Gestion avancée de la mobilité dans les réseaux maillés sans fil

Mehdi Bezahaf, Luigi Iannone, Serge Fdida

Gestion avancée de la mobilité dans les réseaux maillés sans fil

M. Bezahaf[*] — L. Iannone[**] — S. Fdida[*]

Université Pierre et Marie Curie - Laboratoire LIP6/CNRS[*]
{Mehdi.Bezahaf, Serge.Fdida}@lip6.fr

Université Catholique de Louvain - IP Networking Lab (INL)[**]
Luigi.Iannone@uclouvain.be

RÉSUMÉ. *En dépit des efforts déployés dans le domaine de la mobilité dans le contexte des réseaux maillés sans fil (Wireless Mesh Networks -WMN), la gestion de mobilité souffre toujours d'un manque de recherche. Plusieurs travaux intéressants ont été menés sur ce type de réseaux, dont quelques-uns qui proposent des solutions pour gérer la mobilité des clients. Cependant, souvent ces solutions exigent aux utilisateurs de modifier ou rajouter des modules supplémentaires dans la pile protocolaire de leur équipement. Dans ce papier, nous abordons les problèmes liés à la mobilité dans les réseaux maillés sans fil et nous proposons une solution efficace à ces problèmes. Nous concevons et implémentons* Enhanced Mobility Management, *une approche améliorée de la gestion de mobilité, qui détecte efficacement les utilisateurs lors de leurs déplacements. EMM se base sur l'utilisation du cache NDP (Neighbor Discovery Protocol) des clients afin de garder une trace de leur ancienne association au cours de leurs déplacements. Cette trace est utilisée pour mettre à jour les routes sans nécessiter l'installation de logiciels supplémentaires.*

ABSTRACT. *Despite considerable efforts, mobility management in Wireless Mesh Networks (WMN) remains an open issue. Several high performance solutions can be found in the literature, however, they all have the same requirement that refrains them from being widely adopted: they need either modifications or additional modules into the protocol stack of users' equipment. In this paper, we investigate the mobility problem in WMNs and propose a new efficient solution named* Enhanced Mobility Management *(EMM), which does not rely on any modification or additional software on the client side, thus being totally transparent for end-users. EMM takes advantage of the existing Neighbor Discovery Protocol (NDP) cache to keep a track of the last client association and uses this information to trigger an update in order to re-route packets. The measurements we performed show how EMM is able to greatly improve performances.*

MOTS-CLÉS : *Réseaux maillés sans fil, gestion de mobilité, IEEE 802.11.*

KEYWORDS: *Wireless Mesh Networks, mobility management, IEEE 802.11.*

1. Introduction

La gestion de mobilité est un paramètre très important dans les réseaux actuels, puisque les utilisateurs sont de plus en plus mobiles en raison du déploiement massif des technologies sans fil. Cette nouvelle génération de clients qui cherchent à communiquer durant leurs déplacements sans aucune contrainte de connectivité, sans aucun logiciel additionnel à installer et où le changement de réseau est complètement transparent, a poussée la communauté des chercheurs à proposer une nouvelle architecture plus adéquate à leurs besoins. Les réseaux maillés sans fil (Wireless Mesh Network - WMN) [AKY 05] font partie de cette catégorie de réseaux. Les WMNs sont une classe émergente des réseaux sans fil, capable de s'organiser et de se configurer dynamiquement. Ils prennent le principe d'un réseau sans fil basé sur la transmission multi-sauts *c.-à-d.*, les communications entre deux noeuds peuvent être supportées par plusieurs noeuds intermédiaires (appelés Wireless Mesh Router - WMR) dont le rôle est de retransmettre les informations. Leur architecture à deux niveaux concentre le routage sur une partie sans fil stable (le premier niveau - backbone), composée de WMRs qui offrent une connectivité à des clients mobiles (le deuxième niveau). Dans ce contexte, le challenge est de préserver les connexions ouvertes d'un client mobile quelque soit le type de ses déplacements. La gestion de la mobilité dans les réseaux maillés sans fil est répartie en deux phases : une phase de détection et localisation et une phase de transition. La première phase consiste à détecter le plus instantanément possible les clients lors de leurs déplacements et de déterminer leur position dans le réseau à tout moment et avec une bonne précision (*c.-à-d.*, avoir une vue réaliste du réseau). La deuxième phase complète la première en mettant à jour les informations de routage afin de réduire au minimum le temps de déconnexion, éviter la perte de paquets et maintenir les connexions déjà ouvertes. Plusieurs propositions abordent la gestion de mobilité avec de bonnes performances. Néanmoins, elles peinent à être réellement et largement déployées en raison de leur nécessité de modifier la pile protocolaire ou d'installer un logiciel supplémentaire dans l'équipement du client.

Dans ce papier, nous proposons l'approche *Enhanced Mobility Management* (EMM) pour gérer efficacement la mobilité des clients, sans avoir besoin d'installer un logiciel supplémentaire ou de modifier la pile protocolaire de ces utilisateurs. L'approche EMM est basée sur l'utilisation du cache NDP (Neighbor Discovery Protocol) du client. Plus particulièrement, un WMR injecte une entrée particulière dans le cache NDP de ses clients locaux. Cette entrée est utilisée quand un client change d'association, dans le but de reconnaître sa précédente association et de mettre à jour sa nouvelle position (nouvelle association). Les mesures que nous avons effectuées sur notre réseau de test déployé au LIP6 "MeshDVNet" ([IAN 05a], [IAN 05b]), montrent que l'approche EMM est capable d'améliorer énormément les performances de gestion de mobilité en réduisant considérablement le temps de déconnexion et donc les pertes de paquets.

Le reste du papier est organisé de la sorte. Dans la section 2, nous passons en revue les principales solutions de gestion de mobilité dans les réseaux maillés, avant d'analyser dans la section 3 le comportement original et les problèmes liés à la gestion

de mobilité dans MeshDVNet. Nous présentons alors notre approche EMM dans la section 4, suivie des mesures effectuées sur le réseau test MeshDVNet et la discussion des résultats obtenus. La Section 6 récapitule nos contributions et conclut le papier.

2. État de l'art

Les réseaux maillés sans fil [AKY 05] sont souvent sollicités pour diverses raisons : réseaux communautaires, réseaux d'entreprises ou domestiques et réseaux de secteur locaux ou métropolitains. Certains industriels ont même commercialisé des WMNs ([NORT], [CISCO], [STRIX]). L'un des plus connu est le projet MIT Roofnet [BIC 05]. L'étude Roofnet est plus ciblée sur la maintenance et l'optimisation des routes que sur la mobilité des clients. Il existe aussi quelques communautés de WMN telles que NYC wireless [NYCW] et Quail Ridge Wireless Mesh Network [MOH 07].

Dans un tel contexte, le déplacement d'un client abouti souvent sur un changement de sous-réseau, par conséquent l'adresse IP du client n'aura aucune signification dans le réseau visité. Le fait de changer l'adresse IP du client à chaque changement de Point d'Accès n'est pas une bonne solution, car toutes les connexions TCP seront perdues et toutes les applications stockant l'adresse IP souffriront. Pour résoudre ce problème, différentes solutions ont été présentées, telles que Mobile IP ([PER 02], [JOH 04]), HIP [HEN 07] et le mécanisme RendezVous [EGG 04]. Ces approches, permettent aux clients de maintenir la même identité (Identifiant ou adresse IP) pendant la visite d'autres sous-réseaux, comme elles permettent aussi leur localisation. Depuis sa conception, plusieurs améliorations ont été apportées à la version originale de Mobile IP. On peut citer HMIP [CAS 00], qui consiste à hiérarchiser ses agents pour éviter un surcoût de signalisation dans les cas où un client s'éloigne de son réseau mère ; comme on peut citer FMIP [KOO 05], qui consiste à anticiper le déplacement d'un client afin d'obtenir sa nouvelle adresse temporaire avant son déplacement réel. Dans iMesh ([NAV 05], [NAV 06]), les auteurs utilisent une variante de Mobile IP (TMIP) qui est basée sur un serveur centralisé (MLR). Dans d'autres solutions comme HAWAII [RAM 02] et Cellular IP [CAM 00], le client lui-même envoie un paquet de mise à jour à la passerelle du réseau après un déplacement, ce qui permet aussi aux routeurs sur le chemin de mettre à jour leur table de routage. SMesh [AMI 06] utilise le protocole DHCP pour avoir la position des clients. Une adresse IP est attribuée à un client pour une durée de validité de 2 secondes "bail". A expiration du bail, le client transmet une requête DHCP afin de renouveler son bail. Cette requête permet aux APs de positionner le client.

Néanmoins, notre objectif principal est d'avoir un mécanisme efficace de récupération de connexion quand un client se déplace, sans la nécessité d'une pré-installation pour ce dernier, ce qui n'est pas le cas dans la plupart des solutions existantes et les protocoles qui souvent poussent les clients à faire une installation logicielle supplémentaire.

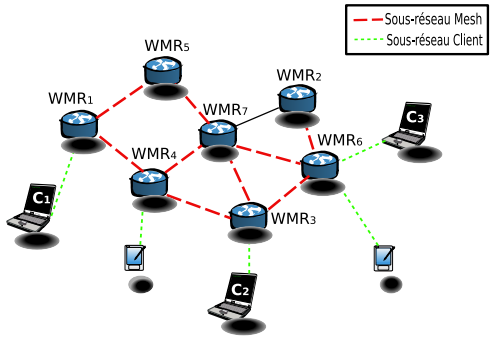


Figure 1 – Exemple de déploiement de MeshDVNet

3. Gestion de la mobilité dans MeshDVNet

Afin d'acquérir une compréhension plus profonde des problèmes de mobilité existants dans les WMNs et pour avoir une étude réaliste, nous avons analysé et effectué plusieurs mesures sur le réseau MeshDVNet ([IAN 05b], [IAN 05a]).

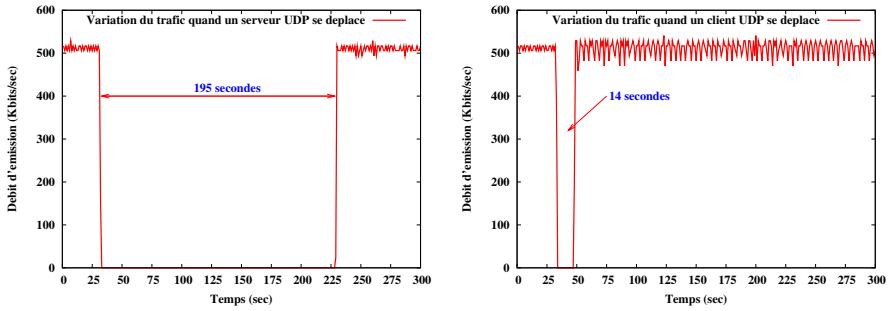
Basé sur IPv6, le réseau MeshDVNet est un WMN de test déployé au LIP6 qui offre une connexion sans fil aux clients et leur permet de communiquer sans aucune pré-installation nécessaire. Pour avoir une vue en temps réel de notre réseau de test et pour contrôler le bon fonctionnement des WMRs, nous utilisons la page Web de supervision qui est publiquement disponible (seulement pour des connexions IPv6) au :

http ://www.infradio-jussieu.lip6.fr/supervision/supervision-mesh-kennedy.html

MeshDVNet est découplé en deux sous-réseaux (Fig. 1) : un sous-réseau formé d'un ensemble de WMRs qui constitue le backbone et un deuxième formé d'un ensemble de clients. Nommés MeshDVbox, les routeurs (WMRs) utilisés dans MeshDVNet sont des Soekris net4521 sous le système d'exploitation Linux (Crux comme distribution), sur lequel le protocole MeshDV tourne. MeshDVbox est équipé de deux interfaces sans fil, une pour communiquer avec les autres WMRs et une pour communiquer avec ses clients locaux.

Afin de comprendre comment la mobilité est gérée dans la version originale de MeshDVNet et pour souligner les problèmes actuels, nous présentons ci-dessous comment une communication entre deux clients, associés à des WMRs différents, est établie. Nous supposons que le client C_1, associé à WMR_1, ouvre une connexion vers le client C_2 qui est associé à WMR_3 (Fig. 1). La communication est établie de la façon suivante :

1 : En utilisant un serveur DNS, le client C_1 peut récupérer l'adresse IP du client C_2 s'il ne la connaît pas. De cela et du fait que C_1 et C_2 sont dans le même sous-réseau logique, le client C_1 envoi d'abord un



(a) Quand le serveur se déplace (b) Quand le client se déplace

Figure 2 – Performances de mobilité dans MeshDV lors d'un trafic UDP

Neighbor Solicitation (NS) en multicast afin d'obtenir l'adresse MAC du client C_2. En recevant ce paquet de sollicitation, le WMR_1 envoi un message MCREQ[1] aux autres WMRs pour découvrir où est-ce que le client C_2 est associé.

2 : Le WMR_3 reçoit la demande en multicast (MCREQ) et répond avec un paquet CRREP[2].

3 : Quand le WMR_1 reçoit le message CRREP, il répond avec un Neighbor Advertisement (NA) au Neighbor Solicitation du client C_1 contenant l'adresse IP du client C_2 associée à l'adresse MAC du WMR_1.

4 : Par la suite, tous les paquets envoyés du client C_1 au client C_2 seront capturés par WMR_1, encapsulés dans des paquets IPv6 et envoyés à WMR_3.

5 : Le WMR_3 décapsulera les paquets reçus pour les retransmettre au client C_2.

Dans un réseau maillé sans fil, la détection des déplacements d'un client peut être réalisée de deux manières différentes : soit c'est l'ancien WMR qui détecte que son client local a changé d'association (détection automatique), soit c'est le nouveau WMR qui détecte un nouveau client et le notifie à l'ancien WMR (détection réactive). MeshDV est basé sur la première approche pour gérer la mobilité de ses clients. Si un client change d'association (*c.-à-d.* de WMR), pendant une communication déjà ouverte auparavant, il arrive à se reconnecter au niveau de la couche physique avec le nouveau WMR, mais la communication avec ses correspondants ne sera pas rétablie complètement et le problème vient de l'ancien WMR où le client était connecté.

En effet, pour ce WMR son client est toujours connecté à son interface, ce qui n'est pas le cas. Donc tous les paquets destinés à ce client seront acheminés par l'ancien WMR à une destination non existante. Ce problème est dû à la mauvaise gestion de mobilité dans MeshDV qui est basée sur la détection des clients, faite par le driver de la carte sans fil. Dans notre réseau de test, le driver utilisé par les cartes sans fil des WMRs est Madwifi [MAD], celui-ci détecte la déconnexion des clients après trois minutes de leur départ. Donc, les WMRs gardent des informations erronées de leurs

1. MCREQ (Multicast Client REQuest) est un paquet multicast, utilisé pour rechercher des clients distants, consiste à trouver le WMR qui gère le client recherché [IAN 05a].
2. CRREP (Client Request REPly) est un paquet unicast, utilisé pour répondre à une requête MCREQ [IAN 05a].

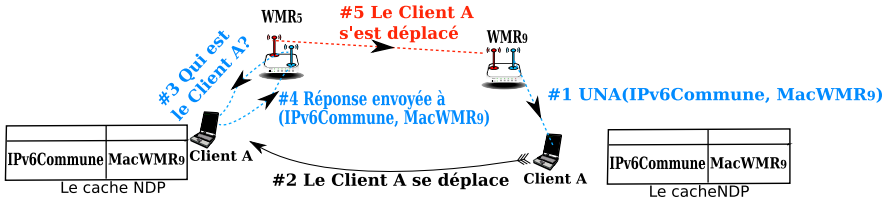


Figure 3 – Exemple de gestion de mobilité avec l'approche EMM

clients locaux qui se sont déplacés, et prennent plus de trois minutes pour se rendre compte. La réduction du timer, au niveau du driver, permet de détecter rapidement un déplacement, mais les beacons de contrôle surchargent enormement le medium.

Nous avons expérimenté différents scénarios de mobilité qui présentent quelques problèmes de performance. Nous nous sommes focalisés sur le temps de déconnexion que le client subit quand il se déplace. Pour cela nous avons utilisé trois types différents de trafic (*c-à-d.* TCP, UDP et Ping) entre deux clients pendant 300 secondes et avons observé le temps de déconnexion quand l'un d'entre eux se déplace. Notez que nous utilisons le programme Iperf [TIR 05] pour générer le trafic TCP et UDP. Comme clients nous avons utilisé des ordinateurs portables Compaq nx7000, sous le système d'exploitation Fedora Core 3. La Figure 2 représente le débit d'émission pendant chaque seconde. Quand le serveur Iperf se déplace pendant un trafic UDP (Fig. 2(a)), nous avons obtenu un temps de déconnexion de trois minutes (détection automatique). Dans le cas où le client Iperf se déplace (Fig. 2(b)), ce dernier se connecte instantanément au nouveau WMR et continu d'envoyer des paquets UDP avec l'adresse MAC destination égale à l'adresse MAC de l'ancien WMR. Cela est dû au cache du protocole NDP (Neighbor Discovery Protocol) [NAR 98], qui n'est pas rafraîchi immédiatement après la connexion physique. Ces deux scénarios sont très représentatifs, puisque le ping et le trafic TCP présentent un comportement très semblable à celui observé dans le premier cas [BEZ 07]. Dans la section suivante, nous proposons notre solution, basée sur le cache NDP, qui résout efficacement tous les problèmes vus dans cette section.

4. L'approche Enhanced Mobility Management

Enhanced Mobility Management (EMM) est une approche améliorée de la gestion de mobilité des clients, où la détection de leurs déplacements est faite entièrement par le nouveau WMR (détection réactive). Notre approche est basée sur le cache NDP qui contient essentiellement les adresses IP des correspondants avec qui un client communique, l'adresse MAC de la passerelle pour les atteindre et inclut aussi des informations sur l'état de leur joignabilité.

Le routeur modulaire Click [KOH 00] est utilisé comme infrastructure logicielle de routage. Dans notre implémentation, chaque WMR utilise le module Click au ni-

veau noyau. Cela signifie que le traitement et le filtrage de tous les paquets reçus sont effectués au niveau Click au lieu qu'ils soient fait au niveau application. Dans l'approche EMM, chaque WMR du réseau, au démarrage, rajoute la même adresse IPv6 en lien locale nommée *Adresse Commune* à son interface client. Cette adresse est statique et peut être manuellement modifiée. Elle n'est jamais utilisée pour échanger des données mais uniquement dans le but de gérer la mobilité des clients. Le message Unsolicited Neighbor Advertisement (UNA) est utilisé pour mettre à jour le cache NDP des clients pendant leurs déplacements (modifier l'entrée de l'*Adresse Commune*).

Présenté dans le RFC 2461 [NAR 98], le message UNA est utilisé par un noeud pour informer ses voisins directs du changement de son adresse de couche liaison. Le principe utilisé dans EMM est similaire au mécanisme de cookies dans les navigateurs Web *c.-à-d.*, c'est les clients eux mêmes qui gardent l'information de leur dernière association WMR. Pour mieux comprendre comment l'injection de ces cookies dans le cache NDP peut aider à gérer la mobilité des clients, prenons le scénario de la Figure 3. Lors de l'arrivée du client A dans le réseau, nous supposons que dans un premier temps il s'associe au WMR_9 puis, il change d'association et passe au WMR_5. Dans ce contexte, le mécanisme EMM NDP cookie fonctionne comme suit :

---

1 : A l'arrivée du Client A dans le réseau et après sa connexion au WMR_9, il reçoit de ce dernier un message UNA *"associe l'Adresse Commune à l'adresse MAC du WMR_9 et enregistre cette information dans ton cache NDP"*. Cela implique une mise à jour du cache NDP du Client A.

2 : Le Client A garde l'Adresse Commune associée à l'adresse MAC du WMR_9 dans son cache NDP pendant son déplacement.

3 : Le Client A change d'association et passe du WMR_9 au WMR_5. Après s'être connecté au WMR_5, ce dernier récupère l'adresse MAC du Client A et dérive son adresse IP[3]. Par la suite, il envoi un Neighbor Solicitation (NS) au Client A avec l'adresse IPv6 source égale à l'Adresse Commune et l'adresse MAC source égale à l'adresse MAC du WMR_5 *"Qui est le Client A ?"*.

4 : Le paquet NS envoyé par le WMR_5 n'est pas utilisé dans le but d'acquérir l'adresse MAC du Client A, qui est déjà obtenue de la couche 2, mais pour connaître l'ancienne association du Client A. Donc à la réception de cette sollicitation, le Client A consulte son cache NDP, où il trouve la cookie (Adresse Commune associée à l'adresse MAC du WMR9). Ainsi, il répond au WMR_5 avec l'adresse MAC destination égale à celle du WMR_9.

5 : Etant donné que l'interface sans fil du WMR_5 est configurée en mode promiscuous, le WMR_5 arrive à voir la réponse du Client A qui est récupérée par le module Click et renvoyée au niveau applicatif. A cette étape, le WMR_5 peut extraire l'adresse MAC du WMR_9 et dériver l'adresse IP du WMR_9 (en se basant sur le EUI-64). Une fois l'adresse IP de l'ancienne association du Client A connue, le WMR_5 n'a plus qu'à envoyer un message CWIT[4] au WMR_9 dans le but de notifier le déplacement du Client A. Au même moment, le WMR_5 envoi un message UNA au Client A *"associe dans ton cache NDP l'Adresse Commune à l'adresse MAC de WMR_5"*, afin de mettre à jour la cookie dans le cache NDP du Client A.

6 : Après la réception du message CWIT envoyé par le WMR_5, si le WMR_9 reçoit un paquet destiné au Client A, il le droppe et renvoi un message Client ERRor (CERR) au WMR qui a généré le paquet (WMR distant) *"le Client A n'est plus associé à moi"*.

7 : En recevant le message CERR, automatiquement le WMR distant envoie à nouveau un paquet MCREQ afin de localiser la nouvelle association du Client A.

La Figure 4 montre le diagramme temporel du mécanisme de gestion de mobilité des clients. Comme nous pouvons le voir, la Figure présente les différents paquets

3. Notez que nous utilisons le 64-bit Extended Universal Identifier (EUI-64) pour obtenir l'adresse IP à partir de l'adresse MAC ([IEE 97], [NAR 99]).
4. CWIT (Client WIThdraw) est un message envoyé en unicast, utilisé par le nouveau WMR afin d'informer l'ancien WMR du déplacement de son client [BEZ 07].

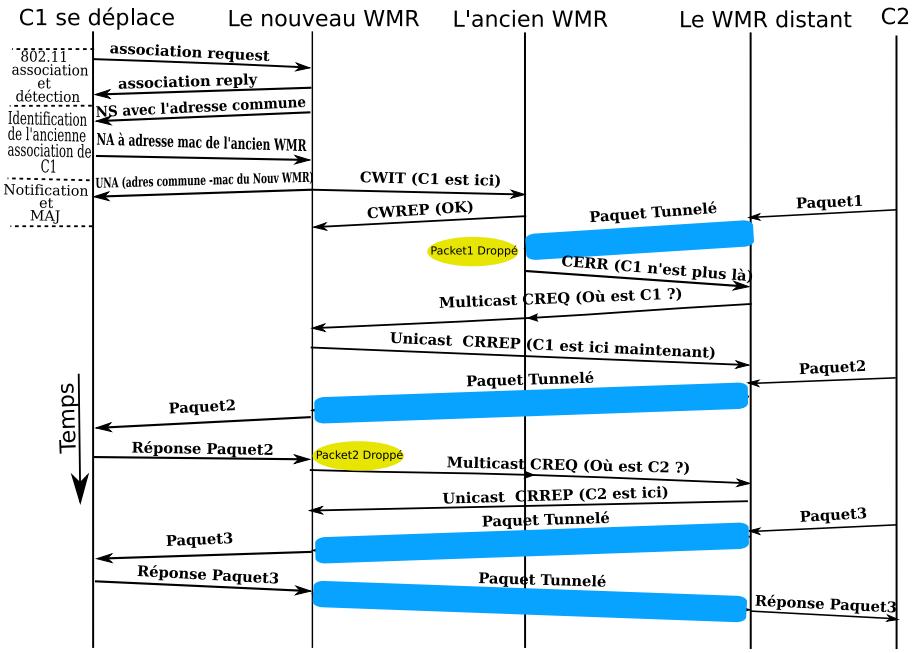


Figure 4 – Diagramme temporel de gestion de mobilité d'un client dans EMM

échangés quand le client C_1 se déplace de "l'ancien WMR" au "Nouveau WMR" durant sa communication avec le client C_2, connecté au WMR distant.

5. Evaluation de l'approche EMM

Afin d'évaluer notre travail, nous avons effectué le même ensemble de test décrit dans la section 3. Nous avons mesuré les délais de déconnexion et évalué le comportement de EMM quand les clients changent d'association. Pour un trafic UDP entre les clients, nous avons obtenu des résultats satisfaisants avec un temps de déconnexion qui est moins de la seconde (Figure 5). Dans le cas où le client qui génère le trafic UDP se déplace, le nouveau WMR doit tout à bord détecter ce client puis, faire une recherche des correspondants de ce dernier. La Figure 5 montre bien que le débit de transmission entre la seconde 233 et la seconde 234 diminue de 500 Kbits/s à 120 Kbits/s, ce qui signifie que la communication est perturbée pendant moins d'une seconde car vers la fin de la seconde la communication reprend (120 Kbits/s). De plus, entre la seconde 234 et la seconde 235 le débit de transmission augmente de 120 Kbits/s à 420 Kbits/s, ce qui signifie qu'il n'y a pas eu de perte de paquets. Notez que pour cette expérimentation, nous ne pouvons pas avoir le temps exact de déconnexion car l'outil utilisé (Iperf) ne nous permet pas d'avoir une granularité inférieur à la seconde.

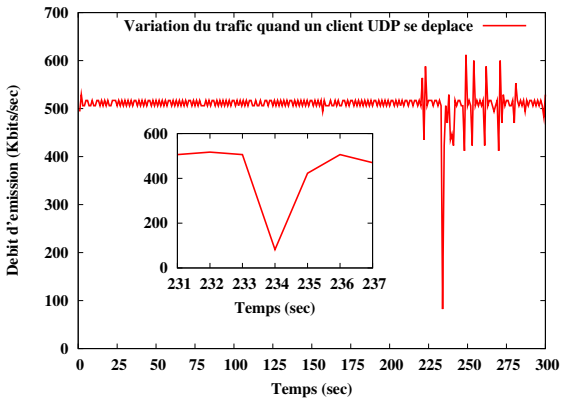
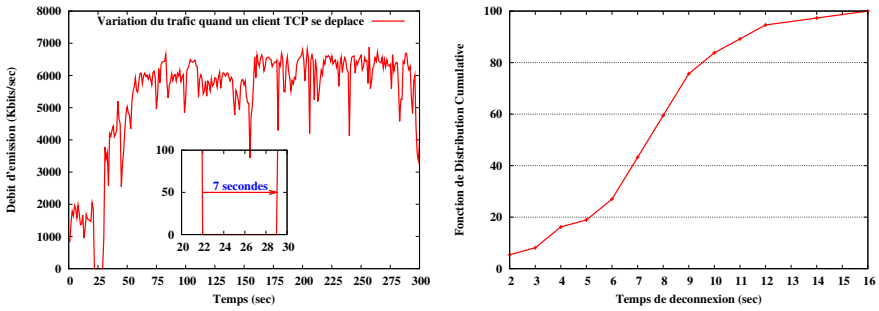


Figure 5 – Performance de l'approche EMM lors d'un trafic UDP



(a) Quand le client se déplace (b) CDF de temps de déconnexion d'un client

Figure 6 – Performances de l'approche EMM lors d'un trafic TCP

Dans le cas d'un trafic TCP entre les clients, la Figure 6(a) montre bien que les résultats ne sont plus les mêmes comparés à ceux obtenus avec un trafic UDP. Comparé au temps de déconnexion d'un client dans la version originale de MeshDV, qui était de 3 minutes, le résultat est nettement meilleur. Malgré cette amélioration, le temps de déconnexion est maintenant de quelques secondes. Cela est dû à la façon dont TCP gère la retransmission des paquets. En effet, en cas de perte de paquets, le protocole TCP utilise le timer RTO (Retransmission TimeOut) dans le but d'assurer la délivrance de ces derniers. Durant RTO secondes d'attente, si un paquet n'est pas acquitté, le RTO double et le paquet est retransmis. Quand un client se déplace et se réassocie à un nouveau WMR, ce dernier en recevant le premier paquet TCP du client ne sait pas à quel WMR est associé le correspondant de ce client. Donc, il envoie un paquet MCREQ sur le backbone et droppe ce paquet. A la fin du RTO, le paquet n'a pas été acquitté, ce qui fait que la valeur du RTO double et le paquet est retransmis. Dans ce cas là, le correspondant reçoit le paquet et l'acquitte. Cependant le WMR à

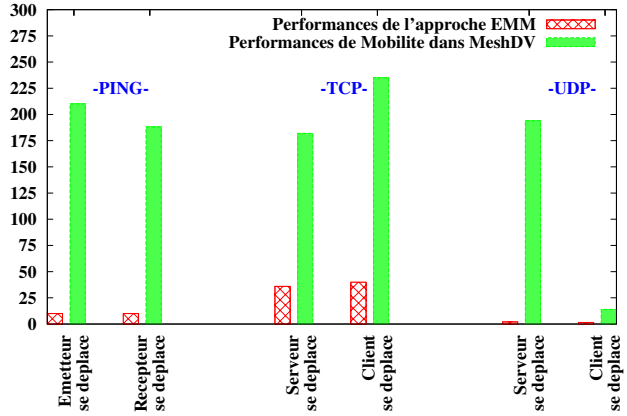


Figure 7 – Comparaison de temps de déconnexion entre la gestion de mobilité dans MeshDV et notre approche

qui le correspondant est associé (WMR distant) fait suivre l'acquittement à l'ancien WMR et non au nouveau. L'ancien WMR droppe ce paquet et notifie au WMR distant que le client à qui le paquet était destiné n'est plus associé à son interface en lui envoyant un message CERR. Cela signifie que la valeur du RTO double une deuxième fois. Ce mécanisme de retransmission, qui double à chaque perte de paquet, rajoute beaucoup de délais. De plus les cartes sans fil dans certains cas, lors du déplacement d'un client, effectuent un scan complet des canaux ce qui augmente la latence de transition IEEE 802.11 d'approximativement 6.9 secondes durant nos tests. Dans ce cas, le temps de déconnexion peut augmenté jusqu'à 30 secondes même si la détection du nouveau client et la mise à jour des tables du nouveau WMR s'effectuent instantanément après la connexion physique [BEZ 07].

La Figure 6(b) représente la fonction de distribution cumulative du temps de déconnexion d'un client en déplacement, lors d'une communication TCP et avec une transition instantanée de la couche IEEE 802.11 (*c.-à-d.* sans que le driver de la carte sans fil ne fait un scan complet des canaux). Du graphe, nous pouvons remarquer que le temps maximum de déconnexion est de 16 secondes, et que dans 95% des cas le temps de déconnexion est inférieur à 12 secondes. Nous remarquons aussi que dans quelques cas le temps de déconnexion peut être court (2 à 5 secondes).

L'histogramme de la Figure 7, représente une comparaison visuelle entre l'approche EMM et l'ancienne version de la gestion de mobilité des clients dans MeshDV. L'approche EMM diminue nettement le temps de déconnexion de 3 minutes à quelques secondes voir même moins de la seconde dans certains cas, ce qui n'est pas une amélioration négligeable.

6. Conclusions

Les solutions existantes pour la gestion de mobilité dans les réseaux maillés sans fil, constituent une sorte de paradoxe, car d'une part, elles offrent des bonnes performances, et d'autre part, elles ont des difficultés à être largement déployées dans des plateformes réelles. La raison d'un tel déficit pourra s'expliquer par le fait qu'elles exigent des installations spécifiques sur les dispositifs d'utilisateurs finaux. Le travail présenté dans ce papier représente nos efforts pour résoudre le problème de gestion de mobilité sans aucun impact, quelque soit l'équipement utilisé par l'utilisateur final. Notre proposition nommée *Enhanced Mobility Management* (EMM) est le résultat d'une analyse profonde que nous avons menée sur le réseau MeshDVNet, déployé au Laboratoire d'Informatique de Paris 6. L'approche EMM utilise les avantages du cache NDP, présent dans toutes les piles protocolaires standards, en injectant une entrée particulière (Adresse Commune). Une telle entrée est utilisée pour récupérer l'identité de l'ancienne association du nouveau client, permettant alors le re-routage des paquets et ainsi la maintenance de la communication en cours avec un temps de déconnexion relativement court. Comme nos mesures le montrent clairement, EMM comparé à la proposition originale de MeshDVNet améliore énormément les performances de gestion de mobilité pour tous types de trafic. Par exemple, dans le cas d'un trafic TCP, le temps de déconnexion est réduit à moins de 20% de la valeur originale. Même plus, dans le cas où une machine qui génère un trafic UDP se déplace, le temps de déconnexion est alors réduit à moins de 0.5% de la valeur originale. Quelques améliorations sont en cours pour éviter la perte des paquets perdus lors d'un déplacement.

7. Remerciements

Nous tenons à remercier Pierre-Emmanuel Le Roux d'avoir mis à contribution ses talents de programmeur. Ce travail de recherche à été subventionné par le Projet IST-WIP (contrat 27402) et par le Projet GigaCom.

8. Bibliographie

[AKY 05] AKYILDIZ I., WANG X., WANG W., « Wireless mesh networks : a survey », *Computer Networks - Elsevier Science*, , n° 47, 2005.

[AMI 06] AMIR Y., DANILOV C., HILSDALE M., MUSĂLOIU-ELEFTERI R., RIVERA N., « Fast handoff for seamless wireless mesh networks », *ACM Press*, , 2006, p. 83–95.

[BEZ 07] BEZAHAF M., « Fast mobility in wireless mesh networks », Master's thesis, University Pierre et Marie Curie (Paris 6), 2007.

[BIC 05] BICKET J., BISWAS S., AGUAYO D., MORRIS R., « Architecture and evaluation of an unplanned 802.11b mesh network », *International Conference on Mobile Computing and Networking (MobiCom)*, , 2005.

[CAM 00] CAMPBELL A., GOMEZ J., KIM S., VALKO A., WAN C., TURANYI Z., « Design, Implementation, and Evaluation of Cellular IP », *IEEE Personal Communications*, , 2000.

[CAS 00] CASTELLUCCIA C., « HMIPv6 : A Hierarchical Mobile IPv6 Proposal », *ACM Mobile Computing and Communication Review (MC2R)*, , 2000.

[EGG 04] EGGERT L., LIEBSCH M., « Host Identity Protocol (HIP) Rendezvous Mechanisms, draft-eggert-hip-rendezvous », IETF, Jul. 2004.

[HEN 07] HENDERSON T., « End-host mobility and multihoming with the host identity protocol, draft-ietf-hip-mm-05 », IETF, Mar. 2007.

[IAN 05a] IANNONE L., « MeshDV : Implementation Draft », *Technical Report*, , 2005.

[IAN 05b] IANNONE L., S.FDIDA, « MeshDV : A Distance Vector mobility-tolerant routing protocol for Wireless Mesh Networks », *IEEE ICPS Workshop on Multi-hop Ad hoc Networks : from theory to reality (RealMAN'06)*, , 2005.

[IEE 97] IEEE, « Guidelines for 64-bit Global Identifier (EUI-64) Registration Authority », IEEE Standards tutorials., 1997.

[JOH 04] JOHNSON D., PERKINS C., ARKKO J., « Mobility support in ipv6, RFC 3775 », IETF, 2004.

[KOH 00] KOHLER E., MORRIS R., CHEN B., JANNOTTI J., KAASHOEK F., « The click modular router », , 2000, ACM Transaction on Computer Systems (TOCS).

[KOO 05] KOODLI R., « Fast Handovers for Mobile IPv6, rfc 4068 », IETF, juillet 2005.

[MOH 07] MOHAPATRA P., WU D., GUPTA D., « Quail Ridge Wireless Mesh Network : Experiences, Challenges and Findings », *International Conference on Testbeds and Research Infrastructures for the Development of Networks & Communities*, , 2007.

[NAR 98] NARTEN T., NORDMARK E., SIMPSON W., « Neighbor Discovery for IP Version 6 (IPv6), RFC 2461 », IETF, Dec. 1998.

[NAR 99] NARTEN T., « Neighbor Discovery and Stateless Autoconfiguration in IPv6 », *IEEE Internet Computing*, vol. 3, n° 4, 1999, p. 54–62, IEEE Educational Activities Department.

[NAV 05] NAVDA V., KASHYAP A., DAS S., « Design and Evaluation of iMesh : an Infrastructure-mode Wireless Mesh Network », *IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WOWMOM)*, , 2005.

[NAV 06] NAVDA V., GANGULY S., KIM K., KASHYAP A., NICULESCU D., IZMAILOV R., HONG S., DAS S., « Performance Optimizations for Deploying VoIP Services in Mesh Networks », *IEEE Journal on Selected Areas in Communication (JSAC)*, , 2006.

[PER 02] PERKINS C., « IP Mobility Support for IPv4, RFC 3344 », IETF, Aug. 2002.

[RAM 02] RAMJEE R., VARADHAN K., SALGARELLI L., THUEL S., WANG S., LA PORTA T., « HAWAII : A Domain-based Approach for Supporting Mobility in Wide-Area Wireless Networks », *IEEE/ACM Transactions on Networking*, , 2002.

[CISCO] « Cisco Systems », *http ://cisco.com*.

[MAD] « Madwifi home page », *http ://madwifi.org*.

[NORT] « Nortel », *http ://nortel.com*.

[NYCW] « NYC wireless », *http ://nycwireless.net*.

[STRIX] « StrixSystems », *http ://strixsystem.com*.

[TIR 05] TIRUMALA A., QIN F., DUGAN J., FERGUSON J., GIBBS K., « Iperf-The TCP/UDP bandwidth measurement tool », 2005.