau, les messages de signalisation qui avertissent de cet événement ne sont propagés que vers des nœuds proches. Une source trouve plusieurs chemins vers une destination. Pour cela, les nœuds maintiennent des informations de routage vers leurs voisins.
- AODV (Vecteur de distance sur demande,Ad-hoc On demand Distance Vector Routing) [PER00] améliore DSDV en limitant le nombre d'informations de routage sur le réseau. Lorsqu'un nœud cherche une route vers une destination, il diffuse une demande de route à travers le réseau. Les nœuds qui reçoivent ces paquets les diffusent à leur tour jusqu'à atteindre un nœud qui possède une information de routage récente vers la destination recherchée ou la destination elle-même. Les nœuds qui relaient des informations de routage mettent également à jour leurs informations de routage.

Lorsqu'un un nœud mobile est crée dans une simulation, le simulateur crée un objet *MobileNode*, un agent de routage et la pile réseau. Ensuite ces composants sont interconnectés et la pile est connectée au canal. Une caractéristique forte des nœuds mobiles est bien sûr de pouvoir se déplacer. NS-2 a été conçu pour exécuter des déplacements en 3D, mais actuellement la troisième dimension n'est pas utilisée (Z=0). Il existe deux mécanismes pour l'utilisateur pour donner du mouvement aux nœuds mobiles :

- Indiquer le point d'origine, la destination et la vitesse explicitement pour chaque nœud mobile. Les mises à jour sont déclenchées chaque fois que l'on exige la position du nœud mobile à un moment donné. Cette solution est plutôt faite pour des petites simulations.
- Générer des mouvements aléatoires : à l'appel d'une procédure, le nœud mobile démarre à partir d'une position aléatoire et exécute des déplacements. Le nœud mobile exécute des mises à jour de routage pour changer de destination et de vitesse.

Indépendamment des méthodes utilisées pour générer les mouvements des nœuds mobiles, il faut définir une topographie. L'espace est considéré comme étant une grille dont il faut donner les frontières (valeurs de x abscisse et y ordonnée).

Les limitations de ce modèle se sont vite fait ressentir. Effectivement, le modèle original de la mobilité permet des simulations de réseaux locaux sans fil et de réseaux ad hoc uniquement. C'est pourquoi une première extension a été ajoutée au modèle, toujours fondée sur le travail des chercheurs de CMU, qui permet de faire des simulations impliquant des nœuds filaires et des nœuds sans fil à la fois. Cette extension, appelée « wired-cum-wireless », utilise le modèle de base de la mobilité décrit ci-dessus.

L'objectif est donc de relier des réseaux locaux sans fil par un réseau filaire. Il se pose immédiatement un problème pour le routage. Le routage des nœuds filaires se fait d'après la topologie grâce au concept de liaison, alors que dans la mobilité le concept de liaison n'existe pas. Un nouveau type de nœud est alors crée pour assurer la liaison entre le réseau filaire et le réseau sans fil. Ce nœud est appelé *BaseStationNode.* Ce nœud est un hybride entre un nœud hiérarchique et un nœud mobile. L'introduction de tel point d'accès a un impact sur l'adressage. Chaque domaine de nœuds mobiles a une adresse unique de domaine et un domaine est défini comme l'ensemble des nœuds mobiles qui sont attachés au point d'accès du domaine. Les nœuds mobiles doivent donc supporter le routage hiérarchique. Un paquet destiné à un correspondant situé en dehors du domaine sans fil sera transmis au point d'accès, si toute fois il existe un chemin jusqu'à celui-ci.

L'adressage hiérarchique fonctionne donc de la manière suivante : il est composé de trois niveaux et noté ainsi : 1.0.1. le premier chiffre indique le domaine, le deuxième indique le cluster et le dernier est l'identifiant du nœud. Une autre extension a été ajoutée dans NS-2 pour supporter l'implémentation de

Sun Microsystem28 de Mobile IP. Cette extension est uniquement basée sur le modèle des nœuds filaires et non sur le modèle de la mobilité fait par CMU. Le scénario de Mobile IP consiste en des agents mère, des agents visités et des hôtes mobiles qui se déplacent de l'un à l'autre. Les agents mères et les agents visités sont grossièrement des points d'accès comme ceux décrits plus haut. Ils sont définis dans *MobileNode/MIPBS*. Ils contiennent un agent d'enregistrement qui envoie les *beacons* et effectue l'encapsulation et la décapsulation des paquets. L'hôte mobile est défini dans *MobileNode/MIPMH* qui a aussi un agent d'enregistrement qui réceptionne et émet des *beacons*. Pour s'apercevoir de la portée et du niveau de simulation qu'il est possible de faire, voici la liste des configurations possibles pour un nœud mobile :

- Type d'adressage : plat ou hiérarchique
- Type du routage ad hoc
- Type de l'objet *LinkLayer*
- Type du protocole utilisé par la couche MAC
- Type de propagation radio
- Interface de la file d'attente
- Type de support physique
- Type de l'antenne
- Type du canal
- Activer ou non les fonctionnalités de routage filaire
- Activer ou non le protocole MIP sur le nœud
- Type de modèle d'énergie / énergie initiale
- Activer ou non un agent de trace sur le nœud
- Activer ou non un agent de trace de routage sur le nœud
- Activer ou non les traces des mouvements des mobiles

Pour le moment, certains de ces paramètres ne peuvent prendre qu'une seule valeur. Par exemple, la couche MAC n'est implémentée que par le protocole 802.11 dans NS-2 de base.

4.3 Point de vue de l'utilisateur

Afin de mettre en application toute la structure présentée ci-dessus, voici quelques exemples de scripts Tcl. Dans cette partie ne sera exposée que les commandes principales. Pour une plus ample prise en main de NS-2 se référer au tutorial de Marc Greis (http://www.isi.edu/nsnam/ns/tutorial/). Comme le simulateur est un interpréteur interactif de scripts, deux possibilités s'offrent à l'utilisateur : soit lancer le simulateur et taper les scripts dans l'interface du simulateur, soit écrire le script dans un fichier « .tcl ». La première solution peut être appréciée au départ pour se familiariser avec les commandes. La seconde vaut pour des simulations de plus grande ampleur et doivent ensuite être exécutées par la commande :

```
ns <fichier>
```

Comme il a été dit plus haut, avant toute action, il faut instancier un objet *Simulator* qui permettra de créer et gérer tout autre objet. Sa création est assurée par :

```
set ns_ [new Simulator]
```

Ensuite, il faut définir la topologie, c'est-à-dire les nœuds et les liens. Voici la description de deux nœuds et d'un lien les reliant :

```
set n0 [$ns node]
set n1 [$ns node]
$ns duplex-link $n0 $n1 1Mb 10ms DropTail
```

Le lien ainsi crée est un lien « duplex » (communication en double sens), avec une bande passante de 10Mb, un délai de 10ms et une file d'attente « DropTail » (élimination de la queue). Pour créer une plus grande topologie, on peut créer les nœuds avec une boucle itérative. Chaque nœud sera alors stocké dans un tableau accessible par un indice. Le code correspondant à la création d'un tableau « n » de sept nœuds est le suivant :

```
for {set i 0} {$i < 7} {incr i} {
set n($i) [$ns node] }
```

Ensuite il faut définir les transferts possibles entre chaque nœud. Pour cela, il faut attacher des agents aux différents nœuds et les connecter ensemble. Voici la création d'un agent CBR :

```
set cbr0 [new Agent/CBR]
$ns_ attach-agent $n0 $cbr0
```

On peut également régler les paramètres des paquets envoyés (taille et période) :

```
$cbr0 set packetSize_ 500
$cbr0 set interval_ 0.005
```

De l'autre côté, il faut créer un agent récepteur sur l'autre nœud et enfin les connecter :

```
set null0 [new Agent/Null]
$ns_ attach-agent $n1 $null0
$ns_ connect $cbr0 $null0
```

Pour voir les effets des tests sur les protocoles de transport, on peut attacher un agent UDP ou TCP à un nœud générateur de trafic. Voici la mise en place d'un trafic CBR par un agent UDP:

```
set udp0 [new Agent/UDP]
$ns_ attach-agent $n0 $udp0
set cbr0 [new Agent/Traffic/CBR
$cbr0 attach-agent $udp0
$udp0 set packetSize_ 536
```

Puis en admettant qu'un agent « Null » null0 a déjà été attaché au nœud 1, on peut connecter les agents :

```
set null0 [new Agent/Null]
$ns_ attach-agent $n1 $null0
$ns_ connect $udp0 $null0
```

Ensuite on définit les évènements du scénario, c'est-à-dire quand le trafic commence, se termine:

```
$ns_ at 0.5 "$cbr0 start"
$ns_ at 4.5 "$cbr0 stop"
```

Enfin il faut lancer la simulation par la commande :

```
$ns_ run
```

La simulation de la mobilité dans NS-2 passe par la déclaration de nœuds mobiles. Celle-ci se passe comme pour les autres nœuds en précisant toutefois la configuration des paramètres. L'attache des agents pour simuler un protocole de transport ou une application est identique. Cependant, on peut définir la position initiale des nœuds mobiles ainsi que leur mouvement de manière très précise :

```
set topo [new Topography]
$topo load_flatgrid 500 500
```

Pour définir la grille, frontière de la simulation, puis :

```
$node_(0) set X_ 5.0
$node_(0) set Y_ 2.0
$ns_ at 10.0 "$node_(0) setdest 20.0 18.0 13.0"
```

Les deux premières lignes indiquent la position initiale du mobile dans la grille. La dernière ligne signifie que le nœud 0 au temps 10.0 va se déplacer en direction de la position (x=20, y=18) sur la grille à une vitesse de 13 mètres par seconde. Bien entendu, étant fait pour évaluer des protocoles, NS-2 permet de conserver une trace de l'échange de paquets. Les possibilités offertes sont décrites dans la section suivante. La génération de fichier de sortie fait partie du code NS-2 par des appels de procédures durant différentes actions dans les simulations. C'est pourquoi, pour visualiser une animation ou simplement enregistrer tous les évènements dans un fichier il faut le spécifier au début du script. Les commandes utilisées pour créer un fichier de sortie sont :

```
set tracefd [open sortie.tr w]
$ns_ trace-all $tracefd
```

et pour envoyer les évènements à NAM :

```
set tracenam [open out.nam w]
$ns_ namtrace-all $tracenam
```

4.4 Statistiques et visualisation

NS-2 fournit plusieurs types de support pour analyser les résultats d'une simulation. D'une part, NS-2 inclus des classes pour suivre à la trace les fluctuations des paquets, pour calculer et enregistrer diverses statistiques sur l'ensemble des paquets ou uniquement pour un certain flux. Ce système de suivi est présenté dans la sous-section suivante. D'autre part, NS-2 travaille de paire avec l'outil de visualisation NAM qui permet de visualiser l'ensemble de la topologie dans une fenêtre graphique.

Système de suivi : NS-2 propose deux types de monitoring :

- Traceur (*Trace*) : enregistre chaque paquet (arrivée, départ ou suppression) sur un lien ou dans une file d'attente. Ces objets sont configurés dans la simulation comme des nœuds dans la topologie de réseau. Ils sont définis dans plusieurs sous-classes pour des évènements précis.

▪ Moniteur (*Monitor*) : enregistre le décompte de différentes quantités comme le nombre d'arrivée de paquets , nombre de bit… Il traque ainsi la dynamique des paquets dans une file d'attente en faisant des moyennes.

L'utilisateur peut demander au simulateur d'enregistrer chaque déplacement des paquets dans un fichier de sortie. Le tableau ci-dessus illustre un fichier de sortie d'une simulation. Il présente 14 entrées de trace de paquets, dont cinq opérations de mise en file (indiqué par "+" dans la première colonne), quatre opérations de défilement (indiqué par "-"), quatre événements de réception (indiqué par "r") et un événement de suppression ("d"). Le temps simulé (en secondes) auquel chaque événement est arrivé est inscrit dans la deuxième colonne. Les deux champs suivants indiquent les deux nœuds entre lesquels le paquet circule. Vient ensuite un nom descriptif pour le type de paquet, suivi de sa taille, codée dans son en-tête IP. Le champ « flag » contient des flags qui ne sont pas utilisés ici. Le champ d'après donne l'identificateur de flux IP. Les deux champs suivants indiquent les adresses source et destination du paquet. Puis il y a le numéro de séquence et un identificateur de paquet unique. Chaque nouveau paquet créé dans la simulation est assigné un nouvel identificateur unique. Ce type de fichier de sortie peut bien entendu être utilisé pour tracer des courbes. Il est possible de demander au simulateur de ne répertorier que les paquets d'un certain type, par exemple que les paquets de contrôle appartenant au protocole pour une meilleure lisibilité.

Action	Temps	Nœuds		Paquet	Taille	Flag	ID flux	Adresses		N°= de séq.	uid
		Source	Dest.					Source	Dest.		
+	1.84375	0	2	cbr	210	---	0	0.0	3.1	225	610
-	1.84375	0	2	cbr	210	---	0	0.0	3.1	225	610
R	1.84471	2	1	cbr	210	---	1	3.0	1.0	195	600
R	1.84566	2	0	ack	40	---	2	3.2	0.1	82	602
+	1.84566	0	2	tcp	1000	---	2	0.1	3.2	102	611

Tableau 4: Format du fichier de sortie.tr

NAM : La conception de protocole demande une compréhension de plusieurs détails, dont le suivi des états d'un grand nombre de nœuds, une analyse de l'échange de messages et doit caractériser les interactions dynamiques pour des trafics concurrents. Habituellement, des traces de paquets sont utilisées pour accomplir ces tâches. Cependant, ces traces ont deux inconvénients majeurs : elles présentent un nombre important de détails, ce qui peut compliquer la compréhension des données, et elles sont statiques, ce qui cache une dimension importante du comportement des protocoles. Les outils de visualisation adressent ce problème en permettant à l'utilisateur de prendre en considération plusieurs informations très rapidement, d'identifier visuellement les modèles de communication et de mieux comprendre les interactions et les causalités.

NAM est un outil d'animation basé sur Tcl/TK pour l'observation des traces de paquet. Les données utilisées par NAM peuvent provenir d'un simulateur ou de tests sur des réseaux réels. Il supporte l'affichage de la topologie, l'animation des échanges de paquets et des outils d'inspection de données divers. NAM a été crée par le laboratoire LBL et s'est considérablement développé durant les dernières années. Le développement de NAM est en collaboration avec le projet VINT. Actuellement, il est développé à ISI dans les projets CONSER (http://www.isi.edu/conser/index.html) et SAMAN (http://www.isi.edu/saman/index.html).NAM interprète un fichier de trace contenant des évènements réseau indexés par le temps de différentes manières. Ces évènements sont principalement les arrivées, départs et suppression de paquets, rupture de lien. Pour les simulations de réseau sans fil, la localisation et les mouvements des nœuds s'ajoutent aux évènements interprétés.

NAM est exécuté avec comme paramètre le fichier enregistré. Lorsqu'on exécute NAM, une fenêtre de travail NAM est créée. Il est possible de faire tourner plusieurs animations avec une seule instance, ce qui permet de mieux comparer certain protocole. On peut entre autre régler le pas de la simulation (de 8µs à 800ms), zoomer sur des zones de la simulation, et manipuler la lecture : on peut mettre pause à tout moment ce qui donne un « arrêt sur image », revenir, avancer sur les étapes de la simulation ce qui permet d'examiner des occurrences particulières. La taille des objets dépend de leurs caractéristiques : l'épaisseur des liens dépend du débit du lien et la taille des paquets dépend de leur longueur en bits et de la bande passante sur le lien. La couleur des paquets peut être utilisée pour

plusieurs raisons ; dans ce cas, elle différencie deux flux de données différents. Les paquets se déplacent de nœuds en nœuds le long des liens et sont mis en file d'attente quand un lien est saturé.

De la même manière que s'est introduite la mobilité dans NS-2, la visualisation de la mobilité dans NAM s'est faite par l'ajout d'extensions. Les nœuds mobiles apparaissent comme les autres nœuds à la différence près qu'ils ne sont pas reliés entre eux.

4.5 Extension pour la mobilité

De nombreux laboratoires de recherche emploient NS-2 pour tester la réaction de nouveaux protocoles dans divers cas de figure. Dans cette section, deux extensions introduites par l'université de Manheim, l'université de Colombie respectivement seront présentés. Il s'agit généralement de fichier modifié ou de nouveaux fichiers introduits dans une version de NS-2. Le code est gratuit et disponible sur les sites respectifs.

NOAH : Cette première extension présentée joue un rôle important dans la gestion de la mobilité dans NS-2. Effectivement, on verra que l'extension CIMS présentée ci-dessous utilise l'agent Noah. Cette extension a été implémentée par Jörg Widmer du laboratoire AT&T31 ACIRI32 à Berkeley pour NS-2 version 6 (ns-2.1b6) ou 7(ns-2.1b7). Noah est un nouvel agent de routage sans fil qui supporte uniquement la communication entre les points d'accès et les nœuds mobiles (en contraste avec les agents DSDV, DSR…). Cet agent permet de faire des simulations dans lesquelles le routage multi-sauts entre les nœuds mobiles n'est pas désiré. En plus, l'agent Noah n'envoie pas de paquets de routage. Cette extension consiste donc en l'amélioration de l'implémentation de Mobile IP existante dans NS-2 par le chevauchement des aires de couverture des points d'accès, la sélection intelligente des agents visités, l'amélioration du processus de handoff. Cette extension inclus en plus un modèle simple de propagation de distance : quand les paramètres du modèle de propagation radio CMU ne sont pas disponibles (qualité de réception…), le modèle simple de propagation de distance permet de spécifier la portée des points d'accès comme une distance (pas d'atténuation du signal). Par contre, quand l'information est disponible, le modèle exact sera utilisé.

CIMS : CIMS(Columbia IP Micro-Mobility Suite) est une extension de NS-2 basée sur les versions 6 (ns-2.1b6) ou 7 (ns-2.1b7) disponibles à l'url http://www.comet.columbia.edu/micromobility. L'extension est disponible en deux versions, selon que l'agent NOAH décrit ci-dessus est déjà installé ou non. Elle a été développée par le groupe COMET de l'université de Colombie, en collaboration avec, laboratoires et entreprises. CIMS inclus les implémentations de Cellular IP, Hawaii et Mobile IP Hiérarchique.

Références

[AND00] Andrew T. Campbell, Javier Gomez, Sanghyo Kim, András G. Valkó, and Chieh-Yih Wan, Columbia University, New York Zoltán R. Turányi, Technical University of Budapest Design, August 2000. “Design, Implementation, and Evaluation of Cellular IP”

[BAD98] Nadjib Badache. "La mobilité dans les systèmes répartis". 45 pages, Janvier 1998.

[BON01] Pierre Reinbold and Olivier Bonaventure Infonet “A Comparison of IP mobility protocols” Technical Report infonet-TR-13. Infonet Group, University of Namur, Belgium December 2001

[CAL98] P. Calhoun, G. Montenegro, and C. E. Perkins, ``Mobile IP Regionalized Tunnel Management," Internet draft, Work in Progress, Nov 1998.

[CAL99] P. Calhoun and C.E. Perkins, ``Mobile IP Network Access Identifier Extension," Internet Draft, Work in Progress, May 1999.

[CAM98] Andrew T. Campbell, Javier Gomez. “An Overview of Cellular IP” Center for Telecommunications Research, Columbia University, New York 1998

[CAM99a] A. Campbell et al., "Cellular IP," Internet draft, draft-ietf-mobileip-cellularip-00.txt, Dec. 1999; work in progress.

[CAM99b] A. T. Cambell, S. Kim, J. Gomez, C-Y. Wan “Cellular IP Performance” Columbia University Z. Turanyi, A. Valko Ericsson October 1999 <draft-gomez-cellularip-perf-00.txt>

[CAS99] Castelluccia C., Bellier L., Toward a Unified Hierarchical Mobility Management Framework, 25.6.1999 < http://www.inrialpes.fr/planete/people/ccastel/draft.txt >

[CIM01]CIMS: The Columbia IP micro mobility Suite NS Source Code Distribution for Cellular IP, HAWAII and Hierarchical Mobile IP, April 2001.

[COM??] COMNET III. http://www.compuware.com/products/ecosystems/comnet/.

[COR00] A. O’Neill and S. Corson, “An Approach to Fixed/Mobile Converged Routing,” Tech. Rep. TR-2000-5, University of Maryland, Institute for Systems Research, March 2000.

[DAS00] Subir Das et al., “TeleMIP: Telecommunication-Enhanced Mobile IP Architecture for Fast Intradomain Mobility,” IEEE Personal Communications, vol. 7, no. 4, pp. 50–58, August 2000.

[DAV92] Davies N., Blair G. S., Cheverst K. and Friday A. "Supporting collaborative applications in a heterogeneous mobile environment". Internal Report No MPG-94-18, Computing Department, Lancaster University, Bailrigg, January 1992.

[DAV93] Davies N., Blair G. S., Cheverst K., Friday A, Raven P., and Cross A."Mobile open systems technology for the utilities industries". In Proceeding of the IEEE Colloquium on CSCW Issues for Mobile and Remote Workers, London, U. K., March 1993.

[DEE91] Deering S., *ICMP Router Discovery Messages*, Request for Comments, RFC 1256, Internet Engineering Task Force, Spetembre 1991.

[DEE95] Deering S.E., Internetwork routing for mobile computers, Synopsium on Wireless Access to Distributed Computing, Columbia University Center for Telecommunications Research, Février 1995

[DRO97] R. Droms, “Dynamic Host Configuration Protocol”, Internet Engineering Task Force Request For Comment 2131, Mars 1997.

[DUC92] D. Duchamp and N. F. Reynolds. "Measured performance of wireless LAN". Technical Report, Computer Science Department, Columbia University, NY, UnitedStates, September 1992.

[FOR94] G. H. Forman and J. Zahorjan. "The challenges of mobile computing". IEEE Computer, 27(4), pp 38-47, April 1994.

[GHA01] Mona Ghassemian "Evaluation of Different Handoff Schemes for Cellular IP" . A Dissertation submitted to King’s College, London for the Master of Science by Research Department of Electronic Engineering King’s College London 2001

[GLO??] Glomosim http://pcl.cs.ucla.edu/projects/glomosim/

[GOM00] Andrew T. Campbell, Javier Gomez, Sanghyo Kim, András G. Valkó, and Chieh-Yih Wan "Design, Implementation, and Evaluation of Cellular IP", Columbia University, New York Zoltán R. Turányi, Technical University of Budapest, IEEE Personal Communications August 2000

[HIL95] S. G. Hild. "A brief history of mobile telephony". Technical Report 372, Computer Laboratory, University of Cambridge, England, 1995.

[IMI92] Imienlinski T. and Badrinath B. R. "Querying in highly mobile distributed environments". Proceeding of the 18th VLDB, pp 41-52, August 1992.

[IMI94] Imienlinski T. and Badrinath B. R."Mobile wireless computing : solutions and challenges in data management". CACM, 37(10), pp 18-28, October 1994.

[ION91] Ionadis John, Duchamp Dan, Maguire Gerald Q., IP-based protocols for mobile internetworking, Proceedings SIGCOMM 91, pp.235-245, Zurich, 3-6 Septembre 1991

[JAC88] Jacobson Van, Micheal j. Karels. Congestion Avoidance and Control. Nov 1988.

[JAI91] R. JAIN, The Art of Computer Systems Performance Analysis : Techniques for Experimental Design, Measurement,Simulation and Modeling, John Wiley & Sons, 1991.

[JAI86] R. JAIN et S. ROUTHIER, Packet trains - measurements and a new model for computer network traffic, Journal on Selected Areas in Communications, vol. 4, no. 6, septembre 1986.

[JOH00] Johnson D., Maltz D., *The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks*, 2000 IETF Draft.

[KAR87] Karn, P. and Partridge, C. “Improving Round-Trip Time Estimates in Reliable Transport Protocols,” Computer Communication Review, Vol. 17, no 5, pp.2-7. Aug 1987.

[LAW94] A. M. LAW et M. G. MCCOMAS, Simulation software for communications networks : The state of the art, IEEE Communications Magazine, vol. 34, no. 3, pp. 44–50, mars 1994.

[MAH??] B. MAH, INSANE An Internet Simulated ATM Networking Environment. http:/www.employees.org/~bmah/Software/Insane/.

[MCC93] S. McCanne and V. Jacobson, “The BSD Packet Filter: A New Architecture for User-Level Packet Capture,” USENIX ’93, San Diego, CA.

[MET95] P. Metzger, W. Simpson "IP Authentication using Keyed MD5,", IETF RFC 1828, August 1995.

[MIL92] D. Mills, "Network Time Protocol (Version 3): Specification, Implementation and Analysis", RFC 1305, Mar 1992.

[MOD??] MODLINE. http://www.simulog.fr.

[MON01]Nicolas Montavont Rapport de D.E.A.Informatique "LA MOBILITE DANS LES RESEAUX IP" 2000/2001 Université Louis Pasteur de Strasbourg

[NAG84] J. Nagle, “Congestion Control in IP/TCP Internetworks”, RFC 896, 1984

[NAR98]T. Narten, E. Nordmark, W. Simpson, “Neighbor Discovery for IP Version 6”, Internet Engineering Task Force Request For Comment 2461, Décembre 1998.

[NET??] NetMaker MainStation. http://www.makesys.com/products/index.html.

[NIS??] NIST ATM/HFC Network Simulator. http://w3.antd.nist.gov/Hsntg/prd_atm-sim.html.

[ONE00] A. O’Neill, G. Tsirtsis, and S. Corson, “Edge Mobility Architecture,” internet draft, draft-oneill-ema-01.txt, March 2000, work in progress.

[OPN??] OPNET. http://www.mil3.com.

[PAR97] VD Park, MS Corson, *A highly adaptive distributed routing algorithm for mobile wireless networks*, Proc. INFOCOM'97, Apr. 1997.

[PER94] Perkins, Bhagwat P., *Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for MobileComputers*, Comp. Comm. Rev., Oct. 1994, pp.234-244.

[PER96a] Perkins C., IP Mobility Support, RFC 2002, Octobre 1996.

[PER96b] Perkins C., IP Encapsulation within IP, RFC 2003, Octobre 1996.

[PER97] C.E. Perkins, and D.B. Johnson, ``Route Optimization in Mobile IP,'' Internet Draft, November 1997.

[PER99] C.E. Perkins and P. Calhoun, ``Mobile IP Challenge/Response Extensions,'' Internet Draft, Work in Progress, May 1999.

[PER00] Perkins C., Royer E., Das S., *Ad Hoc On-demand Distance Vector Routing*, 2000, IETF Draft, http://www.ietf.org/internet-drafts/draft-ietf-manet-aodv-06.txt.gz

[PIT93] Pitoura E. and Bhargava B."Dealing with mobility". Issue and Research Challenges. Technical Report CSD-TR-93-070, Department of computer science, Purdue University, November 1993.

[POS81] Postel J., Internet Protocol, Request for comments, RFC 791, Internet Engineering Task Force, Setpembre 1981

[RAM00] Ramjee R., La Porta T., Thuel S., Varadhan K., Salgarelli L., IP micro-mobility support using HAWAII, Internet Draft, draft -ietf-mobileip-hawaii-01.txt, 7 Juillet 2000.

[REI01] Pierre Reinbold "Un cadre générique de comparaison pour la micro-mobilité sous IP". Groupe Infonet, Université de Namur, Belgique 15 octobre 2001

[RIG97] C. Rigney, A. Rubens, W. Simpson, and S. Willens, ``Remote Authentication Dial in User Service (RADIUS),'' Request for Comments 2138, Apr 1997.

[SHE00] Z. D. Shelby, D. Gatzounas, A. T. Campbell, C-Y. Wan, “Cellular IPv6”, Internet draft, draft-shelby-seamoby-cellularipv6-00.txt, IETF Mobile IP Working Group Document, November 2000.

[STE97] W. Stevens, “TCP Slow Start, Congestion Avoidance, Fast Retransmit, and Fast Recovery Algorithms”, RFC 2001, 1997.

[SUN80] Sunshine, Postel, Adressing mobile hosts in the ARPA Internet environment, IEN 135, University of Southern California Information Sciences Institute, Marina del Rey, California, Mars 1980

[TER89] Terakoa, Yokote, Tokoro, *Muse-IP: A Network Layer Protocol for Large Distributed Systems with Mobile Hosts*, rapport technique Sony CSL SCSL-TR-89-003, Proceedings of the 4 th Joing Workshop on Computer, Communications, Juin 1989

[TER91] Terakoa, Yokote, Tokoro, IP-based protocols for mobile internetworking Transparency, Proceedings SIGCOMM 91, pp.209-220, Zurich, 3-6 Septembre 1991

[THO98] S. Thomson, T. Narten, “IPv6 Stateless Address Autoconfiguration”, Internet Engineering Task Force Request for Comments 2462, Décembre 1998.

[VAL99a] Andras Gergely Valko "Design and Analysis of Cellular Mobile Data Networks” High Speed Networks Laboratory Department of Telecommunications and Telematics Technical University of Budapest Ph. D. Dissertation.Budapest, 1999

[VAL99b]András G. Valkó. Ericsson Research.ACM Computer Communication Review,January 1999. “Cellular IP: A New Approach to Internet Host Mobility”

[WAY93] Wayner P. "Stretching the Ether". Byte, February 1993.

Table des figures

Figure 1.1 : Le modèle des réseaux mobiles avec infrastructure 4
Figure 1.2 : Le modèle des réseaux mobiles sans infrastructure 4
Figure 1.3 : Le principe de réutilisation de fréquence 6

Figure 2.1 : Encapsulation de données TCP dans un datagramme IP 11
Figure 2.2 : Exemple d'établissement de connexion TCP réussie 11
Figure 2.3 : Exemple de déconnexion réussie 12
Figure 2.4 : Exemple d'échange de données 13
Figure 2.5 : Exemple de retransmission de paquet perdu 14
Figure 2.6 : Visualisation de la fenêtre glissante de TCP 15
Figure 2.7 : Evolution de la fenêtre de congestion de TCP Tahoe 16
Figure 2.8 : Evolution de la fenêtre de congestion pendant la phase du démarrage lent 16
Figure 2.9 : Evolution de la fenêtre de congestion pendant la phase d'évitement de congestion 17
Figure 2.10 : Exemple de retransmission rapide de paquet perdu 18

Figure 3.1 : Exemple de deux sous – réseaux avec le support de mobilité 24
Figure 3.2: Tunnelage 25
Figure 3.3 : Encapsulation IP dans IP 26
Figure 3.4 : Enregistrement via un foreign agent 28
Figure 3.5 : Dialogue avec un mobile 29
Figure 3.6 : Routage des données 29
Figure 3.7 : Exemple de routage 30
Figure 3.8 : Routage optimisé 30

Figure 4.1 : Réseau d'accès sans fil et Mobile IP 34
Figure 4.2 : Architecture d'un réseau sans fil 35
Figure 4.3 : Routage 36
Figure 4.4 : Topologie du réseau d'accès sans fil 36
Figure 4.5 : Algorithme de routage uplink 37
Figure 4.6 : Algorithme de routage downlink 38
Figure 4.7 : Une vue schématique d'une passerelle Cellular IP 39
Figure 4.8 : Une simple comparaison des différents schémas de handoff de Cellular IP 39
Figure 4.9 : Point de croisement entre l'ancienne et la nouvelle BS 40
Figure 4.10 : Le trafic de signalisation correspondant au semi-soft handoff 41
Figure 4.11: Paging areas 41
Figure 4.12: Création des correspondances dans les PCs par les paquets paging-update 42
Figure 4.13: Mises à jour des PCs pour un hôte mobile 42
Figure 4.14: Paging packet est routé vers l'hôte mobile en utilisant les PCs 43
Figure 4.15 : Hiérarchie utilisant des domaines 45
Figure 4.16 : Schéma Forwarding 48
Figure 4.17 : Schéma Non-Forwarding 48

Figure 5.1 : Un modèle de réseau pour la comparaison des paramètres du handoff 57
Figure 5.2 : La plate-forme Cellular IP 59

Figure 6.1 : La plate-forme Cellular IP 62
Figure 6.2 : La plate-forme Hawaii 62
Figure 6.3 : Débit TCP « Tahoe » pour les deux schémas de handoffs de Cellular IP 63
Figure 6.4 : Comportement de la fenêtre de congestion pour les deux schémas de handoff de Cellular IP 64
Figure 6.5 : Numéros de séquence sous le hard handoff pour différentes vitesses du MH 65
Figure 6.6 : Numéros de séquence sous le semisoft handoff pour différentes vitesses du MH 65
Figure 6.7 : Numéro de séquence maximum et le nombre de hard handoffs en fonction de la vitesse de HM 66

Figure 6.8 : Numéro de séquence maximum et le nombre de semisoft handoffs en fonction de la vitesse de HM 66
Figure 6.9 : Numéro de séquence maximum en fonction de la vitesse de HM pour les deux schémas de handoffs Cellular IP 67
Figure 6.10 : Débit TCP en fonction de la vitesse de HM pour les deux schémas de handoffs Cellular IP67
Figure 6.11 : Numéro de séquence maximum et le nombre de hard handoffs en fonction de la zone de chevauchement..... 68
Figure 6.12 : Numéro de séquence maximum et le nombre de semisoft handoffs en fonction de la taille e la zone de chevauchement 68
Figure 6.13 : Numéro de séquence maximum en fonction de la taille de la zone de chevauchement pour les deux schémas de handoffs Cellular IP..... 69
Figure 6.14 : Nombre de paquets perdus et le nombre de hard handoffs en fonction de la taille de la zone de chevauchement..... 69
Figure 6.15 : Nombre de paquets perdus et le nombre de semisoft handoffs en fonction de la taille de la zone de chevauchement 70
Figure 6.16 : Débit pour différentes implémentations de TCP en appliquant les deux schémas de handoff du protocole Cellular IP 70
Figure 6.17 : Débit TCP (Tahoe) pour les deux schémas de handoffs de Hawaii 71
Figure 6.18 : Comportement de la fenêtre de congestion pour les deux schémas de handoff de Hawaii... 72
Figure 6.19 : Numéros de séquence sous le Forwarding handoff pour différentes vitesses du MH..... 73
Figure 6.20 : Numéros de séquence sous le Non-Forwarding handoff pour différentes vitesses du MH... 73
Figure 6.21 : Numéro de séquence maximum et le nombre de handoffs en fonction de la vitesse de HM sous le Non-Forwarding handoff 74
Figure 6.22 : Numéro de séquence maximum et le nombre de handoffs en fonction de la vitesse de HM sous le Forwarding handoff 74
Figure 6.23 : Numéro de séquence maximum en fonction de la vitesse de HM pour les deux schémas de handoff Hawaii..... 75
Figure 6.24 : Débit TCP en fonction de la vitesse de HM pour les deux schémas de handoffs Hawaii..... 75
Figure 6.25 : Débit pour différentes implémentations de TCP en appliquant les deux schémas de handoff du protocole Hawaii 76
Figure 6.26 : Comparaison de l'effet de la vitesse sur les schémas de handoff de Cellular IP et de Hawaii 77
Figure 6.27 : Comparaison de l'effet du nombre de handoffs sur les schémas de handoff de Cellular IP et de Hawaii 79
Figure 6.28 : Débit pour différentes implémentations de TCP sous les différents schémas de handoff des protocoles Cellular IP et Hawaii 80

Figure 29 : Techniques d'évaluation des performances de réseau..... 87
Figure 30 : Format des paquets TCP 92
Figure 31: Structure du message route-update..... 93
Figure 32: Structure du champ control information 94
Figure 33 : Format de « path setup update message » 95
Figure 34 : Format du «path setup refresh message» 95
Figure 35 : Configuration DHCP..... 96
Figure 36 : Enregistrement v4 96
Figure 37 : Dés-enregistrement auprès de l'agent mère 97
Figure 38 : Routage triangulaire : du correspondant au mobile..... 97
Figure 39 : Routage triangulaire : du mobile au correspondant..... 97
Figure 40 : Communication v6 99

Table des tableaux

Tableau 2.1 : Récapulatif des caractéristiques des différentes versions de TCP 21

Tableau 4.1 : Résumé des opérations des paging caches et routing caches 43
Tableau 4.2 : Les valeurs typiques des différents timers 43

Tableau 5.1 : Comparaison des paramètres de gestion du handoff 57

Tableau 6.1 : Performances des différents schémas de handoffs par rapport aux vitesses de l'hôte mobile selon les numéros de séquence maximum 78
Tableau 6.2 : Comparaison des performances des différentes variantes TCP par rapport au nombre de handoffs selon les débits reçus par l'hôte mobile 79
Tableau 6.3 : Comparaison des performances des différentes variantes TCP par rapport aux differents schémas de handoffs selon les débits reçus par l'hôte mobile 80

Tableau 4: Format du fichier de sortie.tr 106